PROCEEDINGS

DMS 2004

The 10th International Conference on Distributed Multimedia Systems

Sponsored by

Knowledge Systems Institute, USA

Technical Program
September 8-10, 2004
Hotel Sofitel, San Francisco Bay, USA

Organized by

Knowledge Systems Institute

Foreword

Welcome to the Tenth International Conference on Distributed Multimedia Systems (DMS'2004), and thank you for participating in the conference.

The DMS conference series was established in 1994. Over the years, through continued efforts, the sponsors have worked diligently to improve the quality and appeal of this conference. Universities of Taiwan, USA, Hong Kong, Canada, and Japan have hosted this remarkable event. This year, DMS'2004 was held in the USA, for the third consecutive year.

We are entering a new era of visible and invisible computers. Flexible folded and rolled-up computer screen will become a part of our outfit. A large amount of information will be available to users through such screens at any time and place. All individuals, including elderly people and people with disabilities, will have an equal opportunity to use new technologies and make a contribution in creating the information resources.
Many other exciting things will be developed and available to us in the near future. However, a lot of problems should be solved to comfortably live in this era. What to do with the information explosion, when to replace TCP/IP protocol, how to reach a good level of security and fight against dirty activities on the Internet? These are a few problems to be mentioned.

The series of the DMS conferences can be considered as a part of our efforts to find solutions for such problems. DMS'2004 is a forum for theoreticians, end users, designers and developers of distributed multimedia systems in all fields. To facilitate a fruitful exchange of ideas centered around the core issues in the multimedia systems, six special-purpose workshops were organized. In total, 148 papers were submitted; 46% of them were accepted as regular papers, 28% as short papers, and 24% of the papers were rejected.

We are grateful to all authors and reviewers for their contribution to the technical program, making DMS'2004 a success. Many thanks to the program committee members, the organizers of the workshops, keynote speakers, and the supporting staff.

Finally, we would like to thank Professor Shi-Kuo Chang of University of Pittsburgh for his guidance, advice, and support in organizing DMS'2004.

Nikolay Mirenkov
and Maurizio Tucci
DMS'2004 Program Co-chairs

The 10th International Conference on Distributed Multimedia Systems (DMS'2004)

September 8-10, 2004
Hotel Sofitel, San Francisco Bay, USA

Organizers & Committee

Conference Chair

Shi-Kuo Chang, University of Pittsburgh, USA

Program Committee Co-Chairs

Nikolay N. Mirenkov, University of Aizu, Japan
Maurizio Tucci, University of Salerno, Italy

Workshops Coordinator

Timothy K. Shih, Tamkang University, Taiwan

Program Committee

Nadia Berthouze, University of Aizu, Japan
Carsten Griwodz, University of Oslo, Norway
Kai H. Chang, Auburn University, USA
Shih-Fu Chang, Columbia University, USA
Han-Chieh Chao, National Dong Hwa University, Taiwan
Ing-Ray Chen, Virginia Tech (VPI&SU), USA
Meng Chang Chen, Academia Sinica, Taiwan
Shu-Ching Chen, Florida International University, USA
Yonghee Choi, Seoul National Univ., Korea
Zied Choukair, ENST Bretagne, France
Fadi P. Deek, New Jersey Institute of Technology, USA
David H. C. Du, Univ. of Minnesota, USA

The 2004 International Workshop on Visual Languages and Computing (VLC'2004)

September 8-10, 2004
Hotel Sofitel, San Francisco Bay, USA

Organizers & Committee

General Co-Chairs

Alfonso Crdenas, University of California, Los Angeles, USA
Piero Mussio, University of Brescia, Italy

Program Chair

Kang Zhang, University of Texas at Dallas, USA

Program Committee

Tim Arndt, Cleveland State University, USA
Alberto Del Bimbo, Universita di Firenze, Italy
Marc H. Brown, Vendavo Inc., USA
Shi Kuo Chang, University of Pittsburgh, USA
Ralph Doerner, Fraunhofer AGC, Germany
Jing Dong, The University of Texas at Dallas, USA
George Furnas, University of Michigan, USA
Stephen Guest, Groupworks, USA
Erland Jungert, Swedish Defence Research Establishment, Sweden
Dieter Kranzlmller, University of Linz, Austria
Zenon Kulpa, Inst. of Fundamental Technological Research, Poland
Wei Lai, Swinburne University of Technology, Australia
Robert Laurini, University of Lyon, France
Stefano Levialdi, Universita di Roma, Italy
Kim Marriott, Monash University, Australia
Rym Mili, The University of Texas at Dallas, USA
Nick Mirenkov, University of Aizu, Japan
Marc Najork, Microsoft, USA
Joseph J. Pfeiffer, New Mexico State University, USA
David Stotts, University of North Carolina, USA
Genny Tortora, Universita' di Salerno, Italy
Guido Wirtz, Bamberg University, Germany

The Seventh International Conference on Visual Information Systems (VIS'2004)

September 8-10, 2004
Hotel Sofitel, San Francisco Bay, USA

Organizers & Committee

Conference Chair

Erland Jungert, Swedish Defence Research Establishment, Sweden

Program Co-Chairs

Jian Kang Wu, Institute for Infocomm Research, Singapore
Clement Leung, Victoria University, Australia

Program Committee

Josef Bigun, Halmstad Uni. and Chalmers Uni. of Technology, Sweden
S. K. Chang, U of Pittsburgh, USA
Shih Fu Chang, Columbia U, USA
Kent Cheung, City U of Hong Kong, HK
Tat-seng Chua, National U of Singapore, Singapore
Alberto Del Bimbo, U of Florence, Italy
David Forsyth, U of California, Berkeley, USA
Borko Fuhrt, Florida Atlantic U, USA
Theo Gevers, U of Amsterdam, the Netherlands
William Grosky, Wayne State U , USA
Horace Ip, City U of Hong Kong, Hong Kong
Erland Jungert, Swedish Defence Research Establishment, Sweden
Irwin King, Chinese U of Hong Kong, HK
T. Kunii, Hosei U, Japan
Inald Lagendijk, Delft U of Technology, the Netherlands
Robert Laurini, INSA, Lyon, France
Clement Leung, Victoria U, Melbourne, Australia

Michael Lew, U of Leiden, The Netherlands
Paul Lewis, U of Southampton, UK
Yanxi Liu, CMU, USA
Jinhua Ma, U of Aizu, Japan
Mario Nascimento, U of Alberta, Canada
Fernando Pereira, Institute of Telecommunications, Portugal
Masao Sakauchi, U of Tokyo, Japan
Hanan Samet, U of Maryland, USA
Simone Santini, Praja, San Diego, USA
Raimondo Schettini, DISCO, U of Milano-Bicocca
Timothy Shih, Tamkang U, Taiwan
Arnold Smeulders, U of Amsterdam, The Netherlands
H Lilian Tang, U of Surrey, UK
Ramond Wong, City U of Hong Kong, HK
Marcel Worring, U of Amsterdam, the Netherlands
Jian Kang Wu, Institute for Infocomm Research, Singapore

The 2004 International Workshop on Mobile Systems, E-commerce and Agent Technology (MSEAT'2004)

September 8-10, 2004
Hotel Sofitel, San Francisco Bay, California, USA

Organizers & Committee

General Co-Chairs
Jason C. Hung, Kuang Wu Institute of Technology, Taiwan
Kuei-Ping Shih, Tamkang University, Taiwan

Honorary Co-Chairs
Timothy K. Shih, Tamkang University, Taiwan
Chun-Chia Wang, Kuang Wu Institute of Technology, Taiwan

Program Committee Chair
Anthony Y. Chang, Oversea Chinese Inst. Of Tech., Taiwan

Publishing Chair
Chih-Yung Chang, Tamkang University, Taiwan

Program Committee :
Yuh-Shyan Chen, National Chung Cheng University, Taiwan
Tzung-Shi Chen, Chang Jung Christian University, Taiwan
Lawrence Y. Deng, San John's and San Mary's Inst. Of Tech., Taiwan
Ramachandra A. Dixit, Satyam Computer Services Ltd. India
Hui-huang Hsu, Tamkang University, Taiwan
Lun-Ping Hung, Kuang Wu Institute of Technology, Taiwan
Sheng-Tun Li, National Cheng Kung University, Taiwan
Kuan-Cheng Lin, Kuang Wu Institute of Technology, Taiwan
YingQun Liu, Tsinghua University, Beijing, China
Chikara Ohta, Kobe University, Japan
Magdy Saeb, Arab Academy for Science, Technology & Maritime Transport, Egypt
Prasan Kumar Sahoo, Van Nung Institute of Technology, Taiwan
Ronggong Song, National Research Council, Canada
Willy Susilo, University of Wollongong, Australia
Gustavo A. S. Torrellas, Instituto Mexicano del Petroleo, Mexico
Kuniaki Uehara, Kobe University, Japan
Theodora Varvarigou, National Technical University of Athens, Greece
Pallapa Venkataram, Indian Institute of Science, Indian
San-Yuan Wang, I-Shou University, Taiwan
Gwo-Jong Yu, Aletheia University, Taiwan
Chi Zhang, Florida International University, USA

Workshop Secretary
Chih-Ren Wei, Kuang Wu Institute of Technology, Taiwan

The 2004 International Workshop on Distance Education Technologies (DET'2004)

September 8-10, 2004
Hotel Sofitel, San Francisco Bay, USA

Organizers & Committee

General Co-Chairs

Lawrence Y. Deng, St. John's and St. Mary's Institute of Technology, Taiwan
Anthony Y. Chang, Overseas Chinese Institute of technology, Taiwan

Honorary Co-Chairs

Peter Tuen-Ho Yang, St. John's and St. Mary's Institute of Technology, Taiwan
Chih-Yung Chen, St. John's and St. Mary's Institute of Technology, Taiwan

Program Committee Co-Chairs

Chein-Shing Wang, St. John's and St. Mary's Institute of Technology, Taiwan
Ruei-Xi Chen, St. John's and St. Mary's Institute of Technology, Taiwan

Publishing Co-Chairs

Lin-Her Jeng, St. John's and St. Mary's Institute of Technology, Taiwan
Szu-Chien Luan, St. John's and St. Mary's Institute of Technology, Taiwan

Program Committee :

Rwei-Ching Chang, St. John's and St. Mary's Institute of Technology, Taiwan
Ko-Shung Chen, St. John's and St. Mary's Institute of Technology, Taiwan
K. Y. Chen, St. John's and St. Mary's Institute of Technology, Taiwan
Tzung-Shi Chen, Chang Jung Christian University, Taiwan
Yuh-Shyan Chen, National Chung Cheng University, Taiwan
Chuan-Feng Chiu, Panasonic Taiwan Laboratories Co. Ltd.
Jason C. Hung, Kuang Wu Institute of Technology, Taiwan
Lin-Her Jeng, St. John's and St. Mary's Institute of Technology, Taiwan
Qun Jin, The University Aizu, Japan
Y-K J. Jong, St. John's and St. Mary's Institute of Technology, Taiwan
Dong-Liang Lee, St. John's and St. Mary's Institute of Technology, Taiwan
Sheng-Tun Li, National Kaohsiung First Univ. of Tech. and Sci., Taiwan
T. C. Li, St. John's and St. Mary's Institute of Technology, Taiwan
Shieh-Shing Lin, St. John's and St. Mary's Institute of Technology, Taiwan
Hsu Chun-Liang, St. John's and St. Mary's Institute of Technology, Taiwan
Jian-Hua Ma, The University Josie, Japan
Jen-Kuei Peng, St. John's and St. Mary's Institute of Technology, Taiwan
Chun-Chia Wang, Kuang Wu Institute of Technology, Taiwan
Ching-Sheng Wang, Aletheia University, Taiwan

Workshop Secretary

Fan-Mei Lee, St. John's and St. Mary's Institute of Technology, Taiwan
Shih-Chuan Liu, St. John's and St. Mary's Institute of Technology, Taiwan

The 2004 International Conference on Chinese Language Computing (ICCLC'2004)

September 8-10, 2004
Hotel Sofitel, San Francisco Bay, USA

Organizers & Committee

Honorary Conference Chair

Benjamin K. T'sou, City University of Hong Kong, Hong Kong

General Conference Co-Chairs

Kai Chi, Humboldt State University, California, USA
Eileen V. Moy, Humboldt State University, California, USA

Program Co-Chairs

J. H. Lee, Pohang University of Science & Technology, China
T. S. Yao, North East University, China

K. J. Chen, Academic Sinica, Taiwan
A. C. Liu, Feng Chia University, Taiwan

Advisory Board

S. K. Chang, U of Pittsburgh, USA
Yaohan Chu, Washington D. C., USA
C. N. Liu, Century Development Corporation, Taiwan

K. T. Wang, Chunghwa Telecom, Taiwan
J. S. Ke, III, Taiwan
Ted C. Yang, Feng Chia University, Taiwan

Program Committee

K. Chang, Auburn University, USA
H. H. Chen, National Taiwan University, Taiwan
K. J. Chen, Academic Sinica, Taiwan
C. Chu, University of California, USA
C. C. Chu, Tung Hai University, Taiwan
C. R. Dow, Feng Chia University, Taiwan
J. T. Horng, National Central University, Taiwan
I. L. Huang, Tsing Hua University, Taiwan
W. Hwang, IBM, USA
K. L. Kwok, City University of New York, USA
L. Lam, Institute of Education, Hong Kong
H. J. Lee, S National Chiao Tung University, Taiwan
S. W. Lee, Korea University, Korea
M. K. Leong, Kent Ridge Digital Labs, Singapore
M. H. Li, Asian American Heritage Council, USA

R. Z. Lu, Shanghai Jiao Tung University, China
J. H. Ma, University of Aizu, Japan
M. Ma, Panasonic Technologies Inc., USA
C. Y. Suen, Concordia University, Canada
Timothy Shih, Tamkang U, Taiwan
H. C. Wang, National Tsing Hua University, Taiwan
J. F. Wang, National Cheng Kung University, Taiwan
S. Wang, Renmin University, China
X. W. Xu, Siemens, Erlangen
C. Yang, Chinese Univeristy of Hong Kong, HK
T. Yao, North East University, China
C. F. Yuan, Tsing Hua Univeristy, China
C. Zeng, Fukuoka Junior College of Technology, Taiwan

Table of Contents

Events and human-computer interactions

Virtual and Augmented Reality, Distributed Multimedia Documents

Multimedia streaming

Quality of Service

Network modeling and collaborative applications

Agent and retrieval systems

Editing and Compression of Multimedia Documents

VLC Papers

VIS Papers

Query refinement and benchmarking

Visual features for indexing and retrieval

MSEAT Papers

Mobile Agent and Networking

Scheduling and Control for Multimedia in Networks

Educational Technologies

Intelligence Technologies

ICCLC Papers

Visualization for RFID/Auto-ID Initiative

Tao Lin

SAP Research Center Palo Alto, SAP LLC

3475 Deer Creek Road

Palo Alto, CA 94304, USA

E-mail: tao.lin@sap.com

With the Radio Frequently Identification (RFID) mandates from retailers like Wal-Mart, Metro, Tesco, and Target, the U.S. Department of Defense, the US Food and Drug Administration of US, and Aerospace companies such as Airbus and Boeing, Auto-ID has received a lot of attentions in the past one year. This technology will have a great impact on many industries, such as consumer product, aerospace, transportation, pharmaceutical, defense, airline, and medical industries.

Started from MIT in 1998, Auto-ID initiative intends to provide an infrastructure for automatically tracking and tracing material flow globally. Auto-ID initiative has been transferred from MIT to EPCglobal and becomes an industry standard organization. Electronic Product Code (EPC) that provides a unique identification tagged to the tracked physical object is the fundamental concept for Auto-ID technology. RFID and 2D barcode are examples of Auto-ID. Based on EPC coding, an EPC Network is designed for monitoring material flow that can be potentially cross sites, cross organizations, and even cross countries. The tags, hardware and software systems used in an EPC Network will be from different vendors. Therefore the standards for tags and devices for reading and writing tags, and the interfaces between the systems that form the EPC Network are very critical. Currently EPCglobal is developing a set of standards, such as Tag Standard, Reader Protocol, Reader Management, Application Level Event (ALE) Interface, Object Naming Service (ONS), and EPC Information Service (EPCIS).

EPC Networks will increase the visibility of the today's material flow. Therefore it will improve the quality of the data acquisition to today's enterprise applications, such as Supply Chain Management, Product Life Cycle Management, and Asset Management. Furthermore, the logic of the business processes can be pushed down from today's enterprise applications to the software systems of the EPC Network to support local business decisions and new applications can be developed, such as counterfeit and goods recall.

EPC Network will not only just improve today's business processes but also provide opportunities for new applications. However EPC Network also requires great changes with the current IT infrastructure. It remains a big challenge to many organizations to justify business processes in order to take the advantages of EPC Network. EPC Network will create a lot of research issues, for example, the analysis and reporting of the very large volume of data been generated, the validation of the new business processes with Auto-ID data, the monitoring of the EPC Network. These issues require the assistance of the visualization.

In this presentation, we will introduce the technical concepts and issues for the development and deployment of EPC Network. The practical experience on the development and potential research issues with EPC Network will be discussed. Based on these discussions, we will analyze the usage and requirement of visualization technology for the development and deployment of EPC Network.

Reenvisioning Visual Information Systems: From Signal Analysis to Context-Aware Media Computing

Marc Davis

Assistant Professor
University of California at Berkeley
School of Information Management and Systems
Garage Cinema Research
314 South Hall
Berkeley, CA 94720-4600
EMAIL: marc@sims.berkeley.edu
WEB: http://garage.sims.berkeley.edu
TEL: (510) 643-2253 FAX: (510) 642-5814

ABSTRACT

Multimedia researchers have been trying to solve the problems of multimedia content analysis and media asset management for well over the past decade. It is time to acknowledge that this research has not delivered on its promises. Neither fully automatic signal-based analysis nor manual annotation of media content has provided a workable solution to content-based multimedia access. A new direction and new solutions are needed. In order for media to be as accessible as text, descriptions of their content and structure (i.e., metadata) must be created that are computationally and humanly usable. Unfortunately the low-level features that current algorithms can extract are not sufficient to meet the needs of how humans want to search for and use media content. This "semantic gap" is endemic to current multimedia information systems.

Most prior research in multimedia content analysis has suffered from assumptions that have hampered progress in the field and in making systems that can address users' needs. These assumptions are: 1) that media must be analyzed long after it has been captured (and therefore effectively removed from its context of creation and use); and 2) that multimedia content analysis must be fully automatic and avoid user involvement (and therefore miss out on the possibility of "human-in-the-loop" approaches to algorithm design). By shifting the paradigm of image and video processing from an a-contextual, fully automatic model to a model that leverages the spatio-temporal-social context of media creation and use as well as interaction among devices and people (especially by taking advantage of new programmable networked mobile media capture devices), we have the promise of solving long-standing challenges in visual information systems.

A CHRONOBOT FOR TIME AND KNOWLEDGE EXCHANGE

Shi-Kuo Chang
Department of Computer Science, University of Pittsburgh
Pittsburgh, PA 15260 USA (chang@cs.pitt.edu)

Abstract

The chronobot is a device for time and knowledge exchange. The concept of the Chronobot first appeared in a science fiction short story written by me some twenty years ago. Recently the Industry Technology Research Institute (ITRI) and Institute for Information Industry (III), two leading research institutes in Taiwan, invited me to lead a pioneering project to put my ideas into practice to build a realistic device. The Chronobot was thus conceived. To put it in simple terms the Chronbot allows a group of people to exchange time and knowledge. It is a platform for time and knowledge exchange. This paper describes the concept of the Chronobot, its basic mechanism for time/knowledge exchange and its application to e-learning. The application scenarios, a Chronobot prototype and research issues are also discussed.

1. Introduction

The chronobot is a device for storing and borrowing time. Using the chronobot one can borrow time from someone else and/or return time to the same person. It is a very convenient device for managing time.

The underlying premise of the chronobot is that there is a way to exchange time and knowledge. For example one spends time to acquire knowledge and later uses this knowledge to save time. A group of people can also find some means to exchange time and knowledge. Thus **the chronobot is a device to facilitate the exchange and management of time and knowledge**.

A natural application domain for the chronobot is e-learning and distance education, although we can consider many other interesting application domains for the chronobot. Indeed whenever we need to manage time and knowledge, we can make good use of the chronobot.

2. Application Scenarios

In this section we describe two examples of the chronobot for e-learning and distance education applications.

John is a teenager. His parents recently bought him a chronobot. When John wakes up in the morning, he has breakfast and then takes the bus to the school. On the bus John has some free time. So his chronobot says to John: "You know you have to write a big report on the eating habits of dinosaurs tonight. Why don't you spend some time now to collect some information? There is a rock concert at Point Park tonight. If you get the report done early, maybe your mom will let you go to the rock concert!" John is excited about the rock concert and really wants to go, so he follows the chronobot's advice and puts in some effort to collect and organize information. After the second class period John again has some free time. Again, following the chronobot's advice, John puts in some time to get more pieces of information and label them according to the chronobot's suggestions.

But John's efforts later pay off. After John finishes school he turns to the chronobot who to his delight has already fused the knowledge together to form a rough draft of the report. John only has to do some editing and in less than twenty minutes the report is

completed! But some critical facts need to be checked by John's teacher. So John goes to chronobot's virtual classroom to interact with his teacher Ms. Newman. In ten minutes, John gets all the answers from Ms. Newman. (Unknown to John, it is actually Ms. Newman's chronobot serving as a surrogate to answer his questions, but this does not matter.) It is only six thirty pm, and John proudly shows off the finished report to his mother, who approves John's request for an outing. So John happily goes to the rock concert with his buddies. If John does not have the chronobot, he would be stuck with the report writing task the entire evening and misses the rock concert!

The above example pretty much explains the usefulness of the chronobot. As another example, for a professional media artist George, the chronobot serves the same function of timely knowledge gathering. But it can be even more useful because unlike the teenager who is required to do his homework by himself, George has no such constraints and can rely upon the support from his coworkers. However there is no such thing as a free lunch. In order to ask his coworkers Suzie and Bill to share his workload, George has to put in efforts either earlier (i.e. in the past time) or later (i.e. in the future time) to help them. But at the present time when George needs help, Suzie and Bill will put in their efforts to help George. Through their chronobots, George, Suzie and Bill interact in the virtual classroom. The chronobots in turn through their masters' conversation figure out what they are expected to do. They work independently to retrieve the knowledge previously organized by their respective masters, and then work together to fuse the knowledge into a format useful for George to put in the finishing touches.

3. Characteristics of the Chronobot

Based upon the above description of the application scenarios we can envision some of the general characteristics of the chronobot:

1. The chronobot is a **time manager**. But it is not an ordinary time manager. It can be used to **manage not only one's own time, but also other people's time through time/knowledge exchange protocols**. The exchange protocol is very important, because the unique concept of the chronobot is that **time and knowledge are exchangeable**. The chronobot can manage not only the present, but also the past and the future through suitable time/knowledge exchange protocols.

2. The chronobot is also a **knowledge manager**. It can be used to store knowledge, organize knowledge, retrieve knowledge and **perform information fusion to produce new knowledge**. Information fusion is the key concept. Without information fusion, the chronobot will not be as effective in managing knowledge and saving time.

3. For e-learning and distance education applications, the chronobot offers a versatile **virtual classroom** that combines the functions of chat room, white board, multimedia display device, time manager, knowledge manager and time/knowledge exchanger.

4. Utilizing the time manager, the knowledge manager and the virtual classroom, the **chronobot can interactively provide timely knowledge to the end user**. This is the main characteristic of the chronobot. We often say *time is money*. We also say *knowledge is power*. If we can exchange time and knowledge, then these four entities - *time, money, knowledge and power – all become interchangeable*!

4. From Experiences to Knowledge

Present-day distance education systems are still too rigid and do not lend themselves easily to peer-to-peer learning. Since the Chronobot is designed for time and knowledge exchange, the participants are naturally encouraged to exchange what they have learned. As illustrated in Figure 1, the Virtual Classroom is where such exchanges actually take place. Thus the Chronobot equipped with the Virtual Classroom may circumvent the difficulty in supporting peer-to-peer e-learning.
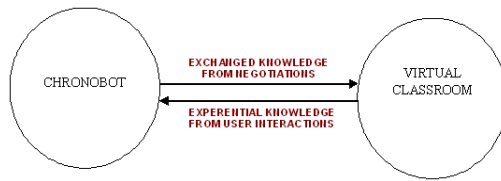
Figure 1. Chronobot and the Virtual Classroom.

Figure 1 illustrates the exchange of knowledge in e-learning. In Figure 1, after a successful negotiation, the Chronobot will send the exchanged knowledge to the Virtual Classroom, where the actual transaction takes place, i.e., one participant will transfer the knowledge to another. On the other hand, the experiential knowledge from user interactions in the Virtual Classroom will be provided to the Chronobot so that the Chronobot has better knowledge about user characteristics and user preferences. In other words, the Chronobot learns more about the User Profile. The VC can also be used to visualize the negotiation process and current negotiation status of the Chronobot.

5. The User Profile

The User Profile *UP* is an abstraction of user's preferences and characteristics. When a user first registers, the user is asked to enter information about self, as well as areas of expertise and so on. The User Profile manager provides a HTML front-end using which new users can register themselves with the system. During the registration, the User Profile manager collects important information from the users such as the UserId, Name, Address, Credit Card details, Areas of Experience, Skill Set, whether the user is an Expert or not, the hourly rate, e-mail address etc. The rationale behind having the credit card information is that if the user defaults, then his credit card is billed depending upon the number of hours defaulted.

6. The Relational Index

Figure 2 illustrates the User Profile and the Relational Index, two important data structures for the Chronobot/VC system.
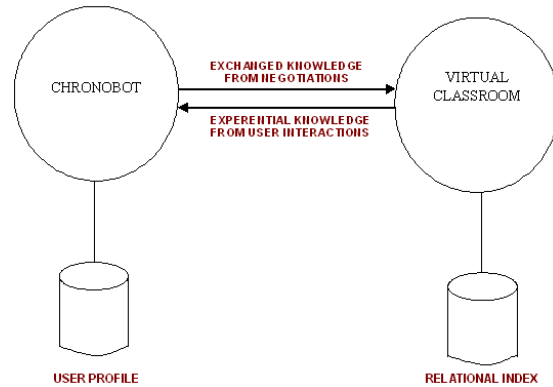
Figure 2. User Profile and Relational Index.

The experiences accumulated in the Virtual Classroom are the most important assets for further e-learning. In practice the experiences are the transcripts of the Virtual Classroom sessions. These transcripts are XML documents. In fact, everything that flows to the Chronobot/Virtual Classroom system is some form of XML document.

The Relational Index *RI* is built to support easy access of the learning experiences. Basically, the session transcripts (XML documents) are stored in an experience-base. The Relational Index is then constructed. It relates learning experiences to user preferences in the User Profile.

7. Determination of the Exchange Rate

The protocol that governs the information exchange is explained in [1]. In the negotiation for time exchange, if the two agents feel their times have different value, it will become necessary to negotiate the exchange rate among the Chronobot agents. In what follows we suggest a mechanism for the determination of the exchange rate.

We would like to express our terminology as follows, when we refer to agents, we refer to Chronobot agents. For two agents agent x and agent y, each agent is characterized by attributes $(x_1, ..., x_n)$, and $(y_1, ..., y_n)$. For the two corresponding attributes x_i and y_i, the information distance measure is denoted by

$d_i (x_i, y_i)$, where d_i is between 0 and 1 (a metric).

The exchange rate between agents x and y, is denoted as follows.

Exchange (agent x, agent y) =

$$e^{(1 / \sum C_{ji} * d(x_{ji}, y_{ji}))}$$

where the summation is over all the terms $C_{ji} * d(x_{ji}, y_{ji})$, and C_{ji} is a scaling constant.

We now illustrate the concept by presenting an example. Let us assume that the two agents' primary skill matches. Therefore $C_1 d_1 (x_1, y_1) = 0$. If the primary skill does not match, $C_1 d_1 (x_1, y_1)$ becomes a big number. For instance, C_1 is 10,000 and d_1 is between 0 and 1, in this case close to 1. Then $C_1 d_1 (x_1, y_1)$ is close to 100,000 and the exchange rate is close to 1. No need to continue.

The two agent's familiarity with subject area also is comparable, so $c_2 d_2 (x_2, y_2) = 0$. If the familiarly does not match, then $C_2 d_2 (x_2, y_2)$ becomes a big number. For instance, C_2 is 1,000 and d_2 is between 0 and 1, in this case close to 1. Then $C_2 d_2 (x_2, y_2)$ is close to 1,000 and the exchange rate is close to 1. No need to continue either.

Finally, the two agents differ in secondary skill. Therefore $C_3 d_3 (x_3, y_3)$ is small and we have an exchange rate that reflects the difference in the two agents' secondary skill. Notice the index function takes care of the re-arrangement of the relative importance of the n attributes. The constants C_j are also important. They take care of the relative scaling of the various attributes. Figure 3 illustrates the determination of the exchange rate as a dynamic process of comparing different attributes to identify the ones that really matter.

8. Just-in-Time e-Learning

A prototype of the Chronobot was implemented [2]. We are now developing a methodology of query morphing for information fusion and integration from the experiences utilizing an ontological knowledge base. A model to fuse information from different sources to dynamically construct the User Profile is also being formulated. Further experimental studies of the Chronobot are currently under way at Taiwan's Industry Technologies Research Institute (ITRI) and Institute for Information Industry (III). One of our goals is to apply the Chronobot to **just-in-time e-learning** to quickly amass learning resources in order to train the unemployed and socially disadvantaged so that those people can acquire new skills quickly to qualify them for certain jobs.
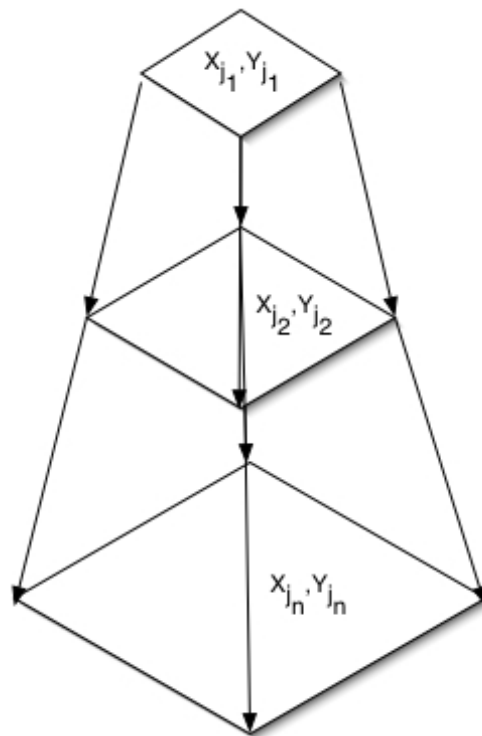
Figure 3. Determination of exchange rate.

References:

[1] Shi-Kuo Chang and Ganesh Santhanakrishnan, "Chronobot: A Time and Knowledge Exchange System for e-Learning and Distance Education," Proceedings of 2004 Int'l Conference on Distributed Multimedia Systems, September 8-10, Hotel Sofitel, San Francisco Bay, 2004, 443-450.

[2] Shi-Kuo Chang, Anu Kapoor, Ganesh Santhanakrishnan and Chirag Vaidya, "The Design and Prototyping of the Chronobot System for Time and Knowledge Exchange," Technical Report, University of Pittsburgh, Pittsburgh, September 2004.

The Dao of Unihan

John Jenkins
Sr. Administrative Director
The Unicode Consortium
650-693-3921
Email: magda@unicode.org

ABSTRACT

Over half of the characters in the Unicode Standard are ideographs. This ideographic repertoire, termed Unihan, is intended to provide complete coverage for all the characters in current or past use in all varieties of Chinese, Japanese, Korean, and Vietnamese.

In this talk, we will give an overview of the structure of the current repertoire of Unihan and its organization. We will discuss some practical implementation issues and how to deal with them, as well as the process by which new ideographs are added to the standard and what the future holds.

We will also provide an overview of the Unihan database. This is a large body of normative and informative data, which is maintained by the Unicode Consortium and included among the data files, which are a part of each release of the standard. We will discuss the nature of the data in the database, how it can be used, and how it can be improved.

Proceedings

The 10th International Conference on Distributed Multimedia Systems (DMS'2004)

Co-Editors

Nikolay N. Mirenkov, University of Aizu, Japan
Maurizio Tucci, University of Salerno, Italy

SCORM-Compliant Reader on Pocket PC

Timothy K. Shih, Nigel H. Lin, Hsuan-Pu Chang, Yuan-Kai Chiu, Huan-Chi Huang
Department of Computer Science and Information Engineering
Tamkang University
nigel@mail.topwise.com

Abstract

Mobile computing devices are becoming extremely popular. Mobile telephones, Personal Digital Assistants (PDAs), and hand-held computers are developing rapidly and most of them have the domination of size which makes them the expected platform for mobile learning. In this paper, we concentrate on the issues of transferring the current PC based Sharable Content Object Reference Model (SCORM) to Pocket PC based and demonstrate the implementation of the mobile learning tool , Adaptive Pocket SCORM Reader.

So many researches have proposed how distance education can be realized on pocket devices. We will also introduce the prototype version of Pocket SCORM Run-Time Environment (RTE). Proposed Pocket SCORM Architecture is able to operate, even when the mobile device is physically disconnected from the network, without losing any students' learning record. Collected records will then be sent back to Learning Management System (LMS), runs at the sever side, after the device is back on on-line. On the client side, mobile device has our Adaptive Pocket SCORM Reader is able to load SCORM compatible courseware.

Keywords: Pocket PC, PDA, SCORM, Distance Education,

1. Introduction

Pocket devices are very suitable for mobile or distance education because of it's carried around, small size and light weight. Although Pocket devices have been improved in both computing power and memory storage recently, they are still with lots of limitation compared with laptop or desktop computers. In short, they don't provide a robust learning experience. Therefore, the platform running on laptop or desktop computers can't be directly transferred into Pocket PC devices. The courseware which is designed for laptop or desktop computers should be modified in some ways in order to be suitable for Pocket devices.

The following points are some differences between the Pocket PC devices and laptop or desktop computers.

- *Displaying Size*

Since display size of the PDAs is much smaller than that of desktop computers, the amount of information that is visible at a time is dramatically reduced. As a result, the number of sub-pages must be larger, or the content must be reorganized or modified. There are many researches that conquer the problem by summarizing content [1] or transform the layout in whole new style [2]. However, most of them are not suitable on educational content. Therefore, the content of courses designed in desktop and running on laptop or desktop computers can't be directly transferred into Pocket PC devices.

- *Connection Mechanism*

Normally, E-Learners will be on-line when they are using laptop or desktop computers. Since E-Learners would stay at the same location longer when using their laptop or desktop computers than using a Pocket PC devices. As a result, they will be kept connected with the LMS, which is running on the server, all the time. This is different from Pocket PC devices because these devices are designed to be portable. This means E-Learners don't stay at the same location for a long period of time. Pocket PCs are carried around to any place where is sometimes without network coverage. Therefore, we need to find out a new connection mechanism which will allow E-Learners to continue their learning even they are disconnected from the internet.

- *Courseware Import and Export*

As we mentioned previously, Pocket PC devices could be disconnected sometimes; however, we hope E-Learners to be able to learn when the network is not existed. In order to enable this functionality, we need to seek a way to allow the courseware to be temporary stored in the hand carried devices. In such way, E-Learners will still be able to learn the course content when they are not on-line. However, there is obstacle to prevent from temporary storing the courseware on the pocket devices because of the storage limitation. This could be a serious problem when the platform is practically applied on the pocket devices.

- *Learning Records Buffering*

For a distance education standard such as SCORM, it requires the LMS to be able to keep tracking on learners' learning records so these records will be used to determine learners' progress or maybe transfer to other LMS where learner continue his education. As a result, it is important for a SCORM compatible platform to be able to store learners' learning records. As a Pocket SCORM platform, we also need to enable our system to capture learners'

learning data. In addition, the recorded data also need to be temporary stored on pocket devices if learner is not connected to the internet. After the learner is back on-line, the system need to transfer these data back to LMS on the server end.

In this paper, we proposed a platform which is known as Pocket SCORM. The proposed platform is focused on the pocket devices. There is also a LMS which will be also SCORM compatible to support the Pocket SCORM platform and the major data store. The rest of this paper is organized as following. In Section 2, we will introduce some related works which are related to SCORM and systems on the pocket device. Section 3 will show the architecture of our Pocket SCORM architecture. In Section 4, we will introduce the Pocket SCORM RTE which has been developed in our lab. Before the final conclusion and the future works, we demonstrate the implementation of Adaptive Pocket SCORM Reader which is part of our Pocket SCORM architecture. It is capable to load SCORM compatible courseware, and adjust the course content to adapt to the features of a pocket device.

2. Related Works

Distance education enables E-Learners to learn without the restrictions of both time and space. There are many web-based courseware have been developed to allow learners to browse course content via a browser. SCORM (Sharable Content Object Reference Model) is a standard which is proposed by ADL (Advanced Distributed Learning) [3]. SCORM is aiming at the standardization of computer based teaching components. There are some papers related to SCORM have been published. There are some advantages such as portability of learning content, the standardized communication interface between the LMS and WBTs, and supporting the reusability of learning object. However, there are some problems as well such as the market value of SCOs, the process of producing WBTs on the basis of different SCO providers, the maintenance of SCOs and WBTs, and the quality of WBTs based on SCOs of different providers. There is a review which discussed these issues can be found in [4]. There is another paper which discussed the Implementation of Content Repository Management System is referenced in [5]. In "Using SOAP and .NET web service to build SCORM RTE and LMS" [6], the XML Web Service based LMS and RTE was introduced. There was another system developed for automating the generation of SCORM-Based multimedia product training manuals was introduced in [7].

Personal Digital Assistants (PDAs) have become new learning tools for distance education. Portability of the PDAs was welcomed by students, and advantageous was advantageous, limitations such as the small screen size, navigation difficulties, and slow and error-prone methods for entering text, made it difficult to read and interact with document on the PDA [8]. Some students' experiences for reading course materials by PDAs were experimented. There are also some applications developed for educational purposes. TekPAC (Technical Electronic Knowledge Personal Assistant Capsule) was introduced in [9]. TekPAC was developed for providing access to readily available electronic information, allowing the user to perform tasks at locations with all schematics, photos, videos and BKMs readily available, and integrating key interventions to raise performance of target audience. PDAs have also been adopted in medical field as a tool for education. There were some PDA Projects at Virginia Commonwealth University being introduced in [10].

3. Pocket SCORM Architecture

In this section, we will introduce our Pocket SCORM Architecture which is shown as in figure 3.1. In proposed architecture, we pointed out two types of connection for a Pocket PC to connect with LMS Server. One type is Pocket PC is directly connected to the server through wired or wireless network to the internet while the other is Pocket PC connects to the server via PC to the internet while Pocket PC is synchronizing with the PC. With wireless technology, some Pocket PCs have built-in wireless LAN card, and some of them have an expansion slot which can be add-in a wired or wireless LAN card. For those wireless LAN cards normally support the IEEE802.11B network protocol. There will be an access point also known as AP to receive the wireless signals. The wireless signals from the wireless LAN card will be transferred to AP first, and if that particular AP is connected to the internet, then signals can be redirected to the destination on the internet. Therefore, if a Pocket PC with the wireless capability, it can connect to the SCORM LMS server directly. For those Pocket PCs which have neither built-in wireless LAN card nor expansion slots for add-in wired or wireless LAN card will connect to the internet through an on-line PC. There are three major components within the Pocket SCORM Architecture. In the following three sub-sections, we will show more details of each of them.
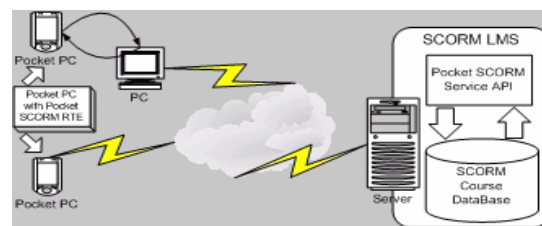
Figure 3.1: Pocket SCORM Architecture

3.1 Pocket SCORM Run-Time Environment

There are six major components which are included in Pocket SCORM Run-Time Environment. All these components work together to form the whole Pocket SCORM Run-Time Environment. They are listed as below:
- *Communication Agent*

The Communication Agent is used when the pocket devices try to communicate with the SCORM LMS Server. When E-Learners try to download the SCORM based courseware from the LMS Server, it will receive the packed courseware and pass it to Data UnPacking Agent. If there are some learning records need to be sent back to the LMS server, the Communication Agent will connect to the LMS Server and send the packed learning records back to the server. If the Pocket PC is connected directly to the LMS server, the Communication Agent will communicate directly with the SCORM XML Web Service Agent. Otherwise, it will communicate with the Synchronization Agent instead of SCORM XML Web Service Agent. We considered using Simple Object Access Protocol (SOAP) [11] to be our transmission protocol to make our services of server side more extendable.

- *Data Packing Agent*

To reduce network load, we implemented the Data Packing Agent. This agent will pack the data before sending it to Communication Agent. For security reason, we also consider to add security protection function for this agent because normally it will pack E-Learner's learning records and some of his privacy information. Therefore, we might need some protection for the E-Learner's privacy.

- *Data UnPacking Agent*

Normally, the courseware will be packed as a Package Interchange File (PIF) before it is downloaded by Communication Agent from LMS Server. Therefore, we need Data UnPacking Agent to unpack the PIF file after it has been downloaded from the LMS server and then restore the original courseware from the PIF.

- *Learning Agent*

When E-Learner starts to study the courseware which has been downloaded from the LMS Server, the Learning Agent will start to keep tracking on the learner's learning records. Since the learner might not be on-line as we pointed out this issue at the beginning of this paper, our Learning Agent will store those learning records in the SCORM PDA Database instead of sending them directly back to LMS Server. After the learner is able to connect to the LMS Server, those temporary stored learning records will be sent back to the Server. As a result, we will not lose learners' learning records even learners are off-line.

- *Pocket SCORM Reader*

Due to the hardware restriction, Pocket PC is small in size. Therefore, normal web-based course content is not suitable for Learners who use a Pocket PC because they might need to use a stylus to control the scrollbars inside the Pocket Browser. This will make learner hard to read the materials on the Pocket PCs. To overcome this drawback, we have designed a Pocket SCORM Reader with the function called "Reflow". The "Reflow" functions will be performed by our Pocket SCORM Reader Data Presentation Module. Normally, a Pocket PC will provide a vertical scroll button which allows user to control the

vertical scrollbar of an application. Our "Reflow" function will adjust the content to make it fit in the width of the display width on our reader. Since the content will fit in the reader, learners need only to control the vertical scrollbar instead of controlling two scrollbars. Our Pocket SCORM Reader allows learners to use only one hand to view the course content. This function extends the flexibility of learning with a Pocket PC. The details of Pocket SCORM Reader will be described in Section 5.

- *SCORM PDA Database*

For the downloaded courseware or learners' learning records, there is a temporary data store which is called SCORM PDA Database. After courseware has been downloaded, the Data UnPacking Agent will unpack the course content and store it into SCORM PDA Database. When learners are studying the courseware, our SCORM PDA Reader will load the course content from the SCORM PDA Database. So learners don't have to be on-line for learning courseware because it has been previously stored in their Pocket PCs. During the learning period, the Learning Agent will track and store the learning records into this data store as well. After the learning records have been transferred to the LMS Server, those records will also be removed from this database as well as unused course content to save the precious memory space of a Pocket PC.

3.2 PC Dock

PC Dock is a layer between Pocket SCORM RTE and SCORM LMS Server. If the Pocket PC without the ability to be on-line, it will require a PC Dock to be able to connect to the LMS server. There is a Synchronization Agent inside the PC Dock. The Synchronization Agent will perform the data transmission job between Communication Agent on the Pocket SCORM RTE and XML Web Service on the SCORM LMS Server. The transmission protocol will be focus on SOAP as well. PC Dock is aimed at providing the internet communication ability for those Pocket PCs without network ability.

3.3 SCORM LMS Server

There are two major components involved in SCORM LMS Server. One is the SCORM Data Repository, and the other is Pocket SCORM Service API. SCORM LMS Server provides distance education courseware and all the courseware is following SCORM Data Model. The learners' information is also contained in the SCORM LMS Server. When a learner connects to the LMS Server, he or she needs to first logon the system before he or she can access any course materials. The uses of these two major components are stated as below:

- *SCORM Data Repository*

The SCORM Data Repository stores all the course materials which follow SCORM Data Object Model. These course materials can not only provide for pocket device users but also desktop or laptop computer users using

browser based application to access the courseware through LMS Server. In the paper, we mainly focused on the Pocket PC devices. Therefore, we only care about how SCORM Data Repository interacts with our Pocket SCORM Service API. Nevertheless, this data repository should also supports any SCORM based API. Furthermore, learners' learning records are also stored in this data repository. These SCORM based learning records should also be able to interact by using either Pocket SCORM Service API or any SCORM based API.

- *Pocket SCORM Service API*

Ideally, Pocket SCORM Service APIs should be same as normal SCORM based APIs. However, due to some limitations of Pocket PCs such as computing power and memory storage, we need to modify the original defined guidelines of SCORM based APIs proposed by ADL and make them accommodate some features which are only found on Pocket PCs. Nevertheless, we hope defined Pocket SCORM Service APIs can be widely applied by other applications. We tried to build Pocket SCORM Server APIs by adopting XML Web Service [12] technology. XML Web Service takes SOAP as its transmission protocol. One of the advantages of using XML Web Service to become our APIs is the accessibility. Since SOAP is loosely coupled protocol by using XML wrapped envelope to invoke APIs, this vantage makes XML Service APIs can be accessed by any platform which follows SOAP protocol to acquire the service. As a result, we hope to implement our Pocket SCORM Service APIs as XML Web Services.

In this section, we have shown the overview of our planed architecture of our Pocket SCORM and components comprised in this architecture. The relationship between each component is shown as figure 3.2.
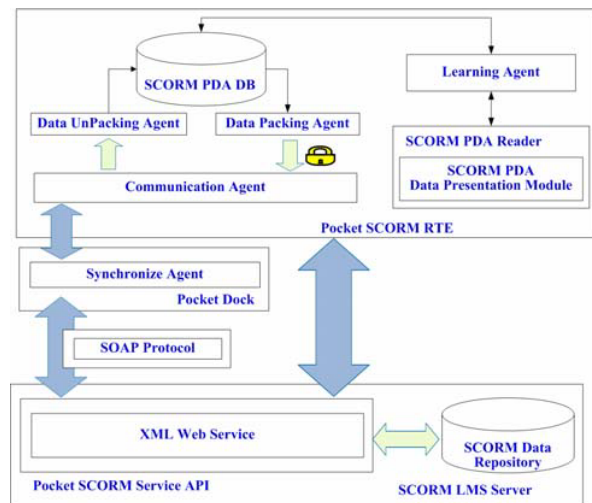
Figure 3.2: Relationship of Components within Pocket SCORM Architecture

4. Pocket SCORM RTE Implementation and demonstration of Adaptive Pocket SCORM Reader

Up to present, we have completed some portion of the whole Pocket SCORM Architecture. There were some components which comprises the Pocket SCORM Run-Time Environment. Some user interfaces and functions will be introduced in this section includes the implementation of our Adaptive Pocket SCORM Reader.

In figure 4.1, there are two user interfaces. On the left hand side, it shows the UI when student try to logon to Adaptive Pocket SCORM Reader. Because we need to track on learners' learning records, the learner needs to provide his identity before studying the course materials. There is also an important issue need to be taken care. We need to make sure is the same user who is studying the courseware.

Figure 4.1: Logon Interface and Course List View

On the right hand side of figure 4.1, it shows a list of imsmanifest files which represent the each different course structure. In SCORM, imsmanifest file contents the course organization information. Our Adaptive Pocket SCORM Reader will list available courseware which has been downloaded into the SCORM PDA Database. Learner can choose one of them and start his learning.

In SCORM there are two major structures of a course were defined. One of them is knowledge based course structure, which is shown as on the left hand side of figure 4.2, and the other one is linear based structure, which is shown as on the right hand side of figure 4.2. Our Adaptive Pocket SCORM Reader is capable to display the course structure according to defined imsmanifest file which learner chose to load.

Adaptive Pocket SCORM Reader is able to load SCORM compatible courseware. There are two display mode provided by our proposed reader.

Figure 4.2: Knowledge based and Linear based Structure

As shown in the left hand side of figure 4.3, the Normal mode displays the course content according to its original design. The original design of the course content is too large to fit in the small display. As a result, learners might feel inconvenient during browsing the courseware because it requires learners operate two scroll bars in order to view the whole page content. Alternatively, our reader provides another display mode which is called MINE mode.

Figure4.3: Display Mode

By using MINE mode to display the course content, the layout of the course content will be reflowed to fit in the display width of the reader. MINE mode enables
learners to use only one hand to operate their pocket device and view the whole page content.

Another major feature of our Adaptive Pocket SCORM Reader is a learner is allowed to make notes while he or she is reading the learning material. The explanatory notes then will be saved. As shown in figure 4.4, these notes written by the learner will be recorded in during MINE display mode.

Figure 4.4: Write personal notes down

5. Conclusion and Future Works

Distance Education provides alternative way of learning. People are benefited from the flexibility of distance learning since it has broken the time and space limitation. To extend this flexibility and to make E-Learners able to learn at any location, we have shown our proposed Pocket SCORM architecture in this paper. The Implemented Adaptive Pocket SCORM Reader was also introduced. We hope our proposed architecture can make E-Learners to learn easier by using a pocket device which can be carried to anywhere and enable E-Learners to learn anytime. The current completed implementation has led us toward to this goal.

The whole Pocket SCORM architecture has not yet completed. The future works include completing the PC Dock and SCORM LMS Server which supports Pocket SCORM Service API. The cooperation with our SCORM LMS Server with other LMS Servers is another issue of our future development. We also hope that we could conduct some real-world experiment by asking students to participate a SCORM based course by using our system under Pocket SCORM Architecture. The actual feedback from learners can contribute the ideas of future modification of our proposed structure.

6. References

[1] Yonghyun Hwang, Jihong Kim, and Eunkyong Seo "Structure-Aware Web Transcoding for Mobile Devices",Internt Internet Computing, IEEE, (Sept.-Oct.2003)

[2] Staffan Björk, Lars Erik Holmquist, Johan Redström, Ivan Bretan, Rolf Danielsson, Jussi Karlgren and Kristofer Franzén, "WEST: A Web Browser for Small Terminals", Proceedings of the 12th annual ACM symposium on User interface software and technology (November 1999)

[3] Advanced Distributed Learning (ADL) (2003), http://www.adlnet.org.

[4] Oliver bohl, Dr. Jorg Schellhase, Ruth Sengler, and Prof. Dr. Udo Winand (2002), "The Sharable Content Object Reference Model (SCORM) – A Critical Review", Proceedings of the International Conference on Computers in Education (ICCE'02)

[5]   Jin-Tan David Yang, and Chun-Yen Tsai (2003), "An Implementation of SCORM-compliant Learning Content Management System – Content Repository Management System", Proceedings of the The 3rd IEEE International Conference on Advanced Learning Technologies (ICALT'03)

[6] H. Lin, Hun-Hui Hsu, and Ching-Tang Hsieh (2003), "Using SOAP and .NET Web Service to Build SCORM RTE and LMS", Proceedings of the17th International Conference on Advanced Information Networking and Applications (AINA'03)

[7] Peiya Liu, Liang H. Hsu, and Amit Chakraborty (2002), "Towards Automating the Generation of SCORM-Based Multimedia Product Training Manuals", Proceedings of 2002 IEEE International Conference Multimedia and Expo, 2002. (ICME '02)

[8] J. Waycott, and A. Kukulska-Hulme (2003), "Students' experiences with PDAs for reading course materials" Personal and Ubiquitous Computing Volume 7, Issue 1 (May 2003), ISSN:1617-4909

[9]   Catherine Weissenborn, and Frank J. Sanchez (2001), "TekPAC (Technical Eletronic Knowledge Personal Assistant Capsule)", 2001 IEEE International Semiconductor Manufacturing Symposium

[10] Kay Sommers, Jane Hesler, and Jim Bostick (2001), "Little Guys Make a Big Splash: PDA Projects at Virginia Commonwealth University", Proceedings of the 29th annual ACM SIGUCCS conference on User services (SIGUCCS' 01)

[11]   World Wide Web Consortium (W3C) (2003), http://www.w3.org/TR/soap12-part0

[12]   World Wide Web Consortium (W3C) (2003), http://www.w3.org/TR/ws-arch

Special Tags for supporting SCORM-Compliant learning Environment

Te-Hua Wang, Yuan-Kai Chiu, Timothy K. Shih, Wen-Chih Chang, Nigel H. Lin
and Chia-Tong Tang
MINE Laboratory
Computer Science and Information Engineering Department
Tamkang University, Taiwan, R.O.C.
{ fatty,u6191033}@cs.tku.edu.tw

Abstract

With the improvement of Internet, the learning activities of the learners have become more and more various and plentiful. Accordingly, the learning behavior in the cyberspace may be different from the one in traditional learning environment. In this paper, we develop an integrated authoring system based on the SCORM(Shareable Content Object Reference Model) specification, and provide some special tags that can be embedded within the learning contents for the learning activities. The special tags can be categorized to four types by different properties of pedagogies. The categorized tags can be taken as the inputs of learning profiles, or as the condition rules of the learning status. Additionally, our proposed authoring system also supports the designing of the various learning materials and the fulfilling with SCORM simple sequencing specification. With the functionalities, the editing and fulfilling the metadata attributes of the learning contents will be easy to made by using the drag and drop operations.

Keywords

Distance learning, SCORM, authoring System, learning activity, learning management system, pedagogic tags.

1. Introduction

For the scope of distance learning, the learning space is based on the network architecture, that is, the contents of learning materials will be more various and complicated than traditional learning. Hence, authoring on the various learning contents will be an important issue for the instructors. Besides, if we take into considerations about the learning efficiency from the learner's perspective, the behavior of navigation will play an important role.

Besides, in most present distance learning environments, the course contents have their varieties and there are no standards for the authoring. In such situations, the luxuriance of course contents was enriched, but relatively, the sharability was accordingly reduced. Even though in the same LMS (Learning Management System), the environment that offers the capabilities for the learners to manipulate the learning activities, the designed course contents cannot reference each other as their sources.

Consequently, many learning contents or course concepts need to be re-built and take necessary disk space for the repository. To solve the issue, many organizations and academic contribute many models to achieve the reusability and sharability of course contents, such as

ULF[2], SOAP[3] and so on.

SCORM is a well-known e-learning standard proposed by ADL [1, 4] in 1997 with the latest released version 1.3. The main purposes of the model focus on the durability, interoperability, accessibility and reusability of learning resources among the distributed learning environments. SCORM is combined with IEEE Metadata [7] Dictionary, IMS [5, 8] Content Packaging, IMS Metadata XML Binding, AICC [6] Content Structure, AICC Data Model, AICC Launch and Communication APIs.

Nowadays, some famous academics and organizations endeavor to build the authoring systems for the SCORM-compatible learning contents. Consequently, we believe that an efficient authoring system based on the user-friendly interface will reduce the cost and time spent while building such learning materials.

By adopting the important concepts specified by SCORM, in our developed system, we want to achieve the costs saving of course contents designing and sharing contents with those who may design the same and duplicate learning contents.

Furthermore, from the perspective of learners within the distance learning environment, the learning activities will be done via the web browsers and some specialized LMSs. However, some pedagogic statistical researches show that, the passive learning materials are important for the learners, but some pedagogical and additional information needs to be involved while learning for enriching the user's learning activities.

Hence we develop an integrated authoring system, which provides the authoring abilities for SCORM-compliant course contents, and also deliberately contains some special tags to enrich the learning efficiency, and to obtain the additional information for the analysis of learner's learning profiles.

This paper is organized as follows. We will introduce the SCORM specifications in the second section. And in the third section, we will illustrate the special tags for the SCORM-compliant learning materials. The conceptions of our innovated authoring system will be specified in Section 4. Finally, the conclusions and some future works are shown in the last section.

2. SCORM Specifications

SCORM is mainly consisted of the content aggregation model, the metadata descriptions, run-time environment and simple sequencing specifications. In the SCORM Content Aggregation Model [9], it totally defines five levels to the descriptive content aggregation. With the

definition, instructors can exchange learning content objects with others easily and reduce the costs of authoring.

The metadata is the description of the learning objects. With the metadata mechanism, it is feasible for querying and searching the content objects. Metadata [11] definitions are divided into 9 categories in the SCORM standard as the following:

- General: The general information of the course objects, such as language, title, etc.
- Lifecycle: The history and status of the course objects, such as version, status, role, etc.
- Meta-metadata: The information regarding the metadata itself.
- Technical: The specifications of the technical requirements, such as format, duration, etc.
- Educational: The educational and pedagogic characteristics of the course objects.
- Rights: The intellectual property rights and conditions of using the course objects.
- Relation: Relationship within the course objects.
- Annotation: The comments of the educational uses of the course objects.
- Classification: The properties used by a specific classification system.

The Run-Time Environment [10] in SCORM consists of the API adapters and the Data Model. The Learning Management System (LMS) will launch SCOs(Shareable Content Object) to the learners. During the SCOs launching, some parameters will be transferred to match the data model and restored as the inputs of the learning profiles. These operations can be traced in the learning management system.

The learning sequence is a novel conception described in SCORM 1.3. With the sequencing specifications, the instructors can setup some different learning routes for the learners according to the different condition rules. The sequence rules are divided into 11 categories, such as the Sequencing Rules, Limit Conditions, Objective Map and so on. The instructors can design the learning activity for a group with more than one learning activity.

3. Special Tags for SCORM-Compliant learning materials

In our developed authoring system, in addition to the abilities of course contents authoring, we define several special tags for distinguishing different kinds of learning activities while navigating the online learning contents. Each of the special tags can be embedded between the lines into the SCORM-compliant learning materials generated from the practical authoring tools. When the learners peruse the manufactured learning materials, the specific tags will manage each of the functionalities given by system or assigned by the course designers.

The special tags in our authoring system are divided into four categories as follows: the Reference Tags, the Navigation Tags, the Answer Tags, and the Auxiliary Tags. The four kinds of additional tags embedded in the learning course contents are different from the general hyperlinks with capabilities of the passive navigation in the learning environment. They will actively communicate with the LMS and provide the applicable learning activities to the learners while learning in the cyberspace. In the following subsections, we will discuss the four categorized special tags in our proposed course authoring system.

Reference Tags
The reference tags can be used in the general learning materials. By using the reference tags, the instructors can provide additional learning resources and some supplementary learning contents to the learners.

With comparing to the conventional tags within the hypertext markup Language specifications, our proposed reference Tags contain implicitly the reference conditions defined by the instructors for the learners. This additional information will communicate with the learner's learning profile and deliver the appropriate reference resources to the learners. In the reference Tags category, some necessary tags are included as following:

- [Video]: The instructor of the learning contents can embed the Video Tags within the leaning document. By doing so, the learners can just move the mouse to hover the special tags and bring out the Video clips in the popup screen and some output devices, such as the PDA-like mobile device.
- [Audio]: The main contribution of Audio tags is to improve the audio progress in the SCORM-Compliant environment. The instructor can input the additional information about the audio learning resources, and the learner can enjoy the audio performance by creating the personal registration.
- [URL]: This kind of reference tags seems to be the same with the HTML definition. But in our proposed URL tags, the instructor may combine several reference Urls into one URL Tag. While the learner performing the onmouseclick action, the popup dialog will provide some necessary links for the learning activities to enrich the learning space.

The above-mentioned Reference Tags will be formed as the visual styles with cascading the specified serial item numbers for referring to the learning materials within the contents repository.

Navigation Tags
The main functionalities of Navigation Tags are defining the learners' learning profiles and the recording of the progresses while course learning. In the specifications of our system, some complex learning activities were involved. By recording certain information while learning, the gathered information will be worthy for the instructors to author the learning contents and define the learning rules to improve the learning efficiency of learners. The navigation tags in our authoring system are categorized as following:

- [Page Tag]: For recording the navigation of course contents, and adding to the learning profile. This tag will be formed as [P ddd], where the ddd is the page number in the course content. The recorded information can be used for analyzing in the learning activity tree.
- [Next Page Tag]: This kind of navigation tags provides the learner to navigate the next page in the well-structured learning contents. As the learner clicking on the tag, the status of learning activity tree will be changed immediately and recorded to the learning profile.
- [Previous Page Tag]: By providing the Previous tags, we can be told that which concepts of the learning materials play the important roles while learning, or which content will have the most reference value.
- [Milestone Tags]: The tags define the learning records of the learner, and contain three detailed items with it: the Attempt Tags, the Objective Tags, and the Cooperative Tags.
  (a) [Attempt Tags]: Mainly for accounting the navigation times of each learner. The recorded information will map to the learning rules set by the SCORM Simple Sequencing rules.

  The detailed Attempt tags are as following:

  [Attempt dd/dd]: From learners' perspective, the Attempt Tags are used for recording the times while they trying to obtain further knowledge within the SCOs. The first parameter "dd" is the SCO number mapping to the course materials. According to the SCO number, the content aggregation will easily be obtained, and the learning status of each learner will be consequently kept tracks. Oppositely, when we want to do the learning in the traditional learning environment, the learners have to read the corresponding SCO to the electronic files. And the system will notice the individual learning status, and make some controls about the learning. The second "dd" parameter means the learning times of each learner. As this paper written, the digits of the two parameters have the ranges from 0 to 1000.

  In this part, we also use the Feedback (% number condition) Tags to scheme the revealing of non-computer outputs, such as the electronic screen within the learning devices. By using such a mechanism, the leaner may use different device for the learning activities, and we can still obtain the learning status of each learner and advise some information to the leaner. The information contains the percentage of the learning activities, the number of the attempt times to the advanced SCOs, and the conditions for displaying the methods to pass the advance SCOs. The conditions contain the following status: All, At least one, At least %, and None.
  (b) [Objective Tags]: The objective tags are designed for recording the different objective of each SCO. In the distance learning environment, the different SCOs might have the same learning objective, consequently, we have to add these kinds of information to the SCOs while learner trying to navigate the course materials. The Objective Tags can be formed as [Objective ddd], where the 'ddd' means the number of the learning objective. There are some detailed tag information contained in our proposed Objective tags as the following:

    [Objective Measure Tag]: This kind of objective tags can apply the measurement to the learning objective. The accomplishment of each SCO depends on the predefined course objective, and different SCO has its own measurement for the objective. Hence we design the Objective Measure Tag, with the form of [Objective dddMddd], to see if the learner has accomplished the course objectives.

    [Objective Weight Tag]: The tag is in highly relation with the above Objective Measure tag. After the measure tag is defined, we try to specify the effect to each learner with the SCO accomplishment. Consequently, we use this tag to define the weight of each SCO learning objective. The tag can be in the form of [Objective dddWddd].
  (c) Cooperative Tags: In traditional learning environment, the learner can discuss the course contents or concepts with other learners. In the distance learning environment, if we want to have the same methodology for achieving the target, we have to define additional special tag for those learners who want to cooperate with others. The Cooperative learning focuses on the IMS learning data model [10]. And by adding the tags to the learning contents, the leaner can figure out the learning status of other learners, and try to cooperate with them while progressing the learning contents.
- [Status Tags]: The Status Tags represent the current learning status of each learner. There are four types within the learning status, satisfied, non-satisfied, completed, and incomplete.
- [Exit Tags]: When the learner wants to quit the current learning course contents, he/she may use the exit tag for telling the learning management system the information, and the system will record and commit the current learning status of the leaner.

Answer Tags

The main functionality of Answer Tags is for the assessment in the distance learning environment. In the evaluation of assessment, we take account of the system automation. Hence, some detailed answer tags for different kinds of assessment developed as follows:

- [Multiple-Choice Tags]: The tags provide the multiple choices for the assessment. When the learner click on this type of tag, the system will popup the corresponding quiz for the evaluation.
- [True-False Tags]: The same mechanism for the simple true/false exams for the learners.
- [Fill-in-blank tags]: After performing the clicking action, the corresponding fill in blank exam will be popup from the exam repository.

After having the assessment, the system will record the results of each leaner, and update the learning status, that is, the ability of each leaner.

Auxiliary Tags

By using the Auxiliary Tags, the learner can decide to open or disable the additional tags functionality of the learning environment. There are four types of auxiliary tags in our proposed system as following:
- [Start]: The Start Tags is used to open the functionality with all the tags described above.
- [End]: To disable the functionality with all the tags described above.
- [Pause]: To pause the functionality with all the tags described above if the leaner need to do something else for a while.
- [Continue]: To continue the functionality if the current learning state is set to be Pause. The system will perform the necessary tasks again with the current learning status.

Because of the LMS will take responsibility for the learning profile of each learner, such as the learning time or the navigation tracks. This is important for the analysis of the learner and the guiding principle of the instructor. Accordingly, in the learning environment, we need some auxiliary tags for aiding the learner with their convenience.

4. System Design

Our integrated Authoring System, which supports the special tags mechanism, combines three main technologies and subsystems: the SCO authoring system, the Special tags embedding system and the Exam Authoring System. Our proposed system architecture is as the figure 1, and there are total three databases we may use in this system, the SCOs database for maintaining the learning materials, the exams database for assessment, and the learner profile database for recording the learning progress of each learner. All the learning contents are based on the specification of SCORM 1.3, hence we can easily obtain the relative learning resources with the sharability.
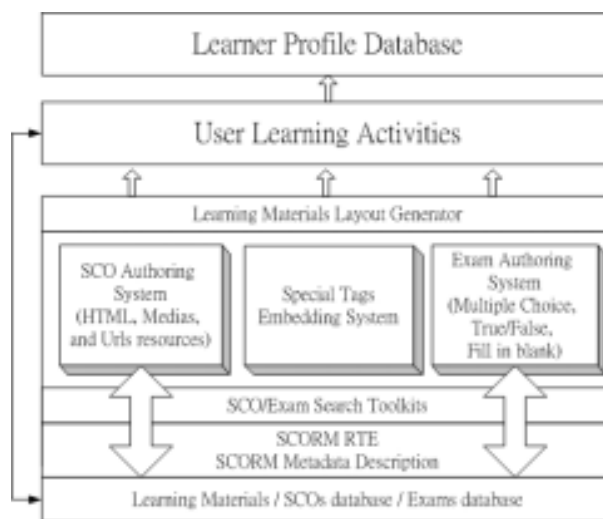
Figure 1. Architecture of our SCORM-compliant Authoring System with special tags

Besides the repository mechanism, in the system, we also provide the easy-to-use query interface for the learners to obtain the desirable learning resources from the integrated SCORM metadata descriptions.

Figure 2 shows the System interface of our integrated Authoring environment. The left-hand side of the appearance is designed for the special tags embedding. The main workplace in the middle is the course materials authoring environment. With such functionalities, instructors can design the SCO with different medias, and conduct the course authoring as the traditional Microsoft FrontPage-like editing environment. After the authoring activities performed, the output of each SCO and the associated medias will be stored in the database repository. The right-hand side of the system tableau is designed for the Sequencing Management of the curriculum specified in the SSS (Simple Sequencing Specification) of SCORM version 1.3. In this working space, the instructors can also fulfill the metadata of each SCO, or proceed the authoring the content of each SCO according to the selection of the items within the popup content menus.

With the completion of all the authoring and products, the learners can go forward the learning activities according to the delegated learning contents. The system will record the learning activities of each leaner corresponding to the special tags the instructors designed, and learning profiles will be stored in the learner profile database. Such a learning profile can be taken into the analysis of each learner, and leads to some advanced course contents to achieve the enrichment of knowledge.
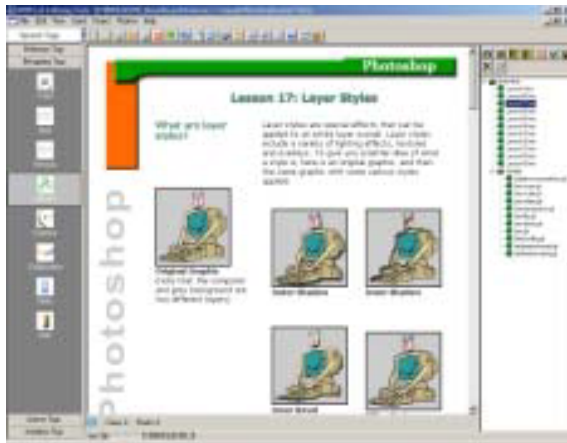
Figure 2. Special Tags Authoring System

In the following subsections, we will elaborate on the main projects in our proposed system.

SCO Authoring System
As mentioned to the on-line educations, the most fundamental element we are expecting for is the learning materials. The learning materials are various and plentiful to trigger the interesting of the learners. In our proposed authoring system, we provide the essential functionalities within the general document editors.
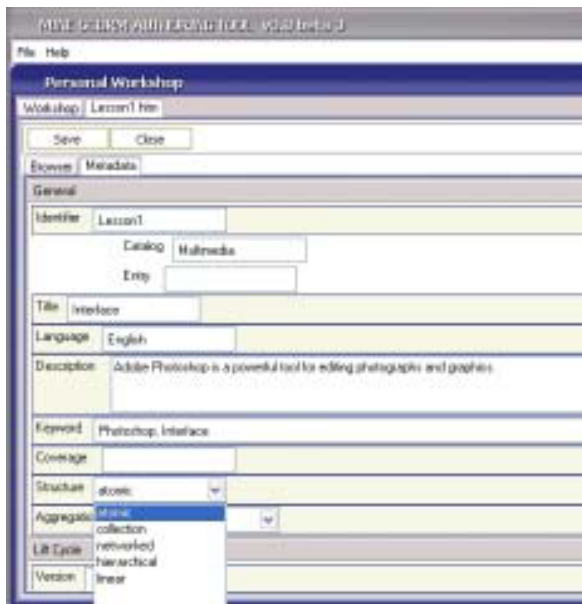
Figure 3. SCO metadata designer

According to the SCORM 1.3 specification, there are nine categories for the metadata description. The metadata is designed for the sharability in the Ilities within SCORM. According to the criterion of SCORM 1.3 specification, it is hard for instructors to edit and fill the numerous metadata attributes of the learning contents and the information of aggregations. However, those metadata indeed contribute the fundamentality for the E-learning. Hence in our system, we provide the automatic and semi-automatic metadata filling

system, as seen in the Figure 3. In the system illustration, we use the drop-down menus for the filling of each SCO metadata. This will remarkably reduce the burden while editing the complicated metadata. Furthermore, some information can be fulfilled as the default value according to the different identification of the instructors.

In our integrated authoring system, we also provide the Simple Sequencing Specification for instructors to standardize the learning sequence of the learning activities. The conditions can be set easily by selecting the dropdown menus.

Special Tags Embedding System
In the special tags embedding system, we use the user-friendly panel-bar-like interface for the embedding functionalities, as shown in figure 4. The Special tags embedding system contains the functionalities for the four categories described above. In each categorized tag, the instructors can fulfill the necessary information about the specified tags, as shown in figure 5, by choosing the item contained in the panel bar, and embed the tags into the appropriate place with the mouse indexing.
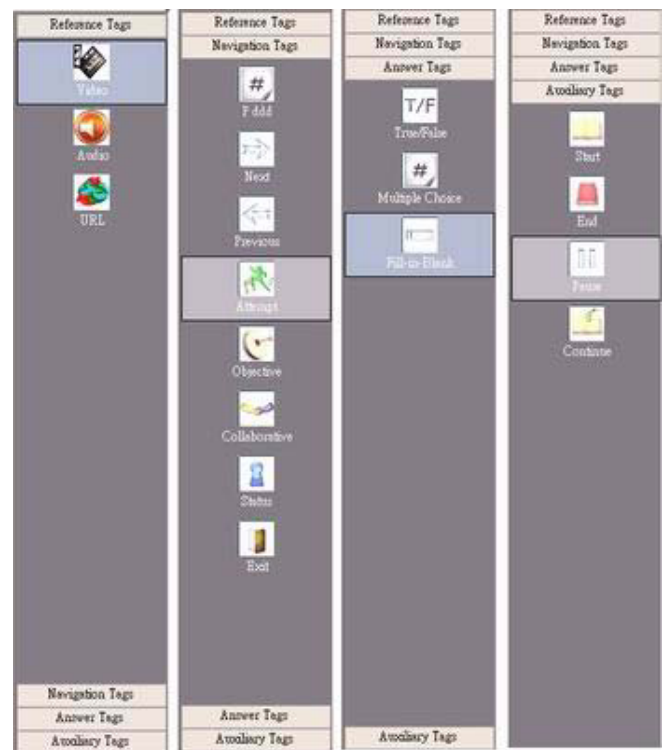
Figure 4. Four categories of special tags for the SCORM-Compliant learning environment

The objective of this subsystem is to insert the specified tags into the corresponding position within the learning contents. After the completion, the output of each SCO will still maintain attributes of the SCORM specifications, and additional information with the special tag were embedded for aiding the analysis of learning activities.
In our proposed system, for avoiding the duplications of

output formats, we take into account the different versions of the learning contents outputs, such as the computer-based outputs, and the PDA versions outputs. The instructors can choose the output type of each SCO easily by selecting the simplified checkbox.
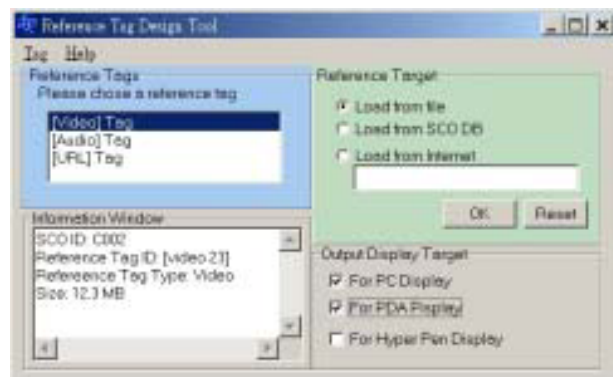
Figure 5. Reference Tag Design Tool

Exam Authoring System
In our integrated system, the course contents are derived from the off-the-shelf pre-designed learning contents or developed via our SCO Authoring System. Learners can process those learning materials and obtain the knowledge within the learning contents. From the educational perspectives, the assessment is an important issue for the learning, especially in the distance learning environment because of the implications. Accordingly in our authoring system, we also concern about the developing of the exam authoring for the instructors to design the suitable exams for the assessment, and the way to deal with such learning results.

There are three types of exam styles contained in our exam authoring system as we mentioned in the Answer Tags specification. We offer an integrated exam authoring system for instructors to design all the three types of examination questions. After the exams authoring, the exams will be automatically stored in the exams repository and communicate with the Answer Tags embedded in the learning contents. The Answer Tags can be defined by the instructors, and be embedded in the appropriate place within the learning contents. When the learner starts the navigation to the learning contents, the Answer Tags will monitor the occurrence that may be included in the learning activities and perform the assessment to the learners. Consequently, the learning results will be taken down into the learner profiles, and influence the learning activities as the condition rules.

5. Conclusion and Future Works
Our integrated authoring system is designed with the ideas of applying the additional learning tags to enrich the learning efficiency in the distance learning environment. And we adopt the standard based on the SCORM specifications. There are three main sub-systems contained in our integrated authoring system, the SCO authoring system for contents authoring and metadata fulfilling, the special tags embedding system for enriching the learning activities, and finally, the exam authoring system for the assessment and evaluation.

The objective of our system is made to reduce the learning contents authoring costs for the instructors, and apply the pedagogical tags corresponding to the traditional learning behaviors. By using the user-friendly authoring interface, instructors can generate the SCORM-compliant learning contents and the learning sequence specifications by performing drag and drop operations. In addition, we also provide the contents authoring workspace for the familiar off-the-shelf course designer. It's desired to be mentioned that we deliberately design some pedagogical learning tags embedded in the learning contents to communicate with the learning activities.

The current version of our integrated system has the adequate capabilities for the authoring of learning contents. In the future, we are expecting for the different oncoming generations of learning, such as the revolution of the online learning activities. Also, we also intend to make the mobile learning environment with different learning devices and learning resources.

Reference

[1] ADL SCORM Version 1.3 Application Profile (2002), Advanced Distributed Learning Initiative, http://www.adlnet.org/.

[2] Universal Learning Format (ULF) Technical Specification (2002) Version 1.0, http://www.saba.com.

[3] SOAP Version 1.2 (2003), W3C, http://www.w3.org/.

[4] Emerging and Enabling Technologies for the Design of Learning Object Repositories Report (2002), Version 1.0, http://www.adlnet.org/.

[5] IMS Question & Test Interoperability (2002), IMS Global Learning Consortium, Inc., http://www.imsglobal.org.

[6] Aviation Industry Computer Based Training Committee. Web site at http://www.aicc.org/ [Last access April 13th, 2004]

[7] W. Hodgins. "*Draft Standard for Learning Object Metadata (LOM) Specification*". Proposed Draft 6.1, Institute of Electrical and Electronics Engineers, Inc., 2001. Electronic version available at http://ltsc.ieee.org/doc/wg12/LOM_WD6-1_1_without_tracking.pdf [Last access September 13th, 2002]

[8] IMS Global Learning Consortium. Web site at http://www.imsproject.org/ [Last access April 13th, 2004]

[9] P. Dodds. "*Sharable Content Object Reference Model (SCORM)*". Version 1.2. Technical report, The SCORM Content Aggregation Model, Advanced Distributed Learning Initiative, October 1 2001.

[10] P. Dodds. "*Sharable Content Object Reference Model (SCORM)*". Version 1.2. Technical report, The SCORM Run-Time Environment, Advanced Distributed Learning Initiative, October 1 2001.

[11] P. Dodds. "*Sharable Content Object Reference Model (SCORM)*". Version 1.2. Technical report, The SCORM Overview, Advanced Distributed Learning Initiative, October 1 2001.

A Semantic-based Automated Question Answering System for e-Learning

Che-Yu Yang, Mao-Shuen Chiu, Chao-Hsun Yang and Timothy K. Shih
Department of Computer Science and Information Engineering
Tamkang University
Tamsui, Taipei, 251, Taiwan, ROC
Email: yang@schummi.com

Abstract

Interaction between the student and the instructor is important for the student to gain knowledge. Also, one of the major tasks on the instructor in e-learning is to reply student e-mails and posted messages. Students usually raise their questions by these two methods in an e-learning environment. In this paper, we introduce a semantic-based automated question answering system that can act like a virtual teacher to respond to student questions online. With the system, not only the instructor can be relieved from the load of answering lots of questions, but also the student can mostly get answers promptly without waiting for the instructor to get online and provide an answer. This would be a big help for both the instructor and the student in e-learning.

Through the process of raising questions and getting answers, the knowledge base will be enriched for future questions answering. Further, not only the students can get answers for their questions, but also the instructors could know what problems students encounter in learning. These would be big aids to both the teaching and the learning.

Keywords: automated question answering, natural language processing, distance education, word sense disambiguation, Wordnet.

1. Introduction

For the interactions between the instructor and the student in distance learning, there is a problem that the instructor cannot be online all the time and it is not possible for the instructor to deal with lots of questions from students in a timely manner. Therefore, an automated question answering system is definitely needed in e-learning. To build an automated question answering system, question-answer (Q&A) sets from students (questions) and the instructor (answers) must be collected first as initial knowledge base for the system.

Most existing question answering systems are based, either directly or indirectly, on models of the traditional Information retrieval (IR) system. Examples of classic models include the probabilistic [17] or Bayes classifier model [21], and the vector space model (Salton et al., 1975 [18]). Many others have been proposed and are being used (Van Rijsbergen, 1986 [21, 22]; Deerwester et al. [5], 1990; Fuhr [7], 1992; Turtle and Croft, 1991 [20]).

Most of these textual materials retrieving approaches depend on a lexical match between words in users' requests and words in database objects. Typically only text objects that contain one or more common words with those in the users' query are returned as relevant. These word-based retrieval systems, however, are far from ideal ─ many objects relevant to a users' query are missed, and many unrelated objects are retrieved. Some researches show that fundamental characteristics of human verbal behavior result in these retrieval difficulties (Bates, 1986 [3]; Fidel, 1985 [6]; Furnas et al., 1988 [8]). Because of the tremendous variety in the words people use to describe the same meaning or concept (*synonymy*), people will often use different words from the author or indexer of the information, and relevant materials will therefore be missed. On the other hand, since the same word often has more than one meaning (*polysemy*), irrelevant materials will often be retrieved.

Textual retrieval systems that utilize automatic indexing techniques to create text representatives from natural language, for better performance, must deal with the problems of polysemy and synonymy. Polysemy, a single word form having more than one meaning, decreases retrieval precision by false matches. While synonymy, multiple words having the same meaning, decreases the recall by missing true conceptual matches.

We try to overcome these problems by indexing textual information by its underlying concepts, rather than the keywords (word forms). All human languages have words that can mean different things in different contexts, such words with multiple meanings are potentially *ambiguous*. Polysemy and synonymy can be handled by assigning different senses of a word different concept identifiers and assigning the same concept identifier to synonyms. Such the process of deciding which of their several meanings is intended in a given context is known as *Word Sense Disambiguation* (WSD). In other words, word sense disambiguation is the task of automatically figuring out the intended meaning (concept) of a word when used in a sentence. Accurate word sense disambiguation can lead to better results for information retrieval. The words in an English question can be "disambiguated" and the query subsequently expanded to include similar words from online dictionaries. Furthermore, the searched objects can also be processed in similar way.

Human beings are especially sophisticated at WSD. For example, given the sentence "*The bank holds the mortgage on my home*", we immediately know that the *bank* here refers to a financial institution that accepts deposits and channels the money into lending activities. Whereas given the sentence "*He sat on the bank of the river and watched the currents*", the *bank* here means the sloping land beside a body of water. But unfortunately, it is very difficult for computers to do the same job effortlessly.

Because meaningful sentences are composed of meaningful words, any computer system that hopes to process natural languages as human do must have information about words and their meanings. This information is traditionally provided through dictionaries, and digitalized dictionaries are now widely available. But most dictionaries are designed and constructed for the convenience of human readers, not for machines. Fortunately, there are a few machine-readable dictionaries emerged and developed continuously. One of the most widely known is the *Wordnet*, developed by George Miller at Princeton University. We will introduce it later in this paper.

2. Wordnet

WordNet is a machine-readable dictionary (MRD) developed by George Miller and his colleagues at the Cognitive Science Laboratory at Princeton University. It is an online lexical database designed for use under program control, it provides a more effective combination of traditional lexicographic information and modern computing (George Miller, 1993 [9]). WordNet is like a dictionary in that it stores words and meanings. However it differs from traditional ones in many aspects. For example, words in WordNet are arranged *semantically* instead of alphabetically. Synonymous words are grouped together into synonym sets, called *synsets*. Each such synset represents a single distinct sense or *concept*. For example, in Wordnet, the synset {car, auto, automobile, machine, motorcar} represents the concept of "4-wheeled motor vehicle; usually propelled by an internal combustion engine".

WordNet stores information about words that belong to four parts–of–speech: nouns, verbs, adjectives and adverbs. There are 129,509 words organized in 99,643 synsets, Approximate 17% of the words in WordNet are polysemous; approximate 40% have one or more synonyms, some 300 prepositions, pronouns, and determiners — although play an important role in any natural language parsing system, they are given no semantic illustration in WordNet (George A. Miller, Richard Beckwith, 1993 [10]).

Wordnet database groups English nouns, verbs, adjectives, and adverbs into sets of synonyms that are in turn linked through semantic relations that determine word definitions and senses. WordNet features a rich set of 299,711 relation links among words, between words and synsets, and between synsets. Table 1 lists some of the semantic relations (links) defined in Wordnet (C. Fellbaum, 1998 [4]).

Table 1. Some semantic relations defined in Wordnet.

Semantic Relation	Meaning	Example
Synonymy	X is similar to $f(X)$	homo, man, human being, human
Hypernym	X is a kind of $f(X)$	Apple *is a kind of* fruit
Hyponym	$f(X)$ is a kind of X	Zebra *is a kind of* Horse
Holonym	X is a part/member of $f(X)$	Wheel *is a part of* a car
Meronym	X has part/member $f(X)$	Table *has part* leg
Antonym	$f(X)$ is the opposite of X	Wet *is the opposite of* dry

The two most typical relations for nouns are *hyponymy* and *hypernymy*. These relations connect two synsets if one referred to by another is "*is a kind of*", or "*is a specific example of*". That is, if synset A is *a kind of* synset B, then A is the *hyponym* of B, and B is the *hypernym* of A (C. Fellbaum, 1998 [4]). For instance, {car, auto, automobile, machine, motorcar} are the hyponyms of {motor vehicle, automotive vehicle}, and {motor vehicle, automotive vehicle} are their hypernyms.

The other typical relations for nouns are *holonymy* and *meronymy*. These relations connect two synsets if one referred to by another one is "*is a part of*". That is, synset A is a *meronym* of synset B if A *is a part of* B. Conversely, B is a *holonym* of A if B *has* A *as a part*. There are three types of holonyms: *Member–Of*, *Substance–Of* and *Part–Of*. Conversely there are three types of meronyms: *Has–Member*, *Has–Substance* and *Has–Part* (C. Fellbaum, 1998 [4]).

Two other relations defined for nouns are antonymy and attribute. Antonymy links together two noun words that are opposites of each other.

Two major relations defined for verbs in WordNet are *hypernymy* and *troponymy*. These relations are similar to the noun *hypernymy* and *hyponymy* relations respectively. Synset A is the hypernym of B, if B *is one way to* A, and B is then the troponym of A (C. Fellbaum, 1998 [4]). Just like nouns, verbs synsets are also linked through the relation of *antonymy*. Other relations defined for verbs are *entailment* and *cause*.

The semantic relations for adjectives and adverbs in WordNet are fewer than those for nouns and verbs, and adverbs have even far fewer relations. The most frequent relation defined for adjectives is *similar to*. As nouns and verbs, the semantic relation *antonymy* links together words that are opposite in meaning to each other for both adjectives and adverbs

We utilize Wordnet to accomplish word sense disambiguation (WSD), the result is - each keyword in a sentence in the documents and query is mapped into its corresponding semantic form (concept) as defined in Wordnet. Not only we can use the semantic meaning of the keywords to index the content, but also this step enables subsequent query expansion based on semantic concepts rather than keywords. We will explain these procedures later.

3. Word Sense Disambiguation

Word sense disambiguation (WSD) is the task of deciding which sense a word has in a given context. It has been very difficult to formalize the automatic process of disambiguation, which humans can do so effortlessly.

Much of the early semantic relatedness study in natural language processing centered around the use of Roget's thesaurus (Yaworsky 92 [23]). As WordNet became available later, most of the new work utilized it (Agirre & Rigau 96 [1], Resnik 95 [16], Jiang & Conrath 97 [12]). One of the many applications of semantic similarity models is for word sense disambiguation (WSD).

Let's now look at some WSD methods that have been proposed based on Wordnet. A common method of measuring semantic similarity is to consider the taxonomy as a tree or lattice in semantic space. The distance between concepts/nodes within that space is then taken as a measurement of the semantic similarity. These methods can roughly be classified into two types. Basically, one is based-on the *edges* (semantic links) between synsets, the other is based-on the *information content*.

Edge-based Approaches

In the line of the edge-based approach, semantic distance is calculated using the edge counting principle. If all the edges (branches of the tree) are of equal length, then the number of intervening edges between two synsets is a measure of the distance. The measurement usually used (Rada et al. 1989 [14], and Lee et al. 1993) is the shortest path between concepts. This relies on an ideal taxonomy with edges of equal length. Unfortunately in taxonomies based on natural languages, the edges are not the same length.

A number of different methods related to distance using edges have been modified to try to correct the problem of this non-uniformity. These modifications include the density of the sub-hierarchies, the depth in the hierarchy where the word is found, the type of links.

(Hirst and St-Onge's, 1998 [11]) measure semantic relatedness based on that, two lexicalized concepts are semantically close if their WordNet synsets are connected by a path that is not too long and that "does not change direction too often".

The use of density is based on the observation that words in a more densely part of the hierarchy are more closely related than words in sparser areas (Agirre and Rigau 1996 [1]). They proposed the *conceptual density* concept for WSD. Given the WordNet as the structured hierarchical network, the conceptual density for a sense of a word is proportional to the number of contextual words that appear on a sub-hierarchy of the WordNet where that particular sense exists. The correct sense can be identified as the one that has the highest density value.

For this density concept to be valid, the hierarchy must be fairly complete or at least the distribution of words in the hierarchy has to closely reflect the distribution of words in the language. Unfortunately, neither of these conditions holds completely. Furthermore, the observation about density may be an overgeneralization, just like simple edge counting is.

Depth in the hierarchy is another attribute often used. The hierarchy of WordNet is deep, the type of link in WordNet is explicit, such as IS-A-KIND-OF and HAS-PART. Some of the studies that have used the above concept include Sussna (Sussna,1993 [19]) who weighted the edges by using the density of the sub-hierarchy, the depth in the hierarchy and the type of link. (Leacock and Chodorow, 1998 [13]) also rely on the length of the shortest path between two synsets for their measure of similarity. However, they limit their attention to IS-A-KIND-OF links and *scale* the path length by the overall depth D of the taxonomy. They all reported improvement compared to straight edge counting.

Information (node) Based Approaches

Those analytic methods described above now face competition from statistical and machine learning techniques. And some hybrid approaches, that combine a *knowledge-rich* source, such as a thesaurus, with a *knowledge-poor* source, such as corpus statistics (Resnik, 1995 [15]; Lin, 1998; Jiang and Conrath, 1997 [12]), have been proposed.

Face above problems with distance related measures, Resnik (Resnik, 1995 [16]) proposed the idea of *information content* of the concepts — he combined together ontology and corpus. By the intuition that the similarity between a pair of concepts may be judged by "the extent to which they share information", Resnik defines the similarity of two concepts defined in WordNet to be the maximum *information content* of their lowest super-ordinate (most specific common subsumer) The Information Content of a concept relies on the probability of encountering an instance of the concept. Resnik used the relative frequency of occurrence of each word in the Brown Corpus to compute this probability.

(Jiang and Conrath, 1997 [12]) also used information content to measure semantic relatedness, but they combined it with edge counting using a formula that also took into account local density, node depth and link type. Their approach uses the notion of information content in the form of the conditional probability of encountering an instance of a child-synset when given an instance of a parent-synset. Thus the information content of the two nodes, as well as that of their most specific/lowest subsumer, plays a role.

Our WSD approach

Word-sense disambiguation is the core of our system. In our system, each keyword in the question and Q&A sets is indexed with its corresponding concept (sense-id) defined in Wordnet. So the performance of WSD is critical for correctly retrieving the answer for user's question. Besides, this would also facilitate subsequent query expansion based on semantic concepts rather than keywords.

Our method is a hybrid approach that combines a knowledge-rich source, Wordnet, with a knowledge-poor source, the Internet (World Wide Web) search. Not only the World Wide Web is the most rich and domain extensive natural language text resource, but also the context on it is very up to date and grows continuously. As the context of webpages on the Internet are composed by

innumerable people, the validity for used as a WSD source is thus guaranteed. Furthermore, there are now a few sophisticated and powerful keyword-based (lexical) World Wide Web search engines can be used to help the task.

We utilize *AltaVista* search engine which supports complex Boolean search through its advanced search function, including brackets, AND, OR, NOT, and NEAR operators. For the context of a question or Q&A sets, the words are paired, and each word is disambiguated by searching the Internet with queries formed using different senses (actually, the word forms in each synsets) of each of the two words. The senses are then ranked by the number of hits with normalization. All the words in the context are processed and senses are ranked in this way. The algorithm is as follows.

Our WSD algorithm

1. Given $T=\{t_1, t_2, t_3 \ldots, t_n\}$, is a set of terms (word forms) from a user question or a Q&A set (or any text document), to disambiguate every terms:

2. For each pair $\{t_i, t_j\}$ of terms from the set T, where $1 \leq i, j \leq n, i \neq j$;

3. Look up in Wordnet for every synsets defined for t_i and t_j respectively, store them in the sets S_i, S_j, thus $S_i = \{s^i_1, s^i_2, s^i_3 \ldots, s^i_m\}$ which is the set of all possible senses/synsets of term t_i, where t_i has m senses/synsets defined in Wordnet; $S_j = \{s^j_1, s^j_2, s^j_3 \ldots, s^j_n\}$ which is the set of all possible senses/synsets of term t_j, where t_j has n senses/synsets defined in Wordnet;   /*note that S_i and S_j are sets of synsets*/

4. From Wordnet, get all the synonymous terms defined in every synsets in S_i and S_j. For example, for synset s^i_1 in S_i we get a set of terms $W^i_1 = \{w^{i(1)}_1, w^{i(2)}_1 \ldots, w^{i(y)}_1\}$, which is the set of all terms representing a single unique concept defined in Wordnet, and for this example, there are y synonymous terms (word forms) defined in synset s^i_1; the same work are done for every synsets in S_i, S_j;

5. For each synsets in S_i (for s^i_a, a=1 to m)   /* $s^i_1, s^i_2, s^i_3 \ldots, s^i_m$ respectively */

6. For each synsets in S_j (for s^j_b, b=1 to n)   /* $s^j_1, s^j_2, s^j_3 \ldots, s^j_n$ respectively */

7. Search the WWW using AltaVista by the query $Q(s^i_a, s^j_b) = (w^{i(1)}_a$ OR $w^{i(2)}_a \ldots$ OR $w^{i(y)}_a)$ AND $(w^{i(1)}_b$ OR $w^{i(2)}_b \ldots$ OR $w^{i(z)}_b)$ /* note that here the AND, OR and brackets in the query are the operators provided by AltaVista's search function */

8. Similarity between two senses — sense s^i_a of the term t_i and sense s^j_b of the term t_j is calculated as, $Sim(s^i_a, s^j_b) = ($ number of hits of $Q(s^i_a, s^j_b)) / (y+z)$ /* divided by y+z for normalization */

9. End for

10. End for

11. Select the pair of synsets having the lagest Similarity value as the correct senses for term t_i, and term t_j respectively.

12. End for

4. The Question Answering System

The architecture of the semantic based automated question answering system is shown in Figure 1. There are six main components in the system, including the questioner assistant agent, the Q&A acquirer, the Q&A database, the lexical parser, the semantic index module and the answer generator.
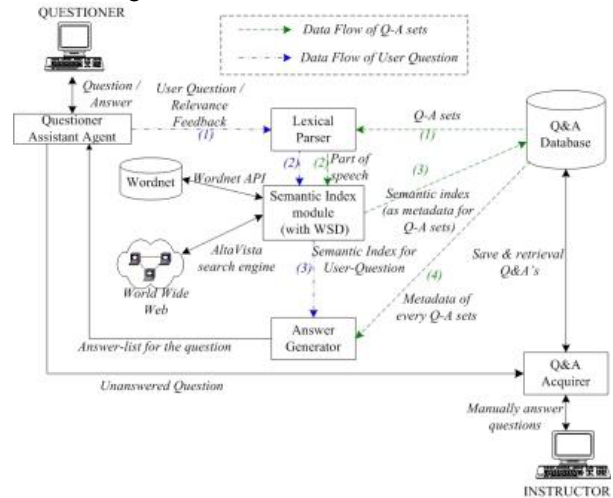
Figure 1. Architecture of the Semantic-based Automated Question Answering

The functions of the components are briefly described as follows.

- Questioner Assistant Agent: The main function of this component is to be the interface between the student/questioner and the Q&A system. The student can send his questions and receive answers via this agent. He can also provide relevance feedback to improve the search results.

- Q&A Acquirer: This component presents unanswered questions from the *Questioner Assistant Agent* to the instructor and provides an interface for the instructor to manually answer the questions. This would form new Q&A sets which are then saved to the Q&A database. We have this component for user's questions might be unanswered by the system due to there is no corresponding answer for it in the Q&A database.

- Q&A Database: This is the repository of all Q&A sets and their metadata (semantic index). It can be collected on a question-by-question basis through the Q&A acquirer, or from a batch file arranged by the instructor.

- Lexical Parser: This component accepts sentences from user question or Q&A sets, and tags part-of-speech (POS) for them. These words combined with their part-of-speech would be used to query the Wordnet for their synsets. As we have mentioned earlier, Wordnet database groups English nouns, verbs, adjectives, and adverbs into sets of synonyms, so it is critical to choose the correct part-of-speech for the words in sentences for later query to Wordnet. For example, in the sentence "I like to swim", the part-of-speech of the term "like" here is a verb, not an adjective. For being as a verb,

the retrieved synset might be {wish, care, like}; while for being as an adjective, the retrieved synset could be {like, similar}. The Maximum Entropy-Inspired (MEI) parser is one of the best, so we integrated it into our system.

- Semantic Index Module: This module is the core of the system. It takes the output (words with their POS from a question or Q&A set) of lexical parser as the arguments to query for the synsets in Wordnet and search the WWW to carry out WSD for these words. In Wordnet, each such synset represents a single distinct sense or *concept*, and is assigned a unique identification, called *sense id*. After we disambiguate words in a text, we get the corresponding concepts/senses defined in Wordnet for the words, and use their sense id as the index for that text. The text here is a user question or Q&A sets.

- Answer Generator: This component computes the match rank between the semantic index of a new incoming question and the semantic index of the Q&A sets in the Q&A database. We use the simple Boolean match to retrieve semantically related Q&A sets to the user question, and rank the match list by equation 1:

For a pair of Boolean matched question q and Q&A set qa_i, the rank for them is

$$Rank(q, qa_i) = \frac{M \times C}{|qa_i|} \ \ldots\ldots\ldots\text{ equation 1,}$$

where M is the total count of instance of any concept matches, C is the count of unique concept match, $|qa_i|$ is the length (number of all words) of the Q&A set qa_i.

For example, if a user question q is indexed with Wordnet sense id's as {2, 5, 30}, and there is a matched Q&A set qa_i, which is indexed as {2, 78, 3, 2, 50, 61, 13, 5, 2}, and the length of qa_i is 25. then the rank of this Q&A set to the question is:

$$Rank(q, qa_i) = \frac{M \times C}{|qa_i|} = \frac{(3+1) \times 2}{25},$$

as there are three instances of match of the sense-id "2", one instance of match of sense-id "5", and there are two distinct sense-id's, "2" and "5" matched.

The answer generator then produces the final answer list and sends it to the question assistant agent.

Relevance feedback is an important strength. After the student receives a list of answers from the system, the student question can be reformulated by relevance feedback. The idea is that questions specified by students sometimes fail to describe completely what those students want. As a result, typical questions miss many relevant answers. However, if the student can identify some retrieved answers as relevant, then the system can use this information to reformulate the original question into a new one that may capture some of the concepts not explicitly specified in the original question, which will hopefully yield improved results.

5. System Implementation

A prototype of the system has been developed. About 2,000 Q&A sets in the area of science have been built for testing. The user can key in an English question in natural language about this area and get answers immediately. The answers are ordered according to their ranking.

For the questioner assistant agent, an interface for the student to ask questions and get answers is developed, as shown in Figures 2 and 3. Under the interface of Figure 2, the student can submit his/her question. And he or she can receive a list of answers from the answer generator as in Figure 3. The student can evaluate the answers for improvement of the system. If no satisfactory answer is shown on the answer list, the question is sent to the Q&A acquirer by the question assistant agent to be presented to the instructor.

Figure 2: The student interface – Ask a question
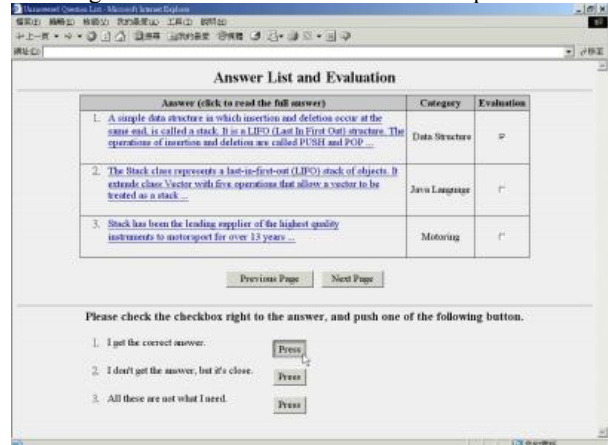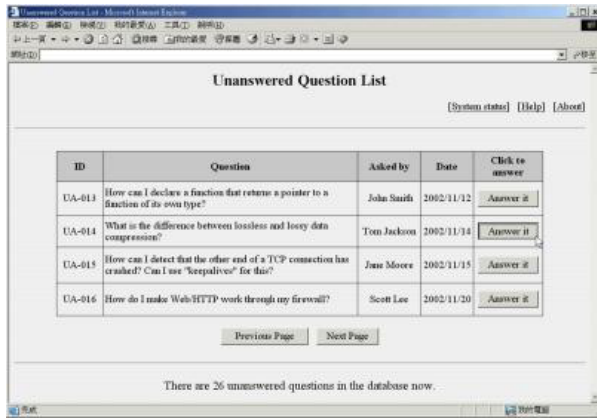
Figure 3: The student interface – Show the answers and get feedback

For the Q&A acquirer, an interface for the instructor to answer questions is also constructed, as shown in Figures 4. Under this interface, the instructor can collect student questions. The interface shows a list of unanswered questions to the instructor. And the instructor can choose any question to respond with an answer. An agent can also help the instructor to retrieve related answers from the Q&A database. Or the instructor can just fill in the answer directly. The Q&A pair is then saved to the Q&A database by the Q&A acquirer.

Figure 4: The instructor interface – List unanswered-questions

6. Conclusions and Future Work

A system designed for automatically answering student questions in an e-learning environment is developed. It can score possible answers and presents the answers to the student in semantic way. It is very important to have such a system in e-learning to help both the instructor and the student.

Most of text retrieving approaches depend on a lexical match between words in users' requests and words in database objects. These word-based retrieval systems, are far from ideal ─ many objects relevant to a users' query are missed, and many unrelated objects are retrieved. This is due to the tremendous variety in the words people use to describe the same meaning or concept (synonymy), and the same word often has more than one meaning (polysemy). We overcome these problems of ambiguity by indexing textual information with its underlying concepts using Wordnet and our proposed WSD method. Students thus can use any words they get used to, to raise their questions in natural language. Furthermore, they can describe their needs freely, rather than striving to choose suitable key terms and use complex query operators.

References

[1]. Agirre and Rigau. 1996. Word sense disambiguation using conceptual density. In Proceedings of the 16th.

[2]. International Conference on Computational Linguistics, pp. 16–22, Copenhagen.

[3]. Bates, M. J. Subject access in online catalogs: A design model. Journal of the American Society for Information Science, 37, 357-376. 1986

[4]. C. Fellbaum, An Electronic Lexical Database, MIT Press, Cambridge, Mass., 1998.

[5]. Deerwester, S., Dumais, S., Furnas, G., Landauer, T., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407.

[6]. Fidel, R. Individual variability in online searching behavior. In C.A. Parkhurst (Ed.). ASIS'85: Proceedings of the ASIS 48th Annual Meeting, Vol. 22, October 20-24, 1985, 6972.

[7]. Fuhr,N. and Buckley, C. (1991).A probabilistic learning approach for document indexing. *ACM Transactions on Information Systems*, 9(3):223–248.

[8]. Furnas et al., Information retrieval using a singular value decomposition model of latent semantic structure. In Proceedings of the Eleventh International Conference on Research & Development in Information Retrieval, pages 465--480, 1988.

[9]. George A. Miller, "WordNet: A Lexical Database," Comm. ACM, Vol. 38, No. 11, 1993, pp. 39-41.

[10]. George A. Miller, Richard Beckwith. Introduction to WordNet: An On-line Lexical Database, Revised August 1993.

[11]. Hirst and D. St-Onge. Lexical chains as representations of context for the detection and correction of malapropisms. In WordNet: An electronic lexical database. MIT Press, 1998.

[12]. Jiang, J.J. and D.W. Conrath (1997) "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy", in Proceedings of ROCLING X (1997) International Conference on Research in Computational Linguistics, Taiwan, 1997.

[13]. Leacock and Chodorow. 1998. Combining local context and WordNet similarity for wordsense identification. In Fellbaum 1998, 265-283.

[14]. Rada, R., H. Mili, E. Bicknell, and M. Bletner(1989) "Development and Application of a Metric on Semantic Nets". IEEE Transactions on Systems, Man and Cybernetics, Vol. 19, No. 1, 17-30

[15]. Resnik, P. (1993) "Semantic Classes and Syntatic Ambiguity", In Proc. of the ARPA Workshop on Human Language Technology. Princeton, 1993.

[16]. Resnik, P. (1995) "Using Information Content to Evaluate Semantic Similarity in a Taxonomy", Proceedings of the 14th International Joint Conference on Artificial Intelligence, Vol. 1, 448-453, Montreal, August 1995

[17]. Robertson, S. (1977). The probability ranking principle in information retrieval. *Journal of Documentation*, 33:294–304.

[18]. Salton, G.,Wong, A., and Yang, C. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620.

[19]. Sussna, M. Word sense disambiguation for free text indexing using a massive semantic network. In Proceedings of the Second International Conference on Information and Knowledge Management, Arlington, Virginia, 1993.

[20]. Turtle, H. and Croft, W. (1991). Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems*, 9(3):187–222.

[21]. Van Rijsbergen, C. (1979). *Information Retrieval*. Butterworths, London.

[22]. Van Rijsbergen, C. (1986). Anon-classical logic for information retrieval. *Computer Journal*, 29:481–485.

[23]. Yaworsky, D. (1992) Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on LargeCorpora. Proceedings of the 15th International Conference on Computational Linguistics (Coling '92). Nantes, France.

AudioPeer: A Collaborative Distributed Audio Chat System[*]

Roger Zimmermann, Beomjoo Seo, Leslie S. Liu, Rahul S. Hampole and Brent Nash
Integrated Media Systems Center
University of Southern California
Los Angeles, California 90089
[rzimmerm, bseo, shihuali, hampole, bnash]@usc.edu

Abstract

Educational tools that utilize the Internet to reach off-campus students are becoming more popular and many educational institutions are exploring their use. We report on a multiuser audio chat system that is using a multicast peer topology and is based on a new audio streaming protocol called YimaCast. The tool is designed to foster collaboration and interactive learning between students, teaching assistants and instructors. The decentralized nature of the audio chat system avoids bottlenecks and allows it to scale to large groups of participants.

We describe the system architecture, its multiple components and how it integrates with the existing distance education infrastructure at our university. We include some preliminary results from our prototype system that demonstrate the feasibility and practicalilty of our approach.

1 Introduction

The expanding capabilities of the Internet to handle digital media streams is enabling new applications and transforming existing applications in many areas. One of the fields that is profoundly affected is higher education. Traditionally, students have attended classes in lecture halls on college and university campuses. The next step was distance education, enabled via video satellite links. More recently, new communications media have broadened the potential audience and allowed anybody with a broadband Internet connection to potentially receive audio, video and slide presentations on their computers.

However, with the basic lecture distribution technology now being available, many educational institutions are grappling with the impact of this technological shift in education. Many pedagogical and policy issues must be resolved in addition to the technological challenges. The University of Southern California's Distance Education Network (DEN) is actively engaging in learning and technological issues to provide both on- and off-campus students with tools that enhance their learning experience. There is evidence from learning and psychology research that a simple one-way communication (i.e., lecture broadcast) is providing less of a learning experience for a student than actually attending an on-campus class would provide. Consequently, we have been engaged with DEN to provide collaboration tools that would make the learning experience more interactive. Our first project is a multiuser audio chat room system that is designed to allow groups of students to discuss assignments, allow teaching assistants to conduct lab sessions, and enable student questions and feedback during lectures.

The multiuser audio chat system involves numerous technical challenges that need to be addressed to build such an application. The number of participants in a chat session may be several dozens, with each student needing to hear and possibly talk to any other person in the session. Additionally, the end-to-end audio latency needs to be kept sufficiently low such that natural interaction is possible. Hence, we aim for our system to be scalable, practical (e.g., work with different types of network connections), integratable into the DEN infrastructure, and extensible with new features (e.g., speaker recognition).

In this study we report on the design and implementation of our AudioPeer chat room that is built on our overlay network multi-cast protocol called *YimaCast*. We have chosen a peer-to-peer (P2P) architecture to allow the chat room system to scale to a large user base without requiring massive resources on a central server system. We present design choices and implementation details of our system and provide some initial experimental results. The contributions of this report are in its description of multiple components that integrate into a fully working implementation. An initial field test of AudioPeer with a pilot class was conducted during the fall 2003 semester.

The rest of this report is organized as follows. Section 2 details the system design and components. Preliminary experimental results are presented in Section 3 while Section 4 surveys the related work. Finally, future extensions and enhancements are described in Section 5.

2 System Design and Components

Our system aims to provide an efficient audio chat application with low audio latency. One of the primary concerns is an efficient interconnection topology and architecture. An immediate first approach would be to connect each participant to a central server that merges incoming audio streams and then distributes the final mixed result to every listener that is connected. The advantage of such a star layout is that the sessions can be centrally managed and the
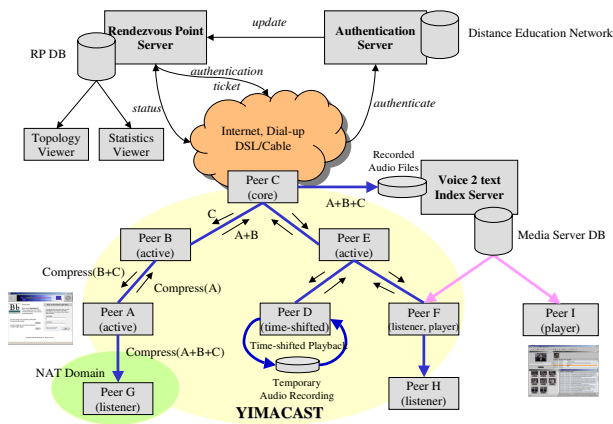
Figure 1. Overall architecture of AudioPeer with YimaCast.

delay for the sound streams to be relayed by the server depends mostly on the distance of the users from the server. The disadvantages of this architecture are that the central server requires a large amount of resources (for example memory and network bandwidth) that is proportional to the number of participants. Furthermore, the server can easily become a bottleneck and also is a single point of failure. Therefore, we adopted a distributed peer architecture where a newly joining user may be connecting to one of her peers who is already participating in an ongoing audio chat session. We call this low bandwidth consuming overlay network *YimaCast*.

One of the challenges with a distributed architecture is that the end-to-end audio latency may be more variable. From existing research we know that for an interactive conversation the delay from the microphone input through the transmission to the audio speaker output should not exceed 150 to 250 milliseconds for a natural conversation. Our audio chat system is designed to work within these limits and to dynamically adjust peer connectivity to optimize the audio transmission. Audio mixing is performed at each multicast member node to reduce the network bandwidth significantly. As a result, many simultaneous audio chat sessions can be supported. While YimaCast manages the multicast tree within a session, multi-session management is provided at the system's level. Our software suite consists of four components as shown in Figure 1: multiple *AudioPeers*, a *rendezvous point* (RP) *server*, an *authentication server*, and a *voice2text indexing service*.

AudioPeer denotes the integrated collection of modules that allow a user to attend audio conferences. It includes an embedded web browser, an application-level multicast connection manager, and an audio manager. From the embedded web browser the user can contact the authentication server and log onto the system. An interface to query and retrieve pre-recorded audio files is also provided. The connection manager handles the application-level connections with other remote AudioPeers through the YimaCast application-level shared multicast protocol.

The audio manager, described in Section 2.2, captures audio samples and plays them continuously. It also mixes incoming audio packets sent from neighboring AudioPeers connected through the YimaCast delivery path, subtracts the mixed stream, and forwards the result to the designated destination nodes.

An AudioPeer can take on one of five different functional roles: active peer, time-shifted peer, passive peer (or listener), recorder, and player. An active peer participates in online discussions. It requires low end-to-end latency with other active peers. A time-shifted peer may suspend the play-out of the current discussion temporarily. In the meantime, it automatically records the incoming audio packets into a file, which can be reproduced when the user requests a resume operation. To catch up with the current on-going session, it may skip some audio packets, such as silence. A listener is a passive user who mostly listens to the current discussion and speaks infrequently. It requires less tight delay bounds, enabling higher audio quality. Usually, active peers are located near the Yimacast core node, while listeners are attached to the leaves of the tree. A recorder, a special case of a listener, receives audio packets and stores them in an audio file. Finally, a player renders pre-recorded audio content stored on the voice2text indexing server.

The distance education network (DEN) at our university provides all registered students with electronically available educational services. The web-based *authentication server* maintains class information, registered user information, and recorded lecture materials. Thus, any user of our system is authenticated through the DEN system. After verifying a user, the server then forwards the login information to the rendezvous point server.

The *rendezvous point (RP) server* is the bootstrap node that enables an authenticated node to join ongoing sessions. To this end, it stores information about users, currently available sessions, and other peers. For administrative purposes it also provides a statistics viewer and an YimaCast topology visualizer. Once an AudioPeer is authenticated, it can freely join and leave sessions at any time without contacting the RP server because YimaCast allows decentralized tree migration by design.

The *voice2text indexing service* allows users to perform a keyword search and retrieve the matching audio fragments through a web interface. Functionally it consists of an audio recorder and an indexing server. The recorder is connected to one of the AudioPeers in every live session and stores the audio packets in a file. The indexing server then extracts keywords and associated audio fragments from these files (currently an off-line process). We plan to use three audio processing plug-ins: speech recognition, speaker identification, and audio classification. Speech recognition generates keyword indices by comparing speech in an audio signal to words in the speech recognition dictionary. Speaker identification identifies speakers in an audio signal by comparing voices in an audio sig-

nal to those of speakers in a speaker identification dictionary. Finally, audio classification recognizes specific audio types such as thunder, applause, or laughter, and stores time stamped information for classification based queries.

2.1 YimaCast Protocol

YimaCast is an application-level multicast protocol designed to serve as a reliable audio streaming platform that provides minimum overall end-to-end delay among all peer nodes. Aiming at high reliability and low latency, the YimaCast protocol dynamically maintains a shared multicast tree among all peer nodes. For space reasons we must restrict our presentation of the YimaCast protocol to a relatively high-level description.

2.1.1 Tree Construction and Maintenance

YimaCast builds and maintains the multicast tree by confining the operation of the AudioPeer client to the well-defined states shown in Figure 2.
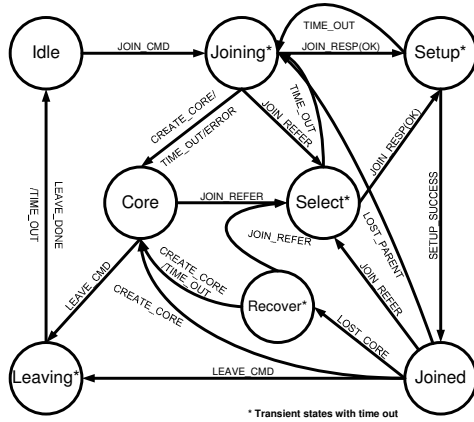
Figure 2. AudioPeer state diagram.

Tree construction is initiated with a connection to the RP server, which replies with information needed for the AudioPeer client to set up an audio connection with the exiting peer nodes. The bootstrap phase is the only time when the YimaCast protocol needs to contact a central server; after that, all tree maintenance is performed in a distributed manner. In order to achieve the minimum end-to-end delay, we must to solve the famous Steiner Tree problem, which is known to be $\mathcal{NP}$-complete [3]. Fortunately, since all the nodes in our tree are multicast members, the Steiner Tree problem is reduced to the minimum spanning tree (MST) problem, which can be solved in polynomial time. There exist several centralized and distributed algorithms to solve the MST problem, e.g., Prim's algorithm [5] and the approach proposed by Gallager et al. [7]. However, these algorithms need to rebuild the complete tree every time when there is a node join or leave operation. They are not suitable for our application which operates in a very dynamic environment. Hence, during the join process, YimaCast uses

a heuristic Shortest Path Tree (SPT) algorithm, which attaches the new node to the nearest known nodes in the multicast tree. The design is to achieve minimum tree delay cost while concurrently minimizing the service interruptions to the existing nodes when modifying the tree.

YimaCast is also designed to handle the dynamics of a distributed environment. At run time, various errors can occur, e.g., the failure to establish a connection with the parent node, the loss of a connection during operation, the loss of the core node, etc. By incorporating a multi-layered error handling policy, YimaCast can repair most of these errors without asking for help from the RP server. Our experiments also show that most of the errors are recoverable in this distributed manner.

2.1.2 QoS-based Tree Optimization

YimaCast addresses two QoS issues related to audio conferencing: the end-to-end delay and playback hiccups. The end-to-end delay denotes the latency between speakers and listeners and the goal is to minimize this delay. Playback hiccups are caused by the variable delivery time of packets over a standard TCP/IP network. Data buffering at the playback side can help to smooth out jitters, but at the price of increasing the end-to-end delay. To solve the conflicting goals between these two QoS issues, YimaCast introduces a dynamic tree optimization algorithm that depends on the participants' individual QoS demands for the audio chat room.

A user may be in either a listening or a speaking mode. Naturally, users who speak more frequently require shorter end-to-end latency because too much delay between speakers will make the conversation uncomfortable. At the same time, users who are mostly listening may tolerate a longer delay. In this case, the focus should be on minimizing the playback hiccups to obtain a better listening experience.

Dynamically reducing the end-to-end delay among speakers is achieved by clustering the speaker nodes. This is carried out by continuously monitoring the behavior of the user and – if a user speaks frequently – gradually migrating her toward the core. On the other hand, if a user keeps silent for extended times during a chat session, the client will increase the audio playback buffer to reduce audio hiccups. With this algorithm, active peers move closer to the core while passive peers are migrated to the edges of the tree, hence the QoS requirements of both groups are met. It is very noteworthy to mention here that the selection of the core node does not significantly affect the performance of the optimization. The reason is that the core node is only used as the generic reference direction in which active peers move together. Once clustered, the delay between the speakers is optimized. Therefore, the optimization result is relatively independent of the core position.
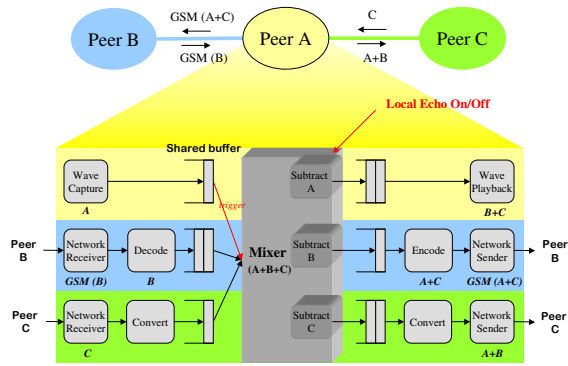
Figure 3. Audio mixer module.

2.2 Software-Based Audio Mixing

The audio mixing algorithm focuses on minimizing the network utilization for our audio conferencing application. Unlike in video conferencing, it is possible to aggregate the uncompressed audio sources through simple arithmetic calculations, preserving the original audio bandwidth.

Table 1 lists the currently supported audio media types and their characteristics: high-quality, low latency (PCM stereo); medium-quality, low latency (PCM mono); low-quality, low latency (GSM.610); and high-quality, high latency (MPEG-1 Layer 3). Among the audio codecs available, GSM.610 is reported to have a small compression delay and a tolerable audio quality. The high-quality high latency audio format is useful because some AudioPeers, participating as listeners or recorders may be connected via a low bandwidth network.

Each peer node is equipped with an audio mixing module that relays the incoming audio from remote nodes to the outgoing connections. Our design allows heterogeneous audio transmissions and each conference participant can create a specific quality of audio samples, such as PCM stereo sound and GSM mono sound. We use a software-based audio mixing algorithm called *decode-mix-encode* [12]. A linear mixing algorithm requires all the input audio bitstreams to be uncompressed for simple arithmetic additions and subtractions. Thus, all incoming encoded bitstreams are decoded into their uncompressed form, and the resulting uncompressed bitstreams are merged to a mixed bitstream. This stream is later used when constructing the outgoing streams for each respective remote nodes.

To illustrate, consider an example with three peer nodes, *A*, *B*, and *C*, connected as shown in Figure 3 (*A* is connected to *B* and *C*). Peer *A* locally captures uncompressed audio samples with a bandwidth of 1.5 Mbps; two unidirectional links between *A* and *B* are constructed to transfer compressed audio packets, say, as 13 Kbps GSM streams. The links between *A* and *C* use uncompressed PCM audio transmissions at 64 Kbps. *A* receives the audio packets β and γ respectively from *B* and *C*. It also generates newly captured audio packets α periodically. With this information the audio mixer performs the following steps.

1. Transcode incoming audio packets into linearly uncompressed audio packets. Uncompressed audio is stored in memory for future subtraction.
2. Mix all uncompressed incoming packets by adding arithmetically, $\phi = (\alpha + \beta + \gamma)$.
3. Subtract the original audio from the mixed stream. For local playback, subtract the original audio packets α from ϕ, resulting in $(\beta + \gamma)$. For *B*, subtract β from ϕ, resulting in $(\alpha + \gamma)$. For *C*, subtract γ from ϕ, resulting in $(\alpha + \beta)$.
4. Encode the subtracted audio bitstreams in the form of the network connection supported audio format. The local playback module, *A*, does not need any encoding. The outgoing audio bitstream, $(\alpha + \gamma)$, for *B*, needs encoding from a 1.5 Mbps PCM bitstream to a 13 Kbps GSM bitstream. For *C*, $(\alpha + \beta)$ needs transcoding from a 1.5 Mbps PCM bitstream to a 64 Kbps PCM bitstream.
5. Packetize the audio bitstreams and send the packets to the remote peer nodes, respectively.

Simple addition, while preserving the volume level throughout a chat session, may cause integer overflows when multiple talkspurts are added simultaneously. Another approach, dividing the original talkspurt by the number of participants or by the number of active talkspurts before adding it to the mixed stream, prevents this overrun problem. However, it requires intelligent floor control mechanism to detect the active talkspurts and generates additional exchange overhead of control messages.

Our implementation uses an augmented version of the simple addition algorithm. It detects overruns before the addition step and lowers the volume level of the audio sources. The mixer is implemented as a single thread with real-time priority. It is blocked until the local capture module sends a signal to wake it up. Immediately, it collects the uncompressed incoming audio samples, aggregates them and subtracts the original data. This mechanism guarantees a continuous hiccup-free audio transmission to the remote nodes. One slight drawback is that it may increase overall end-to-end delay because of queuing delays at the mixing module.

3 Experimental Evaluation

We present some preliminary results from experiments measuring the end-to-end audio delay between two AudioPeer clients in a LAN/WAN environment and identify the primary causes of any delays. We performed our experiments on a Windows platform. Note that all results are subject to improvements due to changes in audio I/O drivers, operating system support, and specialized hardware.

The end-to-end audio delay can be modeled as the summation of the capture interval, the play-out delay, and any network latency. With our experiments, we identified the two primary components of the end-to-end audio delay: the capture interval and minimum play-out delay. The

Supported audio media types				
Compression Type	uncompressed		compressed	
Audio Format	PCM	PCM	GSM.610	MPEG Layer3
Bits per Sample	8	16	8	16
Channels	mono	stereo	mono	stereo
Sampling Rate	8 KHz/16 KHz	48 KHz	8 KHz	48 KHz
Delivery Rate	64 Kbps/128 Kbps	1.536 Mbps	13 Kbps	56 Kbps
Usage Method	LAN	LAN	dial-up modem	cable modem, DSL
Latency Requirement	low	low	low	high
Audio Quality	medium	high	low	high

Table 1. Audio types supported.
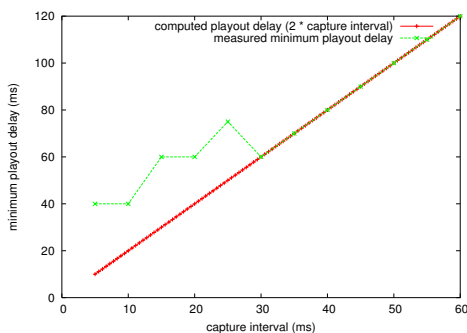
Figure 4. Minimum play-out delay per capture interval.

Figure 5. Measured one-hop end-to-end delay in a LAN environment.

capture interval, in milli-seconds (ms), is the time period between successive callback functions to collect the audio samples from the audio driver. If the capture interval is 20 ms, a callback function is invoked every 20 ms to transmit the captured audio samples from the driver to the application. The *minimum play-out delay*, also represented in milliseconds, is the time used to pre-load the audio samples for smooth audio play-out and to compensate for network jitters and slightly irregular capture intervals.

Experimental setup. The Windows MME (MultiMedia Extension) API was used for waveform capture and playback. To precisely measure the end-to-end audio delay, we used an audio split cable. The two inputs of the cable were connected to the original audio source and the receiving AudioPeer. The output was recorded on another machine and the maximum delay offset between the two inputs was computed using cross-correlation in MATLAB. We repeated this experiment ten times with the same configuration to reduce the statistical errors.

Figure 4 shows the minimally required play-out delay as a function of the capture interval with no audio dropouts. We set the play-out delay as a multiple of the capture interval. We observe that (1) the minimum play-out delay increases as the capture interval increases and (2) the CPU is more heavily loaded as the capture interval decreases. If the capture interval rises above 30 ms, the play-out delay linearly increases by a factor of two. However, the minimum play-out delay levels off at 40 ms.

Figure 5 shows the measured end-to-end latency when the capture interval is 10 ms with varying play-out delays.

In a LAN environment the network transmission delay was less than 1 ms and the capture interval and play-out delay would be expected to dominate the end-to-end delay. Our expectation is confirmed in Figure 5 where the play-out delay is greater than or equal to 70 ms. The small distance between the two curves is caused by the network delay and system overhead. For play-out delays in the range of 40 - 60 ms an additional 10 ms delay is introduced whose cause still under investigation.

From these two figures we find that the optimal capture interval and the play-out delay are 10 ms and 40 ms, respectively. Thus, the minimum one-hop audio delay is 60 ms in a LAN environment. Accordingly, any multi-hop end-to-end audio delay can be roughly represented as *minimum one-hop audio delay (= 60 ms) + end-to-end network delay*. Note that placing mixing modules at intermediate relay nodes may add additional processing delays.

Next, we also measured the end-to-end audio delay in a WAN environment between the USC campus in Los Angeles and Information Sciences Institute in Arlington, Virginia. We placed a loop-back module at the east coast site to receive audio packets and reflect them back to the sender. The average round trip time (RTT) was measured at 70 ms with a small variance. The end-to-end audio delay in the WAN environment exactly matches that of the LAN environment plus the one-way network transmission delay.

These encouraging results show the feasibility of a medium-sized application-level audio chat service in a commodity Windows environment. Similar results have also been confirmed by other research groups [9, 10].

4 Related Work

Our system integrates the following components to create an efficient, easy-to-use, scalable live audio chat service: web-based session management, shared tree based application-level multicast delivery, live recording and retrieval, speech-to-text translation service, and audio mixing. There exists related work in the audio mixing and multicast area, but to the best of our knowledge no system with an integrated feature set like ours has been reported.

Hierarchical audio mixing architectures [13] place all participants at the leaf nodes and locate the mixers at non-leaf nodes. A mixer relays the audio coming from its children to the parent after merging the streams. A root mixer broadcasts the mixed audio to all participants through the IP multicast network. Several commercial products such as the *ClickToMeet* conference product [1], formerly CUseeMe, construct an efficient delivery path (full-mesh or multicast) among the mixers. Because most of the commercial products are integrated with video conferencing applications, they focus on video delivery and cannot fully take advantage of audio mixing. In end-system mixing architectures [12], one of the participants functions as the mixer. Additionally, most of the above cited systems do not consider how to build efficient delivery paths among mixers and participants.

Application-level multicast protocols providing many-to-many connectivity can be classified as mesh based, tree based, cluster based and distributed hash table (DHT) based architectures. An example of a mesh based architecture is Narada [8], where each node constructs a single-source multicast tree from the mesh structure. Due to its centralized nature, Narada does not scale well. Tree based architectures include Yoid [6], HMTP [14] and our proposed YimaCast. NICE [2] was developed as a hierarchical architecture that combines nodes into clusters, then selects representative parents among these clusters to form the next higher level of clusters, and so on. DHT based architectures use hashing mechanisms to generate node identifiers such that nodes close to each other logically have similar node identifiers. Subsequently a multicast tree is created on top of this substrate. Pastry/Scribe [4] and CAN/CAN Multicast [11] are based on this concept. Even though these systems are designed to support a large number of users, all the previous application-level multicast protocols fail to recognize the potential for constructing a more efficient topology if the aggregation of audio packets along the delivery path is allowed.

5 Conclusions and Future Directions

We have presented our design and implementation of a multiuser audio chat system using the peer-to-peer based YimaCast protocol. Preliminary results are encouraging and we are planning further real world tests and deployment in the coming months. We also intend to extend our work in several directions.

First, silence detection can improve the audio mixing process by removing the noise floor that accompanies with every open microphone. Second, our adaptive QoS algorithm that migrates nodes based on their active speaker status requires further investigation and performance tests.

References

[1] Click to meet, http://www.fvc.com/.

[2] S. Banerjee, B. Bhattacharjee, and C. Kommareddy. Scalable application layer multicast, 2002.

[3] M. R. Carey and D. S. Johnson. Computers and intractability. *W.H. Freeman Co. New York*, 1979.

[4] M. Castro, P. Druschel, A. Kermarrec, and A. Rowstron. SCRIBE: A large-scale and decentralized application-level multicast infrastructure. *IEEE Journal on Selected Areas in communications (JSAC)*, 2002.

[5] T. Cormen, C. Leiserson, and R. Rivest. Introduction to algorithms. *MIT Press*, 1997.

[6] P. Francis. Yoid: Your own internet distribution.

[7] R. Gallager, P. Humblet, and P.Spira. A distributed algorithm for minimum-weight spanning trees. *ACM Transactions on Programming Languages and Systems*, pages 66–77, January 1983.

[8] Y. hua Chu, S. G. Rao, S. Seshan, and H. Zhang. Enabling conferencing applications on the internet using an overlay multicast architecture. In *ACM SIGCOMM 2001*, San Diago, CA, Aug. 2001. ACM.

[9] K. Koguchi, W. Jiang, and H. Schulzrinne. QoS measurement of VoIP end-points. In *IEICE Group meeting on Network Systems*, December 2002.

[10] K. MacMillan, M. Droettboom, and I. Fujinaga. Audio Latency Measurements of Desktop Operating Systems. In *International Computer Music Conference (ICMC'01)*, September 2001.

[11] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker. Application-level multicast using content-addressable networks. In *Third International Workshop Networked Group Comm.*, November 2001.

[12] K. Singh, G. Nair, and H. Schulzrinne. Centralized Conferencing using SIP. In *Internet Telephony Workshop*, April 2001.

[13] H. Vin, P. Rangan, and S. Ramanathan. Hierarchical conferencing architectures for inter-group multimedia collaboration, 1991.

[14] B. Zhang, S. Jamin, and L. Zhang. Host Multicast: A Framework for Delivering Multicast to End Users. In *Proceedings of IEEE Infocom*, New York, June 23-27, 2002.

Design of Multimedia Instant Messaging System with Annotation

Zon-Yin Shae and Xiping Wang

IBM T.J. Watson Research Center NY
{zshae, xiping}@us.ibm.com

Ferdinand Hendriks

Hitachi Global Storage Technology
hendriksf@charter.net

Abstract

We have investigated enhancements to Internet based instant messaging (IM) system. Our system allows communication via handwritten information, mixed with other non-textual multimedia information. All information is presented to the user in a graphical user interface using a scrollable whiteboard metaphor. A compact timeline view, allows random access to recorded multimedia messages. Our system allows annotation of items entered anywhere in the IM record, as well as deletion and modification. All users are alerted to annotations by means of two hyperlinks; one at the end of the IM record, the other at the annotation itself. The system is especially useful with the PDAs and Tablet PCs where handwriting input is currently supported. A prototyped IM system in iPAQ PDA with handwriting (ink) instant messaging and annotation will be presented.

1. Introduction

Instant messaging has its roots in well-known Unix commands such as "who" and "talk," which allow users to see who is available as a communications partner, and to communicate nearly instantly. It is currently used more and more as a replacement for e-mail, and as a "control channel" among remote users running another collaborative application. An early protocol developed for instant collaboration is Internet Relay Chat [1]. Popular IM services are AIM (AOL Instant Messenger), ICQ ("I seek you" AOL), Microsoft Instant Messenger (MSN), IBM Lotus SameTime, and many others. IM can also be embedded in web-based applications to allow real-time access to people of similar interests. As such it can promote website "stickiness." See e.g. the Java-based Bantu client [6] and rely on the web application for services such as login and authentication. Despite the incompatibility of different proprietary IM protocols, IM is enjoying very robust growth. The growth would be accelerated into the enterprise environment with the development of SIP SIMPLE standard IM protocol. A typical IM session chat record, sometimes called history, would include a sequence of lines of text.

IM is currently available on many personal digital assistants (PDAs), such as Palm, Compaq IPAQ, and others. PDAs are not large enough to have physical keyboards, and their display screens are quite limited compared to those of desktop and laptop computers. The preferred mode of data input is handwriting with a stylus, or selecting individual characters with a stylus, using a virtual or software-defined keyboard, which is somewhat slow and cumbersome, although shortcuts and predefined phrases may help to speed up frequently used tasks. A stylus produces stroke information. Sometimes it is referred to as electronic ink, or simply ink. Stroke input is, at the minimum, stored as an ordered set of quantized pen locations, but may include many more attributes such as the time at which each location was visited, the pen pressure and pen inclination, user-id of the pen, user-id of the writer and others. Despite the fact that ink is arguably the oldest method of data entry, standardization efforts have lagged. There is renewed interest in standardization by the w3C [2] consortium, owed in part to the increasing importance of handheld devices such as the Palm®, Compaq iPAQ®, Handspring Visor® and others. Ink input is entered in a very natural way on PDAs and tablet PC, while entering text is more awkward. To be sure, other modalities such as speech and gesture also solve some of the problems associated with the lack of a keyboard, but due to a lack of processing power, the predominant data type for input is still text. Thus, the current state of IM lacks the expressivity that stroke input can provide. Furthermore, the record of the IM session is static, and does not allow a user to draw attention to particular messages, let alone add to them, or change them in any way, for all IM participants to see. It is also difficult to navigate among IM entries. This paper presents an ink instant messaging system to solve the above-mentioned problems.

The paper is organized as follows. Section 2 describes the system basic functions. Section 3 describes the scrolling whiteboard metaphor. Section 4 describes predefined form ink message communication. Section 5 describes timeline user interface. Section 6 describes the hyperlinks mechanism. An ink instant messaging and annotation IM system prototype is presented in section 7, and a summary is given in section 8.

2. System Description

The client application window shown in Fig. 1 is divided into the following areas, starting at the bottom of Fig.1:

1.5 Private ink input field. Used for local previewing.
1.4 Awareness field showing registered users and their status.
1.3 Scrollable chooser for input modality: Text, ink, annotation, other multimedia objects.
1.2 Text input field
1.7 Send button.   Causes input to be sent to all participants in the session.
1.8 Clear button. Clears the input area.
1.9 Annotation; written directly into the recording field.
1.1 Public recording field containing text, ink and annotation.
1.6 Scroll bar for the recording field.
1.10 Search button
1.11 Awareness icons that visually express the status of participants.
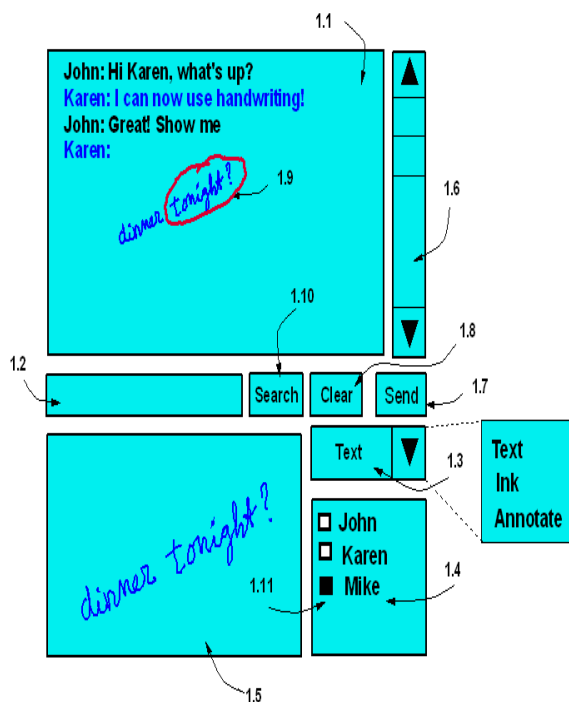
Figure 1. Schematic diagram of GUI

Field 1.3 is the modality selector, with which a user selects the type of instant message to be entered. Three examples are shown: text, ink and annotate. Other modalities can also be supported, in particular audio and video objects. In effect, the modality selector explicitly classifies the type of data entered in the IM record. The

classification is a form of metadata associated with the object. This is useful for later browsing and retrieval of objects in the IM record. The awareness field 1.4 is a standard component of most IM systems. Fig. 1 shows a minimum version of such as field, containing the names of those logged in to the same IM session, together with awareness icons 1.11 which have meanings such as "I am logged in," "I am away," "Do not disturb," etc. In addition this field may contain a timestamp. Field 1.2 contains the input area for text based input. Clicking in this field is equivalent to selecting Text in the modality selector 1.3. Field 1.5 buffers stroke input and other IM objects for input and allows preview. Objects in this field are subject to geometric transformation when displayed in the recording field. This allows for a more compact IM objects that are not text. The "Clear" button 1.8 clears the last entered input object. The "Send" button causes the input to be sent to the server for distribution among session participants and storage in the IM record.

In addition, a user can annotate the recorded IM session with ink, which is in general of a different color compared with the normal ink input (ink entered in the ink input area). Annotations can also be general multimedia objects. The scrollable chooser 1.3 is optional. The intent of the user to choose ink, text or annotation can be expressed by clicking in the private ink input field 1.5, the text input field 1.2 or the annotatable recording field 1.1, respectively. The "Search" button causes the input to be used as key object for content- based query. The content-based query for the ink media is currently a active research topic [4]. The emerging MPEG-7 [5] standard provides for a systematic way to describe the multimedia content of IM sessions so that browsing and querying become much more efficient. Stylus locations are reported in binary format, uncompressed or compressed. Compression of handwritten strokes is known in the art. However, progressive, scalable compression of strokes is an active area of investigation [3]. Progressive transmission of stroke information allows a user to gain an early visual impression of the strokes, while refinements are sent later.

3. Scrolling Whiteboard Metaphor

The recording field presents a scroll metaphor to the user such that the entire scroll is accessible to users as if it were a whiteboard of continuously increasing length as a session progresses. Annotations entered in the recording field are public and distributed to all users. It will be understood that besides text and ink, other media types may be inserted into the recording area, such as images,

video and audio files. The public recording field 1.1 provides a viewport of a generally very long logical recording strip. The recording area represents the contents of all that has been entered into the session. New content is added at the bottom. The graphical user interface gives the user the impression that he/she is looking at part of a very elongated strip of constant width through a viewport. It is clear that the restriction of constant width can easily be removed and a horizontal slider added to the interface so that a user can move left-right in the recording display. However, in most cases a fixed horizontal width will suffice.

4. Filling in of predefined forms

Quite a useful application in collaborative messaging involves the filling in of predefined forms with text and/or electronic ink. In this case, the recording field holds the predefined forms upon session entry. Filling in the forms proceeds as follows:

1. Select the field to be filled in by clicking on the field of interest. This implies a segmentation of the predefined form each having its own specific meaning. The application program detects that a field was selected that awaits user input. For example, a signature.
2. Write the information to be filled in on the form by writing in the user input area for electronic ink, or by typing the information into the text input field.
3. Press the Send button. Repeat steps 1,2,3 upon error.
4. The current information (ink or text) will be overwritten.

As before, electronic ink input will be scaled and repositioned. But now the scaling and repositioning is such that it fits the user input areas of the predefined forms. An example of a predefined form IM client is shown in Fig. 2.

A view of the predefined form 2.1 is presented to the user. This form may have a field with special meaning such as a field 2.2 that can be filled in by the user. After having clicked on field 2.2, the user creates the input 2.3 to be entered into 2.2. The system transforms the input to fit in the field 2.2. The bounding box of 2.3 shall be transformed so as to fit the field 2.2. This transformation can be chosen such that the input maintains aspect ratio. The latter implies uniform scaling in width and height, such that, after transformation, the width does not exceed the width of the field 2.2, and the height does not exceed the height of the field 2.2. Alternatively, the

transformation can be chosen such that the four corresponding corners of the field 2.2 and 2.3 are mapped onto each other.
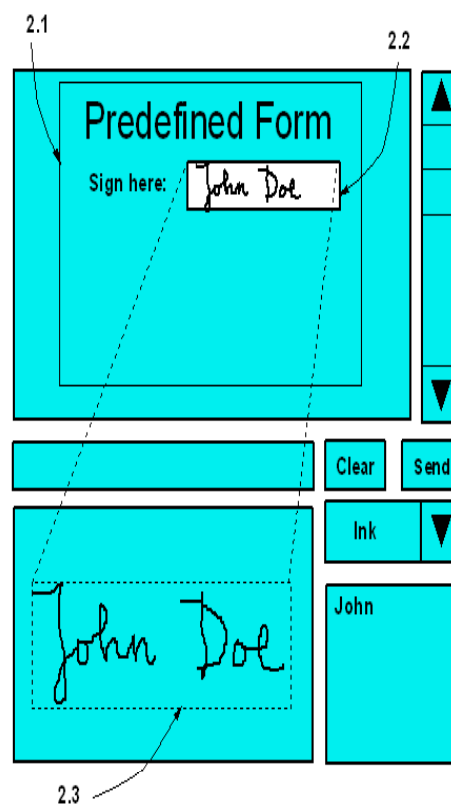
Figure 2. Predefined form application

5. Timeline

In order to facilitate searching the recorded IM session, features such as skip-to-beginning/end and search by content may be part of the user interface. Fig. 3 shows a variation of the client graphical user interface. Some of the public recording area has been reserved to implement a "layered time line." The frame shows the local user's text as a mark 4.1, text from others as mark 4.3 and annotations (from anyone) as mark 4.2. Clicking on a mark causes the recording frame to show a section of the record corresponding to that mark (either ending at the mark, or beginning at the mark). The frame shows a timeline broken in sections, such that each section forms a layer. The intent is that frame 4.4 represents the whole instant messaging session. Distance along each layer represents an elapsed amount of time. Preferably the time is measured linearly and uniformly along each layer. However, especially when the session is particularly long, the distance may be measured non-linearly along each layer, for example, exponentially. This allows us to greatly increase the information

content of the timeline frame (TF) 4.4, but at the expense of readability of the oldest items. The timeline frame allows the user random access to the recorded material, while the scrollbar allows sequential access. It will be understood that the marks of the timeline could themselves be icons that are specific for the modality of the input entered.
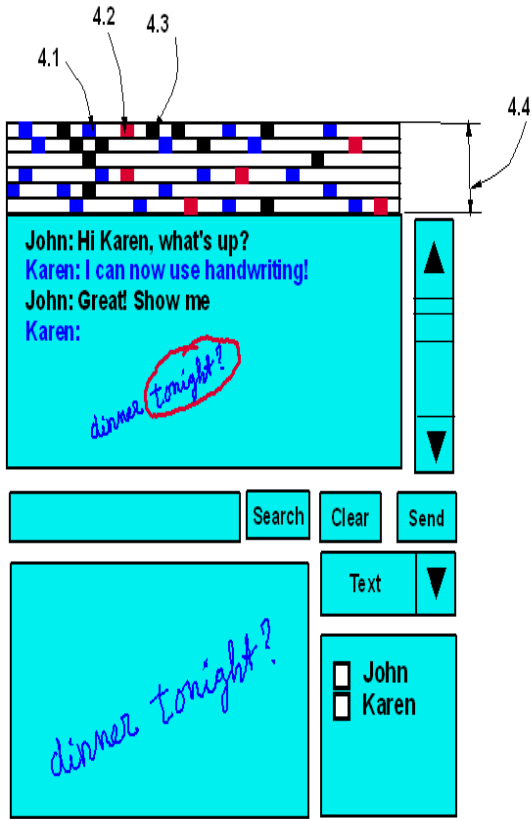
Figure 3. Graphical user interface consists a timeline field.

6. Hyperlinks

All IM messages are objects (that is the object-oriented description of an abstract or concrete entity having attributes, methods and so on). In many cases, it will be necessary to store pointers such as a URL to the actual data describing the object. The present multimedia IM system allows a user to make stroke based annotations anywhere in the recording field. An annotation may or may not be associated with an earlier message. If an association is intended, the user is required to first select the message to be annotated, for example by clicking on it. For example, item 5.4 in Fig. 4 is to be annotated by Bob. Bob clicks on 5.4. indicating that the house will be

the target for his annotation. He then enters the annotation 5.3. Message object 5.4 is now linked to (i.e. associated with) annotation 5.3. Message 5.4 is visible to the user as a bitmap of a house, but the message object may contain many attributes, such as cost, age, number of rooms and so on. In addition to the annotation message object 5.3 a "back link" 5.7 is added to annotation 5.3. The function of this "back link" will become clear later in this section with an application scenario. To facilitate navigation in the recording field, entering an annotation causes an entry at the end of the IM record. This entry is a hyperlink to the actual annotation, and is a message object in its own right. When clicked, it makes other users aware of the fact that another user made an annotation somewhere in the public record. Without this feature, users other than the one making the annotation would not be alerted to annotations. Because a user can scroll to any place in the recording field, what each user sees is not necessarily always the most recent part of the IM session. In addition to the hyperlink in the recording field, the hyperlink 5.1 is also entered as a symbol at the end of the TF 4.4. The actual annotation 5.3 also appears in the TF. The annotation can be associated with an earlier entry 5.4.
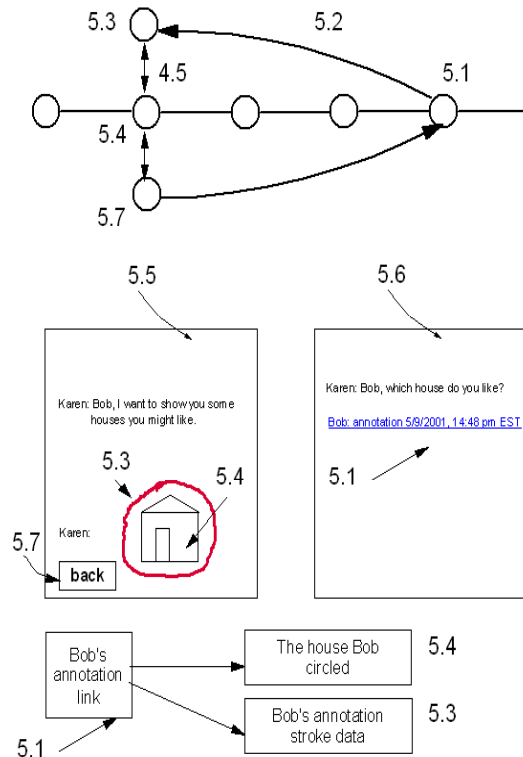
Figure 4. Annotation with hyperlinks, allowing quick jump to and back from an annotation

Fig. 4 illustrates a scenario in which a hyperlink is used to choose a picture of a house 5.4. User Karen has entered several real estate listings on her PDA and has shown them to Bob. She then prompts Bob for the house that he likes best. Bob indicates his preference by scrolling to the house he likes stopping at view 5.5 and circling the house (stroke-based annotation). Karen, if viewing the current view of the recording field immediately sees the hyperlink 5.1 on her screen. If she happens to be at a different view, she is still alerted via the new entry in the TF, which may flash to draw attention to it. To facilitate jumping back to the location in the record from which the annotation was reached, usually the end of the record, a "back" button 5.7 is added to the annotation 5.3. The bottom part of Figure 3 shows the Bob's annotation hyperlink consists of pointer to the annotated object 5.4 and the ink data 5.3. The horizontal line in the upper part of Figure 3 presents the time line of the IM session, the small circles present the objects have been entered in the recording filed in the sessions. Object 5.3 (ink annotation data) and 5.7 (back link) will be created after the annotation of object 5.4. After the "Send" button is clicked object 5.1 (the hyperlink) will be sent and show to the all participants. Every participant can click object 5.1 to look at the annotation object 5.3, and jumping back from there to object 5.1 by clicking object 5.7.
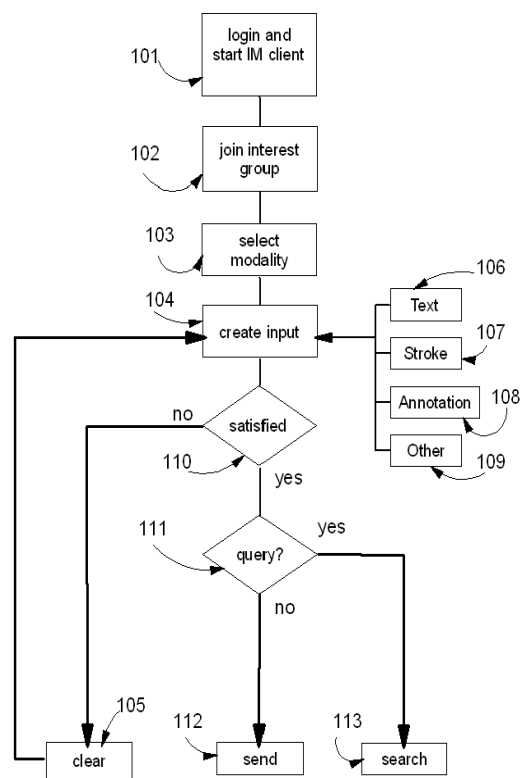
Figure 5. Execution steps at client

7. Ink Instant Messaging Prototype

We have implemented the proposed Ink instant messaging system on iPAQ. The IBM Lotus SameTime is used as the central controller for distribution and management of messages. The client is implemented in pure Java using the SameTime SDK for cross platform portability. The SameTime SDK provides programming interfaces with SameTime server. All of the participants must first log into the same meeting place through SameTime's community service. After authorized, each participant can then communicate with all of other participants or have a private conversation with a desired participant. The communications among meeting participants is controlled and maintained by the SameTime's places service. In order to achieve a natural handwriting speed with the stylus on iPAQ, a special care must be taken on the stroke input. In our implementation, a simple decimation algorithm is used for simplifying the input strokes while maintaining a reasonable accuracy of strokes. The simplified strokes then need to be compressed before transmission for efficiency (details of streaming mechanism had been reported in [3]).

Fig. 5 shows the steps executed at the client. Block 101 starts off the IM client. It includes login and authentication by the server. The user joins an instant messaging topic of interest in 102. If a session for that interest group is in progress, the user is presented with the most recent message or messages. In addition, the messages may contain predefined message objects such as forms that can be filled out. The user is now free to contribute to the IM session in various modalities, selectable in block 103. Modalities include text 106, stroke 107, annotation 108 and other multimedia objects, 109, such as audio and video. Block 104 represents the actions of the user that create IM input in any of the modalities selected in 103. Input can also be inserted via cut and paste from other applications. Deleting the input is supported by clear block 105. Block 110 is a decision block in which the user selects his satisfaction or dissatisfaction with the input which is at this stage still buffered at the client. The user can signal whether the input is to be interpreted as a query 111 using the input as key object. If the input represents a query the input is sent to the server for searching 113, otherwise the input is sent to the server to be stored in the IM record, as represented by block 112. The server

redistributes the input to the networked session participants. The IM record consists of IM objects. The IM objects are ordered temporally, except for annotations. The IM objects are rendered in the recording field to be viewed by the user. The current implementation does not have the search capacity. Ink content retrieval by ink strokes is an on going research issue. Authors have a joint effort with CMU to investigate this issue further [4]. We also have made use of the MPEG4 scene graph [7] to successfully archive and replay the ink strokes synchronously with the voice content. This is especially useful to reproduce the complete IM discussion session in exactly the time sequences.

We have asked different people to evaluate the system and found out that the proposed system is very suitable for PDAs and the speed of the ink messaging input on iPAQs is adequate for tracking natural human handwriting. Since handwriting Ink message is a natural choice for devices using stylus input, the Ink IM is much more desired than text-based IM. Especially the annotation feature and scrollable white board design are very attractive for discussion and history review.

8. Summary

Instant messaging (IM) is often used "when e-mail isn't quick enough" or "spam e-mail is too much" for short text-based notifications and responses. IM is currently touted as the Internet's latest "killer app." This paper investigates the design of multimedia instant messaging system that extends the text instant messaging into multimedia instant messaging. Especially ink instant messaging with annotation and whiteboard metaphor enriches the effectiveness of the instant messaging in the PDA and Tablet PC devices. The scrolling whiteboard metaphor mechanism and timeline as well as hyperlinks user interface has been presented and an IM system using iPAQ PDA has been prototyped and evaluated. Users of this iPAQ IM system have expressed a quite favorable experience doing IM by ink writing. This system would also be very suitable in the Tablet PC. Inter-operability among various commercial IM systems can be achieved by SIP SIMPLE protocol and W3C ink data format standardization.

9. References

[1] J. Oikannen et al. "Internet Relay Chat Protocol," RFC 1459, May 1993.
[2] www.w3c.org/TR/InkML inkXML group and document
[3] Zon-yin Shae et al. "Multi-Resolution and Streaming of Whiteboard Stroke Media" IEEE International Symposium on Multimedia Software Engineering 2000
[4] Howard W. Leung, "Representations, feature extraction, matching and relevance feedback for sketch retrieval", PhD Dissertation, Carnegie Mellon University, department of electrical and computer engineering, June, 2003.
[5] ISO/IEC JTC 1/SC 29/WG 11, 15938-5,"Multimedia Content Description Interface – Part 5 Multimedia Description Schemes" 2002.
[6] www.bantu.com   Bantu enterprise instant messaging
[7] ISO/IETC JTC 1/SC 29/WG 11, 14996-1 MPEG4 systems. 2002.

Biometrics Identification Based on Temporal Templates of Visual Hand Movements

Sanjay Kumar, Dinesh K Kumar, Arun Sharma, and Neil McLachlan
School of Electrical and Computer Engineering
RMIT University, GPO Box 2476V Melbourne, Victoria, Australia 3001
Phone:00-61-3-99253025   Fax-00-61-3-99252007
E-mail: **s2003383@student.rmit.edu.au**

Abstract *This work presents a novel technique of identification of authentic users based on the visual hand actions. The technique uses view-based approach for representation of hand movements, and uses a cumulative image-difference technique where the time between the sequences of images is implicitly captured in the representation of action to generate temporal history templates (THTs) of the hand movement. THTs of different individuals present distinctive 2-D motion patterns. The scale, translation and rotation invariant features have been used for discrimination of the THT for identification. The work explores the identification performance of the technique by the use of scale, rotation, and translation invariant features Hu moments. The recognition criterion is established using K-NN nearest neighbor technique using Mahalanobis distance. The accuracy of accepting an enrolled subject (AAES %) and accuracy of rejecting an imposter (ARI %) are the indicators of identification performance of the technique. The preliminary experimental results from different individuals indicate that: the THT based technique achieves high identification rate when subject specific movements are assigned to the subjects during enrolment.*

Index Terms—Biometric, Identification technique, and computer vision, Hu-moments

1.   INTRODUCTION

In our global information society, there is an ever-growing need to authenticate individuals. With the rapid progress made in electronics, e-commerce and with increased emphasis on security, there is a growing need for overcoming the limitations of the traditional identification and authentication technologies.

Biometrics-based authentication using computer vision technologies is emerging as a reliable method that can overcome some of the limitations of the traditional automatic personal identification technologies. Any human physiological or behavioural trait or knowledge can serve as a biometric characteristic as long as it is universal, distinctive, sufficiently invariant with respect to matching criterion, and these characteristic should be physically and preferably non-invasively measurable [1]. All traditional biometrics measures have certain limitations associated to them e.g. DNA can't be used in certain applications due to issues of contamination, sensitivity, cumberness and privacy; ear dimensions as a biometrics measure has a problem of non unique features; facial biometrics has got problems with aging, face disguise and variable imaging conditions; hand and finger geometry has limited applications and has limitations that these can be copied by the intruders, and although fingerprints are very unique but they also have the problem of fake fingers, storage and imaging conditions problems; iris biometrics is difficult non intrusive and requires co-operation from the individual during enrolment and identification; speech biometrics has the limitation of mechanical variance due to microphone and dependence on subjects' health [1]. Also the fingerprints, facial features, DNA, retinal features are known to be the most common biometrics based on the physiological features that cannot be altered by the authentic user if they so desire. There are a number of publications describing methods using these features as the biometric characteristics.

Biometrics that are related to some activities of the user are also useable. Biometrics such as keystroke analysis and gait analysis are based on the behaviour of the individual [1]. However, not all behavioural biometrics has been examined extensively. One area that has been given light treatment so far is use of hand gestures (movements) as the biometric.

Hand is a dexterous part of the human body and is unique to the gesturer [2], but very little attempt has been made to use the hand gesture as the biometrics. The use of the spatio-temporal template of walking person has been widely used for gait analysis for person identification. In a similar way spatio-temporal template of hand movement (gesture) can be used for person identification. The hand gesture has been extensively used for developing Human Computer Interaction (HCI) applications and similar application. The use of spatiotemporal templates of hand movement as the biometric will open a new area of research with excellent research potential.

This paper reports the development of a new biometrics method based on the hand gestures of the individuals. Based on our earlier research temporal history templates (THTs) generated from the video recordings of the hand movements (gestures) have been used for this purpose [21]. This research uses predefined hand gestures (movements) for the motion-based recognition for person identification. The invariant image moments of the spatio temporal templates of the hand gestures are proposed as the criterion for recognition of the individual are very distinctive to the gesturer [2]. The proposed technique has the advantage of being non-intrusiveness, unique and under the control of the individual without leaving a paper trace.

2.   RELATED WORK

There are two main approaches in the literature about the automatic gait recognition [3]. The first method is based on the model based gait recognition, where a mathematical model describes the subject's movement and the second method applies the statistical description to the set of images. The second approach detects the temporal changes in gait by using optical flow techniques [4] [5].Another automatic gait recognition work, which uses the statistical technique [6] describes the use of statistical approach to the automatic gait recognition is very similar to this work in terms of the features generated from the gait, the work has taken the moments of optical flow and claimed very promising results. Another similar work reported in the literature, which describes motion more closely than just pure statistical technique without implicit knowledge of gait uses velocity moments instead of just pure traditional moments[7]. This method has shown encouraging results on a small database of four subjects and is also immune to noise. In the related work of identifying people by the use of behavioural biometrics gait distinguished people from their walking by extraction of video sequences from their walking patterns [8]. Cunado et al. has also reported similar work that also considers gait as the behaviour biometrics for the purpose of identification [9]. Little and Boyd used frequency and phase features from optical flow information to recognize people from their gait [6].

This work is based on the motion based template research. The motivation of this work comes from the real-time interactive applications developed by Davis and Bobick [10] [11] [12] [13] [14] [15] where they presented a real-time computer vision approach for representing and recognizing common human movements from low-resolution image sequences for its successful development of an application named as "Virtual Aerobic Instructor" [14] . This work is similar to the work done by Bobick and Davis to develop a Virtual Aerobic Instructor for aerobics exercises and Kids Room an interactive room where children can play with the monster in an interactive environment[14].

3.   IDENTIFICATION TECHNIQUE

The philosophy behind the approach for person identification is based on the spatio-temporal templates of hand movements for enrolment and identification. This "THT" is a single static image integrated over time, is very distinctive to the hand, which performs it and is considered to be spatio-temporal hand biometrics. In the enrolment process hand gestures corresponding to the different individual are captured. From the enrolled hand gestures the Temporal History Templates corresponding to the different individuals are generated and stored in a database. From the THT of the various hand gestures global shape descriptors are extracted corresponding to each hand movement. During the identification the enrolled user is asked to perform the test hand gesture, from the test hand gesture THT is generated and features are extracted to be compared with the pre-stored features

of the various THT's of different subjects using K-NN (K-Nearest Neighbour) classifier, "Mahalanobis distance".

3.1.   TEMPORAL HISTORY TEMPLATES

This research is to test the efficacy of THT based method for identification on the basis of the Temporal History Templates (THTs) of hand movements. The representation of THT is based on a view-based approach of hand movement representation, where movement is defined as the motion of the hand over time. The technique is based on collapsing the hand motion over time to generate a static image from the image sequence. This resulting static image can represent the whole sequence of hand movement. This single static image also gives all the properties (shape, direction, where & how) the motion is taking place in the image sequence. This technique is very suitable for short duration, non-repetitive, medium velocity movements making very much suitable for real-time biometric application [16].

3.1.1.   MOTION IMAGE ESTIMATION

For this work a simple temporal difference of frame technique (DOF) has been adopted [13]. The approach of temporal differencing makes use of pixel-wise difference between two or three consecutive frames in an image sequence to extract moving regions [13]. The DOF technique subtracts the pixel intensities from each subsequent frame in the image sequence, thereby removing static elements in the images. Based on research reported in literature, it can be stated that actions and messages can be recognized by description of the appearance of motion [16] [17] [18] [19] [20] [21] without reference to underlying static images, or a full geometric reconstruction of the moving hand [19]. It can also be argued that the static images produced using THT based on the Difference of Frames (DOF) can represent features of temporally localized motion for identification [15] [16] [20] [21] [22].

This process can be represented mathematically as follows
Let $I(x, y, n)$ be an image sequence
 &
let $D(x, y, n) = |I(x, y,n) – I(x, y,n-1)|$

Where $I(x, y, n)$ is the intensity of each pixel at location x, y in the n_{th} frame and $D(x, y, n)$, is the difference of consecutive frames representing regions of motion.

$B(x, y, n)$ is the binarisation of image difference over a threshold of Γ

$$B(x, y, n)= \begin{cases} 1 & if \ D(x, y, n) > \Gamma \\ 0 & otherwise \end{cases}$$

To represent how and where motion the image is moving, we form a Temporal-History Template (THT). In a THT H_N, pixel intensity is a function of the temporal history of

Motion at that point. The result is a scalar-valued image where more recently moving pixels are brighter.
Then THT $(H_N(x, y))$ is:

THT $(H_N\ (x, y))$

$$=\text{Max} \left\{ \sum_{n=1}^{N-1} B(x,y,n)*n \right.$$

where N represents the duration of the time window used to capture the motion. In THT more recent movements of hand actions are brighter than the older positions represented with the darker values[15] [16] [20] [21] [22]. The delimiters for the start and stop of the movement are added automatically in the sequence. The temporal history of the movement in THT is inserted into the data by multiplication of the intensity of each frame with a linear ramp representing time. The THT grey scale images are then generated by temporal integration. Thus time is explicitly encoded in the motion template.

3.2.   FEATURE EXTRACTION

The features are based on the geometrical shape and are invariant to scale, rotation and translation are based on the geometrical normalised centralised moments up to 3rd order and are called as Hu Moments [23].

3.2.1   MOMENTS, THEIR USE UP TO THE 3RD ORDER

To gain a better intuition of how to reason about moments and get an insight into how to derive invariant features based on moments, we next consider several low-order moments and describe their physical meaning.
The definition of the zero-th order moment, m_{00} , of the image $f(x,y)$ is

$$m_{00=} \sum_{x=1}^{N} \sum_{y=1}^{M} f(x, y)$$

The two first order moments, $\{m_{10}, m_{01}\}$ are used to locate the *centre of mass* of the object. The centre of mass defines a unique location with respect to the object that may be used as a reference point to describe the position of the object within the field of view. The coordinates of the centre of mass can be defined through moments as shown below

$$\overline{X} = m_{10}/m_{00}$$
$$\overline{Y} = m_{01}/m_{00}$$

According to uniqueness theory of moments for a digital image of size (N, M) the $(p+q)_{th}$ order moments m_{pq} are calculated
(For p, q   =[0,1,2...)

$$m_{pq} \equiv 1/NM \sum_{x=1}^{N} \sum_{y=1}^{M} f(x, y)\ x^p y^q$$

The m_{pq} are the moments of a digital image of size (N, M) and can be calculated as showed in (eq.4.8)

$$\mu_{pq} \equiv 1/NM \sum_{x=1}^{N} \sum_{y=1}^{M} f(x, y)\ (x-\overline{x})^p (y-\overline{y})^q$$

$f(x, y)$ is intensity function of the gray scale image and μ_{pq} are the centralized moments of the image and can be calculated .
The 1st, 2nd and 3rd order moments calculated using the above equations are,
$\mu_{00}=m_{00}\equiv\mu$
$\mu_{10}=0$
$\mu_{01}=0$
$\mu_{20}=m_{20}-\mu\ \overline{x}^2$
$\mu_{11}=m_{11}-\mu x\ \overline{y}^2$
$\mu_{02}=m_{02}-\mu\ \overline{y}^2$
$\mu_{30}=m_{30}-3m_{20}\ \overline{x}+2\mu\ \overline{x}^3$
$\mu_{21}=m_{21}-m_{20}\ \overline{y}-2m_{11}\ \overline{x}+2\mu\ \overline{x}^2\ \overline{y}$
$\mu_{12}=m_{12}-m_{02}\ \overline{x}+2\mu\ \overline{x}\ \overline{y}^2$
$\mu_{03}=m_{03}-3m_{02}\ \overline{y}+2\mu\ \overline{y}^3$

These centralized image moments are inherently translation independent but in order to achieve invariance with respect to orientation and scale, these moments have to be normalized as following by defining

$$n_{pq}=\mu_{pq}/(\mu_{00})^\gamma$$

Where $\gamma = (p+q)/2+1$ and $p+q \geq 2$

The seven moment-based features proposed by Hu that are functions of normalized moments up to the third order are a digital silhouette or boundary image, and are also rotation independent.
$M_1=(n_{20} + n_{02})$
$M_2= (n_{20} - n_{02})^2 + 4n_{11}^2$
$M_3= (n_{30} -3n_{12})^2 + (3n_{21}-n_{03})^2$
$M_4= (n_{30} +n_{12})^2 + (n_{21}+n_{03})^2$
$M_5 = (n_{30} -3n_{12})(n_{30} +n_{12})[(n_{30} +n_{12})^2-3(n_{21}+n_{03})^2]+(3n_{21}-n_{03})(n_{21}+n_{03})[3(n_{30} +n_{12})^2-(n_{21}+n_{03})^2]$
$M_6= (n_{20} - n_{02})[(n_{30} +n_{12})^2-(n_{21}+n_{03})^2] +4n_{11}(n_{30} +n_{12})(n_{21}+n_{03})$
$M_7=(3n_{21}-n_{03})(n_{30} +n_{12})[(n_{30} +n_{12})^2-3(n_{21}+n_{03})^2]-(n_{30}-3n_{12})\ (n_{21}+n_{03})[(3n_{30} +n_{12})^2-(n_{21}+n_{03})^2]$

3.3.   IDENTIFICATION

3.3.1. MAHALANOBIS DISTANCE

The Mahalanobis distance is a very useful way of determining the "similarity" of a set of values from an "unknown" sample to a set of values measured from a collection of "known" samples. It is computed by the equation below:

$$r^2 \equiv (\mathbf{f} - \mathbf{k}_x)'\ \mathbf{C}^{-1}(\mathbf{f} - \mathbf{k}_x)$$

where r is the Mahalanobis distance from the feature vector $\mathbf{f}$ to the mean vector $\mathbf{k}_x$, and $\mathbf{C}$ is the covariance matrix for $\mathbf{f}$.

Let $\mathbf{k}_1$, $\mathbf{k}_2$... $\mathbf{k}_n$ be the means (templates) for the n-classes, and C_1, C_2, ..., C_n are the corresponding covariance matrices. Feature vector $\mathbf{f}$ is classified by measuring the Mahalanobis distance from $\mathbf{f}$ to each of the means, and assigning $\mathbf{f}$ to the class for which the Mahalanobis distance is minimum.

3.4. IDENTIFICATION AND RECOGNITION PERFORMANCE

The main goal of this research is to test the identification based on the hand movements of individuals, so accuracy is considered as the criterion for performance analysis. Identification requires the subject being identified to lay claim to that identity, so that the method may decide on either accepting the enrolled subject or rejecting the subject. As with any security system, given that the subject is, or is not, a true instance of the enrolled subject, there are four possible outcomes of the errors [1]. The accuracy of any biometric method is generally judged by four error rates.

- Acceptance of Authentic Enrolled Subject (AA) or Genuine Accept Rate (GAR)
- Acceptance of Imposter Subject (IA) or False Accept Rate (FAR)
- Rejection of Authentic Subject (RA) or False Reject Rate (FRR)
- Rejection of Imposter Subject (RI) or Genuine Imposter Rejection (GRR)

The biometric system accuracy requirements depend greatly on the application. In forensic applications, such as criminal identification, FRR rate (and not FAR) is the critical design issue, because we do not want to miss a criminal even at the risk of manually examining a large number of potentially incorrect matches that the biometric system identifies. In some cases the FAR might be one of the most important factors in a highly secure access-control application, where the primary objective is prevent impostors (e.g., at airports). Many civilian applications require, the performance requirements to lie between these two limits of both FAR and FRR. In high-risk applications such as bank ATM card verification, for example, a false match would mean the loss of several hundred dollars, while a high FAR might lead to the loss of a valued customer. As our main goal is to test the THT based method for its identification accuracy for authentication. The first and the fourth identification rates are the main goals to test the efficacy of the method. So AAES (%) and ARI (%) are computed.

AAES (%)
= 100 X Total no of times correctly identifying an enrolled subject/ Total no of enrolled subject attempts
ARI (%)
=100 X Total no of correctly rejecting an imposter/ Total no of imposter attempts

4. METHOD

The method of person identification is logically divided into two separate modules: an enrolment (or training) module and a recognition (or testing) module. In first step the experimentation for enrolment and recognition is carried out and the video sequences of hand movements from the different individuals are captured and stored. From the video sequences of different individuals THTs are computed and stored. Both the enrolment and the recognition module make use of a feature extraction sub-module, which converts the THTs into set of features (Hu moments), which are very distinctive to the hand, which performs the movement. The enrolment module is responsible for enrolling new individuals in the system database. During the enrolment phase, the individual supplies a number of samples of his/her hand movements. A model of the individual is built based on the features extracted from the instances of the hand movements. During the recognition phase, the individual supplies test sample of his/her hand movement, and a measure of similarity is computed between the features of the test hand movement with the available model to establish the identity of the individual, using K-NN nearest neighbor approach using Mahalanobis Distance. The efficacy of the technique is determined by computing the Accuracy of Accepting an Enrolled Subject AAES (%) and Accuracy of Rejecting the Imposter (ARI%).

4.1 EXPERIMENTAL SETTINGS

For testing the efficacy of the method and to test the performance of identification AAES (%) and ARI (%) has to be computed. To compute the AAES (%) and ARI (%), controlled experiments were conducted.

4.1.1.EXPERIMENTS FOR GESTURE (MOVEMENT) DEPENDENT

To test the ability of the method to identify individual based on unique gestures, each subject was assigned unique hand gestures as described in Figure 2. Each subject repeated the movement for 50 times and video recordings of each of these was taken, with (5X 50) 250 video sequences of each subject specific movements from the 5 different participants. The THTs and the Hu moments of each video sequence were computed. The system was trained using 20 of the video recordings. AAES (%) and ARI (%) was calculated and tabulated for the 30 test recordings.

4.1.2.EXPERIMENTS FOR SUBJECT DEPENDENT

To determine if the variation between subjects for the same gesture was identifiable by this technique that combines THT with Hu moments and Mahalanobis distance, each subject was asked to perform a common hand gesture, "moving all fingers clockwise" (common to all subjects). Each subject repeated the movement for 50 times. There were total (5X 50=250 video sequences of

common movements) from 5 different individuals. The THTs of each video sequence was computed and features computed using 7-Hu moments.

First Frame	Intermediate Frame	End Frame	THT
All Subjects With Common Hand Movement "All Finger Moving Clockwise"			
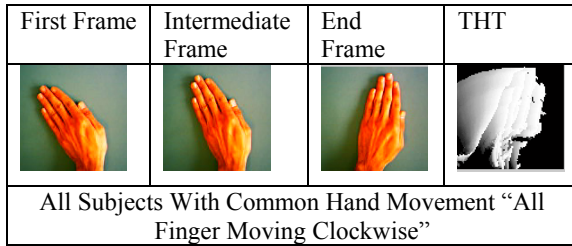
Figure 1 Figure of Hand Movement Assigned to all Subjects "All Finger Moving Clockwise"

During the training of the K-NN classifier, the Hu moments of the THT of each of the 20 video recordings of each of the subject and (4X30=120) of the imposters (the other 4 subjects) were used to identify the person's mahalanobis thresholds. This was then tested using the balance 30 video recordings of the each subject to compute Accuracy of Accepting an Enrolled Subject (AAES %). The other 30 video recordings of the other 4 subjects were used as the imposter samples to compute the Accuracy of Rejecting an Imposter% (ARI%) and this was repeated for each subject.

5. RESULTS & DISCUSSIONS

Table 1-4 describes the results of achieved AAES%/ARI % when the participants conducted common hand gestures and for their unique hand movements over the 5-subject population. Each row corresponds to the individual participant. Table 1 and 2 show the AAES/ARI with subject specific movements, each column represents the unique hand gesture while Table 3 and 4 show result with common hand movements. From the results, it is observed that the THT based person identification is movement dependent and is unable to identify the individual if the gestures are common.

ARI (%) WITH SUBJECT SPECIFIC MOVEMENTS						
SUBJECT IDENTIFIER	MTTF	RIGHT	FIST	HOLD	AFPR	AVERAGE
MTTF		87	89	93	94	90.7 %
RIGHT	93		85	96	95	92.2 %
FIST	85	78		84	90.5	84.3 %
HOLD	96	87	86		93	90.5 %
AFPR	92	88	92	89		90.25 %

Table 1

AAES (%) WITH SUBJECT SPECIFIC MOVEMENTS						
SUBJECT IDENTIFIER	MTTF	RIGHT	FIST	HOLD	AFPR	AVERAGE
MTTF	89	0	0	0	0	89 %
RIGHT	0	92	0	0	0	92 %
FIST	0	0	91	0	0	91 %
HOLD	0	0	0	93	0	93 %
AFPR	0	0	0	0	87	87 %

Table 2

AAES (%) WITH COMMON MOVEMENTS	
SUBJECT IDENTIFIER	AVERAGE
MTTF	58 %
RIGHT	60.4 %
FIST	53 %
HOLD	62 %
AFPR	55 %

Table 3

ARI (%) WITH COMMON MOVEMENTS	
SUBJECT IDENTIFIER	AVERAGE
MTTF	48.25 %
RIGHT	49.0 %
FIST	48.3 %
HOLD	52.25 %
AFPR	54.25 %

Table 4

6.CONCLUSIONS

This paper reports research of a novel method of person identification by the use of hand movements as a new biometric measurement. The low level representation of the action collapses the temporal structure of the motion from the video sequences of the hand movements while removing any static content from the video sequences to generate THTs of the hand movement. THTs of different individuals present distinctive 2-D motion patterns where each pixel describes the function of temporal history of motion in that sequence. The scale, translation and rotation invariant features have been used for discrimination of the THT for identification. On the basis of the preliminary experimental results, it can be concluded that the THT based method can be used for biometric identification with caution. The next step is to test the accuracy on large database and its sensitivity analysis.

7. REFERENCES

1.Prabhakar, S., Sharathpankanti. and A.K. Jain, *"Biometric Recognition: Security and Privacy Concerns"*. IEEE Security and Privacy, 2003.
2.George Panotopoulos, D.P., "Hand Gesture Biometrics:" Caltech Centre for Neuromorphic Systems Engineering, 2001.
3.Nixon, M.S., et al., *"Automatic Gait Recogntion"*. Biometrics: Personal Identification in Networked society, 1999: p. 231-250.
4.Huang, P.S., C.J. Harris, and M.S. Nixon, *"Human gait recognition in canonical space using temporal templates"*. VISP, April 1999.

5. Huang, P.S., C.J. Harris, and M.S. Nixon., *"Recognising humans by gait via parametric canonical space"*. Artificial Intelligence in Engineering, 1999. **13**: p. 359-366.

6. Little, J.J.a.J.E.B., *"Recognising People by their Gait: The shape of the motion"*. Videre: Journal of Computer Vision Research,, 1998. **1**: p. 2-32.

7. Shutler, J.D., M.S. Nixon, and C.J. Harris, *"Statistical Gait Recognition via Velocity Moments"*. Institute of Electrical Engineers, Savoy Place, London, 2000.

8. Niyuogi, S.A.a.E.H.A., *"Analysis and Recognizing walking figures in XYT "*. Proc. IEEE conference on Computer Vision and Pattern Recognition, June 1994.

9. Cunado, D., M.S. Nixon, and J.N. Carter, *"Using Gait as a Biometric, via Phase-weighted Magnitude Spectra"*. First International Conference, AVBPA Crans Montana Switzerland., March 1997.

10. Davis, J., *"Recognizing movement using motion histograms "*. MIT Media lab Technical Report #487., March 1999.

11. Davis, J.a.A.B., *"A robust human-silhouette extraction technique for interactive virtual environments"*. Proc. Modelling and Motion capture Techniques for Virtual Environments, 1998.

12. Davis, J., *"Representing and Recognizing Human Motion: From Motion Templates To Movement Categories"*. International Conference On Intelligent Robots and Systems,Maui, Hawaii, 2001.

13. Davis, J.a.A.B., *"The representation and recognition of human movement using temporal templates"*. Proceedings of Computer Vision and Pattern Recognition, June 1997.

14. Davis, J.a.A.B., *"Virtual PAT: a virtual personal aerobics trainer"*. Proc. Perceptual User Interfaces, November 1998.

15. Aaron F Bobick, J.D., *"The recognition of Human Movements Using Temporal Templates"*. IEEE Pattern Analysis and Machine Intelligence, 2001. **23 No 3**: p. 257-267.

16. Arun Sharma, D.K.K., Sanjay Kumar, Neil McLachalan., *"Representation and Classification of Human Movement Using Temporal Templates and Statistical Measure of Similarity"*. Workshop On Internet Telecommunications and Signal Processing. 2002. Wollongong, Sydney Australia., WITSP'2002.

17. Starner, T.P., A., *"Visual Recognition of American Sign Language Using Hidden Markov Models"*. Proc. Intl Workshop on Automated Face and Gesture Recognition Zurich, 1995.

18. Pentland, I.E.a.A., *"Coding, Analysis, Interpretation, and Recognition of Facial Expressions ",*. IEEE Trans. Pattern Analysis and Machine Intelligence, July 1997. **19, no. 7**: p. 757-763.

19. Little, J., and J. Boyd, *" Describing motion for recognition"*. International Symposium on Computer Vision, November 1995: p. 235-240.

20. Sanjay Kumar, A.S., Dinesh K Kumar, Neil McLachalan., *"Classification Of Visual Hand gestures For HCI"*. ACIVS 2002 Proceedings Of the International Conference On Advanced Concepts for Intelligent Vision Systems. 9-11 Sept 2002. Ghent, Belgium.

21. Sanjay Kumar, A.S., Dinesh Kant Kumar, Neil McLachlan., *"Classification of Visual Hand Gestures Using Difference of Frames"*. Proc. of the Int. Conf. on Imaging Science and Technology, Las Vegas, Nevada, USA , CISST'02. 2002. Las Vegas, USA: (CSREA Press, 2002).

22. Sanjay Kumar, D.K.K., Arun Sharma, Neil McLachalan., *"Visual Hand Gestures Classification Using Wavelets Transform"*. International Journal Of Wavelets and Multiresolution Information Processing (IJWMP), December-2003. **1, No-4**: p. 373-392.

23. Hu, M., *"Visual pattern recognition by moment invariants"*. IRE Transaction Information Theory, 1962. **IT-8, Num. 2**.
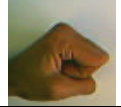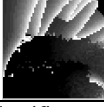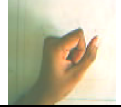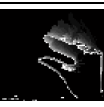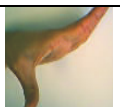
First Frame	Intermediate Frame	End Frame	THT
Subject With Movement "Moving Thumb and Three Fingers"		Subject Identifier " MTTF"	
Subject Movement "Right"		Subject Identifier "RIGHT"	
Subject With Movement" FIST"		Subject Identifier " FIST"	
Subject With Movement "HOLD"		Subject Identifier " HOLD"	
Subject With Movement "All Fingers Point Right"		Subject Identifier " AFPR"	

Figure 2

Usable multimodal multimedia application intended for disabled persons

Samir Benarif, Hichem Djenidi, Amar Ramdane-Cherif, Nicole Levy

PRISM Laboratory, Université de Versailles St.-Quentin,
45, Avenue des États-Unis, 78035 Versailles Cedex, France.

ABSTRACT

Today, every body uses computers, in the everyday life such as accessing an Internet, withdrawing cash, or indirectly in a washing machine or a car. A proportion of the population differs from the average in terms of their physical, sensory or cognitive abilities. This section of the population requires the same access to computers as everyone. This paper describes a multimodal multimedia software environment where speech, eye-gaze, and/or wireless control, virtual keyboard and/or tactile screen are used as input modalities and where, display on monitor screen and voice synthesis, are the output ones. This application is dedicated to paralytics and allows them to navigate in the web and to use the Windows environment. Our application called "InterAct" consists in multimodal multimedia architecture "MMA" based on multithreaded components. To increase the usability of InterAct, we predict some interaction between the user and our application expressed in the specific scenarios.

1. INTRODUCTION

The growing interest in multimodal interface design is inspired in large part by the goals of supporting more transparent, flexible, efficient, and powerfully expressive means of human-computer interaction than in the past. Multimodal interfaces are expected to support a wider range of diverse applications, to be usable by a broader spectrum of the population, and to function more reliably under realistic and challenging usage conditions.

Technology has played a role in helping persons with disabilities live and work independently. The promise of the virtual reality (VR) technologies to provide powerful and unique education and informational experiences is well know [1]. In [2], the researches are focused to give overviews of some of the way that people with different types of disability can use a computer to enable them make music. Other researches focus on designing new graphic manipulators like PadGraph [3] based on adjustments of the mass center of the operator. It allows the user to create graphic objects without use of hand operations. Eagle Eyes [4] allows a person to control the computer by moving eye or head. The method is based on the measurement of electro-oculographic potential (EOG), through electrodes placed on the head. Periodically, new prototypes are designed in laboratories, but without an effective management of all these materials, the disabled persons are still unable to control their computer. Multimedia Multimodal (MM) applications are becoming an important technique used in human-machines systems for evaluation of performance and research [5], but it is also a fundamental tool for people with special needs. A flexible MM application should offer more freedom to the user, since the disabled user can choose the best input/output modalities adapted to its impairments. The MM application should also give him/her the opportunity to keep an input media during the dialog with the intelligent system (in the runtime mode). At last and in a concurrent way, it should react, in real time, to his/her unexpected errors.

The software development for users with special needs is a part of the discipline of humane computer interaction. We develop a MMA based on multithreading components. The different modalities are introduced in the application in order to offer to the disabled users several ways to interact with the computer. We choose at the minimum one device for each principal human sensor (vision, hearing and touch). The diversification of the input modalities offers to the user a choice of the type of interaction needed in the communication with computer and allows our application to be adaptive at the user impairments. A fusion process in our software permits to use several modalities at the same time. Some requirements must be satisfied by the software in term of the functionality and interactivity. The application should be easier to use and should react to the user unexpected errors. In other terms, our MM software (MMS) satisfy some quality attributes like usability, performance and dependability. The usability is important quality in the software architecture especially for the users with special needs; the disabled person must have the same opportunities to use the computer as other people.

2. PRESENTATION OF MULTIMODAL MULTIMEDIA ARCHITECTURE

It is important to distinguish two areas of application of information technology as used by people with special needs. As we are all aware, computers are becoming an increasingly vital part of daily life in work, education and leisure. More people require access to them in daily life and that include people who have impairments which affect their ability to access standard technology. There is, therefore, a need to adapt human computer interfaces so as to make them accessible to such people. The second area of application is what can be referred to as assistive use of computers. That is to say, using technology to attempt to alleviate some of the limitations caused by disability. In such a role, a disabled person would use a computer (or computer-based technology) for tasks for which most people would not use the computer.

Our MMS (which is a part of application called InterAct: Active Interface) assists the disabled person during the interaction with the computer, it plays the role of intelligent intermediate between the user and the computer. It offers to the disabled user certain applications like Internet navigator, but it permits him/her to use all windows applications that procure great freedom. MMS offers to the disabled users the possibility to control the computer in Windows environment via different modalities. During the communication with the computer, MMS application can suggest autonomously to the user some modalities by taking into account his impairments.

MMS is based on components using multithreading technology. The disabled users can use several modalities to interact with the computer. We implement for example the mouse emulator (wireless control mouse), virtual keyboard (keyboard showing on the screen), Eye Gaze System (permits to the user to move mouse with its eyes), tactical screen, voice (speech recognition) and classical mouse and keyboard. The Figure 1 gives an overview of the interface of fusion. The Interpreter components are special components designed to translate the signals sent by the input devices. The Filter component filters messages received from the Interpreter components. The interval of fusion is given by Timer component, it resets Fusion component if the time of the fusion is elapsed. The Fusion component saves the commands in order to merge them. The Visual, Audio and Commands components are linked to the output modalities.
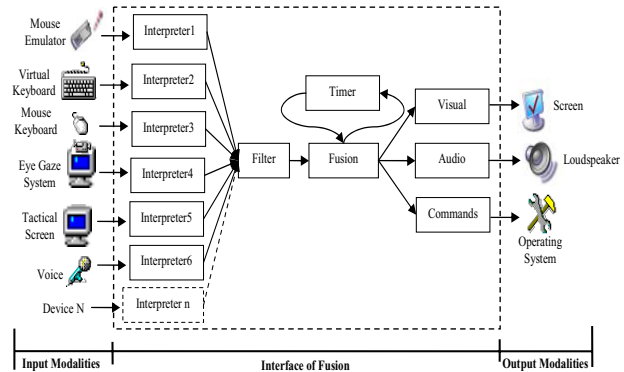
Figure 1: Overview of multimodal multimedia of InterAct architecture

2.1 Description of the Interpreter Components Linked To the Input Devices

We integrate the Interpreter components into the MMS in order to normalize the messages in the interface of fusion. Each Interpreter component is designed especially for one input device. The Interpreter 4 is detailed in the following:

EyeGaze Interpreter component

The Interpreter component of the Eye Gaze System is composed of several components (see Figure 2). The Calibration component is used to calibrate the Eye Gaze System for the user (screen definition and Eye position); the calibration process is activated in the first use of the software. The TrackEye component is a process that captures the position (in the computer screen) of the gaze of the user (the position is expressed in pixels), at fixed frequency. These positions are saved in database. The role of the Converter component is to collect the position (saved in database) and to send it like message to the Filter component. The message contains the command, for example "mouseclick()", which suggests a click of mouse on the screen.
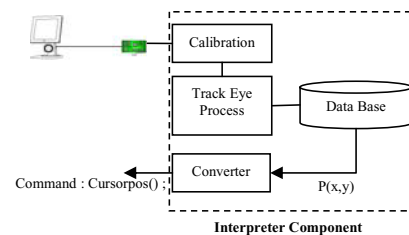
Figure 2: Description of Interpreter component of the Eye Gaze System

Filter component

The set of messages provided by the Interpreter components must be filtered before the fusion. The Filter component has the role of filter, but it has another role, it parameterizes all received messages and identifies the complex and simple commands by using its list (the list contains all complex and simple command).

Fusion component

Fusion component is the engine of the fusion process. It contains a basic MM dialog [6] [7], it gives the user the option of deciding which modality or combination of

modalities is better suited to the particular task and environment.

The commands received by the Fusion component are labialized and saved into table. To describe the fusion process, we chosen an example, in the semantic expression: (command 1→command 2), command 1 is simply followed by command 2. These expressions use the arrow operator for sequential concatenation in the time domain. In the following table, the codes (last column of Table2) are obtained simply by summing the command labels of each semantic code. The obtained codes give information relative at the new merged command for the Fusion component. In the first example of Table 2, two commands (open, that) are merged, the Fusion component means "Open Object" and executes the command labialized $\{(8);(1)\}$.

Set of Sentences	Command meaning	Set of corresponding semantic codes	Set of corresponding codes
{ (open→that); (open) }	Open object	{ (1→7); (1) }	{ (8); (1) }
{ (close→that); (close) }	Close object	{ (2→7); (2) }	{ (9); (2) }
{ (delete→that); (delete).}	Delete object	{ (3→7); (3) }	{ (10); (3) }
{ (paste) }	Past last copied object	{ (5) }	{ (5) }
{ (copy→that); (copy) }	Copy object	{ (4→7); (4) }	{ (11); (4) }
{ (cancel)}	Cancel last command	{ (6) }	{ (6) }

Table 1: Example grammar of authorized sentences

The fusion entities like ((close→open) // click), (click // (delete→open)), etc. or isolated clicks are also ignored by the system. Thus, some errors made by the user are avoided by the model.

2.2 Description of the Fusion Process
InterAct has the ability to manage several modalities in same time. The commands are sent by the input modalities; it can be merged by taking into account the commands that are arrived in the interval of fusion. The Interval of fusion is saved in the Timer component and can be modified by the user (at run-time). Each component of the interface of fusion is used to special task. The Filter component receives messages from all Interpreter components and filters them. Two kinds of messages are identified: simple and complex command

Example of a Complex command
The complex command can be used in fusion process. In this case, the first complex command is stored in the Fusion component. The Timer component is activated in order to count the time of fusion. If the time of fusion is occurred and no other complex command is detected, the Fusion component is reinitialized (the first command is removed). If combination of the complex commands is occurred, the commands are sent to the components of the output devices.

Example (see Figure 3): In this case, the disabled user with the mobility handicap wants to target a file, and after, to copy it. The disabled user activates the Eye Gaze System and the Speech recognition to achieve the previous actions. He says "copy file" on the microphone, the Interpreter component starts the recognition process and sends a new message to the Filter component (step 1). The message contains the command relative to the dictated word of the user. The Filter component sends the message to the Fusion component (step 2) and parameterizes the message like being a complex command. The Fusion component checks the message and saves the first complex command (step 3). It activates the Timer component, which starts the countdown (step 4). The interval of the fusion is configured previously by the user. If the time of fusion is occurred and no other complex command arrives, the Timer component resets the Fusion component (step 5), the commands are erased. If the gaze of the user is fixed on the portion of the screen the Interpreter component sends a new message to the Filter component (step 6). The message contains command "fix", which informs the interface of the position object fixed by the user. The Filter component parameterizes the message like the complex command. If the time of fusion is not occurred, the Fusion component executes the fusion between the first and the second command (step 7) by using the grammar of authorized sentences. The execution of the merged commands is applied by the operating system (step 8); the Audio component uses human voice (step 9) and the Visual component uses visual warnings (step 10) for indicate to the user that the fusion is achieved. The Fusion component stops the Timer component (step 11), which resets Fusion component (step 12).
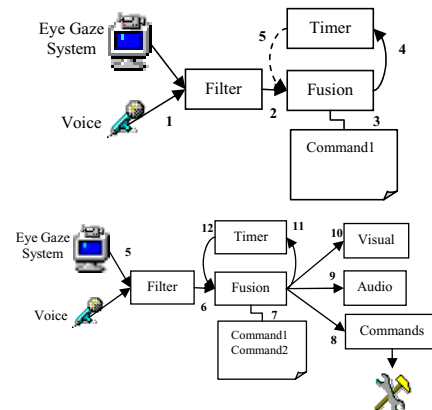
Figure 3: Fusion process in the InterAct software

3. GENERAL SCENARIO OF THE USABILITY TO ASSIST PERSONS WITH SPECIAL NEEDS
One of the key concepts in HCI is usability- that is, making systems easy to learn and easy to use. This entails understanding the factors (psychological, socially, ergonomically, etc.) that determine how people operate the system. The designer and developer must ensure that the systems are suitable to people and produce efficient and successful interaction between the user and the system.

Usability is defined in this way in ISO 9241-11: "Usability is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use".

To assist the disabled users during the use of the computer, our application must characterized by a high degree of usability. The goal of this section is to achieve better system usability through autonomous architectural decisions at run-time mode. Hence, understanding the relationship between software architecture and usability is important to ensure that the system ultimately achieves it. A general usability scenario describes an interaction that some stakeholders have with the MMS under consideration from a usability point of view [8]. We cite some usability scenarios of InterAct like cancelling commands, Using InterAct concurrently with other applications, checking correctness of the user, supporting multiple activities during the communication, navigating quickly into Windows operating system, supporting comprehensive searching, configuring the time of fusion, automating the selection of the input/output modalities, etc.

3.1 Autonomous Managing Of the Input Modalities

The disabled users can use several modalities at same time to interact with the computer. The different modalities introduced in InterAct simulate the mouse and the keyboard events. For example, the tactical screen and Eye Gaze system simulate the mouse functionalities, but the problem occurs when the users attempt to use two devices simulating the same physical device. We impose the input modalities activation rule in order to avoid the conflict devices problem. We organize the modalities in three groups:

- Group 1 (Mouse Simulator): mouse, tactical screen, Eye Gaze System, Wireless control mouse.

- Group 2 (Keyboard Simulator): keyboard, virtual keyboard.

- Group 3 (Vocal Commands): Speech recognition.

The rule is simple; the user can't activate two input modalities belonging to the same group (for example, eye gaze system and the mouse). But one input modality can be used with other modality belonging to the other group (for example, the tactical screen and speech recognition).

The management of the input modalities is expressed by a rule and is included into Filter Component. The stimulus of the scenario is the activation of the new input modality. The Filter Component manages the input modalities autonomously and avoid to the disabled user the generation of the errors during the communication with the computer. If the new input modality is activated, the Filter Component percepts the event and after, tests if this modality is used at the same time with another device belonging at the same group. If the new modality can cause the problem, the scenario consists on deactivation of the new modality. The filter component deactivates the

Interpreter Component of the new modality and its connection .The input modality is disabled.

4. CONCLUSION

This paper describes InterAct software which is a new assistive and adaptable application for the disabled users. InterAct is a multimodal multimedia software. This application is dedicated to paralytics and allows them to navigate in the web and to use the Windows environment. The diversification of the input modalities offers to the user a choice of the type of interaction needed in the communication with computer and permits to our application to be adaptive at the user impairments. Finally, we increase the interactivity of our application with the some usability scenarios.

5. REFERENCES

1. Virtual realism. New York: Oxford University Press.

2. T. Anderson, R&D manager, Drake Music Project. Enabling physically disabled people to play and compose music. Reprinted From Ability Magazine. Spring 1997.

3. G. Evreinov, A. Agranovski, A. Yashkin, T. Evreinova, PadGraph, in: Human-Computer Interaction: Communication, Cooperation, and Application Design, Volume 2 of the Proceedings of HCI International '99 (the 8th International Conference on Human-Computer Interaction), Munich, Germany, (eds.: Hans-J.o.rg Bullinger and J.u.rgen Ziegler), Lawrence Erlbaum Associates, Publishers Mahwah, New Jersey, London, 1999.

4. J. Gips, Ph. Dimattia, F.X. Curran and P. Olivieri, Using Eagle Eyes - An Electrodes Based Device For Controlling The Computer with your Eyes - to Help People with Special Needs, in Interdisciplinary Aspects on Computers Helping People with Special Needs (eds.: Klaus, J., Auff, E., Kremser, W., Zagler, W.L.), ICCHP'96, Linz, Verlag R. Oldenburg, Wien Munchen, 1996.

5. Y. Bellik, D. Burger, Multimodal Interfaces: New Solutions to the Problem of the Computer Accessibility for the Blind. Proc. CHI'94, Boston, 24-28 April 1994.

6. H. Djenidi, S. Benarif, A. Ramdane-Cherif, C. Tadj and N. Levy. Generic Multimedia Multimodal Agents Paradigms and its Dynamic Based Agent Reconfiguration at the Architectural Level. In EURASIP Journal on Applied Signal Processing, , Multimedia Human-Computer Interface. 2004.

7. H. Djenidi, A. Ramdane-Cherif, C. Tadj and N. Levy. Generic Multi-Agents Architecture for Multimedia Multimodal Software Environment. In JOT'2004 : International Journal of Object Technology. Sept/Oct. 2004.

8. Nielsen, J. Usability Engineering. Boston, MA: Academic Press Inc, 1993.86 CMU/SEI-2001-TR-005

CHAMBRE: A distributed environment for the production of multimedia events

Paolo Bottoni   Stefano Faralli   Anna Labella

Claudio Scozzafava

Department of Computer Science, University of Rome - "La Sapienza", Italy

Abstract

We present an environment able to coordinate the operations of distributed computing devices as well as an apparatus of sensors and multimedia renderers. The use of the environment is demonstrated by the description of an artistic event in which inputs coming from light sensors steer the production of sound and visual effects.

1 Introduction

CHAMBRE is a software enabling the use of multimodal inputs to produce multimedial objects with a hypermedia and virtual structure. It can therefore be exploited by designers of multimedia applications or performances to realise artefacts whose users may enjoy a multisensorial experience. It also allows the configuration of several processing units, to strenghten the management of hypermedia aspects and the interaction model. Hence, it makes it possible to construct networks of multimedia tools, able to exchange information. Each multimedia object is able to receive, process and produce data, as well as to form local networks with its connected (software) objects. In CHAMBRE, communication can thus occur on several channels, providing flexibility and robustness.

Figure 1 shows the relation between CHAMBRE and its input and output channels.

The design of CHAMBRE is focused about a notion of interoperability, so that communication between software elements occurs on simple channels allowing the transfer of a simple form of communication, namely formatted strings, with a proper encoding of the type of request and/or information which is transmitted. Special channels can also be devised, as is the case of the use of MIDI channels in the application illustrated in this paper.

A CHAMBRE application is typically built by interactively specifying a dataflow graph, where edges are communication channels, and nodes are processing units exposing ports. These ports can be adapted to receive channels from within the same subnet, or through TCP or MIDI connections.
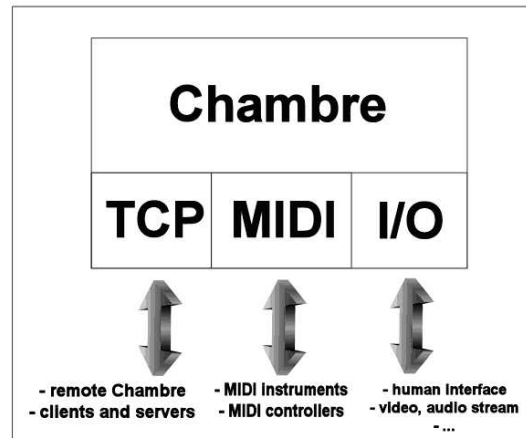
Figure 1. Communication between CHAMBRE **and the external world.**

In this paper, we illustrate the generation of multimedia artistic events, where a performer generates positional events by moving in a dark room. The events are processed by a CHAMBRE network, which interprets them to modify parameters steering the production of musical and computer graphics responses. These are played back to the performers, who can thus tune their movements in orer to obtain some effects, as well as to an audience outside the room, potentially at some other geographical location.

Paper organisation. In the next Section we refer to some related work, both in the area of dataflow architectures and in the area of computer-supported artistic creation. In Section 3, we present the most important features of the architecture. Section 4 describes the use of CHAMBRE for the creation of artistic performances, and Section 5 draws conclusions.

2 Related work

Flowgraph models have often been used for the definition of configurable architectures, due to the facility with which

a user can devise and define the dependencies between elements. An example in the field of the construction of user interface development environments is Amulet [8], where elements are connected by a web of constraints, while a direct connection can be drawn with the LabView [1] or Prograph [3] tools, where connections between elements describe passage of data.

Frameworks for the construction of distributed (object-oriented) systems have usually focused on passage of messages (actually remote procedure calls or method invocation), rather than on data. In these frameworks, the necessity for creation of stubs and proxies usually does not favour reconfiguration. The problem of reconfiguration has been addressed by Coulson *et al.*, by proposing a component-based definition of the middleware itself [2]. Their solution involves an explicit representation of dependencies to determine the implications of removing or replacing a component. In our approach, as it is the receiver's responsibility to deal with data, we can live with the generation of inconsistent connections, by the fact that a component not interested in some input will simply ignore it.

Several researchers and artists are devising ways to employ multimedia tools as creative devices. We draw the attention of the reader to some experiences which are more closely connected to the application depicted here. Co-generation of graphics and music has been the objective of [7, 5], where, however, one medium took control of the other. Flavia Sparacino's work explores the use of a dancer's movements as a way to govern an orchestra of instruments virtually associated with the dancer's body parts [9]. This requires a predefinition of the mappings between movements and controls, whereas our system allows dynamic reconfiguration as well as intervention of other sources for parameter control. Fels *et al.* have proposed MusiKalscope as a grahical musical instrument, combining a virtual kaleidoscope with a system for generating musical improvisations, and provide the performer with virtual sticks through which to drive effects in the two media [4]. Also in this case, the coupling of the two subsystem appears to be a rigid one.

The spatial distribution of performers is stressed in the distributed musical rehearsal environment [6], where an orchestra conductor and the players can be physically removed by one another. The peculiarity of the task, however, makes it necessary the use of specific solutions to preserve the spatial perception of the music sources, which are not necessary here.

3 Chambre architecture

Figure 2 shows a possible configuration of the CHAMBRE architecture, in which a network of objects kept on a single processor, communicates with other similar local networks and with one *Graphics* and one *Audio* servers on different TCP connections, which can carry input and output signals. Further inputs are provided by a camera, whose pictures are processed to detect specific events, and by a MIDI device.

Chambre provides specific drivers for inputs and outputs in MIDI format, as well as managers of TCP connnections.
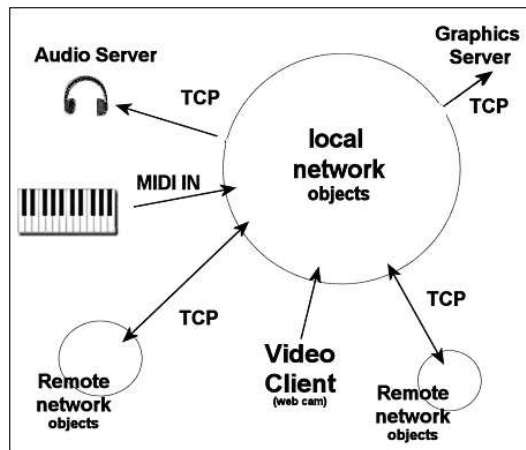
Figure 2. A configuration of a CHAMBRE **distributed application.**

CHAMBRE provides users with several modules (instances of subtypes of an abstract `ProcessingUnit` class), through which they can create any type of (sub)network.

The basic kind of communication in CHAMBRE is realised in an asynchronous way, exploiting the `Observable-Observer` pattern, as supported by the Java Virtual Machine Implementation. Hence, the `ProcessingUnit` class is defined as an extension of `Observable` and an implementation of `Observer`. Every processing unit is associated with objects responsible to maintain the input and output communication. The `Communication` class abstracts from the specific forms of communication, namely local communication, or communication across on a TCP connection or on the MIDI channel. In order to allow a processing unit not only to send out data on the MIDI channel, but also to receive a MIDI stream, a special `Bridge` needed to be defined.

Figure 3 presents a class diagram describing the core of CHAMBRE architecture, i.e. its `ProcessingUnit` class (in the diagram the real name of the class, `UnitaElaborativa`, appears).

Messages exchanged between objects are packaged and unpackaged by suitable interface modules provided by the specialisations of `Communication` and processed by any CHAMBRE object. Processing units can be in different states as regards their ability to receive or send on the different channels, or to communicate with other units in the same local network.

The `arg` parameter of the `update` method (required by the `Observer` interface) in any concrete realisation of `ProcessingUnit` is always packaged as a `String` object. Strings can be formatted in different ways. Basically each string message is considered as a sequence of tokens sepa-
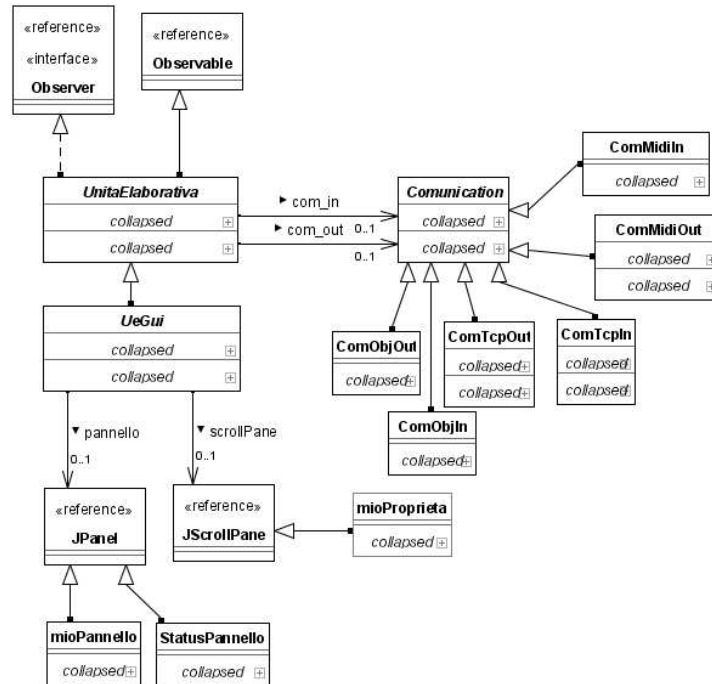
Figure 3. The core of CHAMBRE **architecture.**

rated by spaces. Objects can use special `Parser`s to search the string for tokens at a given position in the string, or recognise suffixes or prefixes, or parse the string looking for attribute-value pairs. It is responsibility of the sender to produce a message in a way which is coherent with the format managed by the receiver. On the other hand, any received message which is not formatted in the way foreseen by the receiver class will be simply ignored, without provoking any disruption of the communication mechanism.

A CHAMBRE application appears therefore as a directed graph of processing units, where the direction of an edge indicates the direction of the communication flow, i.e. which unit acts as a sender and which as a receiver. Units can be multiply connected, as allowed by the registration mechanism of the `Observable-Observer` pattern. It is also possible that a unit must receive inputs from different sources to proceed with its computation. In this case it can exploit helper classes which buffer the inputs and transmit them when all the inputs are ready, or at defined intervals of time.

The development of a CHAMBRE application is realised by defining plugin packages. A CHAMBRE plugin contains a `Plug` class, which inherits from `ProcessingUnit` and implements the `InterfacePlugIn` interface. Figure 4 shows the relations between classes needed to realize a plugin.

This interface specifies the `init()`, `start()`, and `stop()` methods, through which a plugin can act within the network of units to which it has been added. It also allows the definition of methods for the visualisation of the plugin in

the network, as a 32×32 window, and of an icon for creating instances of it. The developer must then provide implementations for the `update` method, as well as for setting up an instance of the plugin in the network and presenting a GUI for interaction with it (i.e. an implementation of the abstract class `UEGUI`.

A number of plugins have been produced and are available within the CHAMBRE environment. They are able to perform arithmetics, geometric transformations, to execute finite state automata, or are signal generators, switches, sliders, etc.

CHAMBRE provides the designer with an interface through which the current configuration of the network and a summary of the available networks is represented. In a collaborative scenario different designers can register the local network for which they are responsible to the *RncServer* plugin. Each designer is provided with an *RncClient* plugin, able to connect itself to the server. A network is elected to maintain the server, so that this instance of *RncServer* maintains the information about the local networks which can therefore be observed by every designer involved in the collaboration.

Figure 5 shows the selection of a network from a combobox and the representation of the selected network, showing information about the features of each node in it, namely the type of messages it can understand, the script through which input messages are processed by the node, the type of messages it can produce, as well as the ports through which MIDI or TCP data can be sent and received.
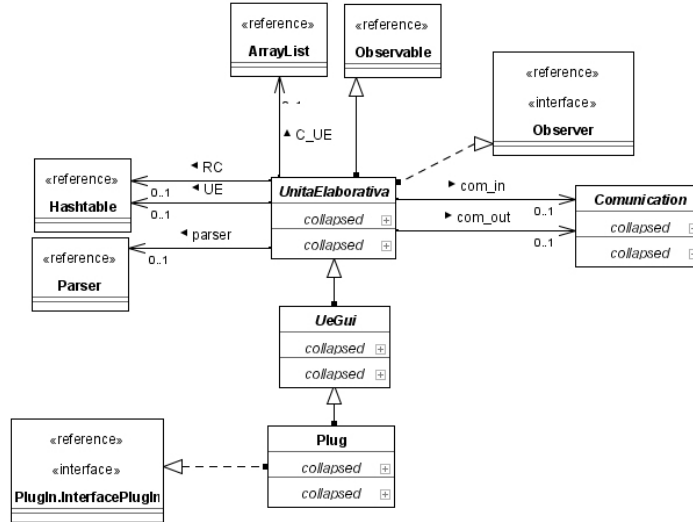
Figure 4. The classes involved in the definition of a CHAMBRE **plugin.**
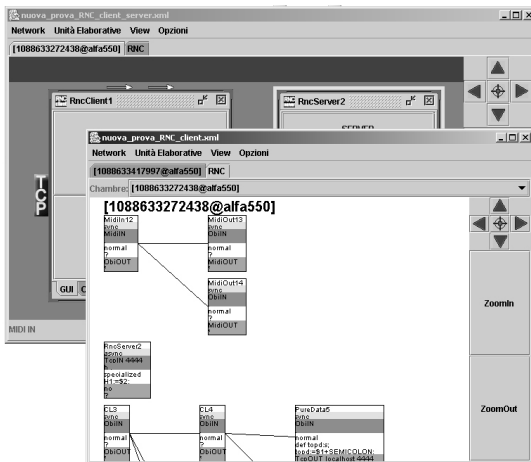
Figure 5. During interaction with a client, a designer can observe a representation of a remote network.

4 Construction of a multimedia event

In this section we illustrate the use of CHAMBRE in the construction of an artistic performance.

A dancer performs movements in a closed environment, with constant light. A reference frame is captured of the environment without the performer. During the performance, a Webcam produces a stream of pictures at a rate of 20 frames per second. The *VideoGrid* application receives this stream and subdivides each picture into a grid of 24×24 subpictures. The *VariationDetection* compo-

nent of *VideoGrid* executes an algorithm to detect variations in the current frame with respect to the reference frame. The result is marshalled into a message of the form $msg1 = (G < dimx >< dimy >< matrix >)$ and sent via TCP to CHAMBRE. G is a label which indicates that the content of the message is a grid, whose size is given by $< dimx >, < dimy >$ and whose content is given as a binary string in $< matrix >$. The *FormDetection* plugin of CHAMBRE evaluates $msg1$ to identify the bounding boxes for each connected component in the grid. It produces a list of forms and creates a message $msg2 = (L < forms >< formlist >< dimx >< dimy >< matrix >)$ where L is the conventional label for this type of message, $< forms >$ is the number of identified forms, and $< formlist >$ is a sequence composed of the four coordinates of the extremal points for each bounding box. $msg2$ is finally sent (again via TCP) to the *GraphicsServer* component which produces 3D representations of the described forms. The representations are projected onto a screen visible both to the audience and to the dancer, who can thus form a feedback loop with the generated images.

Figure 6 shows the fragment of the CHAMBRE network realizing the form detection and its transfer over the network, while Figure 7 shows the process from video capture to form identification for two different frames.

The addition of a "soundtrack" is easily obtained by interfacing some software such as **PureData** or **MAX/MSP** to our application. These tools are indeed well suited to the algorithmic treatment of audio and MIDI data, and are able to support communication via TCP/IP.

In fact, communication with **PureData** occurs via a TCP connection, through which it receives sequences of data interpreted as control parameters to steer functions for sound synthesis. It is therefore possible to realise a
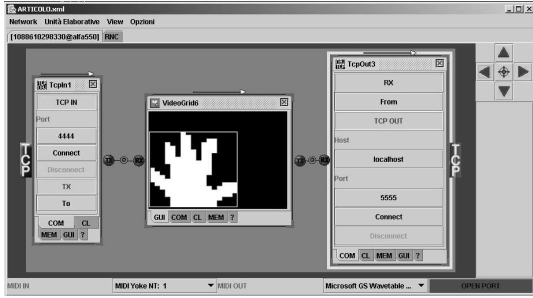
Figure 6. The CHAMBRE **network involved in the detection of forms and communication over the network.**
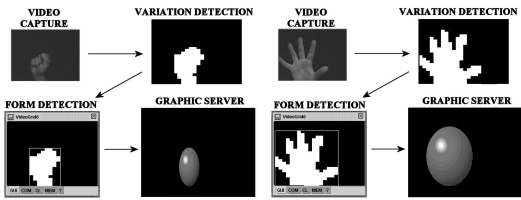
Figure 7. A sketch of the video capture and form detection process.

network of objects which use the content of the messages $msg1$ and $msg2$ to extract control parameters for music synthesis in **PureData**. The dancer can thus modify his or her movements to steer music generation. Figure 8 shows the window describing the state of the communication with **PureData**.
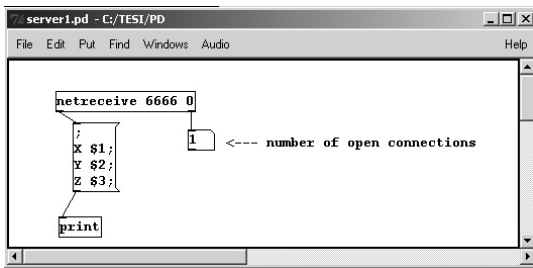
Figure 8. Observing the state of the communication with PureData.

In the construction of the multimedia event, the different tasks can be realized on a local network, as shown in Figure 9, or distributed among several processors hosting the different components and servers, according to the schema of Figure 10.

The different types of multimodal inputs, or multime-
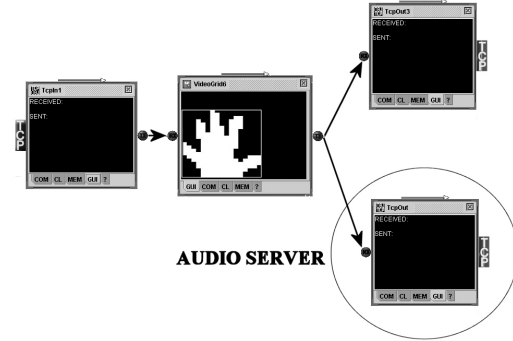
Figure 9. Introducing a processing network between a camera and a PureData **music generator.**
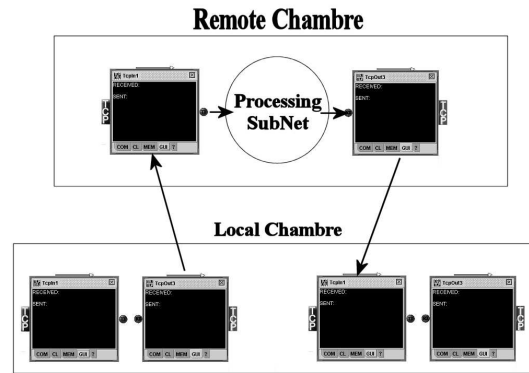
Figure 10. Distributing information processing over a network.

dia results can be the final object of creation, or the basis for further creative processes, realising a full circle of feedback between creator, executor, and computing device. An artist is therefore forced to think of the artistic support as a function of interactive processes operating through different channels. Several installations of this type could also be connected over a network, so that different artists could interact over a geographic network.

5 Conclusions

The CHAMBRE environment for the development of distributed multimedia applications has been presented. It is an open environment based on a simple model of communication among processing units. Developers can exploit the communication primitives and create their application, in the form of networks of plugins, chosen among those provided by CHAMBRE or created in accordance with a specific

interface, and inheriting from an abstract class.

CHAMBRE appears as a very flexible environment for several types of application, in the area of distributed computing as well as of multimedia, hypermedia and virtual reality applications. It can be used in the teaching arena, for scientific or artistic purposes or for complex process control.

We have illustrated its use in the context of artistic creation. Here different types of sensorial interaction (IN/OUT audio and video in the presented application, but haptic plugins can be also devised) provide the artist with a complete development framework. An artist can manage input and outputs stream in an interactive, concurrent way.

The open and naturally interconnected structure of CHAMBRE allows the communication among different workplaces or operators with different specialisations for a given task. It is potentially open to communication with any software able to exploit a TCP/IP connection.

We plan to extend the set of input and output devices controlled by CHAMBRE, possibly including other types of communication channels. our experimentation in the artistic field with the CHAMBRE environment, and envision some enrichment of the proposed performance. For example, by coordinating the input from two cameras, one could recognise movements in the the 3D space, as opposed to simple displacement on the horizontal surface. Another possibility is to exploit a number of light sensors arranged on a grid, to detect the relative positions of several dancers. In all these cases, some pattern detection algorithm on these inputs could be used to drive parameters of different algorithmic operators.

References

[1] Labview: The software that powers virtual instrumentation. http://www.ni.com/labview/.

[2] Geoff Coulson, Gordon S. Blair, Michael Clarke, and Nikos Parlavantzas. The design of a configurable and reconfigurable middleware platform. *Distributed Computing*, 15(2):109–126, 2002.

[3] P.T. Cox, F.R. Giles, and T. Pietrzykowski. Prograph: a step towards liberating programming from textual conditioning. In *Proceedings of the 1989 IEEE Workshop on Visual Languages*, pages 150–156. IEEE CS Press, 1989.

[4] Sidney Fels, Kazushi Nishimoto, and Kenji Mase. MusiKalscope: A graphical musical instrument. *IEEE MultiMedia*, 5(3):26–35, 1998.

[5] Masataka Goto and Yoichi Muraoka. A virtual dancer. "Cindy" — interactive performance of a music-controlled CG dancer. In *Proc. of Lifelike Computer Characters*, 1996.

[6] D. Konstantas, Y. Orlarey, O. Carbonnel, and S. Gibbs. The distributed musical rehearsal environment. *IEEE Multimedia*, 6(3):54–64, 1999.

[7] A. Kotani and P. Maes. An environment for musical collaboration between agents and users. In *Proc. of Lifelike Computer Characters*, 1995.

[8] Brad A. Myers, Richard G. McDaniel, Robert C. Miller, Alan S. Ferrency, Andrew Faulring, Bruce D. Kyle, Andrew Mickish, Alex Klimovitski, and Patrick Doane. The Amulet environment: New models for effective user interface software development. *IEEE Trans. Softw. Eng.*, 23(6):347–365, 1997.

[9] F. Sparacino, G. Davenport, and A. Pentland. Media in performance: Interactive spaces for dance, theater, circus, and museum exhibits. *IBM Systems Journal*, 39(3 & 4):479, 2000.

TANDEM – Transmitting Asynchronous Non Deterministic and Deterministic Events in Multimedia Systems over the Internet

Angela Guercio and Arvind K. Bansal
Department of Computer Science
Kent State University, Kent, OH 44242
guercioa@hiram.edu and arvind@cs.kent.edu
Tel. 330-672 9035 Fax. 330-672 7824

Abstract

The Internet is being used more pervasively for multimedia retrieval, multimedia transmission and rendering. However, little work has gone on in Internet-based multimedia modeling integrating user interaction, asynchronous and non-deterministic multimedia events, multiple multimedia streams, and their synchronization. In this paper, we describe a new and novel XML based multimedia language 'TANDEM' which supports the integration of user interaction, asynchronous non deterministic multimedia events, multiple multimedia streams, and their synchronization. More specifically, we describe 'event constructs', 'stream grouping constructs', 'transformation constructs', and 'synchronization constructs'. We illustrate the constructs with relevant examples.

Keywords: asynchronous, event, Internet, language, modeling, multimedia, nondeterministic, synchronization.

1. Introduction

As the Internet becomes pervasive, the multimedia knowledge base on the Internet will become a source for Internet-based multimedia modeling application based upon user interaction. For an effective use of distributed multimedia data over the Web, we need to develop a series of tools and languages that can help programmers to create such applications.

Current multimedia languages [9] are evolving, and are limited in their ability to model real world phenomena which need integration of multimedia reactivity, asynchronous events, computability, event-based triggering, dynamic altering of multimedia attributes, and synchronization of multiple streams. Languages for modeling distributed multimedia systems must support asynchronous events, loose ordering of events, and automated dilation of time-scale in a group of streams to preserve synchronization.

This paper introduces TANDEM — an XML-based distributed multimedia language for Transmitting Asynchronous Non-deterministic and Deterministic Events in Multimedia systems. The language supports the development of distributed multimedia systems which integrate deterministic and asynchronous non-deterministic events over the Internet. The language is based on a conceptual model that has a trigger – a mechanism which controls the reaction to multimedia content in multimedia streams — as the event generator which instructs the transformers for the appropriate reaction. A trigger accepts multimedia streams (both periodic and aperiodic) as input along with the associated media content, and generates one or more events. The language provides constraint-based synchronization facilities with dynamic scaling for groups of multimedia streams, which have a clock associated with them to control their synchronization. Context-related reactions to situations are detected by an active repository, as are reactions to the external world. The use of a persistent repository takes care of partial conditions and the change in the order of nondeterministic conditions since the truthfulness of partial and nondeterministic conditions can be archived in the active repository.

The major contributions of this paper are:

(1) The language allows the integration of asynchronous events, deterministic multimedia events, and multimedia event-based triggering commands.

(2) The language supports temporal and spatial synchronization of complex media objects.

(3) The language separates and integrates five major components of distributed multimedia reactive systems: triggers; active repository; transformers; grouping of media streams; and synchronization.

(4) The language allows dynamic scaling of the multimedia time base for implicit synchronization.

(5) The language incorporates commands that interrogate the active repository for partial condition verification as well as pattern detection, oblivious data and context-based conditions.

The paper is organized as follow. Section 2 briefly describes related concepts in distributed reactive multimedia systems. Section 3 describes the conceptual model for TANDEM. Section 4 describes the constructs for multimedia events, stream grouping for synchronization, synchronization and transformation constructs. Section 5 presents related works. The last section concludes the paper. Due to space limitations, the grammar for the language and a description of a major nontrivial application will be presented elsewhere.

2. Distributed Reactive Multimedia Systems

Multimedia systems provide interaction involving text, graphics, audio and video. The interaction is obtained via multimedia streams and aperiodic signals. A multimedia stream is a sequence of nested tuples and data values. A multimedia stream can be:

(1) a *continuous stream* as produced by sensors
(2) a *periodic stream* where data is associated with a periodic signal, or
(3) an *aperiodic stream* where the data is associated with an aperiodic signal generated by an event or external interaction.

A *multimedia stream S* has two components: *attribute-set* and *data*. Three attributes *periodic* or *aperiodic*, number of data elements per unit time, and type of data (such as *audio* or *video* or *music or audiovisual* etc.) are essential. Other attributes are specific to the streams, and vary among different types of multimedia streams.

Example 1: The data for the audio stream is a sequence of sampled packets with the attributes: (a_0=periodic, a_1=audio, a_2 = 44100 samples/second, a_3 = no. of channels = 4, a_4 = 16 bits per sample, a_5 = media length, …).

Each multimedia media stream has its own clock which is synchronized to a *common clock*, and is played according to the *playback rate* R of the media type.

In distributed reactive multimedia systems [2], a number of multimedia streams and aperiodic signals are produced in one location (local or remote) and are consumed in another location. Multiple streams are synchronized with each other, are transformed, and the system reacts to asynchronous events. An aperiodic signal interacts with other streams or signals, transforms a stream (or group of streams), and triggers another chain of events. Media types react to external stimuli (user intervention) or their own content, or to some other media type content. Repetition of the same actions does not guarantee the same reaction due to the change in context, past events, or the order of events.

Distributed reactive multimedia systems must be able to react when certain conditions are met. The reaction of the system consists of the generation of one or more events that "respond" appropriately to the presence of some previous phenomena, which consists of the satisfaction of a set of Boolean conditions.

In distributed reactive multimedia system interactions with remote locations is a necessity. Consider an on-line conference in which participants draw on a shared whiteboard object. The drawing must be visible to all of the participants immediately. Actions which do not directly require media streams' attributes modification, should be able to deal with network issues, mobile users, security and alarm exceptions and resource management.

3. The Conceptual Model of TANDEM

The application module of TANDEM is based on the conceptual model for the creation of distributed reactive multimedia systems introduced in [5]. The distributed multimedia application is modeled as reaction graphs (see Figure 1) as follows:

(i) Media generation points are modeled as *sources*, and media rendering (or archival) points are modeled as *sinks*. Sources and sinks can be local or remote. A *sink* is a URI where the multimedia stream is rendered or stored.

(ii) *Transformer* nodes (see Figure 2) apply transformation functions to modify the attributes of the multimedia stream. For example, transformers change the rendering rate of a group of streams, multiplex streams, or reduce the number of channels of a video for rendering.

(iii) *Triggers* provide a general mechanism for actions after a set of Boolean conditions are met. Trigger nodes control the media streams and initiate reaction to the streams. A trigger is associated with sets of multimedia stream groups and a set of conditions. Triggers activate events in response to the satisfaction of a set of conditions. A trigger reaction includes monitoring of external sensors, transforming the attributes of multimedia streams, redirecting a stream to a different destination, starting a new thread of computation, or a cascade of triggers and events. A trigger reaction might require a computation which involves the input streams; in that case the input streams are pre-transformed by one or more transformer modules.

(iv) *Active repository* nodes are associated with triggers. The active repository samples and analyzes the media content for the required conditions, and transmits the outcome to the triggers. An active repository detects and archives *partial conditions* to match the conditions in loose order of occurrence.
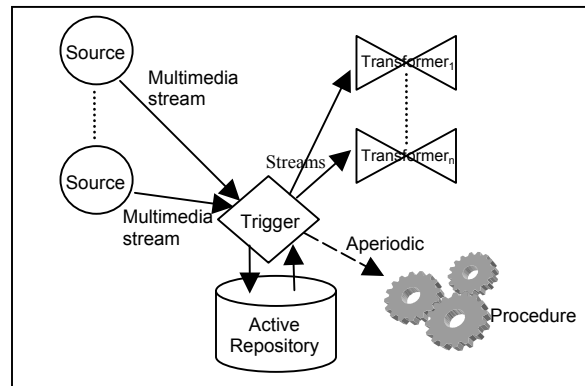
Figure 1. The conceptual execution model

We distinguish two types of triggers: *event-based triggers* and *periodic triggers*. Event-based triggers fire when some constraints (or conditions) are satisfied. For example, the alarm activation of a surveillance camera fires a trigger when a human figure is detected. Periodic triggers fire periodically. For example, a biologist capturing the blooming of a flower will capture and transmit pictures periodically. Triggers are also classified as *continuous* or *discrete*. *Continuous* triggers do not require resetting, while *discrete* triggers are reset every time they go off. Triggers are associated with zero or more streams and aperiodic signals. The trigger communicates with an active repository to archive and retrieve partial conditions and allow nondeterministic ordering of conditions in multiple streams If the reaction requires a computation which does not involve the input streams, the trigger generates an aperiodic signal which causes a procedure-call associated with the activated event, indicated by the gears.
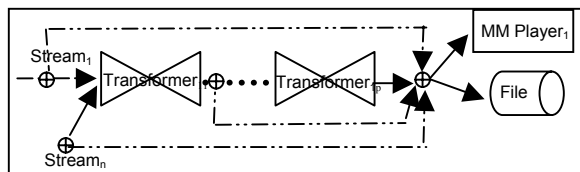
Figure 2. Application of transformers

Groups are used to perform operations on sets of related multimedia streams. Operation on multimedia streams can be either *'isolated'* or *'joint'*. Isolated operations do not affect individual media streams in a group. Joint operations affect every member of the group. Synchronization of streams is a joint operation. Multiple groups can be associated with a trigger. Although a part of triggers, *aperiodic* signals are not part of groups.

Together with the input signals, i.e. groups of periodic multimedia streams (audio, video, text, audiovisual etc.) or aperiodic signals, and the active repository, the trigger forms the basic building block to generate an *event*. The generated events, represented by the dotted links in the above figure, perform the reaction to the specific *conditions* verifications. Event definitions in the trigger define all the conditions that must be verified for the event to occur. The destination of an event can be a *transformer*, a *procedure, or a sink*. A *procedure* is activated when computations (other than stream transformation) are required. Synchronization is required after detecting a required pattern(s) in the active repository. The active repository signals the trigger for the verification of the *partial conditions*, and the action.

4. TANDEM: Framework and Constructs

In this section we introduce generic constructs. Since TANDEM is an Internet language the constructs follow the XML style; however, constructs are generic.

A distributed reactive multimedia application is constructed by specifying the sources, the sinks, the triggers, the transformers, the active repository, and the events. We first define the streams involved in the application. The streams will be input to one or more triggers. Each trigger will contain the description of the groups that participate in the trigger, the loop constructs that we apply to the input streams, the events that the trigger will handle and the destinations of those events. We describe the destinations of each event in the trigger.

The outline of an application framework is:

```
<application ...>
# we describe all the media streams
   <media_stream ...> # mediastream 1 </media_stream>
   ...
   <media_stream...> # mediastream M </media_stream>
# we describe all the triggers
   <trigger ...>
   # define all the loops involving media streams
      <loop ...> ...# loop1 </loop>
      ...
      <loop ...> ... # loopN </loop>
   # define all the groups
      <group ...> ... # group 1 </group>
      ...
      <group ...> ... # group G </group>
   #define events, conditions and trigger destinations
      <event ...> ... # event 1 </event>
      ...
      <event ...> ... # event E </event>
   #describe transformers, actions and destinations
      <transformer ...> ... # transformer 1 </transformer>
      ...
      <transformer ...> ... # transformer S </transformer>
   #describe all the procedures required
      <procedure ...> ...# procedure1 </procedure>
      ...
      <procedure ...> ...# procedure P </procedure>
   </trigger>
   ...
   <trigger>... # trigger T </trigger>
</application>
```

Figure 3. An outline of the application framework

The definition of the multimedia streams in the trigger provides a way to collect information about all the sources (remote or local) needed for the application. For example, a remote source that is sending an audio at a 44,100 samples/sec rate collects the data and values of the attributes as follows:

```
<media_stream name = "mm1">
   <source name = "source1" URI = "192.168.2.102" />
   <type> <audio name = "audio.wav"
       samples_per_sec = "44100" no_of_channels = "2"
       bits_per_sample  = "16"
       scaling_constraint  =  "[0.5, 2]" />
   </type>
</media_stream>
```

Repetition of streams is performed in the *loop construct*. The construct contains the name of the loop and defines the elements, which participate in the loop. For example, the following construct defines a repetition of stream *mm1* 3 times.

```
<loop name = "mm1loop"  times  =  "3">
        <loop_element name  =  "mm1" />
</loop>
```

Nested loops are defined by referencing the name of a predefined loop. The following code repeats *mm1loop* five times.

```
<loop name = "nestedloop"  times  = "5">
        <loop_element  name  =  "mm2" />
        <loop_element  name  =  "mm1loop" />
</loop>
```

4.1 Event constructs

An event is generated when specific conditions are satisfied. An event has a destination according to the action that the event must perform. The events are either tightly integrated with the spatial and temporal constraints or they may loosely coupled. In a tightly integrated constraint the order of the events is very specific and the temporal and spatial constraints are strictly followed. Loosely coupled events may have a non-deterministic order of events with more relaxed constraints.

In the following code fragment, the event "start_video" starts a video clip if all the partial conditions are satisfied. The partial conditions verify the presence of two aperiodic signals: user's right-clicking and the end of another video clip. After both the conditions are satisfied the event is generated. The destination of the event is the "transformer1", which takes as input data stream from the source = "video1".

```
<application name = "example">
...
  <event name = "start_video"
       start  = "0" priority = "1"... >
     <partial_condition name = "cond1"
          signal_type  =  "rightclick"
          presence  = "present" ... />
     <partial_condition cond_name = "cond2"
          signal_type = "video2_end"
          presence  = "present" ... />
     <destination name = "transformer1"
          source  =  "video1"... />
  </event>
...
</application>
```

In TANDEM, the generated events can involve multiple sources or multiple groups of multimedia streams. A generated event is sent to one or more destinations. Remote location interaction can be performed by using multiple remote destinations. The ordering of the events in an application is handled by an optional priority value. The priority, when specified, guarantees the partial order of the events.

4.2 Group constructs

Grouping clusters one or more media streams or groups for synchronization. The groups are both *dynamic* and *hierarchical*. Groups are an elegant and efficient way to specify synchronization on multiple streams as follows:

```
<group name = "soprano">
 <member name =  "mm1"/> <member name = "mm2"/>
</group>
```

The construct creates the group *"soprano"* which groups two media streams, *mm1* and *mm2*. Groups can be modified dynamically by a transformer using group actions as follows:

(i)   *ungroup*, to separate an existing group,
(ii) *add_group*, to add elements to a group
(iii) *delete_group*, to eliminate elements from a group
(iv) *regroup*, to incorporate elements into a new group

In the example described below, the transformer *"ungroup-soprano"* ungroups the elements *mm1* and *mm2* of the group *soprano*. After ungrouping the group *soprano* does not exist. Regrouping is needed for group reconstruction before further use.

```
<transformer name = "ungroup-soprano">
   <action >
     <ungroup > <elements group = "soprano"/>
     </ungroup>
   </action>
   <destination name = "player1" />
</transformer>
```

4.3 Synchronization constructs

Synchronization in multimedia requires the ability to relate the elements involved in the multimedia application both spatially and temporally. The synchronization specifies how those elements will be presented in a specific spatial or temporal order on the rendering device. The required synchronization can be influenced by the events caused by the user interacting with the multimedia system.

The synchronization constructs (i) provide synchronization in the presence of external constraints, such as user interactions, (ii) provide synchronization in the presence of system environmental constraints, such as network traffic or resource availability, and (iii) provide synchronization for user defined media groups.

A *trigger* activates an event every time specific conditions are satisfied. The activation can be started immediately or delayed. In construct (1) the trigger starts

the event immediately, in construct (2) the delay is 9 units. The delay can be determined by a computation as shown in construct (3).

```
<event name = "start_video" start = "0" > ... </event>  (1)
```

```
<event name = "start_video" start = "9" > ... </event>  (2)
```

```
<event name = "synch&start" start = "0"
    var_delay = expression > ......</event>           (3)
```

An event might need to invoke a computation. For example, an event might be delayed depending on conditions of network traffic. An event "invoke_compute" is going to start a procedure "compute_delay" which checks for the network conditions and calculates the appropriate delay. This is done within a period trigger as shown below:

```
<event name = "invoke_compute" start  =  "0">

    # test the presence of the signal "procedure_start"

    <condition name = "cond2" signal_type =
        "procedure_start" presence  =  "present"/>
    <destination name = "compute_delay" />
</event>
```

```
# check and compute the network delay
```

```
<procedure name = "compute_delay" start  = "0">
    <parameters> <param name = "delay"  type = "int"
        mode = "out"/>
    </parameters>
</procedure>
```

Whenever synchronization is required an event must be generated to activate the appropriate reaction. The trigger identifies the specific transformer, and the synchronization is activated. The transformer must contain the synchronization actions to be performed. Since transformers alter the attributes of multimedia streams both spatially and temporally, the synchronized actions must distinguish between spatial and temporal synchronization. For example, consider two media streams are played in *parallel* such that their *start together* and *end together* match. Then the trigger will start an event that will have as destination a transformer that performs the synchronization.

In the following construct, the transformer starts the two streams *mm1* and *mm2* in parallel: *mm2* will start 3 units later than *mm1* and will end at the same time. The stretch required to perform the synchronization must be compatible with the previously defined scaling constraints. If stream scaling constraints are not verified, no modifications will be performed on the stream. The scaling type construct 'Stretch' causes the speed up or the slow down of the playback rate of the media object.

```
<trigger name = "trig1" ...>
...
  <event name = "parallel_start"...>
```

```
...
    <destination name  = "transf1"/>
</event>
```

```
...
<transformer name =  "transf1">
  <action>
    <synchronize>
      <temporal  type  =  "start&end" scaling = "stretch"
          reference  =  "mm1" start  = "0" end = "0">
        <elements name  =  "mm2"  diffstart = "3"
          diffend = "0"/>
      </temporal>
    </synchronize>
  </action>
  <destination name  =  "player1" />
</transformer>
...
</trigger>
```

If the parallel start is applied to a group of streams, the group elements are related synchronously to the reference stream. In the code fragment below, all the media streams of *group1* start 3 units later than *mm1* and terminate at the same time, while all the media streams of *group2* start after 2 units and terminate 2 units earlier with respect to the stream *mm1*.

```
<synchronize>
  <temporal type = "start&end" scaling = "stretch"
      reference = "mm1" start  = "0" end  = "0" />
    <elements name = "group1" diffstart  =  "3"
      diffend = "0" />
    <elements name = "group2" diffstart  =  "2"
      diffend = "-2" />
  </temporal>
</synchronize>
```

Synchronization actions involving spatial constraints are computed by giving the relative values. The following code locates all the elements of *group1*, on the X-axis 3 points after the X-position of *mm1*. Synchronization is handled by the synchronization type "seq_start" as follows:

```
<synchronize>
  <spatial type =  "seq_start" scaling  = "stretch"
      alignment  = "upper_left">
    <elements name  =  "mm1" diffstartX  = "0"
      diffstartY = "0" diffendX = "0" diffendY = "0"/>
    <elements name  =  "group1"  diffstartX = "3"
      diffstartY = "0" diffendX = "3" diffendY = "0"/>
  </spatial>
</synchronize>
```

5. Related Works

The research involving synchronized multimedia streams [8] and the use of high level distributed multimedia language constructs such as event triggering and synchronization constructs for flexible Internet based

modeling is evolving rapidly after the advent of the Internet. The languages like SMIL [9], TAOML [1], VRML [10] and its XML based variants, synchronous language Esterel [3], and distributed multimedia languages concerned with QoS (Quality of Service) [6] have different aspects of multimedia modeling, synchronization constructs and event based constructs.

SMIL [9] models concurrent multimedia streams by synchronizing start and end of the streaming at a specific point of time relative to other streams, and supports spatial synchronization and placement of media streams. However, there is no comprehensive stream group construct in SMIL, and SMIL also does not support frame level synchronization due to the lack of synchronization of periodic signals. The event model (used primarily for user interaction) is independent of the timing model (used for playback). If no events are defined by a host language, event-timing is effectively omitted [9]. For example, if a video is started by the left-click of the mouse and is ended by the right-click of the mouse, in SMIL we cannot guarantee that the left-click followed by the right click results in the media starting and stopping. SMIL also does not support nondeterministic order of events with partial matching of conditions.

VRML [10] and its XML based variants support the notion of events and events triggering other events, triggering computation, and grouping of multimedia components. The events can be modified dynamically in VRML. However, VRML also does not support synchronization of periodic stream at the frame level, and does not support nondeterministic order of conditions or archiving partial conditions.

TAOML [1] has limited expression capability. For example, nested loop or dynamic grouping is available neither in TAOML nor in SMIL. Other languages, such as HQML [6] have focused their attention on the quality of service capabilities; therefore their synchronization abilities are very limited.

ESTEREL [3] is a synchronous deterministic hardware modeling language. Integrated with distributed multimedia sources and sinks constructs, Esterel constructs can be used to model multimedia objects and periodic multimedia streams, and the triggering of a chain of multimedia events. However, Esterel is not an Internet based language, and does not support explicit grouping of periodic streams and nondetermistic ordering of events or storing of partial conditions.

6. Conclusions

In this paper, we have described a distributed multimedia modeling language TANDEM that integrates deterministic and non-deterministic events, asynchronous events, and spatio-temporal synchronization. The language is based upon a novel model [5] developed by the authors which uses a persistent active repository. The active repository supports pattern-based matching and content-based analysis of media streams. The use of the persistent active repository allows us to store partial conditions and relax the order of events. The language has five distinct types of constructs, namely, multimedia definition constructs, group constructs, trigger constructs, synchronization constructs, and transformer constructs. We support multiple stream synchronization using group constructs, support aperiodic signals for user interaction, and continuous streams for sensor information. Trigger constructs are used either to invoke media transformers that alter the attributes of media streams or invoke a procedure call for computation. The language is in an advanced stage of implementation.

References

[1] T. Arndt, S.K. Chang, A. Guercio, "Formal Specification and Prototyping of Multimedia Applications," *Int. Journal of SEKE.*, vol.10, no.4, pp.377-409, 2000.

[2] J. Bacon et al., "Generic Support for Distributed Applications", IEEE *Computer*, pp. 2-10, March 2000.

[3] G. Berry, G. Gonthier, "The ESTEREL Synchronous Programming Language: Design, Semantics, Implementation," *Science of Computer Programming*, vol. 19, no. 2, pp. 87-152, Nov. 1992.

[4] X. Gu, K. Nahrstedt, "An Event-Driven, User-Centric, QoS-aware Midleware Framework for Ubiquitous Multimedia Applications", *Proc. of ACM Multimedia*, Ottawa, Oct. 2001.

[5] A. Guercio, A. K. Bansal, "A Model for Integrating Deterministic and Asynchronous Events in Reactive Multimedia Internet Based Languages", to appear in *IC 2004, Procs. 5th Int. Conf. on Internet Computing*, Las Vegas, June 21-24, 2004.

[6] X. Gu, et al, "An XML-based Quality of Service Enabling Language for the Web", *Journal of Visual Languages and Computing*, vol.13, no. 1, pp. 61-95, 2002.

[7] D.-H. Kim, K.-H. Lee, "An Extended Object Composition Model for Distributed Multimedia Services", *Procs. of the 7th International Workshop on Object Oriented Real-Time Dependable Systems*, January 2002, pp. 279-288.

[8] P.V. Rangan, S. Ramanathan, S. Sampathkumar, "Feeback Techniques for Continuity and Synchronization in Multimedia Information Retrieval", *ACM Transactions on Information Systems*, vol. 13, no.2, pp. 145-176, 1995.

[9] *Synchronized Multimedia Integration Language 2.0 Specification*, http://www.w3.org/TR/smil20/, Aug. 2001.

[10] *Virtual Reality Modeling Language(VRML) 2.0*, ISO/IEC 14772,http://www.web3d.org/x3d/specifications/vrml/ISO _IEC_14772-All/index.html.

[11] *Extensible Markup Language*-http://www.w3.org/XML.

Visual Editor for Designing and Editing Algorithmic Skeletons

Rentaro Yoshioka[1], Yuho Tsuchida[2], Nikolay Mirenkov[1]
[1]University of Aizu, Aizu-Wakamatsu, 965-8580 Japan
[2]Unisystem Co. Ltd., Tokyo, 171-0031 Japan
rentaro@u-aizu.ac.jp

Abstract

A new type of multimedia editor for designing algorithmic skeletons in a cyberFilm is presented. A cyberFilm is a new multimedia format to represent and edit computation through a set of algorithmic features. The algorithmic skeleton is one of these features, in which the algorithmic structure and the associated flow of activities are described. The algorithmic skeleton is represented by a series of stills and scenes. Each still consists of multimedia symbols such as images, text and sounds, and corresponds to one computational step. A scene is a subset of stills which can be considered as related steps and meaningful unit of an algorithm. The use of multimedia symbols in the stills and the composition of stills into scenes must follow the syntax of the cyberFilm language. The design of the skeleton is performed by editing these multimedia symbols in the stills and scenes. This process can be extremely simplified by a special editor supporting high-level operations and intelligent interfaces. This editor allows even a novice user to draw a picture conforming to the language and simplifies the task of extracting and transforming the user's computational intention into the algorithmic skeleton images and animations. In this paper, the different algorithmic features constituting the algorithmic skeletons are described and a number of sub-editors to edit those features are presented.

1 Introduction

Although, computer technology is needed in various fields, programming productivity is insufficient for the demand of the industry[1]. A main cause of this problem is in programming languages. Current major programming languages such as C, C++, Java, C# are text based representations. There is a great gap between their syntax and corresponding semantics. We must encode our mental image of computation into source code in special, strict syntax, and perform the opposite decoding to bring it back to the mental image-level again to understand it. The encoding and decoding tasks require special programming knowledge and description techniques. To master these languages, much time and effort are required, because this type of languages is closer to computer languages and are very abstract. The program written in a text-based language is difficult to understand because they are based on very abstract concepts and notations and meaning of computation is not visible. The difficulty of understanding causes the difficulty of editing, debugging, and reusing.

To solve these problems, various visual languages and environments have been created. Many of these approaches are based on representing modules and components by an icon and allow compositions at level of the icons. They do not visualize internal structures and computations performed inside the module. Also, many visual environments, such as, LabVIEW [2], MATLAB and Simulink [3] express computation by a network of modules as 2-dimensional graph structures, but this is not always the best structure to represent computation. For example, when it is necessary to express processing on meshes and matrices, grid structures are more suitable, and when it is necessary to express hierarchical structures or processes, pyramids and trees are more suitable.

Our approach is to develop and use self-explanatory components for software component representation [4]. A self-explanatory component represents itself with multiple views related to its dynamic, static, and hierarchical features. Features are represented with images, animation, sound, and text. A cyberFilm format is for the representation of multiple views. A cyberFilm contains a set of stills. A still is a view of an objects/processes feature. With multiple view, the user can understand the meaning of the component more deeply and easily. The ease of understanding helps editing, debugging and reusing.

In the cyberFilm, the computation is presented by six representative aspects [4]. The algorithmic skeleton is one of such aspects to show algorithmic computational steps and data structures. Algorithmic skeletons are represented as a series of colored stills. Each still represents one computational step by image symbols. Defining algorithmic skeleton is designing stills with image symbols. However, the editor has to support not only functions to draw a picture like conventional drawing tools do but

higher level functions to extract and transform the user's computational intentions into the algorithmic skeleton stills.

2 Related Work

Conventional description methods for computational algorithms can be divided into two groups with respect to their orientation: computer-oriented methods and user-oriented methods. In other words, the computer-oriented methods place higher priority on program execution while the user-oriented methods place higher priority on explaining programs and algorithms.

A typical example of the first group is the programming languages that most programmers use today, such as C, C++, Java, etc. The merit of these languages is that they are low-level enough to control the very details of the computer and that source code can be efficiently transformed into executable machine codes. The downside is that they are often abstract, implicit, and strict in syntax that it requests the programmer's full attention and hard training to read, write and debug the source codes [5][6]. In this sense, they are not good tools for designing and editing algorithms.

A typical example of the second group is the various tools for algorithm animations and standards for explanatory diagrams. Algorithm animation is a popular field in computer science. Although there are many Java applets to represent algorithm animations on web sites [7], there are a few that allow editing existing animations. Non editable animations are still useful, but it will be much better if existing algorithms could be edited to create new algorithms or to increase understanding by watching how an algorithm changes when some features are modified. Still, tools specialized for animation and presentation creation, such as Macromedia Flash and Microsoft PowerPoint, seem to be accepted widely. These tools have advantages in creating, editing and presenting algorithm animations, but with a serious limitation. Since these tools have no notion of algorithms, a user cannot be controlled on which representation rules to be used or be helped in specifying or modifying a specific feature of an algorithm.

It is important for software components to be both computer-oriented and user-oriented. We need new formats, languages and supporting environments to better support both the user and computer environments in specifying and editing computational algorithms. The cyberFilm format and Active Knowledge Studio [8] are such a format and environment. The work presented in this paper extends the previous results related to cyberFilms by adding two new groups of features to the component. Also

the results add a major new functionality (a cyberFilm editor) to the Active Knowledge Studio. Nevertheless, the results presented in this paper alone can be applied to new software design tools and educational tools for self-learning or distance learning.

3 Language for Algorithmic Skeletons

As already mentioned, an algorithmic skeleton is a set of stills that represent the algorithmic steps of computation. A computational step is visualized as a combination of visual symbols in a still. Such a step is easier to understand within a scene, a special sequence of such stills. This representation method is the same as in conventional animation films. The difference from conventional animation films is that algorithmic skeletons are drawn in the cyberFilm language[4] and can be manipulated through simple operations.

In the cyberFilm language, computational steps are considered as a combination of a structure and a scheme. The structure is an algorithmic shape of a computation. In the still, the structure is shown as a parameterized set of nodes connected by links in 4D space-time. The structure can be of any geometrical construction in a 3D space. The scheme is a computational plan which describes the flow of activities on the structure. The activities are represented as flashing of nodes. There are a number of flashing types. For example, a full flashing node points that operations attached to this node are executed. The attached operations are characterized with flashing colors of nodes. Different colors are to represent different operations.

(a)

(b)
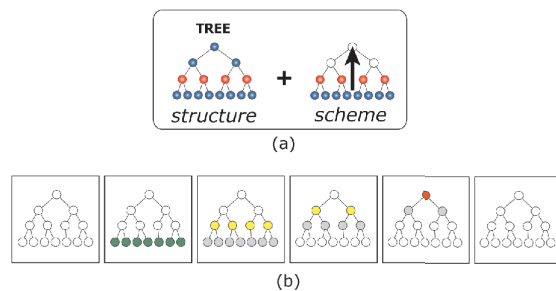
Figure 1 Example of an algorithmic skeleton

Figure 1 is an example of the algorithmic skeleton. It is a combination of a tree structure and a layer-by-layer bottom-up traversal scheme (Figure 1(a)). The resulting stills are shown in Figure 1(b), where the stills are placed in sequential order from the left to right.

3.1 Scene and Still Editors

The purpose of the scene and still editor for algorithmic skeletons is to design algorithmic computational steps by

designing stills conforming to the cyberFilm language. Of course, it is possible to use the existing tools to create similar animations; however, there is no guarantee that the animation created will be correct and suitable as a cyberFilm. By using this editor, a designer will never design "grammatically" incorrect animations because the editor does not allow such operations. A user's computational intention can be received and recorded. The editor can provide explicit and implicit support in designing and editing correct algorithmic skeletons efficiently.

The scenes and stills editor has two interfaces. One is for editing a sequence of stills. This interface allows a user to modify the skeleton in units of schemes, scenes or stills. An example of this interface is shown in Figure 7, and will be described later in section 3.3.

Figure 2 Interface for editing contents of a still

The other interface is for editing the contents of a still, which is shown in Figure 2. The interface consists of three parts: a target frame in the upper-left area, a target selector in the bottom area, and sub-editors in the upper-right area.

The target frame displays a scene or still for editing. The

Figure 3 Target Selector

name of the cyberFilm and the scene and still numbers of the target are displayed on the top part. In the figure, the still is the 5th still in scene 1 of the "MST-prim-fukushima" cyberFilm. There are two buttons in the top-left area, which are buttons to decide show/hide foreground/background items. A user can directly select items in a still by clicking on it with the mouse.

The target selector (Figure 3) is used to select a scene or a still. It consists of two parts. The upper row displays the list of scenes, and the lower row displays the list of stills in the selected scene. A user should just click on the desired scene or still to select it. The selected scene or still is emphasized and displayed in the target frame. This selection switches the mode between scene-edit mode and still-edit mode depending on which was selected.

The scene-edit mode is used to edit items which do not change within a scene according to the cyberFilm grammar. Such items are structures, color and shape of nodes. During this mode, the modifications are reflected to all stills in a scene. This mode is also used to attach foreground and background elements to all stills in a scene.

The still-edit mode is for specifying each still. This mode is used mainly for editing the flashing of nodes. This mode is also used to attach foreground and background elements to a still.
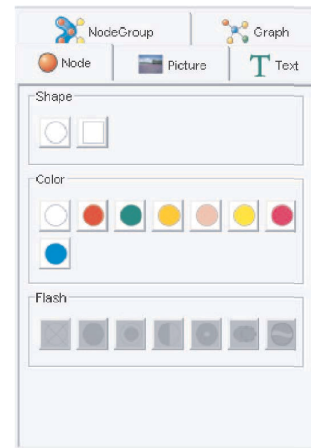
Figure 4 Sub-editor panel

The sub-editors are placed in the upper-right area. Users can switch between subeditors by clicking on the tab labeled by a title and icon. The current set of sub-editors is as follows: structure editors, node editor, picture editor, text editor and node-group editor. Figure 4 is an enlarged view of the sub-editors panel, with the node editor activated. Details of each sub-editor are described in the following sections.
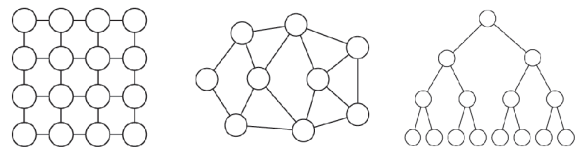
Figure 5 Example of structures

3.2 Structure Editor

Structure Editors are used to modify the size and form of structures. A structure can be of any geometrical 3D construction. Figure 5 shows examples of some structures. There is one editor for each type of structure since special knowledge must be embedded to support structure specific operations. For instance, a 2D-grid editor has the following operations: add-row, add-column, remove row, remove-column, etc. On the other hand, a tree editor has the following operations: add-left-child, add-right-child, add-children, remove-sub-tree, etc. By interacting with these structure editors, a user can modify the size and form of the structure without the danger of performing operations that will result in a change in the structure type.

When a cyberFilm is loaded, an appropriate editor will be loaded automatically according to the structures used in that cyberFilm. In Figure 6, a general graph editor has been loaded in the editing window.

To move a node, the user should first click on the "move node" button, then click and drag the desired node. Other
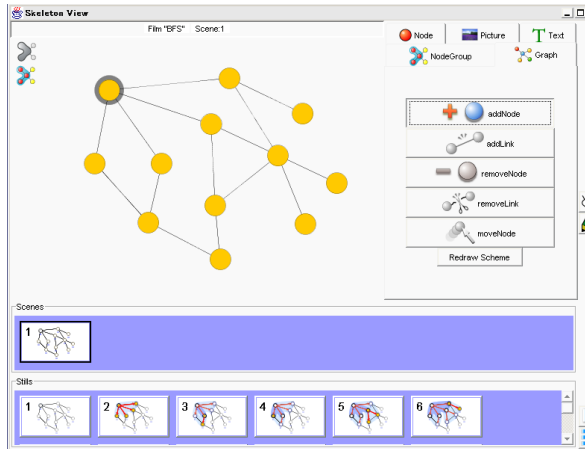
Figure 6 Structure Editor
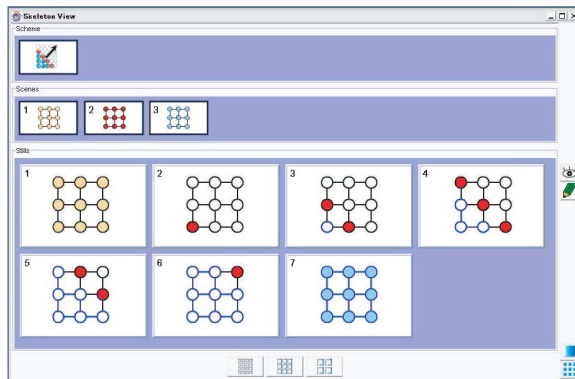
Figure 7 Scheme Editor

operations can be performed in a similar way. The links between nodes are automatically adjusted. When a node is deleted, the unnecessary links are also removed. The result of editing the structure is immediately reflected on all stills in a scene, so that the user does not have to modify every single still.

Note that the foreground and background items are displayed while the structure is being edited. This is to assist users who want to precisely align some node with specific points of a background image or foreground items. Of course, they can also be "hidden" if the user wishes so.

3.3 Scheme Editor

Schemes can be edited in four different levels of detail: the scheme-level, scene-level, still-level and the node-level. For the first three levels, the interface shown in Figure 7 is used.

The scheme-level editing is for completely exchanging the current scheme with another. This means that all scenes and stills are redrawn according to the new scheme. An important point is that the structure will not change. So a user can apply various schemes to the same structure and immediately observe the changes in the flow of computation. The available schemes for a given type of structure will be presented by the editor so that the user need not worry about selecting a scheme that cannot be applied to that structure. For example, tree type schemes are only for a tree structure and cannot be applied to grid structures.
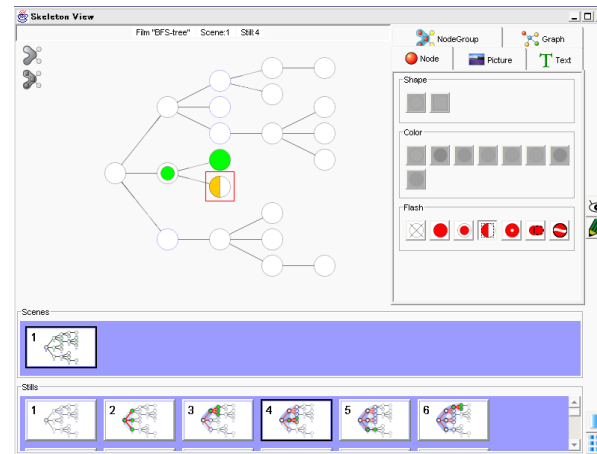
Figure 8 Node Editor (Still-edit mode)

For the scene-level and still-level editing, delete and move operations are supported. At the scene-level, stills can be deleted or moved in units of scenes. At the still-level, indivisual stills can be deleted and moved. Basic functionality of this type of editing has been introduced in [8]. In the current version of the editor, the editor has been

redesigned to judge the validity of move operations. When a still is moved from one scene to another, or from one scheme to another, the editor must check if the still can be added to the destination scene or scheme.

The most detailed editing is the node-level editing. The interface shown in Figure 8 is the node-level editor. In the upper-right of the interface, the sub-editor for editing node attributes is displayed. This editor shows different sets of buttons corresponding to the two modes of editing: scene-edit and still-edit modes. As already mentioned, the target selector is used to select a scene or a still, which in result selectes the mode. When in the scene-edit mode, the sub-editor shows operations to change the color and shape of a node. When in the still-edit mode, it shows operations to change flashing of a node (Figure 8).

3.4 Foreground and Background Editors

Currently, there are three separate sub-editors for editing the foreground and background elements.

1) Picture Editor

The picture-editor is used for adding, moving and deleting images in stills. Images do not affect the algorithmic skeleton, but are very important items that can reduce the level of abstraction and help a user's understanding by providing concrete hints for understanding the skeleton. Figure 9 shows the effect of using an image as a background of a structure. A structure without the image is shown on the left and the same
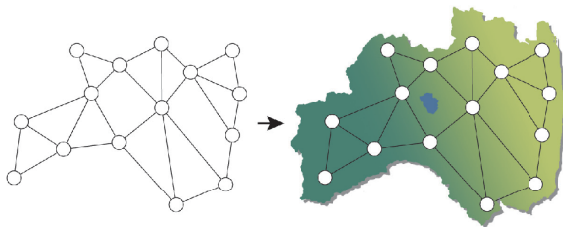
Figure 9 Background images

structure with an image. Like this, additional images increase the understandability and the concreteness of the algorithmic skeleton. By changing the image, different variations can be presented without changing the structure and/or the scheme. The picture editor is show in Figure 10. Images can be set in the front or the back of a structure. A number of images can be placed into a still. The current editor supports images in gif, jpeg and png formats. The image can be set to a scene or a still. When an image is set to a scene, it is set to all stills in the scene in the same size and position. Setting images to a scene is useful for attaching additional hints to items which do not change in the scene, for instance, to attach geometrical meaning to a structure. On the other hand, setting images to a still is also
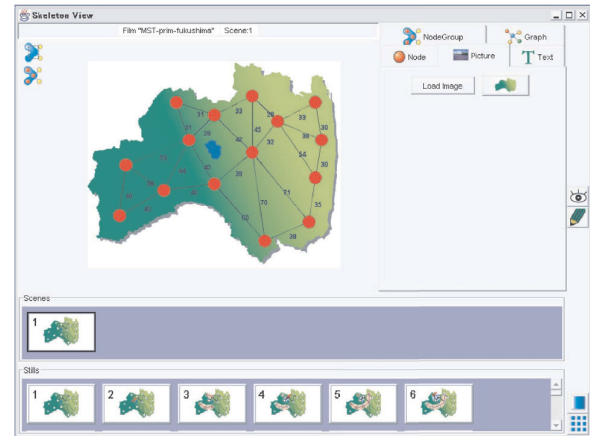
Figure 10 Picture Editor

useful for providing hints only to one computational step.

2) Text Editor

The text editor allows the user to add text to stills. The font type, size and color can be changed. As the same with images, text also does not influence the algorithm itself, but has big influence on the explanation of animations. A moderate quantity of explanatory comments will help users to understand the meaning of animation quickly. It will also be possible to attract attention of users by putting notes only on the stills of important computational steps. Further more, the concreteness of the animation can be increased by attaching captions showing names or values to nodes and links. Figure 11 shows the text editor's interface and the target still with text and a background image. The editor consists of a text-field for typing text, a combo-box for selecting font type, a button for changing the color, and a slider for changing size of font. In the still, a descriptive comment of "Breadth first search" and a
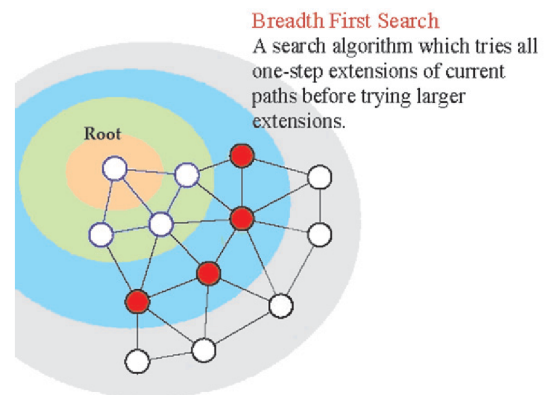
Figure 11 Adding texts to a skeleton

caption for the root node has been attached.

3) Node Group Editor

In the still, special background shapes with color are used to represent subsets of nodes. Different subsets are represented by different colors. Figure 12 depicts a still of the Prim's algorithm. There are two subsets of nodes. The "blue" background shows nodes that are already connected to the minimum spanning tree. Nodes with "pink" background are candidates for the next connection. Details of the description can be found in [9]. The node group editor is used for drawing such background shapes. The interface is represented in Figure 12. The button on the top of the sub-editor interface is for creating a new background shape. The color of the shape can be selected from a listed of un-used colors. The editor prevents the user from adding two groups with the same color. Below this button is a list of buttons corresponding to each declared node group of the still. The modification of a node-group is started by clicking one of these buttons. After selecting a group to edit, the user can add a node to the group by clicking the node on the still. If the user clicks a node which is already in the group, the node is removed from the group. The background shape is modified automatically.

A name and a color of the selected node-group are displayed under the group-list. The name and color are changeable. The name of the group is optional information. Groups are differentiated only by color, but the name will be helpful to add a meaning to a group. In Figure 12, the "blue" group is named "Connected", and the "pink" group is named "Candidates".
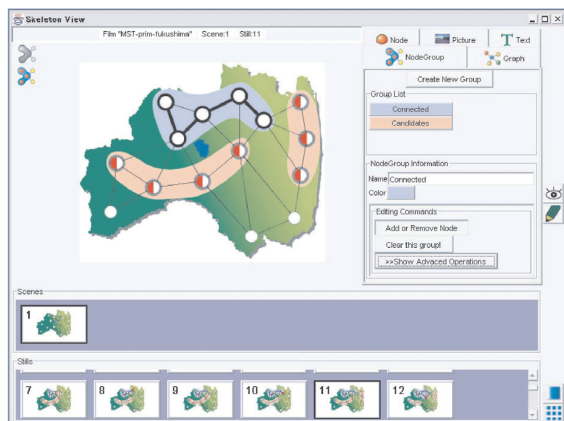
Figure 12 Node group editor

4 Conclusion

A visual editor for designing and editing algorithmic skeletons of computation has been presented. An algorithmic skeleton is defined by a structure and flows of activities on the structure. The algorithmic skeleton is one of the various algorithmic features used in the cyberFilm format to represent computation. In the cyberFilm, algorithmic skeletons are represented by a series of stills and scenes, where each still consists of multimedia symbols such as images, text and sound. The editor is part of a larger programming environment to create and edit software components in the cyberFilm format, and supports the user in designing and editing "grammatically" correct skeletons with high-level operations and intelligent interfaces. In other words, the user does not have to worry whether the design is correct or not: All operations allowed by the editor are guaranteed to be correct. Another characteristic of this editor is that it consists of sub-editors to support the different sub-features of the algorithmic skeleton: structures, schemes, and foreground/background elements. This allows adding new editors in the future to support new types of algorithms, which is important since each structure editor is tailored to a specific structure to provide maximum support to the user.

References

[1] S. Hallem, D. Park and D. Engler, Uprooting Software Defects at the Source, Queue, Vol.1, No.8, 65-71.

[2] Labview, http://www.ni.com/labview/

[3] The MathWorks: MATLAB and SimuLink for Technical Computing, http://www.mathworks.com/

[4] R. Yoshioka, N. Mirenkov, Y. Tsuchida, Y. Watanobe, Visual Notation of Film Language System, In: Proceedings of 2002 International Conference of Distributed Multimedia Systems, San Francisco, CA, USA, Sept. 2002, 648-655.

[5] H. Lieberman, C. Fry, Will Software Ever Work?, Communication of the ACM, ACM Press, Vol.44, No.3, 2001.

[6] R. L. Glass, One Giant Step Backward, Communications of the ACM, ACM Press, Vol.46, No.5, 2003.

[7] The Complete Collection of Algorithm Animations, http://www.cs.hope.edu/algaim/ccaa/

[8] R. Yoshioka, N. Mirenkov, Visual Computing within Environment of Self-explanatory Components, Soft Computing Journal, Springer-Verlag, Vol.7, No.1, 2002, 20-32.

[9] N. Mirenkov, Oleg Monakov, R. Yoshioka, Visualization of Graph Algorithms and Programming in Pictures, In: Proceedings of the 6th IASTD International Conference on Software Engineering and Applications, Cambridge, USA, Nov. 2002, 391-397.

Visual Representation and Editing System for Formula Sequence Specification and Variable Declaration

Dmitry A. Vazhenin, Nikolay N. Mirenkov, Alexander P. Vazhenin

Graduate School Department of Information Systems
University of Aizu, Aizu-Wakamatsu, 965-8580, Japan
email: {d8052102, nikmir, vazhenin}@u-aizu.ac.jp

Abstract

This work is a part of the filmification of methods technology which is based on the visual algorithm representation using an algorithmic CyberFilm concept. Within this concept each algorithm is represented through multiple views and a space-time metaphor. Usually, to specify an algorithm, it is necessary to define space structures and traversal schemes for visiting nodes of those structures. In addition, variables attributes of space structures and nodes should be declared, operations (formula sequences) to be performed in visiting nodes should be defined. A main focus of this paper is a special multimedia subsystem for the variable declaration and the formula sequence definition. This subsystem uses a special multimedia language with high-level constructions and operators in order to make the programming process more efficient and comfortable. Enhanced text-based terms, tables, images and stencils are used for representing the arithmetical and logical expressions. The multimedia language syntax and semantics, user interface and some aspects of automated code generation are described in this paper.

1 Introduction

At present, various programming languages are used for software development. They are very powerful and have a plenty of useful features, but they are still more computer oriented rather then human oriented. So, the programming process is complex for people using conventional text-based programming languages, such as C, FORTRAN, JAVA, etc. Representing abstract things in text is very difficult for the programmers because they have to keep in mind a lot of abstract concepts and should implicitly describe complex structures and relations, states and sequences of objects and dynamic processes. All those features are better to watch and explicitly modify rather then keep in mind. Additionally, these features can be classified and considered separately. Otherwise, even the arithmetical and logical expressions embedded into complicated bodies of nested loops and conditional branches become a serious problem of the programming process.

Therefore, various visual programming languages have been developed to enhance traditional programming systems and solve some problems of the text-based programming languages. Visual languages allow represent knowledge in a graphic way, which helps the user to understand pieces of knowledge and manipulate them better. The computer algorithm animation technologies [1-3], Universal Modeling Language (UML) diagrams [4], formal visualization models for component-based software [5], etc, are representatives of the algorithm visualization techniques. The visual iconic data-flow programming languages are another example of visual programming efforts [6,7]. As a special case we would like to mention the animated visual 3D programming language SAM (Solid Agents in Motion) for parallel systems specification and animation [8]. There also several commercial tools for software development based on mathematical objects manipulations [9,10]. Additionally, it is necessary to mention technologies for automated executable code generation from the visual specification [11] and visualization of dynamic and static aspects of a program execution [12].

The most of the systems mentioned are very specialized and focused on solving or demonstration specific problems. As a rule, they represent an algorithm from one or a few points of view that are not enough to understand the algorithm thoroughly. The multiple view

concept in [13] is oriented to show different aspects of an algorithm including structure, operations, data dependencies, etc, as well as to split program development process.

Our approach is to use a self-explanatory visual programming environment where algorithms are represented in a "CyberFilm" format [14,15]. This format is used as a new type of abstraction combining mathematical and physical concepts. A special system supporting the development, learning and use of self-explanatory components in the CyberFilm format has being implemented. This system is called Active Knowledge Studio (AKS) [15].

The CyberFilm concept uses multiple views to represent various features of computational algorithms. CyberFilm has six groups of views representing different features of computation. In Figure 1, algorithmic skeleton view (a) presents dynamical features of the algorithmic steps, variables and formulas view (b) shows activities (formulas with arithmetical/logical expressions) attached to selected nodes of algorithmic structures. Input and output view (c) includes frames related to inter-component and intra-component interfaces, and integrated view (d) consists of frames with a special structure representing a "summary" of algorithm. There are two additional views: registration view (e) contains registration information such as authors, date of creation, etc, and links and statistics view (f) links to related or referenced CyberFilms, other source of relevant knowledge, and some statistical information, such as number of usage, editing, etc.
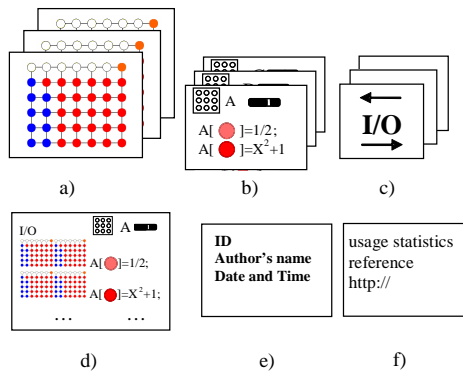
Figure 1. CyberFilm multiple view

Within each group, there is a set of frames representing corresponding algorithmic features. For example, in the algorithmic skeleton view a structure (or a few structures) and scheme of activity on this structure are represented by the frames. The frame structure shows the algorithmic space (and its data) and the series of frames shows the algorithmic activity in time. Usually, frame structures are parameterized sets of nodes and links. It means that, as a rule, to represent (to explain)

corresponding algorithmic features it is not necessary to render large-sized sets. A structure can be any geometrical object in multi-dimensional space, but now we use grids, pyramids, trees, general graphs, matrices and networks as a basis. This paper presents a subsystem of the AKS supporting the development of the b-group frames related to the structure and variables declaration as well as the specification of the activity on nodes.

The a-group frames show an abstract activity on structures by colors, shapes, and types of node flashing. Different colors represent different formulas to be attached to nodes. The shapes depict a hierarchy of nodes; a normal structural node is a circle and a nested CyberFilm is a square. Types of flashing indicate some specific features of operations to be defined on corresponding nodes.

Once the areas of activities are defined, it is necessary to specify concrete computational operations for each type of activity in order to create a working algorithm. These operations are specified via formula sequences attachment process which is described in this paper.

The next section of this paper describes structure definitions and variable declaration mechanism. The third section describes methods and interfaces for the specification and representation of the formula sequence, correctness checking and executable code generation from the formula specification. Section four shows a short example of the formula presentation. The last section contains conclusion and future research topics.

2   Variable specification

2.1   Structure hierarchy

Data structures are one of the key points in programming [16]. So, it is necessary to pay a special attention to data structure selection and definition. The user should be able to choose and define structures easily and have minimal chances to make a mistake. To satisfy these requirements in our system the definition of the data structures is started from the definition of space-time structures. After that we allow a conventional type variable declaration for whole structures as well as for their substructures.

The space-time structures hierarchy is presented in Figure 2. All structures are divided into three classes: *space*, *time* and *observer*. Space-time continuum is the most understandable human environment, because our world consists of 3D space and 1D time. So, the basis of all structures should be 3D space structures and 1D time structure. This level of the data structure abstraction is suitable for visual representation and manipulation. If an algorithm requires more complex data structure, it can be considered as a superposition of subspaces of 3D (such as 1D, 2D and 3D). There are many methods how to represent N-dimensional space; it always can be projected

into 2D or 3D space.

The class of space structures is divided into subclasses according to the structure types: *grid*, *tree*, *pyramid*, *graph*, etc. Examples of space structures are shown in Figure 3, a.
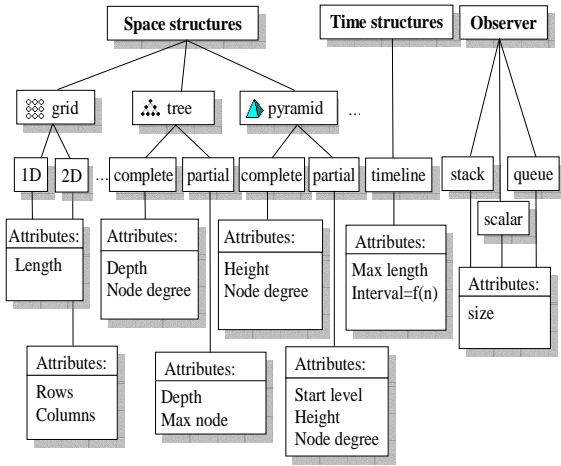
Figure 2. Structures classification

a) space structures

b) zero-dimensional observer structure

c) time structure
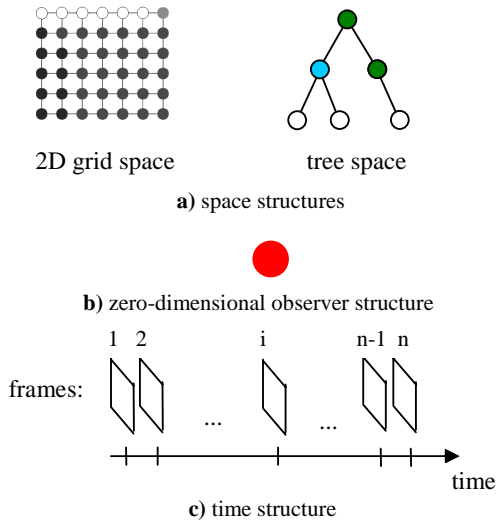
Figure 3. Structure examples

The observer structure is some kind of common zero-dimensional space, a special separated node (Figure 3, b) with indivisible access operations only. They can be defined as operation on scalar, stack, queue, matrix, etc, where all manipulations are "atomic" from the user's point of view. Any existing software component ("black box") may be considered as an operation on such observer variables.

The time structure keeps a history of changes in the variables of space structures. This structure is defined as

1D-timeline where time steps are directly related to frames of algorithmic skeletons (computational steps). Maximum number of steps and size of each step are attributes of the time structure (Figure 3, c). For example, the step size function may be defined as a constant (*f(i)=2* means that the history will contain only every 2nd computational step) or may change during computation (*f(i)=log(i)*, where *i* is a number of computational steps). Time structures and corresponding variables are especially convenient to create and use traces of computation.

2.2 Structure definition GUI

There is a special interface designed to set necessary structure parameters. All structures used in the algorithmic skeleton are appeared as icons and tabbed panels inside a main window (Figure 4).
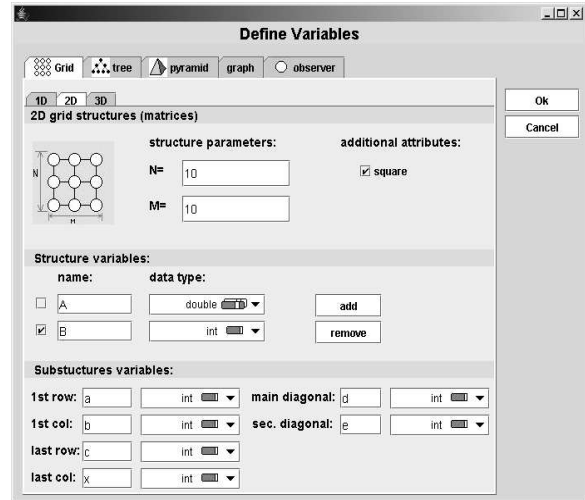
Figure 4. Structure definition and variables declaration window

The user can choose a type and subtype of structure, and enter some numerical attributes. The system in some cases can help the user to define structure correctly. Additionally, some variables may be assigned to the natural substructures (first row, first column, diagonal, etc). A set of natural substructures is predefined for each type of space structures.

3 Representation of computational formula sequences

Algorithmic skeleton shows data structures and some general types of activity on these structures. In order to precisely specify the activity, it is necessary to attach arithmetical and/or logical formula sequences to corresponding nodes. In our definition, a formula sequence is a loop-free sequence of arithmetical and

logical expressions specifying some local activity in active nodes and around them. The expressions are enhanced by using special images, symbols and tables in order to simplify the perception of formulas. Using such representation, the programmer can easier understand relations between a real application and algorithm used for it.

3.1 Index simplification

One reason of the easier understanding above-mentioned is related to possible index expression simplification. It is based on the algorithm design process that is divided into several stages. In the first stage, the user must choose a data structure. Then, in the second stage he/she should design a computation-flow process using skeleton editor tools [15]. In this stage it is possible to select nodes and groups of nodes, where and when some activity should be performed. This selection is supported by special scanning templates. In the third stage, the user should attach a formula sequence to active nodes defined by a scanning template. The formula sequence has several input and output parameters which are necessary to connect the sequence to the scanning template: variable name of the structure, color group id, scanning parameters (i.e. active node coordinates). The output parameters are new node values which were implicitly modified by the formula sequence. All these parameters are hidden, and the user should operate only with visual objects specifying variables and index expressions.

Therefore, a special set of multimedia symbols was developed in order to simplify representation and perception of index expressions. Examples of such symbols are depicted by Figure 5.

3.2 Formula sequence components

Each formula sequence consists of the following parts: expressions, control structures, masks and regular text.

Expressions are necessary to specify data access and manipulations with structure variables. The notation of the expressions is close to the conventional mathematical expressions. Control structures are used to specify branch-forward conditions. The masks are used to disable the node activity as whole. Regular text is used for comments and additional explanations.

3.3 Formula syntax and semantics

The syntax and semantic formalization problems in visual languages are discussed in [17]. Some of them may be solved by using our approach. The formula specification language has several syntactical and semantic rules. Expressions, control structures and text items can be constructed according to these rules.

Masks are defined outside the formula as an additional feature, so they are not included into the syntax. The syntax of proposing language is defined as a set of rules and layouts. Syntactical rules define terms and layouts (Figure 6) to control term visual appearance. Semantic rules are necessary to specify meaning of terms. They are defined using semantic table (Figure 5). The first column of this table contains types of formula symbols; the second column shows visual layout of the symbols; the third, fourth and last columns contains visual and textual examples of formula elements. All terminal symbols have dual semantic meaning for visual and textual representation.

The executable code should be generated according to the substitution rules in semantic tables, and visual objects should be constructed according to these tables also. The semantic table defines correspondence between visual and internal representations. Each type of CyberFilm has its own semantic table (for example, matrix computations and computation on multi-stage networks have different semantics of operations).

The use of the formula object depends on the situation: if it is necessary to display entire formula for watching and/or editing, the visual semantic meaning and layouts are used; if it is necessary to generate executable code, the internal "textual" semantic is used.

3.4 Correctness checking and executable code generation

Syntax correctness checking is rather simple by using syntax diagram and semantic tables mentioned above. During the formula design process the user should operate with visual objects only. In this case, the system will not allow the user to use wrong construction which is not allowed by syntax. Also, each visual object representing some part of the formula should produce linear text according to the syntax. Then, each structure-specific index term must be substituted according to the semantic table. After that, using the linear form of the expressions it is possible to make syntax analysis using target compiler. This stage is to inform user that the particular expression can't be compiled due to some restrictions of the target platform and point the place of possible mistake.

The executable code generation from the formula sequence specification is done as follows. Fist, the linear form of the formula is produced by the formula object, as was pointed above. The executable code should be generated using this linear textual representation with the substitution rules from semantic tables applied.

Thus, the visual terms are represented by their corresponding textual terms. Then the executable code of the formula generated by our subsystem is transferred to program generator [18] in order to produce an executable code for the whole CyberFilm specification.

Symbol type	Visual layout	Examples		
		Visual		Textual
full_flash_node	●	A[●]	Element of A related to full flashing node	A[I][J]
contour_flash_node	◉	A[◉]	Element of A related to contour flashing node	A[c1*I+c2][k1*J+k2]
stencil_node	(stencil symbol)	A[(stencil)]	Element of A related to stencil node	A[I][J-1]
sum (sum of all flashing nodes of specified color)	$\sum$ ●	$x = \sum A$	Sum of A's elements related to all flashing nodes of a specified color	… x=sum(A, COLOR_RED); …
stencil_sum (sum with coefficients of table)	$\sum$ (table)	$\sum \begin{array}{ccc} 0 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 0 \end{array} A[●]$	Sum of A's elements related to a central node pointed by a full-flashing node and its neighbors with coefficients of table	(-1)*A[I-1][J]+ (-1)*A[I][J-1]+ 2*A[I][J]+ (-1)*A[I][J+1]+ (-1)*A[I+1][J]

Figure 5. A part of the semantic table for variables declared on 2D structure

```
…
<math_expression> ::= <function> | <node_data> | <constant> |  …
<node_index> ::= <constant> | full_flash_node | contour_flash_node | stencil_point | …
<active_node_index> ::= full_flash_node
<node_data>  ::= <struct_variable>[<node_index>] | <struct_variable>[<active_node_index>]
<table> ::= {{<constant> [,<constant> ]}[, <constant>]}
<function> ::= sin(<math_expression>) | cos(<math_expression>) | …
<group_function> ::= stencil_sum(<table><active_node_index>) | min(color, <struct_variable>) |
max(color, <struct_variable>) | sum(color, <struct_variable>)  …
…
```

Figure 6. A part of formula syntax diagram

3.5 Formula editor user interface

The user interface of the formula sequences editor is presented in Figure 7. The left area of the window presents formula sequences with their masks; the right area contains buttons for manipulation with formulas; the bottom area contains special formulas for variable visualization. There is one common mask for all formula sequences and several local masks one per formula. The user can use mouse and keyboard to create and modify formulas. There are several groups of buttons: relations, operations and symbols, index icons, functions and operators, visualization. To add or modify a part of the formula, it is necessary to select it by left button of mouse and press corresponding buttons from the right panel of window. The system will not allow to do wrong manipulation. To set or modify string parameter, it is necessary to click appropriate formula component and type a text. To specify variable access it is necessary to click right button and choose the variable name from popup menu, then click an index button. The formula editing system also makes it possible to operate with system clipboard as well as exchange (import/export) formula text with other math software tools using OpenMATH format [19].

Additionally, the user can attach special visualization formulas. These formulas are used to invoke data visualization subsystem with the following parameters: variable name, method of visualization and additional visualization parameters (Figure 7, bottom area).

Figure 7. Formula sequences editor GUI

4 Example

A typical example of our visual formulas is presented by Figure 8, a. This formula looks like traditional

differentiation scheme for the partial differential equation [20]. There are two parts in the formula: a mask and an assignment statement. The mask specifies computing only on internal nodes of the grid. This means that only internal nodes will be used as index parameters in full-flashing indices and stencils. The assignment statement consists of left and right parts. The left part has variable **F** of 2D structure where result of the computation should be stored. The right part has an arithmetical expression with **U** variable as a source of data. Parameters *h* and *l* are global (observer variables). The formula means that variable **F** should be updated in all full-flashing nodes of 2D space. The result of C++ source code generation for the formula is shown in Figure 8, b.

a) visual formula representation

F[_I][_J]=(U[_I][_J-1] - 2*U[_I][_J]+ U[_I][_J+1])/h*h + (U[_I-1][_J]-2*U[_I][_J]+ U[_I+1][_J])/l*l;

b) textual formula representation

Figure 8. Formula example

5   Conclusion and future work

In this paper, a subsystem for formula sequences specification, space-time structures definition and variables declaration in the CyberFilm environment have been introduced. A special formula representation technique using enhanced text-based terms, images and stencils has been developed. In order to simplify the formula representation, the index simplification and table (vector and matrix) formula components enhancements were introduced and implemented. A GUI for these specification, definition and declaration has also been designed and implemented.

The future research of this project will be concentrated on the development of the "activity" view interface. This view will allow the user to extract information about operations in the CyberFilm, modify formula sequences, exchange operations with other CyberFilms, etc. The user will also be able to search formulas by many parameters, sort and group them by types and features of activity. In future, this system should be used not only as an algorithm demonstration tool but also as a programming tool. In this case, the executable code generation will be more powerful including possible automatic optimizations.

References

1. A. Kerrer and J. Stasko. Algorithm Animation. *In S. Diehl (Editor), Software Visualization*, LNCS, 2002, pp. 1-15.
2. J. Stasko, J. Dominique, M. Brown and B. Price (Editors). Software Visualization: Programming As a Multimedia Experience. *The MIT Press*, 1998.
3. M. Brown. Algorithm Animation. *The MIT Press*, 1988.
4. C. Larman, Applying UML and patterns. An introduction to object-oriented analysis and design and the unified process. *Prentice Hall*, 2002.
5. P.T. Cox, B. Song. A Formal Model for Component-Based Software. *In 2001 IEEE Symposium on Visual/Multimedia Approaches to Programming and Software Engineering*, Stresa, Italy, 2001.
6. M. Hirakawa, M. Tanaka and T. Ichikawa. HI-VISUAL Iconic Programming Environment. *In T. Ichikawa, E. Jungert and R.R. Korfhage* (Editors), *Visual languages and applications, Plenum Press*, 1990, pp. 121-145.
7. S. Tanimoto, Programming in a Data Factory. *In Proc. of Human Centric Computing Languages and Environments,* 2003, pp. 100-107.
8. C. Geiger, W. Mueller and W. Rosenbach. SAM - An Animated 3D Programming Language. *In 1998 IEEE Symposium on Visual Languages*, Halifax, Canada, 1998
9. MATLAB. High-level technical computing language. *http://www.mathworks.com/products/matlab.*
10. MathCAD design environment. *http://www.mathsoft.com.*
11. L. Starr. Executable UML. How to build class models. *Prentice Hall*, 2002.
12. R. Oechsle, T. Schmitt. JAVAVIS: Automatic Program Visualization with Object and Sequence Diagrams Using the Java Debug Interface (JDI). *Software Visualization*, LNCS, 2269, 2002, pp. 1-15.
13. N. Mirenkov. "Filmification" of Methods: Programming technology for the 21st Century, Nonlinear Analysis. *Theory, Methods & Applications*, Vol. 30, No. 2, Pergamon, 1997, pp. 779-789.
14. R. Yoshioka, N. Mirenkov, Y. Tsuchida, Y. Watanobe. Visual Notation of Film Language System. *In Proc. of Workshop on Visual Computing at Int. Conf. on Distributed Multimedia Systems*. San-Francisco, Sep. 26-28, 2002.
15. R. Yoshioka and N. Mirenkov. Visual Computing within Environment of Self-explanatory Components. *Soft Computing J., Vol. 7, Issue 1*. Springer-Verlag, 2003
16. B. Kolman, R. Busby, S. Ross. Discrete mathematical structures (4th edition). *Prentice-Hall*, 2000.
17. M. Erwig, Abstract Syntax and Semantics of Visual Languages. *In Journal of Visual Languages and Computing, 9*, 1998. pp. 461-483.
18. T. Ebihara, N. Mirenkov and M. Nemoto. Program generator within multimedia programming environment. *In Proc. of the Sixth International Conference on Human and Computer*, 2003, pp. 8-13.
19. The OpenMath Society, Documentation and Standard. *http://www.openmath.org.*
20. C.H. Edwards and D.E. Penney. Differential Equations & Linear Algebra. *Prentice Hall*, 2001.

PARIS: FUSING VISION-BASED LOCATION TRACKING WITH STANDARDS-BASED 3D VISUALIZATION AND SPEECH INTERACTION ON A PDA

Stuart Goose, Sinem Güven†, Xiang Zhang, Sandra Sudarsky, Nassir Navab

Multimedia Technology Department, Siemens Corporate Research, Inc.
755 College Road East, Princeton, NJ 08540, USA
†Columbia University, 500 West 120th Street, New York, NY, 10027, USA

ABSTRACT

Industrial service and maintenance is by necessity a mobile activity, and the aim of the technology reported is towards improving automated support for the technician in this endeavor. As such, a framework was developed called PARIS (PDA-based Augmented Reality Integrating Speech) that executes entirely on a commercially available PDA equipped with a small camera and wireless support. Real-time computer vision-based techniques are employed for automatically localizing the technician within the plant. Once localized, PARIS offers the technician a seamless multi-modal user interface juxtaposing a VRML augmented reality view of the industrial equipment in the immediate vicinity and initiates a context-sensitive VoiceXML speech dialog concerning the equipment. Integration with the plant management software enables PARIS to access equipment status wirelessly in real-time and present it to the technician accordingly.

1. INTRODUCTION AND MOTIVATION

Siemens is the world's largest supplier of products, systems, solutions and services in the industrial and building technology sectors. Service and maintenance is by necessity a peripatetic activity, and as such one continuing aspect of our research focuses upon improving automated support for this task. Another future trend that we have been focusing on is applying 3D interaction and visualization techniques to the industrial automation domain.

In recent years we have witnessed the remarkable commercial success of small screen devices, such as cellular phones and Personal Digital Assistants (PDAs). Keyboards remain the most popular input device for desktop computers. However, performing input efficiently on a small mobile device is more challenging. Speech interaction on mobile devices has gained in currency over recent years, to the point now where a significant proportion of mobile devices support or include some form of speech recognition.

The ability to model real world environments and augment them with animations and interactivity has benefits over conventional interfaces. However, navigation and manipulation in 3D graphical environments can be difficult, and disorientating, especially when using a conventional mouse. Small sensors can be used to report various data about the surrounding environment and relative movement, etc. One such sensor is that of a small camera.

The hypothesis that motivated this research is that a camera, in conjunction with computer vision algorithms, could be exploited to provide location information, which in turn, could seamlessly and automatically drive the navigation through a 3D graphical world representing selected elements in the real world. In addition to eradicating partially the complexity of 3D navigation, integrating context-sensitive speech interaction could further simplify and enrich the mobile interaction experience. Hence, the PARIS framework was developed for experimenting with the provision of mobile, context-sensitive, multi-modal user interfaces for mobile maintenance.

Figure 1: A mobile maintenance technician using PARIS.

To the knowledge of the authors, this is the first reported VRML-based AR framework that executes entirely on a commercially available PDA. PARIS employs real-time vision algorithms for localizing a technician and offers a multimodal user interface that synchronizes an augmented reality graphical view based on VRML [22] with a VoiceXML [21] speech-driven interface. After automatically detecting when the technician enters the vicinity of a specific plant component, PARIS can engage him or her in a context-

specific speech dialog concerning the corresponding component, as shown in figure 1.

Reported in the remainder of the paper are some novel aspects of PARIS. A brief discussion of related work is provided in Section 2. The system architecture and components are presented in section 3. In section 4, the VoiceXML Lite subsystem that provides location and context-sensitive speech support is reported. Our vision-based techniques for marker tracking and localization are presented in section 5. The industrial scenario and multimodal interaction experience is offered in section 6. Sections 7 and 8 propose areas for further research and provide some concluding remarks.

2. RELATED WORK

This review selectively traces the progress of mobile 3D interfaces, location tracking, augmented reality and speech interaction, and hence the confluence of these technologies as they relate to mobile maintenance.

The benefits of mobile maintenance [20] and virtual environments [4] to the industrial sector have been reported. Navigation and manipulation in desktop 3D graphical environments can be difficult. This need spawned research into novel input and control devices for this purpose [25]. Fitzmaurice et al [6] in 1993 simulated a palmtop computer to, among other things, evaluate how novel input devices can expedite interaction in virtual environments on handheld devices. Hinckley et al [12] describes how a PocketPC was augmented with multiple sensors to offer adaptive interaction with mobile devices, including automatic power on/off, automatic landscape/portrait flipping etc. Mobile Reality [10] is a framework that combines the input from infrared beacons and an inertia tracker to drive automatically the VRML display on a PDA. In contrast to the above, PARIS leverages vision-based localization algorithms executing on the PDA to adjust the viewpoints in the VRML scene correspondingly.

A variety of indoor location tracking technologies have been reported. The Active Badge System [24] facilitates position tracking of people wearing badges in an office environment and, for example, to route phone calls to the closest telephone. Memoclip [2] aims at providing users with location-based messages. When a user approaches a sensor that is physically associated with a reminder, the Memoclip displays the corresponding message. Newman et al [15] describe an AR system whereby the user wears an ultrasonic positioning device. An X-Windows server redirects the user interface of the application to an iPAQ running Linux. Goose et al [10] report a PDA-based hybrid tracking solution that fuses the input from infrared beacons and a three degrees-of-freedom (3 DOF) inertia tracker.

The German Ministry for research and training (BMBF) funds a project called AR-PDA [1]. A few research groups have focused their attention on the use of PDAs for augmented reality applications [3, 7, 8, 9], however none of these perform image processing onboard the PDA. Bertelsen et al [3] use the PDA in conjunction with a barcode reader to access to the data in a water treatment plant. Geiger et al [7, 9] use the PDA to acquire images, wirelessly transmit them to a server for marker detection, scene augmentation and retransmission back to the PDA for AR visualization. Gausemeier et al [9] proceeds in a similar vein but try a method that exploits feature correspondences to the 3D models to estimate the pose and augment the video. By contrast, PARIS performs all processing locally [26, 27]. Wagner et al report a PDA AR solution [23] that perform local processing by leveraging the ARToolkit, whereas PARIS uses a VRML solution.

Ressler et al [19] describe a technique for integrating speech synthesis output within VRML, however the integration of speech recognition is not considered at all. Mynatt et al [14] describe, Audio Aura, to provide office workers with rich auditory cues (via wireless headphones) within the context of VRML for describing the current state of the physical objects that interest them. By contrast, PARIS supports speech in and out for dialog using VoiceXML. In addition, neither approach from Ressler nor Mynatt consider PDAs.

3. ARCHITECTURE

Important technology considerations were to embrace international standards where suitable, execute on a commercially available PDA, and also to leverage any appropriate products and schemes used in contemporary plants. As such, PARIS supports VRML and VoiceXML for the graphic and speech media. The PDA device used was a regular Compaq iPAQ Pocket PC equipped with 200MHz processor, 64Mb memory, a Compact Flash camera and an 802.11b wireless card.

The PARIS framework comprises five significant components or subsystems: Augmented reality management unit, VRML engine, VoiceXML Lite [11, 5], Tracking and localization and Plant management communication. A high-level functional view of PARIS can be seen in figure 2. The inputs are on the left, the outputs are on the right, and wireless communication with the plant management system is below. The following sections explain the function and interaction of these components.

The central command and control center of the PARIS architecture is the Augmented Reality (AR) Management Unit (figure 3) which is responsible for orchestrating all interaction between VoiceXML Lite, the VRML engine and the Tracking and Localization components. The VRML Manager is responsible for initializing and instructing the engine to load the appropriate VRML world. In addition, the VRML Manager is responsible for coordinating and

synchronizing any updates as the technician interacts with the scene. The Video Manager provides an interface to the tracking and Localization software. It initializes the camera, makes appropriate calls to and receives events from the Localization and Tracking component to perform the marker detection. The Voice Manager governs communication with the VoiceXML Lite subsystem. It drives VoiceXML Lite, determining the appropriate VoiceXML file to be loaded based upon the purpose of the user interaction. The Voice Manager gathers parameters from the user during the speech dialog and forwards messages to the VRML Manager for possible visual display.

Figure 2: High-level view of PARIS.

The architecture is driven entirely by the inputs, either from the input sensors or from the user. The typical flow is described below. The tracking and location component processes the video searching for visual markers. Upon successful detection of a marker, the Video Manager receives an event indicating the unique identity of the marker. The tracking and localization algorithm is then stopped, as it degrades the performance of the speech interaction and visualization. The Video Manager then communicates this information to the Voice Manager and the VRML Manager. The Voice Manager instructs VoiceXML Lite to load the VoiceXML file associated with this marker and to begin the speech dialog. The system then engages the user in a speech dialog. As each spoken form input item is elicited from the technician, an embedded JavaScript function is executed to pass the data onto the Voice Manager. The Voice Manager creates a message into which the input data is placed. This data represents information pertaining to the nature of the task and the entities affected. This message is then written to a message

queue. This process is repeated until all of the required speech inputs have been gathered. Unless already loaded, the VRML Manager instructs the VRML engine to load the VRML world associated with this marker and transitions the virtual camera position to the corresponding viewpoint. The VRML Manager then periodically polls the message queue for pending messages from the Voice Manager. Just as HTML and VoiceXML can contain JavaScript, so too can VRML. In addition to geometrical information, the VRML world is imbued with an extensible collection of JavaScript nodes with parameterized functions for performing visual actions to change the VRML scene. The information extracted from a message and maps onto the parameters of these JavaScript functions. The VRML Manager interacts with the VRML engine to set each JavaScript parameter in the VRML node(s) and then invokes the appropriate JavaScript function(s) to perform various visual action(s). Among the visual actions currently provided are highlighting nodes, overlaying nodes, displaying signposts containing textual descriptions, etc.

Figure 3: Augmented Reality Management Unit.

The Video Manager repeats the cycle by re-starting the localization and tracking component.

4. VISION-BASED LOCALIZATION

Coded visual markers are used to support motion tracking and localization, and are used in many industrial sites for photogrammetry. In many cases, marker positions are measured to millimeter precision and stored in databases. Algorithms exist for computing the 3D position and orientation of a camera relative to markers. An example of the coded visual markers employed by PARIS can be seen in figure 4. The rectangular frame is used for the marker detection and for image correspondences. Using the 4x4 coding matrix, more than ten thousand uniquely coded visual markers can be generated. Each marker provides at least eight feature points for image correspondences.

The visual marker-based localization is implemented through motion tracking and camera calibration. A homography-based camera calibration algorithm exploits

the correspondence between a set of coplanar points and their images to estimate the position and orientation of the camera. With every coded visual marker pre-registered within the global coordinate system, the 3D position and orientation of the camera attached to the PDA can be thus determined [27]. Below is a brief description of the camera calibration algorithm.

Figure 4: An example of a coded visual marker.

The pinhole camera model describes the relationship between a 3D point, $\mathbf{M} = [X, Y, Z, 1]^T$, and its 2D projection, $\mathbf{m} = [u, v, 1]^T$, on the image plane as

$$s\, \mathbf{m} = \mathbf{A}\, [\mathbf{R}\, \mathbf{t}]\, \mathbf{M} \qquad (1)$$

where s is a scaling factor, $\mathbf{R} = [\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{r}_3]$ the 3×3 rotation matrix, $\mathbf{t}$ the 3×1 translation vector, and $\mathbf{A}$ the camera intrinsic matrix given by $A = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}$,

with (u_0, v_0) being the coordinates of the camera optical center on the image plane, α and β the focal lengths in image u and v directions, and γ the skewness of the two image axes. Since all 3D points are on the model plane, we construct the marker coordinate system with $Z = 0$. Thus equation.(1) can be rewritten as

$$s\, \mathbf{m} = \mathbf{A}\, [\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{r}_3\ \mathbf{t}]\, [X\ Y\ 0\ 1]^T$$
$$= \mathbf{A}\, [\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{t}]\, [X\ Y\ 1]^T = \mathbf{H}\, [X\ Y\ 1]^T \quad (2)$$

or

$$s\, \mathbf{m} = \mathbf{H}\, [X\ Y\ 1]^T \qquad (3)$$

where $\mathbf{H}$ is the 3×3 homography describing the projection from the marker plane to the image plane. We note

$$\mathbf{H} = [\mathbf{h}_1\ \mathbf{h}_2\ \mathbf{h}_3] = \lambda\, \mathbf{A}\, [\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{t}] \qquad (4)$$

Since at least 8 pairs of correspondences can be obtained from each marker, the homography $\mathbf{H}$ can be determined up to a scaling factor. In many cases, the intrinsic matrix $\mathbf{A}$ is given from off-line camera calibration, then the rotation matrix $\mathbf{R}$ and translation vector $\mathbf{t}$ can be obtained. The final results are then optimized by minimizing the following function for a set of n images, each with m known coplanar 3D points:

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\| \mathbf{m}_{ij} - \mathbf{m}'(\mathbf{A}, \mathbf{R}_i, \mathbf{t}_i, M_j)\|^2 \qquad (5)$$

where $\mathbf{m}'(\mathbf{A}, \mathbf{R}_i, \mathbf{t}_i, M_j)$ is the projection of point M_j in image i. This nonlinear optimization problem can be solved with the Levenberg-Marquardt algorithm [17].

Previous work [3, 7, 8, 9] suggests that a PDA is not powerful enough to perform the real-time image processing and augmentation. The implementation of PARIS reported in this paper proves the contrary, as the PDA is the only computer used for processing. The video analysis, marker-based tracking and localization, image augmentation and AR visualization all execute efficiently on the PDA. The current implementation provides detection of the markers in real-time: image acquisition and marker detection performs well at 10 frames per second or more, depending on other processes being managed by the PDA. It is also able to estimate the position and orientation of the PDA, and therefore its user, relative to the environment. The technician can stand approximately 10 feet away from the marker.

5. INTERACTION AND VISUALIZATION

The plant components designated to be visualized and speech-enabled are each labeled with unique visual markers, as can be seen in figure 1. Corresponding VRML worlds and VoiceXML scripts must be generated or created. The collection of JavaScript functions that provide the visual actions must be embedded or referenced in the VRML file(s). JavaScript for enqueuing a message must also be embedded or referenced in the VoiceXML file(s).

Figure 5: 3D visualization of a pipe assembly with the joints highlighted in red.

Current maintenance practice ranges from completing paper-based forms through to using various flavors of mobile computers for inputting values into fields of server-generated HTML forms. Instead of server-generated HTML forms, PARIS could retrieve and process server-generated VoiceXML forms. PARIS facilitates the quasi-synchronized rendering of the synthesized speech output with its visual counterpart. This feature is necessary for confirming interactions in noisy environments.

Figure 6: A text label is displayed as a result of a speech-driven query for the pressure in the joints.

A scenario was developed in order to test and evaluate the framework in the lab environment. A maintenance technician patrols the plant with her PDA and approaches the first piece of equipment scheduled for repair. PARIS automatically localizes her and displays the pipe assembly, as seen in figure 5.

Struggling to identify the joint at fault, she asks the system to highlight the joint. PARIS then highlights in red the joint and repositions the viewpoint for greater clarity. The technician performs the repair. Once the faulty joint has been replaced and the pipe assembly reconnected and enabled, the technician begins a series of tests to verify that the fault has indeed been corrected satisfactorily. She says "Pressure", to which the framework issues a wireless HTTP query to the Siemens WinCC plant automation software. Upon receipt of the pressure information PARIS announces, "Current pressure is 100 pounds per square inch" while displaying a text label containing the same information (figure 6).

She walks on and climbs a nearby ladder to observe the connection of the pipe assembly to a container tank. PARIS again localizes her and transitions the viewpoint to reflect her new location. She then issues a final check by requesting "Temperature", to which the framework again queries WinCC for the real-time value. Upon receipt PARIS announces, "Current temperature is 300

degrees Celsius" while displaying a text label confirming this value. This can be seen in figure 7.

Figure 7: The pipes are highlighted in red, and a label is displayed as a result of a speech-driven query for the temperature in the pipes.

6. FUTURE WORK AND CONCLUSIONS

While a number of tracking technologies have been proposed in the literature, Klinker *et al* [13] recognizes that the most successful indoor tracking solutions will comprise two or more tracking technologies to create a holistic sensing infrastructure able to exploit the strengths of each technology. We subscribe also to this philosophy, hence current work involves integrating supplementary localization technologies [10], providing support in areas where visual markers are either not present or cannot be detected effectively.

It possible to estimate from the video the position and orientation of the PDA, and therefore its user, relative to the environment. As the VRML worlds can become too large in size for the PDA, current work includes a scheme for downloading and caching of partial VRML worlds. By transmitting to a server the marker identity and the user's location relative to the identified marker, the corresponding partial 3D worlds can be returned. Extensions to support mobile collaborative fault diagnosis are also in development. These include the ability to support a full duplex SIP/RTP voice-over-IP channel and a shared VRML browsing session with a remotely located expert.

Industrial service and maintenance is by necessity a mobile activity. The aim of this research is to improve the automated support for the technician in this endeavor. As such, a framework was developed called PARIS (PDA-based Augmented Reality Integrating Speech) that executes entirely on a commercially available PDA equipped with a small camera and

wireless support. Real-time computer vision-based techniques are employed for automatically localizing the technician within the plant. Once localized, PARIS offers the technician a seamless multi-modal user interface juxtaposing a VRML augmented reality view of the industrial equipment in the immediate vicinity and initiates a context-sensitive VoiceXML speech dialog concerning the equipment.

Important technology considerations were to embrace international standards where suitable (VRML and VoiceXML), execute on a commercially available PDA, and also to leverage any appropriate products and schemes used in contemporary plants. As such, PARIS supports VRML and VoiceXML for the graphic and speech media.

Although industrial mobile service and maintenance has provided the application context throughout this paper, the authors are exploring potential applicability of the technology in other vertical markets, such as healthcare, tourism and building information systems.

7. REFERENCES

[1] BMBF funded AR-PDA Project, http://www.ar-pda.de/

[2] Beigl, M., Memoclip: A Location-based Rembrance Applicance, Journal of Personal Technologies, 4(4):230-234, Springer Press, 2000.

[3] Bertelsen, O., Nielsen, C., Augmented Reality as a Design Tool for Mobile Interfaces, Proceedings of Designing Interactive Systems: Processes, Practices, Methods, and Techniques, ACM Press, pages 185-192, New York, 2000.

[4] Dai, F., Virtual Reality for Industrial Applications, Springer-Verlag, 1998.

[5] Eberman, B., Carter, J., Meyer, D. and Goddeau D., Building VoiceXML Browsers with OpenVXI, Proceedings of the 11th ACM International World Wide Web Conference, Hawaii, USA, pages 713-717, May, 2002.

[6] Fitzmaurice, G., Zhai, Z. and Chignell, M., Virtual Reality for Palmtop Computers, ACM Transactions on Office Information Systems, 11(3):197-218, July, 1993.

[7] Gausemeier, J. , Fruend, J., Matysczok C., Bruederlin B., Beie D., Development of a Real Time Image-Based Object Recognition Method for Mobile AR-Devices, Proceedings of International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction, Africa, February 2003.

[8] Geiger, C., Kleinjohann, B., Reimann, C. and Stichling, D., Mobile AR4All, Proceeding of IEEE and ACM International Symposium on Augmented Reality, New York, 2001.

[9] Geiger, C., Paelke, V., Riemann, C., Rosenbach, W., Stoecklein, J., Testable Design Representation for Mobile Augmented Reality Authoring, Proceeding of the IEEE and ACM International Symposium on Mixed and Augmented Reality, Darmstadt, Germany, 2002.

[10] Goose, S., Wanning, H. and Schneider, G., Mobile Reality: A PDA-Based Multimodal Framework Synchronizing a Hybrid Tracking Solution with 3D Graphics and Location-Sensitive Speech Interaction, Proceedings of the ACM 4th International Conference on Ubiquitous Computing, Göteburg, Sweden, pages 33-47, September, 2002.

[11] Goose, S., Kodlahalli, S. and Lukas, K., VoiceXML Lite: A Standards-based Framework for Speech-Enabling Applications Executing on Commodity Networked Devices, Proceedings of the IEEE International Conference Distributed Multimedia Systems, Florida, USA, September, 2003.

[12] Hinkley, K, Pierce, J., Sinclair, M. and Horvitz, E., Sensing Techniques for Mobile Interaction, ACM UIST, San Diego, USA, November 2000.

[13] Klinker, G., Reicher, T. and Bruegge, B., Distributed User Tracking Concepts for Augmented Reality Applications, Proceedings of ISAR 2000, Munich, Germany, pages 37-44, October, 2000.

[14] Mynatt, E., Back, M., Want, R., Baer, M. and Ellis, J., Designing Audio Aura, ACM International Conference on Computer Human Interaction, Los Angeles, USA, pages 566-573, 1998.

[15] Newman, J., Ingram, D. and Hopper, A., Augmented Reality in a Wide Area Sentient Environment, Proceedings of IEEE International Symposium on Augmented Reality (ISAR), pages 77-86, October 2001.

[16] Nilsson, J., Sokoler, T., Binder, T. and Wetcke, N., Beyond the Control Room: Mobile Devices for Spatially Distributed Interaction on Industrial Process Plants, Proceedings of the Second International Symposium on Handheld and Ubiquitous Computing, Bristol, U.K., pages 30-45, September 2000.

[17] Press, W., Teukolsky, S., Woo, M. and Flannery, B., Numerical Recipes in C: The Art of Scientific Computing, Cambridge University Press, 1992.

[18] Rekimoto, J. and Ayatsuka, Y., Cybercode: Designing Augmented Reality Environments With Visual Tags, Designing Augmented Reality Environments, 2000.

[19] Ressler, S. and Wang, Q., Making VRML Accessible for People with Disabilities, ASSETS 98, Marina del Rey, USA, pages 50-55, April 1998.

[20] Smailagic, A. and Bennington, B., Wireless and Mobile Computing in Training Maintenance and Diagnosis, IEEE Vehicular Technology Conference, Phoenix, AZ, May 1997.

[21] Voice Browser Working Group, http://www.w3.org/Voice

[22] VRML97 Specification, ISO/IEC 14772-1:1997

[23] Wagner, D., and Schmalstieg, D., First Steps Towards Handheld Augmented Reality, ISWC, New York, USA, October 2003.

[24] Want, R., Hopper, A., Falcao, V. and Gibbons, J., The Active Badge Location System, ACM Transactions on Information Systems, 10(1):91-102, 1992.

[25] Zhai, S., Milgram, P. and Drasic, D., An Evaluation of four 6 Degree-of-Freedom Input Techniques, ACM Conference on Human Factors in Computer Systems, Asmterdam, Netherlands, 1993.

[26] Zhang, X. and Navab, N., Tracking and Pose Estimation for Computer Assisted Localization in Industrial Environments, IEEE Workshop on Applications of Computer Vision, pages 214- 221, 2000.

[27] Zhang, Z., Flexible Camera Calibration by Viewing a Plane From Unknown Orientations, IEEE International Conference on Computer Vision, pages 666-673, 1999.

EVE - II: AN INTEGRATED PLATFORM FOR NETWORKED VIRTUAL ENVIRONMENTS

Ch. Bouras[1,2], A. Panagopoulos[2], N. Theoharis[2], Th. Tsiatsos[1,2]

[1] Research Academic Computer Technology Institute, Greece
[2] Computer Engineering and Informatics Dept., Univ. of Patras, Greece

Abstract

In this paper, we present the design and implementation of an integrated platform for Networked Virtual Environments. This platform called EVE-II is an enhancement of the EVE distributed virtual reality platform, supporting stable event sharing for multi-user 3D places, easy creation of 3D multi-user3D places, H.323-based Voice over IP services integrated in a 3D spaces well as many concurrent 3D multi-user spaces.

1. Introduction

Networked Virtual Environments (NVEs) are multi-user virtual worlds, namely computer generated spaces, where participants represented by avatars can meet and interact. Nowadays, the use of NVEs is one of the most promising uses of virtual reality. Using NVEs as communication media, we can offer to members of virtual communities the advantage of creating proximity and social presence, thereby making participants aware of the communication and interaction processes with others [2]. Furthermore, NVEs could be the basis of Educational Virtual Environments where the users could collaborate in order to learn together. In the Educational Virtual Environments the avatars of the users are provided with additional behavior such as gestures, interactivity, movements and voice communication. The following basic requirements should be satisfied in order to implement an integrated platform for NVEs that can also support Educational Virtual Environments [9]: (a) high level of presence of the users, (b) multi-modal user-to-user interaction via chat, voice communication and gestures, (c) user-system interaction, (d) scalability, (e) consistency of the multi-user space and (f) Quality of Services.

In this paper such a platform for supporting NVEs, called EVE-II, is presented. EVE-II is actually an enhancement of EVE platform [1], [3]. Main improvements of EVE that has been integrated in EVE-II concern the sharing of multi-user events, as well as the audio communication among the users as described later in this paper. The emphasis on the new version of EVE platform has been given on the flexibility and extensibility of the architecture, its stability as well as the support of a more easy way for transforming standalone 3D worlds to multi-user places. EVE-II is based on Virtual Reality Modeling Language (VRML) [8] for the representation of the 3D worlds and for describing 3D objects. However, VRML does not provide support for multi-user virtual worlds [3]. Thus, there is a definite need for a multi-user extension behind the VRML. So far there is no standard in this area [3]. The Living Worlds (LW) working group of the Web3D consortium, which has been frozen, has made the first attempt for standardization. Other remarkable proposals are the VSPLUS proposal, which is a simplification of the LW proposal, and the SPIN-3D approach. However the aforementioned approaches for VRML data sharing have some limitations. LW is complex, VSPLUS does not support dynamic created objects and SPIN-3D requires a proprietary VRML browser [3]. EVE-II supports such an extension through a VRML parser satisfying at least the following requirements: (a) conformity with a standard VRML97 browser, and (b) easy transformation of a single user world to a multi-user 3D world. This paper is structured as follows. We initially describe the architecture of EVE-II platform. Afterwards, we briefly describe a prototype for offering e-learning services using EVE-II. Finally, we present some concluding remarks and our vision for the next steps.

2. EVE-II Architecture

EVE-II's architecture (Fig. 1) is based on a client-multiserver platform model. The current form of EVE-II constitutes an open and flexible architecture, which allows and supports the basic functionality that the platform is intended to offer. For that reason the servers on which the

platform relies, are the message server and two application servers, a chat and an audio server. This model offers scalability and flexibility to the EVE-II architecture, because we can add more application servers in order to offer more functionality and furthermore the processing load is distributed among the above set of servers.

In addition EVE-II is characterized from openness due to the fact that is based on open technologies and international standards. More specifically the implementation of the platform is mainly based on:

- VRML, for the representation of the 3D worlds and for describing 3D objects.

- VRML External Authoring Interface (VRML-EAI) [8], for implementing an interface between the 3D worlds and external tools.

- Java, for the development of the client-server model, and the network communication among the different components of our platform.

- H.323 [4], [5] for offering audio conferencing services through the Internet.

EVE-II in comparison with the previous version of our platform (EVE) is improved mainly on the sharing of events, as well as the audio communication among the users.

Concerning the sharing of multi-user events EVE-II, goes beyond EVE and other platforms [3]. Actually the VRML Data Sharing mechanism in EVE was based on the usage of a specific type of file (called SVE), which was maintaining every shared event and shared object in order to facilitate the multi-user communication and the initialization process.

The new approach for the sharing of the multi-user events is based on a VRML parser that has been implemented. This VRML parser runs on the server side, it is an extension of the SVE parser and it helps the server to recognise the shared events without the usage of an SVE file. Exploiting this new approach EVE-II has the following advantages in comparison with EVE:

- It offers enhanced stability through better interface with EAI as well as better support of avatar and avatar's gestures.

- It supports very easy creation of a multi-user space from a standalone one, through the integration of the VRML parser. Actually the shared events are commented out (i.e. marked with a "#"), in the original VRML file and in such a way the standalone world is transferred to a multi-user one.

- It offers server-side syntax checking of 3D spaces in order ti support better and faster sharing of multi-user events.

- It supports execution of shared scripts and VRML routes, and full support of scripts sharing (both on javascript and java format).

- It supports server-side execution of scripts, which offers better sharing of events, even if they are based on time-triggering.

- It supports dynamic insertion of shared object in multi-user places.

- It supports specific PROTOs (such as "chair" for avatar's sitting).

- It supports better initialization process.

Figure 1: Architecture of EVE II

Concerning the audio communication H.323 protocol is supported. H.323 is an ITU recommendation, which defines a network architecture and the associated protocols necessary to voice and multimedia calls establishment. H.323 is a protocol suite that can be used in order to establish, modify and terminate multimedia sessions or calls. These multimedia sessions include both point-to-point and multi-point conferences and Internet telephony applications. Main reason for this choice was the H.323's modular structure that offers flexibility and allows the usage of many well-known codecs and mechanisms for the transmission of the data. Furthermore H.323 supports much more services than voice over IP such as videoconferencing that could be integrated in future versions of EVE platform

In the following paragraphs the main components of EVE-II architecture are described.

2.1 Server Side

The servers on which the platform relies, is the message server and two application servers, a chat and an audio server.

2.1.1 Message Server

The message server is responsible for the manipulation of the virtual worlds that are visited by the users of the system. In addition, this server creates and supports the illusion to the users that they are participants in the above virtual worlds and that they share a common space by updating the view of the world every time that a shared object is modified. Two servers, each of which is used for a specific sequence of operations, constitute this message server. These servers are the Connection Server and the VRML server:

- Connection Server: this server maintains a database, which the system accesses in order to authenticate the user and allow him/her to enter the virtual space of EVE. In addition, the connection server reports every entry or departure that takes place in the platform to all other servers.

- VRML server: this server monitors and records every event that takes place in the virtual space and reports these changes to all participant clients of the platform. Thus, by performing these continuous updates the system assures that the users will have the illusion of sharing a common space. The VMRL server also maintains constantly an updated copy of the world, which is sent to the clients when they enter the system. That way, the new users have the same updated view that the existing users already have.

2.1.2 Application Servers

The application servers are responsible for providing specific functionality to the participants of the virtual world. In the current form of EVE there are two application servers available, a chat server and an audio server.

- Chat Server: this server is responsible for the text chat support. It allows group chat, which means text chatting between multiple users, or whispering, which allows the one-to-one communication between two users.

- Audio Server: this server is responsible for the audio communication between the users of the platform. The audio server uses H.323 as its main protocol. H.323 is a multimedia communication protocol, which can transfer voice, video or data over IP networks, and is especially fitted for this application. The main audio service offered by the platform is the audio communication among all participants in a virtual world, or between pairs of them. So, the audio server is in fact an H.323 MCU, which supports audio conferencing among the platform users. By using H.323, compatibility with a large range of H.323 audio servers and clients is achieved and the use of audio as a separate service of the platform is permitted, whilst the numerous applications of H.323

can enrich the functionality of the platform – for example, by adding video conferencing capabilities in the future.

2.1.3 Client Side

As depicted in Fig. 1, in order the users' client to communicate with EVE's servers and have access to the provided functionalities they need a web browser, a VRML browser, the main EVE client and the audio client.

- Web Browser: The web browser is used for the communication with the web server of the system, which provides an initial interface and entry point between the user's client and EVE's environment.

- VRML Browser: The 3D environment of EVE is implemented using the VRML language. Therefore, a VRML browser, a plug-in, is essential in order to allow the navigation of the user's avatar in the virtual training space.

- Main Client: This client is responsible for (a) the primary connection of the user to the Message Server, (b) the interaction between the user's avatar and the 3D virtual space and (c) the text chat communication between the users of the same virtual space. In particular, the main client, which is a java applet, makes an initial connection to the connection server, which allows it to present the current connection status and when the user is authenticated, it passes on to the vrml server.

  During an initialization phase, the list of the current participants in the virtual space is retrieved, as well as some information about the user avatar. Then, the normal message exchange with the VRML server begins. The first message received always contains the world, in its current state, and the user avatars, so that it completes the initialization phase, and starts normal operation.

  During normal operation, this client is responsible for the interaction between the user's avatar and the 3D virtual space of EVE. In particular, every time that a user acts on an object, this client reports the modification and interaction to the VRML server of the platform that performs the update and transmits it to all other current participants.

  The main client also includes a chat client. This part of the main client is responsible for the text chat communication between the users of the same virtual space. Every time that a message is send from the client's side, this is passed to the chat server that in turn transmits it to the appropriate destinations.

- Audio Client: The audio client is a java applet that records the audio stream from the user's side and transmits it all appropriate destinations, allowing the

audio communication between participants in the same space. As already described, H.323 is used to support the audio services. The audio client communicates only with the audio server, which is used as a Multipoint Conference Unit (MCU), handling and mixing the audio streams that are sent by the clients, and forwarding them to the correct destinations.

2.2 Network Communication

The network communication of EVE, alike its architecture, is focused on providing the available functionality at the best possible performance. Therefore, for the transmission of the packets and the achievement of the communication of the connected clients with the host servers (message server, audio server and chat server) as well as for the server-to-server communication, there are three types of communication supported. Each of these types is found to be optimum for certain kinds of messages. Thus, we categorized the messages exchanged in the EVE communication platform in four basic categories:

- The messages related with the initial connection of a client to a server as well as the messages exchanged between the servers of the platform.
- The position messages that are related with the avatars' position and orientation in the virtual environment.
- The important messages, which correspond to messages that are vital for the consistency of the networked virtual environment (for simplicity reasons, we consider as important messages all messages except for the position messages).
- The messages related to audio streams.

In the following subsections, we describe why a connection type is selected for the corresponding category of messages described above.

2.2.1 TCP Communication

The main characteristic of the TCP communication is the reliability in the transmission of information packets.

Therefore, this type of communication is selected for the cases where the reliable delivery of the exchanged messages is essential and vital for the maintenance of the consistency of the networked virtual environment, even if that interferes some delay in the transmission.

For the EVE platform, this type of communication is selected for the following messages: a) the server-to-server communication, b) the initial connection of a client to the message server, which includes the authentication c) the messages that are vital for the consistency of the networked virtual environment, including the messages

that create the 3D world and the avatars, when a new client enters the system.

A possible failure or loss in the delivery of this type of messages could cause serious inconsistencies in the presentation of the virtual environment and could introduce security issues to the EVE platform.

2.2.2 UDP Communication

The main characteristic of this type of communication is the high speed in the transmission of the information packets. However, one of the main drawbacks of the simple UDP communication is that it cannot assure the reliable and correct delivery of the data packets.

Therefore, this type of communication is selected for the transmission of messages that their possible loss or failure in delivery does not imply a severe impact on the consistency of the virtual world of the connected clients. Such messages are the position message, which carry information about the avatars' position and orientation in the virtual world, and their failure in delivery does not create important scene inconsistencies to the participants.

2.2.3 H.323 based voice communication

As described above, H.323 protocol suite can be used for audio communication, while the transfer protocol used to actually transfer the audio data is RTP. A client exchanges RTP packets with the audio server. As already described, the audio server, which serves as an MCU, mixes the audio streams and forwards them to the clients, making sure that sounds generated by a client are not sent back to it.

3. Case Study: Support of E-Learning Services

NVEs have a good potential to support e-learning services, due to the fact that the can provide the students with an opportunity to experience sensory interactive learning environments, which enable them to move from passive to active learning [3]. In addition, such environments are able to support collaborative learning among students at different locations by allowing them to share experiences about exploring a common environment. We call these environments Educational Virtual Environments [2].

The primary goal of an Educational Virtual Environment is to provide tools in order to reproduce conditions that augment interpersonal interaction in a physical educational environment, e.g. a classroom. This goal is effectively satisfied if the educational virtual environment is represented by 3D virtual worlds where the users are represented by human-like avatars. For this reason, the EVE-II communication platform is exploited in order to support an Educational Virtual Environment.

The implementation of such a prototype gives us the possibility to test every aspect of the system in order to offer specific functionality. The environment that have been implemented is a simulation of a classroom, it combines 2D and 3D features in order to provide the users with communication and collaboration capabilities and necessary tools for realizing collaborative e-learning scenarios, and it is accessible at http://ouranos.ceid.upatras.gr/vr/. The participants in the virtual classroom could have two different roles: tutor (only one user among the participants) and students according to its privileges in the EVE Community.

Figure 2: User interface of EVE-II for supporting e-learning services the training area

The users that participate in the virtual classroom are represented by avatars. The users' avatars are able to make various types of gestures: expressing opinions (e.g. agree, disagree), expressing feelings, mimics (e.g. happy, sad), as well as showing actions (e.g. move learning content, pick learning content). The virtual classroom is supported by audio collaboration, and text chat functionality. Also, it provides a specific place where the users can upload their content and show it to other participants in the course. This space is a 3D presentation table. Moreover this table offers more functionality such as shared whiteboard, or simulation of a brainstorming board. The user interface of the training area is depicted in Fig. 2. More information about the functionality supported by the virtual classroom is available at [2].

4. Conclusions - Future Work

This paper introduces EVE-II, which is a platform for Networked Virtual Environments. This platform can support multi-user 3D spaces along with chat and voice over IP communication. It is based on an open and flexible architecture exploiting well-known and open technologies and standards such as VRML, Java and H.323.

Our next step is the performance monitoring and evaluation EVE-II using networked simulators by conducting the necessary experiments and having this information available in order to trace a path on how the recourses for each type of message should be managed to achieve better performance. Furthermore, the integration of intelligent agents in EVE-II will be a major enhancement of the functionality offered. Intelligent agents can support educational process and they can offer intelligent help to the users for the usage of the system.

5. REFERENCES

[1] Bouras, C., Psaltoulis, D., Psaroudis, C., and Tsiatsos, T. (2001) Protocols for Sharing Educational Virtual Environments. In Proceedings of 2001 International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2001) Split, Dubrovnik (Croatia) Ancona, Bari (Italy), October 09-12, 2001, Vol. II, pp. 659-666.

[2] Bouras C., Tsiatsos T., "Educational Virtual Environments: Design Rationale and Architecture", International Journal of Multimedia Tools and Applications (MTAP), 2003 (accepted under revision).

[3] Bouras C., Tsiatsos T., "Distributed Virtual Reality: Building a Multi-User Layer for the EVE Platform", Journal of Network and Computer Applications (JNCA), Academic Press, 2003 (to appear).

[4] ITU-T, H.323 Recommendation http://www.itu.int/rec/recommendation.asp?type=folders&lang=e&parent=T-REC-H.323

[5] Packetizer, H323 information site, http://www.packetizer.com/iptel/h323/

[6] Araki, Y. "VSPLUS: A high-level multi-user extension library for interactive VRML worlds". In proceedings of the third symposium on Virtual reality modeling language, February 16-20, 1998, Monterey, California, United States, pp.39-47.

[7] Picard, S., Degrande, S., Gransart, C., and Chaillou, C. "VRML data sharing in the spin-3D CVE". In proceeding of the 7th International Conference on 3D Web Technology-Web3D 2002, Tempe, Arizona, USA pp. 165 - 172, 2002.

[8] Web 3D Consortium, http://www.web3d.org.

[9] C. Bouras, T. Tsiatsos, "Architectures and Protocols for Educational Virtual Environments", IEEE International Conference on Multimedia and Expo (ICME2001), Tokyo, Japan, August 22-25 2001.

A State-Transition Model for Distributed Multimedia Documents

P. Bertolotti

Università di Torino

bertolot@di.unito.it

O. Gaggi

Università Ca' Foscari di Venezia

ogaggi@dsi.unive.it

M.L. Sapino

Università di Torino

mlsapino@di.unito.it

Abstract

In this paper we present a state-transition model to describe a multimedia presentation evolution, i.e., its run-time behavior. Each object is modelled as an independent entity with its own behavior and resources allocation: a certain amount of bandwidth, buffer and a display (or an audio channel) for its playback. The evolution of a single media is modelled by means of a finite state machine, in which states transitions are triggered when some specific events occur, provided some conditions hold. The overall presentation is modelled as parallel (or sequential) composition of single media items' executions. The model is well suited for reasoning on multimedia documents dynamics, and to prove properties about them.

1. Introduction

Multimedia presentations can be defined by a collection of different types of media items and a set of spatial and temporal constraints over them. If we consider a distributed environment, media objects are dispersed over a computer network and must be downloaded before playback. Their retrieval from the server(s) is influenced by the network throughput, and buffer resources on the client side must be correctly sized to avoid jitters and stops in the presentation playback.

In this paper we present a model to describe a multimedia presentation evolution, i.e., its run-time behavior. Each object is modelled as an independent entity with its own behavior and resources allocation: a certain amount of bandwidth, buffer and a display (or an audio channel) for its playback. Each object can be considered as a *process* which requires specific *resources*. A correct presentation playback is the result of a correct *scheduling* of retrieval and display of media items.

The purpose of the model presented in this paper is to describe the run-time behavior of a multimedia presentation as the parallel (or sequential) composition of single media items' executions. The model describes the buffers alloca-

tion and deallocation of each media components and can be used as the basis for an algorithm to schedule the download and the playback of complex dynamic elements.

2. A model for multimedia presentations

We refer to the synchronization model for multimedia presentations defined in [3, 4], which we briefly describe addressing the reader to the bibliography for the rationale and the details.

A multimedia presentation is a 4–*tuple* $P = \langle \mathcal{MI}, \mathcal{CH}, \mathcal{E}, \mathcal{SR} \rangle$ where $\mathcal{MI}$ is a set of media items which build the presentation, $\mathcal{CH}$ is a set of channels, i.e., virtual devices used to reproduce media components and mapped to actual resources during their playback, $\mathcal{E}$ is a set of events which will be detailed in Section 3, and $\mathcal{SR}$ is a set of temporal relationships which describe the presentation behavior. An author can design the presentation evolution by imposing a set of temporal constraints among the objects, by means of five synchronization primitives: ***a plays with b*** ($a \Leftrightarrow b$) models the parallel composition of objects a and b; both objects start when either of them is activated, and when object a ends, also b does (if still active); the relation is therefore asymmetric; ***a activates b*** ($a \Rightarrow b$) models the sequential composition of objects a and b; b starts when a ends; ***a is replaced by b*** ($a \rightleftharpoons b$) models the replacement of media item a by b in the same channel; ***a terminates b*** ($a \Downarrow b$) models the stop of media item b, as a consequence of the forced stop of media item a; ***a has priority over b with behavior*** α ($a \overset{\alpha}{>} b$) is used to design presentation behavior during user interactions; media item b is paused ($\alpha = p$) or stopped ($\alpha = s$) when the user starts object a, e.g., through a hyperlink.

3. Description of the system

We consider distributed presentations in which the media to be displayed have to be previously downloaded. This usually requires bufferization. Each channel is therefore associated to a *buffer* to be used by the media item currently occupying that channel.

Buffer management is critical in distributed communication, since it affects the performance and, ultimately, the feasibility of a distributed application, even if limited to media download and presentation. In order to abstract from technical issues which do not limit the model power, we make a number of simplifying assumptions which are plausible in the framework of the multimedia presentations we approach.

First, we assume that the resources provided by the network are adequate, i.e., we face neither QoS problems nor strict real-time constraints in media synchronization; this assumption is acceptable for multimedia presentations, since fine-grain synchronization is resolved by putting synchronized media in the same file, such as multitrack video and audio file. Then, we assume that the time to process a media segment for display, once downloaded, is negligible wrt download time. In the same way, there can be some tolerances, e.g., at the end of a group of objects, there can be a little interval before the next one. Finally, since we assume that the resources are adequate for the whole presentation, we also assume that any parallel combination of media defined in the presentation can be played independently from the media download order, as long as all media are available in core memory when playback starts. In other words, we are interested only in the mutual *logical* relationships among media, and not in performance constraints that can be induced on them by the implementation.

We refer to a simplified double buffer schema over segmented media streams: a media object is divided into segments of equal length, and equal to the length of the buffer. Each time a new segment is required to start, the system switches to the unused buffer and begins to fill it. When the buffer is full, the application can begin media playback (constrained by the synchronization relationships), while the system retrieves from the network a new segment in the other buffer. Other variants (e.g., copying the buffer content into another location, or using a buffer pool) do not introduce significant changes. We shall use the word *buffer* to refer to a buffer area whose allocation policy is not detailed.

We call **pre-fetch** the activity of filling the first buffer for a media item; it defines the minimum delay between download and play in a streaming environment.

Therefore for the remainder of the paper we abstract from any buffer details, and work under the hypothesis that every media is associated with a specific buffer, and that the relevant information about that buffer only concern its being empty, partially filled, or full.

The association of distinct media items with the buffers they are using, is expressed by means of the function $bf(m)$, shorthand for *buffer*, where m denotes the media. Analogously, we assume that every media m is associated to a playback channel, and a stream of data. To denote this association we use the functions $ch(m)$ and $st(m)$.

The relevant information to be checked, when a single media is modelled, is the status of its buffer, its channel and its stream. To check the status of buffers and streams we use the predicates $isEm()$ and $isFu()$, for empty and full respectively.

The channel occupation is given by the function $isUs()$, shortand for *isUsed*: $\mathcal{CH} \rightarrow \mathcal{MI} \cup \{\_\}$ that returns, for every channel, the media item that occupies it. The underscore symbol $\_$ denotes the absence of media item; it is the value used to identify free channels.

In addition, it is possible to verify if the channel associated with a media is available. This control is done using the $isUs()$ function: the channel is available for the media m if the channel is free ($isUs(ch(m)) = \_$) or it is occupied by the media itself ($isUs(ch(m)) = m$). The predicate $isAv()$, that has the media item as its argument, defines the channel availability through these checks.

Each media exhibits its own behavior, which we model in terms of a sequence of different *states* of the media. Media objects are classified as *continuous* media, like video and audio, that once started have their own behavior, and *static* media, like still images, that are simply displayed on the user screen.

Media from the two classes have some states in common, while some other states are specific of continuous or static media items. Both continuous and static media can be **idle**, i.e., not active, waiting to be activated, as well as **init**, that is, pre-fetching data to (dis)play. Continuous media can also be **playing**, the state of media being delivered. The corresponding state for static object is **active**, i.e., actually rendered. Continuous media can be **paused**. When the last segment of a continuous media is playing, the item is in state **terminating**.

If we observe the presentation along time, it can be divided into a number of *states*: these states are more complex than the simple states in which a media is modelled, since the state of a media is embedded in the general state of the system, represented also by means of some *conditions*, that are a set of *facts*, which list a number of atomic conditions that are true in the state. These conditions concern the status of buffer, stream and channel of the media involved in the presentation: this status is checked by means of the predicates and the functions previously introduced in this section.

Therefore, we model the behavior of every single media by means of a *finite state machine*, and the evolution of a presentation, which is a complex dynamic system, as the composition of the machines corresponding to the single atomic components plus the conditions that describe the actual situation of the system.

State transitions are triggered by specific *external* events, that have an effect immediately perceived by the user (requests to start or stop the media) and by *internal*, non observable events, that correspond to some modification in the

internal state of the system, that the user is not necessarily aware of (modifications in buffer status, like the start or the end of pre-fetching phase). In the presence of these events, state transitions are fired, provided some *preconditions*, expressed in terms of logical predicates, hold. The effects of the events on a state of the system are captured by *postconditions* associated to the events.

We denote the set of events that can cause a state transition with $\mathcal{E}$. It includes: $\mathbf{start}_m$, when a media item m is activated; $\mathbf{ready}_m$, when the pre-fetch of a media item is terminated; $\mathbf{pause}_m$, when m playout is temporally interrupted; $\mathbf{stop}_m$, when a user forces the termination of item m; $\mathbf{FF}_m(\mathbf{p})$ (fast forward), when a user asks to play an already active media item m, jumping at position $\mathbf{p}$[1]; $\mathbf{RW}_m(\mathbf{p})$ (rewind), when a user asks to go back to position $\mathbf{p}$ in the playout of media item m; $\mathbf{ending}_m$, when the last segment of the media is starting playing (the item is finishing) and $\mathbf{end}_m$, when a media playout reaches its natural termination.

The external events are *start*, *stop*, *pause, FF(p), RW(p)* and *end*. The internal events are *ready* and *ending*.

An event has a direct impact on the state of the media, causing a state transition of the media, and, more in general, on the state of the system: given an event, its effects are recorded in the new state by (i) deleting, from the current state, those predicate instances which appear negated in the postcondition of the fired transition; (ii) for any positive predicate instance appearing in the postcondition, inserting it in the resulting state. If the predicate instance $p(bf(m_i))$ is the inserted one, (that is, a fact stating something about the buffer of media item m_i), any other predicate $q(bf(m_i))$ appearing in the current state (and concerning the same media item) is then removed (this replacement captures the dynamic evolution of the buffer condition). Some predicates become true (and then are inserted in the current state) as a consequence of the interaction with the environment (these predicates are: $isFu(bf())$, $isEm(bf())$ and $isEm(st())$). That is, some changes that are captured in the state of the system are not induced by any media state transition. They are instead a reaction to some modification in the environment.

4. Single media item and composition of items

We introduce an independent finite state machine modelling a single media item, that encapsulates the functional and timing properties of the media object.

Definition 4.1 (Single Item Finite State Machine) The finite state machine characterizing a *continuous* media item m is $MSM_m = \langle S, s_0, F, next, T \rangle$, where (i)

---

[1]The position $\mathbf{p}$ is determined by the user. Then the system selects the correct segment number and begins to fill the buffer of the media item.

next(**id**, start$_m$) = **in**   next(**in**, ready$_m$) = **pl**
next(**pl**, RW$_m$(p)) = **in**   next(**pl**, FF$_m$(p)) = **in**
next(**pl**, ending$_m$) = **tr**   next(**tr**, end$_m$) = **id**
next(**ps**, start$_m$) = **pl**
next(**x**, pause$_m$) = **ps**, **x** $\in$ { **in, pl, tr**}
next(**x**, stop$_m$) = **id**, **x** $\in$ { **in, pl, tr, ps**}

Table 1. The function next: id=idle, in=init, pl=playing, s=paused, tr=terminating.

$S = \{\mathbf{idle}, \mathbf{init}, \mathbf{playing}, \mathbf{paused}, \mathbf{terminating}\}$; (ii) $s_0 = \mathbf{idle}$; (iii) $F = \{\mathbf{idle}\}$; (iv) $next$ is the function defined in Table 1; (v) T is a set of 4-tuples $\langle s, e, \mathcal{C}, \mathcal{P} \rangle$ describing transitions, where $s \in S$ is the initial state, $e \in \mathcal{E}$ is an event, $\mathcal{C}$ is a set of enabling conditions for the transition from state s when the event e occurs and $\mathcal{P}$ is a set of postconditions, that is, conditions holding after the transition takes place.

A finite state machine MSM_m modelling a static media m is obtained by the MSM_m previously defined removing states **paused** and **terminating** (and the transactions involving these states) and renaming state **playing** in state **active**.

In the following, to characterize state transitions we use the following notation, where $state_i$ is the initial state and $state_r$ is the resulting state: $[\mathcal{C}] \, \mathbf{state_i} \xrightarrow{e} \mathbf{state_r} \, [\mathcal{P}]$.

State transitions take place when an event occurs, and their enabling conditions are satisfied. Preconditions and postconditions mentioned in our transitions only concern local predicates, i.e., predicates whose truth value might be affected by the firing transition. For not mentioned predicates, persistency is assumed.

The set of transitions characterizing a media item is shown in Table 2.

Given this representation, we can model a presentation which contains several media objects, by composing the corresponding finite state machines. Several unrelated media may exist in the presentation, therefore we first model a system containing a number of independent media. Then, we specialize some transition rules, to model synchronization primitives.

In the following definition, we shall use footers to distinguish different media, and the footer corresponding to each media also refers to its states and events, thus distinguishing between analogous states and events for different media.

We will denote with $\mathcal{C}_{s_i,e_i}$ the enabling condition for the transition corresponding to event e_i in the state s_i, for the media item m_i. Analogously for postconditions $\mathcal{P}_{s_i,e_i}$.

Definition 4.2 Let $m_1, \ldots, m_n$ be n independent media items, and $MSM_1, \ldots, MSM_n$ be the corresponding finite state machines. Let $MSM_i = \langle S_i, s_i^0, F_i, next_i, T_i \rangle$, for all $i = 1, \ldots, n$. The overall behavior is modelled

$$[isAv(m_i)] \quad \mathbf{id} \stackrel{start}{\rightarrow} \mathbf{in} \quad [\neg isEm(bf(m_i))\wedge isUs(ch(m_i))=m_i]$$

$$[isFu(bf(m_i))] \quad \mathbf{in} \stackrel{ready}{\rightarrow} \mathbf{pl} \quad [\neg isEm(st(m_i))]$$

$$[true] \quad \mathbf{in} \stackrel{pause}{\rightarrow} \mathbf{ps} \quad [true]$$

$$[true] \quad \mathbf{in} \stackrel{stop}{\rightarrow} \mathbf{id} \quad [isEm(bf(m_i))\wedge isUs(ch(m_i))=\_]$$

$$[isEm(st(m_i))] \quad \mathbf{pl} \stackrel{ending}{\rightarrow} \mathbf{tr} \quad [true]$$

$$[true] \quad \mathbf{pl} \stackrel{FF(p)}{\rightarrow} \mathbf{in} \quad [isEm(bf(m_i))]$$

$$[true] \quad \mathbf{pl} \stackrel{RW(p)}{\rightarrow} \mathbf{in} \quad [isEm(bf(m_i))]$$

$$[true] \quad \mathbf{pl} \stackrel{pause}{\rightarrow} \mathbf{ps} \quad [true]$$

$$[true] \quad \mathbf{pl} \stackrel{stop}{\rightarrow} \mathbf{id} \quad [isEm(bf(m_i))\wedge isUs(ch(m_i))=\_]$$

$$[isAv(m_i)] \quad \mathbf{ps} \stackrel{start}{\rightarrow} \mathbf{pl} \quad [isUs(ch(m_i))=m_i]$$

$$[true] \quad \mathbf{ps} \stackrel{stop}{\rightarrow} \mathbf{id} \quad [isEm(bf(m_i))\wedge isUs(ch(m_i))=\_]$$

$$[isEm(bf(m_i))] \quad \mathbf{tr} \stackrel{end}{\rightarrow} \mathbf{id} \quad [isEm(bf(m_i))\wedge isUs(ch(m_i))=\_]$$

$$[true] \quad \mathbf{tr} \stackrel{pause}{\rightarrow} \mathbf{ps} \quad [true]$$

$$[true] \quad \mathbf{tr} \stackrel{stop}{\rightarrow} \mathbf{id} \quad [isEm(bf(m_i))\wedge isUs(ch(m_i))=\_]$$

$$[isFu(bf(m_i))] \quad \mathbf{in} \stackrel{ready}{\rightarrow} \mathbf{ac} \quad [true]$$

$$[true] \quad \mathbf{ac} \stackrel{stop}{\rightarrow} \mathbf{id} \quad [isEm(bf(m_i))\wedge isUs(ch(m_i))=\_]$$

Table 2. Transition rules for independent media items: id=idle, in=init, pl=playing, ps=paused, tr=terminating, ac=active.

by the finite state machine $MSM = \langle S, s^0, F, next, T\rangle$, where (i) $S = \{\langle s_1, \ldots, s_n\rangle \mid s_i \in S_i, i = 1, \ldots, n\}$; (ii) $s^0 = \langle s_1^0, \ldots, s_n^0\rangle$; (iii) $F = \{\langle s_{f_1}, \ldots, s_{f_n}\rangle \mid s_{f_i} \in F_i, i = 1, \ldots, n\}$; (iv) $next(\langle s_1, \ldots, s_i, \ldots, s_n\rangle, e_{m_i}) = \langle s_1, \ldots, next_i(s_i, e_{m_i}), \ldots, s_n\rangle$, for any $s_i \in S_i$, and any event e_{m_i} on the the media m_i, $i = 1, \ldots, n$; (v) T contains the following transitions: $\forall t$, if $t = \langle s_i, e_{m_i}, \mathcal{C}_{s_i, e_{m_i}}, \mathcal{P}_{s_i, e_{m_i}}\rangle \in T_i$ for a given i then $\langle\langle s_1, \ldots, s_i, \ldots, s_n\rangle, e_{m_i}, \mathcal{C}_{s_i, e_{m_i}}, \mathcal{P}_{s_i, e_{m_i}}\rangle \in T$.

In a presentation some media are related each other, therefore we must consider objects which are temporally related by the synchronization relationships described in Section 2. We translate the temporal relations into different composition of finite state machines, which are mostly based on the one considered above.

Specifically, the finite state machine modelling media items related by any temporal composition $m_i\theta m_j$, $\theta \in \{\Leftrightarrow, \Rightarrow, \Downarrow, \rightleftharpoons, \stackrel{s}{>}, \stackrel{p}{>}\}$ is defined by: (i) applying the definition 4.2, to model the case of the general composition of items m_i and m_j, and (ii) adding some transition rules which will be described later in this section possibly overwriting already existing transition rules. Given a transition

$t \in T$ for an event e from $\langle s_{m_i}, s_{m_j}\rangle$ to $\langle s'_{m_i}, s'_{m_j}\rangle$, if $t' = \langle\langle s_{m_i}, s_{m_j}\rangle, e, \mathcal{C}_{\langle s_{m_i}, s_{m_j}\rangle, e}, \mathcal{P}_{\langle s_{m_i}, s_{m_j}\rangle, e}\rangle \in T$ reaching the same state $\langle s'_{m_i}, s'_{m_j}\rangle$ already exists, t replaces t', otherwise t is added to the set of transitions T.

In order to modify the transition rules described in Table 2 for taking into account the effect of synchronization relationships on the media finite state machines composition, we define the notion of closure of an item, with respect to some synchronization relations, to capture the effects of event propagation among media.

Definition 4.3 ($SC_\Leftrightarrow$) The symmetric closure of a media item a wrt. $\Leftrightarrow$ is the set $SC_\Leftrightarrow(a)$ such that (i) $a \in SC_\Leftrightarrow(a)$, and (ii) for any item $c \in \mathcal{MI}$, if $\exists b \in SC_\Leftrightarrow(a)$ such that $b \Leftrightarrow c \in \mathcal{SR}$ or $c \Leftrightarrow b \in \mathcal{SR}$, then $c \in SC_\Leftrightarrow(a)$.

$SC_\Leftrightarrow(a)$ contains all media items related by a $\Leftrightarrow$ relationship, that are required to start simultaneously, when one of them is activated. From the definition $SC_\Leftrightarrow(b) = SC_\Leftrightarrow(a)$ iff $b \in SC_\Leftrightarrow(a)$.

The set $SC_\Leftrightarrow(a)$ results from the closure of a transitive chaining process. Basically, it includes all items which are transitively connected to a, by means of $\Leftrightarrow$ relationship. As it will be clearer in the following, there are cases in which some of the connected items have to be discarded. In this case, when computing the symmetric closure of a wrt. $\Leftrightarrow$, items connected to a have to be included in the closure only if the "connecting chain" does not include any discarded item. This notion of restricted closure is formalized in the following definition.

Definition 4.4 ($SC_\Leftrightarrow(a)_M$) The symmetric closure of item a wrt. $\Leftrightarrow$, limited by the set of items M, is the set $SC_\Leftrightarrow(a)_M$ such that (i) if $a \in M$ then $SC_\Leftrightarrow(a)_M = \emptyset$, else (ii) $a \in SC_\Leftrightarrow(a)_M$, and for any item $c \in \mathcal{MI} \setminus M$, if $\exists b \in SC_\Leftrightarrow(a)_M$ such that $b \Leftrightarrow c \in \mathcal{SR}$ or $c \Leftrightarrow b \in \mathcal{SR}$, then $c \in SC_\Leftrightarrow(a)_M$.

All the synchronization primitives of the model exhibit an asymmetric behavior. For some of them, a notion of transitive (forward) closure is needed, to deal with the forward propagation of the effects of an event. For the sake of space we introduce a *parameterized* asymmetric closure, in which the parameter rel acts as a place holder for $\Leftrightarrow$ or $\Downarrow$ synchronization primitives.

Definition 4.5 (C_{rel}) Let a be a media item in $\mathcal{MI}$ and $rel \in \{\Leftrightarrow, \Downarrow\}$. The closure of a wrt. rel is the set $C_{rel}(a)$ such that (i) $a \in C_{rel}(a)$, and (ii) for any item $b \in \mathcal{MI}$, if $b \in C_{rel}(a)$ and $b\ rel\ c \in \mathcal{SR}$, then $c \in C_{rel}(a)$.

Six new transition rules define the evolution of a composite presentation in case of events which have a wider impact on the document activating a cascade of simultaneous media activations or stops. Such events are **start**, **ready**,

stop, **ending** and **end**. When the system receives an event $start_{m_i}$, all media items which are related by a $\Leftrightarrow$ relationship, i.e., all media items in $SC_\Leftrightarrow(m_i)$, should begin to fill the buffer. The system controls if their channels are available (media items in $SC_\Leftrightarrow(m_i)_{NotAvailable}$) or if there are some media objects to replace, and in this case changes their states to **init** (and consequently, the states of the replaced objects become **idle**).

Event: $\mathbf{ready_{m_i}}$ for any $i \in \{1 \dots n\}$
Notation:
$\quad NotAv = \{m_k \in \mathcal{MI} | \neg isAv(m_k)\}$
$\quad TR = \{m_k \in NotAv | s_k = \mathbf{tr_{m_k}}\}$
$\quad StartCl = SC_\Leftrightarrow(m_i)_{NotAv}$
$\quad IP = \{m_j \in StartCl | s_j = \mathbf{in_{m_j}} \vee \mathbf{s_j} = \mathbf{ps_{m_j}}\}$
$\quad Ready = \{m_j \in StartCl | isFu(bf(m_j))\}$
$\quad Free = \{m_j \in StartCl | s_j = \mathbf{id_{m_j}} \vee \mathbf{s_j} = \mathbf{ps_{m_j}}$
$\qquad \wedge isAv(m_j)\}$
$\quad Paused = \{m_k | (m_k^p \overset{p}{>} m_k) \in \mathcal{SR}$ such that $s_k \neq \mathbf{id_{m_k}}$
$\qquad$ for some $m_k^p \in SC_\Leftrightarrow(m_i)\}$
Precondition:
$\quad IP = Ready \wedge Free = \emptyset \wedge TR = \emptyset \wedge isFu(bf(m_i))$
Initial state: $\langle \mathbf{s_1}, \dots, \mathbf{s_n} \rangle$
Final state: $\langle \mathbf{s_1'}, \dots, \mathbf{s_n'} \rangle$, where
$\quad \forall m_j \in IP, s_j' = \mathbf{pl_{m_j}}$ if m_j continuous,
$\quad s_j' = \mathbf{ac_{m_j}}$ if m_j static;
$\quad \forall m_j \in Paused, s_j' = \mathbf{ps_{m_j}}$;
$\quad \forall m_j \notin IP \notin Paused, s_j' = s_j$
Postcondition: $\forall m_j \in IP, \neg isEm(st(m_j))$

Table 3. Event: $\mathbf{ready_{m_i}}$.

In case of event $ready_{m_i}$ the system controls if all media items have their buffers full, and in this case the presentation begins its playback (Table 3). Otherwise, the system waits for media objects already buffering and controls if there are other media items in $SC_\Leftrightarrow(m_i)$ for which the channel is now available and begins their bufferization.

If the user stops the playback of an item m_i, the system looks for all media items which must be stopped at the same time, i.e., all objects contained in $C_\Downarrow(m_i)$, frees their channels and changes their states to **idle** again. Otherwise, if m_i naturally ends, the system first notices that its stream is empty (event *ending*) and then that also the buffer is empty (event *end*). When the system receives the event $ending_{m_i}$, it checks what media objects must be started after its end, i.e., media items in $SC_\Leftrightarrow(m)$ such that a relationship $m_i \Rightarrow m$ exists and checks if their channels are available[2], or if some media items can be replaced. Then the system changes the states of media objects whose channels are available to **init** and begins their bufferization (Table 4).

_____
[2]Since the stream is already empty, channel of m_i is considered available for new items.

Event: $\mathbf{ending_{m_i}}$ for any $i \in \{1 \dots n\}$
Notation:
$\quad Started = \{m_k | (m_i \Rightarrow m_k) \in \mathcal{SR}\}$
$\quad NotAv = \{m_k \in \mathcal{MI} | \neg isAv(m_k)\}$
$\quad Repl^{ing} = \{m_j^r | m_j^r \in SC_\Leftrightarrow(m_k)$ for some $m_k \in Started$
$\qquad \wedge isUs(ch(m_j^r)) = m_j$ for some $(m_j \rightleftharpoons m_j^r) \in \mathcal{SR}$
$\qquad \vee(m_j^r \overset{\alpha}{>} m_j) \in \mathcal{SR} \vee m_j = m_i\}$
$\quad StartCl = \bigcup_{m_k \in Started} SC_\Leftrightarrow(m_k)_{NotAv \setminus Repl^{ing}}$
$\quad IP = \{m_j \in StartCl | s_j = \mathbf{id_{m_j}} \vee \mathbf{s_j} = \mathbf{ps_{m_j}}\}$
$\quad Repl^{ed} = \{m_k | (m_k \rightleftharpoons m_k^r) \in \mathcal{SR} \vee (m_k^r \overset{\alpha}{>} m_k) \in \mathcal{SR}$
$\qquad$ for some $m_k^r \in Repl^{ing}\}$
$\quad StopCl = \bigcup_{m_k \in Repl^{ed}} C_\Downarrow(m_k)$
$\quad I = \{m_j \in StopCl | s_j \neq \mathbf{id_{m_j}}\}$
Precondition: $isEm(st(m_i)) \wedge |IP| = |channel(IP)|$
Initial state: $\langle \mathbf{s_1}, \dots, \mathbf{pl_{m_i}}, \dots, \mathbf{s_n} \rangle$
Final state: $\langle \mathbf{s_1'}, \dots, \mathbf{tr_{m_i}}, \dots, \mathbf{s_n'} \rangle$, where
$\quad \forall m_j \in IP, s_j' = \mathbf{in_{m_j}}$;
$\quad \forall m_j \in I, s_j' = \mathbf{id_{m_j}}$;
$\quad \forall m_j \notin IP \notin I, s_j' = s_j$
Postcondition:
$\quad \forall m_j \in I, isEm(bf(m_j))$;
$\quad \forall m_j \in I \setminus Repl^{ed}, isUs(ch(m_j)) = \_$;
$\quad \forall m_j \in IP, \neg isEm(bf(m_j)) \wedge isUs(ch(m_j)) = m_j$

Table 4. Event: $\mathbf{ending_{m_i}}$.

When m_i naturally ends (Table 5), the system stops all media items in $C_\Leftrightarrow(m_i)$, i.e., media items currently playing in parallel, and frees their channels. Then it checks if there are some media objects that must be started after its end, and whose channels are now available. In this case changes their states to **init** and begins their bufferization.

A complete description of these transition rules, which are not detailed here due to lack of space, can be found in [1]; Tables 3, 4 and 5 summarize the events which are used in the example of Section 5.

5. An example

We introduce now an example to show how these rules are used. Let us consider a multimedia presentation about an artwork: an initial video ($intro$) introduces the history period related to the artwork and the artist who made it. At its end, activates another video clip ($vclip$) illustrating the artwork itself. This second clip plays in parallel with a soundtrack ($sound$) and a comment page ($caption$). At the end of $vclip$, a text page ($text$) is displayed, with information about the museum which contains the artwork. The channels are *video*, *audio* and *window*, such that $ch(intro) = ch(vclip) = video$, $ch(sound) = audio$ and $ch(caption) = ch(text) = window$.

Due to space constraints, we only comment a fragment

Event: $\mathbf{end_{m_i}}$ for any $i \in \{1 \ldots n\}$

Notation:

$End = \{m_j \in C_{\Leftrightarrow}(m_i)|s_j \neq \mathbf{id_{m_j}}\}$

$Started = \{m_k|(m_i \Rightarrow m_k) \in \mathcal{SR}\}$

$NotAv = \{m_k \in \mathcal{MI}|\neg isAv(m_k)\}$

$Repl^{ing} = \{m_j^r|m_j^r \in SC_{\Leftrightarrow}(m_k) \text{ for some } m_k \in Started$
$\quad \wedge isUs(ch(m_j^r)) = m_j \text{ for some } (m_j \rightleftharpoons m_j^r) \in \mathcal{SR}$
$\quad \vee (m_j^r \overset{\alpha}{>} m_j) \in \mathcal{SR} \vee m_j \in End\}$

$StartCl = \bigcup_{m_k \in Started} SC_{\Leftrightarrow}(m_k)_{NotAv \setminus Repl^{ing}}$

$IP = \{m_j \in StartCl|s_j = \mathbf{id_{m_j}} \vee \mathbf{s_j} = \mathbf{ps_{m_j}}\}$

$Ready = \{m_j \in StartCl|isFu(bf(m_j))\}$

$Repl^{ed} = \{m_k|(m_k \rightleftharpoons m_k^r) \in \mathcal{SR} \vee (m_k^r \overset{\alpha}{>} m_k) \in \mathcal{SR}$
$\quad \text{for some } m_k^r \in Repl^{ing}\}$

$StopCl = \bigcup_{m_k \in (Repl^{ed} \cup End \setminus \{m_i\})} C_{\Downarrow}(m_k)$

$I = \{m_j \in StopCl|s_j \neq \mathbf{id_{m_j}}\}$

$ChFree = \{c|\exists m \in (I \cup End) \wedge c = ch(m) \wedge$
$\quad isUs(c) = m\} \setminus \{c|\exists m\ c = ch(m) \wedge m \in IP\}$

Precondition: $isEm(bf(m_i)) \wedge |IP| = |channel(IP)|$

Initial state: $\langle \mathbf{s_1}, \ldots, \mathbf{tr_{m_i}}, \ldots, \mathbf{s_n} \rangle >$

Final state: $\langle \mathbf{s_1'}, \ldots, \mathbf{s_n'} \rangle$, where

$\quad \forall m_j \in End \cup I,\ s_j' = \mathbf{id_{m_j}};$

$\quad$ if $StartCl = Ready\ \forall m_j \in Ready,\ s_j' = \mathbf{pl_{m_j}}$
$\quad\quad$ if m_j continuous, $s_j' = \mathbf{ac_{m_j}}$ if m_j static;

$\quad$ else $\forall m_j \in IP,\ s_j' = \mathbf{in_{m_j}};$

$\quad \forall m_j \notin End \notin Ready \notin IP \notin I,\ s_j' = s_j$

Postcondition:

$\quad \forall m_j \in I \cup End,\ isEm(bf(m_j));$

$\quad \forall c_j \in ChFree,\ isUs(c_j) = \_;$

$\quad$ if $StartCl = Ready\ \forall m_j \in Ready,\ \neg isEm(bf(m_j));$

$\quad$ else $\forall m_j \in IP,\ \neg isEm(bf(m_j)) \wedge isUs(ch(m_j)) = m_j$

Table 5. Event: $\mathbf{end_{m_i}}$.

of the presentation; a more detailed discussion on the example is in [1]. Consider the presentation when only the introduction video is playing, i.e., media items are in the following states: $\mathbf{playing_{intro}}$, $\mathbf{idle_{vclip}}$, $\mathbf{idle_{sound}}$, $\mathbf{idle_{caption}}$, $\mathbf{idle_{text}}$ and the following conditions hold: $isUs(video) = intro$, $isUs(audio) = \_$, $isUs(window) = \_$ and $isFu(bf(intro))$.

Suppose $ending_{intro}$ occurs (i.e., the data stream corresponding to the introduction became empty, i.e., $isEm(st(intro))$). The system then controls which items should be activated at the end of video $intro$, i.e., $vclip$, $sound$ and $caption$ and checks if their channels are available as described in Table 4. Therefore, the current states of media items are: $\mathbf{termitating_{intro}}$, $\mathbf{init_{vclip}}$, $\mathbf{init_{sound}}$, $\mathbf{init_{caption}}$, $\mathbf{idle_{text}}$.

When the condition $isEm(bf(intro))$ holds, event end_{intro} occurs. A state transition takes place (Table 5), and the states of the media become $\mathbf{idle_{intro}}$, $\mathbf{init_{vclip}}$, $\mathbf{init_{sound}}$, $\mathbf{init_{caption}}$, $\mathbf{idle_{text}}$. When the buffers of the video clip, the music and the caption page are full, the sys-

tem processes the last $ready$ event as described in Table 3, activating these elements (i.e., moving presentation to $\mathbf{idle_{intro}}$, $\mathbf{playing_{vclip}}$, $\mathbf{playing_{sound}}$, $\mathbf{active_{caption}}$, $\mathbf{idle_{text}}$).

When also the stream of the video about the artwork $vclip$ becomes empty, the system receives the event $ending_{vclip}$, and changes the states of media items beginning the download of the final text pages according to Table 4: $\mathbf{idle_{intro}}$, $\mathbf{terminating_{vclip}}$, $\mathbf{playing_{sound}}$, $\mathbf{active_{caption}}$, $\mathbf{idle_{text}}$. When $bf(vclip)$ is empty, $vclip$ ends. As described in Table 5, the states of the media become $\mathbf{idle_{intro}}$, $\mathbf{idle_{vclip}}$, $\mathbf{idle_{sound}}$, $\mathbf{idle_{caption}}$, $\mathbf{init_{text}}$. Then, event $ready_{text}$ occurs when the buffer $bf(text)$ is full. Therefore, there is a transition to $\mathbf{idle_{intro}}$, $\mathbf{idle_{vclip}}$, $\mathbf{idle_{sound}}$, $\mathbf{idle_{caption}}$, $\mathbf{active_{text}}$ (see Table 3).

6. Conclusion

The abstract formal model introduced so far describes the behavior of a multimedia presentation in terms of resources allocation (buffers and network bandwidth) and synchronization among the media objects.

For this reason it can be used to define and check a sequence of media items download: during the presentation playback, the system calculates *a priori* what happens at the end of a component, i.e., which objects are activated (see Table 4), and finds out a correct scheduling download sequence for their bufferization.

The proposed model is also well suited for reasoning on multimedia documents dynamics, and to prove properties about them. For example, given a set of media items Act, the model can check if it is possible that the presentation reaches a state in which all of them are active. In this case, we need the complete finite state machine associated with the multimedia documents and a description of the initial state of the overall system, expressed in terms of (positive) predicates on media buffers and channels. Then, we can look in the composite finite state machine for a presentation state in which all media $m \in Act$ are in state **active** (if m is a static media) or **playing** (if m is a continuous item) and return the shortest sequence of events to reach that state.

Another interesting property, is the correctness of a sequence of media items download with respect to the modelled behavior of a presentation. The model can check if a given sequence is correct since the finite state machine completely describes the status of the buffers and channels at each moment. Therefore, we can control if the sequence of events $ready$ is compatible with the finite state machine associated to the presentation. For example, if n media items must begin to playback in parallel, they change their states from state *init* to **playing** when the last buffer (i.e., the buffer corresponding to the object with biggest delay)

is full (see Table 3). Therefore the set of possible correct download sequences contains all sequences which respect this property, no matter whether the system begins to fill some buffers before the others.

Other works approach the problem of multimedia scheduling. Candan et al [2] define a model to design and play multimedia presentations. Differently from our approach, it does not describe all the possible run-time behaviors of a multimedia document, but only the dynamic structure as designed by the author, through the use of a graph in which media items are the nodes and the edges are flexible temporal constraints among the objects. A possible presentation schedule (and the resources allocation) can be derived by the graph, but the model is not well suited to check other properties of the document.

In [7] the authors propose a new CPU scheduling technique to improve performance of multimedia and real-time applications, in which the management of events delivery is a critical point to avoid delays. The idea is to coordinate event scheduling and task scheduling by making the multimedia applications *event-aware*. The domain here is little different from the one addressed in this paper and includes multimedia application like virtual worlds or multi-player games.

Paulo et al. [6] describe an approach very similar to the one addressed here. The paper presents a synchronization model based on hypercharts, an extension of the finite state machine formalism. A hyperchart contains timed transitions to specify the temporal behavior of presentation activities whose firing depends on the state of the system, and the system performs a single step at each time unit, reacting to all external changes that happen in that time interval. Hypercharts provide mechanisms for specifying hypermedia requirements such as objects duration, delays, jitters and user interactions, but require a explicit definition of the time instant at which an event occurs, therefore our model allows a easier management of further modifications of a multimedia document.

In [5] Layaïda et al. discuss the effect of uncertainty in the duration of some media objects in multimedia scenarios. Media items can be distributed over the internet and the access delay can be very different. Different from our approach, users interactions with the document cause de-synchronization. The model proposes a scheduling algorithm based on flexibility to solve the problem of re-synchronization.

Our model provides a more general framework that allows to define a correct sequence of download for the media items of a multimedia presentation, as well as to investigate other properties of the real time behavior of the document. This second feature is not considered in the models present in literature.

In the future, we plan to develop a formal system to reason within this model, by properly defining axioms and proof rules, according to the methods usually adopted for program verification and model checking.

Acknowledgements

References

[1] P. Bertolotti, O. Gaggi, and M.L. Sapino. A State-Transition Model for Distributed Multimedia Documents. Technical Report 77/04, Department of CS, University of Turin, http://www.di.unito.it/~bertolot/tech-report.pdf, March 2004.

[2] K.S. Candan, B. Prabhakaran, and V.S. Subrahmanian. Retrieval Schedules Based on Resource Availability and Flexible Presentation Specifications. *Multimedia Systems*, 6(4):232–250, 1998.

[3] A. Celentano, O. Gaggi, and M.L. Sapino. Retrieval in Multimedia Presentations. *ACM Multimedia Systems Journal*, to appear.

[4] O. Gaggi and A. Celentano. Modelling Synchronized Hypermedia Presentations. *Multimedia Tools and Applications*, to appear.

[5] N. Layaïda, L. Sabry-Ismail, and C. Roisin. Dealing with Uncertain Durations in Synchronized Multimedia Presentations. *Multimedia Tools and Applications*, 18(3):213–231, december 2002.

[6] F.B. Paulo, P.C. Masiero, and M.C. Ferreira de Oliveira. Hypercharts: Extended Statecharts to Support Hypermedia Specification. *IEEE Transactions on Software Engineering*, 25(1):33–49, January/February 1999.

[7] C. Poellabauer, K. Schwan, and R. West. Coordinated CPU and Event Scheduling for Distributed Multimedia Applications. In *ACM Multimedia Conference*, pages 231–240, 2001.

Efficient Patching Scheduling for a Multimedia Streaming Proxy

Kuei-Chung Chang, Yan-Yo Lin and Tien-Fu Chen

{ckj, lyy88, chen}@cs.ccu.edu.tw

Department of Computer Science and Information Engineering

National Chung Cheng University, Taiwan

Abstract

Patching can effectively reduce the server workload and the network traffic for streaming servers. It enables a client to receive a multicast streaming by listening to an ongoing transmission of the same video clip for reducing retransmitted frames from the server. Greedy Buffer Reuse (GBR) is a patching algorithm that allows clients to patch from multiple ongoing transmissions. However, multimedia streams multicasting to clients must be stored in the client buffer, and fractions of multimedia streams will be lost due to the limited client buffer size. Existing patching schemes do not cache the transmitted frames patched from the server for reducing nwtwork overheads. In this paper, we first propose a patching scheduling algorithm based on the GBR algorithm to schedule each patching frame on the ongoing streaming to be cached in the proxy. The patching proxy will generate a Proxy Cache Schedule (PCS) to specify which frames are cached on the proxy, and a new channel transmission schedule (NCS_i) to specify how the frames are transmitted on channel i from the server. This will further reduce frames that need to be patched from the server. The simulation results show that the patching proxy can reduce sizeable reduction in transmission overhead with small cache space when a video clip is very popular in the multimedia server.

1. Introduction

Multimedia streaming applications [5] get more and more popular in the Internet. They consume a significant amount of server and network resources due to the high bandwidth and long duration of audio and video clips. If a video is popular in the server, a large number of requests will make the server overloaded and the network congestion. Patching can be expanded dynamically to serve more clients by multicast. It reduces the server and the network overhead by allowing a client to receive a multimedia streaming from an ongoing transmission of the same video clip. *Greedy Buffer Reuse* (GBR) algorithm [14] minimizes the required transmission bandwidth by allowing clients to patch from multiple ongoing transmissions. However, only few fractions of streaming frames will be cached in limited client buffer. Therefore, the client needs to transmit other un-cached frames from the server. The streaming proxy can cache streaming frames and forward those frames to clients. It reduces the server workload and the network traffic if the streaming proxy can efficiently cache frames that will be referenced again in short period. Therefore, our objective is to improve the utilization of the streaming proxy cache by properly scheduling the patch frames in the proxy.

In this paper, we propose a *patching scheduling* algorithm in the proxy to reschedule those frames that will be patched from the server to clients. Our patching scheduling algorithm is based on the GBR algorithm. The GBR algorithm maximized the number of frames that a new client can retrieve from this most recently initiated ongoing complete transmission by scheduling the transmission channel. The streaming proxy uses the patching scheduling algorithm to find a *Proxy Cache Schedule* (PCS) to dynamically cache frames that will be requested again soon. Patching scheduling algorithm cache frames which are lost due to limited client buffer size, thus the proxy cache can be treated as the extension buffer of each client.

The major contribution of this paper is that we introduces the patching proxy that combines the patching algorithm and caching scheduling algorithm to reduce the server and network overheads dramatically. In the large scale system like mobile network, handheld devices can retrieve video streams from the patching proxy, and video streams can be played more smoothly. Proxy-based patching is useful if multicasting capability is not available from the content server to clients.

The remainder of this paper is organized as follows. In Section 2, we review related researches in the multimedia streaming. In Section 3, we describe our system architecture and the patching scheduling algorithm. In Section 4, we show the experiment environment and simulation results. Finally, we conclude our paper in Section 5.

2. Related Work

Video streams typically have high transmission bandwidth requirement even with compressing. For popular video streams, client requests may arrive closely. To

avoid the server crash and reduce network congestion, requests for the same video arriving within short time duration are bunched together and served in a batch by a single data stream [1, 16]. The drawback of aggregating these requests is that it increases playback latency of clients. Other techniques are proposed such as dynamic multicast [4, 18, 19], patching techniques [3, 12] and dynamic caching [6, 2, 8, 10, 11, 13, 17].

Greedy patching algorithm [9] is a simple patching algorithm, which schedules a new regular multicast for a batch only if there is no regular multicast currently serving the same video. *Grace patching algorithm* [9] is to improve the greedy patching algorithm which only share last B time units of the regular stream when the video size of the period between new request and last regular multicast exceeds the client buffer space.

Since the Greedy patching and the Grace patching do not fully exploit the client buffer space or the ability of the client to listen to more than one ongoing transmission. *Periodic Buffer Reuse* (PBR) algorithm [14] maximizes the amount of data that a subsequent client can receive from the existing complete transmission (*regular stream*), even if the client buffer is not large enough to store the entire sliding window of frames.

Although PBR maximizes the amount of data that a client can receive from an existing complete transmission, it does not exploit the potential of receiving data from more than one ongoing transmission. The *Greedy Buffer Reuse* (GBR) algorithm [14] schedules the client to receive frames as late as possible from an ongoing transmission, or directly from the server.

Most people would like to play a video at beginning and usually do not play it to the end. Therefore, the prefix frames of a video are accessed with high probability. That is why the prefix caching [7, 15] is necessary.

3. Patching Model of the Streaming Proxy

3.1. System model

Figure 1 is the model of our proposed patching streaming proxy. The objective of the patching server is to reduce the workload of the multimedia server and to reduce the network congestion by sharing the online streaming frames. In this model, we put the patching server into the proxy and cooperate with the proxy cache. The proxy contained the patching server can reduce the network bandwidth between the video server and clients. In our system model, the buffer is used to keep portion of multimedia streams both in the patching proxy and clients. Frames are stored in the buffer of the client when they arrive before their playback time. When requests arrived simultaneously, they are served as a single batch. As the requests arrive asynchronous, they will be served in the order of the request time slot. Because the streaming proxy must send multimedia streaming from the server to clients and it must cache portion of streaming

frames as soon as possible. The patching proxy produces a *new channel schedule* (NCS_i) for the multimedia server, and determines a *reception schedule* (RS_i) for each client i. It also generates a *proxy caching schedule* (PCS) for the proxy to cache streaming frames. In our system model, we only focus on a single video, because the requests for different streams do not interact to each other.

Figure 1. The streaming service model

3.2. Motivation of patch cache scheduling

We give an example to explain GBR algorithm in the patching server and explain why we use the proxy cache to enhance the patching algorithm. In this example, we assume that there is a video clip composed of 20 frames in the multimedia server and requested by three clients. The second request R_2 and the third request R_3 arrived at time 5t and 6t relative to the first request R_1 respectively. The client buffer is assumed to hold three frames. Request R_1 cannot receive any frames from the ongoing streaming because there is no existing channel for this video. The system may contribute the benefit after the first request.

In Figure 2, we can find out that all frames of request R_1 will be transmitted on channel C_1. The frames of request R_2 may receive from the channel C_1 behind frame 6. The capacity of client 2 buffer is limited to hold three frames, thus each frame behind frame 6 must be checked that the frames received from the existing channels (such as C_1) can be stored in the client 2 buffer. *Frames 6, 7, 8, 11, 12, 13, 16, 17, 18 of request R_2 will be received from channel C_1. Frames 1, 2, 3, 4, 5 are missed when request R_2* arrived. When request R_3 arrives, more ongoing streams could be chosen. *Frames 7, 8, 11, 13, 16, 17 of request R_3* will be received from channel C_1, and *frames 2, 3, 4, 5, 9, 10, 14, 15, 19, 20 of requests R_3* will be received from channel C_2. Therefore, it has an opportunity to make the frames be shared by the clients closer to their playback time. It increases the utilization of the client buffer, and thus share more frames on channels. We can find out that the number of patching frames is reduced when request R_3 patches frames from the server.

Figure 2. Three requests to the same video clip

Figure 4. Objective of the cache scheduling

We can find out that some frames are unable to be stored in the client buffer because of the limited client buffer. It makes the server retransmit the same frames at different time slots. We can use a patching proxy to cache a portion of the video streaming to reduce frames which are patched from the server at different time slots.

For example, we assume that the proxy cache is limited to hold three frames for each video streaming. After request R_1 play the streaming, frames 1, 2, 3 are cached in the proxy, and the proxy will forward those frames to the second client. However, request R_3 will just patch frame 1 from the proxy cache because frames 2, 3 are retrieved from the ongoing transmission channel C_2. It means that frames 2, 3 cached in the patching proxy are useless. So the prefix caching just saves four frames transmitted from the server, as shown in Figure 3. In the prefix caching mechanism, cache objects in the proxy will not be replaced. It means the proxy cannot dynamically determine which frames to be cached at appropriate time. Therefore, the prefix caching would not perform efficiently with the patching algorithm. The number of frames patched from the server can be lowered down when the proxy utilize the cache efficiently. Therefore, our objective is to schedule the proxy cache dynamically for improving the utilization of the proxy cache by using patching scheduling, as shown in Figure 4. Next section, we describe how the proxy dynamically cache the frames appropriately.

Figure 3. Prefix caching with GBR

3.3. Patching scheduling algorithm

The patching scheduling algorithm works with the patching server in the proxy. The patching scheduling algorithm will schedule the patching frames that based on GBR algorithm, thus the patching scheduling algorithm in the proxy is to reschedule CS_i for the request of client i. The proxy will cache the same frames in different CS_i and share these frames among clients. The patching proxy reschedules CS_i to a *new channel transmission schedule* (NCS_i) and a *proxy caching schedule* (PCS). The NCS_i is a vector that represents how the frames are transmitted on the channel i from the server. The PCS is the schedule that the proxy will cache frames at appropriate time slot. The RS_i of the client i is unchanged, because the proxy is transparent to the clients.

Figure 5 represents the data structure and the pseudo code for the patching scheduling algorithm which generates a PCS and NCS_i for client i. CS_i is the channel schedule which generated by GBR patching algorithm. The structure of the element f in PCS is a patching frame class. The vector PLT, as LT in GBR algorithm, keeps track of the latest time. And the vector PLC, as LT in GBR algorithm, maintains the channel on which a frame will be sent from the server.

First, we make a copy of CS_i to NCS_i. We consider the frame j on existing channels to be cached in the proxy before the frame j transmitted to client i. Frame j will be scheduled to be an element of PCS if frame j has already cached in the proxy or frame j can be stored into the proxy cache during the latest time of the frame j be sent from the server to the time before frame j be played. It updates cache occupancy to store frame j if frame j does not exist in the cache during $PLT[j]$, ..., $D_j - 1$. In this case, the frame j will not be transmitted from the server, so we let $NCS_i[j] = 0$. Otherwise, the frame j will be transmitted from the server, so we update the vector $PLT[j]$ to be the time of sending frame j and $PLC[j]$ to the channel of sending frame j. The frames are cached in the proxy according to the schedule generated by the patching schedule algorithm. Each frame has a counter in the cache. The counter is increased if the same frame is scheduled to cache in the proxy. The counter of the frame in the cache is decreased when a client access the frame in the proxy. The frame is removed from the cache when the counter is zero. Finally, we get two schedules that are PCS for the proxy and NCS for the server. The original CS will be placed in the proxy to indicate that the proxy forwards the frames from the server or forwards the frames from its cache.

```
// data structure
Class patchFrame
{
    j; (frame number)
    Dj; (playback time of frame)
    SendTime; (starting time of frame transmitting)
    Channel; (logical channel of frame receiving)
}
// algorithm
patching_scheduling(CS_i, t_i, PLT, PLC)
{
    CS_i:the channel transmission schedule of client i
    t_i:the arrival time of client i
    PLT:keeps track of the latest time of a frame
    PLC:maintains the channel information to transmit a frame

    NCS_i = CS_i
    for j=1, ..., N
    {
        if((CS_i[j]! = 0) AND (PLT[j] ≥ t_i) AND (frame j can store into
        the proxy cache OR frame j has existed in the proxy cache
        during PLT[j], ..., D_j − 1))
        {
            1. Add f to the Proxy Caching Schedule (PCS)
               f.j = j
               f.D_j = Dj
               f.SendTime = PLT[j]
               f.Channel = PLC[j]
               add f to PCS
            2. Update cache occupancy to store frame j if frame j does
               not exist in the cache during PLT[j], ..., D_j − 1
            3. NCS_i[j] = 0
        }
        else if(CS_i[j]! = 0)
        {
            1. PLT[j] = D_j − 1
            2. PLC[j] = C_i
        }
    }
}
```

Figure 5. Patching scheduling algorithm

Client 2		Client 3	
$CS_2[1]= 1$	$NCS_2[1]= 1$	$CS_3[1]= 1$	$NCS_3[1]= 1$
$CS_2[2]= 2$	$NCS_2[2]= 2$	$CS_3[2]= 0$	$NCS_3[2]= 0$
$CS_2[3]= 3$	$NCS_2[3]= 3$	$CS_3[3]= 0$	$NCS_3[3]= 0$
$CS_2[4]= 4$	$NCS_2[4]= 4$	$CS_3[4]= 0$	$NCS_3[4]= 0$
$CS_2[5]= 5$	$NCS_2[5]= 5$	$CS_3[5]= 0$	$NCS_3[5]= 0$
$CS_2[6]= 0$	$NCS_2[6]= 0$	$CS_3[6]= 6$	$NCS_3[6]= 6$
$CS_2[7]= 0$	$NCS_2[7]= 0$	$CS_3[7]= 0$	$NCS_3[7]= 0$
$CS_2[8]= 0$	$NCS_2[8]= 0$	$CS_3[8]= 0$	$NCS_3[8]= 0$
$CS_2[9]= 9$	$NCS_2[9]= 0$	$CS_3[9]= 0$	$NCS_3[9]= 0$
$CS_2[10]=10$	$NCS_2[10]=0$	$CS_3[10]= 0$	$NCS_3[10]= 0$
$CS_2[11]= 0$	$NCS_2[11]= 0$	$CS_3[11]= 0$	$NCS_3[11]= 0$
$CS_2[12]= 0$	$NCS_2[12]= 0$	$CS_3[12]= 12$	$NCS_3[12]= 0$
$CS_2[13]= 0$	$NCS_2[13]= 0$	$CS_3[13]= 0$	$NCS_3[13]= 0$
$CS_2[14]= 14$	$NCS_2[14]= 0$	$CS_3[14]= 0$	$NCS_3[14]= 0$
$CS_2[15]= 15$	$NCS_2[15]= 0$	$CS_3[15]= 0$	$NCS_3[15]= 0$
$CS_2[16]= 0$	$NCS_2[16]= 0$	$CS_3[16]= 0$	$NCS_3[16]= 0$
$CS_2[17]= 0$	$NCS_2[17]= 0$	$CS_3[17]= 0$	$NCS_3[17]= 0$
$CS_2[18]= 0$	$NCS_2[18]= 0$	$CS_3[18]= 18$	$NCS_3[18]= 0$
$CS_2[19]= 19$	$NCS_2[19]= 0$	$CS_3[19]= 0$	$NCS_3[19]= 0$
$CS_2[20]= 20$	$NCS_2[20]= 0$	$CS_3[20]= 0$	$NCS_3[20]= 0$

Figure 6. CS_i **v.s.** NCS_i

PCS			
j	Dj	SendTime	Channel
9	14	8	C_1
10	15	9	C_1
12	18	11	C_1
14	19	13	C_1
15	20	14	C_1
18	24	17	C_1
19	24	18	C_1
20	25	19	C_1

Figure 7. PCS **by patching scheduling**

3.4. Example

We take the same example described in section 3.2. The patching proxy reschedules the CS_i and then generates the NCS_i and a PCS, as shown in Figure 6 and Figure 7 respectively. According to the PCS, the proxy will cache the frames on the specified channel. We can find out that frames 9, 10, 12, 14, 15, 18, 19, 20 are scheduled to be cached in the proxy dynamically to be patched to clients. Figure 8 shows the patching proxy can reduce eight frames which patched from the server by our patching scheduling algorithm, although the proxy cache can hold just three frames. It means that patching scheduling can reduce the server workload and the network congestion efficiently in particular circumstance.

4 Experiments and Discussion

4.1 Simulation environment

The experiments consider a one-hour video stream. The playback rate is 25 frames per second. We generate 500 client requests and multiple clients arriving within the same time unit (*frame time*) are batched and served together. The inter-arrival time is assumed to follow a $Poisson$ distribution. Our basic comparison is performed on a proxy with prefix caching and a proxy using the patching scheduling algorithm. The metric of the measurement is the reduction ratio, and defined as following formula,

$$Ratio = \frac{Patch\ frames\ using\ the\ proxy\ cache}{Patch\ frames\ without\ using\ the\ proxy\ cache}.$$

Figure 8. Patching scheduling with cache

Figure 9. Effect of the proxy cache size

4.2 Effect of the proxy cache size

The utilization of the prefix-cache proxy is proportional to the proxy size. But it is not the same in the patching scheduling algorithm, as shown in Figure 9. We can observe that the reduction ratio of prefix caching is increased when more space of the proxy cache is available, thus the curve of prefix caching mechanism is monotonically increased. The reduction ratio of patching scheduling mechanism does not increase after the size of the proxy cache is larger than *3-minutes*. No matter how the size of the proxy cache is increased, the reduction ratio of the patching scheduling mechanism is saturated. As shown in Figure 8, frames 1, 2, 3, 4, 5 are missed, because the frames are not on the existing ongoing stream when the second request arrived. Therefore, no matter how large the capacity of the proxy cache is, it is unable to cache those frames under the patching scheduling algorithm.

4.3 Effect of request rate

Figure 10. Effect of request rate

The duration of arrived frames affects the proxy efficiency. The inter-arrival time between requests determines the time of a frame staying in the proxy cache because the frame is received from the previous channel and then is kept in the proxy cache until be transmitted to the client. The frame occupies the proxy cache buffer for a long time when the inter-arrival time increases between requests, which will

diminish the proxy efficiency. The arrival time of requests is close to each other when the video stream is so popular. The number of patched frames is reduced when the requests arrive frequently, because the frames can be shared by ongoing streams easily. In Figure 10, the reduction ratio is zero when the inter-arrival time is 10 seconds under the patching scheduling. The reason is that the prefix frames are not on the existing ongoing streams when requests arrived. The reason of the smaller reduction ratio when the inter-arrival time is large enough is that each frame is shared by ongoing stream would occupy the proxy cache for a long time. This reduces the utilization of the proxy cache, and then the frames on the existing ongoing streams are unable to cache in the proxy. Therefore, we can find out that why the curve of the patching scheduling likes a parabolic curve.

4.4 Overall comparison

We want to observe the effects under different client buffer size, different proxy buffer size and different inter-arrival time. We integrate all of the experiment results, as shown in Figure 11.

In these experiment results, we can observe that the reduction ratio of the prefix caching and the patching scheduling are extremely different due to the size of the client buffer under the fixed size of the proxy cache and the small inter-arrival time. In fact, the curve of the patching scheduling algorithm is like a parabolic curve. The effect of the curve of the patching scheduling is that the maximum point of the curve occurs in the small inter-arrival time when the size of the client buffer is reduced. Therefore, the maximum point of the curve of patching scheduling should occur before 10 seconds of inter-arrival time. The reduction ratio of the prefix caching and patching scheduling are closer when the size of the proxy cache is fixed and the inter-arrival time is increased to an extent, such as the inter-arrival time is 10 minutes.

5 Conclusion

In this paper, we study the issues in the multimedia streaming and propose a patching scheduling scheme. Our approach is to give a better patching schedule by using small cache space in the proxy, while it can still reduce the server overhead and the network congestion substantially. We propose a patching scheduling in the proxy to dynamically cache frames that will soon be requested. Therefore, the patching scheduling will virtually increase the capacity of the client buffer. The multimedia server and clients do not need to modify in the system architecture when the system with GBR algorithm is changed to the system with our patching scheduling algorithm. In experiment results, we can observe that the reduction ratio is better than that of the prefix caching or patching scheduling approach when

Figure 11. Results of different conditions

the inter-arrival time is small. It also means that the patching proxy can reduce large the server overhead and network congestion when a video stream is popular in the multimedia server.

References

[1] C. Aggarwal, J. Wolf, and P. Yu. On optimal batching policies for video-on-demand servers, 1996.

[2] E. Bommaiah, K. Guo, M. Hofmann, and S. Paul. Design and implementation of a caching system for streaming media over the internet, June 1999.

[3] Y. Cai, K. Hua, and K. Vu. Optimizing patching performance, January 1999.

[4] Tai T. Do, Du A. Tran, and Kien A. Hua. Layered range multicast for video on demand. In *IEEE International Conference on Computer Communications and Networking*, 2002.

[5] Tai T. Do, Du A. Tran, and Kien A. Hua. Zigzag: An efficient peer-to-peer scheme for media streaming. In *Proceedings of IEEE INFOCOM'03*, April 2003.

[6] Derek L. Eager, Michael C. Ferris, and Mary K. Vernon. Optimized caching in systems with heteroge-

neous client populations. In *ACM Performance Evaluation*, pages 163–185, 2000.

[7] Stephane Gruber, Jennifer Rexford, and Andrea Basso. Protocol considerations for a prefix-caching proxy for multimedia streams. *WWW9 / Computer Networks*, 33(1-6):657–668, 2000.

[8] M. Hofmann, E. Ng, K. Guo, S. Paul, and H. Zhang. Caching techniques for streaming multimedia over the internet, April 1999.

[9] K. Hua, Y. Cai, and S. Sheu. Patching: A multicast technique for true video-on-demand services. In *Proc. ACM Multimedia*, September 1998.

[10] K. W. Ross M. Reisslein, F. Hartanto. Interactive video streaming with proxy servers. In *Proceedings of First International Workshop on Intelligent Multimedia Computing and Networking(IMMCN)*, 2000.

[11] P. Frossard L. Amini O. Verscheure, C. Venkatramani. Joint server scheduling and proxy caching for video delivery. In *IBM Technical Report Num RC21981*, 2001.

[12] Jon Crowcroft Paul P. White. Optimized batch patching with classes of service. In *ACM Computer Communications*, 2000.

[13] M. Handley D. Estrin R. Rejaie, H. Yu. Multimedia proxy caching mechanism for quality adaptive streaming applications in the internet. In *Proceedings of IEEE INFOCOM*, March 2000.

[14] S. Sen, L. Gao, J. Rexford, and D. Towsley. Optimal patching schemes for efficient multimedia streaming, June 1999.

[15] Subhabrata Seny. Proxy prefix caching for multimedia streams, 1999.

[16] Simon Sheu, Kien A. Hua, and Wallapak Tavanapong. Chaining: A generalized batching technique for video-on-demand. In *International Conference on Multimedia Computing and Systems*, pages 110–117, 1997.

[17] Brian Smith Soam Acharya. Middleman: a video caching proxy server. In *Proceedings of NOSSDAV*, June 2000.

[18] M. A. Tantaoui, D. A. Tran, and Kien A. Hua. A multi-multicast sharing technique for large-scale video information systems. In *IEEE International Conference on Communications*, 2002.

[19] Roy Villafane, Kien A. Hua, and Duc A. Tran. Overlay multicast for video on demand on the internet. In *ACM Symposium on Applied Computing*, 2003.

Adaptive Packet Size Assignment for Scalable Video Transmission over Burst Error Channel*

Chu-Sing Yang, Chen-Wei Lee, Yih-Ching Su

Department of Computer Science and Engineering,
National Sun Yat-Sen University, Kaohsiung 804, Taiwan R.O.C

ABSTRACT

This article aims to find an adaptive packet size assignment scheme for scalable video transmission over burst error channel. We have developed an analytic model to evaluate how the channel bit-error-rate affects the video quality of streaming scalable video. A video transmission scheme which combines adaptive assignment of packet size and unequal error protection for raising the end-to-end video quality will be proposed in this article. Several distinct scalable video transmission schemes over burst-error channel have been compared, and the simulation results reveal that our transmission scheme is capable of reacting to the varying channel condition with less and smoother quality degradation.

Key words:
adaptive packet size, scalable video, burst error channel

1. INTRODUCTION

Bit errors and packet losses are commonplace over the wired/wireless Internet. They can severely affect the quality of delay sensitive multimedia applications. In the current Internet architecture it is up to the application to react to the perceived congestion level in the network. The ability of the application to react is enhanced by the availability of simple and efficient loss models. A number of studies have shown that loss patterns exhibit a finite dependence in time [1-2]. The most generalized model that represents quasi-stationary phenomena is a finite state Markov chain. Because of its simplicity and effectiveness, a two state Markov model or Gilbert-model [3-4] is often used to simulate burst loss patterns over wired/wireless channel [5-7].
Scalable video coders for Internet application have been proposed in the past by many authors [12-15]. Among them, Fine Granular Scalability (FGS) [12, 16] in MPEG-4 is one of the highly flexible coding techniques

capable of delivering layered video data with precise rate control. The characteristics of FGS are considered as pluses especially for error-prone heterogeneous transmission environments, e.g. mobile video-on-demand systems.

There is an inherent trade-off between the selection of packet size and the consequent packet-error-rate (PER) on a multi-access channel. For wireless links, the dominant error mechanism is bit error. The longer the packet size, the higher the probability of unrecoverable bit errors, and the higher the probability of packet loss. On the other hand, the smaller the packet size, the higher the header overhead in each packet. In the context of the streaming video, a lossy channel exacerbates the problem if the packet size is not adaptive. This problem is well studied in numerous works. In [9], they try to find the optimal bit allocation in FEC-based video, but they only consider Internet video without wireless video. In [8], a dynamic packet size mechanism in wireless networks was proposed, but the problems of burst-error channel and video quality adaptation have not addressed.

To cope with the sometimes rapidly varying transmission conditions over wired or wireless Internet connections we propose and investigate in this article a video transmission scheme which combines a scalable video coder with unequal error protection (UEP) and adaptive packet size assignment simultaneously. In order to reach the goal of adaptive packet size and UEP assignment, the analytic model in scalable video stream has been analyzed and compared in this work. Our approach does not require any support from the network. It can be employed in any packet-oriented network.

This paper is organized as follows: Section 2 introduces the burst-error channel model; several functions have been derived to calculate packet-error-rate from bit-error-rate. Section 3 describes how the hierarchically encoded video stream is packetized, protected by unequal error protection and transmitted to the receiver. The complete analytical model in scalable video stream is formulated. Furthermore, the adaptive packet size assignment algorithm for scalable video transmission over burst error channel is also explained in this section. The performance evaluation for distinct

scalable video transmission schemes over burst-error channel will be shown in Section 4. Finally, conclusions are given in Section 5.

2. CHANNEL MODEL

A two state Markov model or Gilbert-model is used to simulate burst loss patterns over wired/wireless channel. The two states of the model are denoted G and B (see Fig. 1). In state G, a bit is received correctly, while in state B, a bit is lost. This model is described by average bit error rate P_b and average burst bit error length L_b. The transition probability P_{gb} and P_{bg} can be easily computed using P_b and L_b:

$$P_{bg} = \frac{1}{L_b} \tag{1}$$

$$P_{gb} = P_{bg} \frac{P_b}{1 - P_b} \tag{2}$$

Fig. 1 Gilbert channel model.

To get packet loss patterns, we must have packet error rate P_B (PER) and average burst packet loss length L_B in advance. The calculations on P_B and L_B are as the following:

$$P_B = 1 - ((1 - P_b) \cdot (1 - P_{gb})^{\tau-1}) \tag{3}$$

$$P_{GB} = 1 - (1 - P_{gb})^{\tau} \tag{4}$$

$$P_{BG} = P_{GB} \cdot \frac{(1 - P_B)}{P_B} \tag{5}$$

$$L_B = \frac{1}{P_{BG}} \tag{6}$$

τ : packet length in bits

Fig. 2 shows the relations between P_B (PER), P_b (BER), and L_b. In Fig. 2(a), under the condition of $P_b = 10^{-3}$, P_B will decrease as L_b increases, and a shorter packet length results in a smaller P_B. It has also been observed from Fig. 2(b) that P_B will enlarge

following the increment of P_b for a specific burst status (e.g. $L_b = 70$).

(a)

(b)

Fig. 2 The relations between P_B (PER), P_b (BER) and L_b.

3. ADAPTIVE PACKET SIZE ASSIGNMENT

As illustrated in Fig. 3, bit streams of all layers are interleaved into one block of packets (BOP) and the transmission packets are the rows of the BOP. Note that the source data with length r_i belonging to layer i ($i = 0 \sim l$, layer 0 is the base layer) are filled into the arranged portions (column width $= s_i$) of k_i packets, and

the remaining portions of $n - k_i$ packets (n = number of packets) in the BOP are filled with channel coding redundancy. Therefore, k_i determines the protection level of layer i. To satisfy the buffer and delay constraints for real-time communication, we assume that the BOP buffer size r is fixed. Given the number of packets n, then the packet size $s = r/n$ in Fig. 3 is known too.

Fig. 3 The data structure of BOP.

Packet loss over Gilbert-channel can be modeled with a renewal error process [4]. In other words, the lengths of consecutive inter-error intervals (also called gaps) are independently and identically distributed. Following the development of [4], let $g(i)$ denote the probability that a gap length is $i-1$, i.e., $g(i) = \Pr(0^{i-1}1 | 1)$, where '1' denotes a lost packet and 0^{i-1} denotes $i-1$ consecutive successfully received packets. Similarly, let $G(i)$ denote the probability that a gap length is greater than $i-1$, i.e., $G(i) = \Pr(0^{i-1} | 1)$. Then the probability $R(m,n)$ that $m-1$ packet losses occur in the next $n-1$ packets following an error can be computed by recurrence [4]. Thus,

$$R(m,n) = \begin{cases} G(n), & m = 1, \\ \sum_{i=1}^{n-m+1} g(i)R(m-1,n-i), & 2 \le m \le n. \end{cases} \quad (7)$$

Finally the probability of m lost packets within a block of n packets or $P(m,n)$ is [4]

$$P(m,n) = \begin{cases} \sum_{i=1}^{n-m+1} P_B G(i)R(m,n-i+1), & 1 \le m \le n, \\ 1 - \sum_{m=1}^{n} P(m,n), & m = 0. \end{cases} \quad (8)$$

P_B in (8) means the average packet loss rate of a Gilbert-model.

Combing the BOP data structure with the Gilbert-model renewal error process, the complete analytical

model in scalable video stream from our earlier works [10-11] can be formulated as follows:

$$\eta = $$

$$\sum_{i=0}^{l+1} \sum_{m=n-k_i+1}^{n-k_{i-1}} P(m,n) \left(\underbrace{\sum_{j=0}^{i-1} \chi_j}_{\substack{\text{perfectly} \\ \text{reconstructed} \\ \text{quality}}} + \underbrace{\sum_{j=1}^{n-m+1} \frac{\rho_j R(m, n-j+1)}{P(m,n)} \phi_{ij}}_{\substack{\text{partially} \\ \text{reconstructed} \\ \text{quality}}} \right)$$

$$, \ k_{-1} = 0, \ k_{l+1} = n+1 \qquad (7)$$

where

➤ : means the expected received quality for a BOP .

➤ χ_i : denotes the incremented quality that layer i is received correctly.

➤ ϕ_{ij} : represents the residual quality that the first $j-1$ packets of layer i are received correctly.

➤ ρ_i : indicates the probability that the first error occurs in packet i .

We can define the adaptive packet size assignment problem as to find some n (number of packets) that maximizes the received video quality as follows:

$$\arg \ \max_{n} \{\eta(n)\} \qquad (8)$$

After the searching for n, the adaptive packet size can therefore be found since $s = r/n$, r is BOP buffer size. Here each adaptive packet size was derived and applied for per BOP unit.

Take the adaptive UEP into account, the previous optimization problem can be rewritten as:

$$\arg \ \max_{n,K} \{\eta(n, K)\} \qquad (9)$$

subject to $s = \sum_{i=0}^{l} s_i = \sum_{i=0}^{l} \frac{r_i}{k_i}$ and

$$0 \le k_0 \le k_1 \le \dots \le k_l \le n \qquad (10)$$

where $\mathrm{K} = (k_0, k_1, k_2, \dots, k_l)^t$.

The common test conditions for all of the video streams being used in this paper are as follows:
 (i) frame resolution = CIF format (352x288)
 (ii) constant stream rate = 256 Kbps
 (iii) 1 GOP = 1 intra frame accompanied with 14 inter frames and frame rate = 15 fps
 (iv) sequence length = 9 GOPs
 (v) MPEG-4 FGS method has been adopted to generate the scalable video streams. Each MPEG-4 FGS video stream is composed of one base layer and one enhancement layer.
 (vi) Packet header length: 40 bytes (IP/UDP/RTP).

Fig. 4 compares the analytical and the simulated PSNR values resulting from the adaptive packet size

assignment scheme (8) toward different bit error rate P_b.

In this example, the test video sequence is "Silent", and the burst bit error lengths L_b for those three curves are 20, 40 and 80 respectively. As shown in Fig. 4, the simulation results are very near to the analytical results, which validates the correctness of (8).

Fig. 4 Comparison of the analytical and the simulated PSNR values resulting from the adaptive packet size assignment scheme.

4. SIMULATION RESULTS

In this section the performance evaluation for distinct scalable video transmission schemes over burst-error channel will be demonstrated. At first, fixed UEP (FUEP) is being considered. We compare the performances of several FPS (Fixed Packet Size) schemes and our APS (Adaptive Packet Size) scheme under various channel conditions.

Fig. 5 shows the performance comparison of our APS scheme with some other FPS schemes, where the test sequence is "News" and the channel related parameters are $L_b = 80$ and $P_b = 0 \sim 10^{-2}$. Adaptive UEP scheme has not been taken into consideration here. For the sake of convenience, we take fixed 200 bytes labeled as FPS200 and so on. More specifically, we classify the fixed protection ratios into two levels: low protection ratio (Fig. 5a) or high protection ratio (Fig. 5b). The percentages within the brackets in Fig. 5 legends denote the protection ratios for the base layer (left percentages) and the enhancement layer (right percentages) individually.

From Fig. 5 it is observed that only our adaptive packet size scheme can effectively adapt to the varying channel condition and react to the worse condition with smoother quality degradation. In contrast, no matter how

long the packet size has been decided, the other three non-adaptive schemes will fall in either too large packet size with high P_b for bad channel condition or too small packet size with high header overhead for good channel condition, and the outcome is severe quality degradation.

(a) low FUEP (25%,10%)

(b) high FUEP (75%,50%)

Fig. 5 The performance comparison of our adaptive packet size scheme and three other fixed packet size schemes under (a) low protection ratios or (b) high protection ratios.

Fig. 6 displays performance evaluation for APS and three FPS schemes over burst-error channel with adaptive UEP (AUEP) scheme. The test sequence is "Container"

and the channel related parameters are $L_b = 70$ and $P_b = 10^{-5} \sim 10^{-2}$. It has also been observed from Fig. 6 that APS can effectively adapt to the varying channel condition and react to the worse condition with smoother quality degradation.

Fig. 6 The performance comparison of APS and three FPS schemes with AUEP.

5. CONCLUSIONS

In order to raise the end-to-end video quality over burst error channel, we develop an analytic model of how the channel bit-error-rate affects the packet-error-rate which in turn affects the video quality of streaming video. An adaptive packet size assignment scheme for scalable video transmission over burst error channel was proposed in this article. Given a total available bandwidth, adaptive assignment of packet size and UEP is achieved simultaneously by maximizing the video quality. Simulation results reveal that our method can effectively adapt to the varying channel and react to the worse channel condition with less and smoother quality degradation. Our approach does not require any support from the network, it can therefore be employed in any packet-oriented network.

6. REFERENCES

[1] M. Yajnik, S. Moon, J. Kursoe, and D. Towsley, "Measurement and modelling of the temporal dependence in packet loss," *Proc. IEEE INFOCOM'99*, New York, NY, pp.345-352, March 1999.

[2] R. Singh and A. Ortega, "Modeling of temporal dependence in packet loss using universal modeling concepts," *Proc. 12th Packet Video Workshop*, Pittsburgh, PA, Apr. 2002.

[3] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, vol.39, pp.1253-1265, Sept. 1960.

[4] E. O. Elliott, "A model of the switched telephone network for data communications," *Bell Syst. Tech. J.*, vol. 44, no. 1, pp. 89–109, Jan. 1965.

[5] C. C. Tan, N. C. Beaulieu, "On First-Order Markov Modeling for the Rayleigh Fading Channel," *IEEE Trans. on Communication*, vol. 48, no. 12, pp. 2032-2040, Dec. 2000.

[6] C. Hsu and A. Ortega, "A Lagrangian Optimization Approach to Rate Control for Delay-Constrained Video Transmission over Burst-Error Channels," *ICASSP*, 1998.

[7] H. S. Wang and N. Moayeri, "Finite-state Markov channel: A useful model for radio communication channel," *IEEE Trans. on Veh. Technol.*, vol. 44, pp. 163-171, Feb. 1995.

[8] T. Vu, D. Reschke and W. Horn, "Dynamic Packet Size Mechanism (DPSM) for Multimedia in Wireless networks", MIK Workshop, 2002.

[9] B. Hong and A. Nostratinia, "Rate-constrained scalable video transmission over the internet," *Proc. 12th Packet Video Workshop*, Pittsburgh, PA, Apr. 2002.

[10] Y. C. Su, C. S. Yang and C. W. Lee, "The analysis of packet loss prediction for Gilbert-model with loss rate uplink," Information Processing Letters, vol. 90/3, pp. 155-159, May 2004.

[11] Y. C. Su, C. S. Yang and C. W. Lee, "Optimal FEC assignment for scalable video transmission over burst error channel with loss rate feedback," Signal Processing: Image Communication, vol. 18, pp.537-547, Aug. 2003.

[12] W. Li, "Overview of Fine Granularity Scalability in MPEG-4 Video Standard," *IEEE Trans. on Circuit and System for Video Technology*, vol. 11, no. 3, pp. 301-317, Mar. 2001.

[13] F. Wu, S. Li, and Y. Chang, "A framework for Efficient Progressive Fine Granularity Scalable Video Coding," *IEEE Trans. on Circuit and System for Video Technology*, vol. 11, no. 3, pp. 332-344, Mar. 2001.

[14] D. Taubman and A. Zakhor, "Multirate 3D Subband Coding of Video," *IEEE Trans. on Image Processing*, vol. 3, no. 5, pp. 572-588, Sept. 1994.

[15] B. Kim, Z. Xiong, and W. Pearlman, "Low bit-rate scalable video coding with 3D set partitioning in hierarchical trees (3D SPIHT)," *IEEE Trans. on Circuit and Systems for Video Technology*, vol. 10, pp. 1374-1387, Dec. 2000.

[16] ISO/IEC 14496-2:2001/Amd 2:2002 (Streaming video profile)

A Study on a New Queue Management and Scheduling Technique for a Remote Technical Consultation System

Rafael A. Sierra and Fumitaka Uchio
Graduate School of Systems Engineering, Wakayama University, Japan.
E-mail: s030037@sys.wakayama-u.ac.jp, uchio@wakayama-u.ac.jp

Abstract—In this paper we present an active queue management and packet scheduling technique to be used in the transmission of packets by real-time Internet video applications such as the Remote Technical Consultation System – RTCS.

The RTCS is a system designed to perform real time video consultation to experts over the Internet. In order to minimize the consequences of packet loss along the network, it divides the video image into a Region of Interest, the most important one, and a Region of No Interest, assigns high priority to the former and low priority to the later. According to that priority, under congestion, packets with lower priority are discarded first, ensuring that packets arriving at the receiver can reproduce, at least, the Region of Interest, making the images for consultation always useful.

In this work, we improve the RTCS, with a new technique for queuing management and packet forwarding, based on the calculation and combination of values of remaining time and distance, in order to correct at routers unexpected delays that packets may suffer along the communication path. We design, implement, and simulate the algorithms and also emulate the Sender and Receiver parts of the application. Using a simulator we evaluate and confirm the validity of our proposal.

Keywords—*Real Time Video, Queuing, Scheduling.*

I. Introduction

In this paper we propose a new queuing discipline introducing an active queue management and packet scheduling technique to be used by real-time Internet video applications and in particular by the Remote Technical Consultation System [3] or RTCS.

Thus, in this work, we introduce improvements mainly for the Network component of the RTCS, having as objectives keeping high reliability in the transmission of preferential packets, controlling packet discarding, putting a bound on the maximum transmission delay and reducing jitter among packet flows. Since the RTCS concentrates in transmission of video images, in our proposal we simplify the design of the queuing discipline and concentrate primarily in the transmission of video packets.

Because this discipline can face scalability problems, we must reduce the complexity of the implementation to the minimum possible. In order to do so, we have devised some procedures that allow scheduling packets and managing the buffer queue, avoiding complications in traffic handling and keeping a low processing burden.

More specifically, we research on a technique that can be used inside the Differentiated Services Assured Forwarding PHB [9] for keeping low complexity, performing per-class classification, and, that at the same time, improves the service guarantees it can provide to invoking applications.

Some recently proposed techniques include systems that are neither within the Differentiated Services nor the Integrated Services frameworks. These are complete systems that make use of some of the concepts of DiffServ and IntServ or combine the two frameworks in order provide harder guarantees while reducing complexity. These systems are the Desart system [12], the SCORE system [13], and the Rainbow Fair Queuing system [1].

In the next section of this paper we give an overview of the Remote Technical Consultation System and explain the problems arisen in the first implementation. In the third section we state our research objectives and describe our proposal. We describe the experiments done for the evaluation and present and discuss the results in the fourth section. Finally, we describe our future work and draw conclusions.

II. The Remote Technical Consultation System

A. Description

According to the original proposal in [3], the RTCS performs technical consultation between senders and receivers, using video communication on networks. Senders take video images of objects for consultation and transmit them; Receivers, as experts, diagnose problems and advise senders.

In the RTCS, video images are divided, as depicted in Figure 1, into a Region of Interest -ROI- and a Region of No Interest -RONI-, layered and then encoded. The packets generated are prioritized, with packets belonging to the ROI and lower layers receiving higher priority. Routers forward those packets reflecting the requested priority and transient conditions of the network. If congestion, and packet loss occur at any link of the network path, due to their reduced priority, packets belonging to the RONI may be discarded first, always before discarding RONI packets, so ideally the packets that arrive at the receiver will be enough to reproduce the ROI, making the consultation still useful.

In the first implementation [8] of this proposal, packet forwarding in routers was done following fixed port numbers for every pair Layer-Region. Each one of these pairs of values identifies one flow. This implementation

Fig. 1. The Remote Technical Consultation System

was done using the ALTQ Queuing Framework [11] running on top of the FreeBSD Operating System. The scheduler chosen was WFQ. This implementation achieved suitable Quality of Service for using the RTCS on the Internet with reduced bandwidth. However, some problems were seen as we explain next.

B. Problems

1) Loss of synchrony between ROI and RONI packets: As the transmissions rates of every Layer-Region packet flow are different, every packet flow experiences a different delay. The consequence of this delay variation, or jitter, is the loss of synchrony among the elements of the same image, i.e. images in the ROI and images in the RONI will not be corresponding if played back at the same time.

New experiments made using the DiffServ framework showed a slight increment in the quality of service provided to the application. This is achieved by, according to the DiffServ model, discarding more aggressively packets belonging to the RONI. This allows providing a little more bandwidth to the ROI and other competing flows.

2) Wasted bandwidth: When using WFQ scheduling, due to the reduced bandwidth for one of the flows, e.g. RONI, it is likely that the packets will suffer an extended delay. Applications such as video communication have stringent latency requirements, and if a packet arrives too late, is not useful and it is ignored. Late arriving packets result in wasted bandwidth and a reduction in the quality of service to the application [2].

III. Proposed Technique

In this work, we introduce improvements for the Remote Technical Consultation System with a new technique for doing queue management and packet forwarding. The usefulness of this technique is oriented towards incrementing the Quality of Service provided to (1) the transmission of real time multimedia traffic in general and (2) the Remote Technical Consultation System in particular. During the research process, we have designed simulated and evaluated this technique, which is based on the calculation and combination of values of remaining time and distance in order to correct in routers, unexpected delays that packets may suffer along the communication

path. This proposal is based on the one described on [6]. The main objectives for this technique are: (1) Keep high reliability in the transmission of ROI packets. (2) Control packet discarding. (3) Put a bound on the maximum transmission delay. And (4) Reduce jitter among packet flows - ROI/RONI - to cope with the synchronization problem.

The RTCS consists of three components: Sender, Receiver and the Relay-Forward Network Mechanism that connects them. Sender and Receiver, upon establishing communication, also coordinate and set values for transmission, values such as the maximum delay, the maximum number of hops expected in the communication path as well as the sending rate. These and other values are inserted in every packet and then the packets are sent.

A. Sender Side

In addition to the already described functions, in the proposed technique every packet is given a priority value according to the Layer and Region it belongs to. We call this value Discard Priority. Upon transmission, a packet is also marked with a Time Limit to arrive at the destination and the expected Number of Hops until arrival as a measure of distance. Other values such as Current Time are also added. Time Limit and Number of Hops are combined to form a Relay Priority. The details of this calculation are described later.

B. Relay Mechanism

When a packet enters every router of the communication path, first, information the packet is carrying along is read. Then, according to its Discard Priority the packet is classified into a class. This class corresponds with the region of the image the packet belongs to, i.e. ROI/RONI.

After classifying the packet, its new Current Time is calculated and used to update its remaining time. The new remaining time or new Time Limit corresponds to the previous Time Limit minus the difference between the previous Current Time and the new Current Time. In other words, this is the previous Time Limit minus the time it took the packet leave the previous router and arrive to the current one. The number of Remaining Hops until arrival is also updated though automatically because it corresponds with the TTL.

The Relay Priority is recalculated based on the new Time Limit and the new Number of Hops, and used to schedule the packet with a preferential position inside the output queue. Finally the packet is forwarded giving preference to the class that belongs to the ROI. Next we describe in detail these procedures.

1) Relay Priority Calculation

Time Limit and Number of Hops are combined to form a relay priority. When a packet is going to be sent, a Relay Index –RI– is calculated, where *RI = Time Limit / Number of Hops*. This value expresses the ratio time-distance or in other words the expected delay per hop inside the network. Thus, low Number of Hops and high Time Limit will raise

Fig. 2. Relay Mechanism

the value of the index; higher values help raising the priority valueof the packet. This RI value is also inserted into the packet and is invariable.

As described in figure 2, when the packet arrives at the router, a new RI value is calculated and compared against the value the packet has previously stored in. If the new value is greater than or equal to the previous value, it means that the packet is experiencing only the expected delay, so there is no need for changing anything. Thus the new Relay Priority will be made equal to the remaining time (Time Limit). If the new value is lesser than the previous one, it means that the packet is experimenting a more than expected delay. Then the new Relay Priority will be made equal to the result of the recalculation the Time Limit using the previous index and the number of Remaining Hops: *Relay Priority = RI x Number of Hops.*

2) Scheduling

Using the value calculated for the Relay Priority, the packet is inserted into the queue of its respective class. The queue we are using is a Calendar Queue, described in [2]. The Relay Priority is used to find each packet a preferential position inside the queue, the lower the value, the sooner the packet may be served. However, this is not the only consideration.

Provisions are taken to penalize the Relay Priority according to the time previous arrived packets spent waiting for service inside the queue. As depicted in Figure 3, each queue has an increasing time offset that represents the time elapsed since the first packet arrived to the queue. We add this Time Offset to the Relay Priority and get a Departure Relay Priority and use it to find the final position of the packet inside the queue. In this way, every new packet may be delayed an extra instant according to the time other packets may have been waiting for service. From another viewpoint, with this procedure we are updating / increasing the priority of every packet, while they wait for service.

3) Packet Forwarding and Discarding

Under normal network conditions, every class is given fair service; we forward packets from the queues of both classes. Packets exceeding their Time Limit, i.e. their Time Limits become zero or less, are always discarded; resending them is unnecessary because, being real time traffic, the receiver will no longer be waiting for them. In this case, discarding is done not from the output queue, but when the packet arrives at the router and its values are read.

In the event of congestion, packets from classes with low discard priority, i.e. RONI, are discarded first. In this way we guarantee that the most important packets will keep high reliability. When discarding packets, we do it starting from the head of the queue. For delay-sensitive but loss-tolerant traffic like video transmission, retaining packets that have already waited in the queue for a long time is meaningless. Also the probability of those packets arriving on time decreases. Unlike TCP traffic packets, the packets that follow in the queue will not be sent again despite that previous packets may have failed to arrive so this procedure does not increment the amount of traffic.

The main reason to discard packets from the head of the queue is to notify connections early that there is congestion in the network and allow them to initiate recovery promptly, reducing its transmission rate.

C. Receiver Side

The receiver monitors the number of correctly received packets and the number of discarded packets. Also it keeps monitoring variations in the number of hops traversed in the network by the packets. According to these results, it sends feedback to the sender so it can regulate its transmission rate, adjusting it to the available bandwidth. The sender can also increase or decrease the number of expected hops. The receiver must also reserve a buffer big enough so it can contain all the packets that arrive inside the time limit chosen at the time of connection establishment.

IV. Evaluation

A. System Behavior

We first evaluate the behavior of the Sender and Receiver application that we have constructed and simulated. The purpose of the first experiment is to show the response of the Sender and Receiver application to

Fig. 3. Scheduling of Packets, considering the addition of a Time Offset as a Penalty Function

severe congestion conditions, and also highlight the benefits of preferential forwarding and discarding by comparing it with a standard implementation of the Differentiated Services model.

In this experiment, we set up and try sending, by force, one RTCS flow of 1Mbps over a simple link with only 800kbps of available capacity, with 500kbps for the ROI and 500kbps for the RONI, distributed like Flow A in figure 4. We also setup the same distribution for a DiffServ Core router instead of the RTCS for comparison purposes, with drop preferences good enough to avoid loses of ROI packets. This simulation scenario is run for ten seconds.

As we can see in figure 5, there is a fluctuation of the transmission rate, adapting itself to the available bandwidth. The transmission rate throttles up after a number of successful delivered packets and throttles down after detecting loses. In a real implementation of the application this may well represent the initial stage of connection establishment.

The ROI flow receives more bandwidth only when RONI packets are being lost. After that period as the transmission rate is corrected, both flows recover its intended same rate. In figure 6, we can see that the transmission rate of the flows within the DiffServ setup is similar to the rate within the RTCS setup, although much more variable.

In this experiment we also observe the behavior of the priority dropping mechanism. Packets are being discarded when the buffer fills up. However, the packets being dropped belong only to the RONI, protecting this way the ROI packets that are important for consultation. The same holds true for the DiffServ Setup, thanks to its standard discarding priority mechanisms.

Although both RTCS and DiffServ setups show almost the same behavior, when analyzing the numbers of packet generated and successfully transmitted we see the difference.

In figure 7 we see that the results are quite similar so both configurations work protecting ROI packets. It is necessary to point out that the configuration of the DiffServ discarding options, are tweaked to provide zero loss of ROI packets.

In order to obtain good image for consultation, both regions ROI and RONI need to be present so, in the long run, the results obtained with the RTCS setup would be preferable over the results obtained with the DiffServ setup due to the presence of more RONI packets.

B. Time Limit and Hop Count Effects

Now we observe the effects of different Time Limit and different Hop Count on the packets transmitted at a link. For this experiment, we set up a 2Mbps link and send two 1Mbps RTCS flows A and B, and run the simulation for 10 seconds. These two flows differ once in Time Limit and once in Hop Count. This time we take only the delays experienced by ROI packets.

Fig. 4. Configuration of Simulation Environment

Fig. 5. Throughput fluctuation due to bandwidth sensing and loss detection for the Sender and Receiver Application with a RTCS router

Fig. 6. Throughput fluctuation due to bandwidth sensing and loss detection for the Sender and Receiver Application with a DiffServ router.

Pkts.	RTCS Setup			DiffServ Setup		
	Gen.	Rec.	Dis.	Gen.	Rec.	Dis.
ROI	464	464	-	465	465	-
RONI	464	413	51	465	408	57
Total	928	877	51	930	873	57

Fig. 7. Packets Generated, Received and Discarded with both setups

First, when both flows A and B have the same values the characteristics of the expected delay are as seen in figure 8. The delay bound is set to 200ms for both flows

As in figure 9, having the same hop count, flow A has a lower time limit than flow B. we observe that the delay experienced by Flow A is lower than the Flow B. The delay bounds are set to 150ms for Flow A and 200ms for Flow B.

The same holds true for the same Time Limit but different Hop Count. Flow A has a higher Hop Count than

flow B so as seen in figure 10, packets from flow A spend less time waiting for service.

We see here that our proposal is working as designed, giving preference to packets with tight time limits and longer remaining distances. We do not perform comparison against other disciplines because no one has functions for handling preferentially ROI and RONI packets.

C. ROI/RONI synchrony

As for the loss of synchrony between ROI and RONI flows, we observe the delay experienced by every packet of both ROI and RONI elements of the same flow in conditions of mild congestion, although buffer routers are completely filled caused by a third contending CBR UDP flow, as in figure 4.

This simulation also runs for ten seconds and its results, shown in figure 11, indicate that under mild congestion, the delay that both flows experience is almost the same.

We compare this with the delay experienced by ROI and RONI flows within the DiffServ setup. The delays for both of the conditions, mild congestion and full congestion with packet drop, are almost the same, as seen in figure 12.

We can see that within the DiffServ setup, the delay experienced by ROI packets is steady, however, the delay experienced by RONI packets varies greatly, more over, the difference of delay between these two flows is clearly broad. Moreover, the amount of packet transmission, shown in figure 13, makes the point more clear.

We can understand that by using our proposal we can provide better packet transmission while at the same time we can keep the jitter experienced to a low level.

Considerations:

DiffServ: Our proposal is situated within the DiffServ framework Assured Forwarding PHB [9]. The RTCS traffic constitutes one of the classes and the ROI, RONI

Fig. 8. Two RTCS flows with same values for Time and Distance

Fig. 9. Two RTCS flows with A having lower Time Limit

Fig. 10 Two RTCS flows with flow A having higher Hop Count

Fig. 11. Delay of ROI and RONI packets for a RTCS flow competing for the link using the same time limit, with and without packet loss

Fig. 12. Delay of ROI and RONI packets for a RTCS flow competing for the link, same time limit without packet loss, within a DiffServ setup

Pkts.	RTCS Setup			DiffServ Setup		
	Gen.	Rec.	Dis.	Gen.	Rec.	Dis.
ROI	1087	1087	-	1036	1036	-
RONI	1087	1087	-	1036	1036	-
Total	2174	2174	-	2072	2072	-

Fig. 13. Packets Generated, Received and Discarded by both setups under congestion

match two of the discarding priorities. This queuing model constitutes a subset of a complete queuing system inside a router; therefore we don't provide procedures like traffic admission control. Routers time synchronization: We assume that routers performing time operations implement some methods for keeping themselves aware of their time differences.

Number of Hops: Routes followed by packets, are rather stable. "Routing alteration is more an exception than a rule" [4], [10]. This allows us to set a very tight value for the TTL and monitor its adequacy using signaling in the RTCS application. Using a close to zero value for the hop count will help improving the Relay Priority, however, using a very low value has as a disadvantage the increasing of the possibility of packets being discarded while traversing the network because its TTL became zero.

Time: Real time applications require a maximum limit for the experienced delay during transmission. Typically the end-to-end delay tolerated for audio transmission, according to ITU are 150ms for an ideal unnoticeable delay, and fewer than 400 ms for a low, yet tolerable, quality communication. Over 400 ms, the communication is almost useless [7]. We assume here that the delay for video is the same, as we plan to also integrate audio transmission in our application.

Jitter: The transmission delay is bounded, so as long as the application can hold enough packets in its buffer, no problems with jitter will be perceived. The jitter we want to suppress is the variation of delay among the different flows of traffic in the RTCS, i.e. ROI/ RONI, although the simulation results showed a smooth variation under mild congestion.

V. Conclusion

We have proposed a new technique for active queue management and scheduling at routers, preferentially forwarding and discarding packets and bounding the transmission delay for real time video. This technique will allow us to perform an efficient Remote Technical Consultation on the Internet.

We evaluated the behavior of the system under contingency with other UDP flows and contingency with other RTCS flows with different time bounds. By using priority dropping, under normal conditions ROI and RONI obtain equal throughput, and under contention with other flows, RONI is being dropped until no more BW is available for ROI transmission. When this case occurs, the application throttles down its transmission rate avoiding packet losses.

As for delay and jitter control, we showed that the delay experienced by both ROI and RONI tends to be the same, making the interflow jitter minimum. The experiments also showed that the interflow delay or inter-arrival delay is represented by a smooth curve making the interflow jitter also minimum.

We could not do extensive comparison with any other system because there is no real similarity with existent proposals for doing the kind of queue management and scheduling we propose.

VI. Future Work

The current simulation assumes fair queuing using packets of the same size and performing simple round robin to serve packets. It is necessary to extend the algorithm so we can use it with more flows of mixed characteristics, i.e. many different sizes. We also plan to implement the algorithms of the proposed technique. For this task, as in the first implementation done at our laboratory, we intend to use the FreeBSD Operating System, with the AltQ queuing framework providing the required advanced router functions.

References

[1] Z. Cao, Z.Wang, and E. Zegura. "Rainbow fair queueing: fair bandwidth sharing without per-flow state", In *Proceedings of IEEE INFOCOM 2000*, volume 2, pp. 922–931. Tel-Aviv, Israel, April 2000..

[2] U. Black, *QoS in wide Area Networks,* New Jersey: Prentice Hall, 2000, ch. 1.

[3] N. Iguchi, F. Uchio and T. Takahashi, "A Video Communication Method using Layered Coding and Packet Priority for Remote Technical Consultation on the Internet", *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, 2001.

[4] V. Paxson, "End- to-End Routing Behavior on the Internet", *IEEE/ACM Transactions in Communications*, 1997.

[5] R. Brown, "Calendar Queues: A Fast O(1) Priority Queue Implementation for the Simulation Event Set Problem", *Communications of the ACM*, October 1988.

[6] R. Sierra, N. Iguchi, F. Uchio. "A Time Limit and Hop Based Queue Management and Scheduling Technique for a Remote Technical Consultation System on the Internet", *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, PACRIM 2003*, pp. 931-934.

[7] ITU-T. List of itu-t recommendations, 2001. http://www.itu.int/publications/itu-t/itutrec.htm.

[8] M. Uenoyama, K. Nara, N. Iguchi and F. Uchio, "A priority Control Method for the Remote Technical Consultation System". *Information Processing Society of Japan*, *SIG Note,* October 2002. (In Japanese)

[9] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski , Assured Forwading PHB Group, *RFC 2597*

[10] F. Begtasevic and P. Van Mieghem, "Measurements of the Hopcount in Internet", *Proceedings of Passive and Active Measurement* (PAM2001).

[11] K. Cho, *The Design and Implementation of the ALTQ Traffic Management System*. Doctoral Dissertation, Keio University, 2001

[12] Tsujioka et al. "A Study in Destination-Due-Date Priority Queuing for providing End-to-end Delay Guarantees" *Technical Report of the IEICE Sep. 2002.*

[13] I. Stoica and H. Zhang. "Providing guaranteed services without per-flow management". In *Proceedings of ACM SIGCOMM '99*, pages 81–94, Boston, MA, August 1999.

CHIPS: An End-System Multicast Framework for P2P Media Streaming

Son Vuong, Xin Liu, Ankur Upadhyaya and Jun Wang
Department of Computer Science
University of British Columbia, Vancouver, BC Canada
{vuong, liu, ankur, jwang}@cs.ubc.ca

Abstract

While avoiding many of the conventional difficulties with IP multicast, existing peer-to-peer (P2P) end-system multicast frameworks have their own unique limitations. For instance, such frameworks typically only enable broadcast communication across an entire peer group, rather than allowing finer-grain multicast transmission to select subgroups. Of even greater concern, is that these frameworks do not provide explicit control over the underlying distribution tree topology, thereby precluding the adoption of arbitrary, application specific data distribution policies that enable rich quality of service (QoS) features or the enforcement of other constraints. In this work, Channel Infrastructure for P2P Systems (CHIPS), an end-system multicast framework addressing the concerns above, is introduced. While the proposed framework can be leveraged by a broad range of applications, its particular utility in the distributed media domain is demonstrated with the implementation of a CHIPS-based P2P media streaming system.

1. Introduction

By now, arguments challenging the traditional view that multicast functionality is best handled at the network layer are well established in the literature. This has led to a substantial body of research into end-system approaches to multicast (e.g. Narada [1], ALMI [2], ALMAP [3]), many of which emphasize communication within dynamic peer groups. While such P2P end-system multicast frameworks enable various applications, they nevertheless remain limited in at least two fundamental ways.

First, these end-system multicast frameworks usually support only broadcast communication within peer groups, rather than allowing multicast transmission to select sub-groups of peers. In enabling larger peer-groups and more complex patterns of communication within those groups, this finer level of granularity is desirable for many applications. Unfortunately, no existing P2P end-system multicast framework allows this.

Second, in typical use, these end-system multicast frameworks hide the underlying multicast graph topology and its management from higher layers of software. This is detrimental from the standpoint of flexibility. In particular, by insulating an application from knowledge of and control over the underlying distribution graph topology, it is not possible to engineer traffic in an arbitrary, application specific manner in order to implement QoS features or other constraints appropriate to that application. For example, an application may want to restrict itself to multicast trees that route data only through peers belonging to a certain set (e.g. those peers that have paid for that data). Alternatively, it may want to ensure QoS for certain prioritized peers by only constructing multicast trees that route data to those peers through underutilized high-bandwidth links. In both cases, implementation of these application specific features is made possible with higher-level control of the underlying distribution tree. Nevertheless, no existing framework provides this level of control.

The emerging domain of P2P media streaming is one of many application areas that suffers from these limitations. Several P2P streaming designs have been proposed (e.g. ZIGZAG [4], PeerCast [5], CoopNet [6]), many of which are preoccupied with the formation and maintenance of P2P multicast trees that are optimal according to some application specific criteria. Clearly, the development of such systems would benefit from an end-system multicast framework whose tree management scheme could be easily adapted to such criteria.

To address the aforementioned concerns, this paper introduces the Channel Infrastructure for P2P Systems (CHIPS) framework. At the core of this framework is a label-switching inspired, three-way handshake protocol that enables a member of a dynamic peer group to: (1) advertise a new channel, (2) request subscription to a channel through a particular path of peers and (3) approve a subscription request or reject it, optionally specifying an alternate path in case of the latter. By giving a channel source explicit control of the paths to each of its subscribers, CHIPS enables applications to define customized distribution graph management schemes that, in turn, enable the implementation of rich QoS and other policy requirements. Note that the framework provides a simple plug-in mechanism that allows the specification of such QoS and other policies in a modular fashion.

As part of this work, Multicast Video (MCVideo), a CHIPS-based P2P media streaming system, is developed. While it should be emphasized that P2P media streaming is by no means the only application of CHIPS, the effective N-to-M content distribution achieved in this example system (i.e. N source peers transmitting separate

streams to their own multicast groups) and its successful use of a custom QoS plug-in, demonstrate the success of CHIPS in addressing the two limitations of existing approaches to end-system multicast.

The remainder of this paper is structured as follows. Section 2 outlines the architecture of a typical CHIPS based application and Section 3 details the three-way handshake label switching protocol at the core of the CHIPS framework. Section 4 presents the P2P media streaming application developed using CHIPS, providing a detailed description of the QoS plug-in it uses. Finally, and Section 5 concludes.

2. Architecture

Figure 1 illustrates the structure of a typical CHIPS-based application peer. The salient feature of this architecture is an effective separation of concerns in which multicast infrastructure construction and management is handled by CHIPS and its supporting components, while specification of the QoS and other domain specific policies that govern the design of this infrastructure is relegated to a higher-level application layer. Each of the elements in this architecture is briefly described below.

The application layer on top of the CHIPS framework provides QoS and policy modules that together, contain all logic and data specifying the distribution graph management scheme used by the application. These modules can communicate with CHIPS to gather relevant network statistics (e.g. inter-node loss rate, bandwidth, delay, etc.) which are then used to compute a description of the optimal distribution graph topology. CHIPS can then draw on this description to form and maintain suitable topologies. Note that the modules implement a standard plug-in interface in order to communicate with CHIPS.

The CHIPS framework relies on an end-system multicast middleware to provide application-level routing information that, in turn, is used to construct and maintain a distribution graph (typically a multicast tree) for each channel. This middleware is also used to manage membership in the overall peer-group (e.g. detect new, failed, or exiting peers) and optionally, collect relevant network statistics. While at present the Application Level Multicast Approach to P2P Dynamic Groups (ALMAP) framework is used, communication with the multicast middleware takes place through an adapter to ensure that any suitable replacement can also be employed.

Finally, CHIPS may also rely on application specific unicast middleware to support the basic inter-node communication from which multicast semantics are ultimately built and perhaps also to collect additional network statistics. In a P2P media streaming system, for example, unicast middleware providing access to RTP and RTCP functionality may be appropriate. Again, communication with this component takes place through an adapter to ensure the independence of CHIPS from any single middleware.

Figure 1. Architecture of a CHIPS-Based Application Peer

3. CHIPS Protocol

In this section we introduce the three-way handshake label switching protocol at the core of the CHIPS framework, henceforth referred to as the "CHIPS Protocol". This protocol was designed with the following requirements in mind:

(R1) The protocol should enable the two fundamental goals of the CHIPS framework: (1) to allow fine granularity multicast communication within peer groups and (2) to allow programs built on top of CHIPS to define arbitrary, application specific QoS mechanisms and data distribution policies.

(R2) The protocol should be self-organizing, in the sense that node joins, exiting and failures should be transparent to applications built on top of CHIPS.

(R3) The protocol should be easily adapted to work on top of the various application-level routing protocols implemented by different end-system multicast frameworks. In particular, note that while this paper only considers application-level routing protocols based on shortest paths spanning trees rooted at each member of a peer-group, the CHIPS protocol should not necessarily be confined to this approach.

Section 3.1 provides an overview of the basic protocol, Section 3.2 presents the algorithms used by the protocol to handle node joins, exiting and failures and finally, Section 3.3 briefly discusses how the protocol addresses the second part of requirement R1.

3.1 Basic Protocol

The basic CHIPS protocol is structured as a three-way handshake. Before detailing this mechanism, however, the concept of label switching must be introduced. In simple terms, a "label" is an identifier that can be used to uniquely determine a remote entity or resource. In the context of the CHIPS protocol, a label is an IP address, port number pair identifying a source or

destination of channel data packets (e.g. 142.103.7.14, 3000). In a label switching protocol, such as the CHIPS Protocol, every node maintains a forwarding table, each of whose entries contains a pair of labels: an "In" label and an "Out" label. An example forwarding table is shown in Figure 2. Each such entry in the forwarding table for a node X says that any data packet received from the entity identified by the "In" label should be forwarded to the entity identified by the "Out" label. In the discussion below, note the similarities to other label switching protocols (e.g. Multi-Protocol Label Switching [7]).

Channel ID	In	Out
1	142.103.7.14:3000	142.103.7.18:4000
2	142.103.7.7:4500	142.103.7.56:5000

Figure 2. Example Forwarding Table for a CHIPS Peer

The first step in the CHIPS Protocol three-way handshake is for a peer to start a channel by broadcasting a "content available" message to the entire peer group. This message is broadcasted using application-level routing information provided by the underlying end-system multicast middleware. Note that while this information typically specifies shortest paths spanning trees rooted at each peer, the use of such trees is not a hard requirement.

When a content available message arrives at a node X from its parent node Y in the broadcast graph topology, it contains at least two pieces of information. First, it has the label for Y. Second, it has the accumulated path from the content provider at which the message originated, up to X (e.g. $C, N_1, N_2, …, Y$). When such a message arrives at X, it: (1) records the label of the parent Y from which it received the message as the "In" label in a forwarding table entry for the corresponding channel, (2) modifies the message by replacing the label of Y with its own label and updating the accumulated path information to contain X and (3) forwards the modified message to its children in the broadcast graph topology. The first step of the three-way handshake is complete once the content available broadcast is finished.

The second step of the three-way handshake is channel subscription, in which an interested node subscribes to a channel by sending a unicast "subscribe" message to the content provider. This message goes from the potential subscriber to the content provider following the reverse of the path used to get the content available to the potential subscriber. Upon arrival at each intermediary node Y from a previous child node X, this message contains the label of X, which is used to set the "Out" field of the corresponding router table entry. Y then sets the label field in the message to its own label and forwards it to its parent along the route. Note that in traversing the path back to the content provider we are actually using "reverse" label switching by using the "In" labels at each node to determine the next hop.

The final, optional step of the three-way handshake is subscription approval. If the content provider chooses to accept the subscription request, a "subscription approval" message is sent to the subscriber through the route negotiated in the previous two steps of the three-way handshake. At each node along this route, the corresponding forwarding table entry is marked as valid. Note that by providing appropriate additional parameters in this message, interesting node-based QoS mechanisms can be implemented (i.e. resource reservation, etc.).

Upon successful completion of the three-way handshake, a channel distribution graph (again, typically a multicast tree) containing routes from the content provider to all subscribing peers is established. The channel source can start transmitting content at this point.

3.2 Group Management

Although the basic three-way handshake mechanism described above can be used to establish a content distribution graph for each channel, it does not provide an on-line mechanism for adapting these graphs as nodes fail, or join and leave the peer group. The subsections below detail the algorithms used by CHIPS to handle these cases.

3.2.1 Node Join

The CHIPS protocol can be configured (by using the appropriate policy module) to employ either a subscriber-based or a provider-based approach to handling node joins. In the former, when a new node joins a group it uses the underlying end-system multicast middleware to broadcast a "channel collection" message to all peers. If a content provider decides to respond, it triggers a label distribution procedure similar to that used in the first step of the three-way handshake protocol, but different in two key respects. First, the response to the channel collection request is a unicast message, unlike the broadcast content available message. Second, rather than directly using the routing information provided by the underlying end-system multicast middleware, the topology of the existing distribution graph is used to route the channel collection request and any response.

In the provider-based approach, a source peer interacts with the end-system multicast middleware to listen for new nodes. Upon notification of a node join, it can then proactively trigger a unicast label distribution equivalent to that used above. This latter approach avoids the broadcast channel collection message and allows content providers to take the initiative by proposing routes to the new node from the outset.

3.2.2 Node Exit and Failure

The handling of node exiting and failures differ only in that the former is triggered by a "leave" message explicitly broadcast by a departing node via the end-system multicast middleware, whereas the latter is triggered when the end-system multicast middleware notifies the CHIPS framework of a downed node. In both cases, CHIPS must respond by repairing any damaged

channel distribution graphs.

Once the repair process has been initiated for a particular channel, the corresponding source peer goes into a "channel repair" state in which no new subscriptions are allowed. Using its complete knowledge of the paths to each of its subscribers, this content provider computes the set of nodes in its distribution graph affected by the node exit or failure. It then instructs all of these affected nodes to disconnect from the other nodes (by clearing their forwarding tables), as well as to stop receiving and forwarding data.

At this point, a "change route" message is broadcast through the peer group using application-level routing information provided by the underlying end-system multicast framework. As this message is received by a node X from its parent Y, it is processed as follows. First, X retrieves the entry in its forwarding table for the channel. If it finds that no such entry exists, or that the channel is "inactive" (i.e. no data for the channel is passing to or through X), it then creates or updates, respectively, a table entry that points to Y as the "In" node for that channel. Alternatively, if the channel is active for node X, then no changes to the forwarding table are made, leaving the route to X unchanged. This latter provision ensures that channel repair is transparent to unaffected peers receiving channel data. The change route message is then updated with the label of the processing node and passed along to its children in the topology used by the end-system multicast framework to broadcast the message. Once the "change route" message has been processed, the content provider moves out of the "channel repair" state and the affected nodes may resume their subscription.

3.3 Application Control of Data Distribution

The power of the novel label distribution scheme used by CHIPS lies in the fact that it gives applications greater knowledge of and control over distribution graph topology. This greater knowledge and control is provided in a number of ways. First, for any route from the content provider to a potential subscriber to be usable, that route must first be proposed by the content provider through either a broadcast label distribution step (i.e. that described in Section 3.1) or a unicast label distribution step (i.e. that alluded to in the discussion on provider-based handling of node joins). Second, the content provider has the power to accept or reject any subscription request. Considering these two points alone, it is evident that a channel source effectively has complete control over the paths to each of its subscribers. Finally, while a content provider clearly has a global view of its distribution graph under this scheme, each node in this graph also has knowledge of the route back to the content provider, given the accumulated path information stored in the messages used to propose routes.

Although not explicitly provided in the current implementation of CHIPS, a number of trivial extensions of this framework can provide even greater control. For example, intermediary peers could easily reject the

formation of paths through them for certain channels by allowing them to "cut off" label distribution. In this way, peers are given greater control over the allocation of their bandwidth. Another extension would be to annotate the messages used by the content provider with certain parameters that, in turn, specify the per-hop behaviour of nodes along that route.

In all cases, greater application control of the distribution graph allows the formation and management of more sophisticated graphs – thereby facilitating richer QoS and distribution policies.

4. MCVideo

To demonstrate the viability of a CHIPS-based application, a P2P media streaming system, MCVideo, was developed using this framework. MCVideo consists of roughly 7000 lines of Java code, 3000 lines implementing the user interface and 4000 lines implementing the QoS and policy modules that plug into CHIPS. (As an aside, note that the CHIPS framework itself consists of 6000 lines of Java code and the underlying ALMAP framework, discussed in Section 4.2, consists of an additional 8000 lines.) The subsections below present this application by detailing for it the particular implementations of each of the components above and below CHIPS in the architecture diagram of Figure 2.1.

4.1 User Interface

MCVideo has a simple GUI composed of three tabs: the Message Manager, the Broadcast Manager (Figure 3) and the Reception Manager (Figure 4). The first of these is a trivial instant messenger, used for initial coordination of content providers and subscribers, that allows peers to broadcast text messages to the global peer group.

The Broadcast Manager allows a peer to create, start transmitting and stop transmitting a channel. Note that for MCVideo a channel is simply a multiplexed stream of H.263 encoded video and GSM encoded audio, derived either from default capture devices or a file. As shown in Figure 4, the Broadcast Manager for a peer transmitting a channel also incorporates a visualization tool that

Figure 3. Broadcast Manager Screenshot

illustrates the current topology of the distribution graph rooted at that node (which is always a multicast tree for MCVideo) to distribute data to subscribing peers.

Finally, the Reception Manager presents the user with a list of currently available channels. Subscription to any of these channels can be requested and a list of channels for which these requests are approved is maintained. The user can elect to receive data from any channel to which he or she is subscribed and if data is currently being transmitted on that channel, it will be rendered in a player with basic audio and video controls.

Figure 4. Reception Manager Screenshot

4.2 End-System Multicast Middleware

As discussed in Section 2, CHIPS relies on an underlying end-system multicast middleware for peer membership management and obtaining multicast routing information. MCVideo uses the ALMAP framework for this purpose. In brief, ALMAP employs a semi-centralized approach to manage the dynamic peer group and computes application-level routing information representing a shortest path spanning tree rooted at each node.

4.3 Application Specific Unicast Middleware

In addition to using TCP for the inter-peer communication implementing the three-way handshake label switching protocol underlying CHIPS, CHIPS-based applications can use of any unicast middleware appropriate to their unique data distribution needs. MCVideo, for instance, uses implementations of RTP and RTCP to stream media from one peer to another and collect statistics for the corresponding inter-peer link. These implementations are provided by the Java Media Framework (JMF) [8], an optional extension of the Java 2 Platform providing media support. (Note that the JMF is also used for H.263 and GSM encoding and decoding, as well as for audio and video capture and playback.)

4.4 Policy Module

MCVideo makes use of a trivial policy module that accepts all subscription requests posed to it. Thus, if a particular node asks to receive channel data through any path of peers it specifies, that request will be granted by the source peer for that channel.

4.5 QoS Module

The QoS module for MCVideo specifies a simple mechanism that maintains "optimal" multicast trees rooted at each channel source for content distribution. (Note that the initial multicast trees used to bootstrap the system are always derived from application-level routing information provided by the end-system multicast middleware supporting CHIPS.) Using the MCVideo QoS module, CHIPS is instructed to carry out three tasks: (1) gather network statistics, (2) compute a new multicast tree based on these statistics and (3) adjust the actual multicast tree based on the tree just computed. Each of these parts is detailed in the subsections below.

4.5.1 Gather and Monitor Receiver Statistics

In MCVideo, the CHIPS instantiation at each peer determines an estimated bit rate for the incoming P2P link from its parent in the multicast tree, as well as the loss rates of the outgoing P2P links to its children in the multicast tree, by interacting with the unicast middleware to access RTCP sender and receiver reports. By propagating these statistics up each multicast tree, the source peer for each channel has knowledge of (1) the bit and loss rates for each of the links in its multicast tree and (2) the overall bit and loss rates for each subscribing peer.

The CHIPS instantiation at each source peer monitors these statistics and should it find that the loss rate for a particular subscriber exceeds a predefined threshold for a sufficiently long period, it initiates an adjustment task for this subscriber. This adjustment task consists of two parts: (1) computation of a new multicast tree and (2) modification of the existing tree based on this new tree.

4.5.2 Multicast Tree Computation

Before discussing multicast tree computation, it should be noted that the MCVideo QoS module makes two assumptions about the end-system multicast middleware used to support CHIPS. First, it assumes that this middleware designates a single peer as a centralized controller. Second, it assumes that this controller knows the delay of the direct P2P link between any two members of the global group. Clearly, the ALMAP framework satisfies both of these assumptions by electing a single controller and propagating the results of ping-based delay measurements to this node. As will be evident shortly, these assumptions are only necessary to make sure that the node computing a multicast tree has access to a global view of the P2P network. Distributed approaches that ensure this latter requirement can also be used if an end-system multicast framework not satisfying the assumptions above is unavailable.

In MCVideo, the channel source peers periodically forward their statistics for the bit and loss rates of each of the links in their trees to the CHIPS instantiation at the central controller. This central instantiation, in turn, maintains a fully connected weighted graph that has a

single node for each peer in the global group and each of whose edges represents a P2P link (which may or may not be part of a current multicast tree). The weight *weight*(*e*) of an edge *e* in this graph is defined as follows:

$$weight(e) = \alpha * (\ delay(e)\ /\ max\_delay\ ) +$$
$$\beta * (\ loss\_rate(\ e\ )\ /\ max\_loss\_rate\ ) +$$
$$\gamma * (\ bit\_rate(\ e\ )\ /\ max\_bit\_rate\ )$$

Here *delay*(*e*) is the delay of the P2P link corresponding to *e*, *loss_rate*(*e*) is the loss rate, *bit_rate*(*e*) is the estimated bit rate and *max_delay*, *max_loss_rate* and *max_bit_rate* are the maximum values for these metrics across all links. Obviously, if no bit and loss rate measurements are available for a particular edge *e*, then the corresponding link is not being used and so *bit_rate*(*e*) = *loss_rate*(*e*) = 0. Also, note that $\alpha, \beta, \gamma \in (0, 1]$ and $\alpha + \beta + \gamma = 1$. The selection of these parameters is based on experience and specific application requirements.

When a particular channel provider deems that adjustment of its multicast tree is necessary, it notifies the controller. This controller then makes use of Dijkstra's algorithm on its weighted graph to compute a new multicast tree as the shortest paths spanning tree from the requesting node and returns this tree.

4.5.3 Multicast Tree Adjustment

Although the multicast tree computed by the controller is in some sense optimal, it is not directly adopted by the requesting content provider, but rather, is only used to adjust the path from the channel source to the subscriber whose high-loss rate triggered the adjustment. In this way, the tree is updated with minimal disruption to other subscribers. Also, by limiting fluctuations in the multicast tree topology, the data streams running across this tree have more time to stabilize and so, statistics measured for its P2P links are more reliable.

The algorithm used to adjust the current multicast tree based on the newly computed tree is similar to that used to handle failed nodes. Recall that the content provider knows the paths to all the receivers and thus, can compare the current multicast tree with the new tree. By making such a comparison, the source peer first identifies all nodes affected by the adjustment and computes new routes for them (the details of these steps are discussed shortly). It then tells each of the affected nodes to disconnect itself from all other nodes (by clearing its forwarding table) and cease to receive and/or forward data. The source peer now uses a unicast label distribution step for each of the affected peers to construct the new paths computed for them. Once these paths have been setup, the affected nodes can resume subscription, if previously subscribed, as well as receive and/or forward data again.

The algorithm for identifying affected nodes and computing new paths to them is as follows. Starting from the receiver whose loss rate triggered the tree adjustment, we compare the route to the source peer in the current multicast tree with that in the new tree on a hop-by-hop basis. Upon arriving at the first differing pair of nodes, the children of the original receiver are connected to the current node in the old tree and the receiver itself is connected to the current node in the new tree. The affected nodes are the original receiver and its descendants and the new paths for these affected nodes are as determined above. From the first different point in the new route, the operations above can be carried out recursively until the content provider is finally reached.

5. Conclusions

In this work Channel Infrastructure for P2P Systems (CHIPS), a novel end-system multicast framework, is introduced. Through the use of a label-switching, three-way handshake protocol, CHIPS provides two features not available in previous approaches to end-system multicast. First, rather than restricting itself to broadcast communication within a peer-group, multicast transmission to subgroups is supported, where a subgroup is defined as the set of peers subscribing to a particular channel. Second, knowledge of and control over the multicast graph topology used by a channel is given to the source peer for that channel. This, together with a simple plug-in mechanism, allows applications to implement, in a modular fashion, arbitrary distribution graph management schemes that ensure QoS and various other constraints are observed. While the generality of this framework makes it applicable to many domains, its particular relevance to distributed multimedia has been demonstrated by the implementation of an effective CHIPS-based P2P media streaming system with basic QoS support.

6. References

[1] Y. Chu, S. Rao, S. Seshan and H. Zhang. A Case for End System Multicast. IEEE Journal on Selected Areas in Communication, 20:1456-1471, 2002.

[2] D. Pendrakis, D. Verma, S. Shi and M. Waldvogel. ALMI: An Application Level Multicast Infrastructure. In *Proc. of USITS*, 2001.

[3] S. Vuong and J. Shao. ALMAP: An Application Level Multicast Approach to Peer-to-Peer Dynamic Groups. In *Proc. of PDPTA*, 2003.

[4] D. Tan, K. Hua and T. Do. ZIGZAG: An Efficient Peer-to-Peer Scheme for Media Streaming. In *Proc. of ICC*, 2003.

[5] H. Deshpande, M. Bawa, and H. Garcia-Molina. Streaming Live Media Over Peers. *Stanford Database Group Technical Report (2002-21)*, 2002.

[6] V. Padmanabhan, H. Wang, P. Chou and K. Sripanidkulchai. Distributing Streaming Media Content Using Cooperative Networking. In *Proc. of NOSSDAV*, 2002.

[7] E. Rosen, A. Viswanathan and R. Callon. RFC 3031: Multiprotocol Label Switching Architecture, 2001. URL: http://www.faqs.org/rfcs/rfc3031.html.

[8] Java Media Framework API, 2004. URL: http://java.sun.com/products/java-media/jmf.

Adaptive Video Transmission
Over A Single Wireless Link[+]

Hao Wang, S. Venkatesan

Telecommunication Engineering Department
The University of Texas at Dallas, P.O. Box 830688
Richardson, TX 75083-0688
{haowz,venky}@utdallas.edu

Abstract

In this paper, we propose an adaptive priority-based selective repeat ARQ (PSR-ARQ) transmission scheme for transmitting video over a single-hop lossy wireless link. If transmission errors occur in a lower layer of the video, the errors propagate to all the higher layers of related frames resulting in significant degradation of the video quality at the receiver. Due to the limited channel bandwidth, retransmission of every corrupted packet may be unacceptable for video streaming applications. Our proposed on-line priority-based scheme prioritizes all layers of various frames of a group of pictures and transmits them over a lossy wireless link. The on-line transmission policy achieves good video quality (of small distortion). Simulation results show that the proposed scheme performs better than the traditional scheme for video transmission over wireless channels.

I. Introduction

The trend to streaming stored media over wireless networks is emerging. Streaming video stored at the video server is the technique to achieve smooth playback of video directly at wireless terminals without downloading the entire file before the playback begins. While the demand for such video communication over wireless links has increased considerably, the transmission rate of the wireless channels remains low. In addition, wireless links are error prone. The limited bandwidth and high bit error rate (BER) can be problematic for video streaming over wireless networks. In this paper, we consider the problem of transporting video over a single-hop wireless channel. The single-hop wireless channel can be a dedicated channel as in mobile cellular scenarios or a shared channel as in Wi-Fi (802.11).

Video signals transmitted through the wireless channels are corrupted by two types of errors: (1) Stationary random errors that depend on the mean strength of the received wave or on the distance between the station and the wireless terminal and, (2) Variable error that is caused by the motion of the portable

terminal or Rayleigh fading that is caused by the wave strength dips resulting in burst errors [8]. If the wave strength is 10db smaller than the mean wave strength, BER can be 20 times worse. Faster the portable terminal moves, higher the frequency of distortion caused by burst errors and more the deterioration in video quality.

While forward error-correcting codes (FEC) can be used to reduce the effects of transmission errors at the decoder, the associated increase in bit payload is often unacceptable, particularly in a wireless environment, where bandwidth resources are severely limited. Error concealment techniques have been applied to decoders such that they can extrapolate information from received bits to reconstruct the lost information. However, when large amounts of data are lost due to burst errors, the reconstruction process has insufficient information to work on.

Closed-loop error control techniques like automatic repeat request (ARQ) have been shown to be very effective and have been successfully applied to wireless video transmission [2,4,5]. However, retransmission of corrupted data frames introduces additional delay, which might be unacceptable for real-time conventional services. A combination of ARQ and FEC (hybrid ARQ) is recognized as a good choice for video over wireless.

Liu and Zaiki [7] classify the hybrid ARQ scheme into two categories: type-I and type-II schemes. A general type-I ARQ scheme uses error detection and correction codes for each transmission and retransmission. The receiver attempts to correct the errors by decoding the current received code words. Previously received code words that contain uncorrectable errors are discarded. A general type-II ARQ scheme uses a low rate error correction code. An information packet is first transmitted with some correction bits for error correction. Incremental redundancy bits are transmitted upon retransmission request. The receiver combines the transmitted and retransmitted packets together to form a more powerful error correction code to recover the information. For time-varying wireless channels with burst errors, type-II ARQ scheme may suffer from poor performance. Refinements of ARQ schemes for video have been proposed to alleviate this

problem. Pyun.et.al [2] proposed a hybrid ARQ with interleaving scheme, which reduces the number of retransmission to minimize the delay. Zhang and Kasam [10] proposed a hybrid ARQ scheme, which takes advantages of both type-I and type-II hybrid ARQ, and this partially solves the problem of inadequacy of error correction code to recover the information. For time-varying wireless channels with burst errors, type-II ARQ scheme may suffer from poor performance. Wang, Zheng and Copeland [9] proposed QOS selective repeat ARQ (QSR-ARQ) scheme to enhance the existing data link layer protocols in wireless mobile environments. This scheme delivers I and P frame packets with limited numbers of retransmission to guarantee the delay bound while it transmits B frame packets only once to avoid long delays caused by retransmission [1, 14].

In this paper, we propose a transmission scheme that is priority-based selective repeat (PSR) ARQ. Layered video packets are first sequenced at the sender based on the contribution to the entire video quality and inter-dependency. The base layer of the I-frame is the most significant one and is at the head of the sequence. The highest layer of P/B frame is of least significance and is placed at the tail of the sequence. Our PSR-ARQ scheme increases the chance of successful transmission of important packets by maximizing the number of retransmissions within the delay bound. The PSR-ARQ scheme works at the application layer on tops of the UDP/IP stack. The link layer provides the acknowledgement scheme. For each transmission of data frame at the link layer, the link layer will send the ACK/NACK message to the PSR-ARQ. Based on the link layer acknowledgement, PSR-ARQ controls the video packet retransmission. This requires control traffic between the link layer and the application layer.

II. PSR-ARQ Transmission Scheme

The output of H.263 compression is a sequence of three types of frames: I frame (Intra-picture), P frame (forward Predicted) and B frames (Bidirectional predicted). I frames are reference frames and are self-contained. A P frames specifies the difference between the previous I frame; a B frame is an interpolation between the previous and subsequent frame of I or P type. If an I frame is lost, all the subsequent P and B frames (until the next I frame) are of no value. If a P frame is lost, all the previous and subsequent consecutive B frames (till the next I or P frame) are of no value. Clearly I-frames that start each scene are statistically larger (in size) than the P or B frames because I-frames are self contained.

Layered video coding is very useful in coping with the time-varying nature of wireless channel conditions [3]. A scalable encoder encodes video into several layers. The base layer guarantees a basic display quality and each enhancement layer (correctly received by the receiver) improves the video quality. Only base layer and few enhancement layers are

received correctly if the channel exhibits transmission errors. If the channel condition is good, many enhancement layers (in addition to the base layer) are successfully transmitted and good video quality is achieved at the receiver.

We assume that layered coding encodes an I-frame into a hierarchy of X layers $\{I (BL), I (EL_1)… I (EL_{X-1})\}$, where I (BL) is the base layer while layers I (EL_1) to I (EL_{X-1}) are enhancement layers. Layered coding encodes P-frame and B-frame into a hierarchy of Y and Z layers. Accordingly, $\{P (BL), P (EL_1), P (EL_{Y-1})\}$ are generated for each P frame and $\{B (BL), B (EL_1), B (EL_{Z-1})\}$ are generated for each B frame. The layered video is shown in Figure 2. Note that we have $X>Y>Z$ since I frames are larger (in size) than the related P frames and P frames are larger than B frames.

To reduce the amount of the computation performed at the sender when transmitting, we order all the packets (off-line). The rule is based on the computed distortion (to the entire GOP (Group of Picture) of each packet. A sequence of ordered video packets is available at the sender. The packets contribute to the entire video quality with decreasing significance from the left side to the right side in the ordered sequence. If the given video sequence is I B_1 B_2 P_1 B_3 B_4 P_2 B_5 B_6 P_3, the ordered packet sequence could be similar to the following:

$I(BL)$, $I(EL_1)$, $P_1(BL)$, $P_2(BL)$, $P_3(BL)$, $B_1(BL)$, $B_2(BL)$, $B_3(BL)$, $B_4(BL)$, $B_5(BL)$, $B_6(BL)$, $I(EL_2)$, $I(EL_3)$, $P_1(EL_1)$, $P_2(EL_1)$, $P_3(EL_1)$, $B_1(BL)$, $B_2(EL_1)$, $B_3(BL)$, $B_4(EL_1)$, $B_5(BL)$, $B_6(EL_1)$…..

In the above list, I (BL) represents the base layer of I frame. P_2 (EL_1) represents the first enhancement layer of P_2 and B_2 (EL_1) represents the first enhancement of B_2. By changing X, Y and Z values properly, it is reasonable to assume that each video packet is of the same size statistically.

The algorithm presented in Figure 1 constructs the sequence. Note that if frame P_1 is ordered prior to P_2, its related B frames of P_1 should be ordered prior to the B frames related to P_2 as well.

In our scheme, packets are transmitted by on-line priority-based selective repeat ARQ with acknowledgement of ACK or NACK. The video streaming server keeps two transmission queues. Initially, video packets of first two GOPs put into each transmission queue separately according to the off-line ordering algorithm. Retransmission is required if no ACK is received before timeout or a NACK is received. Upon receiving an ACK, the server deletes the transmitted video packet from the transmission queue. The server selects the first video packet from the first queue for transmission. The reason we choose PSR-ARQ is because of the strong data dependency among video packets. Packets $P_1(BL)$, $P_2(BL)$ and $P_3(BL)$ are useless unless I(BL) is transmitted successfully.

```
j = 1;   //notation: j = 1: BL, j =2: EL₁, j=3: EL₂ …
L = {all packets of all frames of current GOP}
Create an empty queue.

// X, Y and Z are the total number of layers of I, P or B
// frames respectively.

DO
{
Delete X/Z layers of I frame (from the bottom to the top)
from L and add them to the queue (again order it from
bottom to top)

Delete Y/Z layers of all P frames from L and add the deleted
layers to the queue (in the order of decreasing significance)

Delete layer j of all B frames and add the deleted layers to
the queue (in the same order as their related P frames).

j = j+1;

} WHILE (j<=Z)

// The queue now has the ordered the video packet sequence
```

Fig. 1 Video packet sequence off-line ordering algorithm

Each packet will be retransmitted in case of loss or packet corruption, but the maximum number of times a packet can be transmitted is limited. The number of retransmissions of a packet is bounded by the timestamp of current GOP. Packets belonging to the same GOP are buffered at the receiver before the entire GOP is scheduled to be decoded and displayed.

An important component of our on-line PSR-ARQ transmission scheme is the on-line transmission policy at the sender. The on-line policy dictates which packet at the source should be (re)transmitted. For every feasible set of unsent (or retransmission required) video packets, the decision made by the policy must increase the video quality by minimizing the expected distortion of the entire video. The expected distortion of a video packet is defined as the multiplication of absolute distortion of the video packet and the overall transmission failure probability of maximum number of retransmission.

The on-line transmission policy is simple when deciding which packet should be transmitted in current GOP. It selects the first packet of current GOP in the transmission queue. However, the decision is not clear when choosing between the first packet of the current GOP and the first packet of the next GOP. For instance, as shown in the Figure 3, packet I (BL) of GOP M+1 is of greater significance than the unfinished packet, say $B_k(EL_Z)$ in GOP M. But packet $B_k(EL_Z)$ has less

number of chances to transmit. Our on-line transmission policy computes the expected distortion of the next packet of the current GOP and of the first packet of the next GOP. Then it chooses the packet with the smaller expected distortion value to transmit. The computation of expected distortion is given by equations (5) and (6) at the end of section II.

The on-line PSR-ARQ transmission algorithm is shown in Figure 4. Next we evaluate our scheme. A wireless channel can be described as a packet-based burst error channel [11, 12]. One commonly studied analytical model is the Gilbert model (two-state Markov model), which characterizes the bursty nature of wireless channel errors. The model has two states, a good state (S_0) and a bad state (S_1). P_{ij} is the transition probability from state i to state j. P_{00}, P_{01}, P_{10}, and P_{11}, are the state transition probabilities. P_{00} represents the probability of transition from good state to good state. P_{10} represents the

Fig. 2 The layered video frame sequence of a GOP

Fig. 3 The on-line transmission scheme

Sequence all GOPs in the video by number 1, 2...in their temporal order.

Pre-buffer M-1 GOPs at the receiver. The time-bound for following GOP transmission is given by the time-stamp of that GOP.

At sender, for each GOP, order all packets off-line using the algorithm of Figure 1. Initially, put packets of GOP M and M+1 into two transmission queues respectively.

K=M;

DO
{ IF (GOP K exceeds its time-stamp) {
   Dump all packets of GOP k from the queue.
   Put packets of GOP K+2 into that empty queue.
   K++;
   }
 ELSE {
  Compute the expected distortion by sending the first active packet of the GOP K and denotes it as D(1).

  Compute the expected distortion by sending the first active packet of the GOP K+1 and denotes it as D(2).

  IF(D(1)<D(2))
   Transmit the first active packet of the GOP K.
  ELSE
   Transmit the first unmarked packet of the GOP K+1.
  }
}WHILE (the video transmission is not completed)

Fig. 4 On-line PSR-ARQ transmission algorithm

probability of transition from bad state to good state. State transitions occur at discrete time instants. In our work, we assume the time between two successive time instants is equal to the time taken for transmitting one packet at the MAC layer. A packet is transmitted correctly when the channel is in good state and errors occur when the channel is in bad state. The transitions between states occur at each packet instant. The channel state-transition probability matrix can be set up as

$$P = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} 1-P_{01} & P_{01} \\ P_{10} & 1-P_{10} \end{bmatrix}; \quad P(S_0) = \pi_0; \quad P(S_1) = \pi_1.$$

The packet error statistics vary according to the values of the transition probabilities. The transition probabilities can be calculated from the channel characteristics, i.e. the packet-error-rate (PER) and average-burst-length (ABL). Given the

same PER, large ABL implies that errors are bursty. We have

$$ABL = \frac{1}{P_{10}}; \qquad (1)$$

$$PER = \frac{P_{01}}{P_{10} + P_{01}}. \qquad (2)$$

To simulate the burst error, we generate a uniformly distributed random number r in (0,1). The transition from S_0 to S_1 occurs when r is less than P_{01} and the transition from S_1 to S_0 occurs when r is less than P_{10}.

With the assumption that any bit error will cause the packet error. Therefore $PER = 1- (1-BER)^L$, where L is the packet length. When the packet length becomes longer, PER becomes higher.

The time-bound for transmitting current GOP is determined by the duration T_1 before the last frame of the current GOP is scheduled to decode and display. Accordingly, the maximum number of packets F (including retransmitted packets) transmitted for the current GOP can be computed as:

$$F = \frac{T_1 * B}{S}; \qquad (3)$$

Where B is the estimated bandwidth of the wireless link and S is the packet size. B can be estimated fairly accurately and the details appear in the next section.

The maximum number of packets G (including retransmitted packets) transmitted for the next GOP with duration T_2 can be computed as:

$$G = \frac{T_2 * B}{S}; \qquad (4)$$

Suppose in the process of current GOP transmission, the next scheduled layer of current GOP is $X_i (Y_j)$, which has the absolute distortion D $(X_i (Y_j))$. The expected distortion of that layer can computed as:

$$D(1) = D_E(X_i(Y_j)) = PER^F * D(X_i(Y_j)); \quad (5)$$

The scheduled layer from next GOP is $X_a(Y_b)$ with absolute distortion $D(X_a(Y_b))$. The expected distortion of that layer can computed as:

$$D(2) = D_E(X_a(Y_b)) = PER^G * D(X_aY_b); \qquad (6)$$

III. Simulation Studies

We use three video sequences (foreman, waterski and wg_cs_9) to measure the simulated performance. In our

experiments, only the DC value is included in the base layer. Each enhancement layer contains one or more AC values. The higher enhancement layers include AC values with larger frequency than AC values contained in lower enhancement layers. PSNR (peak signal-to-noise ratio) is used to evaluate the layered video quality. Higher the value of PSNR, better the video quality is. Also, we assume that all the motion vectors are decoded correctly.

First, we analyzed video quality with respect to different number of AC values. The PSNR values are shown in table 1 and Figure 5. Second, the gains of PSNR by adding more DCT values are computed by deducting the PSNR value from the previous one in Table 1. Note that Table 1 and Figure 5 show the PSNR values of a typical frame.

Based on the observation of Table 1, we use the video layering policy shown in Table 2.

The layered video packets are transmitted over a wireless link. In equation (5) and (6), estimated wireless bandwidth is required for computation. The available wireless bandwidth can be estimated by the equation (7).

$$B_{next} = k * B_{past} + (1-k) * B_{curr} \qquad (7)$$

Where B_{next} is the estimated wireless channel bandwidth for the next period. B_{past} is the past estimate of wireless link bandwidth and B_{curr} is the measured net bandwidth in the current estimation period. Note that B_{past} of next estimation period equals to B_{next} of current estimation period. The estimation coefficient is k ($0<k<1$).

The estimated and the actual wireless channel bandwidths are shown in Figure 6. Figure 6 shows a good estimation of wireless bandwidth given by equation (7).

We compare our proposed on-line PSR-ARQ transmission scheme with the traditional scheme (SR ARQ). In our simulation, the PSNR values are obtained by averaging the absolute PSNR values of video GOPs transmitted by our scheme and the traditional scheme. The PSNR values of the proposed and the traditional schemes for different PER wireless link are presented in Figure 7. Figure 7 shows that the proposed transmission scheme performs better than the traditional scheme. As the PER increases, PSNR gain of our scheme over the traditional scheme increases.

IV. Conclusion

In this paper, we have presented a scheme for layered video streaming over a wireless link. Our PSR-ARQ scheme minimizes the significant packets' failure ratio by maximizing the number of retransmission within the delay bound. Our

transmission policy achieves good video quality (of low distortion). Simulation results show that our scheme performs better than traditional scheme for video transmission over a single wireless link with limited bandwidth and high burst errors. If the wireless link provides QoS, for example when using 802.11e, it is clear that the wireless link as viewed by our scheme exhibits good behavior (low latency, etc) and hence the video quality will be good. Note that since our scheme is adaptive, the video quality is automatically adjusted based on the quality of the wireless link. Many concurrent schemes can be supported on a single wireless channel of sufficient bandwidth due to the very small real-time processing overhead. Since significant preprocessing is done, supporting live video requires large buffering delay at the sender and hence the end-to-end latency can be large. If audio is encoded in a layered way, our approach can easily be adapted to audio streaming also. The details of the encoding method used in audio will be needed to accomplish this.

Our future work will be video transmission over multi-hop wireless links or wired/wireless links.

Table 1 PSNR with different DCT values

	wg_cs_9	foreman	waterski
DC	24.25	26.47	29.04
DC+AC1	24.66	26.35	29.09
DC+AC1~2	32.84	34.14	34.28
DC+AC1~3	36.02	36.4	36.7
DC+AC1~4	36.66	37.19	36.98
DC+AC1~5	36.89	37.1	37.04
DC+AC1~6	36.78	37.19	37.06
DC+AC1~7	36.87	37.18	37.11
DC+AC1~8	37.48	37.78	37.21
DC+AC1~9	40.65	39.68	38.81
DC+AC1~10	42.68	40.73	40.12
DC+AC1~11	44.14	41.57	40.46
DC+AC1~12	44.74	41.98	40.59
DC+AC1~13	44.61	41.73	40.55
DC+AC1~14	43.82	41.71	40.56

Table 2 Video layering policy

I-Frame	DCT values
Base layer	DC
Enhancement 1	AC 1 - AC 3
Enhancement 2	AC 4 - AC 8
Enhancement 3	AC 9 - AC 14
P/B Frame	DCT values
Base layer	DC + AC 1- AC 2
Enhancement 1	AC 3 - AC 14

Fig. 5 PSNR with different DCT values

Fig. 6 Wireless link bandwidth estimation

Fig. 7 Proposed vs. Traditional Transmission Scheme

REFERENCES

1. S.Varadarajan, H.Q.Ngo, J.Srivastava, "An adaptive, perception-driven error spreading scheme in continuous media streaming" *Distributed Computing Systems, pp.475-483, Apr. 2000.*

2. J.Y.Pyun, J.J.Shim, S.J.Ko, S.H.Park, "Packet loss resilience for video stream over the Internet" *IEEE Trans. Consumer Electronics Vol.48 No.3.pp.556-563 Aug. 2002.*

3. T.A.Lee, S.G.Chan, Q.Zhang,W.W. Zhu, Y.Q.Zhang, "Allocation of layer bandwidths and FECs for video multicast over wired and wireless networks" *IEEE Trans. Circuits and Syst. Video Technol Vol 12, No.12,pp1059-1070, Dec,2002.*

4. B.Girod, N.Farber, "Feedback-based error control for mobile video transmission" *PROCEEDINGS OF THE IEEE, VOL. 87, NO. 10, pp.1707-1723.Oct. 1999.*

5. M.Podolsky, S.McCanne, M.Vetterli "soft ARQ for layered streaming video" *Tech.rep. UCB/CSD-98-1024, University of California, Computer Science Department, Berkeley, CA, Nov.1998.*

6. N.Y.Yin,S.Q.Li,T.E.Stern "Congestion control for packet voice by selective packet discarding" *IEEE Trans. Communications, Vol.38, No.5, pp.674-683.May.1990.*

7. H.Liu, M.E.Zarki "Performance of H.263 video transmission over wireless channels using hybrid ARQ" *IEEE Journal of Selected Areas in Communications, Vol.15,No.9 pp.1775-1786 Dec.1997.*

8. C.H.Chou, C.W.Chen "A perceptually optimized 3-D subband codec for video communication over wireless channels" *IEEE Transactions on Circuits and Systems for Video Technology Vol.6, No.2pp.386-397 Apr.1996.*

9. S. Wang, H. .Zheng, J.A.Copeland, "An error control design for multimedia wireless networks," *IEEE Vehicular Technology Conference (VTC2000-Spring).Vol, 2 pp.795-799, 2000, Tokyo.*

10. Q. Zhang, S.A. Kasam, "Hybrid ARQ with selective combining for video transmission over wireless channels," *IEEE International Conference on Image Processing, Vol.2, pp. 692-695, 1997.*

11. *Deepak S.Turaga, Tsuhan Chen, "Model-Based Error Concealment for Wireless Video"IEEE Transactions on Circuits and Systems for Video Technology, Vo.l12,No.6, pp.483-495 June 2002.*

12. *Sunghyun Choi,Kang G.Shin, "A Class of Adaptive Hybrid ARQ Schemes for Wireless Links",IEEE Transactions on Vehicular Tech,Vol.50,No.3,pp.777-790May 2001.*

13. *Joan L. Mitchell, William B.Pennebaker,"MPEG video compression standard".*

14. S.Varadarajan, H.Q.Ngo, J.Srivastava, "Error spreading: a perception-driven approach to handling error in continuous media streaming",*IEEE*/ACM*Transaction on Networking Vol.10 Issue Feb 2002 Page 139-152.*

Conditional Replenishment Based Multi-Path Transport and Rate Control for Video Streaming in Ad Hoc Networks

Hua Zhu and Imrich Chlamtac

Erik Jonsson School of Electrical Engineering and Computer Science
The University of Texas at Dallas, Richardson, TX 75083
{zhuhua, chlamtac}@utdallas.edu

Abstract–In this paper, we propose a conditional replenishment based framework for video streaming over ad hoc networks. The main challenges for video streaming in ad hoc networks are: (i) link reliability and (ii) bandwidth scalability. Based on conditional replenishment coding (CRC) technique, the proposed framework provides an integrated solution of multi-path transport (MPT) and dynamic rate control for above challenges.

1. INTRODUCTION

The video streaming over multi-hop mobile ad hoc networks has attracted a lot of attentions recently. Possible applications include: video conferencing in remote places without wireless infrastructure, battlefield visual intelligence, rescue operations, etc. However, there are two major challenges for supporting video streaming over ad hoc networks, which are: (i) link reliability, and (ii) bandwidth scalability.

- *Link Reliability*: Due to the mobility of wireless nodes, the topology of an ad hoc network may change often. It is likely that an established end-to-end route may be broken during the transmission, which may cause interruption of the continuous streaming. To improve the quality of service for video streaming applications over ad hoc networks with unreliable links, Multi-Path Transport (MPT) has been proposed. The main idea of MPT is to transmit video streams over multiple paths so that the video playback will not be disrupted by any individual link failure. Multiple Description Coding (MDC) has been proposed to support MPT so that the original video sequence can be divided into complementary yet independently decodable streams.

- *Bandwidth Scalability*: In current wireless access technologies, such as IEEE 802.11, auto-fallback is a very common technique, which allows the nodes to select the best physical-layer transmission rate for any specific link distance and interference condition. Therefore, the link bandwidth may be very different for different paths. Moreover, for a specific link, the available bandwidth may be varied over time due to traffic congestion, resulting in a significant amount of backlogged packets and unacceptable large delay. The variation of available bandwidth is a critical factor in ad hoc communications, and thus may affect the quality of playback. Besides general techniques to provide scalability by temporal and spatial down-sampling, bandwidth scalability can be provided by Layered Coding (LC), which was originally designed to support bandwidth scalability by allowing the sender/receiver to decide to whether to encode/subscribe any or all enhanced layer streams.

Although MDC and LC are originally designed to handle link reliability and bandwidth scalability, respectively, they are not well suited to handle both issues. It is hard to adjust the encoding bit-rate of a specific stream within MDC. On the other hand, LC does not fit well with MPT, since ELs cannot be decoded without the successful reception of BL.

In this paper, we propose a conditional replenishment coding (CRC) based framework for video streaming over ad hoc networks. In the proposed framework, video sequence can be delivered to the receiver through multiple complementary streams. In addition, CRC-based bandwidth scalability and dynamic rate control can be easily integrated for each individual MPT stream. Therefore this framework is an effective alternative for video streaming over ad hoc networks.

The remainder of the paper is organized as follows. Section 2 provides related work of video streaming over ad hoc networks. Section 3 describes conditional replenishment coding algorithm. Section 4 presents the video streaming framework with MPT and dynamic rate control based on CRC technique. Finally, we conclude the paper in section 5.

2. RELATED WORK

General issues, such as error resilience, error concealment, and error correction, have been studied recently for video streaming over wireless and mobile environments [16][17][18][19]. Forward error correction (FEC) codes and/or various automatic repeat request (ARQ) schemes can be used to provide certain error control for video streaming applications [20].

However, in an ad hoc network, packet loss is mainly due to wireless interference and network saturation. In such case, neither FEC nor ARQ is the effective solution to improve the quality of service of video streaming applications as well as the overall network performance, because both techniques consume additional bandwidth and further worsen the network congestion condition. Therefore, in ad hoc networks, bandwidth scalability coding is one of the highly desirable features for ad hoc video streaming.

Layered Coding is proposed to provide bandwidth scalability [1][2][3][4][5][6][7] to streaming applications in packet network environments where available bandwidth is not know a priori and may constantly changing over time. In layered coding technique, the video source is encoded into a Base Layer (BL) and one or multiple Enhanced Layers (ELs). The reconstructed quality, in general, is related to the number of layers used for decoding and reconstruction. The minimal rendered quality is provided at the receiver as long as the BL is received successfully. The video quality can be improved progressively with the correct rendered ELs at the receiver. The successful transmission of BL is critical in layer coding technique, since the failure of the transmission of BL will cause the failure of the entire application. Typically, unbalanced forward error correction (FEC) or certain automatic repeat request (ARQ) mechanism may be implemented to provide extra protection on the BL. Well-known representatives of LC are MPEG-4 [8] and ITU H.264 (or MPEG-4 AVC) [9].

Another desirable feature for ad hoc video streaming is the support for Multi-Path Transport (MPT) [21][22]. Being originally considered for non-real-time data transmission, MPT has raised many attentions for providing reliable video streaming over ad hoc networks. The main idea of MPT is to transmit video streams over multiple paths so that the video playback will not be disrupted by any individual link failure, which is highly likely to happen in multi-hop ad hoc networks. MPT is a general transmission strategy that can be support by any type of coding technique. However, it has been noticed that layered coding may not be well suited for MPT due to the unbalance of BL and ELs. Multiple Description Coding (MDC) [23][24][25][26][27] has been proposed recently to support MPT. By using MDC technique, the video sequence is encoded into multiple streams that are complementary with each other. The decoder is able to decode any stream independently without any information of other streams. MDC is less vulnerable to the failure of a particular stream, and in such sense, it is more suitable for MPT than LC. However, it is worth noting that MDC usually causes more overhead than LC due to the multiple descriptions for streams.

Performance comparison between LC and MDC can be found in [28][29][30][31].

3. CONDITIONAL REPLENISHMENT CODING (CRC)

Conditional replenishment coding (CRC) has been proposed as a compression technique for taking advantage of the similarity between successive frames in video-telephony or video conferencing where video cameras typically are stationary and scenes usually change slowly [10][11][12][13]. Conditional replenishment is appropriate to distance learning or visual phone applications. The rudimentary algorithm behind conditional replenishment is to transmit the pixel intensity value plus an address for each picture element that is changed by more than a certain threshold since the previous displayed frame.

3.1. Video Format

In the video generated by the conditional replenishment algorithm, all video frames have the same format, all intra-coded without motion vector, as shown in Fig.2. The functions and formats of sequence and GOP are the same as MPEG/H.263 standards. Within the frame, each MB begins with an *MB_ptr* field, which stores the sequence number of corresponding macro-block within the frame. The sequence number starts at zero from the upper-left corner of the image, ends at the lower-right corner. The bit length of the *MB_ptr* field is associated with the *image_width* and *image_height* fields. For example, for a frame size of H.263 CIF (352x288), *MB_ptr* field is 9 bits long, which is sufficient to identify the number of macro-block within the frame; while for the size of QCIF (176x144), *MB_ptr* field is instead 7 bits long. Since the removal of GOB (or slice) logical layer, the *MB_ptr* field is also functional as a resynchronization marker. Error will be limited at the boundary of macro-blocks. Followed by the *MB_ptr* field, are the intra-coded luminance, Cr and Cb blocks of the macro-block identified by the 24bits markers and block type headers. It is not necessary that all MBs are encoded in the frame. Only those MBs encoded by the conditional replenishment algorithm stay in the frame, which is identified by the *MB_ptr* field. Also, it is not necessary that all three types of blocks exist in one MB. For example, if only the luminance blocks is required to be encoded based on the criteria of conditional replenishment algorithm for a specific MB, Cr and Cb blocks will not be in that MB. The blocks in one MB follow the order of lum1, lum2, lum3, lum4, Cr and Cb if they are in the MB, as shown in Fig. 1. Because the removals of motion vector (MV) and motion compensation (MC), the computation complexity of our algorithm is low. Especially at the receiver side, the decoding process becomes much simpler without MC, hence the wireless/mobile terminals can be maintained at lower power levels.

Fig. 1. Hierarchical video format

3.2. Block-based Spatial Conditional Replenishment

At the encoder (or transcoder), the luminance plane of incoming frame is first gridded into 16x16 macro-blocks, and each macro-block is associated with four luminance blocks, one Cr block, and one Cb block. To decide whether or not to encode and transmit any type of block, the algorithm uses the *block distance* as the metric. Block distance is calculated as the following equation.

$$\sum_{k=1}^{n} |x_k - p_k| > T \qquad (1)$$

In the above equation, $(x_1, x_2, \ldots, x_n)$ is the block of pixel of current incoming MB, $(p_1, p_2, \ldots, p_n)$ is the block of pixel of the corresponding MB (i.e. same spatial position) in the previous frame. The *block distance* is the sum of absolute difference between $(x_1, x_2, \ldots, x_n)$ and $(p_1, p_2, \ldots, p_n)$. Three thresholds, T_{LUM}, T_{CR} and T_{CB}, are used to determine whether to transmit the corresponding type of block. In practice, for luminance blocks, we run the algorithm together. Only for those blocks, whose block distance is larger than the threshold, will be encoded and transmitted. If the receiver does not receive any macro-block, or the received data are corrupted because of the wireless channel interference, it uses the corresponding macro-block in the previous re-constructed frame.

3.3. Bandwidth Scalability

Based on CRC, the bit-rate of an encoded stream essentially depends on two factors, i.e., three conditional replenishment thresholds (T_{LUM}, T_{CR} and T_{CB}) and the intensity of motion of the encoded video sequence. Increasing the thresholds decreases the required transmission bit-rate as well as the rendered playback quality of the bit stream. We test eight video sequences to illustrate the relationship between encoding threshold and encoded bit-rate. In the test, encoding thresholds have locked ratio as $T_{LUM}=4T_{CR}=4T_{CB}$. Fig. 2 illustrates the ratio (comparing to the maximum bit-rate at the minimum luminance threshold) of conditional replenishment video bit-rate corresponding to a specific luminance threshold, e.g. for video sequence "claire", the percentage of video bit-rate at luminance threshold = 512 or 1024 is approximately 40% or 25% of the maximum bit-rate respectively. By increasing the luminance threshold, the number of intra-coded macro-blocks of the frames decreases. The rates of decreasing are different between video sequences. This difference can be characterized by a parameter, *motion degree* λ, which describes how much difference (on average) it is between successive transmitted video frames. From Fig 2(b), we know that for small λ (video with less and/or slow motion), the video bit-rate decreases rapidly when the luminance threshold is still small. On the other hand, for large λ (video with more and/or fast motion), the video bit-rate decreases slowly. Given the same luminance threshold, large λ is associated with low average PSNR of the video sequence, as shown in Fig. 3. This is reasonable because there is more information loss for high motion video based on the conditional replenishment algorithm. In Fig. 2, video sequences with more motion are convex shape while less motion video sequences are concave shape. Video sequence with medium motion like *waterski.mpeg* is a watershed. λ is a scene-based parameter, which may not be fixed for the entire video. It can be pre-calculated or estimated, and stored in GOP headers to be used as a parameter to adjust the luminance threshold to obtain any desirable video bit rate. The rate control scheme will calculate and adjust the threshold based on λ. However, the exact video bit-rate can never be achieved because of the variable length of macro-blocks and the subjective and statistical properties of λ. Instead of adjusting video bit-rate in fine granularity way, we use a number of bit-rate levels, each corresponding to one particular threshold. A simple threshold-switching scheme is proposed.

Fig. 2. Video source bit-rate ratio (%) to the maximum bit-rate
(a) enlarged area; (b) completed range of luminance threshold [4, 20000].

Fig. 3. Video sequence average PSNR v.s. luminance threshold

4. MULTI-PATH TRANSPORT WITH DYNAMIC RATE CONTROL

As we mentioned in the introduction, MPT and rate control are two effective techniques to handle the unreliable end-to-end path and constantly changing available bandwidth in wireless ad hoc networks, respectively. MDC technique has been designed to support MPT, while LC technique is more specialized for providing bandwidth scalability and rate control. However, so far, neither technique can support a combination of MPT and rate control very well. Comparing with MDC and LC, CRC technique intrinsically suit for such a combination of MPT and rate control.

4.1. CRC-MPT

In order to generate multiple streams for MPT, the original video sequence can be separated into N streams, V_n's, as follows:

$$V_n = iN + n, \quad where \quad i = 0,1,2,\ldots \quad n = 0,1,\ldots(N-1); \qquad (1)$$

For example, if N=3, the original video sequence is divided into three streams, which is shown in Fig. 4. Each stream is encoded by the CRC algorithm with its own thresholds, which can be dynamically adjusted according to the throughput and delay of the transmission path. To achieve MPT, the receiver

should be able to regenerate the video sequence by any single stream. Given a single stream, V_n, received from a specific path, the receiver can reconstruct two successive frames within the stream, and then extrapolates frames that are not included in V_n as follows:

$$x_{n+i} = \frac{(k-i)x_n + ix_{n+k}}{k}, \quad i = 1...(k-1) \tag{2}$$

Fig. 4. Generating multiple CRC streams

Illustrated by Fig. 5, if the transmission path of V_2 in the previous example is broken, the receiver will still be able to extrapolate V_2 by the reconstructed V_1 and V_3.

Fig. 5. CRC-MPT and decoding processing

4.2. Rate Control

We assume that IEEE 802.11-like CSMA access technology is used as the underlying access technology for the ad hoc network. In such a network, available bandwidth is hard to measure due to the wireless interference and channel contention. Since the saturation throughput may be varied due to the number of contending nodes and the average packet length of the ongoing traffic, the available bandwidth cannot be derived simply as the difference between the physical channel bandwidth and the current throughput.

We estimate the saturation throughput and available bandwidth at each node by modifying the method in [14]. We re-define the idle time of a station as the time period that neither is channel busy nor is MAC queue non-empty. This modification combines sending and receiving throughput to get more accurate available bandwidth condition of the node. The second modification is takes account of multi-rate supports. The bandwidth estimation is distributed, of low

computational complexity, and thereby more appropriate for the situation requiring fast and scalable online estimation of available bandwidth.

The notations involved in this section are shown as the following:

$B_{sat,i}$	Current estimate of the saturation throughput at node i
$B_{ava,i}$	Current estimate of available bandwidth at node i
α_i	Percentage of time that MAC queue at node i is busy (non-empty queue or busy channel) in current measurement period
S	MAC layer payload (upper layer packet) length in current measurement period
t_{ack}	The time of receiving successful MAC layer ACK
t_{enqeue}	The enqueue time of a packet seen by applications
$B_{min,n}$	The minimum available bandwidth for stream V_n

The estimate of saturation throughput at node i is updated every time period T_{hello} as the following,

$$B_{sat,i} = avg(\frac{S}{t_{ack} - t_{enqueue}}); \tag{3}$$

where $avg(x)$ stands for the average of all x's calculated in a period of T_{hello}. $b_{ava,i}$ is affected by multiple factors. The more MAC frames in the queue, the longer queue delay for a specific frame, leading to a larger value of (t_{ack}-$t_{enqueue}$) and a smaller $b_{ava,i}$. The average packet length, number of contention nodes, and channel BER, are all indirectly reflected in (t_{ack}-$t_{enqueue}$). Also, $B_{sat,i}$ is calculated as the average of all packets in T_{hello}, which also has the effect of averaging throughputs of all links proportional to the number of packet having been transmitted on those links. If no packet has been transmitted in the last second, we simply use the last $B_{sat,i}$. T_{hello} and the initial value of $B_{sat,i}$ right after the reboot can be configured by the network operator for nodes together or individually. Nevertheless, the value of T_{hello} should be carefully chosen to avoid frequently fluctuation or inaccuracy of current condition of $B_{sat,i}$. A recommendable value of T_{hello} is 1 second. $B_{ava,i}$ is calculated as the following,

$$B_{ava,i} = (1-\alpha_i)B_{sat,i}; \tag{4}$$

Note that, the available bandwidth at node i, $B_{ava,i}$, is estimated as the minimum of the transient available bandwidth based on periodical measurements and reserved bandwidth at node i, which is the total requested bandwidth of all admitted QoS sessions.

When node i is in the saturation condition, i.e., the queue is never empty, $\alpha_i = 1$, which means no spare bandwidth available. In this case, the available bandwidth in the last T_{hello} interval is zero no matter what the saturation throughput is. On the other hand, if $\alpha_i = 0$, which means no packet being transmitted in the last second, $B_{ava,i}$ is equal to the last calculation of $B_{sat,i}$ minus $B_{res,i}$. As stated in [15], the accuracy of the estimates based on calculated saturation throughput and idle time is satisfactory. More important, this method is robust and feasible to be implemented at mobile nodes with the least modification and impact on the MAC or network layer protocol.

With aforementioned method and additional signaling protocol, each node periodically informs the video streaming source node of its estimated available bandwidth. The source node periodically calculates the minimum available

bandwidth, $B_{min,n}$, along the path of each stream V_n. We have

$$B_{\min,n} = \min\{B_{ava,i}\}$$ (5)

where i's are all the nodes along the path of V_n.

A finite set of conditional replenishment thresholds are pre-selected for each video sequence. An initial threshold can be selected as an application configuration parameter. The source node may initiate streaming with the initial threshold for all the streams. In brief, for any stream V_n, if $B_{min,n}=0$, the source node will increase the encoding threshold for this stream to the next larger value in the set, which decreases the encoding bit-rate as well as PSNR. On the other hand, if and only if $B_{min,n}>\delta>0$ for W consecutive T_{hello}'s, where δ can be pre-configured as well, the source node will decrease the encoding threshold for this stream to the next smaller value, which increases the encoding bit-rate as well as PSNR. In such a way, dynamic rate control may be achieved for each stream of the MPT and this does not require any modification on the CRC decoding process at the receiver.

5. CONLCLUSION

Multiple description coding (MDC) and layered coding (LC) have been proposed recently to support to support video streaming over ad hoc networks, where (i) link reliability and (ii) bandwidth scalability are two major problems. MDC emphasizes on the support of Multi-path transport (MPT) to handle the unreliable ad hoc links. On the other hand, LC mainly focuses on the support of bandwidth scalability and rate control. However, both MDC and LC have limitations as an integrated solution to handle both problems above. In this paper, we proposed a video streaming framework based on conditional replenishment coding. We show that the framework can support both MPT and dynamic rate control. Therefore, it is an effective solution for video streaming over ad hoc networks.

REFERENCES

[1] P. de Cuetos and K.W. Ross, "Adaptive rate control for streaming stored fine-grained scalable video," in Proc. Of NOSSDAV, pp. 3-12, Miami, Florida, May 2002.

[2] P. de Cuetos and K.W. Ross, "Unified framework for optimal video streaming," to appear in INFOCOM 2004, Hong Kong, March 7-11, 2004.

[3] U. Horn, K. Stuhlmuller, M. Link, and B. Girod, "Robust Internet video transmission based on scalable coding and unequal error protection," *Signal Processing: Image Communication*, vol. 15, pp. 77-94, 1999.

[4] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," IEEE Journal on Selected Areas in Communications, Vol. 15, No. 6, August, 1997.

[5] R. Rejaie, D. Estrin, and M. Handley, "Quality adaptation for congestion controlled video playback over the Internet," in Proc. of ACM SIGCOMM, pp. 189-200, Cambridge, Sept., 1999.

[6] R. Rejaie and A. Reibman, "Design issues for layered quality-adaptive Internet video playback," in Proc. of the Workshop on Design Communications, pp.433-451, Taormina, Italy, Sept. 2001.

[7] Y. Wang, Q. Zhu, and Y.-Q. Zhang, "Resource allocation for multimedia streaming over the Internet," IEEE Transactions on Multimedia, vol. 3, no. 3, pp. 339-355, Sept. 2001.

[8] ISO/IEC 14496-1, Information Technology - Generic coding of audio-visual objects - Part 1: System. Proposed Draft Amendment 1, Seul, March 1999.

[9] Joint Video Team (JVT) of ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Joint Committee Draft, May 2002.

[10] E. Amir, S. McCanne, and H. Zhang, "An Application Level Video Gateway," In Proc. ACM Multimedia '95, San Francisco, CA, 1995.

[11] X. Liu, "Rate-constrained Conditional Replenishment Video Coding with Adaptive Change Detection," Project report, December 1, 2000, http://ise0.stanford.edu/class/ee368b/Projects/chiao/

[12] S. McCanne, and V. Jacobson, "vic: a flexible framework for packet video," in Proc. ACM Multimedia '95, Nov. 1995 San Francisco, CA.

[13] H. Zhu, H. Wang, I. Chlamtac and Biao Chen, "Bandwidth scalable source-channel coding for streaming video over wireless access networks," 1st Annual Wireless Networking Symposium (WNCG'03), Oct. 2003.

[14] M. Kazantzidis, M. Gerla, and S.-J. Lee, "Permissible throughput network feedback for adaptive multimedia in AODV MANETs," IEEE Int'l Conference on Communications (ICC'01), Vol. 5, pp. 1352-1356, June 11-14, 2001.

[15] S.H. Shah, K. Chen, and K. Nahrstedt, "Available Bandwidth Estimation in IEEE 802.11-based Wireless Networks," the 1st Bandwidth Estimation Workshop (BEst 2003), San Diego, CA, Dec. 2003.

[16] S. Aramvith, I.M. Pao, and M.T. Sun, "A Rate-Control Scheme for Video Transport over Wireless Channels," IEEE Transaction on Circuits and Systems for Video Technology, pp. 569-580, May 2001.

[17] S. Aramvith, C.W. Lin. S. Roy, and M.T. Sun, "Wireless Video Transport using Conditional Retransmissions and Low-Delay Interleaving," IEEE Transaction on Circuits and Systems for Video Technology, Special issue on Wireless Video, pp. 558-565, June 2002.

[18] S. Dogan, et al., "Error-resilient video transcoding for robust internetworking communications using GPRS," IEEE Transactions on Crcuits and Systems for Video Technology, Vol. 12, No. 6, pp. 453-464, June 2002.

[19] J.Y. Liao, and J. Villasenor, "Adaptive intra block update for robust transmission of H.263," Circuits and Systems for Video Technology, IEEE Transactions on, Vol. 10 Issue 1, pp. 30-35, Feb 2000.

[20] H. Liu, and M.E. Zarki, "Performance of H.263 video transmission over wireless channels using hybrid ARQ," IEEE Journal on Selected Areas in Communications, Vol. 15, No. 9, Dec. 1997.

[21] S. Mao, S. Lin, S.S. Panwar, Y. Wang, and E. Celebi, "Video transport over ad hoc networks: multistream coding with multipath transport," IEEE JSAC special issue on Recent Advances in Wireless Multimedia, Dec 2003.

[22] N. Gogate, D.-M. Chung, S.S. Panwar, and Y. Wang, "Supporting image and video applications in a multihop radio environment using path diversity and multiple description coding," IEEE Trans. on Circuits and Systems for Video Technology, vol. 12, No. 9, pp. 777-792, Sept. 2002.

[23] J. Apostolopoulos, T. Wong, W.-T. Tan, and S. Wee, "On multiple description streaming with content delivery networks," IEEE Infocom 2002.

[24] Y. Liu, P. Salama, and E.J. Delp, "Multiple description scalable coding for error resilient video transmission over packet networks," Image and Video Communications and Processing, Proceedings of SPIE, vol. 5022, pp. 157-168, May 2003.

[25] A.C. Begen, M.U. Demircin, and Y. Altunbasak, "Packet scheduling for multiple description video streaming in multipoint-to-point networks," Proc. IEEE Int. Conf. Communications (ICC), Paris, France, June 2004.

[26] V.K. Goyal, J. Kovačević, R. Arean, and M. Vetterli, "Multiple description transform coding of images," in Proc. IEEE Conf. on Image Proc., Chicago, IL, October 1998.

[27] S. Khan, "Multiple description streaming in peer-to-peer and content delivery networks," in Proc. Australian Telecommunications, Networks and Applications Conference (ATNAC), Melbourne, Dec. 2003.

[28] A.R. Reibman, Y. Wang, X. Qiu, Z. Jiang, and K. Chawla, "Transmission of multiple description and layered video over an EGPRS wireless network," IEEE International Conference on Image Processing (ICIP 2000), Vancouver, BC, Canada, Sept. 2000.

[29] J. Chakareski, S. Han, and B. Girod, "Layered coding vs. multiple descriptions for video streaming over multiple paths," in Proc. ACM Intl. Conf. on Multimedia (MM'03), pp. 422-431, Berkeley, CA, Nov 2003.

[30] Y. Wang, S. Panwar, S. Lin, and S. Mao, "Wireless video transport using path diversity: multiple description vs. layered coding," in the Proc. of the IEEE 2002 International Conference on Image Processing, Rochester, New York, September, 2002.

[31] P.A. Chou, H.J. Wang, and V.N. Padmanabhan, "Layered multiple description coding," in Proc. Int'l Packet Video Workshop 2003, Nantes, France, April 2003.

Performance Enhancement of TFRC in Wireless Ad Hoc Networks

Mingzhe Li, Choong-Soo Lee, Emmanuel Agu, Mark Claypool, and Robert Kinicki

{lmz,clee01,emmanuel,claypool,rek}@cs.wpi.edu

Computer Science Department at Worcester Polytechnic Institute

Worcester, MA 01609, USA

Abstract—**The TCP-Friendly Rate Control (TFRC) is a rate-based transport protocol designed for streaming multimedia applications to provide smooth, low delay and TCP-Friendly packet transmission. However, as TFRC was designed for wired networks, it does not perform well in multihop ad hoc wireless networks. Specifically, MAC layer contention effects, such as retransmission and exponential backoff mislead TFRC's congestion control mechanism, resulting in an inaccurate sending rate adjustment. This paper illustrates that an unmodified TFRC's sending rate overloads the multihop wireless MAC layer, leading to increased round-trip times, higher loss event rates, and lower throughput. We propose an enhancement to TFRC, called RE TFRC, that uses measurements of the current round-trip time and a model of wireless delay to restrict TFRC bitrates from overloading the MAC layer, while retaining the desirable TCP-Friendly characteristics. RE TFRC requires minimal changes to TFRC and no changes to the MAC layer and evaluation of RE TFRC shows substantial improvements over TFRC for some wireless scenarios.**

Keywords—**Wireless, Multimedia, IEEE 802.11, TFRC, Ad Hoc**

I. INTRODUCTION

The Transmission Control Protocol (TCP) is the de facto transport layer protocol used in wireless ad hoc networks. Recent research [1], [2], [3], [4], [5], [6], [7] has shown that TCP can perform poorly in 802.11 wireless networks because many of the TCP mechanisms assume a wired network infrastructure.

Designed to support rate-based streaming multimedia and telephony applications over wired networks, the TCP-Friendly Rate Control (TFRC) protocol [8],[1] faces challenges similar to that of TCP on wireless ad hoc networks. However, to the best of our knowledge, there has been very little TFRC-related performance research done for wireless networks.

At the core of TCP/TRFC's wireless challenge is the wireless Media Access Control (MAC) layer of IEEE 802.11. 802.11 uses Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) and the Request-to-Send/Clear-to-Send (RTS/CTS) mechanism to reduce hidden terminal collisions. However, as the 802.11 MAC layer approaches saturation, contention delays and retransmissions caused by the RTS/CTS mechanism become the major cause of TCP/TFRC performance degradation. This behavior is referred to as RTS/CTS jamming [9] or RTS/CTS-induced congestion [10]. Moreover, since TFRC observes loss events after the MAC contention phase, TFRC is unaware of MAC layer congestion and does not compensate for it. Consequently, TFRC overestimates the maximum sending rate, overloads the MAC layer and exacerbates MAC layer congestion. Eventually, the wireless network reaches a sub-optimal stable state with respect to throughput and round-trip time.

Previous research in TCP performance improvements over wireless ad hoc network includes investigating link breakage and routing failure issues [1], [2], [4], link layer solutions [3], [7], MAC layer solutions [5], and TCP protocol modifications [6]. A few recent papers have focused on methodologies to improve TCP throughput by controlling the total number of packets in flight. Fu *et al* [7] present a link layer approach named Link-RED that reduces MAC layer collisions by limiting TCP's sending window, while Cali *et al* [5] limit TCP window sizes directly. While these previous efforts share a common goal with this research, as window-based approaches they are not applicable to the rate-based TFRC protocol. Furthermore, none of these studies attempt to minimize round-trip times which are of critical concern for interactive multimedia applications.

This investigation focuses on the problem of the mis-interaction between TFRC and the 802.11 MAC layer. Specifically, the objective is to make TFRC aware of RTS/CTS-induced congestion such that it chooses a near-optimal sending rate that avoids MAC layer saturation. A major contribution of this paper is the introduction of a new Rate Estimation (RE) algorithm in TFRC to estimate the saturation capacity of the MAC layer. This involves creating a model for round-trip time during MAC layer saturation and deriving a composite TFRC loss event rate that reflects the current MAC layer congestion level. By limiting the sending rate to a value that is lower than the estimated rate, RE TFRC avoids MAC layer congestion. NS-2 simulation results presented in this report comparing RE TFRC with TFRC indicate a 5% to 43% reduction in round-trip times, a 8% to 75% reduction in the loss event rate, and up to 7% improvement in overall throughput. Given that TFRC is intended for multimedia applications, large delay reductions with slight throughput improvements with RE TFRC implies this scheme can improve performance for streaming flows in wireless networks.

The rest of this paper is organized as follows: Section II analyzes TFRC behavior in wireless ad hoc networks and investigates the relationship between performance and a constrained sending rate; Section III details the RE TFRC algorithm; Section IV evaluates the RE TFRC algorithm in several wireless ad hoc network scenarios; Section V summarizes our conclusions and presents possible future work.

II. TFRC PERFORMANCE ANALYSIS

While the RTS/CTS collision avoidance mechanism reduces hidden terminal collisions in the 802.11 MAC layer, repeated

---

[1]The Datagram Congestion Control Protocol (DCCP) has proposed to use TFRC as its congestion control mechanism. See http://www.ietf.cnri.reston.va.us/html.charters/dccp-charter.html.

Fig. 1. Simulation topology

Fig. 2. Offered load and throughput versus the constrained sending rate

MAC layer backoffs and retransmissions can lead to sub-optimal transport layer performance in wireless environments. Fu *et al* [7] demonstrates the impact of hidden terminals on the transport layer protocol.

In TCP-Friendly transport protocols, the sender responds to network congestion by adjusting its transmission rate or window size based on packet loss and round-trip time information gathered from the network. However, since these metrics also include the effects of the RTS/CTS mechanism, MAC layer backoffs and retransmissions when operating on a wireless network, they cannot be used effectively as congestion indicators.

This section shows simulations (via NS-2 [11]) of the TFRC protocol with a constrained sending rate to explore the relationship between the TFRC throughput, round-trip time and loss event rate in multihop 802.11 ad hoc networks. The goal is a meaningful characterization of the effects of 802.11 on TFRC and to gain insight into adapting TFRC's sending rate when transmitting streaming flows over wireless LANs.

A. Simulation Environment

To simplify the analysis of TFRC performance, the chain topology shown in Figure 1 with the default NS-2 802.11 parameter settings is used in a series of simulations. For the simulation results presented, all nodes are immobile, the distance between nodes is set to 200 meters, the transmission range is 250 meters, and the wireless channel capacity is 2 Mbps (the NS-2 default setting).

The throughput equation used for TFRC [8] is a version of the throughput equation for a conformant TCP Reno flow:

$$X = \frac{s}{r\sqrt{\frac{2bp}{3}} + 3p(t_{rto}(1 + 32p^2)\sqrt{\frac{3bp}{8}})} \quad (1)$$

where X is the transmission rate in bytes/second, s is the packet size in bytes, r is the round-trip time in seconds, p is the loss event rate which is the number of loss events as a fraction of the number of packets transmitted ($0 \leq p \leq 1$), t_{rto} is the TCP retransmission timeout value in seconds, and b is the number of packets acknowledged by a single TCP acknowledgment.

Research in [12] establishes the maximum throughput for an ad hoc network to be approximately $\frac{1}{5}$ to $\frac{1}{7}$ of the link capacity. Our simulations show the maximum achievable throughput for TFRC over a multihop wireless network to be significantly lower than the line capacity. Since Bianchi [13] showed that 802.11 MAC layer throughput decreases when offered load exceeds the saturation threshold, this lower-than-expected throughput can be attributed to the RTS/CTS congestion [10] that occurs when the MAC layer becomes saturated.

As expected, all transport layer packet loss in our simulations are caused by MAC layer contention and frame drops when no transport layer congestion is induced. While TFRC is supposed to react to transport layer losses, it is not tuned to respond to MAC layer congestion, and hence does not reduce its sending rate appropriately. The MAC layer congestion also causes an increase in MAC contention time and end-to-end round-trip time. The details of this performance are not given here due to lack of space, but the complete analysis can be found in [14].

B. Rate Constrained Simulations

To clarify the behavior of TFRC in overloading the 802.11 MAC layer in a multihop environment, we modified NS-2 to provide a version of TFRC that had a *manually constrained* sending rate. Figure 2 shows the TFRC offered load and throughput as the constrained sending rate is varied for a seven hop network. As the constrained rate increases, offered load and throughput increase linearly until a divergence occurs at approximately 300 Kbps. Beyond this point, increasing the constrained TFRC rate yields reduced throughput. The observed gap between offered load and throughput at high TFRC rates is due to lost packets.

Figure 3 shows a sharp increase in MAC layer losses starting at about 300 Kbps, as the constrained sending rate increases. Graphs of TFRC's round-trip time and loss event rate (available in [14]) also show a sharp increase at about 300 Kbps. Additional simulations run with typical wireless bit error rates still show constraining the rate of TFRC under 300 Kbps achieves a round-trip time lower than that of unconstrained TFRC.

In rate-constrained mode, TFRC uses the minimum of the constrained rate and the TCP-Friendly rate to control the sending rate. Figure 4 depicts the relationship between the average sending rate, the constrained sending rate and the computed TCP-Friendly rate. Above 300 Kbps, TFRC uses the TCP-Friendly rate to control the sending rate. This implies that TFRC does not keep the sending rate below the MAC saturation point on wireless LANs. Namely, TFRC will select a sub-optimal transmission rate on wireless LANs when the MAC layer is saturated. Thus, the next section presents a new algorithm designed to constrain TFRC and avoid saturating the MAC layer on 802.11 wireless networks.

Fig. 3. MAC layer drop fraction versus constrained sending rate

Fig. 4. Average sending rate and TCP-Friendly rate versus constrained sending rate

Fig. 5. Throughput versus loss event rate (p) for different round-trip times

III. ENHANCING TFRC PERFORMANCE

A. Rate Estimation

From the results in Section II, when unconstrained, TFRC produces an offered load that is above the rate sustainable by the multi-hop 802.11 MAC layer. The MAC layer then suffers from multiple frame retransmissions that increase the round-trip time. Although TFRC eventually receives some packet loss notification because of the frame retransmissions, these packet losses arrive too late for TFRC to curtail its offered load below the saturation point of the MAC layer. To be able to adjust its sending rate to below the MAC layer saturation point, TFRC needs to determine the loss event rate (p) that corresponds to the MAC layer congestion point.

We propose to enhance the performance of TFRC based on aspects of TCP Westwood [15], a TCP variant designed to perform well over wireless links. TCP Westwood uses a bitrate estimation algorithm based on the minimum observed round-trip time and acknowledgment rate to compute a window threshold for TCP. Whenever there is congestion, the TCP congestion window is set equal to the window capable of producing the bitrate estimate (B) assuming no queuing delay (i.e. $window = B \times r_{min}$). We propose a similar algorithm to estimate the MAC layer saturation bitrate. However, instead of us-

ing r_{min}, we use r_{opt}, which represents the minimum round-trip time during MAC layer saturation. r_{opt} is used instead of r_{min} because when the maximum sustainable throughput in the MAC layer is achieved, there is a small queue at individual nodes of a multihop flow. TFRC has a built-in function for estimating the receiving rate, R, which is used as a basis for our modifications.

As described in Section II, TFRC's sending rate is not constrained by a window size but rather by the computed TCP-Friendly rate. TFRC uses an equation based on TCP throughput to compute an estimated TCP-Friendly sending rate, which is a function of the round-trip time (r), loss event rate (p), packet size and time out value (rto). Assuming a fixed packet size (typically around the network MTU) and the default value of $rto = 4 \times r$ (as set in [8]), we simplify the TCP-Friendly bitrate equation in Equation 1 and derive a function for p:

$$X = f(r, p) \qquad (2)$$
$$p = \overline{f}(r, X) \qquad (3)$$

Therefore, the equivalent TFRC loss event rate (p') can be estimated using the inverse function $\overline{f}(r, X)$, and then p' and the current round-trip time measured by TFRC (r_{cur}) can be used to estimate the optimum sending rate (R') that will just saturate the MAC layer:

$$p' = \overline{f}(r_{opt}, R)$$
$$R' = f(r_{cur}, p')$$

Figure 5 depicts the relationship between TCP-Friendly bitrate and loss event rate, where each curve is the TCP-Friendly bitrate for a particular round-trip time.

B. Round-Trip Time Modeling

Realizing the benefits of the proposed TFRC enhancements requires a mechanism to compute r_{opt}, the minimum round-trip time during MAC layer saturation. Previous research on delay modeling of 802.11 networks [16] shows that the average delay (the service time) of a single hop ad hoc network at saturation can be modeled by:

$$\overline{T} = \overline{T}_B + t_s \qquad (4)$$

	DSSS	FHSS
W_{min}	32	16
W_{max}	1024	1024
MAC header	34 bytes	34 bytes
Phy header	24 bytes	16 bytes
ACK	38 bytes	30 bytes
CTS	38 bytes	30 bytes
RTS	44 bytes	36 bytes
Slot time	20 μsec	50 μsec
SIFS	10 μsec	28 μsec
DIFS	50 μsec	128 μsec

where t_s is the time required to to successfully transmit a packet and $\overline{T}_B$ is the average MAC layer back-off time:

$$\overline{T}_B = \frac{\alpha(W_{min} - 1)}{2q} + \frac{(1-q)}{q} t_c \qquad (5)$$

Here, α is the average back-off step size, W_{min} is the initial contention window size, q is the probability of successful transmission, and t_c is the time wasted during a single collision. W_{min} is a physical layer parameter (with a default of 32 for Direct-Sequence Spread Spectrum (DSSS)), while [16] assumes α and q are computable as functions of the number of nodes (n) in the network. t_s and t_c are constants for fixed size packets and can be computed using:

$$\begin{aligned} t_s = \ & rts + sifs + \delta + cts + sifs + \delta + H \\ & + E\{P\} + sifs + \delta + ack + difs + \delta \qquad (6) \end{aligned}$$

$$t_c = rts + difs + \delta \qquad (7)$$

where rts, cts, ack, H and $E\{P\}$ are the transmission times of RTS, CTS, ACK, packet header (physical layer plus MAC layer) and data packets, respectively, and $E\{P\} = P$ for a fixed packet size. δ is the propagation delay. $sifs$ (Short Interframe Space), $difs$ (Distributed Interframe Space) and other specific values for DSSS and FHSS are listed in Table I.

Therefore, given the physical network type and the number of nodes in the network, Equation 6 can be used to estimate the average service time to obtain the delay under MAC saturation conditions.

To extend this model in multi-hop wireless networks, we assume that under saturation conditions, the traffic at each hop is independent, which allows a multi-hop ad hoc network to be divided conceptually into multiple independent, single-hop networks. By using the model on each of the single hops, a cumulative delay for the multi-hop network can be estimated. By assuming RTS/CTS solves the hidden terminal problem in applying the single-hop analysis, we do not need to consider the interference from other nodes outside the transmission range. The N-hop chain network can then be divided into $N - 2$ single-hop networks with four nodes and two single-hop networks with three nodes at the source and destination. The round-trip time at the transport layer (such as in TFRC) is estimated by measuring the time elapsed between sending a data packet and receiving the acknowledgment. Therefore, the round-trip time can be computed as:

Fig. 6. Estimating the optimum round-trip time

$$\begin{aligned} \overline{r}(N) \quad = \quad & \Sigma_{i=0}^{N} T_{data_i} + \Sigma_{i=0}^{N} T_{ack_i} \\ \approx \quad & 2 \times \overline{T}_{data}(3) + (N-2)\overline{T}_{data}(4) \\ & + 2 \times \overline{T}_{ack}(3) + (N-2)\overline{T}_{ack}(4) \qquad (8) \end{aligned}$$

Based on the model, the round-trip time ($r(N)$) from Equation 8 assumes saturation of the MAC layer and can therefore be used for r_{opt} for an N hop ad hoc wireless network.

Figure 6 depicts the round-trip time estimate from this model and the round-trip time obtained by TFRC during simulation. TFRC provides an offered load above the MAC saturation level which causes the round-trip time to increase beyond r_{opt}.

C. Algorithm Summary and Implementation

By combining the loss event rate estimation algorithm from TFRC and the extended round-trip time model, we provide a complete rate estimation algorithm for TFRC, shown in Figure 7.

```
on receiving ack
1.  if (not slowstart)
2.        // compute original TCP-Friendly rate
      X = f(r_cur, ack.p)
3.        // choose modeled RTT or smallest measured RTT
      r_opt = max(r(N), min([Sliding Window]))
4.        // compute new loss event rate given RTT
      p' = f̄(r_opt, R)
5.        // compute new TCP-Friendly
      R' = f(r_cur, p')
6.        // use original rate if new rate is larger
      R' = min(X, R')
7.        // if there is a rate change, do so incrementally
      if (rate_cur > R')
            decrease_rate()
        else
            increase_rate(p')
8.  end if
```

Fig. 7. The rate estimation algorithm for TFRC (RE TFRC)

To make the implementation of the RE TFRC algorithm in Figure 7 more stable and adaptive, a few enhancements were needed. First, at line 2 and 6 of the algorithm, the TCP-Friendly sending rate computation that is fundamental to TFRC is used to ensure appropriate response to transport layer congestion. Second, note as the number of hops or flows increases, the round-

Fig. 8. Distribution of RTS retransmissions

Fig. 9. MAC layer drop fraction versus number of hops

trip time curve of r_{cur} will shift up and go above the r_{opt} curve in Figure 5. In this case, the r_{opt} curve no longer represents saturation. Hence, r_{min} should be used in place of r_{opt}. Therefore, line 3, determines the minimum round-trip time in a sliding window interval and uses the larger of the computed round-trip time ($r(N)$) and the window value to estimate p' in these situations.

IV. PERFORMANCE EVALUATION

The goals of Rate Estimation TFRC (RE TFRC) are to reduce MAC layer congestion, reduce TFRC loss event rate and average round-trip time, and improve throughput without changing the MAC layer protocol. This section evaluates RE TFRC using NS-2 simulations with the same wireless chain topology used in Section II. The first step is a detailed analysis of RE TFRC performance in a seven hop simulation. This is followed by simulation experiment results where the the number of hops is varied from 4 to 15 and other simulations where three flows generate the offered load. The section concludes with a study of the behavior of the RE TFRC in typical Bit Error Rate (BER) network environment.

A. Performance Improvement

A seven hop chain topology was used to compare a standard TFRC implementation against the Rate Estimation TFRC (RE TFRC) algorithm. Since the RTS backoff mechanism drops an RTS frame after seven consecutive collisions, this event represents a packet loss as seen by TFRC. Figure 8 presents the Cumulative Density Function (CDF) for RTS retransmissions for the two simulations. The x-axis is the number of RTS contention backoffs from 0 to 7 where 0 implies no collisions and 7 means TFRC will see this as a loss event. Figure 8 shows that TFRC has a 89% chance of not having to retransmit an RTS while RE TFRC, has a 93% chance of not having to retransmit an RTS, so RE TFRC will experience less backoff delay. Since the backoff algorithm causes exponential growth in backoff delay with an increase in the number of retransmissions, the seemingly small differences in the CDF curves represent significant changes in the contention delay. The reduced collisions result in a lower loss event rate and round-trip time and a smoother sending rate for RE TFRC (see [14]).

Fig. 10. Average round-trip time versus number of hops

B. Multi-hop Performance Evaluation

The next evaluation of RE TFRC involves varying the number of wireless hops from 4 to 15. Figure 9 shows the improvement of MAC layer loss rate for RE TFRC. The MAC layer drop ratio is reduced by between 13% to 66% compared to TFRC.

Figure 10 demonstrates that the round-trip time of RE TFRC is 5% to 40% lower than that of TFRC, and Figure 11 shows that the RE TFRC loss event rate is 8% to 55% less than that of TFRC.

From the results of multi-hop simulations, RE TFRC also shows up to 5% throughput improvement over TFRC when the number of hops is increased from 5 to 15. (see [14]).

C. Multi-flow Performance Evaluation

This section considers situations where three flows are providing the offered load. Table II shows RE TFRC reduces the MAC layer drop rate, TFRC loss event rate and average round-trip times significantly. However, RE TFRC has little effect on throughput in the multi-flows scenarios (see [14]). The "-" in the table means the difference is less than 1%.

D. Bit Error Rate Evaluation

The Bit Error Rate (BER) in wireless networks is usually higher than in wired networks. Typical BER ranges from 10^{-6}

Fig. 11. Average loss event rate versus number of hops

TABLE II

RE TFRC IMPROVEMENT FOR MULTI-FLOW ENVIRONMENT

Hops	7-hop	15-hop
MAC drop fraction reduction	50%	85%
RTT reduction	29%	43%
Loss rate reduction	53%	75%
Throughput improvement	-	-

to 10^{-4} were used to evaluate RE TFRC. A 7-hop wireless network topology with single flow simulation is used to demonstrate the effects of various BER on RE TFRC performance. As shown in Table III, RE TFRC performs considerably better over most of metrics over a wide range of BER.

V. CONCLUSIONS

This paper presents a new algorithm, Rate Estimation (RE) TFRC, designed to enhance TFRC performance in wireless Ad Hoc network environment. RE TFRC estimates a sending rate using an optimal round-trip time based on the network topology and equivalent loss event rate. The optimal round-trip time is estimated by modeling multi-hop contention delay and service time and the equivalent loss event rate is estimated using the inverse TCP Friendly rate equation with the optimal round-trip time. The basic idea is to infer the lower-layer MAC layer jamming in the upper layer TFRC to make it aware of lower layer congestion and reduce the jamming effects.

Our simulations confirm that the RE algorithm can significantly enhance TFRC performance with a reduced round trip

TABLE III

RE TFRC IMPROVEMENT FOR VARIOUS BER

BER	10^{-6}	10^{-5}	10^{-4}
MAC drop fraction reduction	77%	25%	12%
RTT reduction	39%	32%	14%
Loss rate reduction	55%	45%	29%
Throughput improvement	7%	4%	-

time and loss event rate while still providing the same or better throughput than regular TFRC. As TFRC was designed for streaming multimedia and telephony applications, large delay and loss rate reductions with slight throughput improvements implies this scheme can improve performance for streaming flows in wireless networks.

Our current ongoing RE TFRC research is focused on refining the sending rate, loss event rate and round-trip time estimation algorithm. The goal is a more robust RE algorithm that will adapt and remain stable even when the wireless nodes become mobile and the topologies are more complex. Other potential RE enhancements include incorporating particular characteristics of TFRC applications, such as streaming multimedia, in further optimizing performance. Ultimately, the objective is to implement TFRC with wireless extensions on an operational ad hoc wireless network testbed and empirically evaluate its performance.

REFERENCES

[1] Gavin Holland and Nitin H. Vaidya, "Analysis of TCP Performance Over Mobile Ad Hoc Networks," in *Proceedings of IEEE/ACM MOBICOM*, Aug. 1999, pp. 219–230.
[2] Venkatesh Ramarathinam and Miguel A. Labrador, "Performance Analysis of TCP over Static Ad Hoc Wireless Networks," in *Proceedings of the ISCA PDCS*, Sept. 2000, pp. 410–415.
[3] M. Gerla, K. Tang, and R. Bagrodia, "TCP Performance in Wireless Multihop Networks," in *Proceedings of IEEE WMCSA*, Feb. 1999.
[4] Kartik Chandran, Sudarshan Raghunathan, S. Venkatesan, and Ravi Prakash, "A Feedback Based Scheme for Improving TCP Performance in Ad-Hoc Wireless Networks," in *Proceedings of ICDCS*, 1997, pp. 472–479.
[5] Frederico Cali, Marco Conti, and Enrico Gregori, "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit," *IEEE/ACM Transactions on Networks*, vol. 8, no. 6, pp. 785–799, 2000.
[6] Kai Chen, Yuan Xue, and Klara Nahrstedt, "On Setting TCP's Congestion Window Limit in Mobile Ad Hoc Networks," in *Proceedings of IEEE ICC*, May 2003.
[7] Zhenghua Fu, Petros Zerfos, Haiyun Luo, Songwu Lu, Lixia Zhang, and Mario Gerla, "The Impact of Multihop Wireless Channel on TCP Throughput and Loss," in *Proceedings of IEEE Infocom*, Mar. 2003.
[8] Sally Floyd, Mark Handley, Jitendra Padhye, and Jorg Widmer, "Equation-Based Congestion Control for Unicast Applications," in *Proceedings of ACM SIGCOMM Conference*, Stockholm, Sweden, Aug. 2000, pp. 43 – 56.
[9] C. Ware, T. Wysocki, and J.F. Chicharo, "On the Hidden Terminal Jamming Problem in IEEE 802.11 Mobile Ad Hoc Networks," in *Proceedings of IEEE ICC*, 2001.
[10] Saikat Ray, Jeffrey Carruthers, and David Starobinski, "RTS/CTS-induced Congestion in Ad-Hoc Wireless LANs," in *Proceedings of IEEE WCNC*, Mar. 2003, pp. 1516–1521.
[11] Universiy of California Berkeley, "The Network Simulator - ns-2," [Online] http://www.isi.edu/nsnam/ns/.
[12] Jinyang Li, Charles Blake, Douglas De Couto, Hu Imm Lee, and Robert Morris, "Capacity of Ad Hoc Wireless Networks," in *Proceedings of ACM/IEEE MobiCom*, 2001, pp. 61–69.
[13] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications*, Mar. 2000.
[14] Mingzhe Li, Emmanuel Agu, Mark Claypool, and Robert Kinicki, "Performance Enhancement of TFRC in Wireless Networks," Tech. Rep. WPI-CS-TR-03-33, CS Department, Worcester Polytechnic Institute, Dec. 2003.
[15] Ren Wang, Massimo Valla, M. Y. Sanadidi, and Mario Gerla, "Adaptive Bandwidth Share Estimation in TCP Westwood," in *Proceedings of IEEE Globecom*, Nov. 2002.
[16] M. Carvalho and J.J. Garcia-Luna-Aceves, "Delay Analysis of IEEE 802.11 in Single-Hop Networks," in *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, Nov. 2003.

Communication Protocol for Atomic and Causally Ordered Delivery of Multimedia Messages

Satoshi Itaya, Tomoya Enokido, and Makoto Takizawa

Dept. of Computers and Systems Engineering
Tokyo Denki University, Japan
{itaya, eno, taki}@takilab.k.dendai.ac.jp

Abstract

A large-scale network is composed of various types of communication channels. Here, each communication channel supports Quality of Service (QoS) which may be different from others. In group communications, each process sends a message to multiple processes while receiving messages from multiple processes. In addition, messages are required to be causally delivered. A process supporting enough QoS, cannot deliver the messages if another process supporting lower QoS does not receive the messages. Thus, multimedia data cannot be delivered to processes so as to satisfy the real-time constraint if a slower process is included in a group. In this paper, we discuss group communication protocols by which multimedia messages can be delivered to a process with synchronization and QoS constraints.

1. Introduction

In large-scale distributed systems like Peer-to-Peer (P2P) networks [10], large number of processes are cooperating by exchanging messages. In distributed multimedia applications like teleconference and Video on Denamd (VoD), multimedia data is exchanged among multiple processes in networks like Gigabit and ATM networks [1, 4]. Multimedia communication protocols like RTP [11] and RSVP [3] are developed so far, by which a large volume of multimedia data can be efficiently transmitted to one or more than one process. One-to-one and one-to-many communication protocols to satisfy Quality of Service (QoS) requirement like delay time, bandwidth, and loss ratio are discussed in papers [1, 14].

In group communications [5, 12–14], a collection of multiple processes are cooperating by exchanging messages. Here, messages have to be *causally delivered* to multiple destination processes in the group [2]. A message m_1 *causally precedes* another message m_2 if and only if (*iff*) a sending event of m_1 *happens before* a sending event of m_2 [7]. Various types of the group communication protocols which support a group of multiple processes with the causally ordered delivery of messages have been discussed [2, 6, 9]. In a *centralized* approach to realizing group communication, there is one controller process through which processes are exchanging messages. The centralized approach is not suited to realize real-time multimedia applications including multiple processes, especially distributed in a wide-area network due to longer delay time. We take a fully *distributed* approach where every process directly sends a message to destination processes in a group of processes in order to satisfy real-time constraints of multimedia data. Each process receives messages from multiple processes. The process has to causally order messages received from multiple processes by itself. In addition, the process is required to send a message to each destination process while receiving messages from multiple processes so that QoS requirement is satisfied.

Since each communication channel between processes may support different QoS due to distance and congestions, a message sent by a process may not arrive at every destination process at a same time. Every destination process cannot deliver messages received until the slowest destination process receives the messages. Thus, time constraint is not satisfied if a group includes a less-qualified process. We discuss requirements of group communication like real-time constraint in addition to the atomic and causally ordered delivery of messages. We discuss how to synchronize transmissions of messages to multiple processes and receipts of messages from multiple processes so as to satisfy the group communication requirements in the fully distributed group.

In section 2, we present a system model. In section 3, we discuss a model for transmitting and receiving multimedia messages in a group. In section 4, we discuss the atomicity and causality of multimedia messages in group communication.

2. System Model

2.1 Channel

The distributed application is composed of multiple application processes AP_1, ..., $AP_n(n{>}1)$ which are supported by system processes p_1, ..., $p_n(n{>}1)$. A collection of multiple peer processes cooperating is referred to as *group G*. The group communication service is provided for the application processes by multiple system processes. The network is modeled to be a collection of logical communication channels. Processes communicate with each other by taking usage of communication service supported by channels. There is a channel $C_{ij} = \langle p_i, p_j \rangle$ between every pair of processes p_i and p_j in the group G. Each channel $\langle p_i, p_j \rangle$ is *bidirectional* and *synchronous*. Each channel is realized by communication service of the underlying network.

An application process AP_i sends a message m to one or more than one destination process in a group. Let $dst(m)$ denote a collection of destination processes of a message m, which is a subset of a group. Let $src(m)$ show a source process which sends a message m. A message m is transmitted from a process AP_i to every destination process AP_j

in $dst(m)$ via a channel $C_{ij} = \langle p_i, p_j \rangle$. A message m sent by an application process is decomposed into a sequence $pkt(m)$ ($l \geq 1$) of *packets* $\langle t_1, \ldots, t_l \rangle$ [Figure 1]. A packet is a unit of data transmission in a network. Packets are transmitted through a channel of the network. The process p_i transmits a packet sequence $pkt(m)$ to every destination process p_j of the message m via a channel $C_{ij} = \langle p_i, p_j \rangle$. A destination process p_j receives *packets* sent by the process p_i through the channel C_{ij} and assembles the packets into the message m. Then, the message m is delivered to the destination application process AP_j.

message
Application layer
System layer
Channel
Network layer
packets
t_1 t_2 t_l
$AP_1, ..., AP_n$: Application process
$P_1, ..., P_n$: System process

Figure 1. System model.

2.2 QoS

Each channel $\langle p_i, p_j \rangle$ supports Quality of Service (QoS), which is denoted by $Q(\langle p_i, p_j \rangle)$ or Q_{ij}. QoS is characterized by parameters; bandwidth(bw) [bps], packet loss ratio(pl) [%], and delay(dl) [msec]. Each QoS instance is a tuple of values $\langle v_1, \ldots, v_m \rangle$ where each v_i is a value of QoS parameter q_i ($i = 1, \ldots, m$). Let $\mathbf{Q}$ be a set of QoS parameters $q_1, \ldots, q_m$. Let A and B be QoS instances $\langle a_1, \ldots, a_m \rangle$ and $\langle b_1, \ldots, b_m \rangle$, respectively. Each QoS value a_i of the parameter q_i in the QoS instance A is shown by $q_i[A]$. If a_i is better than b_i ($a_i \succ b_i$) for every parameter q_i, A *precedes* B ($A \succ B$). $A \succeq B$ iff $A \succ B$ or $A = B$. A pair of QoS instances A and B are *uncomparable* ($A \parallel B$) iff neither $A \succeq B$ nor $A \preceq B$. A *preference* relation "$\rightarrow$" is a partially ordered relation on QoS parameters $q_1, \ldots, q_m$, i.e. $\rightarrow \subseteq \mathbf{Q}^2$. "$q_i \rightarrow_t q_j$" shows that a process p_t prefers a QoS parameter q_i to another parameter q_j. "$q_i \rightarrow q_j$" shows "$q_i \rightarrow_t q_j$" for some process p_t in a group. The preference relations "$\rightarrow_t$" and "$\rightarrow$" are asymmetric. For example, $bw \rightarrow_t pl$ if bandwidth (bw) is more significant than packet loss ratio (pl) for a process p_t. For every pair of QoS parameters q_i and q_j, $q_i \cup q_j$ and $q_i \cap q_j$ show least upper bound (lub) and greatest lower bound (glb) of q_i and q_j, respectively, with respect to the preference relation "$\rightarrow$". Let P_t be a partially ordered set $\langle \mathbf{Q}, \rightarrow_t \rangle$, named *preference* of a process p_t. $\langle \mathbf{Q}, \rightarrow \rangle$ is $\langle \mathbf{Q}, \underset{t=1,\ldots,n}{\cup} \rightarrow_t \rangle$. For example, $\mathbf{Q} = \{bw, pl, dl\}$ and $bw \rightarrow_t pl$ and $dl \rightarrow_t pl$ for a process p_t. A preference P_t is $\langle \mathbf{Q}, \{bw \rightarrow_t pl, dl \rightarrow_t pl\} \rangle$. A QoS parameter q is referred to as *maximal* in $\mathbf{Q}$ iff there is no QoS parameter q' in $\mathbf{Q}$ such that $q' \rightarrow q$.

Let A and B be QoS instances $\langle 128[\text{Mbps}], 100[\text{msec}], 0.1[\%]\rangle$ and $\langle 64[\text{Mbps}], 100[\text{msec}], 0.01[\%]\rangle$, respectively, for QoS parameters $\mathbf{Q} = \langle bw, pl, dl \rangle$. Here, $64 \succ 128[\text{Mbps}]$, $100 = 100[\text{msec}]$, and $0.1 \succ 0.01[\%]$. If the

bandwidth (bw) and delay time (dl) are more significant than the packet loss ratio (pl) in an application, $bw \rightarrow pl$ and $dl \rightarrow pl$ in a preference P of the application. Here, bw and dl are maximal. The QoS instance A is *more preferable than* B with respect to the preference P ($A \succ_P B$) since the bandwidth and delay time of A are better than B although the packet loss ratio of B is better than A. Let $\mathbf{Q}(A)$ and $\mathbf{Q}(B)$ show sets of QoS parameters of QoS instances A and B, respectively. A preference relation "$A \succ_P B$" with respect to a preference P holds iff

1. For every maximal QoS parameter q_i in $\mathbf{Q}$, $a_i \succeq_P b_i$.
2. If $a_i = b_i$ for every maximal QoS parameter q_i in $\mathbf{Q}$, $A' \succeq_P B'$ for QoS instances A' and B' obtained by removing QoS instances of maximal QoS parameters.

QoS requirement R_i of a process p_t is given a pair of parameters $MaxQ_t$ and $MinQ_t$, which show maximal and minimal QoS required by p_t, respectively. A process p_t is referred to as *satisfy* QoS requirement R_t if $MaxQ_t \succeq_t Q_t \succeq_t MinQ_t$ for every QoS instance Q_t taken by the process p_t. $MaxQ_t$ shows the most preferable QoS which can be realized in the implementation and $MinQ_t$ indicates the least preferable QoS of the process p_t. Suppose a process p_s sends messages to another process p_t. We discuss how much QoS instance a pair of processes p_s and p_t agree on in order to communicate with one another. Let Q be QoS instance. Q is referred to as *satisfiable* for a pair of QoS requirements R_s and R_t of processes p_s and p_t, respectively, if $MaxQ_s \succeq_s Q \succeq_s MinQ_s$ and $MaxQ_t \succeq_t Q \succeq_t MinQ_t$. Q is *maximally satisfiable* for the requirements R_s and R_t i.e. $Q = R_s \cup R_t$ iff Q is satisfiable for R_s and R_t and there is no QoS instance Q' satisfiable for R_s and R_t such that $Q' \succeq_s Q$ and $Q' \succeq_t Q$. Similarly, we define a minimally satisfiable QoS instance ($R_s \cap R_t$) for R_s and R_t.

3. Data Communication Model

3.1 Transmission

A process p_i sends packets $t_1, \ldots, t_l$ ($l \geq 1$) of a message m to every destination process p_{ij} in $dst(m)$ ($j = 1, \ldots, k_i$). There are following ways to transmit a packet sequence $pkt(m)$ ($= \langle t_1, \ldots, t_l \rangle$) to all the destination processes [Figure 2]:

1. *Synchronous* transmission: A process p_i sends each packet t_h to every destination process p_{ij} through a channel C_{ij}. Here, each packet t_h is sent in each channel C_{ij} after t_{h-1} is sent in every channel ($h=1, \ldots, l$).

2. *Partial-synchronous* transmission: A process p_i sends a number n_j of packets to a process p_{ij} while sending a number n_h of packets to another process p_{ih}. Here, the ratio $n_1 : \cdots : n_{k_i}$ is the transmission synchronization ratio of the process p_i to the processes $p_{i1}, \ldots, p_{ik_i}$.

3. *Asynchronous* transmission: A process p_i sends a sequence $pkt(m)$ of packets through each channel independently of the other channels.

Figure 2. Types of transmission.

$\mathbf{Ssnd}(t, C)$ shows a procedure to send a packet t through a channel C. $\mathbf{Asnd}(T, C)$ shows a procedure to send a sequence T of packets t_1, ..., t_l through a channel C, i.e. **for** $h = 1, \ldots, l$ **do** { $\mathbf{Ssnd}(t_h, C)$;}. A packet sequence T is decomposed into a sequence $\langle T_j^1, \ldots, T_j^{g_j} \rangle$ of segments for a process p_{ij}. A segment is a subsequence of the packets. Here, each segment is composed of the same number of packets. Let $|T_j^k|$ show the number of packets in a segment T_j^k. The ratio $|T_1^1| : \cdots : |T_{k_i}^1|$ shows the transmission synchronization ratio for the destination processes p_{i1}, ..., p_{ik_i}. $\mathbf{PSsnd}(T_j^k, C_{ij})$ shows a procedure to send a segment T_j^k of packets in a channel C_{ij}. If the transmission synchronization ratio is $1:\cdots:1$, a process p_i sends packets at a same rate. Here, a notation $F_1 \parallel F_2$ means that a pair of procedures F_1 and F_2 are independently, possibly concurrently performed. For example, $F_1 \parallel F_2$ is realized by creating a thread for each of the procedures F_1 and F_2. The ways of transmissions can be realized by the following procedures:

1. Synchronous transmission:
   for $h = 1, \ldots, l$ **do**
   { $\mathbf{Ssnd}(t_h, C_i) \parallel \ldots \parallel \mathbf{Ssnd}(t_h, C_{ik_i})$; }
2. Partial-synchronous transmission:
   for $h = 1, \ldots, g$ **do**
   {if $T_1^h \neq \phi$, $\mathbf{PSsnd}(T_1^h, C_{i1})$} $\parallel \ldots \parallel$
   {if $T_{k_i}^h \neq \phi$, $\mathbf{PSsnd}(T_{k_i}^h, C_{ik_i})$};
3. Asynchronous transmission:
   $\mathbf{Asnd}(T, C_{i1}) \parallel \ldots \parallel \mathbf{Asnd}(T, C_{ik_i})$;

In the synchronous transmission, each packet is multicast. After a packet is multicast, a succeeding packet is multicast. In the asynchronous transmission, a sequence of packets are transmitted for each channel independently of the other channels. If destination processes have different maximum receipt rates, the source process p_i can take the partial-synchronous transmission. The transmission synchronization ratio shows the receipt ratio of the destination processes.

Each destination process p_{ij} of a message m sent by a process p_i has some QoS requirement R_{ij}. A process p_i has to deliver a message m to every destination process p_{ij} so as to satisfy the QoS requirement R_{ij}. Let $Q_{ij}(t_k)$ show QoS of a packet t_k transmitted in a channel $C_{ij} = \langle p_i, p_{ij} \rangle$. When a group G is established among processes, every pair of processes p_i and p_{ij} do the negotiation on the preference P_{ij} to be used when a process p_i sends messages to the process p_{ij}. Let "$\succ_{ij}$" denote a preference relation "$\succ_{P_{ij}}$". $Q_{ij}(t_k)$ is required to be satisfiable for QoS requirement R_{ij}. In fact, $Q_{ij}(t_k)$ is shown in terms of bits. There are two ways to transmit a message m to multiple processes p_{ij}, ..., p_{ik_i} [Figure 3]:

1. *Quality(Q)-balanced* transmission: For each packet t_h of a message m, $Q_{i1}(t_h) = \ldots = Q_{ik_i}(t_h)$.
2. *Quality(Q)-unbalanced* transmission: For some pair of channels C_{ij} and C_{ih}, $Q_{i1}(t_k) \neq Q_{ij}(t_k)$.

Figure 3. Quality-based transmission.

In the first way, each packet of a message m is sent with same QoS in every channel. That is, a same packet is sent in every channel. In the second way, QoS in each channel is not necessarily same. For example, some channel supports lower bit-rate. In order to synchronously transmit packets to multiple destination processes, packets with smaller number of bits are transmitted.

Let us consider a synchronous transmission of a message m to multiple destination processes in $dst(m)$. If each channel supports enough QoS, a process p_i can synchronously send a same packet in every channel. Here, since each channel supports the same QoS, this is *quality(Q)-balanced* transmission. The Q-balanced, synchronous transmission is referred to as *fully synchronous*. If some channel C_{ij} does not support enough QoS due to congestions and network faults, the process p_i sends a packet t_k with less QoS in the channel C_{ij} than the others. That is, $Q_{ij}(t_k) \prec_{ij} Q_{ih}(t_h)$ for some channel C_{ik} ($h \neq k$). Next, suppose QoS is more significant than the synchronous requirement in an application. The process p_i sends the packets in the channel C_{ij} more slowly than the other channels. That is, the process p_i asynchronously sends packets of the message m. The Q-unbalanced, asynchronous transmission is referred to as *independent*.

3.2 Receipt

A process p_i receives messages from one or more than one process in a group G. There are following ways for a process p_i to receive messages from multiple processes p_{i1}, ..., p_{ik_t} ($k_i \geq 1$) [Figure 4]:

1. *Synchronous* receipt: A process p_{ij} sends a sequence $pkt(m_j)$ of packets $t_{j1}, \ldots, t_{jl_i}$ ($l_i \geq 1$) of a message m_j to a process p_i ($j=1, \ldots, k_i$). The process p_i receives a packet t_{jh} from each process p_{ij} after receiving a packet t_{fh-1} from every process p_{if} ($f=1, \ldots, k_i$).
2. *Partial-synchronous* receipt: A packet sequence $pkt(m_j)$ of a message m_j from a process p_{ij} is decomposed into a sequence $\langle S_j^1, S_j^2, \ldots, S_j^{g_j} \rangle$ of segments. Each segment includes the same number of packets, i.e. $|S_j^1| = |S_j^2| = \ldots = |S_j^{g_j - 1}| = NS_j$ and $|S_j^{g_j}| \leq NS_j$ for a process p_{ij}. The process p_i receives the

hth segment S_j^h from a process p_{ij} after receiving the $(h-1)$th segment S_f^{h-1} from every process p_{if} ($f = 1, \ldots, k_i$). Here, the ratio $NS_1 : \cdots : NS_{k_i}$ shows the receipt synchronization ratio for the processes p_{i1}, $\ldots$, p_{ik_i}.

3. *Asynchronous* receipt: A process p_i receives packets from each process p_{ij} independently of the other processes.

1 Synchronous.   **2 Partial-synchronous.**   **3 Asynchronous.**

Figure 4. Types of receipt.

"$t :=$ **Srec**(C)" shows a procedure to receive one packet through a channel C and store the packet into a buffer t. "$C = \phi$" means "end of transmission". Let T be a sequence of buffers t_1, t_2, $\ldots$, for storing one message, where each buffer can admit one packet. "$T :=$ **Arec**(C)" shows a procedure to receive a sequence of packets into a buffer sequence $T = \langle t_1, \ldots \rangle$ for a channel C. **Arec** is realized by the procedure: **while** $C \neq \phi$ **do** $\{t_h :=$ **Srec**$(C); h := h+1;\}$. Let T be a buffer for storing one segment. "$T :=$ **PSrec**(C)" shows a procedure to receive a segment of packets into a sequence T of buffers. The ways to receive a packet sequence $pkt(m_j)(= \langle t_{j1}, \ldots, t_{jl_j} \rangle)$ of a message m_j are realized as follows:

1. Synchronous receipt:
   while some $C_{ij} \neq \phi$ **do** {
   $t_{1h} :=$ **Srec**$(C_{i1}) \parallel \ldots \parallel t_{k_ih} :=$ **Srec**(C_{ik_i});
   $h := h + 1;$ }
2. Partial-synchronous receipt:
   while some $C_{ij} \neq \phi$ **do** {
   $T_1^h :=$ **PSrec**$(C_{i1}) \parallel \ldots \parallel T_{k_i}^h :=$ **PSrec**(C_{ik_i});
   $h := h + 1;$ }
3. Asynchronous receipt:
   $T_1 :=$ **Arec**$(C_{i1}) \parallel \ldots \parallel T_{k_i} :=$ **Arec**(C_{ik_i});

As discussed in transmission of messages, there are following ways to receive messages from multiple processes with respect to QoS:

1. *Quality(Q)-balanced* receipt: A process p_i receives packets with same QoS from each destination process p_{ij}.
2. *Quality(Q)-unbalanced* receipt: A process p_i receives packets with different QoS from different destination processes.

If a process synchronously receives messages in a quality (Q)-balanced way, the process is referred to as *fully receive* packets. If a process asynchronously receives packets in a Q-unbalanced way, the process is referred to as *independently receive* packets.

3.3 Receipt-transmission

Suppose there are three processes p_s, p_t, and p_u exchanging messages as shown in Figure 5. A process sends messages to processes while receiving messages from other processes. Suppose a process p_t sends a message m_1 while receiving a message m_2 from a process p_s. There are following ways to send messages while receiving messages:

1. *Synchronous* receipt-transmission: A process p_t sends one packet of a message m_1 each time p_t receives one packet of m_2 from a process p_s.
2. *Partial-synchronous* receipt-transmission: A process p_t sends some number n_1 of packets of a message m_1 to a process p_u while receiving some number n_2 of packets of m_2 from p_s. The ratio n_1:n_2 shows the receipt-transmission synchronization ratio of the process p_t to p_s.
2. *Asynchronous* receipt-transmission: A process p_t sends packets of m_1 independently of receiving packets of m_2 from p_s.

In the synchronous and partial-synchronous receipt-transmission ways, every common destination process p_u of messages m_1 and m_2 is required to synchronously receive the messages m_1 and m_2 from the processes p_s and p_t, respectively. The 1:1 partial-synchronous receipt-transmission way is just a synchronous way. In Figure 5, a process sends packets to a pair of processes p_t and p_u. The process p_t sends two packets to the process p_u while receiving three packets from the process p_s. Here, the synchronization ratio is 3:2. The process p_u is required to synchronously receive messages from the processes p_s and p_t with the receipt synchronization ratio 3:2 for the processes p_s and p_t. Here, a segment of a message m_1 includes three packets and a segment of m_2 includes two packets.

There are following types of receipt-transmission with respect to QoS:

1. *Quality(Q)-balanced* receipt-transmission: Packets received from a process p_s and packets sent by another process p_t have same QoS.
2. *Quality(Q)-unbalanced* receipt-transmission: Packets received from p_s and packets sent by p_t have different QoS.

Figure 5. Receipt-transmission synchronization.

4. Atomic and Causal Delivery

4.1 Group communication service

A group is a collection of peer processes which are co-operating to achieve some objectives. In the group communication, a process sends a message to multiple destination processes while receiving messages from multiple processes in a group. Let $s_i(m)$ be a sending event of a message m in a process p_i. The *happens before* relation on events in a distributed system is defined by Lamport [6]. A message m_1 is referred to as *causally precede* another message m_2 ($m_1 \rightarrow m_2$) iff a sending event $s_i(m_1)$ happens before a sending event $s_j(m_2)$ [6]. Every common destination process of messages m_1 and m_2 is required to receive m_1 before m_2 if $m_1 \rightarrow m_2$. Messages are causally delivered by using logical clocks like linear clock [6] and vector clock [7]. In addition, a process receiving a message m can deliver the message m only if every other destination process m surely receives the message m. In the papers [8,9], a protocol to atomically deliver messages in a group in presence of message loss is discussed.

Let us consider a group of four processes p_1, p_2, p_3, and p_4 [Figure 6]. Suppose the process p_1 sends a pair of messages m_1 and m_2 to each of the processes p_3 and p_4. The process p_2 sends a message m_3 to the processes p_3 and p_4 after receiving the message m_1 and then sends a message m_4 after receiving m_2. Here, the message m_1 causally precedes the messages m_3 and m_4 ($m_1 \rightarrow m_3$, and $m_1 \rightarrow m_4$) and the message m_2 causally precedes m_4 ($m_2 \rightarrow m_4$). Suppose the process p_3 receives m_3 and m_4 from p_2 and receives m_1 from p_1 but does not receive m_2 from p_1 due to the communication delay. The process p_3 can deliver m_1 but cannot deliver m_3 and m_4 because the message m_2 following the message m_1 from p_1 might causally precede m_3. The process p_3 has to wait for a message from the process p_1.

Figure 6. Causality.

Next, let us consider a pair of the processes p_3 and p_4 which receive messages from p_1 and p_2. The process p_4 receives the messages m_1, m_3, m_2, and m_4 in this sequence. On receipt of the message m_2, the process p_4 can deliver the messages m_1 and m_3. However, the process p_4 cannot deliver the message m_2 because the other destination process p_3 has not received m_2 yet. The process p_4 has to wait until p_4 knows that the process p_3 receives the message m_2.

If a communication channel C_{13} between processes p_1 and p_3 implies smaller bandwidth than another channel C_{23},

the process p_3 cannot deliver messages from the process p_2. If the process p_1 sends real-time multimedia data, the process p_4 cannot satisfy the real-time requirement even if the process p_4 receives all the messages. The delivery time depends on the slowest process.

Figure 7. Causality in a group.

4.2 Data communication instances

There are two types of requirements for group communication, *synchronization* and *QoS* requirements as discussed in the preceding section. Let **ST** and **QT** be synchronization and QoS types of transmission and receipt, respectively, i.e. **ST** = {Synchronous (S), Partial-synchronous (PS), Asynchronous (A)} and **QT** = {Quality-balanced (QB), Quality-unbalanced (QU)}. A data communication instance is determined by a tuple $\langle s, q \rangle \in \textbf{ST} \times \textbf{QT}$ which is a combination of a synchronization type $s \in \textbf{ST}$ and QoS type $q \in \textbf{QT}$. There are six data communication instances as shown in Table 1. For example, $\langle S, QB \rangle$ shows that a process takes a synchronous and Q-balanced transmission.

Table 1. Types of data communication.

Instances	Types of communication
$\langle S, QB \rangle$	$\langle$Synchronous, Q-balanced$\rangle$
$\langle S, QU \rangle$	$\langle$Synchronous, Q-unbalanced$\rangle$
$\langle PS, QB \rangle$	$\langle$Partial-synchronous, Q-balanced$\rangle$
$\langle PS, QU \rangle$	$\langle$Partial-synchronous, Q-unbalanced$\rangle$
$\langle A, QB \rangle$	$\langle$Asynchronous, Q-balanced$\rangle$
$\langle A, QU \rangle$	$\langle$Asynchronous, Q-unbalanced$\rangle$

A process selects a data communication instance which satisfies application requirements. Then, each process sends and receives messages by using the selected data communication type. If an application requires the strict atomic receipt, a process selects an instance $\langle S, QB \rangle$ or $\langle S, QU \rangle$, i.e. each process synchronously sends and receives messages. If an application does not require strict atomic and real-time communication, a process selects an instance $\langle PS, QB \rangle$ or $\langle PS, QU \rangle$. On the other hand, if a process requires real-time communication, a process selects an instance $\langle A, QB \rangle$ or $\langle A, QU \rangle$, i.e. each process independently sends and receives messages. If quality of data is the most significant for an application, each process takes the Q-balanced way. In another case, if quality is not significant for an application, each process takes Q-unbalanced one. We consider that data

communication instance which satisfies group communication service will be taken in the group.

4.3 Tree-based data transmission

We discuss how to transmit messages to processes in a group. A group G includes processes with different QoS and synchronization services. If every process in a group is required to take a same data communication instance, the group can only support applications with the minimum service to be supported. Hence, the processes in the group G are classified into six types of subgroups shown in Table 1. For example, synchronous and Q-balanced processes compose a $\langle S, QB \rangle$ subgroup. Then, subgroups are interconnected in a tree as shown in Figure 8. Let G_α show a subgroup of type $\alpha \in \{ \langle S, QB \rangle, \langle S, QU \rangle, \langle PS, QB \rangle, \langle PS, QU \rangle, \langle A, QB \rangle, \langle A, QU \rangle \}$. Messages are transmitted by the type α data transmission procedure in each subgroup G_α. In this paper, we consider a two-layered tree structure, i.e. one root subgroup and multiple leaf subgroup.

Figure 8. Division of group.

First, multiple processes in a group G establish transmission tree based on the synchronization and QoS requirements The group G is divided into subgroups each of which support some data communication instances. A root process S sends a sequence of packets to a process F_i in a root subgroup ($i = 1, \ldots, l$). Then, each process F_i forwards packets to processes $L_{i1}, \ldots, L_{il_i}$ in a leaf subgroup.

Figure 9. Data transmission route.

5. Concluding Remarks

This paper discusses a group communication protocol to exchange multimedia messages among multiple processes under some synchronization and QoS requirements. Multimedia messages are exchanged among multiple processes in a group so as to satisfy QoS required by applications. We discussed how to transmit and receive messages;

synchronous, partial-synchronous, asynchronous, quality-balanced, and quality-unbalanced ways. We also discussed the atomic and causally ordered delivery of messages with synchronization and QoS requirements. We are now designing the protocol for exchanging multimedia data in a group of multiple processes.

References

[1] ATM Forum. Traffic Management Specification Version4.0. 1996.

[2] K. Birman. Lightweight Causal and Atomic Group Multicast. *ACM Trans. on Computer Systems*, 9(3):272–290, 1991.

[3] R. Braden. Resource ReSerVation Protocol. *RFC2205*, 1997.

[4] IEEE Computer Society. Gigabit Ethernet. *IEEE Std 802.3z*, 1998.

[5] S. Itaya, T. Tojo, T. Enokido, M. Rozeta, and M. Takizawa. QoS-Based Synchronous/Asynchronous Data Transmission Model in Group Communication. *Proc. of IEEE the 18th International Conference on Advanced Information Networking and Applications (AINA-2004)*, 1:35–40, 2004.

[6] L. Lamport. Time, Clocks, and the Ordering of Events in a Distributed System. *Comm. ACM*, 21(7):558–565, 1978.

[7] F. Mattern. Virtual Time and Global States of Distributed Systems. *Parallel and Distributed Algorithms*, pages 215–226, 1989.

[8] A. Nakamura and M. Takizawa. Priority-Based Total and Semi-Total Ordering Broadcast Protocols. *Proc. of IEEE the 12th International Conference on Distributed Computing Systems (ICDCS-12)*, pages 178–185, 1992.

[9] A. Nakamura and M. Takizawa. Causally Ordering Broadcast Protocol. *Proc. of IEEE the 14th International Conference on Distributed Computing S ystems (ICDCS-14)*, pages 48–55, 1994.

[10] M. Ripeanu. Peer-to-Peer architecture case study: Gnutella network. *Proc. of International Conference on Peer-to-Peer Computing (P2P2001)*, pages 99–100, 2001.

[11] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real Time Applications. *RFC1889*, pages 22–29, 1996.

[12] T. Tachikawa, H. Higaki, and M. Takizawa. Group Communication Protocol for Realtime Applications. *Proc. of IEEE the 18th International Conference on Distributed Computing S ystems (ICDCS-18)*, pages 158–165, 1998.

[13] K. Taguchi, T. Enokido, and M. Takizawa. Hierarchical Protocol for Broadcast-Type Group Communication. *Proc. of the ICPP Workshop on Applications of Ad Hoc Networks (AANET2003)*, pages 21–28, 2003.

[14] T. Tojo, T. Enokido, and M. Takizawa. Notification-Based QoS Control Protocol for Multimedia Group Communication in High-Speed Networks. *Proc. of IEEE the 24th International Conference on Distributed Computing Systems (ICDCS-2004)*, pages 644–651, 2004.

Video-on-Demand for P2P Communities

A. Calvagna, L. Di Lorenzo, G. Tropea
DIIT - Università di Catania
Email: {andrea.calvagna, giuseppe.tropea}@unict.it, livliv@tin.it

Abstract

In this work we present a totally distributed peer-to-peer application aimed at Video-on-Demand. This target, never faced before, offers many challenging problems different from traditional file-sharing. Some of these are sequentiality and real-time constraints, sources' reliability and quality of the offered service. We introduce some new ideas to P2P networks, like affinity and collaboration between peers, linking and association in abstract entities called communities, to the end of lowering message overhead and improving efficiency in resource search. A simulation environment has been developed to test the network; results have demonstrated the scalability and robustness of the architecture and support our theoretical study. **Keywords:** *peer-to-peer; video streaming; affinity-based search*

1 Introduction

Research within peer-to-peer is evolving continually, given the enormous potentialities it offers in terms of resources availability at a very low cost. P2P is also being considered as a possible research field for multimedia and real-time applications, to overcome the scalability limitations of a traditional "many clients-few servers" conception. CoopNet [5] has recently been proposed as a solution to this problem, a peer-to-peer architecture of a mixed type where the server acts more as a network coordinator than a service provider. Still the centralized server remains a single failure point of this Cooperative Network model. When live broadcast streaming is the issue, a hierarchical structure of failure points is not easily avoidable. Nevertheless no significant proposals for totally decentralized networks have been advanced when on-demand streaming is involved. A major benefit P2P introduces in the VoD is its inherent scalability: the common goal is to replace expensive hardware with a network of low cost peers, thus offering a service which can scale to millions of users.

Nevertheless, many factors make a real-time distributed application much different from file-sharing. In traditional file sharing the file can be conveyed without caring about byte-sequence order and with no restraints in transmission time. In real-time film transmission this idea is inconceivable, as sequentiality is the key to make reproduction possible and time requirements are very strict. When a peer assigns part of its upload bandwidth to a corresponding peer, it might then be unable to accept other possible requests, until streaming of that chunk is over. When a peer requests a video stream, we must be sure that there are sources enough to supply the required data rate. Moreover when a peer is halfway in playing a movie, its knowledge of the network state may be already outdated, and service be degraded by the lack of sources. Novel source discovery algorithms and peer aggregation strategies must be introduced, along with a lightweight P2P communication protocol.

2 System Architecture

Routing inside our P2P network is based on the Kademlia protocol. Kademlia routing protocol has a computational complexity which is equivalent to complexity of other protocols (see [4]), but is more flexible because of the quality of the XOR metric [3] it adopts.

Node identifiers (which we here call *IDnode*) are randomly generated 160 bit vectors. For each $0 \le i < 160$, a node holds a list of triples *<IPaddr, port, IDnode>* of the nodes which have distance between 2^i and 2^{i+1} from this node. Thus each node stores 160 lists and every list is called a *k-bucket* because it holds k triples, with $k = log(n)$ and n as the number of peers (usually $k = 20$ for networks of 1 million nodes). Every k-bucket

is ordered by keeping at the head of the list the node that has not been contacted for longer time.

Kademlia uses a k-bucket updating based on standard traffic exchanged between nodes. No node will be eliminated from the lists if it is still active. This leads to a remarkable robustness of the whole network and comes from statistical observation of Gnutella users: it has been proved that the longer a node has been online, the higher the probability is that it stays online one more hour.

2.1 Tracing of resources

Here we introduce many important novel characteristics that make indexing of resources inside our network deeply different from other existing P2P networks.

Each peer is bound to allocate part of its disk space to form a local cache containing film chunks made available for download. Each cache is then part of a large "distributed cache", which is the key element to obtain efficiency and adequate permanence of the films inside the network. Our distributed cache performs duplication of the more requested resources, so as to maintain sources availability proportional to users' requests. The basics of our chunk finding algorithm lie in the efficiency and distributed features of DHT store (see [1]), which we use to assign one or more resources to a peer. Through SHA-1 criptography we obtain a unique key (which we call *IDmovie*) from any given video file; this key is a 160 bit vector, thus in the same space as the *IDnode* vectors.

When the algorithm carries out a lookup request for an *IDmovie*, it will find the list of k nodes "nearest" by the XOR metric to the above vector. These nodes we call the *access points* to the indexed resource and they maintain the necessary information for video files tracing and availability. They serve as starting points in the distributed data structure that keeps track of a film's chunks. Each access point contains a list of links to some of the nodes possessing the first chunk of the indexed film. So the peer that starts the reproduction will contact the access point only once and will be able to contact one of the owners of the first chunk. Besides, each of these owner peers will contain links to other sources of the same chunk, as well as to the owners of the previous and following chunk. So we understand that the resulting data structure is a dynamic linked list that has a high degree of scalability and fault-tolerance. Later, when we need to refer to all the peers containing the same film chunk, these will be referred to as a

Figure 1. The chunk list. Multiple access points can exist. The video file is made up of n chunks.

chunk zone and the data structure will be called *chunk list*, see Fig.1. Thus each chunk list will consist of a list of chunk zones linked to each other. An adequate number of links among the peers is necessary for this structure to work. So each peer must contain a list of k links to elements of the same chunk zone as well as k links to the previous and next chunk zone. Obviously, if the number of copies of a chunk is less than k, the list of links in a peer would be incomplete. List is updated and kept ordered by putting at the head of it the most recent contacts. When a new peer comes into contact, a ping is made of all the elements in list, starting from the latest. If one of them does not answer the ping, the new element is put on top and the other is dropped. If they all answer, the new peer is equally inserted on top, and the last one is the one to drop. In any case each peer of a chunk zone periodically contacts a subset of the elements in list, in order to keep track of its presence in case too long time has elapsed since last contact. In this way, contacts between the peer tend to be as frequent as possible. If chunk i of a film is extinguished, the most evident consequence would be an empty chunk zone. As a result, the elements of chunk zone $i-1$ should keep the links to chunk zone $i+1$, waiting for a peer with chunk zone i to get back online to give rise to a new level in the chunk list. The moment a peer of chunk zone $i-1$, or $i+1$, comes into contact with it, it would immediately insert it in its link list and notify the event to all the links of the same chunk zone, which in turn signal it to their own links. This is necessary in order not to make a chunk list inconsistent, but does not exclude the chance that a peer in not reached by the notification message and thus only keeps links at the chunk zone which

is two steps away (instead of one). For this reason, while scrolling a list, the peer does not take for granted the information of a missing chunk zone, before getting the same answer from more than one peer.

2.2 Resource distribution

An efficient management of this distributed storage should meet two basic requirements:

1. The most popular films, that are the ones mostly required, should enjoy a high duplication so as to allow all the peers to get good quality playback;

2. The less popular films, the ones whose duplication level may be lowered to the minimum, should never be extinguished.

The algorithm devised to quantify the saving priority of a chunk is based essentially on two parameters, i.d. the popularity of the film and that of the chunk. They are estimated in rather different ways, according to the information supplied by the distributed structure of the net. As for a film popularity, the rule is based on the variable rph (request per hour by the access points). It should be remembered that the routing algorithm at the base of the DHT exerts the duplication of information, so there will not be a single access point but at least k. Each of them will probably be consulted by each peer requiring the film.

Each access point thus takes into account the total number of contacts c_{ap} and the number of minutes it has been online t_{on}. We have $rph = c_{ap} \cdot \frac{60}{t_{on}}$.

Once we know that value, we estimate the film popularity pop_f so that $pop_f \in [0, 100]$ and $pop_f = 100 \cdot \max\left(1, \frac{rph}{T}\right)$ where T is a fixed application parameter to weight the number of requests per hour of each film; one may choose to keep it low to enhance the weight of less required films, or high to lower the duplication rate of more obsolete video files.

Estimation related to the popularity of a chunk pop_c is more complex, considered that the number of duplicated chunks is practically unavailable; a peer can only try to approximate it by carrying out the following steps:

1. contact the first peer whose identity is known and ask for its list of k neighbours;

2. contact each of the elements in list and ask for their list of k neighbours;

3. excluding doubled peers, it estimates the filling rate of its own list through a percentage value. To do so, if we assume that all peers have a filled list of k elements and the distinct elements obtained are k_d, the chunk popularity will be given by $pop_c = 100 \cdot \left(\frac{k_d}{k^2+k}\right)$. When the elements in a list are less than k, factor $k^2 + k$ has to be accordingly changed.

Once estimated the two necessary parameters, each chunk is assigned a *score*, given by the contributes of the film score and that of the chunk.

Definition 2.1 (*score*). *The aim of the score is that of quantifying the validity of the chunk's request to occupy a space in the peer's cache.*

Then a comparison between the present chunk *score* and the others now present in cache is made, and if there is a chunk in cache with a lower *score* than the candidate, it will be substituted by the new one. In only one case is this mechanism not respected, i.e. the case in which the chunk is the only specimen present in the net. In this case (see our second requirement at the beginning of this section) the chunk will be maintained.

The rule we have chosen is that of giving a high score to very popular films and a medium-low score to the less popular. But being the latter less privileged in caching, and so at extinction risk, we have chosen to privilege the least diffused chunks, by making the chunk popularity have an influence on their score. The *score* is defined as

$$score = \begin{cases} \max(100, score_f + score_c) & \text{if } pop_f < 50 \\ score_f & \text{if } pop_f \geq 50 \end{cases} \tag{1}$$

and has values in the $[0, 100]$ range. $score_f$ and $score_c$ are the partial scores of the film and of the chunk. As stated before, if a video is considered little popular ($pop_f < 50$), the contribution due to the chunk popularity, is added to the score, so privileging the less popular ones. The threshold to distinguish a popular from a less popular film is given by the rph^* value, where $\frac{rph^*}{T} = 0.5$. We easily obtain $T = 2 \cdot rph^*$.

2.3 Communities

A video-on-demand application is different from a generic file sharing application in another way: distributed content can easily be categorized into a set of thematic areas, which we call *attributes* from now on. This aspect is far from being uninfluential on the structure of the network, since it introduces an aspect of

P2P nets which has been little exploited so far, i.e. locality and the existence of communities. The idea of communities as affinity groups sharing the same interests has already been exploited on the Internet through the "web communities". These are always based on a central authority coordinating their functioning, and the users' participation is only possible through explicit assent. P2P-based communities can be very dynamic and self-managed, since it is the peers themselves that find "similar" users to make links with. We can even find parameters to quantify such affinity, so as to create a dynamic list of "neighbours" (not in the geographical meaning of the term) which may help the peer to make the best of its online experience.

Each user will maintain a list of neighbours that is quite distinct from the k-buckets necessary for routing. It is made up of a fixed number of elements (i.e. $100 \sim 200$ elements) and is continuously updated as new links are made. The size of this list influences the performance of the net. A high number implies a greater involvement of peers in the network and the possibility of getting a greater amount of information. On the other hand, this might lead to excessive communication overhead. All the contacts present in the list of neighbours sharing an attribute with this peer will implicitly be part of the same *community*. It is now necessary to give some definitions for a thorough analysis of the idea of *affinity* between peers, inspired by [2].

Definition 2.2 (*outlink weight*). *It is the weight assigned to each attribute in relation to the percentage of a peer's outlinks that directly reach other peers sharing the same attribute.*

The greater the outlink weight for an attribute, the higher the percentage of neighbours sharing the attribute.

We may also quantify the involvement of a peer in the network, as follows:

Definition 2.3 (*involvement*). *We define involvement of a peer P, within a community interested in attribute A, the sum of A's outlink weights applied to all P's neighbours that contain A among their attributes of interest.*

Evaluation of the affinity between two peers consists in comparing their degree of *involvement* in the attributes of common interest. Given a peer P and the size of the neighbours' list L, if we call $F \leq L$ the number of peers in list that own attribute A, then *involvement* relative to attribute A is given by $involvement =$ $\sum_{i=1}^{F} o_i$ where o_i are the outlink weights of the F peers that contain A. Being $0 \leq o_i \leq 1$, we have that $0 \leq involvement \leq F$. When we want to evaluate the affinity between peer P_1 and peer P_2, we have that $affinity = \sum_{i=1}^{n} involvement_i$ where n is the number of common attributes among P_1 and P_2 and $involvement_i$ are the *involvement* coefficients of the abovementioned attributes.

We can easily point out some special cases: if inside the network we only have movies belonging to a unique attribute A, shared among all the peers, we would obtain $o_i = 1$, $involvement_i = L$ and as a consequence $affinity = L^2$; if the two peers P_1 and P_2 have no attributes in common we obtain $affinity = 0$; if P_1 and P_2 do have common attributes but inside P_1's (or P_2's) list of neighbours there is no peers sharing at least one of those attributes with it, then $affinity = 0$.

List management is dynamic and ruled by the affinity coefficients. When peers come into contact, they exchange their list of attributes and their relative *involvement* coefficient, thus calculating their *affinity* coefficient. If a new peer has a higher degree of *affinity*, it will be inserted instead of another. Thus there is a continuous recirculation of neighbours and gradually the list will be made up of the most "interesting" elements, from the point of view of the peer. On this list of affine neighbours we build some key functions of our architecture, as we will analyze in the following sections.

2.4 Searching resources

The community has an extremely important role for the peers, because it allows them to keep up to date with availability of net resources. Considering the difference between this type of application and file sharing, a mechanism of file search by name is not contemplated. The principle on which the net is organized is thematic search. To this end we exploit peers locality and the above affinity of interest, the key element to build communities.

The reason for the introduction of this mechanism is that, in its absence, a user interested in a film should beforehand know its *IDfile* (i.e. its hash). If on the one hand this corresponds to the idea of "link" for the Internet, on the other the mechanism is not handy, since it would imply an external infrastructure to keep track of the available resources. Our solution is to let resource owners themselves advertise and signal their existence. Here comes into play the idea of community. The management of the information regarding resource availabil-

ity is based in fact on peer location and their willing to collaborate in the common interest.

Each peer keeps a list of the films partially or totally contained by itself or by its neighbours. The list is updated periodically to meet the changing of the cache, the dynamism of the network and the variations of neighbours' lists. The aim is not that of listing the sources of each film (this is done by the chunk list). It is rather to keep *<file name, IDfile>* associations to allow search and playback of the movies. Whenever a new peer is added to the list of neighboring peers, its list of video files is transferred locally, so that each peer contains a list of all neighbours' files. This is a decisive factor to obtain a broad though not invasive view of network resources.

Locally contained films are stored in the *local list*, while films contained in neighboring peers are stored in *second level* lists. If we assume optimistically that each peer contains a list of neighbours distinct from any others, and L is the size of the list, we might obtain the file IDs contained in L^2 peers. If, for example, $L = 200$ we already have the information from 40.000 peers with no messaging overhead inside the network.

2.5 Real-time playback

Real-time playback may be considered as the highest level of our architecture. We have to deal with reliability of service and accurate selection of playback sources. The objectives of our application are the following: 1) assuming that, through all the playback, the state of the network remains unvaried, it must guarantee that the service will come to a successful conclusion keeping high and constant quality. Otherwise user request is rejected; 2) it must minimize playback interruptions by a correct scheduling of stream sources.

When a request to stream a chunk is accepted by a peer, a portion of the upload bandwidth (a *slot*) is assigned to the "client peer" till the entire chunk is streamed. Thus the receiving user is guaranteed a constant bitrate, and no degrading of quality occurs from excessive uploading, since new requests beyond upload capacity are rejected. A chunk may be transferred from multiple sources in parallel, so that each source gives a slot of its upload bandwidth till the bitrate of the film is matched and successful playback can occur. Multiple sources and stream mixing is natively supported by RTP protocol [7].

The problem in choosing the available sources is that peers can leave the network asynchronously. The only

effective way to minimize the risk of loosing bandwidth slots during playback, is to rely on a statistical approach. Gnutella trace files (see [6] and [3]) can be used to build a useful relationship between the probability of remaining online and the actual online time. After an analysis of the trace files we have chosen the function

$$y = 0.92 - 0.4 \cdot e^{-x^{0.9}/100} \qquad (2)$$

to approximate the results, where we call x the online time in minutes and y the probability to remain online, that is the fraction of nodes that stayed online x minutes and remained online for the next $x + 60$ minutes.

Thus by knowing the actual online time t_{online}, the probability to remain online 60 more minutes is given by $p_{online} = y(t_{online})$ and the minimum online time t_{min} for a peer is $t_{min} = 60 \cdot p_{online}$ minutes.

When a peer wants to reproduce a chunk, it goes through the chunk list up to the relative chunk zone and tries to fill in its own list of k sources for this part of the film. Next, by knowing t_{online} for each source, it estimates the possible time the source remains available and compares it with chunk playback time in minutes t_{chunk}, given by $t_{chunk} = \frac{\text{CHUNK\_SIZE} \cdot 8}{60 \cdot bitrate}$, where CHUNK_SIZE is in bytes. All sources having $t_{min} < t_{chunk}$ are immediately excluded. The remaining are sorted by free upload bandwidth and from them the slots are requested. The peer playing the chunk will decide at the end of the chunk whether to keep it in cache (according to the criteria of section 2.2) or not. A peer reproducing a chunk becomes a source for that chunk and enters the corresponding chunk zone, by populating its own list of k elements.

3 Simulation results

We have tried to obtain important information about the scalability of the system, which is our main objective, by considering networks of up to 100.000 peers inside a custom simulation environment we have developed for this project. Most important simulation objects are the peer and the movie.

The peer. Upload and download bandwidth are symmetrical and set to 1200KB/s. Peer parameters are as follows: k is set to $\log N$, being N the number of peers; α is set to 3; L is set to 200 unless otherwise stated; $T = 30$; cache size is set to 40, and it is initially filled up at 30% with chunks belonging to randomly chosen films.

Figure 2. Affinity as a function of time.

The movies. The *attribute* variable is randomly chosen among the 15 different values we have supported; *bitrate* is randomly set between 900 e 1.800 kbps; *length* is in minutes, 80 to 120; file size is derived in kilobytes as follows: $filmsize = \frac{bitrate}{8} \cdot length \cdot 60$; the level of *interest* of a movie is a random between 0 and 100.

3.1 Diffusion of communities

In Fig.2 the most interesting aspect of the links between peers is pointed out. It is the average value of the affinity, which shows the quality of the neighbours for each peer, in terms of similarity of interests and involvement within one's communities. The highest benefits are obtained in large networks, where the average affinity is about 16 times as large as in the small ones. For this test we have also enlarged to $L = 1000$ the maximum value of neighbours and recorded a very high value of contacts (650 per peer on average). So our mechanism is obviously oriented towards large networks and communities.

3.2 Conclusions

We have shown that a P2P architecture can reach valuable results even when strict real-time constraints must be respected. To our knowledge this is the first approach that combines a distributed P2P architecture and the requirements of a VoD application. A new approach consisting in a self-organization of the peers, based on affinity and communities has been introduced and the feasibility of a distributed network of video servers has been proven, in terms of service guarantee and availability of resources. Further investigation must be done, especially in the field of the proposed new search engine,

that is based on the concepts of affinity and involvement of the peers in the network itself.

References

[1] J. Cates, *Robust and efficient data management for a distributed hash table*, Master's thesis, Massachusetts Institute of Technology, May 2003.

[2] M.S. Khambatti, K.D. Ryu, and P. Dasgupta, *Efficient discovery of implicitly formed peer-to-peer communities*, International Journal of Parallel and Distributed Systems and Networks **vol. 5** (2002), no. 4, pag. 155–164.

[3] P. Maymounkov and D. Mazieres, *Kademlia: A peer-to-peer information system based on the XOR metric*, 2002.

[4] D.S. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, and Z. Xu, *Peer-to-peer computing*.

[5] V.N. Padmanabhan, H.J. Wang, P.A. Chou, and K. Sripanidkulchai, *Distributing streaming media content using cooperative networking*, Tech. Report MSR-TR-2002-37, Microsoft Research, Redmond, WA.

[6] Stefan Saroiu, P. Krishna Gummadi, and Steven D. Gribble, *A measurement study of peer-to-peer file sharing systems*, Proceedings of Multimedia Computing and Networking 2002 (MMCN '02) (San Jose, CA, USA), January 2002.

[7] Schulzrinne, Casner, Frederick, and Jacobson, *RTP: A transport protocol for real-time applications*, Internet-Draft ietf-avt-rtp-new-01.txt (work in progress) (1998).

A Threshold based Multicast Technique in a Distributed VoD System with Customer Reneging Behavior

Sonia González, Angeles Navarro, Juan López and Emilio L. Zapata

Abstract— We present a new delivery strategy for distributed VoD systems, inspired in the threshold based multicast technique. Our approach tries to not only minimize the overall bandwidth usage in the system but also to provide a preset quality of service for each customer. Threshold based multicast technique allows a client that arrives before a threshold to receive part of a complete stream (which was serving a previous request) by listening to the ongoing stream of the requested video. In addition, a partial stream is initiated to send the first minutes of the video to catch up with the complete stream. In our strategy, once a complete stream has started, instead of serving the new incoming requests immediately, the requests are delayed a few seconds to form a mini-group first. The mini-group will then be served by a partial stream in order to merge into the complete stream. In that way, the clients batched together in a mini-group share the same partial stream, reducing even further the bandwidth usage. We briefly present an analytical model to help selecting the time of batching to form mini-groups as well as computing the threshold to control when a complete stream is initiated. We include some preliminary experimental results which demonstrate that our strategy reduces both the customer reneging probability and the waiting times when they are compared to previous techniques.

Index Terms— Video-on-Demand, Distributed servers Architecture, Multicasting, Reneging Behavior.

I. INTRODUCTION

IN recent years, there has been an increasing interest in Video on Demand (VoD) systems thanks to the advances in networking and video technology. A promising way to contain the costs in such systems is to build a distributed VoD system made of several small servers interconnected by a network and store (cache) in such servers the videos more frequently demanded by the local clients.

We show in Figure 1 the architecture of our distributed VoD system. In particular our research system consists in N_{serv} servers interconnected by an ATM switch. The bandwidth of the ATM switch is B_{switch} and each link that connects one server with the switch has a B_{link} bandwidth in each direction. Every server has attached N_{term} clients, and can retrieve concurrently N_{stream} streams, being B_{stream} the bandwidth required for servicing a video [1]. The number of videos offered by the system is M.

In this paper we assume that the videos are distinguishable and that they are required according to their **popularity**. In our system, the popularity of the videos is given by the customers' behavior. Basically, the **popularity** of a video in the system

Fig. 1. Distributed VoD system.

is defined as the frequency of requests for this video issued for all the customers in the system. We focus our research on local or metropolitan area networks. In this context, it seems reasonable to assume similar user preferences in all the servers and for this reason we work within global popularities. To keep our proposal general enough, we assume that the popularity follows a Zipf law distribution [2], which models the video popularity quite well [3]. Such distribution depends on the parameter ξ (degree of popularity) that correspond to the degree of skew. The greater ξ, the more skewed on the popularities.

In a distributed VoD system the replication of MPEG-2 videos in all the servers would require huge storage capacity that would make the system prohibitively expensive. So in our approach we reduce the storage requirements by not allocating all the content in all the servers. Only the most popular videos are replicated in all of them. We call **replication degree** to the percentage of videos replicated in all the servers.

Obviously, the user preferences will not be uniform over a given 24-hour period. Not only does the video popularity fluctuate, but the load. The system provider could characterize the load periods, and on a daily basis, for each load period, it could update the video popularities. In this paper, we work on one such load period where the request rate and popularities are known and don't fluctuate. Using that information, the distribution of the M videos among the servers is carried out as follows: first, we assume that the M videos are ordered with regard to their popularity in the system (i.e. the most popular ones first). Using the replication degree, we establish the number s of videos to be replicated, select the s first videos in the ordered sequence and store them in every server. The remaining $M - s$ videos are uniformly distributed among the servers. With this distribution the non replicated videos are stored only in one server. Therefore, all system requests for these videos have to be served only by the corresponding server. This situation is obviously worse than if these videos

were copied in more than one server. We think this is a challenger worst case distribution for our system.

The clients attached to a sever can request videos that are stored in the server (*local request*) or videos that are not stored in it (*remote request*). A local request which is attended (**local service**) consumes local bandwidth for the video stream. A remote request to a server which is served generates a **remote service** from the server that allocates the requested video.

Fig. 2. Threshold-based Multicast.

We note that a remote service consumes more resources than a local service: i) local bandwidth in the remote server, ii) bandwidth in the output link of the remote server, iii) bandwidth in the ATM switch and iv) bandwidth in the input link of the local server which made the request.

To minimize the high amount of bandwidth required if customers are individually served, several strategies has been proposed in order to minimize the server and network bandwidths. One approach is the batching technique where the clients are grouped together to be served by a unique multicast stream. The period of batching could exceed the time that some clients are willing to wait and these ones could renege. The quality of service (QoS) of a VoD system implementing this technique is measured by two factors: the *reneging probability*, that is, the probability that the clients leave the system without receiving service, and the *waiting time* (T_{wait}), which is the average of the period which elapses between the arrival of a request and the time when the service is initiated.

Another approach that tries to reduce the server and network bandwidth usage without introducing any client startup delay is the patching policy [4]. This strategy allows a client to receive part of a complete stream (which was serving a previous request) by listening to the ongoing stream of the requested video. In addition, a partial stream is started to send the first minutes of the video to catch up with the complete stream. To achieve this, clients must be able to receive twice the media playback rate, and to have some amount of local storage (set-top box).

Several patching-type approaches have been proposed in [5], [6], [1]. The basic scheme in these approaches, called *Threshold-based multicast*, introduces a threshold to control the frequency to which a new complete stream is initiated: with this scheme if a client arrives out of the threshold, the server initiates a new complete stream transmission. Figure 2 shows how this scheme works. Let's suppose that, at time 0, Client 1 requests video i (the length of video i is L_i). Then, a complete multicast stream is initiated to serve it. Let's suppose that U_i is the threshold for video i. As we can

see, since Client 2 and Client 3 arrive before U_i, they join the ongoing stream initiated for Client 1 while two partial streams are initiated to serve them. However, since Client 4 arrives after time U_i, a new complete stream is initiated for this last request. The authors demonstrated that this scheme optimizes the used bandwidth and decreases the waiting time when compared to a patching policy without threshold. Others works as [7] and [8] propose approaches where the bandwidth usage is a logarithmic function.

However the reneging behavior of the clients is not taken into account in none of the above mentioned approaches.

In this paper, we present a new delivery strategy for our distributed VoD system, inspired in the threshold based multicast technique, but we also consider reneging.

In our approach, the key idea is to apply the following strategy to the most popular videos: once a complete stream has started, the new incoming requests are not served immediately but delayed a few seconds to form a multicast mini-group. The mini-group will then be served by a partial stream in order to merge into the complete stream. In that way, the clients batched together in a mini-group share the same partial stream, reducing even further the bandwidth usage. A critical point in our approach is the selection of the period of batching to form mini-groups. We will determine this period in such a way that the number of clients in mini-groups be high but do not increase the chance of clients' reneging. In addition, we compute the threshold to control when a complete stream is initiated, which turn to be depending on this period. As we mentioned, the goal of that threshold is to optimize the bandwidth usage. In Section II we present a model to compute the parameters that minimize the used bandwidth while guarantee that the customer reneging probability is smaller than a preset value. The experimental results showed in Section III have demonstrated that our strategy reduces both the customer reneging probability and the waiting times when they are compared to previous techniques. In other words, it improves the QoS of the system.

II. OUR MULTICAST APPROACH

The goal of this paper is to minimize the waiting times for a service to begin in a distributed VoD system with a preset quality of service for each costumer. To this end, we use a scheduling strategy in order to optimize the bandwidth resources. We use the mixed patching-batching multicast technique mentioned in the previous section to schedule the videos. In this section we present the main characteristics of our approach, that we will denote as BITM (Batching Intervals in a Threshold based Multicast).

A. Technique Description

Our technique verifies the following characteristics:

1) We suppose that clients can be impatient, so, they can leave the system without being served if their waiting time exceeds a level of tolerable delay.
2) In each server there is a queue per video. The queues are scheduled following the criterion of Maximum Factored Queue Length (MFQL) [3]. MFQL schedules the videos with the maximum *factored queue length*. This parameter is obtained by applying a weighting factor to each video queue length, where the weight decreases as the popularity of the video increases. This technique

Fig. 3. Our BITM approach.

trades off the popularity of the videos against the time that customers are queued. Therefore MFQL is a fair scheme because balances the scheduling of videos [3].

3) Once a queue is scheduled (for the easy of the explanation let's suppose the queue of requests to video i), a complete stream is initiated to serve the clients in the queue (this happens at time 0 in Figure 3).

4) Let's assume that the next request to video i arrives within the threshold U_i at time t_1 (it corresponds to Client 1 in the mentioned Figure). Then, this request and the next (to video i) arriving during the interval $[t_1, t_1 + t_d]$ (as happens for Client 2) join at time $t_1 + t_d$ to the ongoing multicast session initiated previously at time 0. At the same time, a multicast partial stream is started to send to the group of clients the first minutes of the video, to catch up with the complete stream.

5) A new request to video i arriving in the interval $(t_1 + t_d, U_i]$, as happens with Client S in Figure 3, will define a new mini-group and will be served as in the case of Client 1. However, when Client N arrives, the mini-group is restricted to the clients arriving in $[t_n, U_i]$.

6) Finally, any request to video i arriving out of the threshold is queued until the scheduler provides a new multicast stream for video i (as happens to Client T).

Our aim is to determine the optimal threshold U_i and the maximum batching interval t_{max} to minimize the bandwidth requirement. To this end, we first determine the t_{max} such that the maximum number of clients arriving within it, are grouped together in a mini-group, but the reneging probability doesn't overcome a preset value α. Once t_{max} is obtained, we will derived the threshold U_i that minimizes the used bandwidth. As we will see, the optimal threshold U_i depends on t_{max}.

B. Analytical model

In this section we assume that:

1) The requests arrive to any server following an exponential distribution with rate λ. Then, the requests for video i arrive at each server according to an exponential distribution with rate λ_i, where $\lambda_i = \lambda \cdot p_i$, being p_i the probability of requesting the video i.

2) Each client has different tolerable delay. This behavior may be described by a sequence of random variables mutually independent and with the same density function $(1/MRT) \cdot e^{-x/MRT}$, where $1/MRT$ is the reneging rate request [9], [10].

3) As the videos have different probabilities, they will have different bandwidth usage. So we focus our discussion on a particular video i with a length of L_i seconds and a threshold of U_i seconds.

Let be $F(x)$ the exponential distribution function that defines the reneging behavior. Let $P_i(t)$ denote the fraction of requests for a video i in a interval of length t that leave the system, assuming that the video service begins after the end of the interval t. Due to the Poisson arrival assumption, requests arrive uniformly in [0,t]. Therefore by [9], [11],

$$P_i(t) = \frac{1}{t} \int_0^t F(x)dx \qquad (1)$$

By using the exponential reneging function it follows that

$$P_i(t) = \frac{1}{t} \int_0^t F(x)dx = 1 - MRT \cdot \frac{1 - e^{-t/MRT}}{t} \qquad (2)$$

We want to keep small enough the reneging probability, let say smaller than a preset low value α. Therefore, the maximum interval t_{max} must verify that

$$P_i(t_{max}) \leq \alpha \qquad (3)$$

Approximating the function $e^{-t/MRT}$ by its Taylor polynomial of order 2 we have that, $P_i(t_{max}) \approx \frac{t_{max}}{2 \cdot MRT} \leq \alpha$, thus,

$$t_{max} \leq \alpha \cdot 2 \cdot MRT \qquad (4)$$

As we mentioned in the Introduction, since t_{max} provides the opportunity that a mini-group of requests for a video i shares a partial stream, we expect that more than one client will share the partial stream at the end of a t_{max} period. That happens if

$$\lambda_i \cdot t_{max} > 1 \qquad (5)$$

So, given a probability of reneging α, Equation 4 gives an upper bound for t_{max}. On the other hand, Equation 5 will determine the videos to which our strategy is applied, it is, the videos for which a mini-group is created. Only very popular videos (those that verify Equation 5) would benefit from this feature. For instance, we can choose the reneging probability $\alpha = 0.05$, which is small enough to consider negligible the number of clients that leave the system during t_{max}. This number of clients is lower than $\lambda_i \cdot t_{max} \cdot 0.05$. For example, if $t_{max} = 60$ seconds and $\lambda_i = 0.102$ req/sec the number of clients arriving within during t_{max} is 6, whereas the number of clients that leave at the same period is lower than 0.306.

After t_{max} is determined, we focus in deriving the optimal threshold that reduces the used bandwidth. In our analysis we use the average server bandwidth (denoted as B_i) requirement as the performance parameter in order to find the threshold that optimizes the resources usage. To derive that optimal threshold we model the system as a renewal process [6]. We are interested in the process $\{S(t) : t > 0\}$ where $S(t)$ is the total server bandwidth used from time 0 to t. In particular, we are interested in the average server bandwidth

$$C = lim_{t \to \infty} S(t)/t$$

Let $\{t_j\}_{j=0}^{\infty}$ ($t_0 = 0$) denote the times at which the system schedules a complete stream for video i. These are renewal points in the sense that the behavior of the system for $t \geq t_j$ does not depend on past behavior. We consider the process $\{S_j, N_j\}$ where S_j denotes the total server bandwidth used and N_j the total number of clients served during the jth

renewal epoch $[t_{j-1}, t_j)$. Because this is a renewal process, we will drop the subscript j. Then we have the following result:

$$B_i = \lambda_i \cdot \frac{E[S]}{E[N]} \tag{6}$$

Next, we derive $E[S]$ and $E[N]$. Since the fraction of clients that leave the system is negligible, $E[N]$ is given by $E[N] = \lambda_i \cdot U_i$ that represents the clients that will arrive within the threshold U_i and that will be served with partial streams. Consider now $E[S]$. Since our system has limited resources, some requests will not produce partial streams. Then, an upper bound of $E[S]$ is given by the case of unlimited resources. This bound is as follows,

$$E[S] \leq \left(L_i + \sum_{j=1}^{\lfloor \frac{U_i}{1/\lambda_i + t_{max}} \rfloor} \left(\frac{1}{\lambda_i} + t_{max} \right) \cdot j + U_i \right) \cdot B_{stream} \tag{7}$$

The first term of the equation, $L_i \cdot B_{stream}$, denotes the bandwidth required by the complete multicast stream initiated to serve the video i. For the second term, the upper limit of the sum $\lfloor \frac{U_i}{1/\lambda_i + t_{max}} \rfloor$ is the number of mini-groups that can be formed (see subsection II-A) in an interval of length U_i. The sum denotes the aggregate bandwidth required by the partial streams started to serve the different mini-groups. The third term, $U_i \cdot B_{stream}$, corresponds to the bandwidth required to serve the partial streams for the clients that arrive during the interval $(t_n, U_i]$.

Once determined $E[S]$ and $E[N]$, we replace them in Equation 6 and we derive an upper bound of the average server bandwidth required to service video i as,

$$B_i \leq \frac{L_i + \frac{1}{2} \cdot \frac{U_i \cdot (U_i + 1/\lambda_i + t_{max})}{1/\lambda_i + t_{max}} + U_i}{U_i} \cdot B_{stream} \tag{8}$$

The value of U_i (the optimal threshold) that minimizes the second term of this expression is,

$$U_i = \sqrt{2 \cdot L_i \cdot (1/\lambda_i + t_{max})} \tag{9}$$

Therefore the corresponding value of the lowest upper bound of the average used bandwidth B_i is,

$$B_i \leq \left(\sqrt{\frac{2 \cdot L_i}{1/\lambda_i + t_{max}}} + \frac{3}{2} \right) \cdot B_{stream} \tag{10}$$

At this point, we would like to evaluate the behavior of our approach. For comparison purposes, we use a baseline version of the BITM algorithm, in which the clients are not delayed to form mini-groups. An analytical model for that baseline scheme is derived in [1]. In this scheme, the optimal threshold U_i' is given by $U_i' = \sqrt{2 \cdot L_i/\lambda_i}$ and the average used bandwidth B_i' has as lowest upper bound $B_i' \leq \left(\sqrt{2 \cdot L_i \cdot \lambda_i} \right) \cdot B_{stream}$.

We can note that the analytical values U_i and B_i for BITM depend on the parameter t_{max}. In Fig. 4(a) and Fig. 4(b) we have respectively depicted the optimal threshold and the lowest upper bound of the average used bandwidth for both schemes and for the most popular video among the 100 videos offered by the system. The parameters that we have used are the same that we will use in section III. As we can see from Fig. 4(a) the optimal threshold for BITM is always higher than the optimal threshold derived for the baseline scheme. As t_{max} increases, U_i increases and as we can note in Fig. 4(b) the

bandwidth required by BITM decreases. In Fig. 4(c) we have represented the reneging probability in a interval of length t according to Eq. 2. Figs. 4(b) and 4(c) suggest a trade-off must be considered in the selection of t_{max} because as we can observe in Fig. 4(b) the bandwidth required by BITM decreases with t_{max} and from Fig. 4(c) we can see that the reneging probability increases when t_{max} increases.

The aggregated bandwidth due to the replicated videos which are stored in each server is bounded by the number of streams that can concurrently be served in a local server, i.e.,

$$\sum_{i=1}^{s} B_i \leq N_{stream} \cdot B_{stream} \tag{11}$$

On the other hand, the aggregated bandwidth due to the remote services through the network is bounded by,

$$\sum_{i=s+1}^{M} B_i \leq min(B_{link}, \frac{B_{switch}}{N_{serv}}) \tag{12}$$

From Equations 9, 10, 11 and 12 we get a relationship between the system parameters (B_{stream}, B_{link}, B_{switch}, N_{serv}), the algorithm parameters (U_i, t_{max}) and the client behavior (λ_i). These equations could be used as guidelines to resize the network and servers in our system. We will explore this issue in future works.

III. Experimental results

In this section, we evaluate the performance of BITM in our distributed system through two metrics related with the QoS of the system: the reneging probability (measured as the ratio of clients that leave the system before they receive a service) and the waiting time (T_{wait}). We note that when T_{wait} is computed, the waiting times of the reneging requests will not be counted. On the other hand, remember that only the most popular videos, those that verify Equation 5, are considered to form mini-groups. In the experiments, our technique BITM is compared with the baseline version of the algorithm [1], where the requests are not delayed to form mini-groups, but the clients may leave the system.

A. Parameters of the Simulations

Every server has $N_{stream} = 50$ streams, and $N_{term} = 3000$ clients. Every client generates requests following an exponential distribution with a mean time of $T_{sleep} = 3600$ seconds. We set the parameters MRT and α to 600 seconds and 0.05 respectively. Therefore from equations 2 and 4, the maximum delay is $t_{max} = 60$ seconds.

The system offers $M = 100$ videos whose lengths are uniformly distributed in the interval [3600,10800] seconds and which are distributed as it is mentioned in the Introduction. To model the popularity of the videos we use the so-called pure Zipf law distribution [2], [3] with parameter $\xi = 1$. We analyze the system behavior with different replication degrees that vary from 75%, 50% and 25%.

For every experiment, the simulation time was over 4 hours.

B. Analysis of the results

In Fig. 5(a) we compare the waiting times for the different replication degrees. The X-axis represents the number of servers in the system, N_{serv}, and Y-axis gives the average waiting times in seconds. In Fig. 5(b) we have represented the

Fig. 4. (a) Optimal threshold and (b) The lowest upper bound of the average used bandwidth for the most popular video in BITM and the baseline approaches. We have considered, in both cases, the next parameters: $M = 100$ videos, length of the video 7200 sec, $\lambda_i = 0.102$ req/sec and $B_{stream} = 15$ Mbps; (c) Reneging probability $P_i(t)$: the fraction of requests for a video i in a interval of length t that leave the system.

total reneging ratio (**Total**) as well as the reneging ratio due to remote requests (it is, due to the Non Replicated videos **N.R.**) computed by both approaches with different number of servers and replication degrees.

In Fig. 5(a) we can see that in almost all the cases the waiting times of BITM are the smallest ones. This is because the most popular videos are delayed to form mini-groups, and this fact makes that the required bandwidth is smaller, as it was predicted in the analytical model. Therefore, this confirms that our BITM algorithm reduces the usage of the resources. Moreover, we can note that T_{wait} is almost constant whichever it is the replication degree. It is only for a 25% of replication that T_{wait} decreases from 10 servers in the system. This is due to two reasons. The first one is that for the 25% of replication, the third part of the total reneging ratio is mainly due to the non replicated videos (as we can see in Fig. 5(b)). As there are less remote requests, there are less competition by the interconnection network. Therefore the remote requests that don't leave the system wait less time. The second reason is related to the video distribution that we have assumed. With this distribution, if a system has a low number of servers (2, 4, 6 servers), these have to store more non replicated videos than the servers of a system with a larger number of them (10, 12, 14, 16 servers). Therefore, a server with few non replicated videos store less videos in total and, as a consequence, hasn't to share its local bandwidth among so many videos. So there is less contention by the local bandwidth and, therefore, the clients that don't leave the system wait less than the clients (in the same situation) of a system with few servers.

Now, from Fig. 5(b) we can note that the reneging ratio due to non replicated videos (**N.R.**) increases when the degree of replication decreases. With a low degree of replication (e.g. 25%) a system has to handle more remote services so there is a competition by the network resources. Therefore, the clients have to wait necessarily to get the service. But the more remote services, the more the clients have to wait and, as a consequence, the reneging ratio increases. The difference between the total reneging ratio and the reneging ratio due to non replicated videos is obviously, the reneging ratio due to requests to local videos. From the results, we deduce that with BITM the clients that request local videos renege less than if they are scheduled by the baseline approach. It is, delaying a few seconds the clients to form mini-groups produces that the

local bandwidth is optimized, and therefore more clients are served and don't leave the system.

Let's analyze the reneging behaviour in more detail. Now, Fig. 6 depicts the total ratio of clients that leave the system as well as the total ratio of clients that leave the system before 600 seconds (which is the Mean Reneging Time). In order to evaluate the performance of our algorithm, we have represented the analytical value of the reneging distribution function for 600 seconds ($F(600)$). The ratio of clients that leave before 600 seconds are very similar for both approaches and in addition, they fit the analytical value $F(600)$ in any case of replication. However, we can note that when the replication decreases, the total ratios of the baseline and BITM differ from 2-3% to 5-6%. This is due to the fact that BITM optimizes the bandwidth when some clients are delayed to be served with the same stream. Therefore, the rest of clients wait less (than MRT) because there are more resources. This leads to a decreasing of the total reneging ratio.

We can note also that the total reneging ratios are larger than the analytical value of $F(600)$. The reason is very simple: we have supposed that the clients leave the system following an exponential distribution of mean $MRT = 600$ seconds. Therefore, there are clients that leave the system before and after 600 seconds. As the total reneging ratio includes the clients that leave the system after MRT, this leads to total ratios that are bigger than the analytical value $F(600)$.

Another important fact is that the total reneging ratios decrease when the replication degree decreases in the system. This can be observed in the Fig. 6(c). The reason of this decreasing has just been explained with the decreasing of T_{wait}. With a low degree of replication there are less competition by the local resources (as there are less local videos for servicing), and therefore this leads to a lower reneging ratio due to the local videos, and as a consequence, to a lower total reneging ratio.

In other words, from the reneging behaviour point of view, the most appealing choice would be to design a distributed VoD system based on the bigger number of servers that verify the constraint expressed in Equation 12, and using a small replication degree. However a trade-off must be considered in the selection of the replication degree, because as we saw in Fig. 5(b) the less the percentage of replication, the

Num. Servers	75%		50%		25%		Reneging Ratios
	Basel.	BITM	Basel.	BITM	Basel.	BITM	
2	0.75	0.72	0.75	0.72	0.73	0.67	**Total**
	0.04	0.05	0.11	0.12	0.23	0.24	**N. R.**
4	0.75	0.72	0.74	0.69	0.72	0.65	**Total**
	0.04	0.05	0.11	0.12	0.23	0.24	**N. R.**
6	0.74	0.72	0.72	0.69	0.68	0.64	**Total**
	0.04	0.05	0.12	0.12	0.23	0.24	**N. R.**
8	0.74	0.72	0.71	0.68	0.68	0.63	**Total**
	0.04	0.05	0.12	0.12	0.24	0.24	**N. R.**
10	0.73	0.7	0.71	0.67	0.67	0.61	**Total**
	0.04	0.05	0.12	0.12	0.24	0.24	**N. R.**
12	0.72	0.69	0.7	0.66	0.66	0.61	**Total**
	0.05	0.05	0.12	0.12	0.24	0.24	**N. R.**
14	0.72	0.68	0.71	0.66	0.65	0.6	**Total**
	0.05	0.05	0.12	0.12	0.24	0.24	**N. R.**
16	0.72	0.69	0.7	0.66	0.65	0.6	**Total**
	0.05	0.05	0.12	0.12	0.24	0.24	**N. R.**

(a) (b)

Fig. 5.   Performance results for BITM and the baseline version. (a) The waiting times in seconds; The x-axis represents the number of servers in the system; and (b) Reneging ratios: the odd rows represent the total reneging ratios and the even rows the reneging ratio of the non replicated videos. The first two columns represent the ratios with a 75% of replication in the system; The next two, the ratios with a 50%; The last two ones, the corresponding ratios with a replication degree of 25%.

(a) (b) (c)

Fig. 6.   Reneging ratios: reneging ratio before 600 seconds and total reneging ratio; (a) Reneging ratios in a system with 75% of replication; (b) Reneging ratios in a system with 50% of replication; (c) Reneging ratios in a system with 25% of replication. F(600) represents the analytical mean value of the reneging distribution function.

more reneging for the non replicated videos requests. We will explore further these issues in future works.

IV. CONCLUSIONS

In this paper we have proposed a delivery strategy for a VoD distributed system. Through some preliminary results we have seen that delaying some clients during a short interval leads to optimize the usage of the resources whereas it is maintained a preset reneging probabilities. In addition, we have seen that BITM not only reduce both reneging probabilities and T_{wait}, but also improves the results obtained by the baseline algorithm. In other words, BITM improves the QoS of the system.

REFERENCES

[1] S. Gonzalez, A. Navarro, J. Lopez, and E. Zapata, "Two hybrid multicast algorithms for optimizing resources in a distributed vod system," in *10th International Multi-Media Modelling Conference*, Brisbane, Australia, 5-7 January 2004.

[2] G. Zipf, *Human Behavior and the Principle of Least Effort*. Cambridge, Massachussetts: Addison-Wesley, 1949.

[3] C. Aggarwal, J. Wolf, and P. Yu, "The maximum factor queue length batching scheme for video-on-demand systems," *IEEE Transactions on Computers*, vol. 50, no. 2, pp. 97–109, 2001.

[4] K. Hua, Y. Cai, and S. Sheu, "Patching: A multicast technique for true video-on-demand services," in *ACM Multimedia*, Sept. 1998.

[5] L. Gao and D. Towsley, "Supplying instantaneous video-on-demand services using controlled multicast," in *IEEE ICMCS*, Florence, Italy, June 1999, pp. 117–121.

[6] L. Gao and D. Towsley, "Threshold-based multicast for continuous media delivery," *IEEE Transactions on Multimedia*, vol. 3, no. 4, pp. 405–414, December 2001.

[7] D. Eager, M. Vernon, and J. Zahorjan, "Optimal and efficient merging schedules for video-on-demand servers," in *7th ACM International Multimedia Conference*, Orlando, Florida, November 1999, pp. 199–202.

[8] J. E. G. Coffman, P. Jelenkovic, and P. Momcilovic, "The diadyc stream merging algorithm," *Journal of Algorithms*, vol. 43, pp. 120–137, 2002.

[9] W.-F. Poon, K.-T. Lo, and J. Feng, "A hybrid delivery strategy for a video-on-demand system with customer reneging behavior," *IEEE Trans. on Broadcasting*, vol. 48, no. 2, pp. 140–150, 2002.

[10] D. Gross and C. Harris, *Fundamentals of Queueing Theory*, 3rd ed. Wiley, 1998.

[11] A. Dan, P. Shahabuddin, D. Sitaram, and D. Towsley, "Channel allocation under batching and vcr control in video-on-demand systems," *Journal of Parallel and Distributed Computing*, vol. 30, pp. 168–179, 1995.

A Case for A Mobility Based Admission Control Policy

Shahram Ghandeharizadeh[1], Tooraj Helmi[1], Shyam Kapadia[1], Bhaskar Krishnamachari[1,2]
[1]Department of Computer Science
[2]Department of Electrical Engineering
University of Southern California
Los Angeles, CA 90089, USA

ABSTRACT

Ad hoc networks of wireless devices such as Car-to-Car Peer-to-Peer Networks (C2P2) are an emerging technology. Entertainment applications that manipulate continuous media, audio and video clips, push the limits of these networks due to their stringent requirements. These challenges are magnified by environmental characteristics such as dynamic wireless network connections, mobility, multi-hop nature of the network and the possible lack of a fixed infrastructure. In these networks, an intelligent admission control policy enhances Quality of Service (QoS) observed by the users of the system. This paper makes the case for a decentralized Mobility based ADmission Control (MAD) policy. We develop QoS utility models to quantify the performance of this policy with an environment that employs no-admission control. Obtained results show conclusively that MAD provides orders of magnitude improvement when compared with no admission control.

1. INTRODUCTION

During the past few years, automobile manufacturers have been marketing and selling vehicles equipped with entertainment systems. These systems typically consist of a DVD player, a fold-down screen, a video game console, and wireless headphones. In its present form, storage and content are tied together. This limits the number of available titles to those DVDs and CDs in the vehicle. We envision a separation of storage and content where content is staged on demand across the available storage for previewing. This would provide passengers access to a large repository of titles. In this vision, vehicles are equipped with car-to-car, peer-to-peer (C2P2) devices which form a mobile ad-hoc network (MANET). Each C2P2 is equipped with abundant amount of storage, a processor, and a wireless networking card.

The principle characteristic of continuous media such as video and audio is their high sustained bit rate requirement. If a system delivers a clip at a rate lower than its pre-specified rate without special precautions (e.g., pre-fetching), the user might observe frequent disruptions and delays with video and random noises with audio. These artifacts are collectively termed *hiccups*. A hiccup-free display is the ideal quality of service (QoS) provided to an end user. A second QoS criterion is the observed startup latency, defined as the delay observed from when a user references a clip to the onset of display.

All C2P2 devices may replicate popular titles, but they must collaborate in order to provide users with a large selection of titles. Each C2P2 may contribute a fraction of its storage in a peer-to-peer manner to be occupied by the system-assigned clips. These clips might be referenced by users of those mobile C2P2 devices that are network reachable (including the local user). A C2P2 must offer its passengers only those clips that it can download and play in a hiccup-free manner within a reasonable amount of time (e.g. by satisfying a maximum tolerable startup latency constraint).

Techniques that address these challenges are impacted by the characteristics of the MANET, placement and delivery scheduling of data, and admission control policies in support of a hiccup-free display. Consider each in turn. Mobility, a key MANET characteristic, is the primary challenge because it dictates the life-time of paths between a producer and a consumer of data. When the display of data is overlapped with its delivery, a path repair may incur delays that result in data starvation and hiccup. In addition, topology changes impact the availability of both data and bandwidth. The availability of data is also impacted by its placement across the nodes and its degree of replication. One may replicate data at the granularity of either a clip [1] or a block [2].

Finally, given the high data-rate requirements of continuous media displays and limited bandwidth resources, the system must be configured with an admission control policy to handle multiple simultaneous clip downloads. In particular, the admission control policy must strive to do a "good job" of rejecting those requests that are unlikely to be satisfied within a reasonable amount of time. Section 3 formalizes and quantifies this quali-

tative metric by providing utility models for QoS that consider the number of rejected requests, the number of admitted requests that are successful, and the number of admitted requests that fail to meet a specified startup latency constraint.

Compared to traditional admission control techniques, the C2P2 MANET environment offers the following unique challenges. First the network topology is dynamic. Second the admission control must be performed in a distributed manner as there is no central coordination point for the network. Further, it is hard to estimate the resources that will be available for a clip download (since the topology will change during the download). The key contribution of this paper is the design and analysis of a simple mobility based admission control policy using C2P2/MANET simulation studies. While admission control is well studied in wired networks [4, 8], to the best of our knowledge, besides our prior work with stationary wireless devices [2, 1], there is little if any prior work on this topic.

Recent studies addressing QoS in MANETs are as follows. A survey of QoS metrics in MANETs is provided by [10]. These studies either build on top of the existing routing protocols [11, 6] like DSR, AODV, etc., or integrate the QoS metrics into the routing protocol [5, 9, 12]. There have been studies on both unicast as well as multicast routing. Our primary contribution is to extend the conventional definition of admission control and introduce a simple mobility based admission control policy to study the number of rejected, satisfied and unsatisfied requests in a MANET (C2P2 network).

The rest of this paper is organized as follows. Section 2 describes the framework for the admission control problem and introduces the mobility based admission control policy. In Section 3, we show the performance observed in environments deployed with and without MAD. Due to space limitations a comparison of MAD with other admission control techniques is not presented here. Our conclusions are presented in Section 4.

2. ADMISSION CONTROL

Section 2.1 formalizes admission control as the process of admitting those requests that are able to download a fix-sized file within a specified duration of time. Section 2.2 develops the framework and describes our mobility based admission control policy.

2.1 Introduction and Problem Statement

While it is possible to overlap the delivery and display of continuous media clips using buffering, this strategy runs the risk of incurring hiccups if network resources becomes temporarily unavailable during delivery. In this work, we will focus our attention on the delivery of clips that will be played only after the entire clip is downloaded. We will assume that each request specifies the file size of the clip to be downloaded and a maximum download time (i.e. startup latency).

We formulate the admission control problem as follows: A request to display a clip X requires all blocks of X be downloaded to the requesting C2P2 (denoted as $C2P2_d$) within T_X time units. Note that T_X (the download time or the startup latency) can be significantly smaller than the playback time of the clip X. For example, an audio clip X with a display time of 12 minutes might specify T_X as 1 minute. This means that all blocks of this clip must be materialized in one minute in order for a request referencing X to be admitted to the system. Here, once a request is issued, the decision to either reject or admit the request must be made almost immediately after the request is issued (Instantaneous mode of operation).

Thus, assuming K requests are issued to a system, an admission control policy yield three classes of requests. First, a fixed number of admitted and rejected requests, termed A and R, respectively. Note $K = A + R$. Second, some of the admitted requests might fail, denoted as A_f, $A_f \leq A$. And, finally, some of the admitted requests are serviced successfully, denoted as A_g, where $A = A_g + A_f$. An admission control policy should strive to maximize both A and A_g (ideally, to a maximum of K). By maximizing A, a larger percentage of requests are processed by the C2P2 MANET. By maximizing A_g, a larger percentage of admitted requests are processed successfully. Section 3 presents QoS models that define a policy's utility as a weighted combination of these raw metrics.

2.2 Mobility-based ADmission Control (MAD)

We now propose a framework for solving the admission control problem in a mobile ad-hoc network. We focus on distributed, client-centric, admission control policies that enable the C2P2 client device to either admit or reject a request. For the rest of this paper, $C2P2_d$ is the device with a pending display request.

In order to make an intelligent admission decision, the admission control component of $C2P2_d$ must first obtain information about the state of the network and availability of resources. This information is then weighed against the specifications of the request in order to determine if the request can be admitted. Here, we present the results for the mobility based admission control policy.

In a mobile ad-hoc network such as C2P2, knowledge of the current bandwidth on the path between a client and server is insufficient to determine if a given request can be satisfied. This is because the available bandwidth changes dynamically and might be either greater or lower in the future (due to unpredictable mobility-induced changes in the network structure and network traffic). In order to contend with the dynamic nature of a C2P2 network, flexible but simple admission control policies must be designed that consider different kinds of information about the current state of the network, and try to predict/estimate based on this information whether the required resources for a given request can

be provided. In our admission control framework, we develop two kinds of metrics: a Request Metric (R_m), and an Information Metric (I_m). The request metric R_m (which is normalized to be a number between 0 and 100) is a measure of the amount of resources required for a given request. Hence it depends on the QoS parameter of interest. In particular, we define R_m as the ratio of the requested bandwidth over the nominal maximum wireless link bandwidth B_{max} i.e. $R_m = \frac{\frac{size(X)}{T_X}}{B_{max}} \times 100$

where $size(X)$ is the size of the clip in Mb (MB), and B_{max} is also defined in Mbps (MBps).

While R_m is not policy-dependent, the information metric I_m is, and represents quantitatively the information obtained by the client about the current and future resources available in the network. The information metric is normalized to be a number between 0 and 100. I_m is monotonic in the estimated resource availability — the higher the value of I_m, the more likely it should be that a given request can be satisfied. Note that both I_m and R_m have no units. Now the admission control decision is simply a matter of comparing the two metrics: I_m and R_m, and deciding whether the given I_m predicts sufficient availability of resources for the request with metric R_m. A simple yet flexible form for the admission control policy is simply to take the difference between I_m and R_m and compare with a threshold θ. Thus when $I_m - R_m > \theta$, a policy admits the corresponding request. Otherwise, the request is rejected.

Note that based on our definitions, the threshold can potentially take on values from -100 (which corresponds to allowing all requests) to 100 (which corresponds to rejecting all requests). Thus if θ is too low, there is a danger of greater unsatisfied requests due to bandwidth contention. On the other hand, if it is too high, the number of satisfied requests will be low because too few requests are accepted. The optimal choice of threshold may be an intermediate value and could be scenario-dependent as we shall see in our experiments. Of course, the optimal choice of threshold is also policy-dependent, since I_m is defined differently for each policy. When a request is issued by a client, a policy must either accept or reject requests using the threshold computation.

Content replication and switched proximate server selection are important for robust delivery of high-rate content. The client always chooses the closest server from S candidate servers. When C2P2$_d$'s path to a server breaks at time T_Z, C2P2$_d$ might have downloaded Z bytes of a clip X, $0 \leq Z \leq size(X)$. The bandwidth required to download the remainder of a clip is $\frac{size(X) - Z}{T_X - T_Z}$. If this bandwidth exceeds the wireless network bandwidth, the request is discarded as a failed request. Otherwise, C2P2$_d$ identifies another reachable server and begins downloading the remainder of the clip from that server.

The Mobility-based ADmission control (MAD) policy tries to leverage the mobility pattern in the network by estimating the duration of time a server s_i will be in the radio range of the requesting client C2P2$_d$. This depends on both the direction and speed of s_i and C2P2$_d$. C2PD$_d$ sends out a message (probe) with a certain time to live to obtain the list of proximate servers s_i. Assuming $T(s_i, C2P2_d)$ denotes the duration of time s_i and C2P2$_d$ are expected to be in radio range, for MAD: $I_m = \frac{\sum_{i=1}^{S} T(s_i, C2P2_d)}{\hat{N} T_X} \times 100$. Here, $\hat{N}$ is the total number of nodes reachable from the client via the probe. Note that both the numerator and denominator are in units of time.

This policy only considers the time that a server is within radio range of C2P2$_d$. This is conservative because it ignores the time the servers are reachable via multiple hops.

3. PERFORMANCE EVALUATION

In this section, we quantify the performance of MAD and its comparison to the no-admission control case. We conducted many experiments with different parameter settings. In order to summarize the lessons learnt, we developed utility models to summarize the QoS observed with each policy in one number. These models are presented in Section 3.1. Section 3.2 presents our experimental environment. Section 3.3 demonstrates superiority of MAD to an environment without admission control.

3.1 Utility models for QoS

This section describes three utility models to quantify the QoS metric.

Recall that an admission control policy divides the total number of requests into three groups: R rejected requests, A_f failed admitted requests (due to unsatisfied requirements), and A_g successfully served admitted requests. Different users may value each of these differently, as per their utility model. For the purpose of evaluating our experimental results we consider three distinct utility models that are representative of different levels of QoS that can be provided in a C2P2 ad hoc network. In each model M, we define the joint utility U as a weighted sum:

$$U_M = w_R(M)R + w_{A_g}(M)A_g + w_{A_f}(M)A_f$$

where $w_i(M)$ indicates the weight of component i in model M.

Table 1 shows the weights for three QoS models employed to evaluate the experimental results. The economy model does not penalize a policy for either rejected or failed admitted requests and only rewards accepted requests. The Standard model is indifferent to rejections, but penalizes failed admitted requests as much as it rewards successfully served admitted requests. The Premium model resembles a high QoS environment. It penalizes rejected requests (albeit slightly), rewards admitted requests that are satisfied, and highly penalizes admitted requests that fail. The high penalty for a failed admitted request reflects our intuition that users

Models	Weight of a rejected request, $w_R(\mathrm{M})$	Weight of a successfully admitted request, $w_{A_g}(\mathrm{M})$	Weight of a failed admitted request, $w_{A_f}(\mathrm{M})$
Economy	0	1	0
Standard	0	1	-1
Premium	-0.5	1	-5

Table 1: Three utility models to quantify quality of service with alternative policies.

Figure 1: Thirteen road stretches with 3 stationary servers.

are likely to be annoyed if their C2P2 device initiates a download, spends a considerable amount of time (maybe T_X time units) only to discover that it cannot display a clip because an incomplete file is downloaded.

While these utility models can be applied on a per-user basis and different QoS levels can be integrated within the same system in practice, for the purposes of analyzing our experimental result and comparing the different admission control policies, we shall apply each QoS utility model to the system as a whole for the duration of the experiment.

3.2 Experimental Environment

The results presented in this section are based on a new simulator for C2P2 networks written using the $\mathrm{C}^{\#}$ programming language. The simulator models road stretches and cars that navigate these stretches. A car might be configured with a C2P2 device that provides a fixed amount of network bandwidth. Each C2P2 device implements the DSR [3] routing policy. However, we believe that a choice of the routing protocol does not impact the trends in the observed results. A proactive protocol like DSDV [7] would have shown similar results. We analyzed different scenarios consisting of a different number of road-stretches, different number of C2P2-equipped cars, mobility patterns, and request specifications.

Our basic experiment consists of thirteen bi-directional road-stretches, numbered from 0 to 12, see Figure 1. Three stationary servers (these could be parked vehicles, access points, or base stations) are located on the following road-stretches: 1, 6, and 11, numbered s_0, s_1, and s_2, respectively. The radio-range of each C2P2 spans 3 road-stretches: its current road stretch and its two adjacent road stretches. Thus, a C2P2 on either road-stretch 0 or 2 is in the radio range of s_0 which is on road-stretch 1. Note that there exists four road stretches that are dark because they are not 1-hop reachable by a server. However, given sufficient number of clients and transitive routing of packets across these clients, these

dark road stretches might lit-up as they become network reachable to either one or two servers. We analyzed configurations consisting of 13 possible client C2P2 nodes (in addition to the 3 server nodes), with the simultaneous active displays ranging from 1 to 13 in our experiments. Initially, all cars are assigned to the road-stretches such that they are evenly spaced across the road-stretches. For example, with 13 cars, one car is assigned to each road-stretch. The initial direction of each car is chosen randomly (i.e to move left or right). The speed of each car is fixed at 5 meters per second. Once a car reaches the end of either road stretch 0 or 12, it switches directions and moves toward the opposite end. Within each configuration, the cars that are requesting clips (corresponding to the number of active displays) are chosen randomly and they re-issue requests periodically every 60 seconds.

A total of 100 requests are issued in each experiment. This means that with 10 active displays, each client issues 10 requests on average. All three servers are assumed to contain a replica of the clips being requested. Rejected requests do not consume any network bandwidth and are terminated immediately. Once a request is admitted to the system, it downloads a clip for 60 seconds. The maximum network bandwidth of a C2P2 device is assumed to be 10 Mbps. The referenced clip is a media clip with a bandwidth requirement of 340 Kbps. The display time of each clip is fixed at 12 minutes. We require the clip (size ~30MB) to be downloaded in 60 seconds, requiring a download bandwidth requirement of 4 Mbps.

3.3 A Case for Admission Control

We start with experimental results that justify the use of intelligent admission control policies in an ad-hoc network of C2P2 devices. Figure 2 shows utility of MAD with alternative models when compared with an environment that does not utilize an admission control policy. Figure 3 shows the number of rejected, satisfied, and unsatisfied requests for both MAD and the no admission control case. Without admission control, the

2.a) Premium

2.b) Standard

2.c) Economy

Figure 2: A comparison of Instantaneous-MAD with an environment that employs no admission control. The environment consists of 16 C2P2 devices: 3 stationary servers and 13 mobile clients.

simulator admits a request upon its arrival independent of server availability. Figure 2 shows the performance of MAD for two different thresholds: -20 and -40. The x-axis of this figure shows the number of concurrent clients that initiate the display of a clip, termed number of active displays. This controls the load in the environment and is increased from 1 to 13. The y-axis of this figure is the utility of each model. We present results for all models shown in Table 1. Each presented data point is an average of results obtained from 10 experiments utilizing different seeds. (The random seed impacts the order and identity of clients issuing requests.)

With 3 stationary servers and a maximum wireless bandwidth of 10 Mbps per C2P2 device, our experimental environment supports a total bandwidth of 30 Mbps. All active clients issue their requests at the same time. A total of 100 requests are issued by all clients in this environment. This implies that with two or fewer number of active displays, if there were either no dark regions or unpredictability due to mobility, without admission control, a total of 100 requests would be satisfied.

Figure 3.b shows that only one half of requests are served successfully with either 2 or fewer active displays. This is because of mobility and dark regions that cause an active client to starve for data, failing to download a clip X in T_X time units. This is reflected in the number of unsatisfied requests admitted to the system, see Figure 3.c. A primary observation from Figures 2 and 3 is that an environment with no admission control is clearly inferior to Instantaneous-MAD with θ=-40.

With the Premium model, MAD is superior by several orders of magnitude. With the Standard utility model, MAD remains superior. With the Economy model, however, no admission control outperforms MAD when θ=-20. This threshold renders MAD too conservative, forcing it to admit too few requests.

While all these requests are processed successfully, see Figure 3.c where A_f=0 with θ=-20, MAD does reject requests that can be processes successfully. With a more relaxed threshold (-40), these requests are admitted into the system, enabling MAD to outperform the environment with no-admission control when using the Economy model.

Note that utility of MAD with θ=-20 is a constant positive with the Economy, Standard, and Premium models. When θ=-40, MAD's utility drops as a function of load because the number of unsatisfied requests increases with this threshold, see Figure 3.c.

4. CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

The primary contribution of this paper is a mobility based admission control policy (MAD) to download a continuous media clip in an ad-hoc network of C2P2 devices. Our experimental results demonstrate an environment that employs admission control is superior to

3.a) Rejected

3.b) Successful Admit

3.c) Failed Admit

Figure 3: Rejected (R), successfully admitted (A_g), and failed admitted (A_f) requests with different number of active displays and 16 C2P2 devices: 3 stationary servers, and 13 mobile clients.

one without admission control. MAD can be extended to consider the available bandwidth at each server. We are currently running extensive simulations for such a policy. An obvious research direction is to explore the streaming of a clip across the C2P2 devices. This adds an additional metric which considers the number of hiccups and their duration.

5. ACKNOWLEDGMENTS

6. REFERENCES

[1] S. Ghandeharizadeh and T. Helmi. An Evaluation of Alternative Continuous Media Replication Techniques in Wireless Peer-to-Peer Networks. In *Third International ACM Workshop on Data Engineering for Wireless and Mobile Access (MobiDE, in conjunction with MobiCom'03)*, September 2003.

[2] S. Ghandeharizadeh, B. Krishnamachari, and S. Song. Placement of Continuous Media in Wireless Peer-to-Peer Networks. *IEEE Transactions on Multimedia*, April 2004.

[3] D. B Johnson and D. A Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.

[4] E. Knightly and N. Shroff. Admission Control for Statistical QoS: Theory and Practice. In *IEEE Network, vol. 13, no. 2, pp. 20–29*, 1999.

[5] S. Lee, G. Ahn, X. Zhang, and A. Campbell. INSIGNIA: An IP-Based Quality of Service Framework for Mobile ad Hoc Networks. *Journal of Parallel and Distributed Computing*, 60(4):374–406, 2000.

[6] E. Pagani and G. Rossi. A framework for the Admission Control of QoS multicast traffic in mobile ad hoc networks. In *Proceedings of the 4th ACM international workshop on Wireless mobile multimedia*, pages 2–11. ACM Press, 2001.

[7] Charles E. Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In *Proceedings of the conference on Communications architectures, protocols and applications*, pages 234–244. ACM Press, 1994.

[8] M. Rahin and M. Kara. Call Admission Control Algorithms in ATM Networks: A Performance Comparison and Research Directions. In *Interim Research Report Draft*, 1998.

[9] P. Sinha, R. Sivakumar, and V. Bharghavan. CEDAR: a Core-Extraction Distributed Ad Hoc Routing Algorithm. In *INFOCOM (1)*, pages 202–209, 1999.

[10] K. Wu and J. Harms. QoS Support in Mobile Ad Hoc Networks. In *Crossing Boundaries- an interdisciplinary journal, vol 1, no 1*, Aug 1998.

[11] H. Xiao, W. Seah, A. Lo, and K. Chua. A Flexible Quality of Service Model for Mobile Ad-Hoc Networks. In *IEEE VTC*, 2000.

[12] C. Zhu and M. Corson. QoS routing for mobile ad hoc networks. In *IEEE Infocom*, June 2001.

The COMODIN System: A CDN-based Platform for Cooperative Media On-Demand on the InterNet[*]

Giancarlo Fortino[1], Carlos E. Palau[2], Wilma Russo[1], Manuel Esteve[2]

[1] Dipartimento di Elettronica, Informatica e Sistemistica (DEIS), Università della Calabria
Via P. Bucci cubo 41C, 87036 Rende (CS), Italy
[2] Departamento de Comunicaciones, Universidad Politecnica de Valencia
Camino de Vera s/n, 46022 Valencia, Spain
e-mail: {g.fortino, w.russo}unical.it, {mesteve, cpalau}@dcom.upv.es

Abstract

In this paper we present the COMODIN (COoperative Media On-Demand on the InterNet) system, a media on-demand platform for synchronous cooperative work, which is aimed at securely supporting an explicitly-formed cooperative group of distributed users for (i) requesting an archived multimedia session, (ii) sharing the playback, and (iii) collaborating with each other through questioning and, optionally, synchronous audio/video tools.

At the server-side, the system is based on an IP-multicast-enabled Content Distribution Network, which consists of a coordinated set of heterogeneous and distributed multimedia servers, whereas, at the client-side, the system centers on Java-based applications and applets which interface the cooperative group of users.

Significant applications featured by the COMODIN system encompass Collaborative Learning on-Demand and Distributed Virtual Theaters.

1. Introduction and Motivations

Multimedia streaming services and systems on the Internet are currently fostered by: (i) advances in IP-compliant communication infrastructures (e.g. last-mile, 3G cellular, and satellite networks) [5]; (ii) standard multimedia internetworking formats (e.g., H.261, MPEG), protocols (e.g., RTP/RTCP, RTSP, SIP) and architectures [4]; (iii) commercial interests of public and private companies foreseeing a new, easily reachable market for entertainment.

Thus, nowadays, the Internet-based Media on-Demand (MoD) can fulfill the expectations that were not accomplished by proprietary and highly-expensive Video on-Demand (VoD) systems.

The distinctive key points [16] of the Internet-based MoD are: re-use of existing infrastructures, flexible communication media (e.g., modem, wireless, cable, ATM, LAN, satellite), one service among many others (reverse economics from traditional VoD), quality scalability (from hand-held devices to HDTV), adaptive compression, easy integration with WWW, easy integration with recording functionalities, security through encryption, cheap authoring, a potential lot of contents.

Several Internet-based MoD systems, some of which are also able to record live multimedia streams, have been developed and are currently available:

- *research oriented systems*: IMJ [1], MASH Rover [17], ViCRO^C [9, 10], mMoD [14], MVoD [12], KOM-Player [19];
- *commercial systems*: RealNetworks Media System, Windows Media System, Darwin Streaming Server [6].

However, in the very last years the research focus on Internet-based MoD has been shifting from ad-hoc built systems, such as the above mentioned ones, to Content Distribution Networks (CDNs), which can more efficiently provide diversified media services ranging from TV broadcast to VoD. CDNs are overlay networks on top of the Internet that consist of a set of

---

coordinated distributed servers that are located close to users for the provision of contents in a low-latency fashion [18].

The aforementioned MoD systems basically provide (i) single users with point-to-point transmission and control of selected media sessions and (ii) multiple users with point-to-multipoint transmission of selected media sessions. In the latter, if a user wishes to control the transmission of a media session, which is also watched by other users, the following policies can be adopted:

- *session splitting and proprietary control*: the media session is cloned and transmitted to the user on a different address; users can thus control their own session [1];
- *session control based on the initiator*: only the initiator of the session is qualified to control the session; the others are only passive viewers [12,14,19];
- *shared control of the session*: any control action induces a session state change affecting all session participants [9, 10, 17].

The latter policy is particularly interesting in that it can enable *cooperative playback sessions* in which a group of user cooperatively controls and watches a played-back media session. The MASH Rover system [17] and, particularly, the ViCRO^C system [9, 10] were built to support cooperative playback sessions. However, the main issue preventing their diffusion and exploitation is that they are strongly-based on IP-multicast which is still not widely deployed on the global Internet.

This paper describes the models, protocols and architecture of the COMODIN (COoperative Media On-Demand on the InterNet) system, a media on-demand platform which is able to provide cooperative playback services [11] on the current Internet infrastructure, exploiting IP-multicast only where available.

The system securely support an explicitly-formed cooperative group of distributed users for (i) requesting an archived multimedia session, (ii) sharing the playback, (iii) collaborating with each other through questioning and, optionally, synchronous a/v tools.

The server-side of the COMODIN system is built atop a CDN for media-streaming [13, 15] which was designed according to the architecture of the PRISM system [3] and implemented using Java technologies (e.g. Java Media Framework) and the Darwin Streaming Server [6].

The client-side of the COMODIN system (or COMODIN Client), which was completely implemented using Java and the Java Media Framework, consists of a Swing-based multimedia GUI

supported by an application or by an applet. It allows an user to securely organize a cooperative playback session, play the media session, control the session playback using a VCR panel, and exchange questions with the other users of the group.

The remainder of the paper is organized as follows. In §2 the cooperative playback service model and the related design guidelines are presented; then, a cooperative playback control protocol is defined. In §3 the main components of the server-side part of the COMODIN system are described whereas in §4 the GUI of the client-side part along with some application domains is explained. Finally, conclusions are drawn and on-going research activities discussed.

2. The Cooperative Playback Service

2.1. The Cooperative Playback Service Model

The cooperative playback service [9, 10] allows an explicitly-formed group of clients to cooperatively share the control of the streaming of a multimedia session being played back. It works according to a multicast client/server model which has been in depth analyzed through simulation [8].

Figure 1. Abstract reference schema of the cooperative playback service model

Figure 1 reports the schema of the model. While the Media Streaming Server (MSS) transmits the media playback to all the members of the group, each member can send control commands (e.g., PAUSE, PLAY, SEEK, etc.) which can affect the playback session state of the whole group.

In particular, the MSS holds the session state and changes it each time a control command is accepted. The control command issued by a member is considered to be provisional in that it is the reply of the MSS that modifies the session state. The MSS announces the new state to all the members according to a sort of soft-state paradigm [17], i.e., the session can continue even though some member is not aware of the state change. The *out-filter* is a virtual gate through

which control commands are routed from the Synchronous Cooperative Group (SCG) to the control point of the MSS. The *out-filter* models a specific coordination policy among the clients which can be token-based, voting-enabled or random [8]. The MSS sends control replies and media streams to *in-gate* points which represent mechanisms (e.g., IP-multicast) allowing to multicast messages to group members. In addition, members can interact with each other using specific tools (e.g., question board, audio/video conferencing).

2.2. Design Guidelines

A system implementing the model introduced above should be equipped with the following functionality:

- *Group Organization*, which contains group formation and group management. The former enables the creation of a group of users wishing to work on and control the same playback session. The latter deals with the following issues (i) how to share the starting time of a playback session among the group members; (ii) how to expel from the group a member who interrupts the others using an improper behavior.
- *Media Streaming*, which transmits media content to the group members. It is based on a media transport protocol (e.g., RTP - Realtime Transport Protocol [4]), streaming agents and a multimedia archive. The multimedia archive is a distributed media repository which contains both multimedia sessions dumped from the network and media files (e.g., MPEG files).
- *Cooperative Streaming Control*, which allows the group members to share the control of the playback. Normally, this functionality is embedded in the Archive Control Protocol (ACP) on which the C/S control interaction relies. Control messages are typical commands of a VCR remote controller such as PLAY, PAUSE, STOP, SEEK (i.e., a PLAY with a time range).
- *Joint Work,* which is usually in the form of questioning and annotation. Questioning means that the members of a group can exchange questions and answers about the content of the playback session. Annotation allows tagging a particular point in the session for a discussion proposal, which can be started during or at the end of the playback. Additional synchronous collaborative tools can also be used (e.g. the MBone tools [4])
- *Security*, which provides private cooperative playback sessions through authentication, key distribution, and media encryption.
- *Fault Tolerance*, which includes detection, which deals with the discovery of failures at both

server and client side, and recovery, which involves actions the server and/or the clients have to carry out in order to cope with detected failures.

2.3. The Cooperative Playback Control Protocol (CPCP)

CPCP is a specialization of the COoperative COntrol Protocol (COCOP) [8] which allows to cooperatively send control commands using an *implicit coordination mechanism* among clients with final contention resolution based on the server.

CPCP is currently implemented in Java and can be adapted both to TCP/reliable UDP-based unicast connections and to multicast channels based on the Lightweight Reliable Multicast Protocol (LRMP).

The high-level definition of the REQUEST and REPLY messages exchanged between clients and the server are:

- REQUEST=[T_i|C_K|REQtype], where T_i is the request transmission time, C_K is the identifier of the client issuing the request, and REQtype is the type of request: {PAUSE, PLAY, SEEK *stime*, STOP}.
- REPLY=[T_i|C_K|T_{Ck}|REPopt], where T_i is the reply transmission time, C_K is the identifier of the client issuing the request related to the reply, T_{Ck} is the transmission time of the request related to the reply, and REPopt is a field containing optional data.

The global requests are partially ordered whereas the requests issued by a single client are totally ordered. The session time is kept by the server so that the reply transmission time embodies the instant in which a session state change occurred. The replies are totally ordered.

Figure 2. (a) the client and (b) the server automata of the CPCP

The client and server automata of CPCP are reported in Figure 2.

In the *Ready* state, the client process (or client) can (i) accept a request (UsrReq) from the local user in order to forward the corresponding client request to the multicast group, (ii) sense a client request (Request) sent from a remote client so performing the *implicit coordination mechanism*, (iii) process a server reply (Reply). In the *RequestDone* state, the client ignores other Request and, once it receives a Reply, it passes into the *ProcessDone* state and sets the timer T_C, disabling the user to perform new requests. The client gets *Ready* again after receiving the timer expiration event (FTimer).

In the *Ready* state, the server process (or server) is available to receive and process a Request. After processing the Request and sending the Reply, the server rests inactive for an amount of time T_S which is introduced both to make the clients aware of the session state change, and to regulate the session interactivity. In fact, the group has to wait at least T_S to be able to get another request accepted. In the *ProcessDone* state, the server refuses all incoming Requests. As soon as the timer expires, the server gets *Ready* again. Usually, T_S is set at session set-up whereas T_C is dimensioned upon T_S.

3. The Server-side of the COMODIN system

3.1. The CDN for media streaming

The architectural schema of the CDN for media streaming which constitutes the media streaming platform of the COMODIN system is portrayed in Figure 3. The CDN architecture was inspired by the architecture of the PRISM system [3].

Figure 3. Architectural schema of the media streaming CDN

A client begins a media session by issuing a request to the CDN, the REDIRECTOR component captures the request and, after interacting with the CONTENT LOCATOR component, re-routes the request to the proper MEDIA STREAMING SERVER (MSS) component on the basis of closeness to the client and load conditions. The client can connect to the MSS for receiving and controlling the media session. An MSS can be either the Darwin Streaming Server [6] or a Java-based streaming server [7]. The latter integrates two advanced modules:

- *Secure Streamer Module (SSM)*, which transmits RTP-based MPEG streams encrypted using the Blowfish symmetric encryption technique (see [11] for details);
- *Media Flow Adapter Module (MFAM)*, which continuously adapts the media flow to the client network conditions by means of a rate controller [2].

The CONTENT MANAGER component can redirect a client to a better MSS if the client network conditions worsen too much or the load of the MSS becomes too high.

Cooperative Playback Control Protocol ⟷   Media Streaming →
FE = Unicast Front-End, MFE = Multicast Front-End
UMS = Unicast Media Streamer, USC = Unicast Streaming Controller
MMS = Multicast Media Streamer, MSC = Multicast Streaming Controller
MG = Media Streaming Multicast Group, CG = Streaming Control Multicast Group
CCC = Control Coordination Channel

Figure 4. CDN-based cooperative playback schemas

3.2. Provision of Cooperative Playbacks

In order to provide cooperative playback sessions, the architecture of the CDN for media streaming was enhanced by adding software modules incorporating CPCP (see §2.3) and coordination protocols.

In particular, given that the client C_i is already connected to the MSS_K (see Fig. 4a), when a client C_j wishes to cooperatively control the same cooperative playback session X (CPS_X) of C_i, it sends a request to the CDN for joining CPS_X. The CDN can satisfy the request in several ways, mainly according to the type of access network of the client C_j. Three solutions were implemented:

- *IP-multicast-based* (Fig. 4b): if C_j is on the same access network of C_i and the access network is IP-multicast-enabled, C_j is redirected to and served by MSS_k. In particular, since the Front-End (FE_{Ci}) of MSS_k is serving C_i using a unicast connection, another FE (MFE_{Cij}) is created which can serve C_i and C_j using IP multicast. So, C_i and C_j can receive the same media streams on the multicast group MG and share the media streaming control by sending control commands to the multicast group CG. A FE is composed by a Media Streamer (MS) and a Streaming Controller (SC).

- *Unicast-based* (Fig. 4c): if C_j is on the same access network of C_i and the access network is unicast, C_j is redirected to and served by MSS_k. In particular, another FE (FE_{Cj}) is created and synchronized with FE_{Ci}. C_i and C_j can therefore receive the same duplicated media streams and share their control by sending unicast control commands to their respective FEs which are coordinated with each other by the local Control Coordination Channel (CCC), which allows for the synchronization of the distributed Streaming Controllers (SCs) belonging to the same CPS. If C_j is on a different access network of C_i and the access network is unicast, C_j can be redirected to and served by MSS_w. The same steps as in the previous case are

carried out. The difference is that the CCC is distributed between MSS_k and MSS_w.

- *Hybrid* (Fig. 4d): if C_q requests to join CPS_X, it can be redirected to MSS_k, MSS_w or MSS_t. The multicast or unicast solutions can be used according to the type of access network of C_q. If C_q is redirected to MSS_t, the unicast solution is used. In all cases, the FEs interact through a distributed CCC.

3.2.1. Synchronization of the Streaming Controllers.
Given a set of streaming controllers $\{SC_1,\ldots,SC_M\}$ belonging to the same CPS, when a client C_i sends a REQUEST to SC_j to which it is connected, SC_j, upon reception of the REQUEST, starts synchronizing with the other SCs. After the synchronization phase, the SC, SC_j or another SC, which received a REQUEST simultaneously with respect to SC_j, is chosen to change the playback session state.

The SCs are connected by CCC, a reliable multicast channel through which SCs exchange synchronization, REQUEST and REPLY messages.

The synchronization phase is carried out as follows. Upon reception of a REQUEST, SC_j sends the SYNCHRO message containing a random number in the range [0..1] onto CCC. The receiving SCs, that have not received any REQUEST, reply with the OK message and suspend themselves, waiting for a REPLY. Those SCs that received a REQUEST behave similarly to SC_j. The *winning* SC is the one that sent the SYNCHRO with the highest random number. Finally, only the *winning* SC sends a REPLY onto CCC so that the other SCs can receive and route it towards their attached clients.

Figure 5. The Cooperative Media and Control GUI

4. The Client-side of the COMODIN system

4.1. The Cooperative Media and Control GUI

Figure 5 shows the Cooperative Media and Control GUI of the COMODIN client. It is organized in three main parts:

- *Media*, which allows for the presentation of playbacks (i.e., audio/video RTP-based sessions). It relies on the Java Media Framework API.
- *Control*, which provides the following functions: (i) connection to the COMODIN server-side, (ii) browsing the media archive, (iii) selection and control of a cooperative playback session.
- *Collaboration*, which allows to: (i) monitor the membership of the cooperative playback session, (ii) visualize the result of a control command acceptance, and (iii) send/receive questions using selective filters.

4.2. Application Domains

4.2.1. Collaborative Learning On-Demand (CLOD).
CLOD is an original synchronous/asynchronous remote learning paradigm which enable a group of students (e.g., classmates) to share the view and the control of a playback (e.g., seminar, talk, lecture) so as to go over it by collaborating with each other [10].

4.2.2. Distributed Virtual Theaters (DVT). A DVT is a distributed virtual environment where users (or viewers) meet in order to cooperatively browse, preview and playback movies in private virtual spaces.

5. Conclusions and on-going work

This paper has presented and described the models, protocols and architecture of the COMODIN system, a CDN-based media streaming platform which provides cooperative playback services.

The COMODIN system represents an interesting contribution in that it is able to overcome the issues affecting the MBone-based cooperative media on-demand systems, by efficiently providing cooperative playback services on the current Internet infrastructure by means of a CDN which exploits IP-multicast only where available.

Currently, the COMODIN system is being evaluated on a distributed testbed consisting of two high-performance PC networks connected through a *mroute*-enabled IP-tunnel and respectively located at the Università della Calabria and at the Universidad Politecnica de Valencia. Two kinds of evaluations are being carried out: (i) *usability evaluation* from the client perspective and (ii) *performance evaluation* from the system perspective.

6. References

[1] K.C. Almeroth and M.H. Ammar, "The Interactive Multimedia Jukebox (IMJ): A New Paradigm for the On-Demand Delivery of Audio/Video," *Proc. of WWW*, 1997.

[2] I. Busse, B. Deffner, and H. Schulzrinne, "Dynamic QoS Control of Multimedia Applications based on RTP," *Computer Communications*, **19** (1), Jan. 1996.

[3] C.D. Cranor, M. Green, C. Kalmanek, D. Shur, S. Sibal, C.J. Sreenan, and J.E. Van der Merwe, "Enhanced Streaming Services in a Content Distribution Network," *IEEE Internet Computing*, **5**(4), 2001, pp 66-75.

[4] J. Crowcroft, M. Handley, and I. Wakeman, "*Internetworking Multimedia*," Morgan Kaufmann Publishers, San Francisco, 1999.

[5] A. Dutta, and H. Schulzrinne, "A Streaming Architecture for Next Generation Internet", *Proc. of IEEE Int'l Conf. on Communications (ICC 2001)*, Helsinki (Finland), June 2001.

[6] Darwin Streaming Server, documentation and software, available at http://developer.apple.com/darwin/projects/streaming/, 2003.

[7] G. Fortino, A. Furfaro, C. Guerri, A. Pajares, C. Palau, W. Russo, "A Java-based adaptive media streaming on-demand platform," *Proc. of SCS Euromedia'02*, Modena, Italy, Apr. 2002.

[8] G. Fortino, C. Mastroianni, W. Russo, "Performance Analysis of an Application-level Cooperative Control Protocol", *Proc. of the 2nd IEEE Int'l Symp. on Network Computing and Applications* (NCA'03), Cambridge, USA, 16-18 April, 2003, pp. 305-312.

[9] G. Fortino and L. Nigro, "A Cooperative Playback System for on-demand Multimedia Sessions over Internet", *Proc. of IEEE Int'l Conf. on Multimedia and Expo* (ICME'00), New York, 2000.

[10] G. Fortino and L. Nigro, "Collaborative Learning on-demand on the Internet MBone," in C. Ghaoui (ed.), *Usability Evaluation of Online Learning Programs*, Idea Publishing Group (USA), pp. 48-60, 2003.

[11] G. Fortino, W. Russo, E. Zimeo, "Enhancing Cooperative Playback Systems with Efficient Encrypted Multimedia Streaming", *Proc. of IEEE Int'l Conf. on Multimedia and Expo* (ICME'03)", Baltimore (MD), USA, 6-9 Luglio, 2003.

[12] W. Holfelder, "Interactive remote recording and playback of multicast videoconferences", *Proc. of IDMS'97*, Darmstadt, Germany, Sept. 1997.

[13] B. Molina, C. E. Palau, J. C. Guerri, M. Esteve, "A closer look at a Content Delivery Network implementation", *Proc. of IEEE MELECON'04*, Dubrovnik (Croatia), May 2004.

[14] P. Parnes et alter, "multicast Multimedia On-Demand (mMOD)," available at http://mmod.cdt.luth.se, 1997.

[15] C.E. Palau, J.C. Guerri, M. Esteve, F. Carvajal, and B. Molina, "cCDN: campus Content Delivery Network Learning Facility", *Proc of IEEE Int'l Conf. on Advanced Learning Technologies* (ICALT'03), Athens (Greece), 9-11 July 2003.

[16] H. Shulzrinne, "A Comprehensive Multimedia Control Architecture for the Internet," *Proc. of NOSDAV'97*, St. Luis, Missouri, USA, 1997.

[17] A. Schuett, S. Raman, Y. Chawathe, S. McCanne, and R. Katz, "A Soft State Protocol for Accessing Multimedia Archives," *Proc. of NOSDAV'98*, Cambridge, UK, Jul. 1998.

[18] D.C. Verma, *Content Distribution Networks, an engineering approach*, John Wiley, New York, 2002.

[19] M. Zink, C. Griwodz, and R. Steinmetz, "KOM Player – A Platform for Experimental VoD Research," *Proc. of ISCC'01*, Hammamet (Tunisia), 2001.

Activity Sensing Floor Control in Multimedia Collaborative Applications

Kostas Katrinis, Georgios Parissidis and Bernhard Plattner

Computer Engineering and Networks Laboratory
Swiss Federal Institute of Technology, ETH Zurich
Zurich, Switzerland
E-mail: {katrinis|parissid|plattner}@tik.ee.ethz.ch

Abstract

The present paper introduces a novel approach to coordinate access to shared resources in a computer-mediated collaborative session. Our work borrows activity sensing and collision detection concepts from Ethernet-like protocols and delivers a fully distributed concurrency control (floor control) protocol. The control algorithm offers enhanced robustness and can be parameterized for network environments with varying quality of service characteristics. Additionally, it minimizes human intervention by coordinating a session in a fully automated way. To evaluate the protocol performance, we first model the behavior of collaborators in the context of floor control. Subsequently, we use the derived models to develop a simulation tool for our protocol, covering a comprehensive set of interactions seen in real-life collaborative sessions. Our results indicate high throughput and interactive response times for small group sizes. We finally propose minor alterations to the initial design to improve the responsiveness for larger groups.

1. Introduction

Computer Supported Collaborative Work (CSCW) is being increasingly deployed to enable the cooperation of geographically dispersed users in a cost efficient manner. Familiar application examples comprise of videoconferencing, synchronous distance learning, collaborative editing and distributed simulations. At the same time, network games nowadays account for a considerable merit of the digital entertainment market pie. All the above applications are characterized by a critical transition from the traditional face-to-face interaction to computer-mediated interaction over a communication network. One of the critical issues is the coordination of activities between collaborating entities. Intuitive verbal (voice volume, turn-taking orders) and non-verbal cues (gestures, eye contact, facial expressions) in face-to-face meetings are not valid in a computer-mediated collaborative environment. Here, the designation *floor control* [1] is used to embrace mechanisms that facilitate turn management in a networked collaborative environment. More precisely, floor control mitigates race conditions on the transmission, reception or manipulation of shared data among collaborating subjects. The term "floor" in this context can be thought of as a virtual token, whose possession is the necessary precondition for an entity to access a shared resource. Note, that the term "floor" does not confine the respective implementation to a specialization of token-passing protocols.

Above all, the motivation for floor control stems from the need to achieve awareness [2] in a collaborative session. This is achieved by disallowing anarchic interaction and by introducing a form of controlled concurrency. Beside this primary motivation, we add two supplementary functions covered by floor control. The first is admission control in network environments, where no explicit admission control functionality is available. For instance, floor control is often used in videoconferencing tools to regulate the number of concurrent video streams received by a client with limited downstream capacity. Second, legacy applications – i.e. applications that were not engineered for collaboration – can be integrated without changes into a collaborative environment using floor control middleware. Such a middleware serializes data delivery to the single-input legacy application and conversely distributes the single output of the application to all collaborating partners. An example, where such a middleware becomes useful, is sharing legacy applications (e.g. worksheets, text processing tools) in a groupware environment.

This paper presents *Activity Sensing Floor Control* (ASFC), a distributed floor control protocol based on sensing activity on a shared resource, to which concurrency control needs to be applied to. Our design is valid for coordinating access to fully-replicated shared resources, i.e. resources that maintain identical process instances on every collaborating end system and use group-scoped broadcast for disseminating data to the collaborative environment.

Typical resources that fall into this category are live audio/video streams in a videoconference, shared whiteboards or multiparty network games, where the virtual channel that disseminates the data to a group of participants can be regarded as the shared resource.

The concept of floor control can be further dissected into floor control mechanisms and floor control policies. Mechanisms cover issues like protocol formalism and network topology. Policies refer to the resulting coordination strategy. For example, consider a videoconference with a centralized floor control server. Users place requests to the server and are granted access to the resource according to a queueing algorithm implemented by the server. The centralized topology, the floor control messages and the request-reply model constitute the floor control mechanism. Over this specific mechanism, one can employ several policies by tuning the queueing strategy (e.g. FIFO, priority-based). In the premises of this work, we only cater for providing a distributed floor control mechanism for multimedia collaborative applications. As such, we employ the common policy of explicit floor request and explicit floor relinquishment, allowing at most one active user on the shared resource at any time. Although other policies are possible, evaluating the impact of different policies to the collaboration outcome is out of the scope of this work. Furthermore, floor control protocols are classified by Dommel [1] as either *assistive* or *autonomous*. The first require human intervention (e.g. chairperson-controlled) to facilitate floor management, whereas the second rely on automated processes. ASFC belongs to the autonomous category and does not require human intervention in the floor decision process. Semantically, this has the side effect that ASFC is suitable for sessions, where the ordering of user requests is not relevant. Brainstorming meetings, unstructured discussion groups, educational puzzle games and "quick-answer" games are examples of such sessions.

The rest of this paper is organized as follows. Chapter 2 contains a survey of related work. In chapter 3, we start with modelling a collaborative session in the context of floor control and then describe the ASFC protocol. Chapter 4 elaborates on the evaluation methods and discusses the experimental results. We end up finally in chapter 5 with conclusions and future work.

2. Related Work

Most of prior research on floor control mechanisms comes from the area of conferencing systems. The Conference Control Channel Protocol (CCCP) [3] provides building blocks for managing conferences of various sizes. Its design accommodates a moderator-based floor control scheme for interaction control. Malpani [4] employed similarly a centralized assistive floor control model to provide guidance to chairperson-controlled seminars over the MBone. Research within the CSCW community has rather focused on the efficiency of floor control policies with regard to collaborative awareness and group-work outcome. In [5], the authors evaluated for four different policies the consensus achieved in an online meeting, while Boyd [6] explored theoretically the design space for floor policies in multi-user applications.

Proceeding to past research strongly related to our work, the EMCE system (Experimental Multimedia Conferencing Environment) [7] is – according to the authors' bibliographical research – the first to refer to activity sensing mechanisms to facilitate concurrency control in a collaborative session. The respective prototype uses a primitive voice-activated floor control scheme to force a single user speaking at every instance of time. If two participants started speaking simultaneously, the respective floor control modules suppressed audio transmission after detection of the collision. Collision resolution in the EMCE system relies on the behavior of conferees, i.e. by having all but one of the colliding users withdraw their intention to speak via intuition or explicit coordination (messaging).

The work presented herein is inspired by the EMCE floor control scheme. However, we went on to develop on the idea of activity sensing and give an elaborate description and evaluation of a distributed algorithm for the same purpose. In this sense, we tackle many design and performance issues that were not addressed in the work discussed in [7]. Additionally, we eliminate the reliance of collision resolution on human behavior by assigning this task to automated processes.

The CECED (Collaborative Environment for Concurrent Engineering Design) [8] environment employed the floor control scheme of the EMCE system without developing further on the scheme itself. Recently, Dommel [9] worked on a comparative analysis of several classes of floor control protocols, including the activity sensing case. The method presented in [9] borrows from the analytical evaluation of CSMA protocols and is used to estimate the theoretical throughput of activity sensing protocols. However, the paper does neither deliver a sound protocol design for activity sensing, nor does it contain a practical evaluation, as the present work.

3. Protocol Details

3.1. Protocol Description

In order to formalize the concept of a collaborative session (CS), we first present a characterization of a CS. A collaborative session can be defined as a 4-tuple:

$$CS = (G, U, R, F) \qquad (1)$$

where G stands for the connectivity graph $G = (V, E)$, V for the set of end-system nodes and E for the set of end-to-end links connecting them. Moreover, $U = \{u_1, u_2, u_3, ..., u_N\}$ is the set of collaborating users u_i ($i = 1..N$), where N stands for the group size. The system assumes a one-to-one correspondence between nodes and users, i.e. only a single user resides at a single end system for the entire session lifetime. Furthermore we denote with R the set of resources available for shared access in the premises of the collaborative session and with F the set of floors providing mutually exclusive resource access. We assume a one-to-one correspondence between R and F, meaning that access to each resource $r \in R$ is regulated by one and only one $f \in F$ and conversely. For the sake of simplicity, the last statement implies that each floor administers exactly one (floor-controlled) resource. However, we believe that our protocol can easily be extended to accommodate cases, where a single floor entity regulates access to a group of resources.

The ASFC protocol aims at providing mediation of concurrent access to fully replicated resources in a collaborative session. Mediation is achieved by each user monitoring the activity on a specific resource and contend for resource usage only when the medium is sensed unoccupied. For every resource $r \in R$ that a user u_i is subscribed to, a floor control agent e_i^r manages access to the shared resource by running an instance of the ASFC protocol. We use henceforth the term "agent" to refer to an instance of the ASFC. The agent is started immediately after the creation of the local instance of the resource r and serves the local user during the entire lifetime of the session.

Activity sensing is achieved through real-time monitoring of the incoming data traffic towards the resource (i.e., on the well-known resource port). The monitoring mechanism of the ASFC protocol is conceptually derived from the carrier sense strategy of the Ethernet IEEE/802.3 protocol (CSMA/CD) [10]. The CSMA/CD algorithm enables multiple nodes to share a single medium, provided that each node is equipped with the ability to tap the traffic on the medium. In ASFC, a resource is asserted occupied, if the inter-arrival time between two consecutively received data packets does not exceed a specific time threshold T_{max}; otherwise the resource is considered free. Neglecting for now the case of packet loss, the former statement is valid, if the following two preconditions hold:

a The worst case one-way transit delay (OTD_{max}) experienced by a data packet is fixed and globally known.

b The maximum inter-departure time between two consecutive data packets transmitted by the floor holder is fixed and globally known.

Precondition a) is true for specific network environments, like Local Area Networks (LANs) and networks with QoS guarantees. For the sake of the protocol presentation here, we assume the general truth of precondition a) and discuss at the end of this chapter how to meet this precondition in networks, where maximum network delay and packet loss rate are hard to estimate. Next, we cover precondition b) as follows. We first fix the maximum inter-departure time threshold to the maximum one-way transit delay OTD_{max} of a data packet (assuming uniform data packet sizes). On every new data packet transmission, the floor holder starts a timer initialized to OTD_{max}. If the timer expires, the floor holder broadcasts a "dummy" data packet of minimal size. We claim that exploiting "dummy" packets will not incur a considerable overhead to the network because of their minimal size and small transmission rate (not bursty). And second, because we expect that "dummy" packets will be infrequently used: these packets are essentially used to prevent a floor holder from "losing" the floor unintentionally during activity breaks (thinking time) within his activity slot. This is not necessary for high-rate continuous sources (e.g. audio/video streams) and infrequently needed for other resource types (whiteboards, games). The strategy ensures a maximum inter-departure time of OTD_{max} at the sender and essentially a maximum data packet inter-arrival time T_{max} at the receiver given by:

$$T_{max} = 2 \times OTD_{max} + (2 \times k - 1) \times \gamma_{max} \quad (2)$$

where γ_{max} stands for the predefined maximum transmission time of a single data packet and k is given by $k = N - 1$, where N is the number of the collaborating users as defined above. Ultimately, an ASFC-agent monitors a resource by restarting a timer initialized to the maximum tolerable inter-arrival time T_{max} on the reception of every new data packet. It can be shown that T_{max} also represents the maximum time interval a node's $u_i \in V$ attempt to access a resource can be intercepted by another node $u_j \in V (i \neq j)$. In other words, T_{max} constitutes the **vulnerability time** for a floor acquisition attempt. Clearly the time interval T_{max} plays a significant role in the proper operation of the protocol. This maximum time threshold should be known to every ASFC-agent during initialization (either hard-coded or preferably communicated via a session control protocol).

Note that, if the collaborative session uses a group communication transport (e.g. IP multicast), the transmission overhead $(2 \times k - 1) \times \gamma_{max}$ term in (2) reduces to γ_{max}, due to the fact that each packet is emitted only once at the sender (i.e. $k = 1$). Even in the absence of a group communication scheme, the transmission overhead is negligible compared to OTD_{max} for the group sizes ($5 \leq N \leq 50$) and the client bandwidth capacity (128Kbps) under study. Thus, in what follows, we approximate T_{max} as given in (2) with $T_{max} \approx 2 \times OTD_{max}$.

We model the ASFC-agent as a finite state machine with a state space S = {IDLE, PASSIVE, CONTENDING, AC-

TIVE)} (Fig. 1). Transitions are triggered either by user- or environment-initiated events (in the following text we depict states with capitals and transitions with italic letters).

The floor control agent of a reference user u_i starts as IDLE. In this initial state neither floor requests of the local user are awaiting service, nor is the local user the current floor holder. Consequently, an IDLE agent does not need to sense activity on the resource.

Figure 1. State Diagram of the ASFC-agent. States are illustrated with circles and transitions with directed line segments

In case of user interaction of user u_i (*floor request event*), his agent changes its state to PASSIVE. Subsequently, the agent starts monitoring the resource as follows. It fires a timer initialized to T_{max} and listens for data packets sent to the replicated resource. If a data packet is received before timer expiration, the timer is restarted (*resource occupied*).

However, if the timer expires without receiving a new data packet (*resource available*), the agent assumes a free medium and proceeds to the CONTENDING state. The agent of u_i then broadcasts a probe packet of minimal size claiming access on the resource. At the same time it starts a new timer (also initialized to T_{max}) and monitors the traffic on the resource. If no probe packet is received during a timer period T_{max}, u_i becomes the active floor holder, transits to the ACTIVE state and starts transmitting data packets to the resource (*contention won*). Otherwise, a collision occurs: the colliding nodes remain in the CONTENDING state and initiate a **Binary Exponential Backoff (BEB)** algorithm as introduced in [10] to resolve the conflict (*collision*). We employed the BEB algorithm for collision resolution as it provides fair access to a shared resource and allows for distributed implementation. In the conflict resolution mode each contending node fires up a backoff timer initialized to a value that is uniformly random in the backoff interval b_T:

$$0 \leq b_T \leq (2^n - 1) \times T_{max} \qquad (3)$$

where n denotes the number of unsuccessful resource acquisition attempts in the specific contention round and T_{max}

denotes the vulnerability time of the protocol. A contention round ends, when exactly one of the colliding agents sends a probe packet and does not sense any traffic on the resource for a time period T_{max}. The contention winner moves from the CONTENDING to the ACTIVE state and starts broadcasting data packets to the shared resource (*contention won*). The rest of the colliding agents abandon the backoff mode as soon as they start receiving data packets from the new floor holder and change back to the PASSIVE state (*contention lost*), waiting for the resource to become free to restart contention. Finally, when user u_i signals the completion of his activity on the resource, his ASFC-agent moves to the IDLE state (*release resource*).

Note that our protocol differs from traditional CSMA protocols in the resource probing approach. In the CSMA paradigm, end systems start sending data as soon as they sense the channel free, i.e. no special probing packets are used. When transferring the same method to the ASFC paradigm and specifically to network environments with longer round trip delays one has to be careful. In environments with long round trip delays, two colliding nodes require more time until they sense the collision and defer from data transmission. Consequently, letting nodes probe by initiating data transmission – as performed in CSMA protocols – would overload the network with extra (and probably not useful) packets. We manage to reduce network overhead by controlling the volume of probing information. For this we use the "probe" packet mechanism described above.

3.2. Discussion

According to the above description, one can argue that the ASFC protocol is primarily suitable to be deployed in high speed Local Area Networks (LAN), where maximum latency can be estimated with high probability and packet loss rate is negligible. However, we argue that our protocol is deployable in network environments with probabilistic quality of service characteristics (e.g. Internet). We achieve this via a) an appropriate parameterization of the one-way transit delay OTD_{max} and b) the introduction of misbehavior resolution mechanisms in case of detection of improper protocol operation. In the next section, we examine the performance of our protocol with regard to the OTD_{max} value. We set OTD_{max} to rather high (and thus "safe") maximum transit delays ($OTD_{max} = \{250|500\}$ ms) to cover various network conditions, such as those found in the Internet.

Our protocol handles packet loss events in a reactive manner. Instead of putting effort in loss prevention, we take an optimistic approach and shield the protocol with functionality to resolve non-explicit resource accesses. Consider that, if a probe packet or a train of "dummy" packets is lost, agents in the PASSIVE mode will erroneously assert a free resource and will start contending for the floor.

In some cases this could lead to the situation, where two distinct users claim to hold the floor, violating thus the protocol correctness requirement. To ensure atomicity of a user in the ACTIVE state, we force an active ASFC-agent to immediately defer and transition to the PASSIVE state (*exclusivity violation*), as soon at it receives data packets from another user of the collaborative session. The quantification of the impact of packet loss on the protocol's performance using Internet-like packet loss models constitutes a significant fraction of our ongoing work.

4. Protocol Evaluation

4.1. Simulation Setup

We used the OMNET++ discrete-event simulation framework to realize a packet level simulator for our floor control scheme. Our simulator fully implements the application layer (AL) and session layer (SL) of a collaborating end system. We simplified our simulation environment by abstracting a single transport layer (TL), which carries out the delivery of a packet to the rest of the collaborating peers. The experiment topology comprises of a fully connected graph; vertices of the graph model participating end systems and edges model end-to-end paths connecting any two systems. Whenever a packet is generated by the AL or the SL, it is delivered to the TL for transmission. Subsequently, the TL calculates the effective one-way transit delay (OTD_{eff}) of the packet to each of the collaborating peers. OTD_{eff} is a random variable within the value range [OTD_{min},OTD_{max}] adhering to the Poisson distribution. Finally, the packet is pushed into the network, adding a transmission time overhead to the overall packet delivery time. The packet transmission time is specified using the assigned node bandwidth (B) and the packet size (P_{data} or P_{probe} as maximum data and probe frame size respectively). Note that our model assumes links with zero packet loss and therefore, our simulation results do not capture the influence of loss incidents on the protocol performance. At simulation warmup, the nodes exchange handshaking messages and agree on a common maximum one way delay. The latter is used by the algorithm to specify the protocol's vulnerability time. After the end of the bootstrap phase, the simulator generates the floor request patterns of a collaborative session: it generates floor requests randomly and assigns them to idle nodes, together with the (random) time interval for which the node will hold the shared resource when it becomes active. In the following, we describe the patterns used to construct the floor control request and activity time samples, which form a critical input for the quality of the simulation results.

4.2. Input Workload Generation

It is critical to choose cautiously the value ranges of the independent variables that model the behavior of collaborative users; otherwise the simulation results will not resemble the protocol behavior in real-life scenarios. First, we model the effective activity time T_{eff}^a, i.e. the time that a participant uses the resource, when becoming the floor holder. Due to the lack of statistical data of user-behavior in similar application scenarios, we model T_{eff}^a as a uniformly distributed random variable with boundary values T_{min}^a and T_{max}^a respectively. For the selection of the activity time boundaries, we have defined two scenarios with different degrees of interactivity.

In *scenario-1*, we capture collaborative sessions, where speakers rather spend time on a predefined topic or participate in collaborative tasks with relatively long lasting turns (e.g. document editing, graphical designs) than often interrupting the speaker. Complementary to this, *scenario-2* models sessions with spontaneous short turns, like for instance distributed quiz shows or question-and-answer sessions. We believe that the above two settings cover a major part of activity time patterns seen in real-life collaborative sessions and therefore, allow us to generalize our simulation results. Additionally, we model the behavior of collaborating users with regard to floor request generation. For this, we use a centralized approach defining a random variable K, which represents the number of new floor requests generated during an activity time. Taking into consideration that a not yet serviced floor request is buffered at an end system, the random variable K takes values in the interval $[0, N - H]$, where N is the group size and H the number of users, who possess buffered floor requests that will be serviced in future contention rounds. Again, due to the lack of measured user behavior, we have chosen a distribution of K, imitating interaction patterns in real-life collaborative sessions. We observe that for constant group size, it is highly probable that the contribution of the actual speaker (or more generally resource user) will provoke a small to moderate (relatively to the group size) number of new floor request generations. However, there is always a rather low probability of an activity causing a burst of simultaneous floor requests. Such bursts occur for example, when a hot topic is touched by the active speaker during a meeting or when a key step is taken by the active floor holder during a collaborative problem solving session. Based on these assumptions, we model the number of new floor requests K with a Poisson distribution, i.e. the probability $P(k)$ that k new requests are generated during a single activity round is given by:

$$P(k) = \frac{e^{-\lambda} \cdot \lambda^k}{k!} \quad (4)$$

where λ determines the mean value ($E[K] = \lambda$) and the

standard deviation ($\sigma[K] = \lambda$) of K. Throughout the simulation experiments, we let the parameter λ grow linearly to the group size according to the function:

$$\lambda(N) = 1 + \beta \cdot (N - 5), (N \geq 5) \qquad (5)$$

where N stands for the group size and parameter β controls the impact of the group size on the density of new floor requests. We conducted our experiments using two distinct values for parameter β, namely $\beta_1 = 0.04$ and $\beta_2 = 0.1$. The value β_1 captures a scenario of low to moderate number of floor requests, whereas with the value β_2 we test our algorithm against more aggressive interaction modes. Note that in (5) λ only depends on the group size. One might argue that K should also depend on the size of each activity round, since a longer speech would probably cause a greater amount of floor control requests by the listeners. This would obviously preclude the modelling of the random variable K with a stationary distribution. For simplicity reasons and due to the small variance in the chosen activity time values, we model λ as independent of the length of the activity time.

4.3. Simulation Results

The results presented in this section were produced as follows. We start the simulator and the floor request generation module assigns requests to collaborating nodes for a predefined simulation time point $t_{last} = 3600$ sec. If after the simulation period nodes with waiting floor requests exist, we continue the simulation until all outstanding requests have been satisfied (t_{fin}). We first measure the throughput of the system to evaluate the utilization of the shared resource under the previously discussed input workload. The throughput T is given by the overall activity time over the entire session duration t_{fin}, as shown in (6). Note that we implemented our simulations such that at least one outstanding floor control request exists after the completion of every activity period.

$$T = \frac{Overall\,Activity\,Time}{t_{fin}} \qquad (6)$$

Intuitively, the throughput depends on the magnitude of activity times, the protocol overhead (e.g. vulnerability time) and the interaction modus (number of colliding requests). To test this hypothesis we created eight datasets with varying input values for each parameter, as shown in Table 1. For each dataset, we alter the value of exactly one parameter, while keeping the rest of the parameters constant. For instance, we used Dataset1 and Dataset5 to test the influence of the interaction modus to a session with relatively large activity times (30s to 210s) and low overhead ($OTD_{max} = 250$ms).

Dataset	OTD_{max}	T^a_{min}	T^a_{max}	β
1	$250ms$	$30s$	$210s$	0.44
2	$250ms$	$10s$	$30s$	0.44
3	$500ms$	$30s$	$210s$	0.44
4	$500ms$	$10s$	$30s$	0.44
5	$250ms$	$30s$	$210s$	0.11
6	$250ms$	$10s$	$30s$	0.11
7	$500ms$	$30s$	$210s$	0.11
8	$500ms$	$10s$	$30s$	0.11

Table 1. Sets of simulation input parameters

Figures 2 and **??** present the achieved throughput of the protocol for the eight datasets. We observe that the throughput decreases slowly with an increasing group size in the scenarios with large activity time slots (30s to 210s), remaining above 88%. This is comprehensible, if one considers the difference in the order of magnitude between useful activity time and resource acquisition time samples. As shown later in our results, we found the average resource acquisition time to be between 1 and 9 seconds. Compared to the time spent on the resource (uniform distribution in [30s,210s]), the average overhead for a resource access is considerably low, even in the case of aggressive interactivity patterns (Datasets 5 and 7). Moreover, we observe that across the scenarios with large activity time, the utilization of the shared resource was nearly independent of the vulnerability time (OTD_{max}). This motivated us to experiment with higher values of vulnerability time to test the protocol deployment in network environments with no fixed worst case packet delivery time guarantees (e.g. Internet), without sacrificing its high throughput. On the contrary, the scenarios with short activity slots yielded high throughput only for small group sizes. We observe that utilization decreases steadily as group members grow, reaching its lowest value

Figure 2. Protocol throughput for mild interactivity patterns ($\beta = 0.44$)

Figure 3. Protocol throughput for aggressive interactivity patterns ($\beta = 0.11$**)**

Figure 4. Average waiting time versus group size for the low interactivity scenarios

Figure 5. Average waiting time versus group size for the high interactivity scenarios

across all measurements (50% in dataset 4). This is due to the fact that resource acquisition time is comparable to the resource usage time (uniformly distributed in [10s,30s]). Also note that our interactivity model assumes that the number of new floor requests generated during a single activity slot is independent of the size of the time slot. Hence, small slots equal to a larger floor request generation rate and consequently to more and lengthier collisions with a negative impact on the utilization of the resource.

Apart from efficiency in terms of resource utilization, responsiveness is of equal importance for a floor control protocol used in a collaborative application. Therefore, we need to study the time between consecutive resource usages (contention time) as well. During this time interval, the resource is seen by each user as idle and thus we need to limit these idle slots to a few seconds, otherwise the protocol would degrade the effectiveness of the online collaboration. For an ASFC-agent that will try to acquire the floor in the next contention round, we define as *waiting time* the time interval:

$$Waiting\ Time = t_{occ} - t_{free} \qquad (7)$$

where t_{occ} is the timestamp that the resource is sensed to have been occupied - either by the local agent (*contention won*) or by another agent (*contention lost*) - and t_{free} stands for the absolute time that the resource was first sensed free during this contention round. For each simulation run, we averaged the waiting time samples assessed by each agent and depicted the results versus group size as shown in Fig. **??** and **??**. The results manifest that the current protocol design is suitable for rather small groups of collaborating entities (roughly up to 10 members). Beyond this group threshold, the waiting time grows unacceptably high (up to 9.6 sec) for an interactive application.

4.4. Discussion

It is straightforward that a contention round is proportional to the vulnerability time (OTD_{max}). Throughout our evaluation we studied only cases with fairly large vulnerability time values to cover various types of network environments. For instance, our results do not capture the case of a collaborative session comprising of participants, who are geographically close to each other. In this case the OTD_{max} protocol parameter will decrease and so will the waiting time measurements. In this first work on ASFC we confined ourselves to protocol design, simulator development and performance evaluation of principal scenarios. Evaluating the protocol performance in different network settings and for further interaction scenarios is for us ongoing research.

Finally, we demonstrate here an algorithm that gives constant waiting times, irrespective of the one-time delays (bounded by OTD_{max}) of the underlying network. Intuitively, long contention times occur when a large number of

participants contend for the floor. In particular, the BEB algorithm yields collision rounds that are logarithmic to the number of colliding participants. A simple solution that gives constant waiting time is to let every contending agent select after the first collision uniformly a random integer out of the interval $[1, L]$, where L is the number of colliding agents in this specific round. The first "dummy" collision is needed to let all agents specify the number L. For the rest of this round only agents that picked numbers in the interval $[1, M]$ will continue contending for the floor, whereas the remaining $L - M$ agents will return to the PASSIVE state. The duration of average waiting time versus group size measured at scenarios with more intensive interactivity contention rounds can then be controlled by setting the constant M to a desired value (obviously as a function of the vulnerability time). Keep in mind that this strategy does not only provide fixed waiting times, but will as well improve throughput due to lower protocol overhead.

5. Conclusions and Future Work

This paper proposes a distributed floor control protocol to coordinate access to fully replicated shared resources in collaborative environments. The scheme is robust due to its fully distributed implementation; failure of any single node does not lead to failure of the protocol. For example, if the end system hosting the actual floor holder fails, data packets will stop being transmitted. The users in PASSIVE state will then assert the resource free and will start a new contention round that will yield a new floor holder. Furthermore, our protocol minimizes human intervention in floor management. Usually, human interference increases the operational fault probability and increases response time. ASFC is rather based on automatic processes that act on behalf of the user's intention. This has the additional advantage of offering transparency to the end user; users are not bothered by explicit floor control commands or notifications.

Apart from the significance of the protocol features, we presented a model for a collaborative session in the context of activity sensing floor control and explored the design space for constructing a respective simulation tool that resembles real-time scenarios. Our evaluation showed that the protocol achieves high throughput, but fails to scale to large group sizes with regard to response time. To overcome this limitation, we proposed a strategy to limit the protocol's waiting time.

The integration of the latter strategy into the protocol design and its subsequent evaluation is one of our planned tasks in the premises of this work. More importantly, we plan to enhance our simulation tool with a packet loss model derived from Internet measurements and then evaluate the influence of packet loss to the protocol performance. Fi-

nally, we consider applying fairness criteria to the selection of backoff intervals by the BEB algorithm. Currently, we employ a flat uniform selection of backoff intervals. It might be relevant for specific collaborative scenarios, such as those where users with small participation time require priority over users monopolizing resource usage, to perform research on different backoff strategies.

6. References

[1] Dommel H.P. and J.J. Garcia-Luna-Aceves, "Floor control for multimedia conferencing and collaboration", ACM Mutlimedia Systems, vol. 5, iss. 1, p. 23-28, 1997.

[2] Beaudouin-Lafon M. and A. Karsenty, "Transparency and awareness in a real-time groupware system", in Proceedings of the 5th Annual ACM Symposium on User Interface Software and Technology, p. 171-180, 1992.

[3] Handley M., I. Wakeman, and J. Crowcroft, "The conference control channel protocol (CCCP)", in Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, p. 275-287, 1995.

[4] Malpani R. and L.A. Rowe, "Floor control for large-scale MBone seminars", in Proceedings of the 5th ACM International Conference on Multimedia, p. 155-163, 1997.

[5] McKinlay A., R. Procter, et al., "A study of turn-taking in a computer-supported group task", in People and Computers VII: Proceedings of HCI '93, 1993.

[6] Boyd J., "Floor control policies in multi-user applications", in INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems, p. 107-108, 1993.

[7] Aguilar L., J. J. Garcia-Luna-Aceves, et al., "An architecture for a multimedia teleconferencing system", in Proceedings of the ACM SIGCOMM, p. 126-136, 1986.

[8] Craighill E., R. Lang, et al., "CECED: a system for informal multimedia collaboration", in Proceedings of the 1st ACM International Conference on Multimedia, p. 437-445, 1993.

[9] Dommel H.P. and J.J. Garcia-Luna-Aceves, "Efficacy of floor control protocols in distributed multimedia collaboration", Cluster Computing, vol. 2, iss. 1, p. 17-33, 1999.

[10] "IEEE standards for local area networks: carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications", 1985.

MORA: a movement-based routing algorithm for ad hoc networks [*]

Giulia Boato and Fabrizio Granelli

DIT - University of Trento
Via Sommarive 14, I-38050 Trento (Italy)
E-mail: {`boato,granelli`}@dit.unitn.it

Abstract

In an ad hoc environment with no wired communication infrastructure, mobile hosts necessarily operate as routers, in order to provide network connectivity. Since mobile ad hoc networks change their topology frequently and without prior notice, routing in such networks becomes a challenging task.

In this paper we present MORA, a movement-based routing algorithm for mobile ad hoc networks. The algorithm is completely distributed, since nodes need to communicate only with direct neighbors in their transmission range, and utilizes a specific metric, which exploits not only the position, but also the direction of movement of mobile hosts.

1 Introduction

Mobile ad hoc networks consist of wireless hosts that communicate with each other in the absence of a Þxed infrastructure . In an ad hoc wireless network, mobility and bandwidth are two key elements representing research challenges. Not all hosts are within the transmission range of each other and communication is achieved by multi-hop routing. Due to the hosts mobility, the topology of the network can change with time and nodes have to build and update their routing tables automatically and effectively.

Traditionally, multi-hop routing for mobile ad hoc networks can be classiÞed into proactive and reactive algorithms. In proactive routing algorithms, each node in the mobile ad hoc network maintains a routing table that contains the paths to all possible destinations. If the nodes in the network are reasonably mobile, the overhead of control messages to update the routing tables becomes prohibitive. Moreover, storing large routing tables in low-cost mobile nodes might be too expensive. Reactive routing algorithms, on the other hand, Þnd routes only on demand. When a node needs to send a message to another node, the sender needs to ßood the network in order to Þnd the receiver and determine a path to reach it. This process can still use a signiÞcant amount of the scarce available transmission resources. A detailed review of routing algorithms in mobile ad hoc networks can be found in [1, 2].

An interesting approach is represented by position-based routing algorithms [3, 4], which require information about the physical position of the participating nodes. The distance between neighboring nodes can be estimated on the basis of incoming signal strenght or time delay in direct communications, or may be available using GPS.

In this paper, the problem of routing in an ad hoc network is considered. An alternative MOvement-based Routing Algorithm (MORA) is presented, which exploits not only the position, but also the direction of motion of mobile hosts. The paper is organized as follows. Section 2 introduces the new method, which is then analyzed in Section 3. In Section 4 a comparison with existing algorithms is presented. Finally, Section 5 concludes the paper.

2 The proposed method

The desirable properties of any routing protocol include simplicity, loop-free operation, convergence after topological changes, small storage, reduced computational and transmission overhead. In a position-based routing algorithm, each node makes a decision to which neighbor to forward the message based only on the location of itself, its neghboring nodes, and destination. In our approach, this decision is taken considering also which direction neighbors are moving in, in order to exploit this knowledge to optimize data path. None of the existing strategies to forward packets (MFR, NFP,...) takes into consideration that hosts in ad hoc network are moving in directions that can introduce unpredictable changes in the network topology , which can hamper the stability of the links and routes. Moreover, the system is made more robust by avoiding centralized information management, and easier to set up and operate.

The metric used in MORA is a linear combination of

---

the number of hops, arbitrarily weighted, and a target functional, which can be calculated independently by each node.

2.1 The functional F

The idea is to create a functional that each node can independently calculate, which depends on how far the node is from the line connecting source and destination, sd, and on the direction the node is moving in. The target functional should reach its absolute maxima in the case the node is moving on sd and it should decrease as the distance from sd increases. Moreover, the more a node moves towards sd, the higher should be its value, i.e. for a Þxed distance from sd the functional should have a maximum if the node is moving perpendicularly to sd.

Let d_0 be a reference distance metric, chosen on the basis of the application context (e.g. 1 meter, or 10 cm). Let $x = \frac{d}{d_0}$ be the adimensional distance of the current node from sd and $y = \frac{l}{d_0}$ the adimensional distance from destination of the point of intersection between sd and its perpendicular starting from the node current position . The functional F is a function of $x \in [0, \infty]$ and $\alpha \in [-\pi, \pi]$, where α represents the angle between the line of direction and the perpendicular line to sd .

Figure 1. Definition of d, l and α

For the purpose of the paper, the functional F is deÞned as follows

$$F_{\delta,\gamma}(x, \alpha) = sin\frac{|\alpha|}{3} e^{-|x|} + cos\frac{\alpha}{3} e^{-\frac{(x-\delta)^2}{\gamma}}$$

where δ and γ are two parameters set on the basis of the application. With such deÞnition of F, more weight is given to nodes moving on sd, and also to nodes moving towards it (see Figure 2) as required above. In fact: *i)* for $x = 0$ there are 2 absolute maximums, for $\alpha = \pm\frac{\pi}{2}$ respectively; *ii)* for $0 < x < \epsilon$ (ϵ arbitrarily small) the trend is the same as above; *iii)* for $x \to \infty$ the function decreases; *iv)* for $x = \delta$ there is a relative maximum corresponding to $\alpha = 0$; *v)* for $x \in [\delta - a_{\delta,\gamma}, \delta + b_{\delta,\gamma}]$ ($a_{\delta,\gamma}$ and $b_{\delta,\gamma}$ constants deÞned with the choice of δ and γ) there is a maximum corresponding to $\alpha = 0$.

The idea is to favor relatively stable paths and not necessarily those with smaller number of hops. Moreover, by

Figure 2. $F_{\delta,\gamma}(x, \alpha) = sin\frac{|\alpha|}{3}e^{-|x|} + cos\frac{\alpha}{3}e^{-\frac{(x-\delta)^2}{\gamma}}$ **con** $\delta = 2, \gamma = 15$

carefully setting δ and γ, it is possible to modify the inßuence of node$\tilde{\text{Q}}$ direction of movement and therefore the curvature of functional F.

The functional F will be sampled and put into a look up table. In this way, each node does not need to calculate F for any computation, but it can easily obtain the value corresponding to a given combination of x and α with a table lookup.

2.2 The metric m

Another degree of freedom of the metric employed in MORA is the weight assigned to each node, which can be used to represent trafÞc conditions, application constraints, etc. The goal of the weighting function is to obtain a fair distribution of the available resources through the overall network. For the purpose of the paper, the function W, deÞned for $y \in [0, y_{src}]$, is given by

$$W(x, y) = \begin{cases} 1 & 0 \le w(x, y) < 0.1 \\ -log_{10} w(x, y) & 0.1 \le w(x, y) \le 10 \end{cases}$$

where $w(x, y) \in [0, 10]$ is the weight of node i with coordinates x, y.

The following metric can be deÞned, for $y \in [0, y_{src}]$

$$m_{\delta,\gamma}(x, y, \alpha) = \frac{1}{2}(W(x, y) + F_{\delta,\gamma}(x, \alpha))$$

where both $W(x, y)$ and $F_{\delta,\gamma}(x, \alpha) \in [-1, 1]$ and therefore $m_{\delta,\gamma}(x, y, \alpha) \in [-1, 1]$. Due to the fact that x and y are the coordinates of node i and α depends on the node i, in following sections we refer to $m_{\delta,\gamma}(x, y, \alpha)$ and m_i without distinction. The reader should note that, by choosing such metric, the higher the value of m_i the higher the probability node i is selected for the path from source to

destination. Moreover, if node i is congested and therefore $w(x, y) \to 10$, then $W(x, y) \to -1$.

3 The MORA routing protocols

U-MORA

The first version of the routing protocol is called Unabridged-MORA, because the core idea is similar to source routing on IP networks and it does not support scalability. In position-based routing algorithms, usually a short probe message is used for destination search, that collects routing information from destination to sender, and finally data are send from source to destination. Our approach exploits this small packet, not only to localize the destination, but also to get information about the best path between source and destination at that moment and for the near future (see Figure 4).

In this algorithm a similar short probe message, used to localize the destination, once it is received, it is sent back from the destination node through several routes, imposing strictly increasing values of y to avoid loops and values of F bigger than a certain threshold η to avoid flooding. Each node j, except the source node, receiving the packet through the link (i, j), updates the value of the function

$$M_j = M_i + m_j$$

by calculating the functional (see Section 2.1) and increasing the number of hops (with relative weight). The packet is updated with the node identifier and the updated value M_j In such a way each M_i received by the source identifies a single path to destination. The state diagram for node j is presented in Figure 3.

probe message is received from node i with value M $_i$

node j

idle   active

if F >η and y$_j$ > y$_i$ m$_j$ is calculated and the packet is forwarded with the updated M $_j$

Figure 3. U-MORA flow chart

Supposing that different probe packets, following different routes, pass through the same node j, in order to avoid computational and traffic overheads, node j has to make a decision depending on functions M_i of packets arriving simultaneously (in a predefined time-window). After the decision, node j updates the function to get M_j and forwards

only one probe message. All other packets getting to node j after the decision will be discarded.

When a fixed timeout T expires, the source has a set of different reliable paths to the selected destination and the corresponding functions M_i. The source node has to take a decision to which path to use. Such decision obviously depends on the weight of each node, its position and direction of movement.

destination

2 → 4

$m_1 + m_2$ $m_1 + m_2 + m_4$ $m_1 + m_2 + m_4$

1

m_1 5 $m_1 + m_3$

3   source

$m_1 + m_3$

Figure 4. Metric computation example

Figure 4 presents a simple example . The destination node 1 computes $M_1 = m_1$, while the intermediate nodes update the received function, i.e. $M_2 = M_1 + m_2$, $M_3 = M_1 + m_3$, $M_4 = M_2 + m_4$. The source node 5 get two packets, in which an available path and M_3, M_4 are respectively recorded: if $m_1 + m_3 > m_1 + m_2 + m_4$ data are then sent through the path $5 \to 3 \to 1$, rather than through $5 \to 4 \to 2 \to 1$.

U-MORA requires a relevant amount of traffic, due to the several possible paths and therefore to the several packets trasmitted. The following routing algorithm, D-MORA, presents another use of the metric m, defined in such a way to reduce control traffic overhead.

D-MORA

The second version of MORA is scalable and it is called Distributed-MORA.

Let $k \in \mathbf{N}^+$. Again the short probe message is sent from destination back to source. Every k hops the current node receiving it polls for information its neighboring nodes, considering only those with bigger value of y in order to avoid loops (y is related to the distance from the destination as in Section 2.1). The probe message is then forwarded to the neighbor with the higher value of m, attaching path information as in U-MORA. The state diagram for node j is presented in Figure 5.

If a node has no possibilities to forward the packet, it removes its identifier from the packet, increases the value of k by one and returns the packet to the node from which it originally received it.

probe message is received from node i with value M_i

node j

if k > 0 and $y_j > y_i$ then the packet is forwarded
after decreasing k of one; if k = 0 and $y_j > y_i$
m_i is calculated for each neighboring node i and the
packet is forwarded to the node with maximum
value of m_i after resetting k•1

Figure 5. D-MORA flow chart

4   Comparison with existing routing schemes

This section outlines potential advantages and disadvantages of the MORA approach with respect to other existing routing algorithms taking as a starting point the taxonomy of position-based routing protocols proposed in [4]. Table 1 reports the selected features of some routing algorithms. Exploiting the knowledge of the hosts position could be not enough in a conÞguration with frequent topological changes. In such a situation it is important to try to garantee links stability and therefore robustness of the routing protocol. The idea behind MORA is to take also into consideration also the direction of movement of the nodes in order to try to Þnd a solution to this critical problem.

If only position information is used, it is possible to lose some good candidates to forward the packet. For example, considering Location Aided Routing and Distance Routing Effect Algorithm for Mobility if one host, moving in direction of sd, is out of the Órequest zoneÓit will be never considered. Similarly, Most Forward within Radius doesnÕt care if the selected next hop is moving in the wrong direction (exactly in direction of the source node for example). Another advantage of MORA over LAR and DREAM concerns ßooding: U-MORA signiÞcantly reduces ßooding, while D-MORA quite completely eliminates it (experimental results are under evaluation).

Depth First Search could appear similar to D-MORA, since the decision among direct neighbors is taken by minimizig a distance function. However, with this method links are unstable if the topology is highly dynamic.

Shortest-path-based solution are also very sensitive to small changes in local topology and activity status. On the contrary, MORA is adaptive to Ósleep periodÓoperation, since power consumption is extremely reduced for inactive nodes and only a few nodes are involved in packet routing.

Detailed simulation is in progress using the Global Mobile Simulator (GloMoSim) Library [5]. From a preliminary analysis, U-MORA approximately generates the same number of events as AODV or LAR with very similar exe-

Table 1. Routing protocols classification

Method	Pos inf	Path str	Metric	Scal
shortest path	no	single	N hop	no
MFR	pos	single	N hop	yes
LAR, DREAM	pos	ßood	N hop	no
DFS	pos	single	N hop	yes
power aw	pos	single	power	yes
U-MORA	pos/mov	mult	comb	no
D-MORA	pos/mov	single	comb	yes

cution time: in the case of 10, 50, 300 nodes uniformly placed in 300 sm, as well as for 10 nodes moving *random-waypoint* (pause 2 s, min/max speed 0,1/10 m/s).

5   Conclusions

In this paper, a motion-based routing algorithm for ad hoc networks (MORA) is proposed. The algorithm is completely distributed, since nodes need to communicate only with direct neighbors in their transmission range, and utilizes a speciÞc metric, which exploits not only the position, but also the direction of movement of mobile hosts. Considerations outline that MORA represents a good solution in cases of high-mobility of the terminals.

Future work will provide a detailed numerical and statistical evaluation of the MORA routing protocol and extensive comparison with other existing approaches. In addition, the problem of accuracy in the knowledge of the position of the nodes will be studied.

References

[1] J.Broch et al. *A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols*. Proc. 4th ACM/IEEE IntÕl. Conf. Mobile Computing and Networking MOBICOM, Dallas, USA, 1998, pp. 85-97.

[2] E.Royer and C.Toh. *A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks*. IEEE Personal Communications, Vol 6, pp. 46-55, April 1999.

[3] M.Mauve et al. *A Survey on Position-Based Routing in Mobile Ad Hoc Networks*. IEEE Network, Volume 15, Issue 6, pp. 30-39, November/December 2001.

[4] I.Stojmenovic. *Position-Based Routing in Ad Hoc Networks*. IEEE Communic. Magazine, pp. 2-8, July 2002.

[5] UCLA Parallel Computing Laboratory and Wireless Adaptive Mobility Laboratory. *GloMoSim: A Scalable Simulation Environment for Wireless and Wired Network Systems*. http://pcl.cs.ucla.edu/projects/domains/glomosim.html.

A Middleware System for Flexible Intercommunications

Koji Hashimoto and Yoshitaka Shibata
Faculty of Software and Information Science, Iwate Prefectural University
152-52 Sugo, Takizawa, Iwate, Japan 020-0193
{hashi, shibata}@iwate-pu.ac.jp

Abstract

Although we can construct intercommunication environments by high quality video multicast streams such as DV (Digital Video) streams on super high speed networks, it is difficult for the reasonable network service users to join the high quality communication session. If we can use transcoding functions that include translator and mixer functions defined by RTP, it is able to transcode a media stream format into another stream format. Media transcoding functions are usefulness for intercommunication on heterogeneous computer networks. However, it is difficult for distributed communication systems to prepare the suitable transcoding nodes in advance because the required intercommunication environments are not always static. Accordingly, we should consider dynamic transcoding functions with relocatable decision methods. In this paper we propose a novel middleware system that realizes dynamic transcoding functions by extendable media stream.

1. Introduction

The advent of high performance computers that can process and integrate audio, video, graphics and text on interconnected high speed networks has constructed multimedia communication environment. At ordinary times, we actually can use multimedia applications such as IP telephone, IP radio, Video-on-Demand system, multimedia teleconference system and so on. We can communicate with remote locations easier than before by using these multimedia applications.

When we communicate with each other as long as we use current communication systems we must select media codec and format according to available network bandwidth and computing power. Moreover, we must prepare some communication systems that are constructed from software and hardware to some locations for communications, but if we can use super high speed network such as JGN (Japan Gigabit Network) [1] we can construct intercommunication environments by high quality video multicast streams such as DV (Digital Video) that uses about 28.8Mbps per a stream.

However it is difficult for us to join the high quality communication session on a business trip or from my home. To know all communication environments of all participants is very difficult, and even if we can join the high quality DV session, realtime communications may fail by lack of bandwidth and computing power. If translator and mixer functions defined by RTP [2] are available on the communication environments, it is able to transcode the DV stream into another format of stream such as MPEG, M-JPEG and H.263. Suppose that the required intercommunication environments are static and the total number of participants are limited, required transcoding functions can be located into suitable intermediate nodes in communication path in advance. However the preparation is very difficult because the communication environments are not always static. Therefore we should consider on demand transcoding functions with relocatable decision method.

As network platform that can be customized by user, Active Network [3] has been proposed. The Active Network users can program about communication flow and each packet behavior [4,5,6]. In addition, the users can update program without shutdown of network equipments. However, a new system architecture is required over transport layer in order to dynamically relocate transcoding functions and regulate frame rate and image quality in according to users' environments and QoS (quality of service) [7,8] requirements.

On the other hand, as middleware platform that can migrate executing program, Aglets [9], Voyager [10], Plangent [11] have been proposed. They are mobile agent platform and various applied researches are progressing on the mobile agent platforms. The mobile agents that perform given tasks while migrating on some hosts connected with computer networks are one

of important base technologies for constructing dynamic intercommunication environment.

In this paper we propose a novel middleware system that realizes media processing and session management by using mobile agent based extended media stream.

2. MidField System

Fig. 1 shows the system architecture to realize constructing dynamic intercommunication environment. The middleware system name is MidField (**Mid**dleware for **F**lexible **i**ntercommunication **e**nvironment by re**l**ocatable **d**ecision). This system architecture is between the application and the transport layer. The system is constructed by 3 layers and 4 vertical planes and offers multimedia communication functions to the application layer. Stream Plane is constructed by synchronization, data transform and media flow control layer, and performs multimedia stream processing. Session Plane performs management of communication sessions. System Plane monitors network traffic and CPU rate in the local host, and performs admission tests for QoS requirements from system user. Event Process Plane processes various events that are created in the system.

Figure 1. MidField System Architecture

In the Session Plane of MidField System, information of both participants and media streams is handled. It should be considered that some new functions for handling this information are required by various session types. Therefore if both these information and the handling functions can transfer to intermediate node that performs to transcode, the system will have flexibility for updating new management functions for new session types. In addition, if there are no suitable func-

tions to process media in local computer system, the system should have some functions to get required media processing functions from the other computer systems.

The system is able to construct intercommunication environment on computer networks dynamically according to the environment of users and QoS requirements from users. Fig. 2 shows an abstract of MidField Session. A MidField Session consists of at least one multicast session and offers peer-to-peer communication to system users. In Fig. 2, three system users (MF1, MF2, MF3) join to a MidField Session. At this point, both MF1 and MF2 have enough communication environments to transmit DV stream. On the other hand, MF3 can't transmit DV stream because the environment doesn't have enough bandwidth and computing power to handle DV stream. Therefore MF3 requests to use MPEG4 stream to join the MidField Session. In such case, MidField System locates required transcoding functions into suitable node on computer networks to communicate with each other. In this example, both MF4 and MF5 transcode from DV stream into MPEG4. By using transcoding functions, MF3 can join to the MidField Session.

Figure 2. MidField Session

2.1 Functional Modules

The functional modules of MidField System are shown in Fig. 3. The functions of each plane in MidField are offered to application program by Stream Interface, Session Interface and System Interface. Each processing in MidField for these interfaces is performed by Stream Agent, Session Agent and System Agent. Stream Agent has a Stream Segment for RTP streaming and transmits RTP streams in the Stream Segment and transcodes media data of RTP packet. Session Agent has a Session Property that includes information of MidField Session. As the need arises,

these agents make clone of it and can migrate to the other MidField Systems. System Agent monitors cpu rate and network traffic in local computer and performs admission test when system users join to MidField Session.

Figure 3. Functional Modules

Agent Place manages all agents in the local system and performs creation, migration and termination of agents, and has Agent Loader and Agent Server for realizing of agent migration. Agent Place is also an agent itself in MidField System.

These agents can exchange messages with the other agents. Connection Manager in Event Process Plane manages objects of connected socket interface and performs exclusive control for message delivery with no problems. On the other hand, System Event Manager has mechanisms to process various events that are created by each agent in the system. If an agent creates some events, the agent posts the events to System Event Manager and the events are stored into Event Queue. The stored events are processed by idled Event Processor that has a Thread. Thus, MidField System can perform unified event management and priority based event processing.

2.2 MidField Stream

Fig. 4 shows an overview of extendable MidField Stream. The StreamAgnet0 has an input stream from DV capture device and the output is a DV over RTP stream to the multicast session A. The StreamAgent1 receives the DV over RTP stream as an input stream and creates an output DV over RTP stream to another multicast session B. This stream agent doesn't transcode the RTP stream, but relays the source RTP stream from a multicast session to another multicast session. On the other hand, the StreamAgent2 also receives the DV over RTP stream. And the received RTP stream is transcoded into MPEG4 and PCM over RTP streams to multicast session C.

Figure 4. Extendable MidField Stream

Here, it is the most important point that these stream agents perform in suitable transcoding nodes. The next section shows how to use the extendable MidField Streams in a MidField Session.

3. Dynamic Configuration

The MidField System creates some RTP sessions in a MidField Session as the need arises. Then MidField System locates transcoding functions between the RTP sessions to configure a flexible intercommunication environment. A RTP Session is on a IP multicast session. Therefore, in order to send a RTP stream, it should be checked that there are enough bandwidth and CPU power at the sender and all receivers. On the other hand, in order to receive a RTP stream, the system can use dynamic mechanism to locate transcoding functions in suitable intermediate node in according to bandwidth and CPU power of the receiver.

In fig. 5, there are 5 MidField Systems on computer networks. MF1 and MF2 will communicate with each other by using DV multicast streams, then MF3 will be able to use MPEG4 streams to join the communication session of MF1 and MF2.

At first, this session agent in MF1 creates a new MidField Session. Next, the session agent in MF2 joins the MidField Session. Now, fortunately both MF1 and MF2 have a communication environment enough to use DV multicast streams. So, MF1 and MF2 start to communicate by DV multicast streams.

Well, the MF3 wants to join the DV session. However, unfortunately MF3 doesn't have a communication environment enough to use DV streams. Accordingly MF3 can't join the DV session, so MF3 joins another session for MPEG4 streams. When the MF3 joins the

Figure 5. Dynamic Configuration of a MidField Session

MPEG4 session, the stream agents in MF1 and MF2 start to search for transcoding node. And then MF4 and MF5 are selected by the stream agents in MF1 and MF2 respectively. Then, the DV multicast streams are extended for MF3.

On the other hand, when the MF3 starts to send a MPEG4 video stream to the MidField Session, MF3 also searches for transcoding node to relay the MPEG4 stream to the upper DV session. In fig. 5, MF5 is selected as the rely node, and finally MF1 and MF2 will be able to receive the MPEG4 stream from MF3 via MF5.

Thus, MidField System constructs dynamic inter-communication environments according to available resources.

4. Stream Segment for Transcoding

Fig. 6 shows a configuration of a StreamSegment for media transcoding. The StreamSegment has a MediaProcessor for processing of media stream and the MediaProcessor realizes media processing by connecting some suitable Plug-In modules. The MediaProcessor can take a capture device or file or RTP stream as input. The input data stream is processed at unit of a frame or a packet. On the other hand, processed media data stream are outputted by the unit of a frame or a packet.

In the Plug-In modules, each RTPSender and RTPReceiver performs RTP packets transmission. The RTPPacketizer and RTPDepacketizer perform packetization and depacketization respectively. The MediaProcessor can process a media stream by connecting these Plug-In modules that include various codecs, demultiplexers and rendered. In addition, these Plug-In modules have control modules to control media processing as external interfaces. Therefore a StreamAgent can control media processing.

Figure 6. Configuration of a StreamSegment

A. Receiving RTP Packets

As describe above, RTPReceiver can receive RTP packets. Fig.7 shows a data flow and processing methods to deliver depacketized RTP packets to a suitable codec module.

The RTPReceiver uses two socket interfaces for RTP and RTCP, and has an RTPReceiveQueue to deliver incoming RTP packets to RTPDepacketizer. The RTPReceiveQueue has two packet queues; one is filled with incoming RTP packets and another is free queue. Some pointers to an allocated memory bloc for incoming packets go to and from these two queues.

Figure 7. Receive and Depacketize RTP Packets

At first, the RTPReceive() method in fig. 6 receives a RTP packet, then the DispatchRTPPacket() method dispatches the RTP packet to a suitable RTPReceiveQueue. In MidField System, the RTPReceiveQueue(s) are managed per RTP sender's CNAME associated with SSRC value in RTP packets. Therefore, RTP packets can be dispatched into suitable RTPReceiveQueue. If the RTPReceiver has a suitable RTPReceiveQueue for an incoming RTP packet, the GetFreePacket() method takes a free pointer and copies the incoming packet payload to the pointer. Then, the copied packet payload is added to the filled queue by PutFillPacket() method. On the other hand, The RTPDepacketizer takes incoming packet payload by using GetFillPacket() method and copies the payload to an allocated memory bloc for a frame data. When the RTPDepacketizer finishes copy of a frame data from RTP packets, the RTPDepacketizer delivers the frame data to suitable codec.

B. Sending RTP Packets

The RTPSender in fig. 8 sends RTP packets. Fig. 8 shows a data flow and processing methods to deliver outgoing RTP packets.

The RTPPacketizer takes a frame data from a connecting codec module and appends packetized RTP packets to the RTPSendQueue.When the RTPPacketizer finishes packetizing the frame data the StartTransmission() method is called. The StartTransmission() method sends a transmission start message to the RTPSender. Then the RTPSender starts sending RTP packets by using the methods GetFillPacket(), RTPSend() and PutFreePacket(). In MidField System, a set of RTPSendQueue and RTPReceiveQueue is created in a MediaProcessor per a SSRC value that is included in RTP packet. Therefore the system can identify some senders in a RTP session.

Figure 8. Packetize and Send RTP Packets

5. Prototype System

In order to implement a prototyped MidField System, we use Java (Ver.1.4), C and C++ programming languages on Windows XP. Above described codec functions are realized by using some filters of DirectShow (DirectX 9.0) [12]. Some RTPPacketizer and RTPDepacketizer are implemented for each media type as the filter of DirectShow. The RTPSender and RTPReceiver are implemented by using Lucent Technologies RTPlib 1.02b [13].

So far we have implemented a prototype system that includes the System Plane, Event Process Plane and Stream Plane in fig. 1. Although the functions of Session Plane are not available yet, now we are testing implemented functions. In the Stream Plane, the prototype system has been able to process DV, MotionJPEG, MPEG4 and PCM streams. Fig. 9 shows a simple sample code for sending and transcoding a RTP stream by using external interfaces of the prototype system. At first, the MidField System instance is created by MFSystem.invokeSystem() method. Next, the instance creates a new MFStream instance. In this sample code, the MFStream instance takes capture device DVCam as

input. The output is DV stream. Then, the streaming process starts by calling sender.start() method.

After that, a new MFStream instance is created as transcoder. In this example, the transcoder's input is the DV stream and output is MPEG4 and PCM streams. Here, these IP multicast sessions for input and output are different. Moreover, in line 17, xcoder.migrate() method can be called for migration to suitable transcoding node. After this call, the transcoder will migrate to another MidField System that is running for media transcoding. Then, the transcoder will perform the stream data processing according to given input and output.

```
01: // Invoke the MidField System
02: MFSystem mfs = MFSystem.invokeSystem();
03:
04: // Create a new send stream and start to send the stream
05: // IN : Video capture device (DVCam) , OUT : RTP stream (DV/RTP)
06: MFStream sender = mfs.newStream();
07: sender.setInput(
        "capture://DVCamera&Default;");
08: sender.setOutput(
        "stream://StreamAgent00/239.192. 0. 0/20000/2&DV_RTP;");
09: sender.open();
10: sender.start();
11:
12: // Create a new transcoding stream, migrate and start to transcode the stream
13: // IN : RTP stream (DV/RTP), OUT : RTP stream (MPEG4/RTP)
14: MFStream xcoder = mfs.newStream();
15: xcoder.setInput(
        "stream://StreamAgent00/239.192. 0. 0/20000/2&DV_RTP;");
16: xcoder.setOutput(
        "stream://TranscodingAgent01/239.192. 0.1/20000/2&MPEG4_360X240_RTP;
        stream://TranscodingAgent02/239.192. 0.1/20000/2&PCM_RTP;");
17: xcoder.migrate("destination_hostname");
```

Figure 9. Simple Sample Code for Streaming

6. Conclusion

In this paper we have proposed a novel middleware system for realizing flexible intercommunication. The system architecture is based on mobile agent technology and can construct flexible intercommunication environment by using relocatable mobile agent. Now, we are implementing the prototype system on Windows XP. Although the functions of Session Plane are not available yet, now we are testing implemented functions. In the Stream Plane, the prototype system has been able to process DV, Motion JPEG, MPEG4 and PCM streams.

As future works, we will design required modules for MidField Session in detail and implement the modules. The modules include session announcement protocol, exchange method for session locator and relocatable decision method according to CPU rate and available network bandwidth. In addition, we will construct pro-

totype system and evaluate the functions for dynamic configuration of intercommunication environments.

References

[1] http://www.jgn.tao.go.jp/english/index_E.html
[2] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson. : RTP: A Transport Protocol for Real-Time Applications, RFC 1889, January 1996.
[3] Andrew T. Campbell, Herman G. De Meer, Michael E. Kounavis, Kazuho Miki, John B. Vicente, and Daniel Villela : A Survey of Programmable Networks, ACM Computer Communications Review, April 1999.
[4] Michael Hicks, Jonathan T. Moore, D. Scott Alexander, Carl A. Gunter, and Scott M. Nettles : PLANet: An Active Internetwork, Eighteenth IEEE Computer and Communication Society INFOCOM Conference, pp.1124-1133, March 1999.
[5] Dan Decasper, Zubin Dittia, Guru Parulkar and Bernhard Plattner : Router Plugins: A Software Architecture for Next Generation Routers, ACM SIGCOMM'98, pp.229-240, September 1998.
[6] Andrew T. Campbell, Herman G. De Meer, Michael E. Kounavis, Kazuho Miki, John B. Vicente, and Daniel Villela : The Genesis Kernel: A Virtual Network Operating System for Spawning Network Architectures, 2nd IEEE International Conference on Open Architectures and Network Programming(OPENARCH'99), March 1999.
[7] Matti A. Hiltunen, Richard D. Schlichting, Xiaonan Han, Melvin M. Cardozo, and Rajsekhar Das : Real-Time Dependable Channels: Customizing QoS Attributes for Distributed Systems, IEEE Transactions on Parallel and Distributed Systems, Vol.10, No.6, pp.600-612, 1999.
[8] Gary J. Nutt, Scott Brandt, Adam J. Griff, Sam Siewert, Marty Humphrey, and Toby Berk : Dynamically Negotiated Resource Management for Data Intensive Application Suites, IEEE Transactions on Knowledge and Data Engineering, Vol.12, No.1, pp.78-95, 2000.
[9] http://www.trl.ibm.com/aglets/
[10] http://www.recursionsw.com/products/voyager/
[11] Ohsuga A., Nagai Y., Irie Y., Hattori M., and Honiden S. : PLANGENT: An Approach to Making Mobile Agents Intelligent, IEEE Internet Computing, Vol.1, No.4(1997), pp.50-57
[12] http://www.microsoft.com/windows/directx/
[13] http://www-out.bell-labs.com/project/RTPlib/

Toward a Multi-Mobile Agent System on Semantic Web Service: Design Support in Supply Chain

Daisuke Maruyama
University of Aizu Graduate
School of Computer Science
and Engineering,
Aizu-Wakamatsu City,
Fukushima, Japan
E-mail: dai@ebiz.u-aizu.ac.jp

Incheon Paik
School of Computer Science
& Engineering,
the University of Aizu,
Aizu-wakamatsu,
Fukushima, Japan
E-mail: paikic@u-aizu.ac.jp

Yuu Watanabe
University of Aizu Graduate
School of Computer Science
and Engineering,
Aizu-Wakamatsu City,
Fukushima, Japan
E-mail: yuu@ebiz.u-aizu.ac.jp

ABSTRACT

In the supply chain process, the initial design phase is especially important from the perspective of cost reduction, and a more efficient system to support the design phase such as automating agent is required. The semantic Web provides an excellent environment for the information infrastructure that integrates product design attributes: component-cost, quality, function and technology. The infrastructure makes it possible for agents to support the design stage efficiently with semantics. In this paper, an automating agent system to support product design is designed and implemented with multi- and mobile agent approaches in collaboration with the information infrastructure. Reduction of network traffic, split application, and parallel operation of the product design task are achieved. The performance is also evaluated based on the measurement of time cost for the product design. This system did not achieve a satisfactory level of performance. Nevertheless, the multi-mobile agent system has potential for real supply chain networks with adaptation of flexibility and cooperation.

Keywords

Supply Chain, Semantic Web Service, Design Support, Multi-Mobile Agent

1 INTRODUCTION

E-business on the Internet is rapidly becoming ubiquitous. The methodology and technology that make business processes more efficient such as automation of Supply Chain Management (SCM) and Web service are attracting the attention of many researchers.

SCM is a critical part for e-Business. It can reduce a series of costs and raises corporate profits by coordinating and integrating flows of materials and finances moving in a process from supplier to manufacturer, wholesaler, retailer and customers.

The information infrastructure [1][2] integrates four product design attributes: component-cost, quality, func-tion, and technology. The infrastructure considers the relationship of the attributes and utilizes ontological methodology. It also provides agents and users aim interoperable service interfaces to access its integrated attributes and supports the product design stage.

The semantic Web service provides a powerful environment for software agents to perform assigned task automatically and efficiently with well-defined, flexible and expressive information. It even enhances agent capability.

In the SCM process, time and development terms of products are very important because companies compete with each other. In the design phase, a more efficient system to support product design such as automating agents is required to reduce the time cost.

In this research, a multi-mobile agent system to support product design in the semantic Web service is proposed. The initial agent system [2] tackled this design support using semantic Web service, and this agent system used just static stationary agent. However, the potential of agents can be tapped by adopting multi- and mobile agent technology. The multi-agent is an agent system with multiple agents that divide a large goal into manageable sizes and cooperate with each other to achieve the goal. This cooperative ability is very attractive because the agent system can achieve assigned jobs flexibly, dynamically and efficiently. Mobile agents [3] also have the ability to travel from site to site in a heterogeneous network and reduce network load and overcome network latency. In this paper, we describe the construction of a more useful agent system to support product design with this technology.

The organization of the paper is as follows. In Section 2, we describe the importance of design phase in the SCM process and the characteristic of the information infrastructure on semantic Web service [2] that supports the design stage. Section 3 reviews mobile agent technology and its java implementation. Section 4 explains the initial agent architecture to support product design. In Section 5, the architecture of multi-mobile agent system is described. Section 6 shows experimental environment, method and result. In Section 7, we discuss the multi-mobile agent system and

evaluate it. Conclusion and future work are given in the final section.

2 INFORMATION INFRASTRUCTURE FOR DESIGN SUPPORT IN SUPPLY CHAIN

In the SCM process, the consideration of the initial plan and design stage is especially important because careful construction of this stage leads to reducing the unnecessary cost of redesigning and modifying the product. The developing cost of the product is small in the initial phase, but the cost becomes higher toward the end of the production stage, so the supporting design phase in a supply chain is extremely significant from the perspective of cost reduction [1].

The semantic Web environment where our agents will work was developed as the information infrastructure for design support [2]. The infrastructure coordinates four design attributes information: component-cost, quality, function, and technology with their relation and ontological paradigm and offers a combined Web service interface to agents. With this environment, the agent can provide high-quality service such as multilateral inspection for design attributes to support the design phase.

3 JAVA MOBILE AGENT TECHNOLOGY

3.1 Mobile Agent Technology

It is necessary for distributed systems to perform multiple interactions in order to achieve assigned jobs. These systems essentially need high network traffic because of interactions over the network. A mobile agent has the ability to move its own code and state to a remote location, so the agent can move to a remote host and complete the distributed job at the remote machine, and then return itself or only its result over the network. This characteristic converts interactions over the network into interactions on a local host and reduces network traffic. The reduction of network traffic leads to protection against network latency. (see Figure 1)

Figure 1. Mobile agent and reduction of network traffic.

3.2 Java Mobile Agent Implementation

The Aglet Mobile Agent Development Kit (ASDK) [4] developed by IBM Japan and utilized in this research is an agent framework made by Java language for mobile agents in the Internet. ASDK provides schemes for moving code, data and state information from one site to another as a Java class library. The mobile agent program can be implemented easily using this API (Application Programming Interface). The agent program with ASDK is platform independent and highly portable because of its Java-based implementation.

4 THE INITIAL AGENT ARCHITECTURE TO SUPPORT PRODUCT DESIGN

The initial design support agent [2], described in Figure 2, retrieves and procures attribute information of component parts and reasons about which component combination is appropriate for user preference with an if-then rule-based reasoning engine [5][6] and design product. The client agent accesses the service interface of the information infrastructure and procures attribute information directly.

This agent system has problems in that the information procurement requires high network traffic because each interaction occurs over the network. Moreover, the network load exponentially increases as the number of supplier sites grows, and over-concentration of client transactions occurs because a client is only one yet needs to deal with a lot of information from multiple suppliers. This results in taking a lot of time to design products. Reduction of network traffic and the split application and parallel operation of product design tasks are needed to save time costs.

Figure 2. Initial agent architecture to support product design.

5 THE ARCHITECTURE OF A MULTI-MO-BILE AGENT SYSTEM TO SUPPORT PRODUCT DESIGN

Figure 3 shows the architecture of the multi-mobile agent system to support product design. This architecture makes it possible for client agents to perform split application of an assigned task and parallel processing of product design leading to the reduction of network load by essentially freeing the procurement of attribute information and the reasoning transaction from the client agent. Required tasks are divided into supplier-based blocks and delegated to independent mobile agents, and then the mobile agents perform the job autonomously.

This system involves a client, a supplier, and mobile agents and stationary agents for clients and suppliers. The client creates the mobile agent and dispatches the mobile agent to the supplier site which has attribute information of component parts. The mobile agent retrieves and procures attribute information. It then reasons and designs a product automatically according to user preference and returns the designed data to the client.

Figure 3. Multi-mobile agent architecture to support product design.

5.1 Mobile Agents

In this system, there are two kinds of mobile agents. The first is the leader mobile agent, and the second is the worker mobile agent. Mobile agents visit supplier sites and procure component attributes information and design desired products. Figure 4 shows the flow from mobile agent creation to mobile agent disposal.

Figure 4. Sequence diagram that shows the flow from mobile agent creation to mobile agent disposal.

5.1.1 Leader Mobile Agents

The leader mobile agent is a leader agent of the worker mobile agents and manages them. The leader mobile agent is created and initialized with the conditions of user preference, a list of suppliers to be visited and user account information. The leader is dispatched to one supplier site by a client stationary agent. Then it generates the worker mobile agents for only the number of supplier sites and initializes them with the user preference and the account information. The leader mobile agent allocates one supplier sites as the destination for each worker mobile agent and dispatches each to its assigned site. Then the leader mobile agent waits for the result message of product design from the mobile worker agents, the request and instruction message from the client stationary agent. When the result message comes, the leader mobile agent receives the message and sends the result to the client stationary agent in an arrival order. If the instruction of agent disposal comes, the leader mobile agent sends disposal message to the worker mobile agents. After the confirmation of the worker mobile agent disposal, the leader mobile agent disposes itself (see Figure 4).

5.1.2 Worker Mobile Agents

The worker mobile agent is an actual worker that supports product design. The worker mobile agent is created for only the number of supplier sites. It is initialized with the information of user preference and user account and then assigned a destination and dispatched to the supplier site by the leader mobile agent. The worker mobile agent authenticates the user based on name and password and communicates with the supplier stationary agent, getting the attribute information and reasoning about if component attribute information appropriate for user preference exists. If attribute information that meets user demands exists, the worker mobile agent designs the product automatically and sends the designed data to the leader mobile agent. After the first design, the worker mobile agent stays in the assigned supplier site and waits for new request and instruction message from the leader mobile agent. If new request message comes, the worker mobile agent performs design based on new user preference and sends the result to the leader mobile agent. If disposal message comes, the worker mobile agent disposes itself (see Figure 4).

5.2 Client Site

The client site is composed of the client graphical user interface (GUI) and the client stationary agent. This client mediates between the user and the agent system, supporting user data entry and the resulting data presentation and controlling the creation and disposal of the mobile agent.

5.2.1 GUI

The client GUI helps the user to input individual preference data and to instruct the creation of the mobile agent. It displays the results and the condition log of mobile agents to the user.

5.2.2 Client Stationary Agent

The client stationary agent creates the mobile agent and initializes it with the user preference and list of suppliers to be visited. Then the client stationary agent dispatches the mobile agent to the supplier site. If the dispatch of mobile agent is finished, and the user requests a new preference, the client stationary agent sends the new request as a message to the mobile agent without a new dispatch of mobile agents. After reception of the designed product data from the leader mobile agent, the client stationary agent shows the result to the user through the GUI.

5.3 Supplier Site

5.3.1 Information Infrastructure

The information infrastructure includes the well defined database of component attribute information and provides the interface to access the database and pull up demanded data flexibly and efficiently.

5.3.2 Supplier Stationary Agent

The supplier stationary agent waits until the worker mobile agent visits. When the mobile agent arrives at the supplier site and sends the message for authentication, the supplier stationary agent inquires for authentication of the user based on name and password sent by the mobile agent to the information infrastructure through the interface. If the authentication is successful, the stationary agent waits for the request message of the mobile agent. When the request comes, the stationary agent accesses the interface, retrieves and procures the requested data, and returns the data to the mobile agent.

6 EXPERIMENT

6.1 Environment

In this experiment, the following environment was constructed and utilized. Table 1 shows the experimental environment of the client and supplier site. One computer as a client site and eight computers as supplier sites are equipped.

Table 1. Supplier and client site environment

Location	The University of Aizu
Workstation	SUN Blade 100
Operating System	SUN OS 5.8
Aglet Software	ASDK 2.0.2
JDK Version	1.4.2_03

6.2 Method

An experiment regarding two different architectures, the initial agent system and the multi-mobile agent system, is performed for evaluating the multi-mobile agent system to

support product design. This experiment is performed in the following situation.

The initial agent system:

- The case of using thread processing
- The case of not using thread processing

The multi-mobile agent system:

- The case of processing starting from the initialization and dispatching of mobile agents
- The case of processing starting from the completion of mobile agent dispatch

The time taken in milliseconds for designing a product, which is the time from the user giving an instruction of product design to the display of the design result, was measured 10 times for each architecture with a common design preference. The average time was considered the product design time (time cost) of the architecture. The experiment was implemented cases in which there were two, four, and eight supplier sites, and the time taken for dispatching mobile agent to supplier sites was also measured. The conditions of user preference were that light source brightness be greater than 7.5 as a quality product design attribute, the price of the product be medium, the age of the user be young, and the life time of product be long as user's preferences.

6.3 Results

Figure 5. Comparison of time cost between the initial agent system and the multi mobile agent system

Figure 5 shows the experimental results, the time taken for product design for each architecture. The black dot describes the time cost of the initial agent system with no thread processing. The white square dot shows the time cost of the multi-mobile agent system in the case of processing starting from the initialization and dispatching of mobile agents. The black square indicates the time cost of the multi-mobile agent system in the case of processing starting

from the completion of mobile agent dispatch. The white triangle dot shows the initial agent system with thread processing.

7 EVALUATION AND DISCUSSION

7.1 Evaluation of Multi-mobile Agent System

In every cases, with two, four, or eight site, the multi-mobile agent system was able to shorten the time it takes for designing a product compared to the initial agent with no thread processing, and the time cost was suppressed in spite of increases of supplier sites. These results mean that the parallel operation using multiple mobile agents had the desired effect. However, the performance, while not so bad, was inferior to the initial agent system with thread processing. While the initial agent system performed product design with simple methodology, the multi-mobile agent system needed several steps such as the initialization and dispatching of mobile agents in order to start product design. This additional transaction is thought to be a major cause of the performance disparity.

7.2 Discussion

MAgNET opened the possibility of mobile agent to apply to procurement in e-Business [7]. And it showed simple procurement scenario and experimental result through the network globally. Our work has improved the static architecture in the initial agent system [2]. Although the initial agent system with thread processing was the best methodology to support product design based on the experimental results [2], there are merits to utilize this multi-mobile agent system. The advantages are the reduction of network traffic, the protection against network latency, and system flexibility. These experimental results were based on an experiment performed using the local area network of The University of Aizu. As this network has uniform network conditions, careful consideration of network latency and reduction of network traffic were not needed. Real supply chain networks are composed of various conditions of bandwidth, and enormous quantities of data are traded over the networks. It is necessary to respond to narrow-band connection flexibly and to process of a lot of information. The multi-mobile agent system has the capability of meeting these necessities with its system characteristics. This system also has flexibility in that it is possible to construct more useful services such as the notification of the infrastructure service update easily by extending the capability of supplier stationary agents. That is why this system is considered beneficial and worthy of utilization.

8 CONCLUSION

An automating agent system to support product design was designed and implemented with multi- and mobile agent methodology, and the performance was evaluated based on the measurement of the time cost for the product design. In this experiment, this system did not achieve a satisfactory level of performance, but the agent system had positive features that the initial agent system did not have. It has the potential to show real talent in the real network.

The multi-mobile agent system constructed in this thesis is one applied example of multi-agent technology. In multi-agent systems, multiple agents achieve an assigned job in collaboration with each other. This cooperative ability leads to flexibility and efficiency of problem solving. By applying it appropriately, the multi-agent technology has potential to be one key methodology in constructing an efficient system and converting existing system into a more flexible system.

In this paper, an efficient method in which the mobile agent was dispatched to a supplier site according to supplier's location, distance and traffic route from client to supplier site was not argued. It is very important to consider where the mobile agent is dispatched based on the network structure in order to save unnecessary time cost and make use of this agent system.

In the future, it will be necessary to research an algorithm for dispatching mobile agents suitable for the network structure and more general multi-agent architecture, and mobile agent details, including network latency considerations, and adaptive communication.

ACKNOWLEDGMENTS

REFERENCES

[1] I. Paik, W. Park. "Software Component Architecture for an Information Infrastructure to Support Innovative Product Design in a Supply Chain," Forth coming in JOCEC (Journal of Organization Computing and Electronic Commerce), Lawrence Erlbaum Associates, NJ.

[2] I.Paik, H.Akimoto and S.Takami, "Automating Design in Supply Chains on Semantic Web Services," Proceedings of 13th WWW Conference Workshop, NY 2004.

[3] Danny B. Lange and Mitsuru Oshima, *Programming and Deploying java Mobile agents with Aglets*, Addison-Wesley, 1998.

[4] "IBM Aglets Software Development Kit," http://www.trl.ibm.com/aglets/ (current Apr. 2003)

[5] Joseph P. Bigus, Jennifer Bigus. *Constructing Intelligent Agents using Java, 2/Edition*, John Wiley & Sons 2001.

[6] Stuart Russell, Peter Norvig. *Artificial Intelligence: A Modern Approach, 2/Edition*, Prentice Hall 2002.

[7] P. Dasgupta, N. Narasimhan, L.E. Moser and P. M. Melliar-Smith, "MAgNET: mobile agents for networked electronic trading," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 11, No. 4, July/August 1999, pp. 509-525.

LOAD-BALANCING IN DISTRIBUTED RETRIEVAL SYSTEM

Ming Hong Pi, Anup Basu and Hua Li[1]
Department of Computing Science
University of Alberta, Edmonton AB, T6G 2V4, Canada
[1]Dept. of Mathematics and Computer Science, University of Lethbridge
Lethbridge, AB, T1K 3M4
Email: {minghong, anup}@cs.ualberta.ca, huali@cs.uleth.ca

ABSTRACT

Limitations on the bandwidth to a server makes the retrieval of a large-size high-resolution image, even compressed, very time-consuming. To speed up this process we have replaced the retrieval process from a single server with the retrieval from multiple servers, i.e., simultaneously retrieving different part of an image from multiple server sites. Specifically, in a related work we have implemented a block-based distributed retrieval strategy for Wavelet compressed images. In the paper, we model block-based distributed retrieval as a combinatorial optimization problem, and prove this problem is NP-Complete. We propose a heuristic algorithm to solve the optimization problem. Experimental results are given to show that the heuristic algorithm performs reasonably well in the initial tests performed.

Keywords: Load balancing, Computational complexity, NP-Completeness, Heuristic algorithm

1. BLOCK-BASED DISTRIBUTED RETRIEVAL

Thus far, conventional retrieval across the Internet relies on transmission efficiency from a server to a client. However, due to the server's bandwidth limitation, the retrieval of a large-size high-resolution image, even compressed, is very time-consuming. Hence, there is a compelling demand of multiple servers retrieving the same image in a parallel mode. A distributed retrieval system provides a scheme to speed up downloading. Multiple connections are opened between multiple servers $\#_1, \#_2, \ldots, \#_N$ and a single sink (client), and data is synchronously downloaded into the same sink from multiple servers. Because multiple connections together have more bandwidth capacity than a single connection, their collaboration shortens downloading time. In such a distributed retrieval system, one concern is how to equalize (load balance) the connections workload. The need for load balancing arises in a parallel and distributed system, there have been several books dealing with various aspect of this topic [1-2]. However, it still is an important and challenging area of research and attracts many researcher's attention [3-4].

Block-based distributed retrieval implies, partitioning an image I into a set of blocks ($I = \sum_{i=1}^{M} B_i$), and assigning multiple servers concurrently to retrieve them. It aims at minimizing retrieval time by scheduling retrieval task appropriately on multiple servers. Let t_{ij} be the time for transferring block B_i from the server $\#_j$ into the sink, the assignment of M blocks must be load balanced to make the sum of retrieval times at each server as close as possible. The problem can be modeled as the following combinatorial optimization problem:

$$\min \sum_{j,k=1}^{N} \left(\min\left\{ \sum_{i=1}^{M} t_{ij} \delta_{ij} \right\} - \min\left\{ \sum_{i=1}^{M} t_{ik} \delta_{ik} \right\} \right)^2 \quad (1)$$

$$\text{Subject to } \sum_{j} \delta_{ij} = 1 \quad i = 1, 2, \ldots, M$$

Here, δ_{ij} is block assignment function and is defined as:

$$\delta_{ij} = \begin{cases} 1 & if \quad B_i \in \#_j \\ 0 & otherwise \end{cases}$$

$B_i \in \#_j$ means B_i is transferred by $\#_j$, $\sum_{j} \delta_{ij} = 1$ means a block must be transferred by a server, rather than multiple servers. Solving (1), the minimum retrieval time of M blocks is:

$$T = \max_{1 \leq j \leq N} \left\{ \sum_{i=1}^{M} t_{ij} \delta_{ij} \right\}$$

At first, let us measure the complexity of (1). Block assignment set Ω is defined as:

$$\Omega = \{ \delta_{ij} : \sum_{j=1}^{N} \delta_{ij} = 1, \ i = 1, 2, \ldots, M \}$$

Ω contains N^M cases ($|\Omega| = C_N^1 \times C_N^1 \times \ldots \times C_N^1 = N^M$). Finding the minimal retrieval time by exhaustive enumeration has exponential complexity and is intractable as the number of servers N and number of blocks M

increase. For the convenience of theoretical analysis, t_{ij} is assumed to be an integer, otherwise, a tiny time atom can be introduced as the quantization factor to quantize t_{ij} into an integer.

To reduce the complexity of the problem, let us assume a distributed system consists of two servers and a sink, with transfer times of M blocks being expressed as a $M \times 2$ matrix:

$$Q = \begin{bmatrix} p_1 & q_1 \\ p_2 & q_2 \\ \vdots & \vdots \\ p_M & q_M \end{bmatrix} \qquad (2)$$

Here, both p_i and q_i are positive integer. (1) can thus be simplified into finding a partition $I_1 \cup I_2 = I$, such that

$$\min \left| \min(\sum_{i \in I_1} p_i) - \min(\sum_{i \in I_2} q_i) \right| \qquad (3)$$

To intuitively explain the above problem we give two examples:

$$Q_1 = \begin{bmatrix} 2 & 5 \\ 6 & 7 \\ 9 & 4 \\ 5 & 2 \end{bmatrix} \quad Q_2 = \begin{bmatrix} 2 & 5 \\ 6 & 7 \\ 1 & 4 \\ 5 & 2 \end{bmatrix} \qquad (4)$$

For the first, minimal retrieval time is max{2+6,4+2}=8, the partition is $I_1 = \{1,2\}$, $I_2 = \{3,4\}$. Let us transform Q_1 into Q_2 by a slight modification, then the minimal retrieval time is max{6+1,2+5}=7, the partition is $I_1 = \{2,3\}$, $I_2 = \{1,4\}$. Next, we remove "min" from (3) and further simplify, and show that even this simplified version is NP-Complete.

2. PROOF

We restate the above problem using the instance and question by removal of "min" in (3) and further simplification, and call it the Load Balance Problem (LBP). To show LBP is NP-Complete, we reduce Subset Sum Problem (SSP) [5] into LBP. For this purpose, we describe SSP.

LBP:

Instance:

Finite set of pairs $Q = \{(p_i, q_i) | i \in I = \{1,2,\ldots,M\}\}$ with $p_i, q_i \in Z^+$

Question:

Does there exists a partition $I_1 \cup I_2 = I$, such that

$$\sum_{i \in I_1} p_i = \sum_{i \in I_2} q_i$$

SSP:

Instance:

Given positive integers $a_1, a_2, \ldots, a_M$, B

Question:

Is there $I' \subseteq I = \{1,2,\ldots,M\}$, such that $\sum_{i \in I'} a_i = B$

LBP is in NP:

First note that LBP belongs to NP because given a partition as a guess, the equality question above can be verified in polynomial time.

REDUCTION:

To show LBP is NP-complete all we need to do is show that a known NP-complete problem can be reduced to LBP in polynomial time.

Given an instance of SSP, a known NP-complete problem, we generate an instance of LBP. Let k be the maximum index such that $\sum_{i=1}^{k} a_i \leq B$ and $\sum_{i=1}^{k+1} a_i \geq B$, and

$$\begin{cases} p_i = \begin{cases} a_i & 1 \leq i \leq k \\ B - \sum_{n=1}^{K} a_n & i = k+1 \\ 0 & k+2 \leq i \leq M \end{cases} \\ q_i = a_i - p_i & 1 \leq i \leq M \end{cases} \qquad (5)$$

Clearly the conversion is polynomial, in fact it is $O(M)$. For any partition (I_1, I_2) of I, define sum difference as

$$D(I_1, I_2) = \sum_{i \in I_1} p_i - \sum_{i \in I_2} q_i$$

Because $\sum_{i \in I} p_i = B$, it follows that

$$D(I_1, I_2) = B - \sum_{i \in I_2} p_i - \sum_{i \in I_2} q_i = B - \sum_{i \in I_2} (p_i + q_i)$$
$$= B - \sum_{i \in I_2} a_i$$

Clearly, if (I_1, I_2) is a solution to an LBP instance, i.e. $D(I_1, I_2) = 0$, then $\sum_{i \in I_2} a_i = B$. This implies I_2 is a solution to an SSP instance. For example, if $A = \{3,7,2,5,9\}$, by (5), A is converted into Q_3

$$Q_3 = \begin{bmatrix} 3 & 0 \\ 7 & 0 \\ 2 & 0 \\ 0 & 5 \\ 0 & 9 \end{bmatrix} \quad Q_4 = \begin{bmatrix} 3 & 0 \\ 7 & 0 \\ 1 & 1 \\ 0 & 5 \\ 0 & 9 \end{bmatrix}$$

The partition $I_1 = \{4,5\}$, $I_2 = \{1,2,3\}$ is a LPB solution. Accordingly, $I_2 = \{1,2,3\}$ is the SSP solution. When $B = 11$, A is converted into Q_4, both LBP and SSP have no solutions.

The conversion (5) is formally simple, but, regrettably, some p_i or q_i can become zero after the conversion. To overcome the disadvantage, another conversion to reduce SSP into LBP is defined as follows

$$
\begin{cases}
p_i = \begin{cases} 2ka_i & i = 1,2,\dots,M \\ 1 & i = M+1,\dots,2M \\ kB + kS + M + 1 & i = 2M+1 \end{cases} \\
q_i = \begin{cases} 1 & i = 1,2,\dots,M \\ ka_i & i = M+1,\dots,2M \\ kB & i = 2M+1 \end{cases}
\end{cases}
\quad (6)
$$

Here, $S = \sum_{i=1}^{M} a_i$, $k > M$. Clearly, the conversion is polynomial. Let

$$
\lambda_i = \begin{cases} 1 & if \quad p_i \quad is \quad chosen \\ 0 & otherwise \end{cases} \quad i = 1,2,\dots,M
$$

$$
\gamma_i = \begin{cases} 1 & if \quad q_{i+M} \quad is \quad chosen \\ 0 & otherwise \end{cases} \quad i = 1,2,\dots,M
$$

If $\{\lambda_i\}_{i=1}^{M}$ and $\{\gamma_i\}_{i=1}^{M}$ is the above LBP solution, it implies

$$
\sum_{i=1}^{M} \lambda_i p_i + \sum_{i=1}^{M} (1-\gamma_i) p_{i+1}
$$
$$
= \sum_{i=1}^{M} (1-\lambda_i) q_i + \sum_{i=1}^{M} \gamma_i q_{i+1} + q_{2M+1} \quad (7)
$$

Notice that $p_{2M+1} \gg q_{2M+1}$, therefore, q_{2M+1} must be chosen. In place of p_i and q_i in (7) by (6), we obtain

$$
k \sum_{i=1}^{M} (2\lambda_i - \gamma_i) a_i = kB + \sum_{i=1}^{M} (\gamma_i - \lambda_i) \quad (8)
$$

(8) implies $\sum_{i=1}^{M} (\gamma_i - \lambda_i)$ must be divisible by k, thus

$$
\gamma_i = \lambda_i \quad i = 1,2,\dots,M \quad and \quad \sum_{i=1}^{M} \lambda_i a_i = B
$$

Therefore, SSP is reducible to LBP in polynomial time, which implies (1) is NP-Complete.

3. PROPOSED HEURISTIC ALGORITHM

Hu presented some suggestion on how to design an efficient algorithm when facing a new problem in his book [6], based on his suggestion, we build a heuristic algorithm based on swaps. We focus our attention on the distributed system consisting of two servers and a sink, whose transfer time of M blocks is listed in (4). At first, we find the minimum of both elements in each row, maximize the difference between two column sums, and then by means of swaps, lower the difference until there is no further improvement. The actual steps are described as follows:

Step 1:
Find $\min(p_i, q_i)$ and calculate $S_1 = \sum_{i \in I_1} p_i$ and $S_2 = \sum_{i \in I_2} q_i$

and $D = |S_1 - S_2|$ ($I_1 = \{i : p_i < q_i\}$, $I_2 = I \setminus I_1$).

Step 2:
When $S_1 > S_2$
$S_{1,i} = S_1 - p_i$, $S_{2,i} = S_2 + q_i$ $i \in I_1$,
$E = \min_{i \in I_1} |S_{1,i} - S_{2,i}| = |S_{1,k} - S_{2,k}|$
If $0 < E < D$
$I_1 = I_1 - \{k\}$, $I_2 = I_2 + \{k\}$, $S_1 = S_1 - p_k$,
$S_2 = S_2 + q_k$, continue step 2,
Otherwise
If $E = 0$ $I_1 = I_1 - \{k\}$, $I_2 = I_2 + \{k\}$, go to end
Else go to end
When $S_1 < S_2$
$S_{1,i} = S_1 + p_i$, $S_{2,i} = S_2 - q_i$ $i \in I_2$,
$E = \min_{i \in I_2} |S_{1,i} - S_{2,i}| = |S_{1,k} - S_{2,k}|$
If $0 < E < D$
$I_1 = I_1 + \{k\}$, $I_2 = I_2 - \{k\}$, $S_1 = S_1 + p_k$,
$S_2 = S_2 - q_k$, continue step 2,
Otherwise
If $E = 0$ $I_1 = I_1 + \{k\}$, $I_2 = I_2 - \{k\}$, go to end
Else go to end

4. EXPERIMENTAL RESULTS

First, let us take Q_2 as an example to illustrate proposed heuristic algorithm.
Step 1: $I_1 = \{1,2,3\}$, $I_2 = \{4\}$, $S_1 = 9$, $S_2 = 2$, $D = 7$.
Step 2: $S_{1,i} = \{7,3,8\}$, $S_{2,i} = \{7,9,6\}$, $E = 0$,
$I_1 = \{1,2,3\} - \{1\} = \{2,3\}$, $I_2 = \{4\} + \{1\} = \{1,4\}$,
go to end.

We implement proposed heuristic algorithm in C++ and use a TXT file to record and trace swaps, and two column

sums S_1 and S_2. The chosen element is marked by a * in the TXT file. Many experiments have been done based on random transfer time matrices generated by a seed, and have been used to validate that the proposed heuristic algorithm is a very efficient one. For fixed M, swap times depend upon the row minimum distribution. When the majority of the row minimum is located at the 1st or 2nd column, more swaps are needed. We randomly select an instance to illustrate the trace process (column 1-3 in Table 1), where, $M = 10$ & seed = 20. After two times swap, an approximate solution is obtained, but there exist a slight difference between the approximate solution (Column 3) and the exact solution (Column 4). We consider two extreme situations: retrieval time of two servers for each block is close and different, the experimental results are listed Table 2.

Table 1: The trace of swap and both column sum S_1, S_2

2*	14	2*	14	2*	14	2*	14
8	8*	8	8*	8	8*	8	8*
1*	14	1*	14	1*	14	1*	14
14	10*	14	10*	14	10*	14*	10
15	10*	15*	10	15*	10	15	10*
4*	17	4*	17	4*	17	4*	17
7	5*	7	5*	7*	5	7*	5
1*	12	1*	12	1*	12	1*	12
11	10*	11	10*	11	10*	11	10*
5	1*	5	1*	5	1*	5	1*
$S_1 = 8$		$S_1 = 23$		$S_1 = 30$		$S_1 = 29$	
$S_2 = 44$		$S_2 = 34$		$S_2 = 29$		$S_2 = 29$	

Table 2: Retrieval time of two servers for each block is close and different

1-5 blocks		6-10 blocks		1-5 blocks		6-10 blocks	
11*	12	16*	16	12	2*	2*	18
14	14*	5*	6	16	4*	11	1*
5*	6	2	4*	5*	18	15	1*
15	15*	12	12*	12	1*	4*	17
3	2*	10*	8	2*	18	13	3*
$S_1 = 47$ $S_2 = 47$				$S_1 = 13$ $S_2 = 12$			

Table 3: Multi-dimensional case

1-4 blocks				5-8 blocks			
19	17	10*	18	11	9	19	13*
7	11	12	6*	19	8	17	4*
4	4*	10	10	3*	11	13	8
4*	17	18	10	8*	14	18	16
9-12 blocks				13-16 blocks			
14	13	4	2*	19	10*	12	12
5*	11	18	8	17	8*	9	14
12	8	11	8*	11	6*	9	15
6*	18	19	10	2*	19	8	7
17-20 blocks							
1*	9	7	17				
1	15	19*	11				
9	2*	19	18				
19	14	4*	4				

The above-mentioned swap algorithm can be easily extended into a distributed system consisting of multiple servers and a sink. In the multi-dimensional case, swap is always done between two columns whose cumulative sums are maximum and minimum, each column cumulative sum has to be recalculated after each swap, this procedure continues until there is no further improvement. Here we randomly select an instance, $M = 20$, $N = 4$ & seed=20. Its solution is listed in Table 3, and cumulative transferring time allocated to four servers are respectively $S_1 = 29$, $S_2 = 30$, $S_1 = 33$, $S_2 = 33$.

5. CONCLUSION

In this work we modeled the underlying problem for block-based distributed retrieval of a wavelet image from multiple servers. We proved that the general problem is NP-complete even for a simplified scenario and proposed a heuristic algorithm. Experimental results show that the heuristic solution is quite close to the exact solution for a limited number of blocks and servers.

REFERENCES

[1] C. Xu and C. M. Lau, Load Balancing in Parallel Computers Theory and Practice, Kluwer Academic Publishers, 1997.

[2] B. Veeravalli, G. Debasish, M. Venkataraman, and T. M. Robertazzi, Scheduling Divisible Loads in parallel and Distributed System, IEEE Computer Society Press, Los Alamitos, California,1996.

[3] A. Y. Zomaya and Yee-Hwei Teh, "Observations on Using Genetic Algorithms for Dynamic Load-Balancing", IEEE Trans. On Parallel and Distributed Systems," vol.12, no.9, pp. 899-911, Sep. 2001.

[4] T. Thanalapati and S. Dandamudi, "An Efficinet Adaptive Scheduling Scheme for Distributed Memory Multicomputers", IEEE Trans. On Parallel and Distributed Systems," vol.12, no.7, pp. 858-768, Jul. 2001.

[5] M. R. Garey and D.S.Johnson, Computer and intractability, A guide to the theory of NP-Completeness, San Francisco: Freeman 1979, page 223

[6] T. C. Hu, Combinatorial Optimization, Addison-Wesley, 1982.

Library architecture for searching software components by their algorithmic features

Yutaka Watanobe, Rentaro Yoshioka, and Nikolay N. Mirenkov
Graduate School Department of Information Systems
University of Aizu, Aizu-Wakamatsu, 965-8580, Japan
E-mail:{d8061103, rentaro, nikmir}@u-aizu.ac.jp

Abstract

A library architecture suitable for searching software components by their algorithmic features is presented. One of the main goals of the library is to provide an efficient search mechanism for a large number of reusable software components. To achieve a higher degree of reusability in comparison with conventional libraries, a library in which each component is stored as a cyberFilm [11] is considered. A cyberFilm is a component format that can represent software as a set of algorithmic features, such as structures, schemes, formulas, and I/O operations. These algorithmic features provide a common, high-level and yet precise description of computation. The library takes advantage of these features in the searching operations. A set of software modules responsible for manipulations with corresponding features constitute the library architecture. In this paper, an overview of the attributes derived from different features to classify and access cyberFilms is presented. Applicability of the attributes to support searching interfaces is also described.

1 Introduction

In spite of the many difficulties of programming, a greater population of people is required to be involved in the information technology and development of information resources for supporting the future growth of our society. Software component libraries have become an important part of software creation. It is extremely rare that we create programs completely from scratch. In the simplest form, we use code fragments created by others as includes and imports, or simply make calls to standard libraries of the programming languages. In a more sophisticated form, we program within the framework of some component-based technologies, such as Microsoft's COM, JavaBeans, OSGI, etc. [4]. These frameworks help in reducing the developer's load related to the tasks of accessing, composing, and executing components. In either case, building new solutions by combining existing components improves quality and supports rapid development. However, it is very difficult for users with limited knowledge to understand and reuse these libraries. Such libraries include a large variety of algorithms. For example, Java class libraries and Standard Template Library [7] provide a wide range of functions from basic data structures and GUI gadgets to high performance algorithms. Library based programming tools, such as LEDA [8] also provides a lot of useful algorithms. Currently, it is up to the user to find an appropriate class or component and discover how to use them. The most common approach in performing this search is reading text and diagrams of thick documents. It is possible to say that we are still facing the productivity problems in programming [5-6]. One possible method for the user to acquire the right components corresponding to his/her needs is retrievals based on algorithmic features. In many case, the users know what to do, but do not know how to do. It means, the library architecture should support problem solutions through extracting or predicting algorithmic features from the problem description.

In this paper, a library architecture for searching software components by their algorithmic features, is presented. It is based on the self-explanatory components approach [9-10]. In this approach, a component is created in a "cyberFilm" format. One notable characteristic of the component is that it can represent software as a set of algorithmic features. The set of algorithmic features includes at least three groups: multimedia algorithmic skeletons (AS), variables and arithmetic/logic expressions, and input/output operations. The library architecture includes a set of modules responsible for manipulations with corresponding features. A feature can be considered as a set of attributes, and we present two main sets of attributes important for the searching operations. The first group is attributes mainly to identify each component and to summarize its algorithmic features. The second group is attributes to describe how the

component is related to other components and how it has been used by other users. In other words, the idea is to describe the component from internal and external perspectives to allow a search from multiple points of views, thus increasing the possibility of finding a suitable component from a large library for various users. Most of these attributes are automatically discovered/generated by supporting modules of the library when the user registers/accesses the cyberFilm. Applicability of the attributes to support searching interfaces is also described.

2 Related works

In addition to papers from the introduction, we would like to mention some other works related to software component search. SPARS-J (Java Software Product Archiving, analyzing and Retrieving System) [2] is a component search system where source codes of classes or interfaces are parsed and ranking methods based on frequency of words to discover relations between various components are used. Agora [3] is also a component search system that shows a useful integration of Web search engines and component introspection. However, these systems work only with the source code format. They provide effective searching of components such as Java classes and JavaBeans. As a rule, many object-oriented language researchers focus on fixing Java's trouble spots or extending the language itself [1]. However, it is difficult to extract additional high-level descriptive information, such as algorithmic features, from them.

In our framework, algorithmic cyberFilm components are acquired in a library as pieces of "active" knowledge. In most cases, the user should not create cyberFilms from scratch. He/she can retrieve a suitable cyberFilm from the library. Users can embed their algorithmic ideas into the skeletons of existing cyberFilms to create new cyberFilms by relatively simple operations. The modified cyberFilms are stored into the library again, and will be used for generating new cyberFilms. It is worth mentioning that the same computational scheme can be used to represent many specific algorithms by attaching different sets of formulas, variables, and input/output operations to the skeletons. In other words, such cyberFilms can be used easily to create a variety of new cyberFilms (algorithms) by substituting the existing sets with new sets.

3 Internal Attributes

The architecture supports mainly two groups of attributes in the cyberFilm for searching software components. The first group is a set of internal attributes to uniquely identify and summarize the algorithmic features of a component. The second group is attributes to describe how the component is related to other components and how it has been used by other users. Each attribute takes a part in the creation of a classification code. The classification code is a coded tuple of attributes of a corresponding cyberFilm component. These attributes are used for the searching operations. In this paper, we do not mention how the codes are created, but we focus on the extraction of the attributes from the algorithmic features of a cyberFilm component and on the types of these attributes.

The first column of Table 1 depicts the current list of internal attributes. The attributes are defined by the user through special editors to specify algorithmic features. These attributes are used to produce different sections of the classification code. In this section, details of these internal attributes and how they are implicitly defined by the user in a language of micro-icons, are described.

Table 1. Table for Internal Attributes

Internal Attributes
Structures
Schemes
Masks
Static irregularities
Type of compositions
Mathematical notations
Variables
Stencils
Input/Output structures
Input/Output communication types
Complexity based on AS
Complexity based on full features
Complexity defined by authors
Title
Author
Date
ID

- **Structure.** A structure or a combination of structures is one of the most interesting and the most likely to be required attribute for searching computational algorithms as well as algorithmic skeletons. The structure is a space shape of computation, and determines a model of corresponding applications and problems. Figure 1 (a) depicts examples of micro-icons to specify basic structures such as 2-D grids, 3-D grids, trees, graphs, particles in 3D space, etc. In many cases, these basic icons will be used to specify structures, but also additional icons that represent more specific features of a structure such as additional properties or different layouts, can be used for a variety of applications and user's intentions. Figure 1 (b) shows examples of additional micro-icons to specify constrained structures for more special types of computation, such as circular trees, cone trees, bipartite graphs, complete graphs, and balanced trees.

Figure 1. Examples of micro-icons for structures

- **Mask.** A mask is specifies static irregularity on regular structures. In other words, a mask modifies the space structures so that the user can characterize computational shapes in more detail. Figure 2 represents examples of micro-icons for the masks related to 2-D grids and 3-D grids. Both simple and complex geometrical configurations and patterns of irregularity can be represented.

Figure 2. Examples of micro-icons for masks

- **Scheme.** A scheme defines a partial order of node scanning on the corresponding structures. Figure 3 shows examples of micro-icons for possible computational schemes. This attribute also allows to specify sequential or parallel processing, directional characteristics and patterns of computational flows.

Figure 3. Examples of micro-icons for computational schemes

- **Static irregularity.** To add additional levels of irregularity to the computational schemes, a set of masks can be used. In fact, these masks are applied not to space structures but to computational schemes on the structures. A mask can inhibit a set of nodes from execut-

ing formulas regardless of the computational scheme. For example, the leftmost micro-icon of Figure 2 defines computation only on external nodes of a 2-D grid structure.

- **Type of composition and levels of hierarchy.** For composite cyberFilms, a great variety of spatial (for example, shown in Figure 1), temporal, and hierarchical structures, and a combination of spatial and hierarchical structures, are supported. Also, in hierarchical compositions, **levels of the hierarchy** can be important attributes to characterize corresponding components. In temporal compositions, special structures for iterative and branching constructs are supported as shown in Figure 4.

Figure 4. Examples of Temporal Composition

- **Mathematical notations/functions.** One of essential points of this library is that components can be searched by kinds of mathematical notations/formulas used. Figure 5 shows examples of micro-icons for the mathematical notations/functions.

Figure 5. Examples of micro-icons for mathematical notations/functions

- **Variable.** This attribute includes a list of (structure, data type) pairs for all variables declared in the component.

- **Stencil.** A stencil is used as a high-level construct for specifying indexes in formulas [9]. Figure 6 shows examples of micro-icons for stencils.

Figure 6. Examples of micro-icons for stencil

- **Input/Output structures and their data types.** Input/output structures include 1-D/2-D/3-D grids, trees, pyramid, and other structures. This attribute is a set of pairs (a structure and a data type such as integer, double precision integer, float, double precision float, character, string, etc.). Kinds of I/O structures and data types occurred before, after, and during the process of computation, can be specified.

- **Complexity based on algorithmic skeletons.** Complexity of algorithmic skeletons can be estimated from a combination of schemes and structure sizes. This complexity is based on the number of computation steps represented by the algorithmic skeleton and does not consider complexity of operations performed in the structure nodes.

- **Complexity based on full features.** It is like in previous case with additional taking into account complexity of operations in the structure nodes.

- **Complexity defined by component designers.** Some times, it is difficult or unsafe to derive exact complexities of corresponding components automatically. As an attribute, the author can evaluate the complexity by himself.

- **CyberFilm title.** A title is a conventional name which can be used as a keyword for searching operations.

- **Author.** This attribute includes a personal name, as well as a name of company, university, or research organization. For example, a search based on this attribute can provide a list of components registered by a particular group of people or organizations.

- **Date of creation.** Currently, date is automatically set to the registration data. Also, this date implies latest modification date.

- **ID.** When a component is registered, a URL-like address is attached to the component. In this address, the component is available in the cyberFilm format. It means that a direct observation of the component feature can be used for searching operations.

Above mentioned attributes are recorded for each component. On assumption, each cyberFilm can be applied to various application fields. It will be interesting and useful to search components by application fields as well. However, depending on creators and users, definitions of such attributes may be different. People in different fields have different knowledge and use different terminology. So, clusters of cyberFilms depending on people's knowledge or professional orientation should also be prepared in the library.

4 External Attributes

In this section, a set of external attributes about how a component is related to other components and how it has been used by other users, is considered. Relations between components and statistical attributes s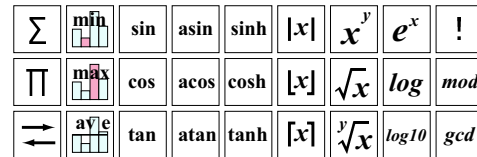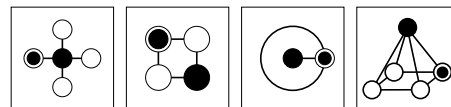uch as frequency of usage, can be attractive search keys. After the registration or specification, any activity with a cyberFilm component in the library will be recorded and used to update the statistics and links between the cyberFilm components. Table 2 shows the current set of external attributes. In this section details of the attributes are considered.

Table 2. Table for External Attributes

External Attributes
Latest referred date
Number of usage
Activity rate
Number of updates
Parent
List of children
Number of inheritance
List of source links
List of target links
Frequency of usage based on user's activity

- **Latest referred components.** A set of components that have been used recently (for example, this month) should be supported. This value can be used to derive activity rates as below.

- **Number of usage.** This value can comprise famousness of corresponding components. Also, this value can be used to derive the activity rates.

- **Activity rate.** This value computed from the latest referred components and the number of usage. This attribute is important not only for searching, but also for library arrangement. For example, passive knowledge such as components that are used seldom for a long time, will be removed to a special section of the library by the system automatically. Also, in some search interfaces, multiple search results are ordered based on the activity rate. That is to say, the most active component will appear first in a list of candidates.

- **Number of updates.** This value is incremented every time when the component is modified. In some sense, it is like a version number. Definitely, a variety of components improves the library quality.

- **Parent and list of children.** As mentioned previously, a library component can be derived from existing cyberFilms by editing/composing operations. A parent as well as a list of children provide genetic (historical)

information of the components. These attributes enable search based on inheritance relationship between components.

- **List of links.** The links associate composite cyberFilms. There are two types of links: for target and for source components.

Here, it is important to mention that a cyberFilm has an attribute to record the frequency of usage. It would be useful to know which components were frequently used by users from a certain field or profession. The library maintains tables to provide mapping between component IDs and frequencies for each different groups.

5 Applicability of the attributes

In this section, applicability of the attributes presented in previous sections is considered. These attributes are used by special modules of the library to perform searching operations. We do not mention about concrete search algorithms and their efficiency, but we focus on kinds of search methods applicable with the attributes through a special set of library interface panels.

Figure 7 shows an example of an interface for searching by a narrow down technique. Users can specify preferred values for several attributes to find matching components. The specification of the attributes is based on icon selection or if necessary on keyboard input of texts or numeric values. Users can select kinds and numbers of attributes. Figure 7 shows that our focus is on searching with attributes related to algorithmic skeletons and input/output formats.

Figure 7. An interface for search on queries

Figure 8 shows an example of an interface for search by specifying possible ranges of attributes values. This interface shows components that match several attributes simultaneously in a two or three dimensional surface. It has vertical and horizontal groups of axes to specify value ranges of several attributes. The number of axes and corresponding attributes are changeable. Operations such as "union," "intersection," "difference," and "complement" of these attributes are supported. Selected components are presented by colored squares according to their features.

Figure 8. An interface for search on values of attributes

Figure 9 shows an example of an interface for searching on the interrelationships between components. These searching operations are based on traversing library components connected by links on a three dimensional spherical surface. Such information visualization technique is presented in [12].

A main point of this searching method is considering different kinds of interrelationships. Some of them are related to the embedded inheritance, others are created as a result of the component and library use. Basically, a set of links implies relations between components, so they are changeable according to users purposes. One of useful relationship is similarity. The proximity of any two components are derived by how algorithmically (based on the attributes) similar they are. As a rule, lengths of the links depict distances between components. Consequently, components that have similar attributes are gathered, while less similar components will appear distantly from one another. The distances between two components and their locations are computed based on the classification codes. The inheritance relationship uses the list of parents and children. Also, links to other components can be considered. A cyberFilm can include other cyberFilms. This link corresponds to such a relationship. The user can search application cyberFilms that refer

Figure 9. An interface for search on interrelationships

to a selected cyberFilm.

These searching operations enable the users not only to search target components, but also to discover new knowledge and algorithms for them. The user can start searching from any component and then traverse in a direction based on his/her intentions. Also, the interface will support searching on multiple attributes. In such case, the attributes are distinguished by using different colors of links, and intensities of relevance ratios are represented by thickness of links.

6 Conclusion

A library architecture that is suitable for searching software components by their algorithmic features has been presented. Contents of the library are software components presented in the cyberFilm format. A cyberFilm represents software by a set of algorithmic features. Two groups of attributes have been introduced to efficiently and effectively search these components. The first group is a set of internal attributes to uniquely identify the component and to summarize its algorithmic features. The second group is a set of external attributes to describe how the component is related to other components and how it has been used by other users. The architecture is based on modules responsible for extracting component features and transforming them into a set of attributes to produce a special classification code. These attributes and code are used to implement efficient searching operations. To specify the operations, a special set of interface panels has been introduced.

References

[1] A. P. Black, Post-Javaism, IEEE Internet Computing, Vol.8, No.1, 2004, 96, 93–95.

[2] T. Yamamoto, Overview of Component Search System SPARS-J, International Workshop on Community-Driven Evolution of Knowledge Artifacts, University of California at Irvine, 2003.

[3] R. C. Seacord, S. A. Hissam, and K. C. Wallnau, Agora: A Search Engine for Software Components, IEEE Internet Computing, Vol.2, No.6, 1998, 62–70.

[4] G. T. Heineman, and W. T. Council, Component-based Software Engineering: putting the pieces together, Addision-Wesley, NHH(2001). New York, 1998.

[5] F. P. Brooks, No Silver Bullet: Essence and Accidents of Software Engineering, *Computer*, Vol.20, No.4, 1987, 10–19.

[6] S. Hallem, D. Park, and D. Engler, Uprooting Software Defects at the Source, Queue, Vol.1, No.8, 2004, 65–71.

[7] Nicolai M. Josuttis The C++ Standard Library A Tutorial and Reference, Addision-Wesley, 2001.

[8] LEDA, http://www.algorithmic-solutions.com/

[9] R. Yoshioka, N. Mirenkov, Visual Computing within Environment of Self-explanatory Components, Soft Computing Journal, SpringerVerlag, Vol.7, No.1, 2002, 20–32.

[10] N. Mirenkov, A. Vazhenin, R. Yoshioka, T. Ebihara, T. Hirotomi, T. Mirenkova, Self-explanatory components: a new programming paradigm. International Journal of Software Engineering and Knowledge Engineering, world scientific, Vol.11, No.1, 2000, 5–36.

[11] R. Yoshioka, N. Mirenkov, Y. Tsuchida, Y. Watanobe, Visual Notation of Film Language System, Proceedings of 2002 International Conference of Distributed Multimedia Systems, San Francisco, California, USA, 2002, 648–655.

[12] S. K. Card, J. D. Mackinlay, and B. Shneiderman, Information Visualization, Morgan Kaufmann Publishers, 1999.

Video Editing with Change of Camera Motion

Masahito Hirakawa and Kasumi Nakashita
Interdisciplinary Faculty of Science and Engineering, Shimane University
1060 Nishikawatsu, Matsue 690-8504, Japan
hirakawa@cis.shimane-u.ac.jp

Abstract

Videos are active and rich media which are enjoyable for the people. Recent advancement of digital technologies, specifically the development of consumer digital video camcorders and mobile phones with camera, makes us possible to take videos with ease. This then requests us to have a facility of editing videos which have been taken, so as to fit to their preference. While many video editing software are available, it is allowed in most cases just to cut a video into segments and arrange them along a timeline.

In this paper we propose a new editing scheme allowing the user to specify desirable camera motions, specifically pan, tilt, and zoom, to a video shot after the shot has been taken. Image mosaicing which is a well-known technique in computer vision and computer graphics is used as a means of interacting with the user, by knowing the fact that the outer shape of a mosaic image conveys the information of camera motions included in the shot.

1. Introduction

Videos are active and rich media that people feel fun and enjoyable. Development of consumer digital equipments, e.g., video camcorders and mobile phones with camera (camera phones), allows us to take videos casually. Those equipments are small enough to bring, and thus taking a video is becoming a common activity for us in our daily life. Furthermore it is reported in [1] that worldwide unit sales of camera phones will reach nearly 150 million in 2004. Camera phone sales are expected to experience a compound annual growth rate, CAGR, of 55% to reach 656 million units in 2008.

Here those who take such videos are mostly non-professionals – people who have less knowledge of taking good videos. They just press a recording button and take a video without any scenario in advance. They leave what they take in a video to chance. As the result, video materials are not good enough in their image quality and structure to watch repeatedly after they have been taken. Video editing is becoming more and more important and crucial for the success in video utilization [2], [3].

There are various commercial products, such as Adobe Premiere and Apple Final Cut, which help the user to edit video materials he/she has recorded. Here, roughly speaking, facilities provided in such software are to cut a video into segments and then arrange those video segments with certain transitions and titles. In other words, editing of structural aspects (i.e., the order of video segments) is possible, which is called intra-segment editing. Yet other powerful tools/facilities allowing the user to edit the contents of each video segment are requested. We call this type of video editing inter-segment editing. Researches on inter-segment editing have been active in recent years [4]-[6]. However, still further studies are needed.

This paper presents a new idea of inter-segment editing. The user can edit a video shot in a scene by specifying desirable camera motions - specifically, pan, tilt, and zoom - as a postproduction condition. For example, while an original shot is taken without any camera motion, it is possible to generate a shot featuring a continuous pan with an additional zoom-in in a certain part of the shot in order to call viewer's attention to a target zoom-in object.

Image mosaicing [7] which is a well-known technique in computer vision and computer graphics for creation of a comprehensive overview of a video shot content is used as a means of interacting with the user. The video content is visualized in a form of mosaic image. The user specifies his/her preference in video editing over the given mosaic image. It is noted here that the specification of camera motion is also given in a mosaic form. This uniformity improves the user-friendliness of the system.

The rest of the paper is organized as follows: Chapter 2 describes background and motivation of this research. Chapter 3 explains a user-interface of the proposed system. A method of generating a final video shot is explained in Chapter 4. Finally, in Chapter 5, we conclude the paper.

2. Research Background and Motivation

2.1 Non-linear video editing

Throughout the 1990s many manufactures introduced non-linear video editing systems, such as Adobe Premiere and Apple Final Cut. While traditional videotape-based editors are linear in the sense that they require editing a video sequentially from beginning to end, video edits in non-linear video editing environments can be done at any order and at any point in the timeline. This random accessibility of non-linear editing increases the usability and performance in complex editing tasks.

Furthermore, video editing systems provide a variety of video effects. For example, the user uses color correction tools to adjust hue, saturation, and brightness throughout a video shot. And the camera view command in Premiere, for example, gives the impression of a camera looking at your shot from different angles - rotate, flip, and zoom. It should be noted, however, that the shot itself doesn't change. The video image (or shape) is transformed something like a flying plane in a 3D space.

In addition to those commercial-based video editing tools, researchers have been interested in developing new and powerful editing tools. For example, Hitchcock system in [4] identifies the type and amount of camera motion. It uses the data to identify shots for inclusion in the final video and to select their start and end points. [5] presented video editing operations that support complex postproduction modifications, including changes in camera motion, with the idea of spatio-temporal volumes. However, user interaction with the system for editing is not a main concern.

2.2 Mosaicing

Mosaicing [7] is an already established technique in many disciplines and has been applied to various domains. Video browsing and retrieval are examples [8], [9], where the capability of mosaic image – the content is understandable "at a glance" – helps the user to specify his/her query.

A mosaic image is created by piling up frames of a given shot. In this process, the camera motion is first estimated, and successive frames are then overlapped. Figure 1 shows an example mosaic image. For reference, the first, middle, and last frames of the shot are presented as well.

Here, an interesting feature of mosaicing in our context is that a different combination of pan, tilt, and zoom results in a different shape of the mosaic image. The degree of panning and tilting is observable through the size of horizontal and vertical blank space (i.e., black region in the mosaic image), respectively, as illustrated in Fig. 2. The scale of video frames corresponds to the degree of zooming. Smaller the video frame, greater the zoom-in factor.

Figure 1   An example of mosaic image

Figure 2   Mosaic image with pan, tilt, and zoom operations

Thanks to this characteristic, editing of a video with the change of camera motion becomes possible. For example, while a source video contains only a pan operation, the user can make a different video featuring the tilt and zoom.

Yan and Kankanhalli proposed the idea of using mosaicing for video editing [6]. However, in their trial, mosaicing is adopted for detecting video segments towards detection and removal of lighting and shaking artifacts.

3. User Interface for Video Editing

As explained, a video is formed into a 2D mosaic (panoramic) image in advance. Specification of editing in camera motion is given by placing a desirable mosaic

shape, i.e., outer frame of mosaic image, onto the original mosaic image. This uniformity reduces the semantic gap and helps the user to carry out the task.

Figure 3 shows an editing interface of the proposed system, consisting of a command menu and a main editing window. When a source video is specified by the user, the corresponding mosaic image is generated by the system and presented in the main editing window, as in the figure.

Figure 3   Example of a visual sentence specifying the edited camera motion

A camera frame denoted by a rectangle box in Fig.3 is a primitive component used to specify a preferable camera work, and is defined by dragging a mouse on the mosaic image. Semantically it specifies static parameters on the position of camera center and the zoom factor. Dynamic features, that is, the movement of a video camera, are specified in terms of a sequence of camera frames, forming a visual sentence, as in the figure.

More specifically, the position of a camera frame within the editing window specifies which portion of a source video scene is framed. Its size is inversely proportional to the zoom factor. That is, a small camera frame means that the camera is focusing on details of the source video scene, and vice versa. Lines connecting the camera frames provide a visual feedback on the movement of video camera.

For example, the visual sentence in Fig. 3 features three camera frames. It describes a combination of pan (to the right), tilt (to the above), and zoom-in (between the second and the third camera frames). A visual clue on the outer frame of the visual sentence, or target mosaic image, gives a fruitful feedback to the user.

Here, in specifying the visual sentence as a combination of separate camera frames, those camera frames denote the timing where meaningful transitions in at least one of the camera parameters occur. This idea has been borrowed from graphics animation, where keyframes are used in a similar manner.

Meanwhile, since the final outcome is a video, the system has to interpolate frames between camera frames. Here we suppose that parameters between the frames vary linearly. This assumption may, however, not be preferable in some cases. More powerful capability of changing motion parameters is requested. Let us here consider a facility of editing the speed of camera motion.

Since the camera motion is a temporal feature, the most conventional approach would thus be of using a timeline. Images corresponding to the camera frames are arranged as keyframes on a timeline. Change of the camera motion speed can be specified by dragging a target camera frame image. If a camera frame image is moved closer to its adjacent camera frame image, it is interpreted as speed-up, and vice versa.

Meanwhile, we propose using a spiral-based visualization scheme that one of the authors presented in [10], as the alternative to the linear timeline-based approach. The spiral-based visualization was proposed originally for identifying periodical patterns of temporal events on a timeline.

Though both 2D and 3D spiral representations are presented in [10], we adopt the 3D form of spiral in this trial. It is noted here that the form is not just like a coil formed by a wire around a uniform tube. Additional interactive function is provided, at which the keyframes are aligned, resulting in a non-uniform spiral as appeared in Fig. 4. Square green boxes in the figure correspond to keyframes. A longer diameter shows a longer period between the associated keyframes.

Figure 4   Spiral-based speed control

Adjustment of the camera speed is carried out by making the diameter longer or shorter depending on the user's preference. He/she can easily recognize the temporal pattern (or periodicity) of the frames by looking at the shape of spiral. Also the user may switch their representation from symbol/box to thumbnail, and vice versa.

4. Generation of Resultant Video

Next we explain a method of generating a resultant video along the given visual sentence being composed of multiple camera frames.

As explained in 2.2, a mosaic image is created by piling up video frames, where successive frames are overlapped with making an adjustment of their position with respect for the camera motion (see Fig. 5). Camera frames forming a visual sentence, or a target mosaic image, are specified on the source mosaic image through the interface explained in the previous chapter. The system first identifies frames in the source video, which are associated with the visual sentence. And then a specific portion of each frame is extracted as a frame to be included in the resultant video.

Figure 5   Video frames associated with the target mosaic image

Here the portion to be extracted may belong to multiple frames. In other words, there are multiple choices in selecting a frame for the portion, as illustrated in Fig. 6. We have to choose one of them in advance of the further processing.

Figure 6   Multiple choices in selecting a frame

Our interest goes to image quality – the number of pixels in a target image portion. In the case of pan and tilt, as you can imagine, all frames have an equal image quality while the scene in those frames may be different. Therefore we can choose any one of the possible frames. In the case of zoom, however, image quality with respect to the target image portion differs frame by frame. We choose the one which includes the whole portion of the target image and takes the greatest zoom-in parameter. In the case of Fig. 6, χ is the frame to be chosen.

Finally, all portions which correspond to the target mosaic image are obtained. Here, as illustrated in Fig. 7, those portions have different image size depending on the zoom factor specified in the visual sentence. In order to generate a complete video file, they are expanded or reduced in their image size, resulting in a sequence of frames whose size is the same.

Figure 7   Resize of frame portions

source video

resultant video

Figure 8   Source video and resultant video to the visual sentence specification in Fig. 3

Figure 8 shows some snapshots of the source and resultant videos for the given specification of visual sentence in Fig. 3.

5. Conclusion

In this paper, we presented a new video editing tool which enables the user to edit a video by giving desirable camera motions which include pan, tilt, and zoom. The resultant video can thus take different camera motions from the original/source video. The idea of image mosaicing has been applied for both the representation of a source video and the specification of a user's requirement. This unified interface is helpful to the user in manipulating the system. We have developed a prototype system and the experiments provide compelling evidence for the efficacy of the proposed method.

Much more still remain to be investigated. One serious problem is that, in case of using a zoom-in operation, a part of the video frame is extracted and expanded. This degrades the image quality of the resultant video frame. Further study on image processing is necessary.

References

[1]   http://www.infotrends-rgi.com/home/Press/itPress/ 2004/3.11.04.html

[2]   M. Davis, "Editing out Video Editing," IEEE MultiMedia, Vol.10, No.2, pp.54-64, Apr./June 2003.

[3]   J. Casares, A. C. Long, B. A. Myers, R. Bhatnagar, S. M. Stevens, L. Dabbish, D. Yocum, and A. Corbett, "Simplifying Video Editing Using Metadata," Proc., ACM DIS Conf., pp.157-166, 2002.

[4]   A. Girgensohn, J. Boreczky, P. Chiu, J. Doherty, J. Foote, G. Golovchinsky, S. Uchihashi, and L. Wilcox, "A Semi-automatic Approach to Home Video Editing," Proc., ACM UIST, pp.81-89, 2000.

[5]   E. P. Bennett and L. McMillan, "Proscenium: A Framework for Spatio-Temporal Video Editing," Proc., ACM Multimedia, pp.177-184, 2003.

[6]   W. Q. Yan and M. S. Kankanhalli, "Detection and Removal of Lighting & Shaking Artifacts in Home Video," Proc., ACM Multimedia, pp.107-116, 2002.

[7]   M. Irani, P. Anandan, J. Bergen, R. Kumar, and S. Hsu, "Mosaic Representations of Video Sequences and Their Applications", Signal Processing: Image Communication, special issue on Image and Video Semantics: Processing, Analysis, and Application, Vol.8, No.4, May 1996.

[8]   J. Assfalg, A. Del Bimbo, and M. Hirakawa, "Mosaic-based Query Language for Video Databases," Proc., IEEE Symposium on Visual Languages, pp.31-38, 2000.

[9]   M. Hirakawa, K. Uchida, and A. Yoshitaka, "Content-based Video Retrieval using Mosaic Images," Proc. International Symposium on Cyber Worlds: Theories and Practices, pp.161-167, 2002.

[10] K. P. Hewagamage and M. Hirakawa, "An Interactive Visual Language for Spatiotemporal Patterns," Journal of Visual Language and Computing, Vol.12, pp.325-349, 2001.

A New Post-processing Method for VQ Compressed Image

Ju-Yuan Hsiao[1], Kai-Jung Shih[2] and A-Pei Tsai[3]

[1] Department of Computer Science and Information Engineering
National Changhua University of Education, Changhua 500, Taiwan R.O.C.
E-mail: hsiaojy@cc.ncue.edu.tw

[2] Department of Computer Science and Information Engineering
National Chung Cheng University, Chia-Yi 621, Taiwan, R.O.C.
E-mail: kjshih@cs.ccu.edu.tw

[3] Department of Information Management
National Changhua University of Education, Changhua 500, Taiwan R.O.C.
E-mail: beta@mail2000.com.tw

Abstract

Vector quantization (VQ) is an efficiency and importance technique for digital image compression. In order to reduce its data size, we would like to encode the VQ compressed image again. To avoid degrading the quality of reconstructed image, we employ lossless technique to keep the quality. This method is combining the concept of state codebook, lossless encoding method and statistic table of index occurs. A comparison of bit-rate is made between our method, SOC and LAS. We use several data sets to demonstrate that our results are better than other methods.

Keywords: Vector quantization, variable length coding, lossless encode

1. Introduction

The industry of digital image processing is currently an area of rapid growth. These technologies originated from two principal application areas: improvement of pictorial information for human interpretation; and processing of image data for storage, transmission, and representation for autonomous machine perception. Since the image size maybe large, we need some compression techniques for image data storage and transmission. The image compression techniques aim at reducing the data rate to within the network bandwidth capacity or optimize storage. These techniques were based on the feature of images, such as its two-dimension data characteristic and (a little) difference toleration by human eyes.

There are many compression methods and storage formats for digital images. These methods can be classified as two kinds for image compression mainly: lossless[6] and lossy compression. Lossless compression methods are guaranteed that the restored image is numerically the same as the original one but restricted to lower compression rates and less efficiently. Lossy compression methods, on the contrary, can compress image within very high compression rate and keep the reconstruction quality in the acceptable scope. Therefore, lossy compression methods are used in many fields.

Since lossy compression methods cause some degradation in the reconstructed image. In order to provide a quantitative measure of the effect, the PSNR (Peak Signal to Noise Rations) value is calculated and displayed. For an 8 bits gray image, the measure of PSNR is follow:

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \quad dB \tag{1}$$

$$MSE = \frac{1}{N \times N}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}[X(i,j) - X_d(i,j)]^2 \tag{2}$$

Where $N \times N$ is the image size, $X(i,j)$ is the pixel value in original image, and $X_d(i,j)$ is the pixel value after decompressed. In general, a lower MSE or a higher PSNR means the reconstructed image quality is better. But it is not absolutely. Sometimes a little part changes may cause an obviously influence in human visual. When we evaluate the quality of reconstructed image, we can use both the objective and subjective aspects at the same time.

Within the lossy compression methods, we can also classify them in two ways: transform method and vector

quantization (VQ) [1][2][3]. Transform methods are based on image data transformation, quantization and coding. The JEPG image format is the most popular lossy transform method. It used the Discrete Cosine Transform (DCT) method to prune the insignificant part and keep important information for compression. VQ is a direct and efficient method which skipping the part of transformation and quantization. The image is divided into small and non-overlapping blocks which size is $m \times m$. Each block is regarded as a vector contains $m \times m$ dimensions. In the encode phase, each vector x_v is computed the Euclidean distance with the codewords (or codevectors) of codebook. And using the index of the nearest distance codeword represents this block. These indices are composed an index table. The table is indicated as the compressed image. In the decode phase, the decoder reconstructs the image just using the index to perform table look-up to get the codeword from the codebook.

Nevertheless, the original VQ methods takes too much time in encoding phase since we need to search codeword one by one and compare their Euclidean distance. Many researches have focus on improving the VQ encoding efficiency. One is to enhance the VQ encoding efficiency, such as Principal Component Analysis (PCA) or Euclidean distance table look-up, to improve the encoding speed. The other way is to improve VQ compression rate by reducing the VQ index table, i.e. codebook, [7][10][11][12]. These methods, such as Search-Order Coding (SOC) [8] and Locally Adaptive Scheme (LAS) [9], are usually exploiting the relation between neighborhoods within a codebook to save more storage size. Since the codebook is the major part of VQ method, so these methods in the second encode phase using lossless compression for keeping the quality of reconstruct image.

In this paper, we propose a novel method to enhance the VQ compression rate by reduce the VQ codebook size. It combines the concept of state codebook of Full Search VQ (FSVQ, will be discuss in Section 2), lossless encoding method, and the statistics table of the index frequency. This paper is organized in the following manner. The measure quality of reconstructed image and related works are presented in Sections 2. Section 3 details our method. In section 4 the experimental results is demonstrated that our method is better than others. Finally, conclusions are given in Section 5.

2. Related works

In this section, we address the research background of this paper. We shall firstly describe the basic idea of VQ and FSVQ, which is the central element of our proposed method. Then, we will briefly introduce two different lossless encoding techniques, Reserved Bit (RB) and Reserved Codevector (RC).

2.1 State codebook of Full Search VQ (FSVQ)

In FSVQ[4][5], each input block is encoded by searching the nearest distance codeword in the state codebook not in the super codebook. In general, the FSVQ encoder consists of a finite set of state codebooks. It regards the previously encoded blocks as a parameter to input the next-state function to predict the next one. There are mainly two different approaches used for designing the state codebook and next-state function. One consists a large codebook, called the super codebook, for each state a small subset of the super codebook is selected. The other is determining the state and then creating a state codebook separately. In our method, we encode the index table of VQ compressed image, to choice the state index-book by using the occur frequency of previously indices.

2.2 Encoding Method

The predictor in next-state function of the FSVQ is not always right as full-search. In FSVQ, in order to reduce the calculation time, it just finds the nearest distance in the sate codebook. If the prediction is correct, the FSVQ performs well. Otherwise, the best matching codeword is not in the state codebook, the quality degraded. So we can't guarantee that the quality of reconstructed image is good as full-search VQ. Two different lossless encoding techniques are described as followed.

2.2.1 Reserved bit (RB) technique

Assume that the super codebook has C codewords with $C=2^c$, and state codebook has D_{RB} codewords with $D_{RB}=2^d$. The C and D_{RB} bits ($C > D_{RB}$) are indicated the index of super codebook and state codebook, respectively. In this approach, an extra bit as a flag is needed to indicate the situation. For the current input index, the encoder compares the index and codewords in state codebook. If the index lies inside the state codebook, a "0" bit followed by the state codebook index is transmitted to the decoder. Otherwise, a "1" bit followed by the super codebook index is transmitted. In another word, in the former case $1+d$ bits are transmitted. The latter case needs $1+c$ bit to transmit. When the decoder received the first bit, the next c or d bits are picked up is decided. Receiving the first bit is "0", reconstruct the index using the state codebook, while is "1" the decoder employ the super codebook.

2.2.2 Reserved codevector (RC) technique

This method does not need to insert an extra bit for each codeword. The super codebook size is the same as RB technique, but the state codebook size is different. The state codebook has $D_{RC}=2^d-1$ codewords. If the current index locates in the state codebook, the state codebook index is transmitted using d bits. If not so, the index is outside the state codebook, a sequence of d "1"s followed by the super

codebook index. In brief, either d or $d+c$ bits are needed depended on whether the predict value is correct or not. In this way, the decoder must receive the first d bits. If the first d bits represent a number is smaller than D_{RC}, the decoder looks up the codeword in the state codebook. If the receives number is equal D_{RC}, it needs to pick up the followed c bits and then to look up the codeword in the super codebook.

The RC and RB methods provide different bit-rate. In general, if the wrong states occur frequently, RB has good performance. Otherwise, RC provides a lower bit-rate.

3. The proposed method

In this section, we shall first address the design of the Occur Frequency table, which is a two-dimension table keeps the frequency of the codeword appeared on the VQ compressed image. Then, the proposed lossless codebook compression on VQ technology will be presented.

The lossless encoding approach consists of state index-book, occur frequency table and lossless encoder. Fig. 1 describes the process.
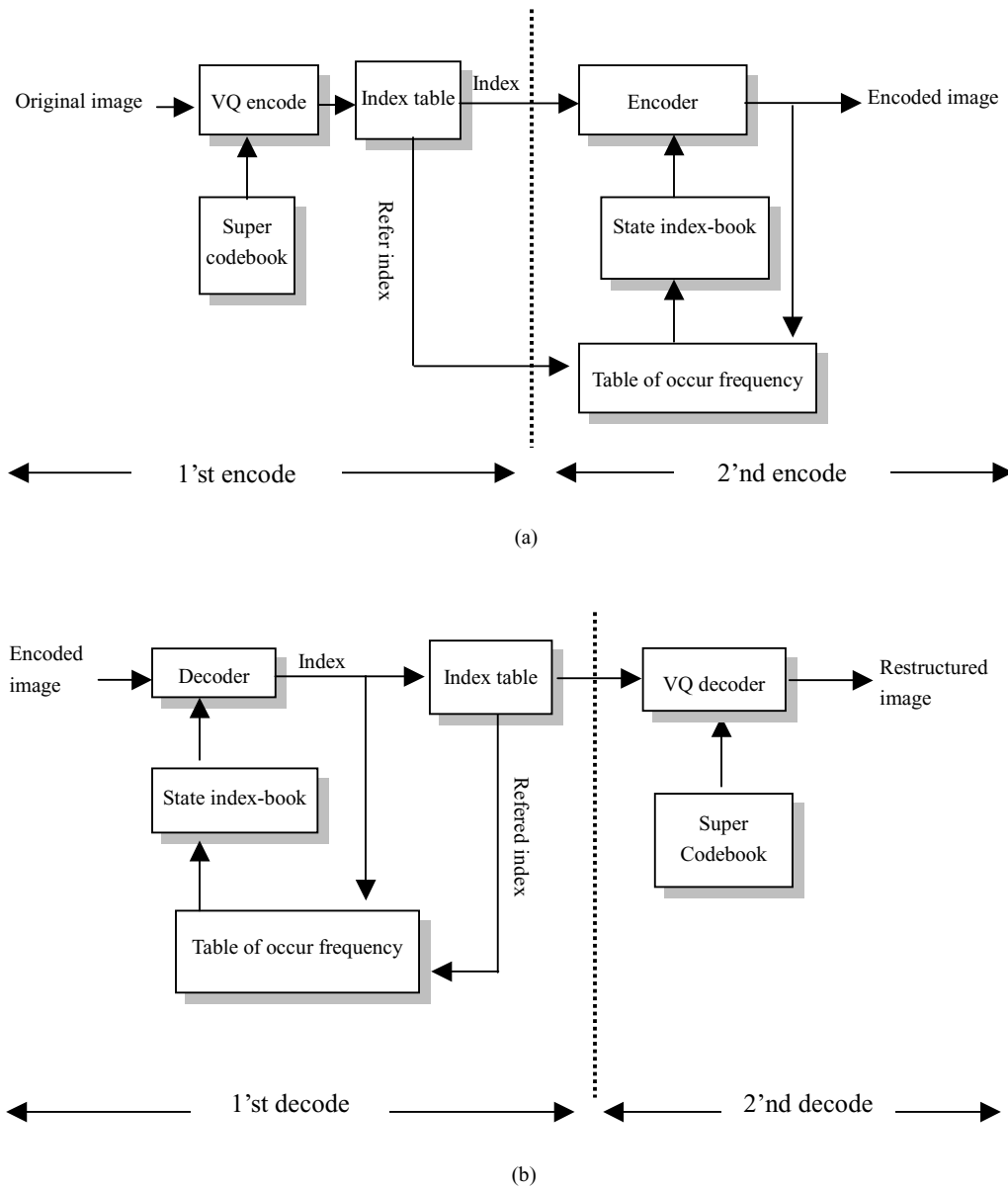

(a)

(b)

Fig. 1. Encoder and decoder of our method: (a) encoder scheme (b) decoder scheme

In the first encode phase, the original image is decomposed in to small blocks and then calculate the Euclidean distance. Finding the best matching index according the shortest distance rule. In the second encode phase, the two-dimension table of occur frequency is initialed. When the index inputs to encode, it must take the refer index to generate state index-book, and then perform lossless encoding by using RC or RB. The next process is described as follow.

3.1 The occur frequency table

In order to generate the codeword more correctly, we use a two-dimension table, called occur frequency table, to add up the index occurs times. In our method, we choice the left index as reference to predict the current encode index. As we know, in fig.2, the encode index "F", its neighbors are A, B, C, E, G, I, J and K. Before the index "F" is encoding, four indices (A, B, C and E) have encoded. So, we can refer the four difference aspects to predicate the current state. From experimental results, we decide to choice the left index as a seed to generate the current state index-book. It means that when we want to encode the index F, we can find out the reference index is equal to E, to choice the higher L frequency index to compose the state index-book. And then the encoder compares the index F and the candidate indices in the state index-book to encode the index. Final, we must add 1 to the (E, F) in the table.

A	B	C	D
E	F	G	H
I	J	K	…
…	…	…	…

Fig. 2. The index table

3.2 Determination of the state codebook size

There are two kinds of state index-book size: fixed and variable. Using the fixed size is an easy design approach. The smaller size doesn't mean the lower bit-rate, because of the right index may be not locate in the state index-book. In the situation, we need to recode the index in super codebook. According to the statistics, more than 70% index can be found in the state index-book that its candidate index size is 8. So we can set two thresholds to determine which size is need. According the rate (R) to determine which size is selected.

$$R = \frac{Count_{max}}{Count_{total}} \qquad (3)$$

In Formula (3), $Count_{max}$ indicates the maximum frequency; $Count_{total}$ indicates the total occur time. If we apply the RC method, the decision is as followed.

$$Size^{RC}_{state-codebook} = \begin{cases} 1 & ,if \quad R \geq T_1 \quad or \quad Count \leq 1 \\ 3 & ,if \quad T_1 > R \geq T_2 \ or \quad 1 < Count \leq 3 \\ 7 & ,if \quad R < T_2 \end{cases} \qquad (4)$$

In Formula (4), the *Count* indicates occurs time. If we employ the RB method, the decision is as follow:

$$Size^{RB}_{state-codebook} = \begin{cases} 2 & ,if \quad R \geq T_1 \quad or \quad Count \leq 2 \\ 4 & ,if \quad T_1 > R \geq T_2 \quad or \quad 2 < Count \leq 4 \\ 8 & ,if \quad R < T_2 \end{cases} \qquad (5)$$

In above Formula, T_1 and T_2 are predicted thresholds. According to experimental, while using RC method, the thresholds are 0.8 and 0.4. And the RB method, we set the thresholds equal to 0.6 and 0.3.

4. Experiment results

The proposed method was evaluated using thirteen test images. In the table 1 and 2, we can see the bit-rate using the fixed size of RC and RB approach individually.

Table 1
VQ and our method (RC) results for different image. State codebook size is fixed.
(Super codebook size=256, image size=512×512)

	Our method (State codebook size)			VQ
	1	3	7	
Airplane	0.335480	0.281624	0.303265	0.5
Baboon	0.490997	0.495979	0.507732	0.5
Barb	0.373230	0.366402	0.384289	0.5
Boat	0.290344	0.277748	0.309887	0.5
Girl	0.387909	0.350533	0.337780	0.5
Gold	0.353699	0.350197	0.361370	0.5
Lena	0.434052	0.418434	0.413433	0.5
Lenna	0.330505	0.300728	0.310558	0.5
Pepper	0.347351	0.311043	0.313610	0.5
Sailboat	0.305267	0.311653	0.342571	0.5
Tiffany	0.232269	0.196449	0.226635	0.5
Toys	0.176178	0.200020	0.244915	0.5
Zelda	0.366150	0.327492	0.312695	0.5

Table 2
VQ and our method (RB) results for different image. State codebook size is fixed.
(Super codebook size=256, image size=512×512)

Image	Our method (state codebook size)			VQ
	2	4	8	
Airplane	0.289070	0.288815	0.319862	0.5
Baboon	0.468567	0.452694	0.445061	0.5
Barb	0.358738	0.356724	0.369205	0.5
Boat	0.279404	0.293186	0.324078	0.5
Girl	0.353264	0.332369	0.338135	0.5
Gold	0.344425	0.342693	0.353794	0.5
Lena	0.408966	0.388447	0.384998	0.5
Lenna	0.311687	0.303074	0.323868	0.5
Pepper	0.319271	0.307972	0.324593	0.5
Sailboat	0.306961	0.318386	0.344219	0.5
Tiffany	0.219936	0.228046	0.272770	0.5
Toys	0.200390	0.238071	0.284786	0.5
Zelda	0.339752	0.316830	0.322590	0.5

From the two tables, the bit-rate of all image are lower than VQ and when the state index-book size is equal to 3 or 4 more than half images have better compression ratio then other fixed size. We can find that smooth image using the RC encode method can get the higher compression ratio, otherwise the complex image use RB method is better then RC.

Compare the fixed and variable size, we can find the result in table 3, only the image "baboon" using RC variable size, the bit-rate is larger than fixed size.

Table 3
Bit-rate results (bpp) for fixed size versus variation size.
(super codebook size=256、image size=512×512)

| Image | Our method | | | |
| | RB | | RC | |
	Fix	Var. (0.6,0.3)	Fix	Var. (0.8,0.4)
Airplane	0.288815(4)	0.280109	0.281624(3)	0.276489
Baboon	0.445061(8)	0.439564	0.490997(1)	0.499130
Barb	0.356724(4)	0.337948	0.366402(3)	0.358437
Boat	0.279404(2)	0.273067	0.277748(3)	0.272919
Girl	0.332369(4)	0.324173	0.337780(8)	0.330250
Gold	0.342693(4)	0.318237	0.350197(3)	0.326111
Lena	0.384998(8)	0.371674	0.413433(8)	0.400154
Lenna	0.303074(4)	0.288719	0.300728(3)	0.285976
Pepper	0.307972(4)	0.296913	0.311043(3)	0.296059
Sailboat	0.306961(2)	0.293163	0.305267(1)	0.300495
Tiffany	0.219936(2)	0.201637	0.196449(3)	0.178421
Toys	0.200390(2)	0.199089	0.176178(1)	0.164928
Zelda	0.316830(4)	0.298225	0.312695(8)	0.291660

Most of image with the variable size have lower bit-rate than the fixed. Because of the size is variable by the adjacency refer index, make the size is suitable for each encode index.

Generally, the SOC method refer the adjacency 4 or 12 indices has better results. In table 4, compare the SOC and our method.

Table 4
SOC and our method result.
(super codebook size=256、image size=512×512)

| Image | SOC（reference） | | Our method | | | |
	4	12	RB (4)	RB (8)	RC (3)	RC (7)
Airplane	0.328128	0.375839	0.288815	0.319862	0.281624	0.303265
Baboon	0.479965	0.467834	0.452694	0.445061	0.495979	0.507732
Barb	0.432381	0.437515	0.356724	0.369205	0.366402	0.384289
Boat	0.337395	0.381821	0.293186	0.324078	0.277748	0.309887
Girl	0.406837	0.412399	0.332369	0.338135	0.350533	0.337780
Gold	0.426910	0.428177	0.342693	0.353794	0.350197	0.361370
Lena	0.433708	0.434998	0.388447	0.384998	0.418434	0.413433
Lenna	0.433342	0.427185	0.303074	0.323868	0.300728	0.310558
Pepper	0.400566	0.411407	0.307972	0.324593	0.311043	0.313610
Sailboat	0.414207	0.411514	0.318386	0.344219	0.311653	0.342571
Tiffany	0.247604	0.334717	0.228046	0.272770	0.196449	0.226635
Toys	0.285049	0.351721	0.238071	0.284786	0.200020	0.244915
Zelda	0.427048	0.422516	0.316830	0.322590	0.327492	0.312695

From this table, it can demonstrate that our method (fixed size or variable size) has lower bit-rate than SOC. This is because the reference scope of our method is all of the previous encoded indices. By counting the index occur frequency continuously; the probability of find out the correct index in the state index-book is higher.

Table 5 and 6, we compare the SOC, LAS, Optimal FSVQ and our method.

Table 5
SOC, LAS and our method that the state codebook size is vary results.
(Super codebook size=256, image size=512×512)

| Image | LAS（8x8） | SOC | Our method | |
			RB	RC
Airplane	0.2924	0.3013	0.280109	0.276489
Boat	0.3042	0.3064	0.273067	0.272919
Gold	0.3827	0.3678	0.318237	0.326111
Tiffany	0.2285	0.2338	0.201637	0.178421
Toys	0.2188	0.2433	0.199089	0.164928

Table 6
Optimal finite-state VQ and our method results.
(Super codebook size=256, image size=512×512)

| Image | RB | | RC | |
	Our method	OFSVQ	Our method	OFSVQ
Lenna	0.289	0.271	0.286	0.249
Sailboat	0.293	0.303	0.300	0.300

These number are referred on [8][9] and [4] respectively, our method which the state index-book size is variable. From table 5, six images are tested; our proposed method also has better result. In table 6, we compare two encoding methods; the OFSVQ has lower bit-rate for the image "Lenna". For the image "Sailboat", our method using RB encode method is better than OFSVQ; and the bit-rate of RC method is equal to OFSVQ. Because of the OFSVQ collects a great deal of information before encode phase, this is vary efficient for the smooth image such as image "Lenna". If the image is not smooth, the statistic may be not effective such as "Sailboat". Thus, we believe that the proposed compression method using encoded index of the image itself as a reference can get the good results, and does not need to perform the collection before encode the image.

5. Conclusions

Our method represents a new lossless compression method applied on the indices of the VQ compressed image. The main contribution reported in this paper is to reduce the bit-rate by varying the state index-book. A statistic table and state index-book were developed. This is variable length coding method. In the encode phase, two conditions occur, one belonging to the case when the right index is inside the state index-book and another one belonging to the case that is not located in it. Simulation results showed

the bit-rate that applied fixed state index-book can reduce about 60% as compared to the traditional VQ. In order to make the size more suitable for encoding, we set two thresholds to determine the state index-book size. Compared to the other methods, the reference indices are increase in our method, and the bit-rate is lower.

Reference

[1] R.M. Gray (1984), Vector quantization, *IEEE ASSP Magazine*, pp.4-29.

[2] A. Makur and S.S. Selvi (2001), "Variable dimension vector quantization based image watermarking", *Signal Processing*, pp.889-893.

[3] R.Y. Li, J. Kim, and N. Al-Shamakhi (2002), "Image compression using transformed vector quantization", *Image and Vision Computing*, pp.37-45.

[4] A. Cziho, B. Solaiman, I. Lovanyi, G. Cazuguel, and C. Roux (2000), "An optimization of finite-state vector quantization for image compression", *Signal Processing: Image Communication*, pp.545-558.

[5] Y.C. Liaw, W. Lo and Z.C. Lai (2002), "Image restoration of compressed image using classified vector quantization", *Pattern Recognition*, pp.329-340.

[6] G. Deng (2002), "Transform domain LMS-based adaptive prediction for lossless image coding", *Signal Processing: Image Communication*, pp.219-229.

[7] M. Xu and A. Kuh (1996), "Image coding using feature map finite-state vector quantization", *IEEE Signal Processing Letter*, pp.215-217.

[8] C.H. Hsieh and J.C. Tsai (1996), "Lossless compression of VQ index with search-order coding", *IEEE Transactions on Image Processing*, Vol. 5, No. 1, pp.1579-1582.

[9] C.C. Chang, C.H. Sung and T.S. Chen (1997), "A locally adaptive scheme for image index compression", *Proceedings of the 1997 Conference on Computer Vision, Graphics, and Image Processing*, Taichung, Taiwan, pp.93-99.

[10] S.B. Yang and L.Y. Tseng (2001), "Smooth side-match classified vector quantizer with variable block size", *IEEE Transactions on Image Processing*, Vol. 10, No. 5, pp.677-685.

[11] C.H. Hsieh, K.C. Chuang and J.S. Shue (1993), "Image compression using finite-state vector quantization with derailment compensation", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 3, No. 5, pp.341-349.

[12] S.A. Rizvi and N.M. Nasrabadi (1997), "Finite-state residual vector quantization using a tree-structured competitive neural network", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 7, No. 2, pp.377-390.

Dynamic Motion Estimation Search Window Size Calculation in H.264 Standard Video Coder

Gianluca Bailo, Massimo Bariani, Ivano Barbieri, Marco Raggio

Department of Biophysical and Electronic Engineering
University of Genova
Via Opera Pia 11 A,
16146 Genova,
ITALY

Abstract

The JVC/H.264 standard video coder can obtain high compression rates together with important video quality improvements if compared with existing standards. On the other hand, the encoder complexity, if compared with previous video-compression standard algorithm, is largely increased, mainly because of the high computational cost of the Motion Estimation module.

In this paper, we propose an innovative algorithm for reducing the complexity of the Motion Estimation (ME) module. The main idea is to make dynamically modifiable a previously static H.264 coder parameter: the size of the search window in motion estimation. A Motion Detection module has been added to the video coder in order to compute the search window size. Detailed motion estimation is performed only when and where it is required. The Motion Detection algorithm is based on blob coloring [1] over sub-sampled binary images generated by block wise threshold image difference.

In the proposed solution, a reduction in the number of calculated SAD (Sum-of-Absolute-Differences) allows decreasing the encoder complexity, especially in low complex sequences (e.g. video-surveillance and video-telephony applications). H.264 [2] has been used for validating the proposed approach; nevertheless there are no algorithm's constraints in use it for other video coding standards based on motion estimation.

1. Introduction

The wide range of multimedia applications based on video compression (video telephony, video surveillance, digital television, distance learning, mobile applications) leads to different kind of requirements for a video-coding standard (image quality, compression efficiency). Today there are new high band request from actual multimedia services and efficient video coders are needed in order to obtain a sensible reduction of video streams high band occupation.

The new video coding standard H.264 seems to be answer to fulfill these requirements. The ITU-T recommendation H.264 (also referred to as MPEG-4 part 10) has been jointly published by the Moving Pictures Experts Group (MPEG) of ISO/IEC and the Joint Video Team (JVT) of ITU-T.

Compared with previous standards, H.264 introduces many new features in all the aspects of the video encoding process. Firstly it can use 7 blocks configuration (see Figure 1). If all blocks are active, exhaustive motion estimation is performed and better quality and compression performance are achieved.

Figure 1 – Block configurations for ME

Another H.264 key feature is the multiple frame reference; in this case motion estimation is performed N time (where N is the number of reference frames) in order to improve quality and compression. Anyway the computational complexity is increased N times respect to the single motion estimation application.

H.264 achieves 50% bit-rate save at same quality compared with existing video coder standards as H.263 [3], but the complexity of the encoder has been increased of more than one order of magnitude (while the decoder is increased by a factor of 2) [4].

The high compression rate together with the good quality obtained by the H.264 standard make it suitable for a large variety of applications (video-surveillance, video telephony on mobile terminals). Moreover the introduction of new features like the new frame types called SP-frames and SI-frames [7] allow implementing bit-stream switching and "VCR functionalities" (random access, fast forward, fast backward, etc.) in a more efficient way [7]. Therefore, Video-Surveillance operations like sequence retrieval, switching between video stream coming from different cameras and navigation in video-streams can be better performed using H.264 algorithm.

Several H.264 application areas require high power efficiency (especially in the video encoder part) in order to work on embedded systems and mobile terminals. This requirement implies the need to dramatically reduce the complexity of the H.264 video encoder.

Algorithm analysis shows that mode-decision and ME modules are the most complex in the H.264 encoder (especially when Rate-Distortion Optimization is used) [5]. This is mainly due to the great number of SAD calculation in ME (e. g. 1500 millions in the first 50 frames of the standard sequence *foreman,* using 5 reference frame and all-blocks configuration activated). There are many algorithms performing a reduction of the number of SAD calculation based on spatial and temporal correlation of motion vector [8].

In the following, we will introduce a different approach, based on motion detection and blob coloring algorithms. The algorithm proposed in this paper can significantly reduce H.264 ME computational cost and achieves the best performance in sequences with low complexity (where spatial movement is limited in time), as we can have in both video-surveillance and video-telephony environments. This approach is useful for each configuration feature of the H.264 coder.

2. Proposed Algorithm

The proposed algorithm is based on the idea that exhaustive motion search is useful only in video sequences containing large motions and not in low complex sequences. H.264 can use its entire feature set if there is the real need to do it. The H.264 ME full-search can be applied just when the motion detection algorithm identifies motion and, specifically in the region where the motion is detected. In particular, the motion detection module is utilized to calculate a parameter that is usually a constant, the *search window size*. Motion detection evaluates where and when to set a large window for the motion estimation; otherwise a minimum-size window will be used. Since this parameter strongly influences the number of calculation performed in the video coder, the dynamical setting of this value can save a lot of time in the compression process with minor effects on the quality of the produced video sequence.

The scheme in Figure 2 explains where the motion detection module works in the window search estimation.

Figure 2 - Algorithm position in H.264

This algorithm consists of several steps performed for each frame of the input video sequence. First of all the current frame is divided into 8x8 blocks (Figure 3), then the image difference with threshold is computed.

Figure 3 – Frame 8x8 blocks

The difference is calculated between the current frame and a reference frame (the latest intra-frame). A threshold (called *detection threshold*) is applied to the resulting difference, producing a binary image of black and white pixels: if the original value is less than the threshold, the pixel value is assigned to 0 (a white pixel); otherwise it is assigned to 255 (a black pixel). As we will show in the following, the detection threshold is a critical parameter in the efficiency-quality tradeoff.

The sum of the white pixels is evaluated for every 8x8 block. If the resulting sum value is greater than a fixed white-pixel density the status of the related block becomes activated, else it is not activated. Obviously, the greater is the detection threshold the smaller is the density of the white pixels. Therefore, this parameter determines the sensitiveness of the motion detection algorithm (setting a great threshold implies the detection of considerable movements only).

At this point we have a set of activated and not-activated blocks. Each block can be viewed as a pixel of a reduced-size binary image. Blob coloring [1] is performed over this sub-sampled binary image obtained from the described steps. The result is an array of elements, each one composed by two coordinates identifying motion object boundary (bold rectangles in Figure 4).

Figure 4 - object boundary in bold

The information obtained applying the blob-coloring algorithm, is utilized to define an approximate motion image (Figure 5). This is an approximation of the blob-coloring output result, because we need as little as possible complexity in motion detection algorithm.

```
0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 1 1 0 0 0 0 0
0 0 0 0 1 1 1 0 0 0 0 0
0 0 0 0 1 1 1 0 0 0 0 0
0 0 1 1 1 1 1 1 0 0 0 0
0 0 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0
```

Figure 5 - Motion Images Example

During the video coder motion search, the motion image is scanned in order to decide the search window size for every H.264 macro block. Larger window size will be applied for macro blocks where motion has been detected.
Otherwise the video coder will use a minimum-size window.

3. Results
The proposed algorithm has been validated with version jm60a [6] of the reference JVT software. The reported tests have been performed using four standard sequences in QCIF format (tests have been also performed on CIF format with analogue results). *Akiyo, hall* and *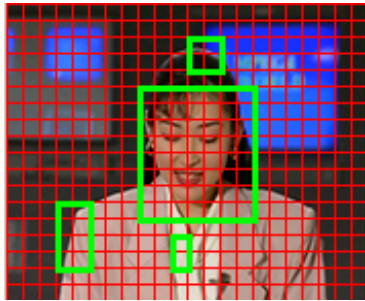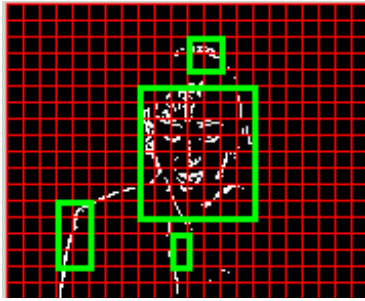salesman* have been selected on the basis of application of interest (video-surveillance and video conference), while *foreman*

has been utilized for comparison purpose in order to test the proposed algorithm with a more complex sequence. In the following tests, we encoded the sequences at 30fps. The CAVLC entropy coder and Hadamard were used for all tests, with quantization values of 28. The H.264 activated block configuration is 8x8 16x16.
From the results reported in this section it can be noted as our proposed scheme can simplify the encoder complexity.
The performance is measured evaluating three aspects: time spent to compress the selected video sequences, size of compressed sequences, and achieved quality. These values are compared to the reference software using a range of values for the detection threshold parameter; the zero value corresponds to the original H.264 coder without applying the presented algorithm. Figure 6 shows the gain obtained by this algorithm in terms of compression time for the selected standard sequences. The proposed approach achieves from 50% to 60% encoding time reduction in typical video-surveillance and video-telephony sequences. Anyway, also the worst-case *foreman* shows a good computational time reduction (about 30% for threshold equal to 5, about 60% for thresholds greater than 40).
The H.264 coding efficiency is only slightly decreased, as can be noted from the graphs in Figure 8 – Size of compressed sequence Foreman , 8, and 9 (also in these cases the detection threshold values are on the x-axis).
The quality is evaluated using the SNR Y, measured by the jm6.0a test model [6]; in Table 1, SNR Y values are shown in decibel for different ranges of detection threshold values. As previously explained, the proposed algorithm is strongly influenced by the detection threshold parameter, as the estimation of detected movements depends on the threshold value (using a very high value not all of the motions can be detected, otherwise a very low value can cause the detection of background noise as relevant motion). This can explain the reduction of the compression time together with a minor reduction of image quality when increasing the value of this parameter. Setting the detection threshold to the maximum value will cause a unitary window size, as the motion detection module will never signal the presence of motion in the input sequence.

Figure 6 - Compression time

Figure 7 – Size of compressed sequence Akiyo

Figure 8 – Size of compressed sequence Foreman

Figure 9 - Size of compressed sequence Hall

Threshold	akiyo	foreman	hall	salesman
0	38.28	36.13	37.11	35.55
5-25	38.28	36.13	37.10	35.55
30	38.28	36.12	37.10	35.55
40-45	38.28	36.11	37.10	35.55
50-70	38.28	36.10	37.09	35.55
75-95	38.28	36.09	37.09	35.55

Table 1 – SNRY comparison

4. Conclusions

In this paper, an innovative algorithm for ME complexity reduction based on motion detection has been presented. The proposed algorithm has been integrated in the JVT reference software jm60a [6] and validated using standard CIF and QCIF sequences. Tests show significant results in video-surveillance and video-telephony standard sequences where the proposed approach allows performance improvement applying exhaustive motion estimation only in rarely situations. The motion detection algorithm has a very low complexity due to sub-sampled image blob coloring application allow an H.264 encoder speed-up. Moreover motion detection is a part of many last generation video-surveillance applications.

Future development will focus on H.264 blocks scalability.

5. References

[1] D. H. Ballard and C. Brown, *Computer Vision*, New Jersey, Prentice Hall, 1982.

[2] ISO/IEC 14496-10, ITU-T Rec.H.264, Joint Video Specification, October 2002.

[3] ITU-T Recommendation H.263, "Video coding for low bitrate communication", Feb. 1998

[4] S. Saponara, C. Blanch, K. Denolf, and J. Bormans, "The JVT Advanced Video Coding Standard: Complexity and Performance Analysis on a Tool-By-Tool Basis", Packet Video 2003, Nantes, France, April 2003

[5] P. Yin, H. C. Tourapis, A. M. Tourapis, and J. Boyce, "Fast Mode Decision And Motion Estimation For Jvt/H.264", International Conference on Image Processing - ICIP 2003, Barcelona, Spain, September 2003

[6] JVT Reference Software version jm 6.0a, Bs.hhi.de/~suehring/tml/download/

[7] M. Karczewicz and R. Kurceren, "The SP- and SI-Frames Design for H.264/AVC", Ieee Transactions On Circuits And Systems For Video Technology, Vol. 13, No. 7, July 2003

[8] R. Korada and S. Krishna, " Spatio-Temporal Correlation based Fast Motion Estimation Algorithm for MPEG-2," 35th IEEE Asilomar Conference on Signals, Systems and Computers, California, November 2001

Proceedings

The 2004 International Workshop on Visual Languages and Computing (VLC'2004)

Editor

Kang Zhang, University of Texas at Dallas, USA

Visualizing the Construction of Decision Trees Using Treemaps

Cristiane Santana
Manoel Mendonça,
Daniela Cruzes
Salvador University - Unifacs , Rua Doutor José Peroba, no 251,
STIEP,CEP 41770-235, Salvador - BA, Brazil
E–mail: christiane.Santana@unifacs.br, mgmn@unifacs.br, daniela@unifacs.br

Abstract

Most of the approaches published in the literature proposes a completely automatic process to generate decision trees. These approaches miss valuable expert tacit knowledge input during the construction of the tree. This paper describes an approach for interactive construction of decision trees. The approach is user-centered. It combines the strengths of the user and the computer to build better decision trees. The user provides domain knowledge and evaluates intermediate results of the algorithm. The computer automatically creates patterns satisfying user constraints and generates appropriate visualizations of the produced tree. A tool was developed to support this approach. It combines treemap visualization, visual data mining mechanisms, and the J48 (Weka) algorithm to interactively build a decision tree.

Index Terms—Visual Data Mining, Decision Trees, Classification, human-computer interaction.

1. Introduction

DURING the last decade, data repositories have grown faster than our ability to analyze them. The area of Knowledge Discovery from Databases (KDD) has appeared as an attempt to balance this equation. KDD aims to extract non-trivial, previously unknown and potentially useful information from data repositories [8][14]. KDD is a highly iterative process that goes from the definition of the analysis goals to the extraction and assimilation of knowledge from the data repository.

The main activity of this process is data mining. Data mining is characterized by the use of algorithms to extract useful knowledge from pre-processed data sets. Classification is one of the most common tasks in data mining [14], and the construction of decision trees is one of the most popular classification methods. Decision Trees are intuitive, easy to interpret, relatively fast to construct and, compared to other classification methods, have equal or better accuracy [7].

Many approaches for decision trees construction have been proposed in the literature. However, most of them, focus on the algorithm and follows a completely automatic process for constructing a decision tree. In these approaches, the expert configures only a few parameters before the start of the tree construction process. As the expert is not involved in the tree construction process itself, precious human knowledge is wasted instead of being inserted on the final model [2][3]. This knowledge includes tacit domain and context knowledge that resides in the expert head and was gained over the years of experience on data gathering and analysis. This knowledge is not coded formally in the data so it cannot be automatically incorporated by machine learning algorithms.

This work describes an interactive approach for the construction of decision trees, which combine mechanisms for user intervention with a traditional decision tree construction algorithm [7]. This approach allows the intervention of the expert at any time during the tree construction. This enables the expert to verify and, if desired, direct the choice of classification attributes and splitting points of the tree. This allows the expert to interactively bring his domain and context knowledge to the tree construction process. In this user-centered approach, the expert and the computer can both contribute to the tree building process: the user providing domain knowledge and evaluating intermediate results of the algorithm; and, the computer automatically extracting patterns, satisfying the user constraints, and generating appropriate visualizations of these patterns.

In order to make this approach work, one needs to use effective interaction mechanisms between the machine and the experts. For that, we use visual data mining techniques that combine powerful visualization techniques with interactive query mechanisms [15]. In particular, we use a hierarchical visualization technique called Treemap [5][17][18], combined with graphical widgets for fast tree exploration and querying. These mechanisms enable the experts to visualize and explore the intermediate trees built by the J38 classification algorithm [11]. This structure allows human interaction to be performed with the computer during the whole classification process.

2. Decision Tree Classifiers

Decision tree classifiers [7] learn a discrete-valued classification function, which is represented by a decision tree. Figure 1 shows an example of a tree for deciding if one can play outside or not. The tree has leaf and intermediate nodes. Each intermediate node corresponds to an attribute test. Edges symbolize all possible outcomes of the test in the node. The leaf node contains the label of one of the existing classes (in the example, yes or no). A path from the root to a

leaf node defines a classification rule in the decision tree. In Figure 1, the left most path contains the rule: IF outlook=sunny AND humidity<=75 THEN yes (one can play outside).
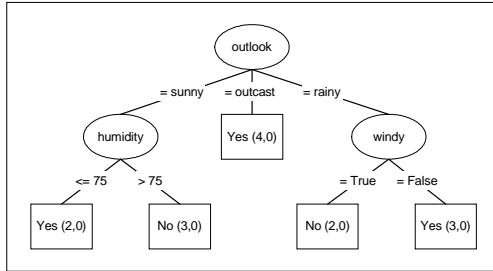

Figure 1. A Decision Tree

The construction of the decision tree consists of two phases: (1) the construction itself; and, (1) the pruning of the tree. In the first phase, the tree is constructed by recursively partitioning the training set until each partition consists mostly of records of the same class. This set is then labeled as a leaf node. For each intermediate node, an attribute is selected. This attribute must not have been used yet in the classification path. The chosen attribute is the one that promotes the maximum segregation among the records with respect to the classification criteria. If the selected attribute is categorical, a sub-tree is created for each of its possible values. If the attribute is numerical, the algorithm verifies the value (or values) that better splits the data with respect to class segregation. A split (normally binary) is then created, with tests of the type "less than (<)" and "equal or greater than (≥)" the chosen split value.

The construction of the tree is guided by the objective decreasing the difficulty of classification. Thus, the choice of the attribute that will constitute the decision node, as well as the selection of the split point for numerical attributes, is carried through the maximization of the discrimination between the classified classes [6][7].

Many times the resulting trees have very specialized rules. These trees adjust excessively to the peculiarities of the training set, and tend to produce deductive models of the training set instead of inductive models of the real world. On those trees, leafs nodes are supported by a small number of examples that represents isolated facts and do not reflect reality. The pruning phase consists of simplifying the constructed tree to remove its excessively specialized parts. This generates less complex and more significant decision tree structures.

3.   Interactively Building Decision Tree Classifiers

Many approaches for decision tree construction have been proposed in the literature. Most of them, however, focus on completely automated algorithms for tree construction, allowing only a few parameters to be configured before the start of the algorithm. Typically, the user involvement in this process is limited to the choice of the data and the parameterization of the algorithm that will be used. Once the construction process starts, the algorithm does not allow user intervention or the visualization of intermediate results, only the final model is shown to the user [2].

Ankerst proposed an interactive model for decision tree construction [3]. In this model, the user cooperates with the computer in the construction of the tree. The user can choose the next node to be expanded, select the attribute that will partition the data as well as its split points, or to leave those decision to the system.

3.1. An Approach for Interactive Decision Tree Construction

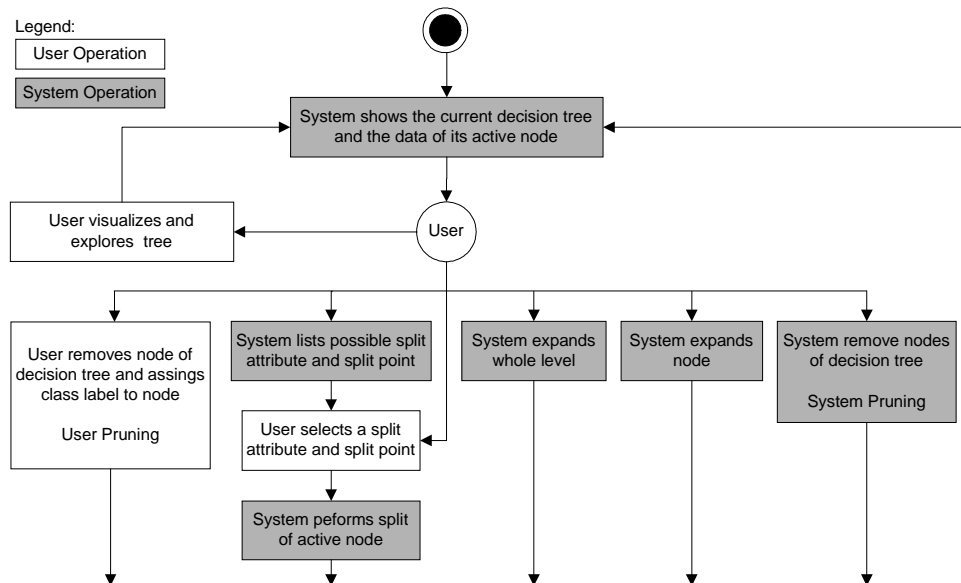In our approach, the construction of a classification tree is

Figure 2. Approach for Interactive Decision Tree Construction

highly interactive. For this, we adapted Ankerst model to the approach shown in Figure 2. In this figure, the activities performed by the user are shown as white rectangles and the activities performed by the computer are shown as gray rectangles.

The approach is centered on the user. After the beginning of a work session, the user can choose one of the following operations:

1) Manually remove a node of the tree. This corresponds to a manual pruning, where the current node is transformed into a leaf node and all its children are removed from the tree;

2) Ask the system to suggest a list of possible classification attributes and splits points for a specific node. The attributes are ordered by its mathematical information gain for class segregation;

3) Choose an attribute and its split point, and ask for the system to execute the expansion of a node based on this attribute;

4) Ask the system to explore the data. The systems opens the tree exploration tool and allows the user to browse and query the tree constructed up to that moment;

5) Ask the system to automatically expand a node of the tree. The system automatically chooses the attribute with bigger mathematical gain and expands the node;

6) Ask the system to automatically expand a level of the tree. The system automatically expands all nodes in the current level of the tree;

7) Ask to the system to prune the tree constructed up to that moment. The user inputs the parameters for pruning, and the system automatically removes from the tree all nodes that it judges too specialized for effective classification.

The approach described in Figure 2 has some important differences from the model originally proposed by Ankerst: (1) it allows the expansion of a complete level of the tree; (2) it permits tree pruning at any time during its construction; and, most important of all, (3) the visualization and interactive exploration mechanisms are quite different. Ankerst approach is pixel-based [4][12]; in it, each pair attribute-value is represented by one colored pixel in the visual screen. We propose the use of treemaps [18] associated with query devices and details on demand controls [13][15]. We argue that these resources are instrumental to the success of the approach shown in Figure 2. They are intuitive and allow the user to explore the characteristics of these trees and the data associated with their nodes.

4. A Visual Classification Tool

4.1. Treemaps

As said before, the proposed decision tree construction approach uses treemap visualization to allow user-computer interaction. Treemap is an information visualization technique proposed by Shneiderman for visualizing hierarchical structures [17][18]. They use 100% of the available space for information visualization, mapping hierarchies into rectangular regions. This contrasts with the traditional representation of trees that uses lines to establish the connection between parents and children nodes of a hierarchy. The traditional type of representation has two significant disadvantages: (1) a great portion of the available visual space is spent in the organization of the nodes; and, (2) non-trivial hierarchical structures generate trees difficult to visualize.

Because it uses all the available space for drawing, treemaps allow the efficient visualization of large hierarchies

Figure 3. Visual Data Exploration of a Classification Tree

Figure 4. (a) Interactive Classification - Node Selection (b) Expansion of Fig 4a selected node (c). Decision Tree automatically constructed

that can be in the order of thousands of items [17]. It is also very efficient in coding node attributes using the rectangles size and color.

We have built a visual data-mining tool that uses treemaps as its main visualization paradigm [1]. A screenshot of the tool, named TreeMiner, is shown in Figure 3. Part 1 of this figure shows the use of a treemap to present a hierarchy. The users can choose interactively the data variables that define the shown hierarchy (Part 2) and examine .any record of the hierarchy in detail (Part 3). They can also dynamically associate any variable of the data with visual attributes like color, label, and rectangle size (Part 4).

4.2. Interactive Classification

Based on the TreeMiner and a traditional decision tree constrution algorithm, we wrote a tool to support the approach shown in Figure 2. The new tool allows for the use

of the TreeMiner visualization structures, dynamic query and details on demand devices, while executing a decision tree construction algorithm. The visualization of intermediate trees is brought up to date at each step of the tree construction process, allowing the expert to explore it, and re-direct the tree building process if necessary.

The new tool has all the TreeMiner funcionalities, except for the dinamic association of variables with the visual attributes "color" and "size". In the new tool, the visual attribute "size" is used to represent the number of records that supports a node of the decision tree, and the visual attribute "color" is used to represent node's purity degree – i.e., the number of records of the classified attribute that belongs to a given class.

In the new tool, each class (e.g., yes and no) is a associated with a distinct color (e.g., red and yellow). The purity degree of a leaf node can be visualized by the clarity

of the color representing it. This way, the tool creates an intuitive representation for the decision tree nodes support and purity at any point of its construction.

Figure 4 (c) shows, as a treemap, the same decision tree presented earlier in Figure 1. This representation not only shows the values of the classifying attributes, but it also shows the purity and support of its leaf nodes through its rectangle color and size. This way, the treemap offers a one shot view of the decision tree and its class distribution.

Besides the visualization, the developed tool supplies interaction mechanisms based on visual data mining techniques to assist the user to explore the tree at any point of its construction. Upon selecting a node (figure 4a-1) in the tree, the information about that node is shown in right superior corner of the screen (figure 4a-3). Three informations are shown to the user: (1) the dominant class of the node; (2) the amount of registers that supports the node; and (3) the total number of impurities in the node (i.e., the number of records that do not belong to the node dominant class). Moreover, if necessary, the user can also request the list of all the registers that support the selected node.

By selecting a node in the interactive mode of a decision tree construction, the user can also choose the split attribute, based on the calculated gain information (Figure 4a-2).

The tool also has in its bottom right corner filters to explore the decision tree, see Figure 4c. The "depth filter" hides deeper nodes of the tree, allowing to the user to quickly examine simplified trees, and take pruning decisions. The "class filter" redraw the tree using only the selected classes. The "impurity filter" allows the interactive verification of nodes that have impurity values over or under the set values. And, the "number of records filter" allow the verification of the nodes that have support over or under the set values. It is important to observe that the time between user interaction with these filters and the tool updating of the visual screen is practically zero. With these interaction resources, the user can quickly decide which node to remove or to expand, or if the obtained tree is already satisfactory.

4.3. Some Implementation Details

Besides the TreeMiner developed at Salvador University [1], the tool uses functionalities from Weka[1] (Waikato Environment will be Knowledge Analysis) functionalities. Weka is an environment for knowledge discovery that was developed at the University of Waikato in New Zealand [9][11]. It implements algorithms for several data mining techniques, including the J48, an improved version of the C4.5 for building decision trees [16].

From Weka, we used the structure for creating classification sessions, its resources for data preparation, its resource for algorithm parameterization and the base of the J48 algorithm for constructing decision trees. The algorithm, however, had to be adapted in order to make it possible the

interaction of users during tree construction. It was adapted in three points: (1) to allow the tree construction level by level, (2) to allow the tree construction node by node, and (3) to list the attributes´ information gain and their splitting points for any given node. For the first functionality, the algorithm had to be modified to work breadth first (level by level) instead of depth first. Figure 5 illustrates the old and the new sequence of node construction carried by the algorithm.

Figure 5. Data Exploration

Besides the mentioned adaptations in the Weka and the TreeMiner tools, the following functionalities are being added to the tool: (1) a functionality for user annotation during the whole decision tree construction process; (2) filters to assist data selection for the training sets; (3) a functionality for mapping numerical attributes into categorical ones; and, (4) a functionality for allowing the assignment of weights for the attributes in the J48 algorithm.

The tool is currently being used to explore and build classification models from empirical software engineering data.

5. Conclusion

The current classification approaches allow a limited participation of the expert during the decision tree construction process. These approaches do not take advantage of the experts´ knowledge with respect to the data and domain being mined. With a cooperative classification approach, the expert and the computer can contribute with their best. The computer providing the capacity to recognize mathematical patterns. The experts providing their ability to interpret and promote a deeper understanding of these patterns. This combination should generate more trustworthy models.

This work defined an approach and implemented a tool for the interactive construction of decision trees. The defined approach is adapted from the cooperative classification model originally proposed by Ankerst. The implemented tool enacts this model, using treemap visualizations and visual data mining query devices to create efficient mechanisms for supporting user-computer interactions during the construction of decision trees.

The tool was developed by combining functionailities from a visual data mining tool, TreeMiner, with an environment for knowledge discovery, Weka. The resulting tool supports all phases of classification process, from the data selection to the interactive exploration of the obtained results.

For future work, we intend to include new mechanisms for

visualizing and interacting with the decision trees. In particular, we intend to add a module that uses traditional tree drawing methods to visualize and interact with the produced trees. This will allow the comparison of the treemap approach with more traditional approaches of tree visualizations and exploration. We also intend to incorporate a module for visually comparing generated trees by overlapping them on the computer screen at each iteration of the tree construction process.

References

[1] Almeida, M.; Mendonça Neto, M. Using Treemaps to Internalize Knowledge (in Portuguese). In the Proceedings of the First Brazilian Workshop on Information Systems and Knowledge Management, SBC, Fortaleza, CE, Brazil, 2003. v. 1, p. 1-11.

[2] Ankerst, M., Elsen, C., Ester, M., Kriegel, H.-P.: Visual Classification: An Interactive Approach to Decision Tree Construction. In ACM SIGKDD 5th Int. Conf. On Knowledge Discovery and Data Mining, San Diego, CA, USA, pp. 392-396 (1999).

[3] Ankerst, M., Ester, M., Kriegel, H.-P.: Towards an Effective Cooperation of the User and the Computer for Classification. In ACM SIGKDD 6th Int. Conf. On Knowledge Discovery and Data Mining (KDD 2000), Boston, MA, pp. 179-188 (2000).

[4] Ankerst, M.: Visual Data Mining with Pixel-Oriented Visualization Techniques. Proc. Workshop Visual Data Mining (2001).

[5] Babaria, K.: Introduction to Treemap. University of Maryland. [on line] http://www.cs.umd.edu/hcil/treemap3/TreemapIntroduction.pdf (2001).

[6] Barlow, T., Neville, P.: Case Study: Visualization for Decision Tree Analysis in Data Mining. Proc. of IEEE Symposium on Information Visualization - INFOVIS´01 (2001).

[7] Breiman, L., Friedman, J., Olshen, R., and Stone, C.: Classification and Regression Trees. Belmont, CA: Wadsworth (1984).

[8] Fayyad, U., Piatetsky-Shapiro, G., Smyth, P.: From Data Mining to Knowledge Discovery in Databases. In AI Maganize, pp. 37-54 (1996).

[9] Frank, E. and e. al: Weka [http://www.cs.waikato.ac.nz/ml/weka/], The University of Waikato.

[10] Garner, S.R.: ARFF-the WEKA dataset format. World Wide Web hypertext document at http://www.cs.waikato.ac.nz/~ml/weka/arff.html.

[11] Garner, S.R.: WEKA: The Waikato Environment for Knowledge Analysis. In Proc. of the New Zealand Computer Science Research Students Conference, pages 57—64 (1995).

[12] Keim, D. A.: Designing Pixel-Oriented Visualization Techniques: Theory and Application. In IEEE Trans. on Visualization and Computer Graphics, vol 6: IEEE Computer Society, pp. 59-78 (2000).

[13] Keim, D. A.: Information Visualization and Visual Data Mining. In IEEE Trans. on Visualization and Computer Graphics, vol 7, n° 1, pp. 100-107 (2002).

[14] Mendonça Neto, M.; Sunderhaft, N. A State of the Art Report: Mining Software Engineering Data. State of the Art Technical Report DACS-SOAR-99-3. U.S. Department of Defense (DoD) Data & Analysis Center for Software, Rome, NY, 1999. Also available in: http://www.dacs.dtic.mil/techs/datamining/datamining.pdf

[15] Oliveira, M., Levkowits, H.: From Visualization to Visual Data Mining: A Survey. In IEEE Transactions on Visualization and Computer Graphics, vol 9, n° 3, pp. 378-394 (2003).

[16] Quilan, J. R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, CA (1993).

[17] Shneiderman, B., Johnson, B.: Treemaps: A Space-Filling Approach to the Visualization of Hierachical Information Structures. Proc. of IEEE Information Visualization, pp. 275-282 (1991).

[18] Shneiderman, B. Tree visualization with tree-maps: 2-d space-filling approach. ACM Transactions on Graphics, vol. 11 , n°. 1, pp. 92-99 (1992).

Using Visualization to Bring Context Information
to Software Engineering Model Building

Daniela Cruzes and Manoel Mendonça
Salvador University - Unifacs , Rua Doutor
José Peroba, no 251, STIEP,CEP 41770-235,
Salvador - BA, Brazil E-mails:
daniela@unifacs.br, mgmn@unifacs.br

José Carlos Maldonado
Instituto de Ciências Mat. e de Computação
Universidade de São Paulo, Av. Trabalhador
São-Carlense, 400, C.P. 668 CEP 13560-970
São Carlos, SP, Brazil. E-mail:
jcmaldon@icmc.usp.br

Mario Jino
Electrical Engineering Department -
DCA/FEEC at /UNICAMP. C.P. 6101 CEP
13083-970 Campinas (SP), Brazil. E-mail:
jino@dca.fee.unicamp.br.

Abstract

Software engineering model building is not an immature science anymore. Over the last decades, a large number of model building approaches have been proposed in the literature. These approaches include both statistical and machine learning-based techniques. However, most of them are essentially data-driven, in the sense that very little input is given by humans during the model building process. This paper argues that model building should consider expert context information during the model building process and that visualization and interactive data exploration are important mechanisms to do that. We propose a model-building framework in which context information is considered during all activities of the model building process. We finish by presenting a case study in which visual data exploration is used to bring context information to a traditional model building approach.

Index Terms: model building, software engineering, data mining, and visualization.

1. Introduction

SOFTWARE engineering data repositories are now commonplace in software organizations and data mining is pointed as the tool of choice to explore this data. Formally, data mining can be defined as the extraction of new, useful, and non-trivial information from data repositories [1,5]. Data mining represents a shift from verification driven to discovery driven data analysis. The main goal of verification driven data analysis is to validate hypotheses postulated by data analysts, usually to confirm an idea. The tool of choice for verification driven data analysis is inferential statistics.

The main goal of discovery driven data analysis is to automatically, or semi-automatically, extract useful information from volumes of data. This "useful" information manifest itself in the form of patterns, rules, and ready to use prediction, classification, and estimation models. There is a large spectrum of techniques that can be used for discovery driven data analysis.

In software engineering, a good portion of the work on discovery driven data analysis has focused on model building [3,4,8]. Much of it even precedes the data mining age. The majority of the old cost estimation modeling can be classified as discovery driven, as those models are built directly from data.

It is our position that model building is the tip of the iceberg. A good model is where one wants to arrive, but some issues have to be considered in order to get there. This paper discusses how data mining geared towards visualization and interactive data exploration can help with that. It presents a framework for data collection and analysis that highlights the need for collecting and considering context information during the model building process. The paper finishes by presenting a case study in which visual data exploration is used to bring expert context information to a traditional model building approach.

2. Context Information

Data-driven model building is an activity that has much to profit from learning about the data. Some of the questions a modeler wants to answer before he builds a model from a data set are: (1) where the data comes from? (2) In which context it was gathered? And, (3) what is the quality of this data. There are two complementary approaches for answering these questions: (1) store context information together with the data; and (2) extract context information about the data.

The former is the best approach because context information is easier to collect when the data itself is colleted. However, one cannot predict all future usages of a data repository. It is impossible to foresee all the context information one may needed to record. Besides, data collection costs money and there is a trade off between how much one can spend and how much data (and context information) one wants to collect.

2.1. Gathering Context Information

Although it costs money, there a minimum set of context information that we consider necessary to store with any data set. Model building greatly profits when

there is information about the original goals of the data storage, the process of data collection, artifacts used during data collection, and perceived data quality (reliability, accuracy, and precision of the data) [7].

Just as important as gathering and recording context information, it is to revise and update it every time the data is reused. New data analysis goals, assessed data quality, lesson learned, insights, and obtained models should be recorded whenever possible. We see context information as a live knowledge repository that is born with the collected data, but grows and evolves as the data is used.

2.2. Extracting Context Information from Experts

Unfortunately, the context information associated with data repositories is usually coarse. In many cases, important context information is not coded at all. The most common scenario is that software engineers acquire useful knowledge during the production, collection, preparation, and analysis of data, but this knowledge is not transcribed in an explicit format to future data users. We call this the tacit knowledge problem [11].

It is key to transform as much tacit knowledge as possible into explicit information. Tacit knowledge tends to be lost with time and people relocation.

2.3. Extracting Context Information Through Data Mining

The previous sections talked about directly gathering context information. Frequently, this is not possible. Most of the times, the context information associated with data repositories is insufficient. A solution to this problem is to try to extract part of this context information directly from experts and available data.

Data mining is about extracting useful information from data, so it can help experts to extract context information. It helps them to identify relevant questions about the data. Questions that they would not have thought to ask before.

Next session argues that visualization and interactive data exploration has much to offer to the experts by helping them to explore and better understand the data during a model building process.

1. Our Perspective on Data Mining And Model Building

Figure 1 breaks down data mining activities in three levels: model building, pattern extraction, and interactive data exploration. They can be seen as a pyramid where the exploration of data in the base creates the fundamentals for the extraction of patterns

and construction of models on the top [5].

The aim of interactive data exploration is to help to describe complex information and better understand what is going on in the data. It can be seen as a data-driven extraction of context information. This level helps with data pre-processing, data cleaning, data transformation, data set selection, and identification relevant variables. Techniques based on visualization and visual data mining can be successfully used for interactive data exploration [2,6].

Figure 1 – Data mining and model building.

Pattern extraction aims at identifying useful patterns in data. Techniques like cluster analysis and association discovery helps to identify groups of homogeneous data sets and associations between relevant variables. They also may have strong impact on data understanding, selection of data sets, and selection of independent variables for model building.

Model building should be the last step. It must be grounded on data understanding. It is the tip of the pyramid and should be supported by interactive data exploration and pattern extraction.

2. Bringing Background Knowledge to Model Building

Figure 2 proposes a model building process that considers context information. The top of the figure shows the process activities and the decision points of this process. The left bottom lists the context information need during the modeling process. As discussed, the sources for this information can be experts´ tacit knowledge, recorded context information (knowledge repository), and mined context information.

The new, generated, context information is seen on the right under the activities. This information includes: modeling goals, assessed data quality, lessons learned, insights, and produced models. This information should be added to a knowledge repository for future usage.

The process activities are as follows:

1) **Problem Definition and Modeling Goals** – beginning of the process. Goals, initial parameters, limitations and expectations are stated.

2) **Obtaining and Understanding Stored Data and Gathering Knowledge** – data is assembled and explored. The available context information about the data is gathered. This information may come from the repository, from data exploration or from the tacit knowledge of software engineering experts. Best modeling techniques are identified.

3) **Assessing Data Quality** – data reliability, precision and accuracy are estimated based on the collected context information. The experts´ judgment has a great influence in the process; they should use the gathered context information to assess data quality and discard data that are not trustworthy.

4) **Model Construction** – the modeling algorithm is applied. This step is dependent on the results of the previous steps. It is interactive and iterative with the others activities. The experts will rebuild the model many times, until he is satisfied with it.

5) **Models Evaluation –** the generated models are evaluated against a testing data set, and measures of the model's precision, error and accuracy are taken. This activity finishes when the experts are satisfied with the produced model. If not satisfied, the experts can backtrack to some of the previous modeling activities.

6) **Conclusions, Recommendations and Packaging** – It consists of describing the model and organizing all context information about the model. This information should be packaged in with all the necessary information for re-creating the model from the same data. All the context information for the correct use of this model should be made available. Important context information may include: (1) data quality assessment; (2) the model sensitiveness, accuracy, precision, limitations, and assumptions; and, (3) information about the model building process.

We have been using our model-building framework over empirical software engineering data. The next two sections present this work, focusing on the use of visual data exploration techniques to bring context information to the model building process.

3. Work Context

In order to illustrate the role of visualization on software engineering model building, we have been using data collected during the Readers Project. The project is a collaborative research effort to evaluate software defect detection techniques through controlled experiments [12,13]. It involves researchers from the Fraunhofer Center MD, the University of Maryland, and several universities in Brazil. The project backbone is the replication of controlled experiments to produce more empirical data about the techniques we want to evaluate.

In this paper, we use data from the Perspective Based Reading (PBR) Experiment [14]. PBR is a family of reading techniques based on scenarios to improve the effectiveness of inspections over natural language requirements documents. The PBR experiment is aimed at comparing the efficacy and efficiency of PBR against Checklist techniques for detecting defects in software requirements documents.
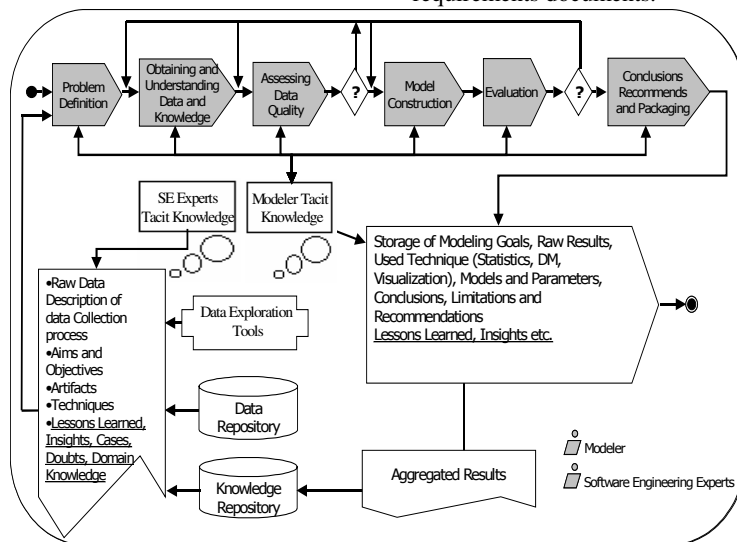

Figure 2 – Model Building Framework

The experimental design of the PBR experiment is shown in Table 1. Subjects were divided into two groups of nine people. Both groups applied Checklist on the

Table 1 – PBR Experimental Design

1º Day	Group 1 – 9 Subjects			Group 2 – 9 Subjects			
	Theory Checklist and Training						
	ATM			PG			
2º Day	3 Designer	3 Testers	3 Users	3 Designer	3 Testers	3 Users	PBR
	Theory PBR and Training						
	PG			ATM			

first day and PBR on the second. The experimental artifacts are two requirements documents describing an Automated Teller Machine (ATM) and the operation of a Parking Garage (PG). The groups applying PBR were divided into three subgroups of three subjects. Each subgroup applied the technique from one of the PBR perspectives, either a Designer (D), a Tester (T) or a User (U). Depending on the perspective, the reader must follow specific scenario in order to uncover software defects. Once the defect is uncovered, the subject must assign a type to the found defect based on the taxonomy of the defects.

The dependent variables of the PBR experiment are: number of defects found, effectiveness, and efficiency. The independent variables are: reviewed document, technique, perspective, experience, defect type, defect class, subject native language, subject experience as manager, developer, tester and analyst, and subject experience using and writing requirements documents.

4. Using Visualization in a Model Building Process

This section shows how we used visual data exploration to support the construction of models following the framework shown in Figure 2. It focuses on building classification models for defect detection effectiveness from the data gathered out of the PBR replications. In particular, it describes the building of decision trees, one of the most popular classification methods [5,6,9].

Although, we could have used different visualization techniques, we focused on treemap visualizations to illustrate our case study. Treemaps is an information visualization technique effective for visualizing hierarchical structures [11]. They are well suited to the problem at hand: to build decision trees. Figure 3 shows a treemap that presents the defects detected during one of the PBR replications. The defects are hierarchically organized by defect class (omission and commission) and defect type (incorrect information, incorrect fact, ambiguous information; extraneous information, miscellaneous). In the example, the defects are also

colored by effectiveness of the subject who discovered it.

The sections below use data from two PBR replications to illustrate how we used treemaps to address the issues raised in Figure 2, namely, better understand the data, assess data quality, build models, and evaluate these models.

4.1. Obtaining and Understanding Data

The data was obtained directly from the replicators,

Figure 3 – A Treemap to Visualize Defects Classes and Types

but we did not have much context information on it. In order to understand the data, three tables were assembled, one for each replication and one combining all data from the two replications. The next step was to use visualization to understand data and pre-select relevant variables. The goal is to identify equivalent variables and relationships between dependent and independent variables. This process is highly interactive and iterative with the quality assessment and model building activities.

The idea is to use visualizations from different perspectives to better understand the data. Figure 3 is a good example of one of those visualizations. It shows the distribution of the effectiveness by the classes and type of defects. The figure points out that there is no strong relationship between class-type of defects and effectiveness, showing that these variables are not critical for the models we want to build.

4.2. Assessing Data Quality

Even when the data is gathered in a controlled environment, software engineering experiments has

Figure 3 – Experience of the Subjects as a Manager

many sources of noise and variation. As with any type of measurement, it is always necessary to assess data quality. In our case study, we could indeed find problems in data quality through data visualization. An example is the coding of the experience of the subjects. These variables were numerical and supposed to be coded in years of experience. We knew that the subjects were undergrads or first year grad students It was improbable that a student could have an experience in testing of 24 years, or an experience as manager of 12 years, as seen in Figure 4. We hypothesized that some of the subjects answered the experience questions in months. Looking at the background questionnaire answered by the subjects, we found out the experience questions were written as: How many years/months of experience do you have as a software manager? Our hypothesis was confirmed.

4.3. Model Construction

The goal of the models we wanted to construct was to study the impact of different context variables on the effectiveness of defect detection during the replications. For the construction of the model, we used an algorithm for building decision trees adapted from Quinlan´s C4.5 algorithm [9]. We implemented a tool to make treemap visualization and user interaction possible during the construction of decision trees [10].

Many trees are built during the model building process shown in Figure 2. For each new tree, we needed to go through the activities of the framework, and evaluate the tree built. The process is highly iterative. There are feedback loops between data

understanding, assessing data quality, model building, and model evaluation. For example, the first trees we built for different replications were very different from one another. We had to step back and understand that this variation was caused by the experience variables we discussed earlier. Visualization helped much with this process.

Figure 4 shows an example of a decision tree expressed as a treemap. It shows in its hierarchy a precedence of information gain for the variables: document used, English language knowledge and technique being used. The importance of the technique for the effectiveness wasn't surprisingly, but the high information gain for the type document was. This motivated us to evaluate if this model made any sense.

4.4. Model Evaluation

Empirical models are usually statistically evaluated, in terms of precision and accuracy, against a testing data set. Our model-building framework proposes to go beyond that and to use context information to understand the model itself.

In our case study, for example, we wanted to understand if indeed the document type was an important variable for the subjects´ effectiveness, and why this was so. To answer these questions, we build the treemap shown in Figure 8.
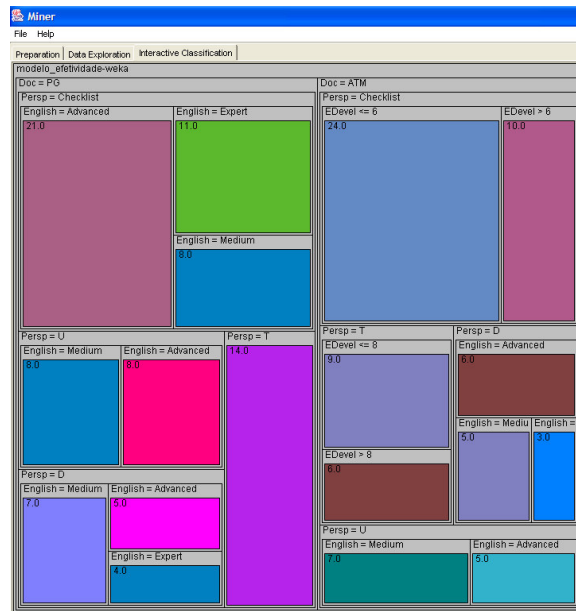

Figure 4 – A Decision Tree Expressed as a Treemap

Figure 8 shows a treemap (not a decision tree) in which the records are hierarchically organized by effectiveness, language knowledge, and document type. Each rectangle represents a defect detection event and is
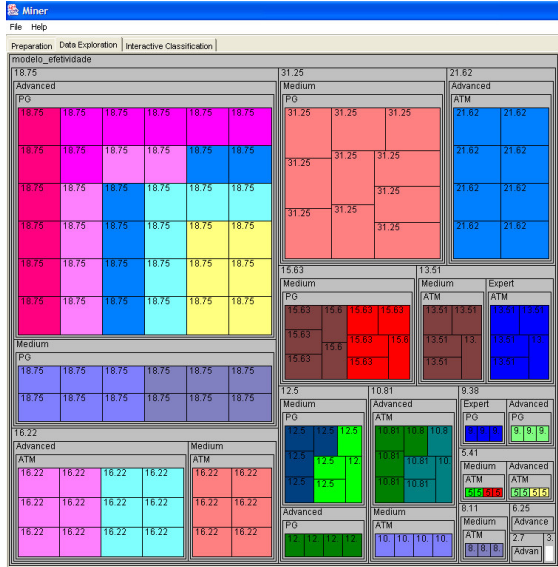
Figure 5 – New Information

colored by the subject ID. Based on the colors, we were able to observe that subjects performed better, in terms of defect detection effectiveness, when revising the Parking Garage (PG) document.

This was surprising information because previous experiments, done in the USA, did not show significant differences on defect detection effectiveness by document type. The experts raised the hypothesis that cultural or language factors play a role in making the ATM document more difficult to Brazilians than to American students.

5. Conclusion

Model building is not a strait process. It should encompass interaction and iterations. A model is as good as the information we use to build it. The systematic association of good context information with models and data repositories facilitates model usage and data understanding. Ideally, a systematic process for knowledge acquisition must be created to support the collection and packaging of context information.

This paper argues that data collectors should record useful context and background information during the data collection process. It also argues that data-modeling efforts generates new context information that should also be recorded. Even when the data itself is not augmented, data repositories should continuously incorporate new context information about data usage and modeling efforts.

However, the paper points out that, even in quasi-ideal scenarios, important context information is usually missing. It then argues that visualization and visual that

exploration have a key role to play in supporting better data understanding and model building. These techniques help experts to bring background knowledge and context information to the model building process.

The paper goes through a case study where treemap visualizations were used to understand empirical software engineering data, assess its quality, and build and evaluate decision tree models from it.

The experience showed that the process is highly interactive with the modeling experts, and highly iterative in the modeling activities.

References

[1] Fayyad U., Piatetsky-Shapiro, P., Smyth. P. The KDD process for extracting useful knowledge from volumes of data, Communications of the ACM, 39(11), pages 27-34, November 1996.

[2] Keim, D. Information visualization and visual data mining. IEEE Trans. on Visualization & Comp. Graphics, 8(1):1-8, Jan 2002.

[3] Khoshgoftaar, T. M., Allen, E. B., Jones, W. D., and Hudepohl, J. P. Classification tree models of software quality over multiple releases. Proceedings of the Tenth Int. Symposium on Software Reliability Engineering, pp. 116-125, Boca Raton, USA, Nov 1999.

[4] Langseth, H. Bayesian Networks with Applications in Reliability Analysis. Ph. D. thesis, Department of Mathematical Sciences, Norwegian University of Science and Technology, 2002.

[5] Mendonça, M. G. and Sunderhaft, Nancy L.. Mining Software Engineering Data: A Survey. A DACS State-of-the-Art Report: http://www.dacs.dtic.mil/techs/datamining/datamining.pdf.

[6] Oliveira, M.C.F., Levvkowitz, H. "From visual data exploration to visual data mining: a survey". IEEE Trans. On Visualization and Computer Graphics, vol 9, nº3, p378-394-jul-set/2003.

[7] Pipino, L., Lee, Y., and Wang, R.."Data Quality Assessment," Communications of the ACM, April 2002. pp. 211-218.

[8] Porter, A. A. & R. W. Selby (March 1990) Empirically Guided Software Development Using Metric-Based Classification Trees. IEEE Software, p. 46-54.

[9] Quilan, J. R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, CA (1993).

[10] Santana, Christiane, Cruzes, D. S., Mendonça, M. G. Visualizing the Construction of Decision Trees Using Treemaps In: International Workshop on Visual Languages and Computing (VLC´2004), 2004, San Francisco.

[11] Shneiderman, B., Johnson, B.: Treemaps: A Space-Filling Approach to the Visualization of Hierachical Information Structures. Proc. of IEEE Information Visualization, pp. 275-282 (1991).

[12] Shull F., Basili V. R., Carver J., Maldonado J. C., Travassos G. H., Mendonca M., and Fabbri S., "Replicating Software Engineering Experiments: Addressing the Tacit Knowledge Problem", In Proc. of Int. Symposium on Empirical Software Engineering, October 2002, pp. 7-16.

[13] Shull, F., Carver J., Travassos, G.H., Maldonado, J.C., Conradi, R., Basili, V.. Replicated Studies: Building a Body of Knowledge about Software Reading Techniques. In Lecture Notes on Empirical Software Engineering, edited by Natalia Juristo and Ana Moreno, World Scientific, 2003.

[14] Basili, V., Green, S., Laitenberger, O., Lanubile, F., Shull, F., Sorumgard, S., Zelkowitz, M.; The empirical investigation of perspective-based reading, *Empirical Software Engineering: An International Journal,* vol.1, no.2, 1996, pp.133-164.

Visual Analysis of Data from Empirical Studies

Rogério Eduardo Garcia
Maria Cristina Ferreira de Oliveira
José Carlos Maldonado
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo
Av. Trabalhador São-Carlense, 400, C.P. 668
CEP 13560-970 São Carlos, SP, Brazil
regarcia,cristina,jcmaldon@icmc.usp.br

Manoel Mendonça
Unifacs – Universidade Salvador
Rua Doutor José Peroba, nº 251, STIEP,
CEP 41770-235, Salvador – BA, Brazil
mgmn@unifacs.br

Abstract

Exploratory visualization techniques may complement statistical data analysis, helping users to understand and treat data from empirical studies. Visualization becomes particularly interesting as data sets grow large and more diverse. This paper discusses how visual representations and exploratory data visualization may be applied to support and enhance analysis of data sets collected in empirical studies on reading techniques applied to requirements documents. Information Visualization techniques are used to analyse data from experiments comparing different reading techniques, allowing us to identify advantages and limitations of visual approaches as compared with traditional statistical techniques.

1. Introduction

In order to successfully attain project goals and deadlines software development teams must choose suitable models and supporting techniques [5]. Empirical Software Engineering attempts to evaluate models and techniques, registering how they perform in practical contexts, with the goal of establishing a knowledge database to support decision making for development. As such, empirical studies aim at providing evidence of the quality and productivity of software development methods, techniques and tools [2, 6, 8, 9].

Experimentation processes are typically conducted to validate previously formulated hypotheses. Hypotheses state an assumption on how dependent variables are influenced by the independent ones. Thus, in an experimental design the independent variables are isolated to investigate if the hypotheses hold, which is usually determined by a statistical analysis. As data is gradually accumulated, it may be difficult to observe non-anticipated relationships and patterns in the data applying only standard statistical techniques. Visual representations may help data analysts to convey information better, and visualization techniques provide an alternative approach to explore data sets produced in empirical software engineering.

In this work visualization techniques have been employed to support knowledge presentation, hypothesis-oriented data analysis (confirmatory visualization) and, to a lesser extent, undirected data exploration (exploratory visualization) of data from experiments comparing different reading inspection techniques applied to software requirements documents. This is part of an effort to identify how visual techniques can be integrated into a systematic data analysis process in this context. We plan to revisit previous work by Carver [4], who proposes a systematic approach for building hypotheses both top-down, using knowledge available in the literature, as well as bottom-up, using experimental data, to study the relationship between an inspector's characteristics and his or her effectiveness in an inspection. Our interest is to investigate how visual representations may assist the hypothesis formulation process. We illustrate such uses on data collected in two experiment replications conducted in the scope of the Readers Project [10], that congregates Brazilian and American researchers on an effort to produce and integrate a large body of results from controlled experiments on families of technologies.

This paper is organized as follows: in Section 2 we discuss work on the empirical studies on reading techniques for reviewing requirements documents, and describe the experiment whose data is used to illustrate the potential role of visualization in data analysis; in Section 3 we present a brief overview of Information Visualization; in Section

4 we describe how Visual Data Analysis has been applied both in hypothesis-driven and exploration-driven data analysis. In Section 5 we provide a discussion and perspectives for further work.

2. Empirical Studies on Software Requirements Analysis: A Case Study

Researchers from the Empirical Software Engineering Group (ESEG) at the University of Maryland have conducted experiments observing the application of several reading techniques, aimed at evaluating and comparing their efficacy and efficiency. In particular, Basili et al. [2] proposed Perspective Based Reading (PBR), which consists of a family of reading techniques for detecting defects in software requirements documents. PBR provides a process to review requirements documents in which the reviewer assumes one particular perspective: *Designer* (D), *Tester* (T), or *User* (U). Depending on the perspective, the reader must follow specific guidelines to conduct the revision process. Experiments have thus been designed to compare PBR against other reading techniques, particularly with the Checklist approach for defect detection. Several experiments have been replicated in the scope of the Readers project, producing data on PBR [10].

The original experimental design of the PBR experiment, run at the University of Maryland, addressed questions such as: *1) Do teams applying PBR detect more defects than teams applying Checklist? 2) Do individual reviewers using PBR and Checklist find more defects? 3) Does the reviewer's experience affect his or her effectiveness?* Shull et al. [10] pointed out that the original experimental design left some open questions, and the experimental design for a set of replications of the original experiment extended the roll of questions to be investigated, such as: *4) Does a reviewer, individually, find out different defects applying PBR and Checklist techniques? 5) Do the PBR perspectives have the same effectiveness and efficiency? 6) Do the PBR perspectives find different defects?*

Following the design specified in the Lab Package, shown in Figure 1, subjects were divided into two groups for the experiment. In the first day, subjects in both groups used the Checklist technique to review one of two requirements specification documents: Automated Teller Machine (ATM) or Parking Garage (PG). In the second day, each subject was trained in one of the PBR perspectives – *Designer*, *Tester* or *User* – to review the document that she/he had not revised yet: those who had reviewed the ATM document applied one PBR perspective to review the PG requirements document, and vice-versa. For each defect occurrence observed the subject should register the page on the correspondent requirement document and classify the defect according to a given taxonomy: *Ambiguous Infor-*

mation (A) – Information is ambiguous; *Inconsistent Information* (II) – Two sentences contradict each other; *Incorrect fact (IF)* – Some sentences assert a fact that cannot be true; *Extraneous Information (E)* – Information is provided, but is not needed or used; *Miscellaneous Defect (MD)* – Other defects; *Omission (O)* – Necessary information has been omitted.

	Group A			Group B			
First Day	Training			Training			Checklist
	ATM			PG			
Second Day	(D)	(T)	(U)	(D)	(T)	(U)	PBR
	Theory PBR						
	Training			Training			
	PG			ATM			

Figure 1. Experimental Design [2]

3. Information Visualization

Visualization techniques can be categorized, according to the type of task supported, in *presentation*, *confirmatory* and *exploratory*.

Presentation techniques assume that facts to be presented are known and fixed *a priori*. *Confirmatory* visualization techniques are useful to support confirmation or rejection of hypotheses about the data held by the analyst – the visualization may motivate (or not) additional statistical analyses on the data. *Exploratory* techniques create representations from raw data, and are useful when there are no *a priori* hypotheses – for example, if the user has little idea of what to search for in the data, or is pre-processing data to improve quality for further input into analytical algorithms and tools. With interactive exploration, a user conducts an undirected search for structures, and gradually forms mental hypotheses that may be confirmed by appropriate visualizations or statistical analysis [1]. These different visualization categories may be applied to data from empirical studies to support different analysis tasks.
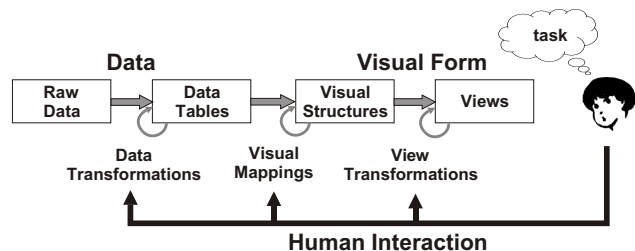

Figure 2. Visual Analysis Process [3]

A typical Visual Analysis Process follows the pipeline described by Card et al. [3], depicted in Figure 2. The pipeline describes the tasks and interactions for producing

a human perceivable, interactive visual representation from raw data. Arrows indicate multiple transformations over the data: once raw data is "cleaned" and instances and relationships organized into a data table, it is possible to create Visual Structures. These combine spatial substrates, marks and graphical properties and may be interactively modified by the user to reach his/her goals. Views from the visual structures are obtained by setting visualization and viewing parameters. One may identify two phases in the process: Data Treatment and Visual Treatment. Although connected, they occur in different spaces (the data space versus the representation space) and, therefore, require different types of support. Flexible interaction with visual representations is a major requirement to explore possibilities and draw conclusions, and such capability must be provided by the visualization tools.

In general, these tools can obtain data from several sources, from text flat files – in standard or custom file formats – to data base management systems. The data collected in the experiment replication and used in this study was summarized in datasheets, which were then converted to text files. Attributes describe subjects' background experience as a software analyst, tester and developer; subjects' previous experience on writing and inspecting requirements documents; the document inspected; the inspection technique; the reported defects and their classification. The resulting Data Table (see Figure 2) – was formatted as required by the different tools used. We used *SpotFire DecisionSite* [1] and *XmdvTool* [11] to generate visualizations for both exploratory and confirmatory analysis, and also a local implementation of Parallel Coordinates for exploratory analysis.

The general process described in Figure 2 was thus instantiated in this context, and results are presented in the next section.

4. Visualization Applied to Empirical Software Engineering: PBR Data

Visualization techniques may assist in different stages of a data analysis process with different roles. One may identify the major stages of a process supported by visual analysis tools starting from Wohlin et al.'s [12] description of a statistical analysis process: a Pre-Analysis stage involves data processing and treatment, including data set reduction when applicable; Hypothesis Verification, which is in the core of the analysis; and two additional stages that may rely on heavy support from visual techniques, Synthesis – which involves presenting results – and Meta-Analysis – which refers to analyzing data from multiple experiments or replications.

Figure 3. Visualization applied in empirical studies

Figure 3 shows our view of the potential application space of visualization techniques in the analysis of empirical data, considering three axes representing the type of user (Designer of the experiment, Replicator or Subject), the stage in the analysis process (Pre-processing, Hypothesis Verification, Synthesis or Meta-Analysis), and the nature of the tasks to be supported (Presentation, Confirmation, Exploration). The front plane, for example, shows the potential usefulness of visual techniques to an experiment Designer. Exploratory visualization techniques are potentially useful in Pre-processing, Hypothesis Verification and Meta-analysis, and Visual Presentation techniques are useful in the Synthesis stage. Visual Presentations may also convey information to Subjects and Replicators, as well as to the Designers themselves. The size of the circles drawn in the figure reflects our feeling regarding the potential usefulness of visual representations for a specific combination of analysis stage, task nature and type of user in the analysis process. Thus, we believe, for example, that visualizations are likely to be very useful to Designers (front plane) in several stages, for example for conducting Confirmatory and Exploratory tasks in the Analysis and Meta-analysis stages, whereas experiment Replicators (middle plane) are likely to benefit a lot from Presentation techniques in the Synthesis stage and from Confirmatory visualizations in the Analysis, though they are usually not concerned with Meta-Analysis. The back plane shows that Subjects might resort to effective visual Presentation techniques, for example, to understand results and evaluate their own performance (Synthesis tasks).

In the above scenario, visual analyses conducted in this study concentrate mainly on the front and middle plane regions, considering the experiment Designers' and Replicators' perspectives. Analyses were conducted with two dif-

Figure 4. Defects found by Technique and Defect Type (R1)

ferent goals. First, confirmatory visualizations that could actually support qualitative "hypothesis" confirmation or rejection were produced. Here, interactive visualizations were the main resource to identify and group the attributes of interest – identified as the dependent and independent variables defined in the experimental design to visually "show" how the data behaves regarding a certain hypothesis. Once the relevant information is identified, it may be conveyed by a proper visual presentation organized to show the target (dependent) attribute in relation to the secondary (independent) ones. At this po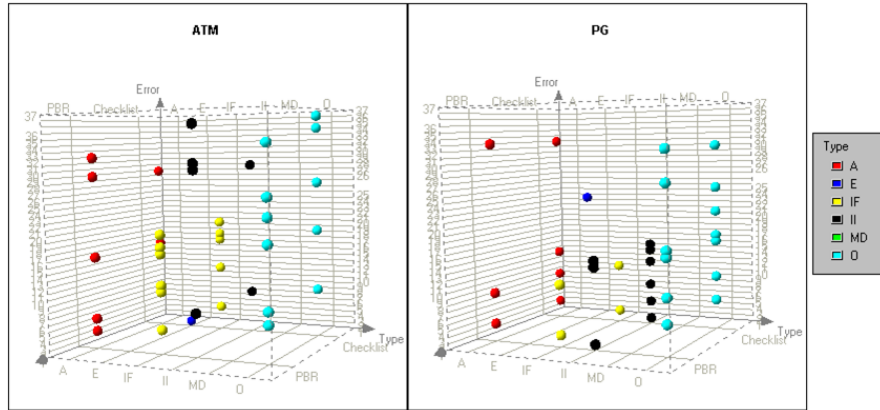int, simple presentations, such as pie and bar charts, usually suffice. Secondly, exploratory analyses were conducted trying to look at the data without directing the search by hypotheses. Undirected exploration was limited, however, as the study was restricted to a small data set from two replications.

4.1. Hypothesis-Driven Visualization

Initial visualizations were inspired by the Questions posed by the experimental design, trying to visually identify as trends in a graphical representation the known statistical results. Although data from two replications on PBR experiment have been used, the results presented are from the first replication – the results obtained using data from replication R2 are quite similar. Focusing on Question 2 (*Do individual reviewers using PBR and Checklist find more defects?*), an example of a hypothesis-driven visualization that considers the type of the defects is presented in Figure 4. In this visualization, the horizontal axis (x) depicts defect type, the vertical axis (z) represents the defect identification numbers, and the reading technique is on the y axis. This visualization allows observing the distribution of reported defects and their types, comparing the behavior of both PBR and Checklist techniques. One may observe, for example, that defects of the type *Omission* (O) have been found more frequently than other types and that very few defects of type

Extraneous Information (E) have been found.

Figure 5. Defects detected by Document and Technique – color maps average Experience (R1)

Focusing on Question 3 (*Does the reviewer's experience affect his or her effectiveness?*), the visualization in Figure 5 allows observing the role of reviewers' previous experience and background on performance. Subjects' experience as software *Developer*, *Manager*, *Tester* and *Analyst*, plus how much experience *Using Requirements Documents* and *Writing Requirements Documents* were considered. Figure 5 shows the number of defects detected by each group, for each Document and Technique, with the group´s average experience level mapped to a grey scale: white represents the minimal experience coefficient, varying gradually to black for the maximal experience coefficient, as shown in the figure. This figure shows that the less experienced group had a significant improvement when applied PBR.

Question 6 (*Do the PBR perspectives find different defects?*) is concerned with the defects revealed by the different PBR perspectives. If we handle Checklist as an additional perspective we may compare the number and type of defects detected by the four perspectives. Figure 6 shows

a scatterplot: the x axis shows the subjects and the y axis shows the defects, identified by their number in the PG document. One observes that defects in the beginning of the document (low identification numbers) were detected by a greater number of subjects for any perspective and any technique. This suggests a verification of the experimental process, as it seems that even with enough time for each review session, subject's effectiveness decrease with time. However, before considering investigating this issue any further, a quick look at the distribution of defects in the documents (Figure 7) shows that defects are indeed concentrated in the beginning of the documents for both documents. Thus, this result may not be due to subject behavior. To verify the influence of defect distribution the Lab Package may be evolved, making feasible the use of requirements document with different defects distribution.

Figure 7. Defect distribution in documents

4.2. Exploratory Data Visualization

Although limited within the scope of data from two experiments, we investigated how visualization could assist an undirected exploratory process on the data, separately. In this context, we tried to produce visualizations capable of conveying potentially interesting and unknown relationships amongst independent experimental variables. Figure 8 shows a Parallel Coordinates [7] visualization of two PBR replications data (R1 and R2), focusing on the attributes that register subject's previous experience as *Manager*, *Developer*, *Analyst*, *Tester*, *Using Requirements* and *Writing Requirements*. The range of values for these four attributes was normalized and experience values are expressed in months.

Figure 8(a) shows that 14 out of 18 subjects have exactly the same experience on *Using Requirements* and *Writing Requirements*, and four of the subjects have quite similar experience in general. The same is not observed at Figure 8(b): there are no straight correspondence among these experiences (*Using Requirements* and *Writing Requirements*), but it is possible to keep this idea considering the small variance among them. It might be interesting to verify if this holds in other replications, as this information might be considered for attribute reduction in future replications. Carver [4] has discussed some specific skills that may affect software development experience, suggesting a study about their particular influence on subject's effectiveness. A visual analysis may help to decide whether it is worth to conduct further statistical analysis on all the experience metrics collected from subjects. For example, it may not be worth to consider separately the experiences on *Using Requirement* and *Writing Requirement*.

5. Conclusions and Further Work

This work is an initiative towards introducing visual techniques in the process of analyzing data from empirical

Figure 6. Defects detected by subject (R1)

(a) First Replication (R1)

(b) Second Replication (R2)

Figure 8. Experience of Subjects

studies in software engineering, using as a vehicle experiments conducted to evaluate reading techniques applied to requirements documents. We observed that introducing visual techniques in the process is not immediate. Training on techniques and domain knowledge are required to avoid creating naïve presentations, that may induce mistaken observations and conclusions. Knowledge about the experimental design is fundamental for a proper selection of attributes and attribute combinations for visual presentation. Although visual techniques do not substitute statistical analysis, our case study illustrates that visualizations may improve interpretation of the experimental data.

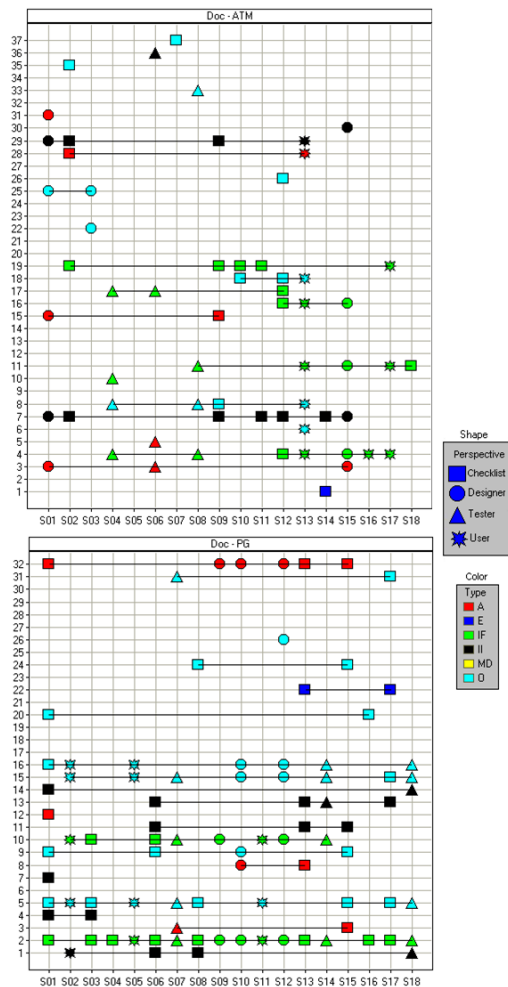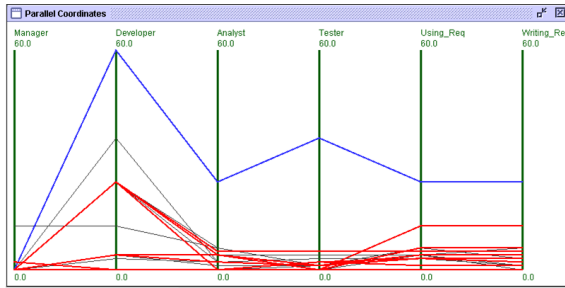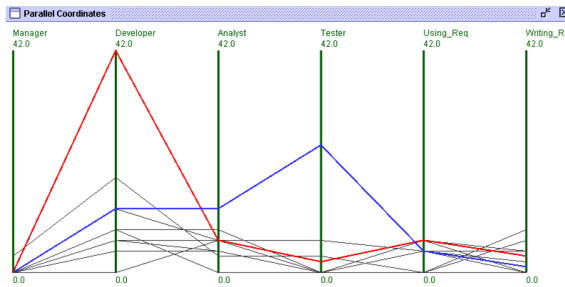Exploratory visualization can assist not only in hypothesis verification, but also in observing trends in data. In both situations it may avoid repeated statistical analysis that may not be worth running, as some facts and trends can be observed with visual interactive techniques and multiple graphical representations. At this point, we believe that visual analysis may be useful to support systematic investigations on data sets to identify sets of relevant hypotheses, such as the one conducted by Carver [4]. Visual representations bring agility and flexibility to data analysis. Also, possible enhancements to the Lab Package were identified, such as the need to investigate the impact of defect distribution through document on subjects' performance.

The major effort to conduct a meta-analysis on the data collected in a set of – possibly evolving – experiments is quite likely to benefit from visual analysis. As further work,

we are investigating how visual data analysis can support meta-analysis and deal with the difficulties introduced by different data organizations. Also, new hypotheses might be proposed for future experiments using exploratory visualization and visual data analysis.

Acknowledgments

References

[1] M. Ankerst. Visual data mining, 2000. Ph.D. thesis, University of Munich, published by www.dissertation.de.

[2] V. Basili, S. Green, O. Laitenberger, F. Lanubile, F. Shull, S. Sorumgard, and M. Zelkowitz. The empirical investigation of perspective-based reading. *Empirical Software Engineering: An International Journal*, 1(2):133–164, 1996.

[3] S. Card, J. MacKinlay, and B. Shneiderman. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers, San Franciso, CA, 1999.

[4] J. C. Carver. *The Impact of Background and Experience on Software Inspections*. PhD thesis, University of Maryland, 2003.

[5] CeBASE. NSF Center for Empirically Based Software Engineering, 2004. (URL:http://www.cebase.org/www/home/index.htm – Access: Jan/2004.

[6] P. Fusaro, F. Lanubile, and G. Visaggio. A replicated experiment to assess requirements inspections techniques. *Empirical Software Engineering: An International Journal*, 2(1):39–57, 1997.

[7] A. Inselberg and B. Dimsdale. Parallel coordinates: A tool for visualizing multidimensional geometry. In *Proceedings of IEEE Visualization '90*, pages 360–375, Los Alamitos-CA, October 1990. IEEE Computer Society Press.

[8] E. Kamsties and C. M. Lott. An empirical evaluation of three defect-detection techniques. In W. Schäfer and P. Botella, editors, *Proceedings of the Fifth European Software Engineering Conference*, pages 362–383. Springer–Verlag, 1995.

[9] A. A. Porter, L. G. Votta, and V. Basili. Comparing detection methods for software requirements inspections: A replicated experiment. *IEEE Transactions on Software Engineering*, 21(6):563–575, 1995.

[10] F. Shull, J. Carver, G. H. Travassos, J. C. Maldonado, R. Conradi, and V. R. Basili. *Empirical Software Engineering*, chapter Replicated Studies: Building a Body of Knowledge about Software Reading Techniques, pages 39–84. River Edge, NJ, 2003.

[11] M. Ward. Xmdvtool: Integrating multiple methods for visualizing multivariate data. *Proceedings of Visualization '94*, pages 326–333, 1994.

[12] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén. *Experimentation in Software Engineering: An Introduction*. Kluwer Academic Publishers, Boston (USA), December 1999.

Visualizing Software Architecture: a Code Driven Approach

W. Eric Wong[†] and Yu Qi
Department of Computer Science
University of Texas at Dallas
Richardson, TX 75083

Abstract

We describe a method for extracting architecture of a program[1] from its source code. Such architecture is represented by imposing the execution counts computed from the dynamic traces collected from program execution on a static call graph at the function level or a control flow graph at the basic block level. A tool, VSAC, was developed to automate the extraction and present the results graphically. An example of using VSAC on a C program is also reported to explain the use of this tool in detail.

Key words: Software architecture, dynamic execution trace, call graph, control flow graph, execution count, basic block

1. Introduction

A C program[2] can be modeled by a graph with a set of interconnected nodes. Depending on the granularity, each node in the graph can represent a function or a basic block in a function. Two nodes, α and β, are connected if control can flow from α to β. A basic block, also known as a block, is a sequence of consecutive statements or expressions containing no transfers of control except at the end, so that if one element of it is executed, all are. This, of course, assumes that the underlying hardware does not fail during the execution of a block. Hereafter, we refer to a basic block simply as a block.

With this in mind, in this paper the architecture of a program is represented in terms of its static structure such as call graphs or control flow graphs and the execution counts collected from dynamic execution traces. More specifically, the architecture is represented by imposing the execution counts computed from the dynamic traces collected from program execution on a static call graph at the function level or a control flow graph at the basic block level.

Information captured in such an architectural model can help programmers understand and maintain their programs. For example, when a node has an unexpected number of execution counts (such as some code that should not be executed by a test case but is executed), it gives an indication that there might be a program bug in the code which decides the corresponding control flow. Another example is that such information can be very useful in performance analysis by identifying which function(s) or which block(s) and their surrounding structures are responsible for most of the executions.

To automate the extraction and present the results graphically, a tool, VSAC (standing for **V**isualizing **S**oftware **A**rchitecture in **C**ode), was implemented on top of ATAC which is part of the Telcordia Software Visualization and Analysis Toolsuite (also known as χSuds) [1,6]. Given a C program, ATAC instruments it at compile time by inserting probes at appropriate locations and builds an executable based on the instrumented code. For each program execution, ATAC saves the execution traceability in a trace file. With this information, VSAC can compute how many times a function is invoked and how many times a block in a function is executed. Such information along with the static structure of a program (either call graphs at the function level or control flow graphs with respect to individual functions) is then displayed in a user-friendly graphical interface.

The rest of the paper is organized as follows. Section 2 explains the use of VSAC with an example. Section 3 presents an overview of some related studies. Our conclusions and recommendations for future research are in Section 4.

2. An Example on the use of VSAC

In this section, we explain the features of VSAC by way of a running example *eparser* – a simple expression parser which parses and evaluates constant expressions, that is,

---

[†] Corresponding author. Tel: 972-883-6619; Fax: 972-883-2399. *Email address:* ewong@utdallas.edu; *Web URL:* http://www.utdallas.edu/~ewong.
[1]In this paper, we use "program," "application," and "software" interchangeably. We also use "functions" and "procedures" interchangeably.
[2]The method discussed in this paper also applies to other programming languages such as C++ and Java. For illustration and the associated tool development, however, we have to select a specific language -- C in our case.

expressions with no variables. For example, it takes as input a string containing a numerical expression, such as (10 – 5) * 3, and computes the proper answer. The source code for the *eparser* program, which consists of eleven user-defined functions, is contained in two files: *eparser.c* and *elib.c*.

We compile the source code with ATAC, generate an instrumented executable (*eparser*), and collect the static information. Next, we run *eparser* on a test case (say t_1) to generate a dynamic execution trace file (*eparser.trace*). Figure 1 shows the main VSAC window display based on the static and dynamic information we have collected. On the left of the menu bar, there are four pull-down menus: File, Analysis, Graph, and Help. All features of VSAC can be invoked through these pull-down menus except for Summary and Update which are invoked by clicking on the corresponding buttons on the right of the menu bar. There are four additional buttons in the left panel, which allow the user to conveniently invoke a function or block call graph, or zoom in/out the current graph. The window below the zoom in/out buttons displays some general information about the program being analyzed. In our case, it shows that *elib.c* and *eparser.c* are the two functions being analyzed. The window on the right, which is initially blank, is used to display, by default, the function call graph. The number at each node is the execution count of the corresponding function. For example, the number "2" at node *eval_exp2(elib.c)* implies the function *eval_exp2* in file *elib.c* is executed twice. If function A calls function B, there is a directed edge in the graph from A to B and the associated number gives the number of invocations of B by A. For example, there is an edge from *eval_exp2(elib.c)* to *eval_exp3(elib.c)* with a number "4"; this means *eval_exp3* is called four times by *eval_exp2* because of the execution of *eparser* on t_1.

Suppose we run the executable on another test case (say t_2). New trace information will be generated and appended at the end of *eparser.trace*. In fact, VSAC continuously monitors this trace file to see if any new trace information has been added to it. If so, it highlights the Update button in red to indicate this. We can click on the Update button to tell VSAC to incorporate the new trace information from t_2 into its display. Figure 2 shows the updated function call graphs. Compared with Figure 1, we notice that the number at the node *eval_exp2(elib.c)* changes from "2" to "6" which implies the function *eval_exp2* in file *elib.c* has been executed four more times because of t_2. In addition, the number at the edge from *eval_exp2(elib.c)* to *eval_exp3(elib.c)* changes from "4" to

"11"; this means *eval_exp3* is called seven more times by *eval_exp2* because of the execution of *eparser* on t_2.

Figure 1 The call graph at the function level after executing t_1

Figure 2 An updated call graph at the function level after executing t_2

In Figures 1 and 2, only part of the call graph is displayed in the VSAC window because the graph is too large for the display window. We can use the zoom out feature to reduce the size of the graph. Figure 3 shows a reduced graph of the one displayed in Figure 2. On the other hand, we can use the zoom in feature to focus only on a small enlarged portion for a better view.

In addition to displaying the execution counts directly on function call graphs these counts can also be listed in a tabular format. An example of this appears in Figure 4 which lists the execution counts of all the functions in the *eparser* program.

Figure 3 A reduced call graph after zooming out the one displayed in Figure 2.

Function Name	File Name	Execution Counts
eval_exp	elib.c	2
eval_exp2	elib.c	6
eval_exp3	elib.c	11
eval_exp4	elib.c	16
eval_exp5	elib.c	16
eval_exp6	elib.c	16
atom	elib.c	12
serror	elib.c	0
get_token	elib.c	32
isdelim	elib.c	32
main	eparser.c	2

Figure 4 Execution counts of functions in the *eparser* program

The listing in Figure 4 can be sorted by "Function Name," "File Name," or "Execution Counts". For example, Figure 5 shows the list sorted in an ascending order by the "Execution Counts".

Function Name	File Name	Execution Counts
serror	elib.c	0
main	eparser.c	2
eval_exp	elib.c	2
eval_exp2	elib.c	6
eval_exp3	elib.c	11
atom	elib.c	12
eval_exp6	elib.c	16
eval_exp5	elib.c	16
eval_exp4	elib.c	16
isdelim	elib.c	32
get_token	elib.c	32

Figure 5 The sorted list with respect to the execution counts

Similar to call graphs at the function level where each node represents a function (such as Figure 1 and Figure 2), we can generate a control flow graph (or a call graph at the basic block level, namely, a block call graph) for each function with each node representing a block and each edge indicating a possible control flow from the block at the head of the edge to the block at the tail of the edge. Figure 6 shows the control flow graph for function *eval_exp2* in *elib*.c. Same as before, the number at each node or edge is the execution count of that node or edge. From this figure, we make the following observations:

- If the cursor is moved onto a node, the source code of that node will be displayed. For example, the code *while ((op = *token) = = '+')* is displayed when the cursor is at B_1.
- The number at node B_2 is 8 which means block B_2 is executed eight times.
- The numbers at the edge from B_2 to B_4 and from B_2 to B_3 are 2 and 6, respectively. This implies of the eight executions of B_2, two are followed by an execution of B_4 and six by B_3.
- The number at node B_4 is 5 which is larger than 2 (the number at the edge from B_2 to B_4). This is because B_4 can also be executed after the execution of B_1. In our case, three of the eleven executions of B_1 are followed by an execution of B_4. As a result, the total number of execution counts for B_4 is 5 (the sum of 2 and 3).

Figure 6 The control flow graph for function *eval_exp2* in *elib*.c. The text next to B_1 gives the corresponding source code at B_1.

Click on the Summary button and you will see the execution counts for every block in this function as displayed in Figure 7.

Block Name	Execution Counts
B0	6
B1	11
B2	8
B3	6
B4	5
B5	5
B6	5
B7	2
B8	3

Figure 7 Execution counts of each block in *eval_exp2* in *elib.c*

3. Related Studies

Several tools and methods have been developed to identify software artifacts in the subject program and organize them into higher level abstractions. One example is ManSART [2,3], a software architecture recovery tool, which automatically recovers architectural features from source code by using data flow, control flow, and static program slicing. In the context of large-sized applications with multiple languages, ManSART dec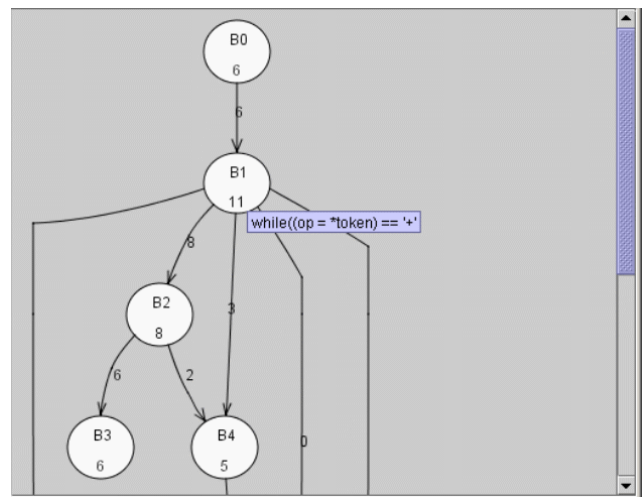omposes the source code according to different languages or modules, invokes component analysis for each decomposed piece, and then interprets and integrates the results of localized, language-specific source analyses. One significant problem of this approach is that it is entirely based on the static analysis of the program without including any dynamic information from its execution.

Another example is Rigi [5] which has a similar goal as ManSART for understanding large software systems through their architectures. It semi-automatically extracts, manipulates and analyzes program artifacts through a distributed, multi-user repository. It also allows software analysts to identify the relations among different pieces of code and integrate these pieces based on their experience and domain knowledge of the program being analyzed. As a result, Rigi can be used to help programmers understand legacy software. However, the process for such an understanding makes heavy use of human cognitive abilities.

A third example is an approach proposed by Chris Pal [4] for extracting and visualizing the dynamic component level interactions obtained at runtime. For a given program, it uses Rigi to extract the static structure, runs the compiled program in a debugging mode with pre-defined breakpoints in each function, analyzes the file generated by debugging call stack traces for each breakpoint, and finally determines a set of call relationships between procedures. The major disadvantage of this approach is that such call relationships are incomplete unless all possible scenarios are considered which might not be possible for a software system without appropriate documentation.

4. Conclusion and future research

In this paper we propose a way to construct an architectural abstraction of a program in terms of its static structure (call graphs or control flow graphs) and execution counts collected from dynamic execution traces. A tool, VSAC, was implemented to support this process. The extracted architecture is displayed in a graphical interface with zoom in and out features. The benefit of using this abstraction model in program debugging and performance analysis is also discussed. An important future study is to explore how the architecture extracted by using our method complements other architectures which are also extracted from the source code. Another study is to apply VSAC to real-life context applications to examine how it can help programmers understand and maintain their programs.

References

1. H. Agrawal, J. R. Horgan, J. J. Li, and W. E. Wong, "Mining System Tests to Aid Software Maintenance," *IEEE Computer*, 31(7):64-73, July 1998.
2. M. P. Chase, S. M. Christey, D. R. Harris, and A. S. Yeh, "Recovering software architecture from multiple source code analyses," in *Proceedings of the ACM SIGPLAN-SIGSOFT workshop on program analysis for software tools and engineering*, pp. 43 - 50, Montreal, Quebec, Canada, July 1998.
3. D. R. Harris, H. B. Reubenstein, and A. S. Yeh, "Recognizers for extracting architectural features from source code," in *Proceedings of the Second Working Conference on Reverse Engineering*, pp. 252 - 261, Toronto, Ontario, Canada, July 1995.
4. C. Pal, "A technique for illustrating dynamic component level interactions within a software architecture," in *Proceedings of the 1998 Conference of the Centre for Advanced Studies on Collaborative Research*, pp. 134-146, Toronto, Ontario, Canada, November 1998.
5. K. Wong, "*The Rigi User's Manual - Version 5.4.4*," Department of Computer Science, University of Victoria, Canada, June 1998.
6. χSuds User's manual, Telcordia Technologies, 1998.

PDAGraph: Event-Driven, Visual Scripting for Handheld Computers

Shea Armstrong
saarmstr@cs.uwaterloo.ca
University of Waterloo
Waterloo, Ontario, Canada

Yael Kollet
kollet@cs.dal.ca

Trevor J. Smedley
smedley@cs.dal.ca

Dalhousie University
Halifax, Nova Scotia, Canada

Abstract

This paper describes PDAGraph, an event-driven, component-based visual programming language for "power users" of Personal Digital Assistants (PDAs). PDAGraph will give users the ability to create customized applications, taking advantage of existing PDA applications and hardware modules. It uses a visual programming language derived from Prograph but adapted for non-professional programmers and for use on a small screen. We describe our initial design here and then discuss how we expect to further develop our design to prepare for user testing.

1. Introduction

With the ability to put more and more computing power into a smaller package, computer hardware companies started in the early 1990's to develop handheld computers, such as Apple's Newton and Palm's Pilot. These devices are now commonplace, and are likely to become even more widely used than desktop computers. The widespread use of desktop computers created a demand from end users for customisability and programmability: it is clear that there is a similar demand from users of handheld devices. Already, there are simple tools such as spreadsheets and database products for handhelds, that allow some level of end user programmability.

We discuss here the approach we are taking in researching the provision for end-user programmability of PDAs, our current progress in our work in this area, and a discussion of our future directions. We have previously reported some preliminary discussion of this work [2].

2. Related Work

There are currently a variety commercial products which allow end-user customization of PDA's running the Palm OS. HotPaw Basic [5], Quartus Forth [10] and OrbWorks Pocket C [9] are all textual programming languages which operate under the PalmOS. Creating useful programs with these products on the PDA is tedious as all text must be entered using Graffiti (the simplified script used with handwriting recognition on the Palm OS). Furthermore, it is impossible to view much of the code at any one time due to the small screen size. This is to say nothing of the problems end users have in creating useful programs using general purpose textual languages such as C and BASIC even without the restrictions of a handheld device.

Very little academic research has been done regarding customization and scripting of PDA's. Hyperflow [6] is a visual programming language developed by Dr. Dan Kimura designed for use by children on a pen-based (but not strictly a hand held) computer system. Hyperflow is a dataflow-based visual language intended for a range of software development. Hyperflow was implemented within the PenPoint Operating System [4].

3. Overall Approach

We are taking a component based, event driven approach to providing programming capabilities to end users of PDAs. We define three different types of components. *External Components* provide the equivalent of an Application Programming Interface between PDAGraph scripts and other hardware devices or software applications. These would be created by professional programmers, and loaded as 'plug-ins' in the PDAGraph environment. For example, the developer of a Global Positioning System (GPS) module might create an external component plug-in to provide the ability to access the GPS information from within a PDAGraph script. Similarly, external components would be developed to access things like the ToDo list, Memo and Calendar data maintained as part of the PalmOS.

User Interface Components hold collections of forms (the PalmOS equivalent of a window). Elements on these forms can be defined to supply information to scripts, take information from scripts, and/or trigger the execution of scripts. Forms in UI components are created by the end user, on the PDA, and are re-usable.

Finally, *Script Components* contain the graphical code created by the end user to provide the functionality they desire. Script components, like UI components, are reusable, as we will see in our example in Section 4. Scripts are written in a data-flow language which evolved from Prograph [8]. We have made adaptations from Prograph, since it is a general purpose programming language aimed at professional programmers using desktop computers, and PDAGraph is a domain-specific language aimed at end users for use on handheld devices.

An *Application* is simply a set of components, which are bundled together, and which could be complied into a stand alone application.

3.1 Execution Model

Our execution model is based on events. Events may be system, user, or program generated (they may come from external, UI or script components respectively), and are referred to as *Triggers*. All execution is performed in
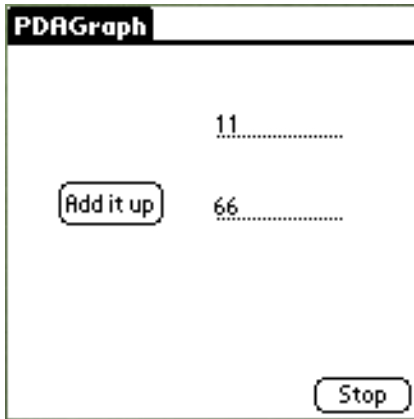
Figure 1 — Application Interface

Figure 2 — Triggered Method

response to triggers. We have special methods, called *Trigger Methods* which are configured to respond to them, and operations called *Triggerring Operations* which are used to generate triggers. We are still exploring a number of issues relating to the processing of triggers. For example, if, in the process of executing a script in response to a trigger, another trigger is generated, is the first processing paused while the new trigger is processed, or should the new one be queued for later processing? Furthermore, if responses to triggers are queued (and there must be some queuing, since more than one method can respond to a single trigger), then is the queue processed in LIFO or FIFO order, or perhaps something else altogether? We are working on designing user tests to determine the answers to these and related questions, and will report on this work at a later date.

4. Example

We will illustrate the current status of our PdaGraph work using an example. The example is a program (with a user interface) to add up the numbers from 1 to n. We will show three different ways of solving this problem with PdaGraph. First a recursive solution, and then using two different forms of iteration. All of the figures shown are screen captures from our current implementation.

In Figure 1 we see the interface for the application (running in our interpreter). The user enters a value in the top text area, presses the button, and the result is displayed in the lower text area.

In this form, the input text area is defined as *askable* (an item from which information can be extracted in a script), the output text area as *tellable* (an item to which information can be provided in a script), and the button as a *trigger* (as discussed in Section 3.1 this can be used to cause the execution of a script).

In Figure 2 we see the triggered method that is executed in response to pressing the "Add it up" button. This uses an *ask operation* called "input" to get the value from the input text field, calls a method called "sum" to do the com-

putation, and then deposits the result into the output field of the form using the *tell operation* at the bottom.

In Figure 3 we see the body of the "sum" method, which simply uses a conditional operation to compute the sum. In Figure 4 we see the three parts that comprise the conditional operation. The leftmost of these is used to define the test for the conditional operation. The result of the test is fed into the small output bar at the bottom of the pane, and this determines which of the two other panes is executed. If the result is True, the centre of the three conditional panes is used, supplying 1 as the output. If it is false, the rightmost pane is executed, which takes the input, subtracts 1 from it, calls the "sum" method recursively to calculate the sum for n-1, and then adds on the input to produce the desired result.

In Figure 5 we see another implementation of the body of the "sum" method, this one done using a definite iteration control structure. The pane on the left shows the body of the sum method, with the call to the looped operation. The remaining two panes show the two parts of the iteration structure. The first of these, in the centre of the figure, shows the controls on the iteration, and the specification of the inputs and outputs to the body of the loop. In this case we see that the loop will start at one, step by

Figure 3 — Sum Method

Figure 4 — Conditional Recursive "sum" Method

one, and continue to the value input to the looping operation. The left input to the body of the iteration will be the index of the iteration. The right input will initially be 0, and then, on each subsequent iteration, the output from the body will be looped around to this input. When all iterations are complete, the output of the looping operation will be the last value output from the body. Finally, on the right in this figure we see the definition of the body of the loop, which simply adds the current index to the cumulative sum so far.

In Figure 6 we see another implementation of the "sum" method, this one using an indefinite iteration. Again, the pane on the left shows the sum method definition with the call to the looping operation. The remaining two panes show the definition of this looping operation. The pane in the centre of the figure shows the specifications of the indefinite iteration, and the inputs and outputs to the body of the loop. This indicates that the first output from the body will be used to test for continuation of the loop. The remaining two outputs are paired with the two inputs, indicating that the outputs of these will be looped around to the corresponding inputs for the following iteration. For the first iteration of the body, the left input to the body will be the input to the looping opera-

tion, and the right input to the body will be 0. Finally, the rightmost pane shows the contents of the body. Note that this is more complicated than in the previous figures, since this particular computation lends itself well to recursion or definite iteration, but is awkward to express with indefinite iteration. In any case, we see that the left input is used to keep track of which iteration (counting down from the limit) and the right input is used to keep track of the sum so far. When the body is executed, one is subtracted from the left input, and this value is output on the centre output, to be looped back around for the next iteration. This value is also tested against zero, and this value is sent to the leftmost output, to be used to test for continuation of the loop. We also see that the two inputs are added together, and the result sent to the rightmost output, calculating the sum so far.

5. PDAGraph Structure

As discussed above, an Application can be made up of three different types of components: External, Script and User Interface. External components provide the following types of methods that can be called from within scripts: Simple, Ask, and Tell. They also provide triggers. UI Components contain forms, but also provide Ask and Tell methods, as well as triggers.

Figure 5 — Definite Iterative "sum" Method

Figure 6 — Indefinite Iterative "sum" Method

In defining a Script Component, a user can create Simple, Ask, Tell, and Triggered methods. Restrictions on inputs and outputs to the various types of methods, and notes on their usage are shown in Table 1.

Type	Inputs	Outputs	Notes
Simple	Any	Any	A "general-purpose" user defined method
Ask	0	≥1	Used to extract data from another component
Tell	≥1	0	Used to supply data to another component
Triggered	0	0	Execution caused by an event

Table 1: Method Types and Arguments

The body of a method is made up of a number of different operations, each of which may have inputs and/or outputs, connected together with data links. The different types of operations are: Input Bar, Output Bar, Simple, Ask, Tell, Triggering, Conditional and Looping. Restrictions on inputs and outputs to the various types of operations, and notes on their usage are shown in Table 2.

5.1 Ask and Tell Operations and Methods

Semantically, there is no difference between an ask method and a simple method with only outputs. Similarly, there is no semantic difference between a tell method and a simple method with only inputs. However, we feel that conceptually, there is a difference between performing some sort of arbitrary computation, and accessing data. In order to make PDAGraph concepts more closely aligned with end user conceptualisations, we have created separate methods and operations for data access.

5.2 Conditional and Repeated Execution

As illustrated in our example in Section 4 we are using special-purpose operations for conditional and repeated execution. This is in contrast to Prograph, in which any method can contain conditional execution instructions, through the use of method cases and Next Case annota-

Type	Inputs	Outputs	Notes
Input Bar	0	Any	Any method which allows inputs includes an input bar operation. The outputs on the input bar correspond and provide access to the inputs to the method.
Output Bar	Any	0	Any method which allows outputs includes an output bar operation. The inputs on the output bar correspond and provide access to the outputs from the method.
Simple	Any	Any	This type of operation specifies a call to a built-in or programmed method.
Ask	0	≥1	Used to access data provided by a component.
Tell	≥1	0	Used to provide data to a component.
Triggering	1	0	When a triggering operation receives an input of true, it fires it's trigger, which may be used to initiate the execution of triggered methods. Multiple trigger operations with the same name within a component refer to the same trigger.
Conditional	Any	Any	Used to perform conditional execution.
Looping	Any	Any	Used to perform repeated execution.

Table 2: Operation Types and Arguments

tions on operation. Similarly, in Prograph, any operation can be annotated to repeat its execution, with annotations for looping inputs, or for iterating across a collection of objects, and the termination of repeated executions is controlled through a variety of controls. We feel that our approach, although less flexible, will be more amenable to end users than the Prograph approach. In particular, even in the case of infrequent use, a user is 'directed' through the construction of a conditional or repeated execution by following the tabs, as opposed to Prograph, where the various annotations need to be remembered or relearned each time.
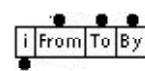
The conditional operation is fairly straightforward, with the first tab being used to compute a boolean test value, which is fed into a small output bar at the bottom of the form with a single output on it. Depending on the result of this test, either the second (true) or third (false) tab is executed. We are also considering a more general form of conditional execution, where the value fed into the test result operation can be any primitive data, and then multiple tabs can be created, each with an associated value. The tab to be executed would be the one with the value which matches the value fed into the test result operation. A default tab could also be created. Alternatively, we could have pairs of test/computation tabs followed by an (optional) default tab, much like chained if-then-else statements in traditional languages.

The iteration operation is more involved. It is made up of two tabs, one used to define the type of iteration, the controls on the repetition, and the inputs and outputs to the body of the iteration. The other tab defines the computation that is to be carried out iteratively.

In the first tab, the programmer chooses between definite iteration ("Counter") and indefinite iteration ("Conditional"). In the case of definite iteration, a control structure is included in the dataflow diagram, . This structure accepts inputs for the start ("from"), stop ("to") and step ("by") of the iteration, and provides an output ("i") that gives the index of the current iteration, which can be connected as an input to the body of the iteration. In the case of indefinite iteration, a different control structure is included in the dataflow diagram, . This control takes a single input, which is connected to an output on the body. The control stops the execution of iterations of the body when the value becomes False.

With both definite and indefinite iteration the first tab also include specifications to the inputs and outputs of the body. These can be either simple (or) or looped (or). A simple input indicates that the same value will be provided to the body of the iteration each time the body is executed (with the exception of a simple

input connected to the output on the definite iteration control structure, which provides the index of the iteration). A simple output, when connected to the output bar of the tab, will provide the value computed by the last iteration of the body to the output of the iteration operation when it completes execution. Looped inputs and outputs always occur in matched pairs. On the first iteration, the value connected to the looped input is provided to the execution of the body. On subsequent iterations the value output on the looped output is cycled around, and provided as input to the next iteration of the body. If the looped output is connected to the output bar, then after the last iteration, the final value on the looped output is output from the looped operation.

6. Concluding Remarks

We have reported on our continuing progress in our work on providing end user programming capabilities to users of handheld computers. We are not yet at the point where we are able to perform user testing, but we are looking forward to the lessons we will learn when we shortly reach that stage in out research. We anticipate that there will be significant changes and adaptations to the work we have shown here, but we also expect that we will see some promise for the continuation of this work.

Besides continuing work on defining and refining the PDAGraph language we are spending considerable effort investigating questions relating to the usability of the system. We use or plan to use a variety of interface techniques to ease the programming task for the end user. For example, rather than writing names of things, the user will usually choose items from lists, and names need only be entered when something is first created. We are also investigating techniques for displaying more information on the screen at one time, from things as simple as automatically abbreviating names and labels, to more sophisticated techniques borrowed and adapted from information visualisation research.

There are two areas where we are particularly active at the moment, as described in the following subsections.

6.1 Automatic Layout

Interaction with the PDA is done primarily using a pen on the touch sensitive screen. As a result, it is imperative that construction of the scripts be tailored to that method of input. One feature which we feel will greatly assist in this area is incremental automated script layout [3]. Traditionally, this is a very difficult graph theory problem; however our domain poses many useful constraints. The small screen size limits the number of objects that can on the screen at one time. Also, it is assumed the user will place each object approximately where he/she feels it should go and as a result only incremental changes will be necessary. We feel it may be beneficial if the movement of objects/ links is done using animation such that the user can easily

follow.

Many factors contribute to aesthetically-pleasing data-flow diagrams. The most obvious being minimization of edge crossings. Other factors which must be taken into consideration are bend minimization, symmetry maximization, edge length uniformity, and graph area minimization. A tension exists between these factors in that the optimality of one often conflicts with the optimality of others. Proper design and implementation of this portion of the system would ultimately result in increased user efficiency through elimination of tedious time consuming tasks in rearranging graphs on the screen.

We have made some initial progress in this area [1].

6.2 Message Flow

For simplicity of use there is a need in PdaGraph for a high-level message-flow view that will allow the end-user to see how different components of the users' programs connect and interact, to design the program, and to create the different components. The components connect to each other through message-flow links. Graphical user interface items (such as buttons and text fields) can be generated automatically after the user defines the message flow between a form inside a UI component and any other component.

Visually, the high-level view of the program will show the user the different components as icons. UI components and scripting components will have distinguishable icons, and each external component will be uniquely identified. Two components that interact with each other will be connected by a message-flow link. The user will be able to decide visually where the execution of the program starts.

From this level of view, the user can add more components and add links between components. The user can also view the content of components by clicking on the icons, and access a detailed list of messages that flow between the two components by clicking on the message-flow link. As the scripting component is the environment where all the computing and assignments are done, there cannot be message flow between external and UI components without passing through a scripting component.

In general messages can flow from:
- UI component to a scripting component
- Scripting component to UI component
- External component to scripting component
- Scripting component to external component
- Scripting component to scripting component

The message-flow link is just a possible link between the two components and can have zero or more actual messages flowing between the two, linking methods and UI objects.

The use of UI components and scripting components as a library for forms and methods is probably too complicated for beginners and might even be time consuming for power users to learn. Thus there is a need to have a simpler high-level view in order to help novice users to learn PdaGraph quickly. The simplification of the UI and scripting components can be achieved by having a UI component as one form and a scripting component as one method. This will help a novice end-user to navigate through components in a new environment, as writing visual languages can be an acquired skill.

We have made some initial progress in this area, including a user study of this aspect of the system [7].

7. References

[1]   S. Armstrong. Incremental Automated Layout of Dataflow Diagrams on Small Screen Devices. Dalhousie University Honour's Thesis. 2002.

[2]   S. Armstrong, Y. Kollet and T. Smedley, Visual Scripting for Handheld Computers. In Proceedings of IEEE 2002 Symposia on Human Centric Computing Languages and Environments. 2002.

[3]   K.F Bohringer, Frances Newbery Paulisch. Using constraints to achieve stability in automatic graph layout algorithms. SIGCHI: ACM Special Interest Group on Computer-Human Interaction, 43-51, 1990.

[4]   Carr, R. and Shafer, D. The Power of PenPoint, Addison-Wesley, 1991.

[5]   HotPaw Software for PalmOS. Retrieved March 7, 2002, from http://www.rahul.net/rhn/hotpaw/

[6]   T. D. Kimura. Hyperflow: a uniform visual language for different levels of programming. Proceedings of the 1993 ACM conference on Computer science 1993, Indianapolis, Indiana, United States, pp 209-214

[7]   Y. Kollet. Message-Flow Programming in PdaGraph. Dalhousie University Master's Thesis. 2004

[8]   Pictorius Incorporated. (1993) *Prograph CPX User's Guide.*

[9]   PocketC for PalmOS. Retrieved March 7, 2002, from http://www.orbworks.com/PalmOS

[10] Quartus Forth (Palm OS Version). Retrieved March 7, 2002, from http://www.quartus.net/products/forth

Visualizing Intra-procedural Data-flow Interactions to Help Locate Faults

Marcel R. Karam

American University of Beirut, Lebanon

Trevor J. Smedley

Dalhousie University, Halifax, Canada

Abstract

One of the most expensive and resource-consuming tasks in the testing cycle is locating faults. To do so, testers must identify suspicious source code elements such as statements, controls, and variables involved in failures. This paper presents a new technique that visualizes tested source code elements to assist with these tasks. Our visual modeling technique uses visual artifacts and colors to represent the participation of each source code element in the outcome of the execution of a function under test, or intra-procedural testing. Based on this visual mapping, a user can inspect and identify source code elements involved in failures, and potentially locate faults. This paper also describes a tool that implements our technique.

1. Introduction

Reducing the number of faults in software systems is clearly desirable. Among the tasks required to reduce the number of faults in software development is locating errors. This is a resource consuming task that can have a significant impact on the cost of the delivered software. Inevitably, techniques and tools that can automate the process can provide significant savings, and evidently reduce the time required to locate faults. This can have a significant impact on the overall cost and quality of the software development and maintenance cycle.

Observations made by Spafford et.al. [8] suggests that one of the tasks of locating faults begins with the activity of identifying program statements involved in failures. Developers, in an attempt to locate faults, use tools to manually trace the program, and with a particular input, encounter a point of failure, and then backtrack to try to find suspicious program elements and potential causes. With large programs and large test suites, the huge amount of data produced by such an approach, if reported in a textual form, may be difficult to interpret, and very time consuming [8].

Herold et.al [5] implemented a visualization technique that provides a global view of the results of executing a program with an entire test suite. The technique uses color to visually map the participation (pass and fail) of each program statement in a test suite. Based on this visual mapping, a user can inspect the statements in the program, and identify potential faulty statements. Our approach, while similar in the use of a visual mapping scheme and color, focuses on providing more complex mapping of a source code elements (statements, control, and data-flow interactions) in individual functions. To further assist testers in locating faults in a function under test, our technique employs a scheme that visually maps, not only the statements of a function, but rather all function elements, including control and dataflow interactions. Visually representing static data-flow interactions of a function under test is particularly complex due to the overwhelming number of data flow interactions, especially in the presence of loops.

Our technique uses visual modeling artifacts that are interactive. That is, after a test case is applied, these visual artifacts are colored to reflect, based on a particular code-based testing criterion (All-nodes, All-edges or All-du chains)[†], the testedness of the source code elements of a function under test. The decision to use colors as an indication of testedness in our technique/system is inspired by the work in Burnett et. al. [3]. In that work, empirical studies showed that colors are very effective in reflecting the testedness of form-based programs.

One attempt to visualize statements and controls of a function under test was introduced in Combat [6]. The visual *CFG* supported by Combat represented blocks as numbered circles, and control between blocks as edges with arrows. Static data-flow interactions were not however visually represented.

This paper presents the details of our visual mapping technique that solves the visual mapping of statements, controls, and data-flow interactions of a function under test. The paper also presents a description of a tool, Visual Testing Environment for C, (VTEC), that implements the technique. Our new testing technique, as we will show, helps to quickly identify and locate untested elements, especially when these elements correspond to complex dataflow interactions.

The remainder of this paper is organized as follows: in Section 2 we briefly describe both control-flow and dataflow-testing methodologies for imperative languages, and discuss related work. In Section 3 we discuss, in the context of a C example, our visual modeling technique. In Section 4 we describe VTEC, the prototype that implements the technique, and demonstrate, using an example, its effectiveness in visually identifying and locating untested program elements and their corresponding portions of the source code. Finally, in Section 5 we conclude with some directions for future work.

2. Background and Related Work

Most test adequacy criteria for imperative languages are defined in terms of abstract models. The most commonly used abstract model is the Control Flow Graph (CFG). Con-

---

† The All nodes, All-edges, and All du-chains testing criteria are explained in Section 2

trol flow analysis in a *CFG* focuses on how conditional code constructs explicitly reflect how the program execution paths follow the control flow in the graph. Data flow analysis in a *CFG* focuses on how variables are bound to values, and where these variables are to be used. A path in a *CFG* from a variable definition to its use is commonly known as a definition-use chain or du-chain.

Two practical code-based testing methodologies are available for imperative languages: control-flow testing; and data-flow testing. Control-flow testing focuses on traversing all nodes (statements) or all edges (controls) in the *CFG*, while data-flow testing focuses on traversing some or all du-chains. Traversing all du-chains is also known as All-dus. Two kinds of uses are important to All-dus: computational use or *c-use*; and predicate use or *p-use* [4]. Given a definition of a variable v in a block or node $b_i \in CFG$, we say that a node $b_j \in CFG$ contains a computational use or *c-use* of v if there is a statement in b_j that references the value of v. We also say that a block $b_k \in CFG$ contains a predicate use or *p-use* of v if the last statement in b_k contains a predicate statement where the value of that variable is used to decide whether a predicate is true for selecting execution paths. Each block b_i that contains a predicate statement is referred to as a predicate block.

2.1 Prograph's Visual Dataflow Environment

The visual artifacts we use to construct the source code elements of a function under test, is based on a subset of Prograph's [7] visual constructs. We next informally introduce the visual syntax subset of Prograph that is of interest to our current work. Figure 1 depicts the following operations: *roots, terminals, input bar; output bar; local;* and *multiplexed-locals*. The *input bar* represents a method's input parameter list. The *roots* (implicit variable definitions) on the *input bar* represent the formal parameters of the method. As depicted in Figure 1, *r1* represents the formal parameter on the *input bar*. The *output bar* represents a method's output parameter list. The *terminals* (implicit variable uses) on the *output bar* represent the parameters returned by the method. As depicted in Figure 1, t_1 on the *output bar* represents the parameter returned by the method. In visual dataflow languages such as Prograph, variables are not explicitly defined. That is, users write programs by creating icons that are connected via datalinks. In general, a datalink is created between a *root* on one operation, and one or more *terminals* on another. In Prograph, *roots* serve as vari-

Figure 1—A subset of Prograph.

able definitions and *terminals* connected to those *roots* via a *datalink* serve as variable uses. As depicted in Figure 1, the *root* r_2 on *L3* is connected to the *terminal* t_2 on *L4*.

A *local* operation is analogous to a parametrized *begin-end* block in a standard procedural language. As depicted in Figure 1, "*L3*" represents a *local* operation. Control can be applied to most operations in Prograph. An operation can be annotated with *next case on success* or *next case on failure* controls. The *next case* control is analogous to a conditional statement in textual languages. Once triggered, depending on the condition being checked, execution of the current case immediately terminates, and execution is initiated in the next case. As depicted in Figure 1, *L1* and *L2* are annotated with *next case on success* and *next case on failure*; respectively.

Local operations in Prograh can be annotated with a *repeat-control*. The result is a *multiplexed-local* operation that is analogous to a *while* or *for* loop in imperative languages. As depicted in Figure 1, the *local* operation *L6* is annotated with a *repeat-control*. A *local* or *multiplexed-local* operation, when double clicked, opens up a new window with its own *input bar* and *output bar* to represent the input and output to and from the operation. *Roots* and or *terminals* on most operations in Prograph can be annotated with a *loop-control*. The result is a *multiplexed-local* with a *wrap-around link*. A *wrap-around link* is a *datalink* that starts at the *root* of a *multiplexed-local* and is wrapped around the latter to enter into the appropriate *terminal*. As depicted in Figure 1, r_3 on *L5* is annotated with a *loop-control*. The *wrap-around link* on *L5* is formed by wrapping r_3 and t_3 with a *datalink*.

2.2 Related Work

Harold et. al. [5] have presented a technique that uses statement coverage or All-nodes to aid in the task of fault localization. Their technique visually maps each source code statement in a program to a short, horizontal line of pixels. This "zoomed away" perspective lets more of the software system be presented on one screen during testing. Their technique is implemented in a tool called Tarantula. Tarantula colors statements in a program to show their participation in passed and failed test cases. Our work, while similar in the use of a visual mapping technique and color, "zooms in" on all source code elements (statements, control and data-flow interactions) of a function under test. That is we visually map each source code element using visual artifacts to expose more knowledge of the source code. These visual artifacts are then appropriately colored to reflect their testedness according to testing criteria such as All-nodes, All-edges, and All-dus. This "zoomed-in" approach allows testers to apply more stern testing criteria such as All-edges and All-dus, and at the same time *zoom in* on the testedness of the visual representation of the source code elements. As a consequence, faults have a higher chance of being uncovered and visualized.

In other related work, Burnett et. al. [2] implemented an abstract model structure, Cell Relation Graph (CRG) to test form-based visual languages. The *CRG* made it possible to apply a variety of testing criteria. Later work by Burnett et. al.
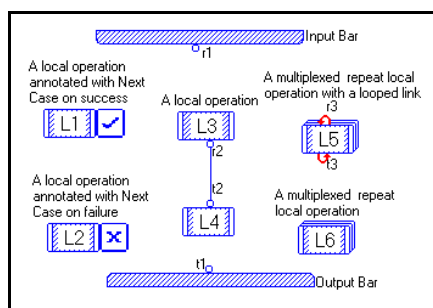
[3] proposed the use of visual elements and colors to communicate to the user the testedness of cell formulas under a particular test adeqaucy criterion. Our work parallels that in Burnett et. al. [3]. That is, we make use of visual elements/artifacts to represent source code elements of an imperative function under test, and use colors to communicate the testedness of each element.

3. Our Visual Modeling Technique

Given a function F in a program Q under test, let $CFG_{(F)}(V, E)$ be the control flow graph corresponding to F, such that V and E are the set of nodes and edges $\in CFG_{(F)}$, respectively. Our visual modeling technique in VTEC takes a $CFG_{(F)}$ and produces a Visual-$CFG_{(F)}$ or V$CFG_{(F)}(N, C, R, T, D)$ such that: N is the set of visual nodes (*locals* or *multiplexed-locals*); C is the set of visual controls *(next case* annotations) that may be applied to a subset of N; R is the set of visual *roots* representing variable definitions in the $CFG_{(F)}$; T is the set of visual *terminals* representing variable uses in the $CFG_{(F)}$; and D is the set of visual *datalinks* connecting a subset of R to a subset of T, representing the data flow interactions in the $CFG_{(F)}$.

Next, we illustrate, in the context of the C example that is depicted in the window labeled "main" in Figure 3 and its $CFG_{(C)}$ in Figure 2, the construction process of the V$CFG_{(C)}$ that is depicted in windows "main" and "n3" in Figure 3. The C program of Figure 3 computes the square root of a number between 0 and 1 to an accuracy e, such that $0 < e <= 1$.

3.1 Visual Modeling of the Control Flow

After we perform static control flow analysis on a $CFG_{(F)}$, we build a $VCFG_{(F)}$ in VTEC as follows:

For each $CFG_{(F)}$, and each *multiplexed-local* operation, we construct the appropriate visual *input bar* and *output bar* in the V$CFG_{(F)}$. For each non predicate block $b_i \in CFG_{(F)}$ we construct a *local* $n_i \in N$. For example the *local* labeled *n2* in the $VCFG_{(C)}$ of Figure 3 corresponds to block $b2$ in the $CFG_{(C)}$ of Figure 2.

For each predicate block $b_i \in CFG_{(F)}$, we construct a *predicate-local* $n_i \in N$ in the V$CFG_{(F)}$. A *predicate local* n_i is a

Figure 2 — A C example and its CFG.

local that is annotated with a ☑ on its right-hand side, and a ☒ on its left-hand side, to represent the predicate block's true and false edges or outcomes, respectively. For example, the *predicate-local* n_1 in the $VCFG_{(C)}$ of Figure 3 corresponds to block b_1 in the $CFG_{(C)}$ of Figure 2.

For each block $b_i \in CFG_{(F)}$ representing the entry of a loop, we construct a *multiplexed-local* $n_i \in VCFG_{(F)}$. For example, the *multiplexed-local* n_3 in the $VCFG_{(C)}$ of Figure 3 corresponds to the loop that starts at node b_3 in the $CFG_{(C)}$ of Figure 2. As depicted in Figure 3, the window labeled "n3" contains the set $<n_4, n_5, n_6>$ of *locals* $\in N$ in the V$CFG_{(C)}$ of Figure 3 that corresponds to the set $<b_4, b_5, b_6>$ of blocks in the $CFG_{(C)}$ of Figure 2.

3.2 Visual Modeling the Data-flow Interaction

After we perform static data flow analysis on a $CFG_{(F)}$, we map the data-flow interactions to a $VCFG_{(F)}$ as follows:

For every formal parameter f_i in $CFG_{(F)}$ we construct a *root* $r_i \in R$ on the *input bar* of the $VCFG_{(F)}$. The set of *roots* $R= <r_1, r_2,..., r_n> \in VCFG_{(F)}$ corresponds to the set of formal parameters $F= <f_1, f_2,..., f_n> \in CFG_{(F)}$. F and R are both sorted in an ascending alphabetical order, and placed from left to right on the *input bar* of the $VCFG_{(F)}$.

For every formal parameter f_i that reaches live (without encountering a redefinition), the end of the method, we construct a *terminal* t_i on the *output bar* of the $VCFG_{(F)}$. We use standard data flow analysis and methods to determine whether a formal parameter f_i can reach the end of a method before a redefinition of f_i is encountered. With imperative languages, a definition of a variable v reaches a point p in a subroutine S if there is an execution path from the definition to p along which v is not redefined or killed [1].

For every formal or actual variable definition v_j in $b_i \in CFG_{(F)}$, that reaches, live, the end of b_i, we construct a *root* r_j on the *local* n_i. For example, in the $CFG_{(C)}$ of Figure 2, the definition of X in b_1 reaches, live, the end of b_1 and thus we construct a *root* on n_1 of the $VCFG_{(C)}$, as depicted in Figure 3.

For every formal or actual variable definition v_j in b_i that reaches, live a use(s) in a block other than the one it was defined in, we construct a *terminal* t_j on the *local* $n_i \in N \in VCFG_{(F)}$. For example, in the $CFG_{(C)}$ of Figure 2, variable X is defined in b_1 and reaches, live, via the path (b_1, b_3, b_7), a use in block b_7. Thus, we construct a *terminal* t_i on n_7, as depicted in the $VCFG_{(C)}$ of Figure 3.

Let P be the set of complete paths in a $CFG_{(F)}$. We model the data flow interactions between variables definitions and their uses in a $VCFG_{(F)}$ as visual *datalinks*. To construct these visual *datalinks* in VTEC, we define a family of variable uses in a $CFG_{(F)}$: direct use; extended use; and wrap-around

Figure 3 — VTEC, a C example containing an error (left) and its VCFGs (right).

use. Each variable use in this family can be either a *c-use* or a *p-use*.

We say that a direct use from $(b_i$ to $b_j) \in CFG_{(F)}$ corresponds to an execution path $M \subset P$ iff the definition of a variable v_d in b_i reaches, live, a *c-use* or a *p-use* of v_d in b_j, and b_j is the executional successor of b_i. A direct use $\in CFG_{(F)}$ is constructed in VTEC as a direct *datalink* D_L in the $VCFG_{(F)}$ that connects a *root* $r_d \in R$ to a *terminal* $t_d \in T$. For example, in the $VCFG_{(C)}$ of Figure 3, we construct a D_L between the *root* labeled X on n_1 and the *terminal* on n_7.

We say an extended use from node $(b_i$ to $b_k) \in CFG_{(F)}$ corresponds to an execution path $M \subset P$ iff (1) the definition of a variable v in b_i, $b_i \in$ to a block representing a loop L_B, and v reaches, live, a *c-use* or a *p-use* of v in b_k, such that b_k exists outside the static scope of L_B, and (2) the definition of a variable v in b_i, b_i exists outside L_B, and v reaches, live, a *c-use* or a *p-use* of v in b_k, such that $b_k \in L_B$. Therefore, an extended use $\in CFG_{(F)}$ is constructed as a extended *datalink* E_L in the $VCFG_{(F)}$ in two cases: (1) a definition inside the body of loop that reaches, live, a use outside the loop, or (2) a definition outside the loop body that reaches, live, a use inside the body of the loop. For example, in the first case, the definition of X in b_6 of the $CFG_{(C)}$ of Figure 2 reaches, live, a

use in b_7 via the path $\{b_6, b_3, b_7\}$. Thus, we construct in the $VCFG_{(C)}$, as depicted in Figure 3, an E_L from the *root* labeled X on n_6 to the *terminal* on the *output bar*, and from the *root* labeled X on n_3 to the *terminal* on n_7. The *terminals* on the *output bar* of a *multiplexed-local* represent ports through which definitions of variables inside the loop reach uses either outside the loop or back in the loop via the loop edge. In the second case, for example, the definition of D in node b_1 of the *CFG* in Figure 2 reaches, live, a use in node b_6 via the path $\{b_1, b_3, b_4, b_6\}$. Thus, we construct, as depicted in $VCFG_{(C)}$ of Figure 3, an E_L from the *root* labeled D on n_1 to the *terminal* on n_3, and, as depicted in Figure 3, the link is extended from the *root* labeled D on the *input bar* of Figure 3 to the *terminal* on the n_6.

We say a wrap-around use from node $(b_i$ to $b_j) \in V$ corresponds to an execution path $M \subset P$ iff the definition of a variable v in b_i, $b_i \in L_B$, and v reaches, live, a *c-use* or *p-use* of v in b_j, such that $b_j \in L_B$, and i possibly equals to j. A wrap-around use $\in CFG_{(F)}$ is constructed in VTEC as *wrap-around datalink* $W_L \in VCFG_{(F)}$. Thus, a W_L is constructed for every variable defined inside the body of a loop that reaches, live, uses back in any block inside the loop via the loop edge. We construct a W_L by: (a) constructing a D_L from the *root* where

the variable is defined to the appropriate *terminal* on the *output bar* of the *multiplexed-local*; (b) constructing a *loop-link* ⬍ on the D_L constructed in (a); (c) wrapping the link around the *multiplexed-local*, and into the appropriate *loop-terminal*; (d) constructing D_L from the appropriate *root* on the *input bar* of the *multiplexed-local* to *terminals* that are associated with the uses; and (e) constructing *loop-links* on the D_L(s) constructed in (d). If the D_L(s) that are to be constructed in (d) already exist, we simply skip the construction process of (d), and apply (e). A *loop-link* is constructed to differentiate between the definition of a variable (outside the body of the loop) that reaches, live, *c-uses* or *p-uses* directly inside the loop, and a redefinition of the same variable (inside the loop body) that reach, live, *c-uses* or *p-uses* via the loop edge.

To illustrate, consider the $CFG_{(C)}$ of Figure 2. The definition of X in b_1 reaches, live, c-uses in b_4, and b_6. Thus, as depicted in the *VCFG* of Figure 4, one E_L is created from n_1 to n_4 and another E_L is created from n_1 to n_6. Also, in the $CFG_{(C)}$ of Figure 2, the definition of X in b_6 reaches, live, via the loop edge, *c-uses* in b_4, and b_6. Thus, as depicted in the $VCFG_{(C)}$ of Figure 3, we (a) construct a D_L from the *root* that is associated with X in n_6, to the appropriate terminal on the *output bar* of the *multiplexed-local* n_3 (b) construct a *loop-link* on the D_L that was constructed in (a), (c) wrap the link around the *multiplexed-local* n_3, and into the third right-hand side *loop-terminal*, (d) skip the construction process since the $D_{L(s)}$ that are associated with the uses in n_4 and n_6 already exist, and (e) construct the appropriate *loop-links* on the already constructed D_L(s). This completes the construction process of the $VCFG_{(C)}$.

The set of visual *locals*, *predicate-locals,* and *multiplexed-locals* $N = <n_1, n_2,... n_m> \in VCFG_{(F)}$ and *datalinks* (D_L, E_L, and W_L) can be directly manipulated by the user. That is, a user can double click on any n_i and view the source code that corresponds to $b_i \in CFG_{(F)}$. The user can also double click on any *datalink*, and view, in the source code, the variable associated with that particular data flow interaction. The layout of *locals*, *multiplexed-locals*, and *datalinks* are movable in VTEC to give the user more visual flexibility and control during a testing session.

4. Prototype

We have built a visualization system VTEC, that implements the visual mapping described in Section 3. VTEC is written in Java. It takes as an input a program, and generates the $VCFG_{(F)}$ for each function $F \in P$. VTEC supports three code-based testing criteria: All-nodes; All-edges; and All-dus. Test cases can normally be extracted based on either the specification or the logic of the source code; however, in the current implementation of VTEC we manually create test cases. Users can right click in the source code window of a function

F, or the window displaying the generated $VCFG_{(F)}$, and choose a testing criterion and input the test cases. Once a chosen test case is run, VTEC reflects, based on a particular testing criterion, the testedness of the visual artifacts of the $VCFG_{(F)}$. We next show how these visual testing feedbacks can aid the user in visually locating either statement(s), control(s) or data flow interaction(s) responsible for untraversed source code elements.

4.1 An Example

To best illustrate a testing session in VTEC, we test using all three supported testing criteria the erroneous C program that is depicted in Figure 3. This program computes the square root of a number between 0 and 1 to an accuracy e, such that $0 < e <= 1$. The error in the program lies in lines 19 and 20 that should be interchanged. The test suite results of Table 1 is used here to give, in tabular form, the testedness of the C program of Figure 3 under all three supported testing criteria in VTEC. As, depicted in Table 1, the third test case results in a 100% coverage of All-nodes and All-edges, but only 98% All-dus. The All-dus criterion is not satisfied since the path (b_6, b_3, b_4, b_6) is not traversed with regards to the definitions of X and C in b_6. This means that the definitions of X and C in b_6 do not reach their uses over the loop edge into statements 19 and 20, respectively.

4.2 Reflecting the Testedness With Color

Under All-nodes, after any test case T in a test suite TS is executed, VTEC colors green each *local* $\in VCFG_{(C)}$ if its corresponding block $b_i \in CFG_{(C)}$ is traversed, otherwise red.

Under All-branches, after any test case T in a test suite TS is executed, VTEC colors the *next case* visual annotations ([✓] and [✗]) of each *predicate-local* or *predicate-multiplexed local* $n_i \in VCFG_{(C)}$ as follows: (1) we color ([✓] and [✗]) green to reflect that the true and false edges that are associated with the predicate block $b_i \in CFG_{(C)}$ are traversed; (2) we color [✓] green and [✗] red to reflect that the "true" edge has been traversed; (3) we color [✓] red and [✗] green to reflect that the "false" edge has been traversed; and (4) we color [✓] and [✗] both red otherwise. For example, after the third test case in the test suite of Table 1 is executed, VTEC, as depicted Figure 4, colors the *next case* visual annotations [✓] and [✗] on the *predicate-locals* n_1, n_3 and n_4 green. The G and R labels in Figure 4 represent the Green and Red colors, respectively. We annotate the visual artifacts with G and R appropriately to reflect the testedness, since colors are not supported in the final printed version of Figure 3 or Figure 4.

Table 1: All Nodes, All-edges and All-dus test suite.

P	E	output	Nodes	Edges	Dus
1	--	invalid domain	28.5 %	12.5 %	0.0 %
0.9	1	0.0 "wrong result"	28.5 %	12.5 %	12.0 %
0.9	0.004	0.625 "wrong result"	100.0 %	100.0 %	98.0 %

Figure 4 — The visual reflections of tested and untested du-chains.

Under All-dus, after any test case T in a test suite TS is executed, the *datalinks* (D_L, E_L or W_L) that are associated with a traversed du-chain are dealt with as flows: (1) if the *datalink* corresponds to a du-chain whose variable definition v in $b_i \in CFG_{(F)}$ reaches a *p-use* in $b_j \in CFG_{(F)}$, VTEC colors the appropriate *datalink* in the $VCFG_{(F)}$ fully green to reflect that the true and false edges that are associated with the predicate block $b_j \in CFG_{(C)}$ are traversed, one green and one red to indicate that only one edge has been traversed, and red otherwise; (2) if the *datalink* corresponds to a du-chain whose variable definition v in $b_i \in CFG_{(F)}$ reaches a *c-use* in $b_j \in CFG_{(F)}$, VTEC colors the appropriate *datalink* in the $VCFG_{(F)}$ fully green to reflect that the du-chain has been traversed, and red otherwise.

For example, in the $CFG_{(C)}$, and under the All-dus testing criterion, no amount of further testing or test cases would result in traversing the definition of C and X in block b_6 that reaches uses in block b_4. To reflect these untested du-chain in the $VCFG_{(C)}$, we color in red, as depicted in Figure 4, the *loop-links* of the W_L(s) that start at the *roots* that are associated with the definition of C and X at n_6, wrap round n_3, and end at the first and third left-hand *terminals* on n_6, respectively. All other *datalink*s will be colored green to indicate that they have been traversed.

4.3 Locating the Variable(s) Associated With Untraversed Du-chains

To help VTEC users locate the variable involved in untraversed du-chains, the user can simply double click on the red *locals*, *predicate-locals*, or *datalinks*, and the environment will take the user into the source code where the suspected statements, controls, or variable du-chains be highlighted. This activity is analogous to that of modern *IDE*s where a user can double click on reported complier error and the *IDE* takes the programmer to the suspected line in the source code where a syntax error has occurred. Our testing system parallels that in providing the user/tester with visual clues as to where in the source code, an untraversed statement, control, or du-chain has occurred.

5. Conclusion and Future Work

We have implemented a system/technique that solves the problem of incorporating and visually representing both control-flow and data-flow characteristics of a function under test. We have demonstrated that these visual artifacts can reflect, according to a particular testing criterion, a program's testedness in a way that gives the user/tester visual clues to untraversed source code elements, and in turn help locating faults.

We are currently in the process of preparing a usability study to empirically further investigate the effectiveness of this system. Furthermore, the study will look into determining a benchmark in relation to other existing testing tools.

6. References

[1] Aho, A. V; Sethi, R; Ullman, J. D., *Compilers, Principles, Techniques, and Tools.* Addison-wesley, Boston, MA, 1986.

[2] Burnett, B; Li, L; Rothermel, Gregg., Testing Strategies for Form-Based Visual Programs. *IEEE International Conference on Software Reliability Engineering,* pp. 96-107, November 1997.

[3] Burnett, M; Dupuis, C; Li, L; Rothermel, G., What You See Is What You Test: A Methodology for Testing Form-Based Visual Programs. *IEEE International Conference on Software Engineering,* pp.198-207 1998.

[4] Frankl, P; Weyuker, E. J., A Data Flow Testing Tool. *In ACM Softfair Proceedings* Dec. 1985. ACM, New York, 46-53.

[5] Harrold, M. J.; Jones, J.A.; Stasko, J., Visualization of Test Information to Assist Fault Localization, *ACM International Conference on Software Engineering,* Orlando, Florida, U.S.A., May 2002.

[6] Harrold, M. J.; Priyadarshan K., Combat: A Compiler Based Data Flow Testing System.

[7] Shafer Dan, *The Power of Prograph CPX*, The reader Network, 1994.

[8] Spafford, H. E.; Pan,H.; and DeMillo, R. A. Failure and fault analysis for software debugging. In Proceedings of COMP-SAC 97, pages 515-521, Washington, D.C., August 1997.

Towards Comprehensible Control Flow in Visual Data Flow Languages

Anthony Cox
amcox@cs.dal.ca

Simon Gauvin
gauvins@cs.dal.ca

Trevor Smedley
smedley@cs.dal.ca

Faculty of Computer Science, Dalhousie University, Halifax, Canada

Abstract

The use of graphics is a common technique for providing programmers with added information for program comprehension tasks. However, the development of visual programming languages has provided opportunities for programmers to use graphics directly within programming languages. In this paper we investigate the use of control flow notation within visual data flow programs. Our investigation is centered around the comprehensibility of boolean expressions that control the execution of data flow and how variations in these notations can effect the comprehension of programs. We report on a user study of visual boolean expressions and a design to improve the task of program comprehension.

1. Introduction

The collection of conditionals, and their evaluation to determine program behavior at runtime, is a significant aspect of the control flow of a program. Understanding of control flow is central to program comprehension [20] and is often cited as the first cognitive task programmers perform when trying to understand a program [3]. Reading and comprehension of a set of conditionals results in the construction of higher semantic abstractions that can aid in understanding of a program's behavior. To create these higher forms of understanding a programmer is required to evaluate the numerous conditionals within source code.

Program maintenance and evolution accounts for more than 90% of the total cost of software [23]. Of this percentage, 50-90% of that time is simply spent understanding previously written source code [5]. A significant aspect of understanding source code is understanding the control flow of the program. Since conditionals are a major aspect of the control flow of a program, if conditionals can be evaluated more quickly, the time needed to understand a program will be reduced. If conditionals are difficult and time consuming to evaluate programs will be difficult to comprehend and time consuming to maintain.

One of the promises of visual programming languages (VPL) is to provide opportunities for reducing programming time by making code faster and easier to comprehend. Specifically VPLs are able to do this by providing a closer mapping between the problem domain and the expression of the domain solution using a VPL notation [2][26]. This mapping is, however, not automatic and must follow a set of clear guidelines to improve comprehension [13]. The visual nature of the VPL can provide affordances to programmers seeing code for the first time, thus supporting comprehension and maintenance tasks [16][19]. Because program comprehension is a difficult and time consuming task, we would like to aid in comprehension by providing a representation that integrates control and data flow. In the current phase of this work, we examine the reported difficulties of the present control flow notation of the Prograph data flow language [6] to examine programmer's ability to understand conditional expressions. We report on an experiment that compares the present Prograph-style control flow notation to an alternative notation. In this experiment, we use time spent in understanding code fragments as a measure of the ease of understanding of control flow notations. We discuss related work on boolean expressions from a cognitive and programming perspective and then present the visual control flow notations that are the subject of our experiment. This is followed by the presentation of the experiment, its results, and some concluding remarks.

2. Related Work

Our work intersects two areas of research, program comprehension and visual programming. As conditionals are evaluated on the basis of boolean logic we first explore the cognitive issues with regard to their comprehension. We then examine visual techniques that have been used to program boolean expressions.

2.1 Boolean Expression Evaluation

A Boolean concept is simply a cognitive mapping between elements of a domain and the boolean values of true or false. Evaluation of Boolean concepts was examined by Posner when testing subjects about the truth of a simple statement "Is the star above the cross" [21]. This sentential evaluation of a conditional expression was performed after showing subjects a picture of a star and a cross. Posner found that participants performed more rapidly when the statement was posed in the positive form than in the negative, such as "Is the star not above the cross." Posner explained this behavior by observing that cognitive functions evaluate Boolean concepts in the positive form and then perform a second stage to evaluate the meaning through negation. From experience we know that code that is expressed in terms of a negated expression is harder to comprehend and thus it is regarded as poor programming in practice [12].

Novice users of Boolean expressions generally do not understand AND, OR, and NOT constructs. The terms

'and' and 'or' are used in natural language differently than in formation of query logic, thus errors are made when using knowledge of English [12]. A common mistake was to substitute AND for OR when translating an English sentence to a query. The NOT Boolean operator is often misinterpreted in terms of scope due to ambiguity [12]. Combinations of Boolean logic, where evaluation of an expression is based on several sub-evaluations, is more difficult to comprehend since sub-expression results need to be held in working memory while other sub-expressions are being evaluated.

2.2 Visual Boolean Expressions

Michard [18] attempted to represent Boolean expressions graphically using Venn diagrams to show operations on sets. Each attribute of an expression is expressed as a circle where users form queries by pointing at portions of the intersecting circles. In a user study of twelve college students he found that subjects performed better using the graphical query interface. Another graphical Boolean interface, based on relative positions of tiles, was developed by Anick et. al. [1].

Visual Boolean expressions for SQL queries have been of interest for some time due to the inherent difficulties in formation of Boolean expressions [4][27]. The Filter/Flow system used data flow as a the base model, with a visual source and result display using graphical links showing the flow of records between query attributes [28]. Using this system participants in a user study made fewer errors evaluating Boolean expressions, in some cases one-third as many errors as compared to SQL.

Program comprehension has been shown to be improved through the use of control flow annotations added to textual programs [7]. Studies have also shown that data flow is harder to understand than control flow [10]. Intuitively this seems natural since we tend to think in terms of sequences of instructions when performing a task rather than the flow of the work that is produced while performing the instructions. However, in cases where instructions are complex or even unpredictable, data flow as a model greatly simplifies the ability to define program behavior [24].

Evidence that data flow is an appropriate notation for programming is the existence of a number of commercially successful VPLs based on data flow including LabVIEW [15], VEE [14], Simulink [8], Softwire [25] and Prograph [6].

However, the specification of control flow within these languages continues to be a problem. In LabVIEW conditionals are built from logic gates that are displayed within the language and connected with links that carry Boolean data. Logic gate notation is extremely well known and, since LabVIEW is targeted to the engineering domain, it would seem natural that such a representation would fit. However, observations by Green [11] suggest that engineers have a difficult time comprehending programs written in this notation.

We also see difficulties with control flow notations in the Prograph family of languages [6][9][22]. These use a particular control flow notation that is specified by visually annotating the source code displayed as a data flow graph. We and our colleagues have observed that first year computer science students being taught programming using Prograph reported having difficulty understanding this particular aspect of the language. Green et. al. [11] also noted that these controls were difficult for test subjects to comprehend. They attributed these difficulties to what they call "hard mental operations." That is, a programmer must stop and carefully evaluate a conditional to in order to understand the behavior of a program. Meyer et. al. [17] reported that the number of control choices also contributed to comprehension difficulties.

3. Visual Control Flow Notation

In this section we present the control flow notation used in the Prograph family of languages. We provide an informal definition and show how the notation is used to build conditionals that control the flow of a data flow program. We also present the alternative notation that we have developed and used in our study.

A Prograph program is made up of operations (built-in functions and operators or calls to user defined functions) connected by datalinks. Data flows between the operations on the links. The execution order of a set of operations is controlled by the arrival of data at the operations — an operation can only execute once all of its data is available.

Although there exist pure data flow languages where the execution of operations is determined only by the availability of data, this is not a practical programming methodology, and visual data flow languages usually include some method of specifying the flow of control. In Prograph this is provided by annotations on operations.

3.1 Controls

In Prograph, a control annotation, or simply control, is a visual annotation attached to the body of an operation. The control is evaluated during execution to determine if the data flow graph of a program will continue after the execution of the operation is complete, or branch to some other sub-graph.

Figure 1 shows examples of some Prograph operations with annotations. Each operation has inputs on the top (terminals) and outputs on the bottom (roots). When an operation executes, the input values are used, output values are calculated, and then propagated along the links connected to the roots. Operations also have annotations, such as those shown on the right hand side of the bar and snafu operations. These annotations are used to control the execution flow of Prograph programs. Besides the two annotations shown here, Prograph also has a variety of other annotations to support repeated execution. How-

ever, these other controls are not discussed here and their examination is left as future work.

When an operation executes, besides returning values for its outputs, it also produces an execution message. This execution message has one of two values, "success" or "failure." It is this execution message that is tested by the control annotations for use in controlling the

Figure 1: Prograph Operations

flow of the program. Conceptually, this is equivalent to having an additional boolean output which can be tested for control flow purposes.

The control annotations on operations are made up of two parts, a trigger state (success or failure) and an action (for example, next case or continue). It should be noted that the absence of an annotation on an operation means that the operation has the control "Continue on Success" which is the default behaviour for an operation.

3.2 Control Trigger State

A control has a trigger state, which has one of two values: success (designated by a checkmark, as shown on the snafu operation in Figure 1) or failure (designated by an X, as shown on the bar operation in Figure 1). In order to determine if a control will "fire" it's action, the execution message from the operation is compared against the control trigger state. If they match, then the action is fired, otherwise it is not. In order for a programmer to figure out the behaviour of an annotated operation, he must first determine the execution message from the operation, and

Result	Operation	Control
Match is true, therefore do control action.	0 0 $=$	✔
Match is false, therefore do not do control action.	0 1 $=$	✔
Match is false, therefore do not do control action.	0 0 $=$	✘
Match is true, therefore do control action.	0 1 $=$	✘

Table 1: Biconditional Variations

then apply a boolean match function (i.e. a function which returns true when its two inputs are the same) to the execution message of the operation and the control's trigger state.

There are four possible combinations of execution message and control trigger state for an annotated operation. The combinations are illustrated in Table 1. These combinations show a sample conditional using the equality operator with two integer inputs. The operation will evaluate to true or false and emit an execution message of success or failure respectively. The match function is then evaluated using the trigger value and the trigger state of the control to determine if the control action is fired. Of the four combinations only the first and last combination result in the action being fired.

3.3 Control Action

The action of a control is the behavior it performs when it fires. There are several different actions that can be used to control program flow. Each action is represented as a unique icon which contains the check or X for the control trigger state and indicates to the programmer what the program will do when the control fires. In our current study, we use only the next case control, illustrated by the

☐ icon. The next case control stops execution of the current case, and executes the first operation in the next case in the sequence of cases of a method or local operation. This provides functionality equivalent to that of an if or case statement in a language such as C.

3.4 Comprehension of Control Flow

As we have shown, a control is composed of two distinct concepts. The first is the control trigger state, represented by icons placed in the center of the control icon. The second is the control action, represented by an icon surrounding the trigger state and forming the "body" of the control that is attached to an operation. The three parts – operation, action, and trigger state – form a conditional statement that evaluates to cause a particular behaviour. To comprehend this conditional statement a programmer must interpret these three parts and the result it produces. It is this evaluation that has been identified as being difficult and that forms the focus of our investigation.

Upon further analysis, we can see that controls consist of two independent Boolean expressions that are tightly coupled. Both of these expressions must be evaluated independently before the "final" result of the control can be determined and the behavior of the expression understood. The first expression is that of the operation itself. The programmer must first evaluate the code to which the control is attached. As we have seen this can be a conditional expression or simply a single operation. In all instances the operation's execution message provides a Boolean result that matches the result of the operations

Result	Operation	Control
Operation returns true, therefore do control action.	0 0 =	
Operation returns false, therefore do not do control action	0 1 =	

Table 2: Unconditional Variations

logical result. Next, the programmer must read the control, with its Boolean trigger state, and compare it against the result of the operation's Boolean state to determine the final Boolean result of the conditional. This design produces what we call a biconditional expression where each evaluation of a conditional in the language requires two Boolean conditional expressions to be evaluated.

3.5 Alternative Notation

We have created an alternative control flow notation which uses a stateless control that has only an action. With this design the resulting program comprehension task requires the evaluation of only a single Boolean expression, i.e. the expression provided by the operation. If the operation evaluates to true the stateless control will fire, otherwise it will not. As shown in Table 2 this control design provides only two possible results compared to the four results possible with biconditional controls of Table 1.

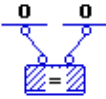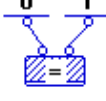The difference presented in the state-ess control is that the result of the operation determines the execution of the control and not the result of the Boolean expression between the operation state and the control state. This means that one of the Boolean expressions required to evaluate the conditional has been removed resulting in a unconditional statement. The icon of the unconditional statement, as a whole, represents the action of the control (to stop) and not its state, since it has none.

4. User Study

In our user study we compare two control flow notations. Participants were asked to complete a single task of evaluating a Boolean expression. The main goal of the study was to determine which of the two control flow notations allowed the participants to perform the task correctly in the least amount of time. However, the study also demonstrates that programmers can effectively understand control flow elements in a predominately data flow representation.

The hypothesis is that there is a difference between the two conditionals and that the unconditional expression will take less time to comprehend than the biconditional expression, and that the subjects' responses will be more accurate using the unconditional notation.

4.1 Participants

The study consisted of 14 male participants. The participants were programmers with 6 with no university training, 7 undergraduates, and 1 with graduate level education. Experience was spread with 1 having <5 years, 11 having 5-10 years, and 2 having >10 years programming experience. Ages were clustered with 12 between 26-35, 1 <18, and 1 >35 years of age. None of the participants had any experience with visual or data flow programming.

4.2 Apparatus & Procedure

The task we were testing was the evaluation of visual data flow conditional expressions. Task test items consisted of viewing a code snippet and pressing a button in response to a question regarding the code. The experiment used the two variations of control design described in Section 3.

The study was delivered to all participants though the Internet via a web browser. The interface consisted of a set of web pages and forms written in PHP and two Java applets embedded within the pages. In performing the study participants navigated through a total of eight pages. Interaction with each page consisted of reading and clicking on controls (buttons, checkboxes, lists) with the mouse. No typing was required. Data was collected by submission of forms and through interaction with the Java applets. This information consisted of participant profiles, test results, and survey results. The use of Java applets to collect timing information minimized the effects of network latency on the collected data.

Participants started at an introduction page that explained the procedure of the experiment. The interaction with the participants was in three phases: demographic questionnaire, first experiment and second experiment.

The first phase consisted of single page that presented a brief questionnaire to obtain consent and collect general demographic information such as age, gender, and years of programming experience. We also included questions pertaining to their knowledge related to the experiment such as if they had previous VPL, data flow, or control flow experience. Preference and experience with textual languages was also collected.

The second and third phases were identical in structure but different in experimental content. The ordering of these two phases was randomly chosen for each participant. Each phase consisted of three web pages that sequentially presented a tutorial, test, and survey. The tutorial consisted of a Java applet that displayed a series of pages of information that contained the content of the tutorial.

The test was also presented in a Java applet within a web page. It consisted of three panels. A top panel presented a question. The center panel displayed a code snippet. The bottom panel displayed a set of three buttons. The procedure in performing the test was to read the ques-

tion in the top panel, evaluate the expression in the center panel, and press a button in the bottom panel in answer to the question posed in the first panel.

Participants started the test by first reading the text in the web page and then clicking a start button on the applet when ready. The text in the web page asked the participants to respond to the test as quickly as possible while maintaining accuracy. The question in all cases for both tests was "What does the following code cause the program to do". The buttons provided to answer this question were labelled "Continue", "Stop", "Don't Know". Upon each press of a button a new code snippet would be presented for to a total of 10 questions (note that only 9 responses were used — the first question was used as a practice to allow the subject to familiarize himself with the format). After the first practice question, the questions were presented in a random order. Once complete participants were asked to move on to the survey page. The survey page consisted of six multiple choice questions rating their impressions of the test they had just performed.

The pages were designed to prevent the use of the browser back or refresh button preventing users from visiting previously viewed pages. This was important since we did not want to have participants view the tutorials more than the allocated duration, nor did we want them to attempt to re-try the tasks.

To reduce the number of variables possibly effecting the experiment we did not include any participants with previous Prograph experience. We reasoned that past experience would cause an unfair advantage when evaluating biconditionals and a disadvantage for conditionals since habit formation with biconditionals would cause increased interference in expression evaluation. This meant that all participants were effectively learning the syntax of the data flow language for the first time.

4.3 Results

The focus of the experiment was to determine if there was a significance in the difference between unconditional and biconditional expressions. The results of the two surveys were analyzed using paired, 1-tailed t-tests. The three factors analysed were expression type (unconditional vs. biconditional), speed of response, and accuracy. We chose a significance level of $\alpha=.05$.

The mean calculated for biconditional response time was 6042ms and for unconditionals was 3880ms. Standard deviation for biconditionals was found to be 2097ms versus unconditionals of 1207ms. Calculation for significance between the two groups was performed using a paired, 1-tailed t-test on response time and showed significance with $t(13) = 5.016$, $p < 0.001$.

The mean calculated for biconditional correctness was 89% and for unconditionals was 96%. Standard deviation for biconditionals was found to be 20% versus unconditionals of 9%. Calculation for significance between the

two groups was performed using a paired, 1-tailed t-test on correctness and did not show significance with $t(13) = -1.571$, $p = .07$.

We explored order effects with respect to task ordering using a one-way ANOVA with the following results: Biconditional correctness, $F(1,13)=.050$, $p>.05$; Unconditional correctness, $F(1,13)=1.242$, $p>.05$; Biconditional time, $F(1,13)=9.305$, $p<=.01$; Unconditional time, $F(1,13)=0.045$, $p>.05$.

4.4 Discussion

The t-test results on response time provide a good indication that biconditionals are more difficult to comprehend than unconditionals which would have caused the added time required to respond for biconditionals. The differences in the mean, 64% faster for unconditionals, provides an indication that unconditionals are simpler and easier to understand.

The t-test results for correctness did not show significance in the difference between the biconditionals and unconditions, but there is a trend towards increased correctness in the unconditional case.

The 89% and 96% correctness values indicate the programmers have no difficulty in manipulating control flow annotation in a data flow language. Thus, Prograph provides a mean for integrating both control and data flow information without obscuring the conditional elements of control flow.

An order effect was found for biconditional time, which we potentially attribute to a learning effect. It is likely that performing the unconditional task first assisted participants in performing the biconditional task.

5. Conclusion

In this paper, we explored the representation of control flow in Prograph, a visual data flow language. The integration of the two types of flow provides a notation that can be used for the development of improved program comprehension tools. This combination avoids the limitations existing in pure data flow languages to create a visual representation for simultaneously viewing both control and data flow. In future research, we intend to demonstrate that allowing programmers to view both types of control will improve their ability to leverage data flow information during program comprehension tasks. The high correctness values that we obtained indicate that programmers have no difficulty in extracting control flow information from the merged representation. Users with no experience in visual programming had little difficulty in evaluating Prograph control structures.

Apart from the possible benefits to program comprehension, this research also suggests a technique for improving the representation of control flow in the Prograph family of languages. Significant improvement in time and a trend towards improved correctness suggests that unconditionals are easier to understand and formu-

late. The improvement in time provides additional verification of earlier findings with respect to Boolean systems. As expected, the time taken to manually evaluate a uniconditional is significantly less than for a biconditional expression.

While existing models of program comprehension exploit a programmer's ability to extract control flow from source code, little attention is paid to data flow. This research has the potential to address this omission by using existing research in visual programming languages to provide improved and more effective representations. The simultaneous representation of control and data flow creates an integrated representation that can be used as the foundation for new and better comprehension tools and environments.

In future work, we plan to explore more complex expressions and expand our range of testing to include other control structures such as loops, repeats, and locals.

6. References

[1] Anick, P. G., Brennan, J. D., Flynn, R. A., Hanssen, D. R., Alvey, B., Robbins, J. (1990). A direct manipulation interface for Boolean information retrieval via natural language query. Proceedings of the ACM SIGIR Conference 1990, 135-150.

[2] Blackwell, A. F., Green, T. R. G. Does metaphor increase visual language usability? In 15th IEEE Symposium on Visual Languages (VL99), pages 246-253, 1999.

[3] Brooks. L. R. Spacial and Verbal Components of the Act of Recall. Canadian Journal of Psychology, 22(5), pp. 349-369. 1968.

[4] Clark, G. J., and Wu, C. T., (1992). DFQL: Dataflow Query Language for Relational Databases, Naval Postgraduate School, Department of Computer Science, Monterey, CA.

[5] Corbi, T.A. Program Understanding: Challenge for the 1990s. IBM System Journal, 28(2):294--306. 1989.

[6] Cox, P.T., Giles, F., Pietrzykowski, T. Prograph: a step towards liberating programming from textual conditioning. IEEE Workshop on Visual Languages, Rome, Italy, Oct. 4-6, 1989.

[7] Cross, J., Hendrix, D. and Maghsoodloo, S. The effectiveness of Control Structure Diagrams in Source Code Comprehension Activities. IEEE Trans. on Software Eng., 28(5), pp. 463-477, 2002.

[8] J.B. Dabney; T.L. Harman, Mastering Simulink 4, Prentice Hall (2001)

[9] Gauvin, S. ReactoGraph: A Visual Programming Language for the Development of User Interfaces. Dalhousie University Masters Thesis, 2003.

[10] Good, J., Brna, P., Cox, R. Program comprehension and novices: Does programming language make a difference? Proceedings of the 19th Annual Conference of the Cognitive Science Society, pp. 936. 1997

[11] Green, T. R. G., Petre, M. Usability analysis of visual programming environments: a 'cognitive dimensions' framework. Journal of Visual Languages and Computing, 7, pp. 131-174. 1996.

[12] Greene, S., Devlin, S., Cannata, P., and Gomez, L. (1990). No IFs, ANDs, or ORs: A study of database querying. International Journal of Man-Machine Studies, 32(3):303-326.

[13] Gurr, C. Isomorphisms in Visual Languages. In 15th IEEE Symposium on Visual Languages (VL99), pp 246--253, 1999.

[14] R. Helsel, Visual Programming For HP-VEE, Prentice Hall (1997)

[15] G.W. Johnson, R. Jennings, LabVIEW Graphical Programming, McGraw-Hill (2001)

[16] McGrenere, J. and Ho, W. "Affordances: Clarifying an Evolving a Concept", CHI Conference Proceedings, 2000.

[17] Mark Meyer and Tim Masterson. Towards a Better Visual Programming Language: Critiquing Prograph's Control Structures. Consortium for Computing Sciences in Colleges-Northeastern Region, Ramapo College of New Jersey, April 28-29, 2000.

[18] Michard, A. (1982). A new database query language for non-professional users: Design principles and ergonomic evaluation. Behavioral and Information Technology, 13, 279-288.

[19] Norman, D.A. "Affordance, conventions, and design." Interactions, 6(3), pp. 38-42. 1999.

[20] Pennington, N. "Comprehension Strategies in Programming," Proc. Empirical Studies of Programmers: Second Workshop, pp. 100-112, 1987.

[21] Posner, M.I. Orienting of attention. Quarterly Journal of Experimental Psychology, 32, 3-25, 1980.

[22] C.C. Risley and T.J. Smedley, Visualization of Compile Time Errors in a Java Compatible Visual Language, IEEE Symposium on Visual Languages, Halifax, Canada, 1998, 22-29.

[23] Seacord, R., Plakosh, R. and Lewis, G. Modernising Legacy Systems: SOftware Technologies, Engineering Processes, and Business Practices. SEI Series in Software Engineering. Addison-Wesley. 2003.

[24] J.A. Sharp, Data Flow Computing, Ellis Horwood (1985)

[25] Softwire, http://www.softwire.com, (accessed March 2004)

[26] Walenstein, A. Cognitive Support in Software Engineering Environments: A Distributed CognitionFramework. PhD thesis, School of Computing Science, Simon Fraser University, May 2002.

[27] Weiland, W. and Shneiderman, B. (1991). A Graphical Query Interface Based on Aggregation/Generalization Hierarchies, Department of Computer Science Technical Report CSTR-2702, University of Maryland, College Park, MD.

[28] Deji Young and Ben Shneiderman. A Graphical Filter/Flow Representation of Boolean Queries: A Prototype Implementation and Evaluation. Journal of the American Society for Information Science, 44(6): 327-339, 1993.

Control Constructs in Visual Meta-Programming Language[1]

Mikhail Auguston

Department of Computer Science

Naval Postgraduate School, Monterey, CA, USA

maugusto@nps.edu

Abstract

Two-dimensional data flow diagrams provide representations of meta-programs that expose potential parallelism. This work suggests visual notations for data structures, data flows, pattern matching, conditionals, iteration and synchronization. The framework provides encapsulation means for hierarchical rule design and default mapping rules to reduce screen real-estate requirements. The representation supports practical reuse of generic data structures for program representation, abstract syntax type definitions for common programming languages, and related default mappings (e.g. parsing and deparsing).

1 Introduction

Compiler and program generator design is a domain that has been studied extensively. The following domain features are among the most common for language processor design and have contributed to our language.

- The architecture of a language processor in most cases can be represented as data flow between components (e.g., the famous compiler data flow diagram on page 13 in the "Dragon Book"[1]).

- Context-free grammars are used to specify syntax and to serve as a basis for parser design.

- Attributes associated with data items, and attribute dependency and propagation schemes are of great relevance.

- Tree and graph traversal and transformation are common for optimization and code generation tasks.

- Pattern matching (e.g., with respect to regular expressions or context-free grammars) is a useful control structure for this problem domain.

Data-flow diagrams are most commonly used to represent dependencies between data and processes in visual programming languages, for instance, in LabVIEW [5] and Prograph [7]. Two-dimensional diagram notation could significantly improve readability of meta-programs. Some of these ideas have been explored in our previous work [3]. The more detailed introduction into our visual meta-programming notation is presented in [4]. This paper focuses on control structure aspects and provides new examples of meta-programming rules.

2 Data flow diagrams

Detailed rationale for data-flow diagram notation and a survey of related work can be found in a previous paper [3]. Briefly, a meta-program is rendered as a set of two-dimensional data flow diagrams that shows the dependencies between data and functional computations. Each diagram defines a single function called a rule.

Each diagram represents a single function with several inputs and outputs. The rectangular boxes in our notation denote data objects, and ovals denote patterns, that could be matched with data objects. A rectangular box representing a data item may have names associated with its input ports, those names can appear in the expression within the data box and are visible only within this box. A simple data flow delivers values one at a time. There is never more than one data item in a simple data flow channel between sender's output port and receiver's input port.

A stream is similar to a simple data flow, except that it contains an unbounded sequence of values, and can deliver as many of those values as needed to match the guarding patterns of a rule. A data box fires when and only when the following conditions are satisfied:

- all input values are delivered to the corresponding input ports;

- all data flow output values produced at the previous execution cycle are consumed from output ports by input ports of the connected nodes downstream (output streams are always available for output).

A data switch node has one or more input ports and the same number of output ports on the "True" and "False" edges. Input ports have names used within a Boolean expression associated with the data switch node. If the Boolean expression evaluates to True the input data is forwarded to the "True" edge output ports, if it evaluates to False, the data items are passed to the "False" edge ports.

This construct demonstrates that it is possible to integrate data flow and control flow mechanisms into a single notation, as opposed to, e.g. [6].

Data object patterns are used to visualize structure of objects in order to provide access to object components and associated objects. An object pattern may be placed in any part of the data flow and is matched with the object connected to the pattern's input.

If pattern matching is successful the output object is passed downstream. If pattern matching fails (or all branches of an alternative pattern fail) then the entire diagram execution fails, and the diagram sends to its outputs a default value Null, unless the pattern has been provided with the 'Failed' output route. The pattern sequence consumes as many objects from the stream as it can successfully match. The semantics of pattern matching is derived from RIGAL language[2].

3 Types and default mappings

Basic predefined types include `char` (characters) and `int` (integers). Aggregate types are trees (ordered tuples of heterogeneous objects), which are useful for abstract syntax representation, sets and lists (sequences of homogeneous objects that could be dynamically augmented). Types can be parameterized.

Example of a list type definition.

```
text :: [char]
```

Example of a type definition with several alternatives (union type).

```
expr :: int | id | (op expr expr)
```

This effectively declares that types int and id are subtypes of expr in the scope of this definition.

Selected rules may be declared as default mappings. There can be at most one default mapping from the given source type to the given target type. The type system uses name equivalence, so that it is easy to create many disjoint but isomorphic subtypes of a given type. This facilitates control of default mappings. It means that corresponding rule calls are optional in the diagrams, and input and output data boxes may be connected directly. This helps to save some screen real estate and to make diagrams less crowded and more readable. Typically default mappings are introduced for text-to-abstract syntax (parsing) and for abstract syntax-to-text mappings (de-parsing, or abstract syntax-to-concrete syntax mappings).

Yet another kind of default mapping is associated with construction and concatenation operations for tuples, sets and lists.

Definitions of abstract syntax types for common programming languages and related parsing and de-parsing default mappings may be valuable assets for reuse.

4 Examples of parsing and generation rules

The following rule `#expression` illustrate the main control structures of our notation on a simple arithmetic expression parsing example for the following grammar.

```
expression::=  term  ( "+" term )*
term::= identifier | number |
        "(" expression ")"
```

The input source code is a stream of strings representing tokens. It is assumed that there is a lexical module that filters out comments, spaces, tabs, end-of-line characters from the stream before it is fed to the parsing rules. Each of the rules `#expression` and `#term` outputs an object of the type `expr` representing the abstract syntax tree for the expression parsed.

Remarks for the rule #expression

- A pattern may have several alternatives. The alternative pattern is denoted by gray vertical bar at the left hand side of diagram. The alternatives are applied in order of appearance, if the first alternative fails, the input stream is restored to its state before the first alternative and the next alternative is applied until one of alternatives is successful. If all alternatives fail, the entire alternative pattern also fails.

- The entire sequence of patterns in the rule consumes part of the input stream delegated from the calling rule. A rule pattern can consume arbitrary many data items from a stream.

- When a pattern fails, all data items that were matched by parts of the pattern are restored to the stream, including indirectly invoked sub-patterns.

- The rule #expression may be used as a pattern. If pattern matching encapsulated in the rules is successful, the rule also is successful and returns value, which may be used by the calling rule.

- The application of pattern sequence "+' and #term may be repeated zero or more times (indicated by the ellipsis '…'), each successful pattern matching contributes two elements to the tuple constructor (as the boxes of the types `op` and `expr`, correspondingly).

- Nesting boxes and forwarding output of pattern matching inside the resulting box of the type `expr` provide an intuitive visualization for the tuple constructor.

- If the first branch of the alternative pattern (i.e. '+' #term) fails, the second branch unconditionally succeeds (since it does not have any pattern to match and does not consume any values from the input stream), and produces the default value (also Null in this case). Any value at the input port of string constant data box ("done" or "not

yet") fires this data box and sends the corresponding   string value down the data flow.

#expression: **Stream** [string] -> expr

Fig.1 Parsing the grammar rule `expression::= term ( '+' term ) *`

- The connectors marked FINISHED are used to avoid arrow intersection, and in fact are representing a data merge node.

- The iteration of the pattern sequence ('+' #term) will produce a nested tuple of the type `expr` representing left-associative abstract syntax tree for the arithmetic expression parsed. After the current tuple has been completed (i.e. all three components have been supplied), the whole tuple object is passed down the flow to the data switch, and the tuple constructor is ready for the next iteration.

We will illustrate use of pattern matching for tuple traversal and default mappings on the example of generating textual prefix representation of the arithmetic expressions. The rule #generate takes as an input an object of the type expr and outputs the text with the prefix form. We suppose that the rule #generate: expr -> text has been de-

clared as a default mapping in the scope of our example and that there are default mappings defined for int -> text, id -> text, and op -> text.

Remarks for the rule #generate.

- The input is an abstract syntax object for an expression and a pattern for this object (nested oval boxes) provides an access to its subcomponent retrieval.

- A pattern may check the type of the object.

- A constructor box for text represents a concatenation operation for text objects. The concatenated components may be either results of default mappings, or constants of the text type (like '(' and ')') directly embedded into the constructor box.

#generate: expr -> text

int:

id:

expr:

op:

expr:

expr:

text:
`(`

text:

text:

text:

`)`

Fig. 2 Example of text generation

5   Conclusions and work in progress

We expect the following advantages of this approach.

- Visualization of data and data flow provides better readability and uncovers parallelism in data processing.

- The language provides systematic and consistent correspondence between constructors and patterns.

- Data streams and patterns give a flexible and expressive framework for parsing rules supporting extended BNF notation.

- Control mechanisms, such as data switch, iteration and recursion well fit with data-flow notation and provide a transparent and expressive language to define different kinds of meta-programming algorithms.

At the moment of this writing the interpreter for the core of data-flow language is already implemented, and work is in the progress on the graphical editor.

References

[1] A.Aho, R.Sethi, J.Ullman, Compilers: Principles, Techniques, and Tools, Addison-Wesley, 1986

[2] M.Auguston, "RIGAL - a programming language for compiler writing", Lecture Notes in Computer Science, Springer Verlag, vol.502, 1991, pp.529-564.

[3] M.Auguston, A.Delgado, Iterative Constructs in the Visual Data Flow Language, in Proceedings of IEEE Symposium on Visual Languages, Capri, Italy , IEEE Computer Society Press, 1997, pp.152-159

[4] M.Auguston, V. Berzins, B.Bryant, "Visual Meta Programming Language", in Proceedings of the 2001 OOPSLA Workshop on Domain-Specific Visual Languages, Tampa, Florida, pp. 69-82 (http://www.isis.vanderbilt.edu/oopsla2k1/Papers/papers.htm)

[5] E.Baroth, C.Hartsough, Visual Programming in the Real World, in Visual Object-Oriented Programming, Concepts and Environments, (ed. M.Burnett, A.Goldberg, T.Lewis), Manning 1995, pp.21-42

[6] W.Bennet, Visualizing Software, Marcel Dekker, Inc., NY 1992

[7] P.T.Cox, F.R.Gilles, T. Pietrzykowski, "Prograph", in Visual Object-Oriented Programming, (ed. M.Burnett, A.Goldberg, T.Lewis), Manning 1995, pp.45-66

[8] Glaser H., Smedley T., PSH - the next generation of command line interface, in Proceedings of the 11th International Symposium on Visual Languages, VL'95, IEEE Computer Society Press, 1995, pp.29-36

[9] T.D.Kimura, Object-Oriented Dataflow, in Proceedings of the 11th International Symposium on Visual Languages, VL'95, IEEE Computer Society Press, 1995, pp.180-186

An Interactive Visual Query Interface on Spatial/temporal Data

Xin Li and S. K. Chang
Dept. of Computer Science, University of Pittsburgh, USA
{flying, chang}@cs.pitt.edu

Abstract: In this paper a visual approach for the query interface on the spatial/temporal data is described, which is designed for the users who know nothing about query languages. Based on concept of elementary query and query operators, a complex spatial/temporal query could be built as a compound query which is broken down into several interactive iterations. In each of iterations, users just need to build a simple elementary query and select a query operator, which greatly reduces the mental load of users. To make query building more efficient, a novel approach based on the query pattern retrieving and matching is explained and used in the interface. An experimental prototype has been implemented to demonstrate the usability of the interface.

Keywords: spatial/temporal query, query interface, query operator, query pattern;

1. Introduction

Query on the objects which may change over the time and space become more and more important and complex. For example, when the hurricane comes, it is crucial to know the movement of its range over the time. Meanwhile some related objects could also be important. For instance, it is of great interest for people to know the houses and residents which are in the hurricane area, and moving airplanes which are interfered by the hurricane. This process makes queries even more complex. There are many interesting problems related to this kind of spatial/temporal query. Query interface is one of most important issues, which interacts with users to build query and visualize result.

In this paper a visual approach for interactive query building interface is described in detail, which is designed for the user who knows nothing about query languages. It is analyzed based on the Σ Query Language (ΣQL) [1], which is a powerful SQL-like temporal/spatial query language. However, it can be applied for any query languages. In other words, it is query language independent.

It is worth noting that one of main usability issues in HCI (Human Computer Interface) is to decrease user's mental load [3]. To achieve this goal, in our approach, complex query can be built by several steps and each step is expressed in a straightforward visual form. User always has the chance to see the result of query before moving to next step. A new concept - *query path* is proposed in this paper

which basically is the sequence of related queries from initial (simple) one to final (complex) one. Similar query paths could be generalized as a *query pattern* during query processing. Utilizing query patterns, similar queries can be built in few steps based on previous experience.

This paper is organized as follows: Section 2 presents ΣQL query language as the background knowledge. Section 3 describes the related works and the motivation of our approach. In Section 4, *elementary query* and *compound query* are defined, and *query operators* especially spatial / temporal operators are described in detail. To reuse the repeated structures in complex query, *query path* and *query pattern* are defined in Section 5. Section 6 presents algorithms for *query pattern* retrieving and matching. An experimental prototype is described in Section 7 to show the visual appearance of the interface by an example, followed with the future work in Section 8.

2. Background

Query language is one of key issues in query processing. In spatial/temporal query, it is crucial for the query language to deal with spatial/temporal relation effectively and efficiently. ΣQL is an extension of SQL, which can represent spatial /temporal queries using a natural way for users.

ΣQL query language explicitly deals with spatial/temporal information in query with the clause "CLUSTER" [1, 2]. In the same cluster, objects have same characteristics (such as having a same time value). For example, from a video source we want to find a red car followed by a truck within 10 minutes. The corresponding query in ΣQL is:

```
    SELECT object
    CLUSTER ALIAS OBJ1, OBJ2
    FROM
       SELECT t
       CLUSTER *
       FROM video_source
    WHERE OBJ1.type='car' AND
           OBJ1.color='red' AND
           OBJ2.type='truck' AND
           OBJ1.time<OBJ2.time AND
           OBJ1.Time>OBJ2.time-10
       Figure 1. Example of a ΣQL Query.
```

In this Query, OBJ1 and OBJ2 will be retrieved. Each one is several collections of the same objects which have the same

time value. In [2, 7], a query system for ΣQL is described in detail including query processing, refinement and optimization. However, another problem is that users have to learn ΣQL query language before they can use the query system. Even for the expert, it is not easy to write a query in ΣQL by hand. As mentioned in [6], in most information retrieving applications on temporal/spatial data, a visual query building tool is of great significance.

3. Related Research

Broadly, the query interface can be categorized into two groups: textual interfaces and non-textual interfaces. The ΣQL scheme can be treated as a textual interface, which is a text-based extension to SQL [1]. For the non-textual interface, which we are focusing on, usually there is a visual language supporting it, called as a visual query language. Several approaches on visual query languages are proposed in [3,9,10]. In these approaches, the picture elements conveying different meaning are connected together to form a query. For example in [3], a simple query can be represented by a data input element and a data output element connected by a filter symbol. A filter also can connect different simple queries to get more complex queries. In [9], iconic symbols which have various meaning are composed by spatial relation (such as horizontal or vertical concatenation) to generate a query. From the style of connecting elements, all these approaches can be distinguished into two groups: connection-based [3] and geometric-based [9].

The foundation of the connection-based query languages is E-R (Entity-Relation) diagram, which is widely used in database design. But unfortunately for naïve users who have no knowledge on database and query language, it is not easy to understand E-R diagrams. Another constraint for both approaches is: when query is getting complex, all these diagrams will become much more difficult to build and manage.

To eliminate the accumulated complexity, complicated query are broken into several separated iterations in our approach. It is worth noting that in many cases, users are not completely clear on what they are really trying to find before starting, or they may change their interest after the result of queries shows up. For example, a commander wants to find all tanks of his troop in a battle field. Once tanks are shown on the map, objects of threat near these tanks highly probably will become the target of next query. In other words, queries could be generated step by step and coupled together getting more and more complex. This is another reason to make query building interactively.

4. Elementary Query and Compound Query

Here is an example to show the processing of interactive query building: the query in figure 1 can be rewritten in following equivalent way:

```
(2)    SELECT OBJECT
(2)    CLUSTER t ALIAS OBJ1,
(1)        SELECT OBJECT
(1)        CLUSTER t ALIAS OBJ2
(1)        FROM video-source
(1)        WHERE OBJ2.type='truck'
(2)    FROM Video-Source
(2)    WHERE OBJ1.type='car' AND OBJ1.color='red'
                AND OBJ1.time<OBJ2.time AND
                OBJ1.Time>OBJ2.time-10
```
Figure2 breaking down the complex query

Using this reorganizing, the query can be built in two iterations. The query for the first iteration is the statements marked with (1) *selecting all the truck objects from video source*, which is organized by time. The statements with mark (2) are the query for the second iteration. Besides the object, time, space and some properties needed to be specified as same as first step, the relationship with the first step is another thing needed for the second step. From this point of view, the second iteration can be divided into two parts: one part is building a simple query –*"selecting all the red cars from video source"*; the other one is choosing a relation operation to combine these two queries. For users, the relation between different queries can be treated as an operator like a function which is applied on two queries. In this example, the query operator is working on the temporal relation -*Objects in query (1) appears before Objects in query (2) within 10 minutes.*

As shown in Figure 2, each complex query can be treated as a *compound query* constructed in several iterations. During each of iterations, user needs to construct a basic query, called an *elementary query*, which is defined as follows:

Elementary-Query = <Object, Source, Time, Space, Direction>

The five items are explained as follows:
- Object: object type, such as 'car', 'truck', etc.
- Source: the data source which query will be applied.
- Time: time of interest (TOI) [5].
- Space: area of interest (AOI) [5].
- Direction: set of constraints on movement, color and other properties of objects, such as {object.color=red AND object.moving=True}.

A compound query is defined recursively based upon query operators:

Compound-Query = Query-Operator (Elementary-Query, Elementary-Query) | Query-Operator (Elementary-Query, Compound-Query)

Using Query operators, elementary queries are composed by the nested query in ΣQL. But this composition rule is implicit for the users. In other words, users don't need to know the nested ΣQL. What is required is an understanding

of the meaning of different query operators.

5. Query Operators

Query operator specifies the relation between the object in current query and previous one. It is represented as a constraint which could nest two queries in ΣQL. For example, in Figure2, the nesting constraint is "OBJ1.time < OBJ2.time AND OBJ1.Time>OBJ2.time-10". In general, query operators can be categorized into three groups: direction operator, spatial operator and temporal operator.

i. Direction Operators
Direction Operator describes the relation on the common properties of the retrieving objects except time and space. In most temporal/spatial queries, there are only limited number of common non-temporal and non-spatial properties for objects, such as object, type, color, moving and background. For example, for the property "object", there are relations such as coexistent and associate. For "type" of two objects, there are relations such as Equal, Similar, and Different, which are predefined in domain knowledge.

Formally, direction operator is defined as a 2-tuple:
Direction operator=<Common Property, Relation Predicate>

For example, direction operator α =<type, similar>, it applies on the two queries Q_1 and Q_2. Assume the Object1 and Object2 are targets of Q_1 and Q_2, α (Q_1 , Q_2) generates a new query nesting the Q_1 and Q_2 by predicate "similar(Object1.type, Object2.type)".

ii. Spatial Operators
Spatial operator specifies the spatial relation between the retrieving objects. There are many methods to specify a special relation between objects [5]. For a graphic user interface, the following two approaches are the most natural ways:
1. Orientation & Distance: It is easy for user to specify spatial relation using orientation and a distance such as "*Is there a red car in the 0.1 miles northeast of my house?*" (Figure3).

Figure 3. Orientation and Distance to specify spatial relations.

2. Predicate & Distance: Spatial predicate [4] is useful in the scenario which users don't care about the orientation, The spatial predicate classifies the special relation of two objects as *disjoin, meet, overlap,*

coveredBy and *inside* (figure 4). Meanwhile, a distance is also needed to specify how far between centers of objects, which may depend on the shape of objects.

Figure 4. Predicate of Spatial relations.

ii. Temporal Operators
Temporal operator specifies the temporal relation between two retrieving objects. Temporal relation can be treated as one case of spatial relation (Figure 5), because the time property of every object can be visually represented as segments on time line.

Figure 5. Temporal Relation could be treated as a special case of spatial relation.

So we also have following relation between the time of two objects: disjoin, meet, overlap, coveredBy and inside. Using more natural words about time, they are *before, end-meet-start, overlap, co-start and within*. Meanwhile a "distance" is also needed, called as *time interval* between two objects.

6. Interactive Querying, Query path and Query pattern

As shown in previous sections, a compound query can be represented by **n (n=1, 2, 3……) elementary queries** with **n-1 query operators**. The main function of the user interface is to provide a visual interface for users to compose elementary queries and query operators into a compound query. However, for a complex query, this may make the query building even more time consuming and error prone. An approach to utilize repeated structures in queries is proposed based on following two concepts: *query path* and *query pattern*.

Query path p is defined as a sequence of queries (Q_0, Q_1, Q_2, ……, Q_n), in which
$$\forall i = 1,2,...n \quad \exists op \in \text{Query-Operators} \quad \exists q \in \text{Elementary-Query} \quad Q_i = op (q, Q_{i-1})$$
Basically query path shows the track of an interactive query building form Q_0 to Q_n. In each step, a query operator and an elementary query are applied on the previous one.

A Query path p = (Q_0, Q_1, Q_2, ……, Q_n) is *complete*, if and only if
$$Q_0 \in \text{Elementary-Query}$$
Complete query path is a history of an interactive query

building from an elementary query.

Given query paths $p1=(Q_0, Q_1, Q_2, \ldots, Q_n)$ and $p2 = (Q'_0, Q'_1, Q'_2, \ldots, Q'_m)$ $(m \leq n)$, $p2$ is **sub-path** of $p1$, e.g. $p2 \subset p1$ if and only if
$$\exists s, 0 \leq s \leq n-m \quad \forall i = 0,1,\ldots m \quad Q'_i = Q_{s+i}$$

Query Pattern **P** is defined as a triple $<Q_{path}, S, L >$ in which
- Q_{path} is a *query path* such as $\{Q_0, Q_1, Q_2, \ldots, Q_n\}$;
- S is a set of variables, which is $\{x_1, x_2, \ldots, x_m\}$. $x_1, x_2, \ldots, x_m$ are variables in the query $Q_i(i=0,1,\ldots n)$ in $Q_{path} = \{Q_0, Q_1, Q_2, \ldots, Q_n\}$;
- L is a linguistic label for the pattern.

While a query pattern is generalized from query paths, some constant values in the queries may be marked as variables indicating that these values are exchangeable while doing a pattern matching. The linguistic label is a tag for a pattern in natural language. It is used as an index for user to choose patterns.

Given query Patterns **P1**= $<Q_{path}, S, L >$ and **P1**= $<Q'_{path}, S', L' >$, **P2** is *sub-pattern* of P1 e.g. **P2** $\subset$ **P1**, if only if I and II:
    I. $S' \subset S$
    II. Q'_{path} is a sub-path of Q_{path}, e.g. $Q'_{path} \subset Q_{path}$

7. Query pattern retrieving and matching

Query pattern is important because it could cover most of frequently used queries and make them simple. In our approach, a set of query pattern $S_{pattern}$ is stored and maintained during query processing. The query patterns are retrieved dynamically and manually. At the initial state, there are no query patterns in $S_{pattern}$. When one query is finished, the query path is shown and user is asked to choose the variables and input a linguistic tag for it. The following pseudo-code shows the algorithm for adding a new query pattern into $S_{pattern}$:

```
procedure addPattern(Spattern: Set of pattern; P: pattern)
var P0:pattern
begin
    for i:=0 to Spattern.size()-1 do
    begin
        P0:= Spattern.Item(i);/*get the pattern in the set one by one*/
        /* If P0 is sub-pattern of P, delete P0 from the set */
        if (P0⊂P) then Spattern.delete(P0);
        /* If P is sub-pattern of P0, no need to add P, quit*/
        if (P⊂P0) then return ;
    end;
    /* If P is not any pattern's sub-pattern, add it into the set*/
    Spattern.add(P);
end;
```

A query pattern matching occurs every time when an elementary query is finished. For example, the current query path is $p=(Q_0, Q_1 \ldots Q_i)$ $(i=0,1,\ldots)$, when the query Q_{i+1} is done, query path becomes $p'=(Q_0, Q_1 \ldots Q_i, Q_{i+1})$. All of query pattern P=$<Q_{path}, S, L >$ in $S_{pattern}$ with $p' \subset Q_{path}$ (instantiate the variables in Q_{path} to match p') are shown for user to choose. All the matched patterns are listed by the linguistic label. A result preview is also shown to let user know what the result will look like if this query pattern is taken.

8. An Experimental Prototype

An experimental prototype of the interactive visual query interface has been implemented. The system diagram is shown in Figure 6. The user request comes from the query interface to query plan generator, which produce a ΣQL query to the query executor. After the query is executed, the query and result together are sent to query pattern matcher. The matched query patterns will be presented to user interface as well as query result.

Figure 6. An Architecture of the Interactive Query System.

The main user interface is shown in Figure 7. On the left of main window, there are some objects shown on a map which is a background. The red rectangle on the map shows the AOI (Area of Interest) for the current query. As we discussed in previous sections, an elementary query has five items needed to be specified: Object, Source, Time (TOI), Space (AOI) and Direction. The Space (AOI) is shown on the map, the other four items are on the right side.

What follows is an example to illustrate query construction, pattern retrieving and matching. Suppose we have the first iteration as shown in Figure7. To make ΣQL query simple, the constraint about TOI and AOI is omitted. The query we build in this iteration is:
    SELECT OBJECT
    CLUSTER * alias object1
    FROM video-source
    WHERE (object1.type='truck') and
    (object1.color='red') and (object1.moving=True)

Figure 7. Main Interface of Interactive Query System.

If the result of this query is uninformative, the second iteration is applied as in Figure 8. The elementary query shown is a query search any objects. The query operator chosen is a direction operator α =<type, similar>.

Figure 8 the Second Iteration

The query we built so far is expressed in ΣQL as follows:

    SELECT Object
    CLUSTER * alias object2
    FROM video-source
    WHERE similar (object2.type, object1.type)
    AND object1 in
            SELECT OBJECT
            CLUSTER * alias object1
            FROM video-source
            WHERE (object1.type='truck') AND
            (object1.color='red') AND
            (object1.moving=True)

Click button "Query", the result will be shown on the map. Suppose the result is still uninformative, the third iteration is applied and shown in Figure 9. It is worth noting that the object in the elementary query is "object2", it means searching the same object as the second iteration. In the system, "object1", "object2"… are all reserved words which represent the result of the first, second … iterations. Set object of elementary query as the previous object is a way to refine previous query by just adding more nesting constraints. In this example, the new constraint is introduced as a spatial operator in format <Predicate, Distance>, which

says "the object is 0.1 mile around the retrieved objects"
The query we built until the third iteration is:

    SELECT Object
    CLUSTER * alias object2
    FROM video-source
    WHERE similar (object2.type, object1.type)
    AND distance(object2,object1)<0.1
    AND object1 in
            SELECT OBJECT
            CLUSTER * alias object1
            FROM video-source
            WHERE (object1.type='truck') AND
    (object1.color='red') AND (object1.moving=True)

Figure 9 the Third Iteration

Suppose the result is still uninformative, the fourth iteration is shown in figure 10. This iteration is still refining the previous one by adding a temporal constraint, which says "the object appears before the retrieved objects in any time".

Figure 10 the Fourth Iteration

The query we get in fourth iteration is:

    SELECT Object
    CLUSTER * alias object2
    FROM video-source
    WHERE similar (object2.type, object1.type)
    AND distance(object2,object1)<0.1
    AND object2.time<object1.time
    AND object1 in
            SELECT OBJECT

```
        CLUSTER * alias object1
        FROM video-source
        WHERE (object1.type='truck') AND
(object1.color='red') AND (object1.moving=True)
```

Iterations as above could be applied as many as necessary. Suppose the user is satisfied with the result, a window shown in Figure 11 will pop up to ask users inputting query pattern manually. The query path is shown on the top of window. There are two lists below it. The left one contains all the constant values in query path. Using the first button in the middle, selected constant can be set as a variable which will be listed in the right list. User can set variables back to constants at any time using the second button.

Figure 11 Manually Query Pattern Retrieving

The query pattern is added into the $S_{pattern}$ using the algorithm described in the section 7. This pattern could be matched and applied to build similar queries in the future. For example, a new query is built to search a moving car from video-source, the pattern just retrieved is matched by replacing "truck" with "car" and setting color "red" to "not specified", noting "truck" and "red" are all variables in the pattern.

Figure 12 Interface for Choosing Query Patterns

Once the elemental query are inputted and executed, all the matched query pattern will be shown in a window shown in Figure 12. The matched patterns are indexed by the linguistic labels, which are listed on the right part of the window. A result preview is also provided, which is showing the result if user takes this pattern. To save processing time, only partial result (restricted AOI and TOI) are retrieved and shown on the preview window.

9. Future work

In this approach, complex queries on temporal/spatial data are built interactively through a visual interface. *Query pattern* is proposed to capture repeated structures in queries, and reused them in following query building process. It becomes crucial for system to find similar queries and group them as a pattern. In current approach, similar queries are connected by *variables* in *query patterns*, which are selected by hand. However, it is time consuming and error prone, therefore how to retrieve query pattern automatically is a important issue for our future research. Ontology knowledge will be introduced into query pattern retrieving and matching process.

References:

[1] S-K. Chang, G. Costagliola, E. Jungert and F. Orciuoli, *Querying Distributed Multimedia Database Data Sources in information Fusion Applications*, Journal of IEEE transaction on Multimedia, 2004.
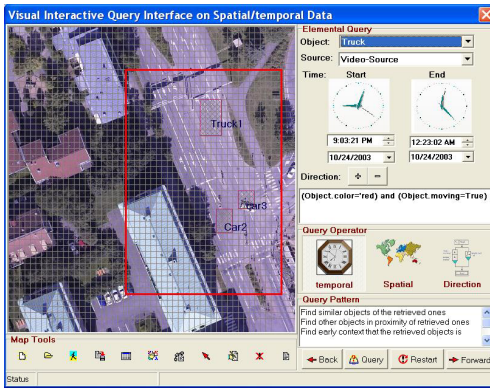
[2] S-K. Chang, W. Dai, S. Hughes, P. Lakkavaram and X. Li, *Evolutionary Query Processing, Fusion and Visualization* Proc. of Int'l Conf. on Distributed Multimedia Systems, September, 2002.

[3] A. Abdelmoty and B. El-Geresy, *Qualitative Tools to Support Visual Querying in Large Spatial Databases,* Proc. of Int'l Conf. on Visual Languages and Computing, 2003.

[4] M. Erwig and M. Schneider, *A visual language for the evolution of spatial relationships and its translation into a spatio-temporal calculus*, Journal of Visual Languages & Computing, Volume14, Number 2, April 2003.

[5] K. Silvervarg and E. Jungert, *Aspects of a visual user interface for spatial/temporal queries*, Proc. of Int'l Conf. on Visual Languages and Computing, 2003.

[6] L. Paolino, F. Pittarello, M. Sebillo, G. Tortora and G. Vitiello, *WebMGISQL-A 3D Visual Environment for GIS Querying*, Proc. of Int'l Conf. on Visual Languages and Computing, 2003.

[7] S-K. Chang, *Query Morphing for Information Fusion* Proc. of IMAGE: Learning, Understanding, Information Retrieval, Medical, Cagliari, Italy, June 9-10, 2003.

[8] G. Costagliola, A. Delucia, S. Orefice and G. Polese, *A Classification Framework to Support the Design of Visual Languages*, Journal of Visual Languages and Computing, Volume 13, Number 6, December 2002.

[9] A. Narayanan and T. Shaman, *Iconic SQL: Practical Issues in the Querying of Database through Structured Iconic Expressions,* Journal of Visual Languages and Computing, Volume 13, Number 6, December 2002.

[10] Y. Lee and F. Chin, *An Iconic Query Language for Topological Relationships in GIS*. International Journal of Geographical information Systems, Volume 9 Number 1, 1995.

Visual specification of spatial/temporal queries in a sensor data independent information system

Karin Silvervarg, Erland Jungert and Tobias Horney,
FOI (Swedish Defence Research Agency)
Box 1165, S-581 11 Linköping Sweden
{karin, jungert, tobho}@foi.se

Abstract - Modern query languages will in many cases be concerned with a large variety of heterogeneous data sources, of which most will correspond to different sensors and where the input data, in particular, may be of spatial temporal type. This requires not only structures for analysis, manipulation, data fusion and storage of such data but also methods for specification of the queries as well as visualization of the query result will be required. Spatial/temporal queries become especially complex since they have to deal with multiple dimensions. For this reason, means to support inexperienced end-users in defining spatial/ temporal queries must be developed. In this work a visual method for the specification of this type of queries is proposed for an environment where the input data sources basically are sensors of various types and where the queries can be made in a sensor data independent way.

Index Terms **- query language, visual user interface, spatial, temporal**

1. Introduction

The next generation query languages will generally require input from a large number of non traditional data sources where the data in many cases will be of heterogeneous type. Basically, these data sources will be sensors that may be located on different platforms. These sensors will also be of a large variety of types. However, common to most of these sensor data sources are that they will deliver data of spatial and temporal type that for the most part are intended for multimedia applications, which will have a strong influence on how to handle, analyze, and visualize the information that will correspond to the result of the queries. Furthermore, all these aspects will not only have a strong impact on the design of the query system itself but also on the way the queries are given. Such systems also require capabilities to automatically *select* sensors and algorithms for sensor data analysis without any user interference. The situation is complicated further because the selection of the sensor data analysis algorithms depends on aspects like weather and light conditions at the time when data are registered. Hence, it is necessary to design the query system so that the users do not need to have a deeper technical understanding. Systems with these capabilities are said to be *sensor data independent* [1] since queries can be applied independently of which sensors and which algorithms that are actually used. The motivation for the introduction of this con-

cept is that query system designed to handle multiple sensor data must be simple to use. That is, a high technical competence in using these complex information systems should not be required. Sensor data independence can from a practical viewpoint be carried out by means of an ontology combined with an ontological knowledge base [2], [3].

Sensors attached to information systems generate large quantities of heterogeneous data. In these data facts can be found that generally are redundant with respect to the problem to be solved. Furthermore, the data are often associated with various types of uncertainties due to limitations in the sensors. For this reason, a tool that can help separate the redundant information and fuse the relevant information is needed. However, such tools need to be general and efficient with respect to the variety of problems that may occur and that may require different combinations of information. For this reason the query language must include built-in support not just for sensor data independence but also for *sensor data fusion* [4] as well. Efforts must also be made towards the design of *usable systems* [5]. The latter is of great importance since most users will place a higher trust, or confidence, in the system at the same time as they will be able to concentrate their efforts on their primary tasks. In this work, the efforts towards the design of a usable user interface for a query language based on a visual language approach will be introduced. The query language, primarily designed for heterogeneous sensor data, is called ΣQL. In particular, in this work a visual user interface for the application of user-defined queries is introduced; a preliminary study of this approach was given in [6]. A more thorough description of ΣQL can be found in [7], [8]. A discussion of the set of sensors and their sensor data analysis algorithms currently used in the query system is given in [9].

Many attempts have been made to make visual interfaces for SQL, but only a few have touched the issues of spatial and/or temporal queries. An attempt to handle temporal data is given by MQuery [10], but this approach is very simple and does not for instance, handle Allens [11] time intervals. Chittaro et al [12] on the other hand have a system for handling those time intervals, but it is by no means a complete query language. Abdelmoty et al [13] have made a representation of visual queries in spatial databases with a filter-based

approach. Bonhomme et al [14] have proposed a visual representation for spatial/temporal queries based on a query-by-example approach. Chang [15] has made an approach to use c-gestures for making spatial/temporal queries in what he calls a sentient map. All of these have influenced our approach to a visual query language for both spatial and temporal queries.

The general structure of this work can be described as follows. Section II introduces issues and background on how to query spatial/temporal systems. Section III gives an overview of the system that the query processor is a part of. Section IV describes the functionality and visualization thereof in the user interface. Section V contains some larger examples of how different queries looks, followed by the conclusions and directions for future work in Section VI.

2. Problem formulation

Querying multiple sensor data sources includes many complex subtasks, for instance, analysis and fusion of the sensor data but also selection of the sensors. The latter aspect is concerned with the determination of the sensors that depending on, e.g., the actual weather and light conditions will register data with the highest quality, which in turn puts the focus on the fact that sensor data always are associated with various types of uncertainties. However, these problems have already been in focus for some time and they relate to the sensor data independence aspect discussed above. There are also other aspects of importance that concerns questions like how should a user apply queries to the system, i.e. the user interface. Of concern here is not only which input information that should be given by the user but also how it should be given and finally the question arises how the information returned by the system should be presented. Of importance when formulating the research problem is to remember that most input and output data is of spatial/temporal type. An aspect that a user must consider when applying a query concerns what will be asked for. The answer to this question is various types of objects that can be extracted by the analysis programs from the sensor data. The objects may for instance correspond to vehicles but it is not sufficient to just return the object types since the user may also be interested in different spatial and temporal relationships. Important spatial relationships are *topological relations*, *directions* and *distances.* Temporal relations are somewhat different and may concern *relative relations*, e.g. object *a* is before objects *b* in time. Time relations between intervals similar to those proposed by Allen [11] are also of concern. Finally, the obvious case with discrete time observations must also be handled, e.g. object *a* was observed at 16.00 o'clock. Clearly, queries that allow the combination of spatial and temporal conditions must be possible to apply. The approach taken in this work is based on a visual interface and

queries can logically be split into two main groups although from the user's perspective they are combined. The first main group concerns the querying of objects from the sensor data; answers to this group returns information about the object and its attribute and status values. Associated with the object type information is also a belief value, which indicates how much the system believes in the answer. This belief value is clearly a quantification of the uncertainty associated with the sensors and the data they produce. The second group concerns aspects normally found in where-clauses and involved aspects like specifications of the spatial/temporal relations and other types of conditions.

Figure 1. An overview of the information system.

3. The query system

An overview of the structure of the query system can be found in figure 1. The system is, from top to bottom, divided into a visual user interface, a query processor and the sensor nodes to which the sensor data sources are attached. The query processor includes a knowledge system that operates in conjunction with an ontology. This part of the system supports automatic selection of both the sensors and the algorithms for the sensor data analysis.   In a first set-up of the query processor the actual sensors have been a digital-camera, an infrared camera and a laser radar. However, the system is not limited to these three sensor types but other types can, on demand, be attached as well. The sensor nodes include means for target recognition. For this purpose a database containing a library of target models is attached to the ΣQL-processor. The target models stored in this library are used by the image analysis process to recognize objects found in the sensor data inquired by the users. A meta-database containing the available information that has been registered by the sensors is also attached to the query processor. The query system includes, contrary to conventional query languages, a sensor data fusion module. The purpose of this module is primarily to fuse information extracted from the sensor data that most of the time emanates from multiple sensors and, whose sensor data generally are of heterogeneous type.

Figure 2. Simple selection of AOI, IOI and object type.

4. The visual user interface

A. Functionality

The user interface has to be sensor data independent. The reason for this is that sensor technologies are constantly developing, thus it is almost impossible for a user to have specific knowledge about the capabilities of all sensors. It should be enough for a user to have a general understanding about problems associated with sensor data e.g. uncertainty and all sensors can't measure all possible attributes. To achieve sensor data independence the system cannot and should not work with concepts related to the sensors, but instead work with concepts that are relevant to the user. The basic questions that should be answered in a spatial/temporal environment is *where?, when?* and *what?*. We have come to call these area of interest (AOI), time-interval of interest (IOI) and object types. We have chosen to let the user mark the AOI on a map and IOI is in its simplest form set by deciding the start and end points in time, see figure [2]. IOI can be more advanced by allowing repetition, i.e. answering the same query several times but with different IOI. Object types can in its simplest form just be chosen from a list, but the user often has higher demands than that. He may not be looking for all vehicles, but only for all moving vehicles, or all vehicles moving along a certain road. We need to have a way to restrict the object types and put them in relation to each other. Our solution to this is to have a structure with both object types and relations. The object types are still simply selected from a list, but after that the user can apply relations to them, thus putting restrictions on the query result.

Objects belong to all kinds of object types that can be found by the sensors, objects that have properties that may vary over time and space, e.g. vehicles, people. Objects can also be of geographical type e.g. roads, towns. The properties of the objects are important. They can be categorized as spatial, temporal or other. The spatial attributes or status values are attributes that relate to geography e.g. position or area. Temporal attributes are related to time, e.g. the point of time when it was observed or the time interval during which it exists. Other attributes concerns width, length, color, velocity etc., i.e. all conceivable attributes that are neither related to geography or time.

All objects can be affected by relations. This is the way to specify the details of the query. The type of objects is already set, but the relations delimits the answer to include only those objects that have the attributes that are wanted. The relations can be unary or binary. Unary relations are for example "color equals blue" or "velocity is greater than 50 km/h". The binary relations can be either undirected or directed. Directed means that the order of the involved objects matter, for instance the

relation "before" gives a different result depending on the order of the involved objects, whereas the result of "equals" does not.

The spatial relations can be of topological and of directional types. The global directional spatial relations includes members of the {north_of, south_of, ..., north_west_of} and the local directional spatial relations include members of the {in_front_of , behind, ...}. All theses directional spatial relations are binary directed. Temporal relations are typically Allens [11] 13 relations relating time intervals to each other, e.g. before, starts, equals, but there are also relations that relate to a specific time e.g. before 2 o'clock. Finally there are also relations that are both spatial and temporal. They are related to movement and change over time, e.g. show me anything that was there at t1 but not at t2.

B. Visualization

Visualization of the parts *where?*, *when?* and of the simple part of *what?* of the query seems relatively easy, so in this paper we have focused on the more advanced part of *what?,* i.e. the part that was described in the last section. The user interface functionality is built up around a work area and palettes. The work area is the space where the object types and the relations are placed and set in relation to each other. The palettes contain the object types and the relations, organized according to which kind of attribute it concerns, to be easily navigated.

When the user selects an object type and places it in the workspace it is visualized as a box with the type of object written inside. Then the user can select the relations that put restrictions on the objects. The relations are also visualized as boxes, where possible the relation is explained with an icon rather than in text, since that is often simpler for a human to grasp [14]. To distinguish the relations further from the objects they also have a different color on the border of the box, see figure 3.

Figure 3. The user has selected the object types vehicle and road, and the relation *inside*.

Figure 4. The query "Find all vehicles that is not inside roads" where not is used on the *inside* relation.

The user connects the object types to the relations with arrows. The location of the object types and relations in the work area is irrelevant, only the arrows matter. In the case of directed binary relations the icon of the relation is colored in two colors and the arrows from the object types to the relation is colored in the same way to make it easy for the user to understand which role each object has in the relation. Some object types are merely a support for getting the answer. In the query "find all vehicles inside roads" the vehicles are the object types that are of interest to the query, while the roads merely are support for finding the relevant vehicles. This is visualized both by making the arrow from the supporting object dotted instead of solid and by writing the type of the resulting object below the relation, see figure 5. If both objects that are participating it the relation are used the text below the relation will be the most specific object that contains both object types, e.g. bus and car gives vehicle. Relations can also be connected to other relations, thus building a more complex graph.

Figure 5. The query "Find all vehicles inside roads" where *inside* is a directional relation.

There are three logical operators; "and", "or" and "not" that have been added for the completeness of the language. The operator "not" does not really have to be used since it is always possible to use the inverse of a relation instead. If the user wants to use it, it is a property of the relation. It is visualized by a red line diagonally across the relation icon, see figure 4. The operators "and" and "or" are quite similar to each other. Since both are used frequently in more advanced queries we have added a visual shorthand notation for them. Relations that operate on the same object types can be grouped vertically instead of combining them with an "and". Similarly

relations can be grouped horizontally instead of using "or", see figure 7. Both directed and undirected relations can be combined in this way, but the user has to make sure that the objects get the correct role in each of the directed relations. Even if not all of the relations operate on the same object types they can still be grouped, but at least one object type must be the same for all of the relations. An outer box will be added around the relations that is part of the "and"/"or". The arrow for the object type that is part in all relations inside the box goes to this outer box. For all the other object types their arrows goes directly to the relation in which it is involved. This will be shown later in figure 8.
Figure 6.

Figure 7. Two equal ways of making the query "Find all vehicles within 10 km north of towns."

One thing that is common in many query languages, that we do not have, is an end-point in the graph. Instead, after the query is evaluated, the user can select any box in the graph. In this way the actual partial result in that node can be viewed, thus letting the user look around in the graph to see which relations that restrict the query a lot and which ones that restrict it only marginally, i.e. which results that include either too few or too many alternatives.

5. Illustrations

In the last chapter we described the functionality of how queries can be formulated, we also showed some simple examples. In this chapter some more advanced examples will be shown.
The first example, see figure 8, is a purely spatial query. The query is taken from [13]. The query is:

Find the supermarkets within a buffer of 0.5 km of a motorway or are outside and north of a town whose population is greater that 10000.

It contains unary relations like type restriction and population. It also contains both directed and undirected binary relations. The motorways and supermarkets are compared separately from the supermarkets and towns and then the results are merged by grouping them in an OR-relation.

Figure 8. Find the supermarkets within a buffer of 0.5 km of a motorway or are outside and north of a town whose population is greater than 10000. [13]

The second example, see figure 9, contains both spatial and temporal relations. This query is taken from [14]. The query is:

Which trucks did drive in a riverside expressway (up to) five hours before a flood?

Figure 9. Which trucks did drive in a riverside expressway (up to) five hours before a flood? [14]

In this query the motorways are first related to the rivers to find the riverside expressways. These roads are related to the trucks to find all the trucks that were driving on those roads, finally that is compared with the flood to restrict it to the trucks that drove up to five hours before the flood.

6. Conclusions

In this paper a user interface for a query language intended for sensor data sources where the sensor data are of heterogeneous type. As a consequence the query language must include facilities for sensor data fusion. Of importance is also that the query language must be sensor data independent. Sensor data independence in this context means that the user must be able to interact with the system without any knowledge about the attached sensors and their data. This means primarily that the users should be able to focus their attention on just their primary working problems and not on which sensors and sensor data that should be used for information gathering and analysis.

The structure of the user interface presented here is basically designed for the manipulation of spatial/temporal queries. The queries can be split into two parts. The first part is concerned with gathering of information from the actual sensor data while the second part is concerned with selection of information from the first step that generally depends on various conditions that are identified by the user. Thus the second part is similar to where-clauses in traditional query languages, that is where-clauses in e.g. SQL. However, the main differences between the query language described here and traditional query languages depend on the feasibility to handle spatial/temporal queries. A main problem in this context has been to develop a visual user interface with this property. At this time, however, the structure of this user interface is partly preliminary since no user tests have been made so far. Nevertheless, future research will be concerned with such a study in order to determine the usefulness of the proposed visual approach. Still, we believe that the approach taken here is quite powerful and that it can be used by both by inexperienced users with simple needs and by more experienced users with more sophisticated needs. Work is also going on to develop means that in a more powerful way will support the less experienced users, see [16].

7. References

[1] Jungert, E., Silvervarg, K. and Horney, T., *Ontology driven sensor independence in a query supported C^2-system*, Proceedings of the NATO workshop on Massive Military Data Fusion and Visualization: Users Talk with Developers, Halden, Norway, September 2002.

[2] Horney, T., *Design of an ontological knowledge structure for a query language for multiple data sources*, FOI, scientific report, May 2002, FOI-R--0498--SE.

[3] Horney, T., Jungert, E., Folkesson, M., *An Ontology Controlled Data Fusion Process for Query Language,* Proceedings of the International Conference on Information Fusion 2003 (Fusion'03), Cairns, Australia, July 8-11.

[4] *Handbook of Multisensor Data Fusion*, D. L. Hall & J. Llinas (Eds.), CRC Press, New York, 2001.

[5] Nielsen, J., *Usability Engineering*, Morgan Kaufman, New York, 1994.

[6] Silvervarg, K. and Jungert, E., *Aspects of a visual user interface for spatial/temporal queries*, Proceedings of the workshop of Visual Language and Computing, Miami, Florida, September 24-26, 2003, pp 287-293.

[7] Chang, S.-K., Costagliola, G., Jungert, E. and Orciuoli, F., *Querying Distributed Multimedia Databases and Data Sources for Sensor Data Fusion*, accepted for publication in the journal of IEEE transaction on Multimedia, 2004.

[8] Chang, S.-K. and Jungert, E., *Query Languages for Multimedia Search*, In Principals of Visual Information Retrieval, M.S. Lew (Ed.), Springer Verlag, Berlin, 2001, pp 199-217.

[9] Ahlberg, J., Klasén, L., Grönwall, C., Ulvklo, M., Jungert, E., *Automatic Target Recognition on a Multi-Sensor,* Platform Proceedings of the Swedish Symposium on Image Analysis, Stockholm, Sweden, March 6-7, 2003, pp 93-96.

[10] Dionisio, J.D.N. and Cardenas, A.F., *MQuery: a visual query language for multimedia, timeline and simulation data*, Journal of Visual Languages and Computing, v 7, n 4, Dec. 1996, p 377-401

[11] Allen, J. F., *Maintaining knowledge about temporal intervals,* Communications of the ACM, vol. 26, no. 11, pp 832-843.

[12] Chittaro, L. and Combi, C., *Visualizing queries on databases of temporal histories: new metaphors and their evaluation,* Data & Knowledge Engineering, v 44, n 2, Feb. 2003, p 239-64

[13] Abdelmoty, A. and El-Geresy, B., *Qualitative Tools to Support Visual Querying in Large Spatial Databases*, Proceedings of the workshop of Visual Language and Computing, Miami, Florida, September 24-26, 2003, pp 300-305.

[14] Bonhomme, C., Aufaure, M.-A. and Trépied, C., *Metaphors for Visual Querying of Spatio-Temporal Databases*, Advances in Visual Information Systems. 4th International Conference, VISUAL 2000. Proceedings (Lecture Notes in Computer Science Vol.1929), 2000, p 140-53

[15] Chang, S.-K., *The sentient map*, Journal of Visual Languages and Computing, Vol 11, No. 4, August 2000, pp 455-474.

[16] Chang, S.-K., Jungert, E., *Iterative Information Fusion using a Reasoner for Objects with Uninformative Belief Values*, Accepted for the 7th international Conference on Information Fusion (Fusion'04), Stockholm, Sweden, June 28- july 1, 2004.

Modeling and Retrieval of
Multimedia Data in Temporal Semantic Abstraction*

Yan Jianfeng, Tang Yan, Li Zhanhuai

Department of Computer Science & Engineering,
Northwestern Polytechnical University,
710072 Xi'an, Shaanxi, P. R. China
jfyan@mail.nwpu.edu.cn
lizhh@nwpu.edu.cn
http://www.nwpu.edu.cn

Abstract. The temporal feature is one of the most important features in multimedia data, such as video and audio. Based on the definition of Multimedia Object and Multimedia Data Unit (MDU), this paper focuses on the descriptive algorithm of generalization and aggregation, which are two kinds of important temporal semantic abstraction. Furthermore, we introduce the operating methods of our multimedia data model supporting temporal queries.

1 Introduction

Many works have been done in order to introduce temporal features into the multimedia data model. Time line is a directed descriptive method for media objects. It uses absolute time to annotate temporal features of media objects in time axis. Breiteneder proposed a model based on time line [1]. This model can distinct differentiation in expressive scene and transmitted time. However, it does not have related description for temporal relationship, either introduces temporal abstraction, therefore what can be described for semantic is limited.

OCPN [2] is able to describe comparatively simple temporal relationship like synchronism and continuity. However, this model can't provide abstract description for temporal features, and cannot support description for user's interactive operation.

Allen has concluded thirty correlations in two temporal intervals [3]. For the point of view of application, Schloss [5] has discussed a method to build layered model multimedia data, and proposed a method to operate temporal

relation through calculations of multimedia event. Little in [6] has proposed TIB (Time-Interval-Based model), which not only describes absolute temporal intervals in object, but also describes relative temporal relationships.

According to the rich semantic queries of multimedia temporal/spatial features, Megalou[7] proposed a method for multimedia temporal/spatial semantic abstraction, which divides multimedia objects into conceptual objects and presentational objects, and the formal describe the content information of multimedia information, while the latter describe temporal/spatial feature of the conceptual objects. Although this work provided an operable semantic model for temporal/spatial semantic abstractions, temporal/spatial semantic abstractions of multimedia object would change according to different application and background. And so do the related generalization layers and aggregation granularities. Therefore it is difficult to use this model to do everything.

In this paper, we will discusses description and query for absolute temporal intervals and relative temporal relationships. Through the analysis of generalization and aggregation, we also propose a structural algorithm and query method of relative temporal relationship constrains in multimedia data content information management, and these can be used in different generalization layers and satisfy different aggregation granularities.

2 The MDU Data Model

We represent a group of objects, which have certain relationships with each other, by a data structure named Multimedia Description Unit (MDU). The following is the formal definition of MDU.

Definition 1. A **Multimedia Description Unit** (MDU) is a triplet $u:<O,C,R>$. Here, O is a set of all objects that this MDU needs to describe. C is a set of all possible relationship types that maybe exist among objects. R is a subset of the mapping set $2^O \times 2^O ?$ C, which represents all possible relationships between objects.

Following Allen's definition, C consists of at least the following 13 temporal relationship elements, *equal, starts, finishes, meets, overlaps, before, during, starts^{-1}, finishes^{-1}, meets^{-1}, overlaps^{-1}, before^{-1}, durings^{-1}*. Temporal constraints exist between the temporal intervals of multimedia objects and the temporal relationships of those objects. For example, two objects that have same attribute values in temporal interval must have a temporal relationship called *equal* between them, which means that they appear and disappear at the same time.

3 Temporal Semantic of Generalization

In the temporal queries of multimedia data, besides the querying about thirteen basic temporal relationships by Allen, it is always necessary to answer the queries on generalization temporal semantics. Here, generalization is to abstract a more normal and higher layer temporal relationship from two or more basic temporal relationships. For example, the query *to find an object set that appears at the same time with object A,*

or *to find an object set that appears after object A.* These two queries do not need strict basic temporal relationships. It only needs to query through generalized temporal semantics relationship between objects.

Definition 2. Temporal relationship generalization of a multimedia description unit *u* is the generalization abstraction of temporal relationship between objects in *u,* written as $Gen_T(u, hierarchy_i(H))$, which means a new multimedia description unit *u'* that is got from *u* through temporal generalization. Here, *hierarchy_i* means a certain temporal generalization abstraction *i* in the form of a tree; *H* means a certain abstract node in this generalization abstraction. If all children of *H* are basic temporal relationship, we call *H* **direct generalization** of basic temporal relationship; if not, we call **indirect generalization**.

The temporal direct generalization is defined as:

$$u' = Gen_T(u, hierarchy_i(H)) \Leftrightarrow$$
$$u'.O = u.O, u'.C = u.C \cup H, u'.R = u.R \cup R'$$
, in which:
$$R' = \{(O_i, O_j) \to H \mid \forall O_i, O_j \in u'.O,$$
$$\exists c \in C, ((O_i, O_j) \to c) \in R \wedge H$$

is the direct generalization abstraction of *c}*

If abstract node *H* can be got through many times abstraction of basic temporal relationship *c,* such as the indirect generalization *sequential* of the basic temporal relationship *before*[(-1)] and *meets*[(-1)], temporal relationship generalization of MDU can be got through by a recursive algorithm. To a known generalization abstract structure, we can abstract the basic temporal relationship to every corresponding node in generalization structure; therefore it has been extending the description ability of MDU in the system.

4   Temporal Aggregation Semantic

In the data management of multimedia information system, another important semantic abstraction, aggregation semantics, is needed, and this semantic abstraction can aggregate some parts to a whole. For example, there is a video section, which is about the best scenes in a football game, and has described every team member as an object by MDU, including the temporal relationship between objects. However, if we want to answer the query like *to find all scenes including shooting team members and resist team member*s, we have to abstract some objects and their temporal relationship to a certain group, so we can make the objects like shooting team members and resist team members as a new shooting scene object. The essence of temporal aggregation semantics is to construct temporal relationship in higher layer granularity from temporal relationship in some lower layer granularity on temporal interval. Here, object in coarse granularity has maps to a group objects in fine granularity and the temporal relationship between them.

Definition 3. Temporal relationship aggregation in a multimedia description unit *u* is the aggregation abstraction of a set of object' sets *{O¹,O²,...,Oᵐ} on u,* (in which $\bigcup_{i=1}^{m} O^i \in u.O$), written as $Agg_T(u,\{O^1,O^2,..., O^m\})$, which represents the new multimedia description unit *u'* gotten from temporal aggregation on *u* . If the set *{O¹,O²,...,Oᵐ}* only includes one set of objects, it is named as **basic aggregation**; otherwise, it is named as **complex aggregation**. To the

basic aggregation, if $O^l=\{O_1,O_2,...,O_n\}$, then we can define:

$$u' = Agg_T(u,\{O_1,O_2,\ldots,O_n\}) \Leftrightarrow$$
$$u'.O = u.O \cup O'$$

Here, the object O' is a new object aggregateed from the set of objects $\{O_1,O_2,...,O_n\}$, whose object identifier, which is different from other identifiers, is created by the system, and the set of attributes, which is shown as following, is gotten from objects in $\{O_1,O_2,...,O_n\}$:

$$O'.ID = New(MO), Attibutes(\ O'.ID) =$$
$$\bigcup_{i=1}^{n} Attibutes(O_i.ID)(O_i \in \{O_1,O_2,\ldots,O_n\})$$

Temporal interval attribute I in the new multimedia object O' can be gotten from granularity approximation operation. Let G_i be the granularity before aggregation; and G_j be the granularity after that. The formal calculation is defined as following:

$$O'.I = \bigcup_{i=1}^{n} App_{G_i}^{G_j}(O_i.I)(O_i \in \{O_1,O_2,\ldots,O_n\})$$

New object will be created after aggregation semantic abstraction, and has extended the former temporal relationship set of multimedia description unit. Therefore, some operation is required to get the temporal relationship type set C and temporal relationship set R' of u'. The detailed operation is shown as following:

$$C'=\{aggregate, aggregate^{-1}\},$$
$$R' = \{(O_i,O') \to aggregate, (O',O_i) \to$$
$$aggregate^{-1} \mid O_i \in \{O_1,O_2,\ldots,O_n\}$$

Then, we can get a new complete multimedia description unit u' as following:

$$u'.C = u.C \cup C', u'.R = u.R \cup R'$$

5 Retrieval Based on Multimedia Temporal Abstraction

5.1 Select Operation

Select operation chooses a substructure of MDU, whose entry condition can be multimedia objects, temporal relationships or just a certain temporal condition.

Definition 4. Select operation $u'=s_F(u)$ can form a new MDU according to different condition, in which F is the entry condition represented as MDU x, and x can be defined by user or an existed MDU in system. Selection operation can get a minimal MDU u' which consists this MDU, which satisfies the following condition:

1. For the object $O_i=(oid_i,(a_1:v_1, ..., a_m:v_m))$ in any x, we can always find a mapping $O_i'=(oid_i',(a_1':v_1', ..., a_n':v_n'))$ in u', which satisfying:

$$(\forall a_k \in Attributes(oid_i))(\exists a_l \in Attributes(oid_i'))$$
$$(oid_i.a_k = oid_i'.a_l')$$

and, O_i' is the **mapping object** of O_i;

2. Relation type in x is the subset of relation type in u'. $x.C \subseteq u'.C$.

3. Relationships between objects in x and u' satisfy:

The essence of entry condition F in select operation is to provide a variable to user queries. When object relationship type set and object relationship set are empty in x and only consists of object set, select operation has transformed into an easy query, which is only focused on multimedia object, and the operation will return all needed objects

and relationship. When *x* merely consists of object relationship type set, the query has transformed into the query which is only focused on multimedia object relationship, and the operation returns all object set and relationship set satisfying the above relationship.

5.2 Temporal Projection Operation

Temporal projection operation can revise MDU according to temporal interval of given query, and the revision is based on changing temporal characteristics of multimedia object. And this will cause the change of relationship among objects, so, we can get the desired new MDU.

Definition 5. Temporal projection operation $u' = ?_F(u)$ creates new MDU under some certain entry condition *F*, which is represented by temporal interval (t_s, t_e). The operation will obtain the maximal MDU u' that can be restrained in this temporal interval, satisfying the following conditions:

1. In object set of MDU *u*, the temporal interval of all objects must be included in (t_s, t_e). If no common part exists between temporal interval attribute value of multimedia object and (t_{begin}, t_{end}), then delete this multimedia object. Otherwise,

$$\forall O_i \in u.O, O_i = (oid_i, v),$$

$$if \ oid.I.t_s < t_{begin} \ then \ oid.I.t_s = t_{begin}$$

, and,

$$\forall O_i \in u.O, O_i = (oid_i, v),$$

$$if \ oid.I.t_e > t_{end} \ then \ oid.I.t_e = t_{end}$$

2. Temporal restrain should be maintained after temporal interval has changed, therefore the relationship between objects in MDU need to be recalculated. For

$\forall ((O_i, O_j) \rightarrow c_i) \in u.R$, if $O_i \notin u.O$ or $O_i \notin u.O$, we delete the relationship from *u.R* directly, and then delete c_i from *u.C*; otherwise, we calculate c_i' according to Table 1 and Table 2, and then revise the relationship between objects as following:

$$u.C = (u.C - \{c_i\}) \cup \{c_i'\}, u.R =$$

$$(u.C - \{(O_i, O_j) \rightarrow c_i\}) \cup \{(O_i, O_j) \rightarrow c_i'\}$$

5.3 Joining Operation

Data independence is a basic characteristic that data management system should provide. For multimedia data, the advantage of providing data independence is to share and reuse data, which means basic multimedia source material can reform new multimedia document. The joining operation can combine different MDUs into a new MDU.

Definition 6. The joining operation $u' = u_1 \mapsto u_2$ means that u_2 is joined after u_1, in which all objects and the relationship between them in u_1 don't change, and the relationship between objects in u_2 don't change. Temporal interval attribute value of all objects need move from left to right temporal point in all objects of u_1, and we can obtain a new MDU that satisfies the following condition:

Any object $O_i = (oid_i, v)$ and the relationship between them in u_1 satisfy:
$$(\forall O_i \in u_1.O)(O_i \in u'.O) \wedge (\forall c \in u_1.C)$$

$$(c \in u'.C) \wedge (\forall r \in u_1.R)(r \in u'.R)$$

Any object $O_j = (oid_j, v)$ and the relationship between them in u_2 satisfy:
$$(oid_j.I.t_s = oid_j.I.t_s)$$

$$+ Temp\_R(u_1) - Temp\_L(u_2),$$

$$(oid_j.I.t_e = oid_j.I.t_e)$$
$$+ Temp\_R(u_1) - Temp\_L(u_2),$$
and,
$$(O_j \in u'.O) \wedge (\forall c \in u_2.C)(c \in u'.C)$$
$$\wedge (\forall r \in u_2.R)(r \in u'.R)$$

u' is the biggest MDU that satisfies the above conditions.

6   Conclusion

Multimedia temporal features management and temporal semantic abstraction are the focal problems that need to be solved in the research domain of multimedia data management. Based on absolute temporal management and temporal relationship research, this paper proposes a multimedia data model based on multimedia object description, and discusses the descriptive algorithm of generalization and aggregation, which are the two important temporal semantic abstractions. Finally, we introduce a multimedia temporal query method based on the above description.

Our future researches will be focused on the realizing of the above model, including temporal query processing, query optimizing and the design of query language of multimedia data.

References

1. C.Breiteneder, S.Gibbs and D.Tsichritzis, Modeling of Audio/Video Data, In proceedings of the 11th International Conference on the Entity-Relationship Approach, Karlsruhe, Germany, October 1992, pp 322-339

2. T.D.C.Little, A digital On-Demand Video Service Supportin Content-based Queries, In Proceedings of ACM Multimedia 1993, Anaheim, USA, August 1993. pp427-436

3. J.F.Allen, Maintaining Knowledge about Temporal Intervals, Communications of the ACM, 26(11), November 1983, pp832-843

5. G.A. Schloss, M.J. Wynblatt, Building Temporal Structures in a Layered Multimedia Data Model, In Proceedings of ACM Multimedia 1994, San Francisco, USA, August 1994.pp271-278

6. T.Little, A. Ghafoor, Interval-based conceptual models for time-dependent multimedia data, IEEE Transaction on Knowledge and Data Engineering, 1993, 5(4). Pp551-563

7. E. Megalou, T. Hadzilacos, Semantic Abstractions in the Multimedia Domain, IEEE Transactions on Knowledge Data Engineering, 2003, 15(1). pp136-160

Smart Decision Module for Streaming 3D Meshes over Lossy Networks

Hui Li, Parinkumar Shah, B. Prabhakaran

hxl015300,parinshah,praba@utdallas.edu

Department of Computer Science

University of Texas at Dallas, Richardson, TX 75083

Abstract— Existing 3D multi-resolution compression techniques assume no loss during the transmission. The reliable transport protocol such as TCP seems to be an excellent solution. However, time sensitive applications find TCP suffering long delay in the presence of network congestions. This motivates us to use the unreliable transport protocol such as UDP. Yet UDP introduces the distortion between the constructed mesh and the original mesh due to the packet loss. In this paper, we modified CPM (Compressed Progressive Meshes) and proposed a smart decision module. The smart decision module can intelligently select the transport protocol for each geometric sub-layer to achieve the minimum delay and distortion under a given network bandwidth and loss ratio. Simulations show that this smart decision module can correctly evaluate the transmission time and expected distortion and thus make the optimal transport decision.

Index Terms— Compressed Progressive Meshes,MPEG,VRML

I. INTRODUCTION

Real time 3D applications such as distributed 3D games, 3D animations and interactive Virtual Reality require a huge amount of data transmitted in a certain time frame. For example, the uncompressed "Happy Buddha" model is about 42 MB [1]. The existing Internet has two major limitations on 3D streaming: the delay caused by limited bandwidth and the distortion by packet loss. Therefore, it is desirable to compress 3D models first and then transmit over Internet. Different compression techniques for 3D meshes have been proposed. They can be broadly categorized into:

- *Single-resolution technique:* For single resolution, a 3D mesh is compressed into a single representation with high compression ratio. However, single-resolution techniques are not suitable for the network transmission because the rendering process starts only after the entire mesh is downloaded.
- *Multi-resolution technique:* Multi-resolution compression techniques include Hoppe's PM [2] [3], Taubin's PFS [4] and Pajarola's CPM [5]. Multi-resolution schemes compress the original mesh into the base mesh and a sequence of refinements. The size of the base mesh is relatively small compared with the original mesh. The rendering process begins after receiving the base mesh. The following refinements improve the model progressively.

CPM assumes no transmission loss and does not have error recovery mechanism. TCP meets this requirement. Any lost

Fig. 1. CPM edge collapse and vertex split. The vertex spanning tree is shown with thick lines.

packet is retransmitted. However, TCP suffers long delay in the presence of packet loss due to congestion control mechanism. This is not suitable for real time applications and motivates to use UDP. However, UDP is not reliable and the distortion increases due to the packet loss. Chen [6] and Ghassan [7] transmitted the base mesh and high priority refinements via TCP and the rest via UDP. They found that for certain combinations, the transmission time is improved with little degradation in visual effect. However, the transport scheme of Chen [6] and Ghassan [7] is at the batch level: the entire batch is transmitted via TCP/UDP and this transport scheme is not optimal.

Proposed Approach: To alleviate this problem, we propose:

- A modified CPM(MCPM) model that will help in identifying the average distortion encountered at the client side at the sub-layer level while rendering the received 3D model.
- A hybrid transmission protocol with a smart decision module to transmit 3D mesh compressed by MCPM.

MCPM decomposes the geometric data into sub-layers based on the important digit. With this decomposition, we can evaluate the distortion of each sub-layer. In our hybrid transport protocol, we transmit the base mesh and structural data over TCP. The proposed smart decision module makes decision on which transport protocol to use for each geometric sub-layer based on the relative importance. This decision module can achieve the minimum delay and distortion under a given network bandwidth and loss ratio. The transport scheme is not restricted to TCP/UDP and can be generalized to RTSP/RTP in MPEG-4 standard [8].

II. COMPRESSED PROGRESSIVE MESHES

CPM representation of a mesh is stored as the base mesh M_0 and subsequent batches M_i. The base mesh is a coarse representation of the original mesh. At compression, a vertex spanning tree of the current mesh is constructed and a number

Fig. 2.   (a)CPM and MCPM representations of batch M_i. "S" for structural and "G" for geometric data. (b)CPM and MCPM representations of the geometric data in a batch. The transmission order is shown with arrowed lines.

of edges are selected for "edge collapsing" transformation. "edge collapsing" collapses the two ending points of the selected edge to the midpoint. The reverse transformation at decompression is called "vertex split". A new vertex is inserted into the mesh. Figure 1 shows the base mesh M_0 and the mesh with M_1. From M_1 to M_0, two edges A'A" and F'F" are collapsed to their midpoints A and F respectively. By applying "edge collapsing", one vertex is removed from the current mesh and the midpoint is marked as "1" in the vertex spanning tree. These "edge collapsing" transformations are then encoded into one batch with the structural and geometric data as shown in Figure 2(a). These information is used for "vertex split" at decompression. The structural data contains the indices of the two cut-edges. In Figure 1, edge e and f are the two cut-edges for vertex A and edge g and h are for vertex F. With the structural data, "vertex split" knows how to connect the new vertex with the neighbors. For example, new vertex A" is connected with vertex C and P. The geometric data contains the error vector E between the estimated split-vector SV' and the original split-vector SV. The split-vector is defined as the vector between two ending points for "edge collapsing": $A' - A''$ and $F' - F''$ in Figure 1. With the split-vector and the split vertex, the positions of the two ending points for "vertex split" can be completely determined. For example, the positions of A', A" for the "vertex split" of A can be obtained by $A = \frac{A'+A''}{2}$ and $SV = A' - A''$. The original split-vector is obtained by adding the error vector in the geometric data to the estimated split-vector, i.e.,$SV = SV' + E$. The estimated split-vector is computed with Butterfly prediction [5].

When CPM representation is streamed over lossy networks, some information could be lost. Since the amortized size of geometric data is about twice of the structural data [5], it is optimal to carefully select the transport format for the geometric data.

III. MCPM

In CPM, if the structural data is lost during the transmission, the decoder cannot recover from this error and the mesh could crash. Therefore, we transmit the base mesh and the structural data over TCP. The only part left is the geometric data. Proposed MCPM decomposes the geometric data into sub-layers and leaves the structural data unchanged as shown in Figure 2(a). With MCPM, we can evaluate the distortion and make the transport decision for each sub-layer instead of the entire batch with CPM.

A. Important digit

When we transmit 3D meshes in CPM over UDP, geometric data is chopped into packets and sent over networks. Because UDP is not reliable, some packets are dropped during the transmission. The receiver has no choice but uses the predicted values. This introduces the distortion between the constructed mesh and the original mesh. We define the average distortion $\bar{D}$ at the receiver as

$$\bar{D} = \sum_{i=1}^{B} \frac{m(i)}{N(i)} \times (D[i] - D[i+1]) \qquad (1)$$

Where B is the number of batches, N(i) is the number of packets and m(i) is the number of lost packets during the transmission of batch i. D[i] is the distortion of batch i. Equation(1) gives the average distortion $\bar{D}$ based on the distortion of each batch $D[i]$.

In MCPM, the geometric data is decomposed into sub-layers based on the "important digit" position as shown in Figure2(b). The error vector of a split vertex v_i consists of three coordinates (x_i, y_i, z_i). We assume each coordinate has at most m digits after quantization, i.e., $x_i = x_{i1}x_{i2}x_{i3}x_{i4}.x_{im}$ (here we only show x coordinate, the same transformation is applied to y and z). We take x_{i1} as the first important digit, x_{i2} as the second and so on. Therefore, there are m important digits. Then we put the ith important digit of all split vertices in one batch into sub-layer i. If a vertex does not have this digit, we use a special symbol to represent it. Figure 2(b) shows how to decompose the geometric data in a batch with N split vertices into m sub-layers. The data layout for MCPM is different from CPM. For CPM, the order of the transmission is from the top to the bottom as shown in Figure 2(b), i.e., $(x_1, y_1, z_1...x_N, y_N, z_N)$. For MCPM, the formation of a sub-layer is from the top to the bottom for each important digit. The order of the transmission is from the left to the right, i.e., from sub-layer one to sub-layer m as shown in Figure 2(b). Equation (1) can be modified to calculate the average distortion based on the distortion of each sub-layer by changing D[i] to D[i] [j], N(i) to N(i,j) and m(i) to m(i,j) where j is the index of the sub-layer. Equation(1) can be rewritten as

$$\bar{D} = \sum_{i=1}^{B} \sum_{j=1}^{S} \frac{m(i,j)}{N(i,j)} \{D[i][j] - D[i][j+1]\} \qquad (2)$$

where S is the number of sub-layers for each batch. Next, we discuss how to compute D[i][j].

B. Distortion of Geometric Sub-layer

In this section, we dicuss the computation of the distortion caused by the loss of sub-layer j of batch i, i.e., D[i][j]. We denote D[i][j] as the average distortion caused by all split vertices in this batch as

$$D[i][j] = \frac{\sum_{k=1}^{N_i} d(k,j)}{N_i} \qquad (3)$$

where N_i is the number of split vertices in this batch. d(k,j) is the expected distortion caused by split vertex k if sub-layer j is lost and all sub-layers before j are received. Because sub-layer

(a) Bufferfly prediction   (b) Dependence tree

Fig. 3. MCPM dependence tree and Bufferfly prediction

(a) Distortion   (b) Relative importance

Fig. 4. Stanford "bunny"

C. Relative Importance

Figure 4(b) shows the relative importance for Stanford "bunny" model. Note that relative importance does not monotonically decrease. Sub-layer in the finer batch can have larger relative importance. This indicates that it is not optimal if the entire batch is transmitted via TCP/UDP as in Chen [6] and Ghassan [7]. The transport decision should be made for each sub-layer according to the relative importance value. In the next section, we discuss the smart decision module.

IV. SMART DECISION MODULE

We have two assumptions about the networks. First, the underlying network is a QoS network. It has separate admission control modules for TCP/UDP requests since UDP is not TCP-friendly. With this assumption, the network reserves certain bandwidth for TCP/UDP. Second, the network drops packets randomly with a constant loss ratio.

A. Smart Decision Module

Table I shows input parameters to the smart decision module. The objective is to decide which transport protocol to use for each sub-layer in order to minimize the delay and distortion. First we sort sub-layers into a list on relative importance in decreasing order. We call this list as relative importance list. As in Figure 4(b), it is possible that in the relative importance list, sub-layers in the finer batch appear before sub-layers in the coarse batch. Then, we define the cost function as:

$$Cost(x) = transmission(x) + \alpha \times distortion(x) \quad (5)$$

where x is the index of the sub-layer in the relative importance list and α is the weight of the distortion. This cost function computes the total effect of the transmission time and expected distortion when first x sub-layers in the relative importance list are transmitted over TCP and the rest over UDP. The decision module iterates all possible x and chooses the x with the minimum Cost.

1) Transmission Time: X partitions the relative importance list into two parts: first x sub-layers through TCP and the rest through UDP. The transmission time is expressed as

$$Transmission = \sum_{m=1}^{x} \frac{size(m)}{t} + \sum_{m=x+1}^{B \times S} \frac{size(m)}{u} \quad (6)$$

Based on our network assumptions, UDP rate is a constant. However, TCP rate is not a constant although certain bandwidth t is reserved. To remedy this problem, we use average

j contains the jth important digit of the error vector, the ratio of the expected distortion introduced by this sub-layer to the previous sub-layer j-1 is a constant β. With β, we only need to compute d(k), the distortion due to the loss of error vector for vertex k and then distribute this distortion to sub-layers in this batch with equations: $d(k) = \sum_{j=1}^{S} d(k,j)$ and $d(k, j+1) = \beta d(k,j)$, where $1 \leq j \leq S-1$. Due to the dependence across batches, the distortion is propagated from the current batch to the finer batches as stated in Ghassan [9]. For example, in Figure 3(a), V is split to A and B in the current batch and A/B may be used to compute the estimated split-vector in the subsequent batches. To calculate such propagation, first, we construct the dependence tree shown in Figure 3(b) as follows. The computation of the estimated split-vector involves all vertices in Figure 3(a). However, different vertices have different effects depending on the connectivities with A and B. If a vertex lies in this neighborhood, we draw a line from this vertex to the split vertex in the dependence tree with a case number derived from Bufferfly interpolation [5]. The calculation is processed from batch one to the finest batch. Then we calculate d(k) with the formula below:

$$d(k) = \sum_{l=i+1}^{B} \sum_{u \in \Gamma_u} D_u \quad (4)$$

Where vertex set Γ_u includes vertices affected by k. D_u is obtained by tracing the link from vertex k to vertex u on the dependence tree from the next batch to the finest batch. This computation is performed at the pre-processing stage.

Figure 4(a) shows D[i][j] for Stanford "bunny" model [1]. MCPM has nine batches and each batch has four sub-layers. As shown in Figure 4(a), the distortion decreases monotonically. Note the amount of distortion reduced by each sub-layer is different. We define the amount of distortion reduced by a sub-layer divided by the size of this sub-layer as "relative importance". Relative importance value of a sub-layer represents the benefit we can obtain if this sub-layer is correctly transmitted.

Fig. 5. Transmission time for l=0.04,0.1 and 0.12

Fig. 6. Expected distortion for l=0.04,0.1 and 0.12

TCP rate t'. t' is obtained by observing the average TCP rate when streaming the base mesh. Also, Equation(6) can be used to determine x when the transmission time is specified.

2) Expected Distortion: The expected distortion is given by Equation(2). We need to evaluate the number of packets received for each sub-layer. Here we assume sub-layers transmitted via TCP do not have any loss. For sub-layer j with N(j) packets over UDP, the expected distortion can be expressed as:

$$ED(j) = \sum_{m=0}^{N(j)} \frac{mProb(m)}{N(j)}(D[i][j] - D[i][j+1]) \quad (7)$$

Prob(m) is the probability that m packets are lost. Since the network drops packets randomly, Prob(m) is a binomial distribution and the expected value is $l \times N(j)$. Substitute it into Equation(7), we have $ED(j) = l \times (D[i][j] - D[i][j+1])$. Sum ED(j) on all sub-layers via UDP, we obtain the expected distortion. Equation(7) can be used to compute x if expected distortion is specified.

V. Simulation Results

The testbed consists of a server, a client and an intermediate node installed with dummynet [10]. We configured the dummynet node as a gateway and all traffic is captured by this node. We set forwarding rules of dummynet to emulate the network bandwidth and loss ratio.

Figure 5 shows the comparison of the transmission time between simulations and Equation(6). It is obvious that the transmission time increases as more sub-layers are transmitted over TCP. The reason is that the average TCP rate is much lower than UDP. Note the increase of transmission time is not linear due to the larger data sizes of finer batches.

Figure 6 shows the comparison of the expected distortion between simulation and Equation(7). The distortion decreases as more sub-layers are transmitted via TCP. Note the sharp decrease of the expected distortion when the first several sub-layers in the relative important list are transmitted over TCP. This is because the first several sub-layers have larger relative

Fig. 7. Optimal number of sub-layers via TCP

importance values.

Figure 7 shows the optimal number of sub-layers in the relative importance list is transmitted over TCP with different α. As α increases, the number of sub-layers through TCP increases. The reason is that larger α means more preference on the distortion and thus requires more sub-layers through TCP. Note given the same α, more sub-layers are transmitted over TCP for l=0.1 than l=0.12 and this is due to the relatively small TCP rate for l=0.12. Therefore, loss ratio affects both TCP rate and the expected distortion. As loss ratio increases, it is not optimal to use TCP to transmit more sub-layers.

VI. Conclusions and discussions

3D streaming techniques have been developed rapidly during the last four years. Different compression schemes, transport protocols and error protection techniques have been proposed and some of them have been accepted as standards in VRML or MPEG. However, due to the bandwidth and loss limitations, there is still work to be done. In this paper, we proposed a smart decision module for streaming 3D models in MCMP format over QoS networks. The smart decision module can select the transport protocol for geometric sub-layers to achieve the minimum delay and distortion.

References

[1] "Large geometric models archive," Georgia Institute of Technology. [Online]. Available: http://www.cc.gatech.edu/projects/large_models
[2] H. Hoppe, "Progressive meshes," in *Proceedings SIGGRAPH 96*, 1996.
[3] H.Hoppe, *Efficient Implementation of Progressive Meshes*, Technical Report MSR-TR-98-02, Microsoft, 1998.
[4] G.Taubin, A. Gueziec, W. Horn, and F. Lazarus, "Progressive forest split compression," *Siggraph'98 Conference Proceedings*, July 1998.
[5] R.Pajarola and J.Rossignac, "Compressed progressive meshes," *IEEE Trans. Visual. Comput. Graphics*, vol. 6, pp. 79–93, Jan. 2000.
[6] Z. Chen, B.Bodenheimer, and J. Barnes, "Robust transmission of 3d geometry over lossy networks," in *Proceeding of the eighth international conference on 3D web technology*, 2003.
[7] G.Al-Regib and Y. Altunbasak, "3tp:an application-layer protocol for streaming 3-d graphics," ICME, 2003.
[8] *MPEG4 International Standard*, Moving Picture Experts Group Std. ISO/IEC 14 496:1998, 1998.
[9] G. Al-Regib and Y. Altunbasak, "An unequal error protection method for packet loss resilient 3-d mesh transmission," INFOCOM, 2002.
[10] L.Rizzo, "Dummynet: a simple approach to the evaluation of network protocols," *ACM SIGCOMM Computer Communication Review*, Jan. 1997.

Generating Proofs with Spider Diagrams Using Heuristics

Jean Flower, Judith Masthoff and Gem Stapleton

Visual Modelling Group

University of Brighton

Brighton, UK

Email: {j.a.flower,judith.masthoff,g.e.stapleton}@brighton.ac.uk

Abstract— We apply the A^* algorithm to guide a diagrammatic theorem proving tool. The algorithm requires a heuristic function, which provides a metric on the search space. In this paper we present a collection of metrics between two spider diagrams. We combine these metrics to give a heuristic function that provides a lower bound on the length of a shortest proof from one spider diagram to another, using a collection of sound reasoning rules. We compare the effectiveness of our approach with a breadth-first search for proofs.

I. INTRODUCTION

Simple diagrammatic systems that inspired spider diagrams are Venn and Euler diagrams. In Venn diagrams all possible intersections between contours must occur and shading is used to represent the empty set. Diagram d_1 in Fig. 1 is a Venn diagram. Venn-Peirce diagrams [11] extend the Venn diagram notation, using additional syntax to represent non-empty sets. Euler diagrams exploit topological properties of enclosure, exclusion and intersection to represent subsets, disjoint sets and set intersection respectively. Spider diagrams [4], [7], [8],

Fig. 1.   A Venn diagram and a spider diagram.

[10] are based on Euler diagrams. *Spiders* are used to represent the existence of elements and *shading* is used to place upper bounds on the cardinalities of sets. A spider is drawn as a collection of dots joined by lines. The spider diagram d_2 in Fig. 1 expresses the statement "no mice are cats or dogs, no dogs are cats, there is a cat and there is something that is either a mouse or a dog". Sound and complete reasoning rules for spider diagram systems have been given [7], [8], [10].

As argued in [3], it is important for automated diagrammatic reasoning systems to produce proofs that are easy to understand by humans. For this reason, our ambition is to produce diagrammatic proofs using diagram transformations instead of converting to first order logic and using existing theorem provers. An existing theorem prover for spider diagrams successfully writes proofs [2], but they can be long and unwieldy. In [3], we presented a new approach to proof writing in diagrammatic systems, which is guaranteed to find shortest

proofs and can be extended to incorporate other readability criteria. We applied the A^* algorithm and developed an admissible heuristic function to guide automatic proof construction. However, the work presented in [3] was limited to the simple case of so-called unitary spider diagrams. Here, we extend that work to the significantly more challenging general case of so-called compound spider diagrams.

II. SPIDER DIAGRAMS

We now informally introduce the spider diagram system.

A. Syntax and semantics of spider diagrams

In this section, we will give an informal description of the syntax and semantics of spider diagrams. Details and formal definitions can be found in [8]. A **contour** is a labelled closed curve in the diagram used to denote a set. The **boundary rectangle** is an unlabelled rectangle that bounds the diagram and denotes the universal set. A **zone**, roughly speaking, is a bounded area in the diagram having no other bounded area contained within it. A zone can be described by the set of labels of the contours that contain it and the set of labels of the contours that exclude it. A zone denotes a set by intersection and difference of the sets denoted by the contours. A **region** is a set of zones.

A **spider** is a tree with nodes, called **feet**, placed in different zones. A spider **touches** a zone if one of its feet appears in that zone. The set of zones a spider touches is called its **habitat**. A spider denotes the existence of an element in the set represented by its habitat. Distinct spiders represent the existence of distinct elements. A zone can be **shaded**. In the set represented by a shaded zone, all of the elements are represented by spiders. So, a shaded zone with no spiders in it represents an empty set. A **unitary diagram** is a finite collection of contours (with distinct labels), shading and spiders properly contained by a boundary rectangle.

The unitary diagram d_2 in Fig. 1 contains three contours and five zones, of which one is shaded. There are two spiders. The spider with one foot inhabits the zone inside (the contour labelled) $Cats$, but outside $Dogs$ and $Mice$. The other spider inhabits the region which consists of the zone inside $Mice$ and the zone inside $Dogs$ but outside $Cats$.

Unitary diagrams form the building blocks of **compound diagrams**. To enable us to present negated, disjunctive and

conjunctive information between unitary diagrams, we use connectives: $\neg$, $\sqcup$ and $\sqcap$. If D_1 and D_2 are spider diagrams then so are $\neg D_1$ ("not D_1"), $D_1 \sqcup D_2$ ("D_1 or D_2") and $D_1 \sqcap D_2$ ("D_1 and D_2"). The semantics of compound diagrams extend those of unitary diagrams in the obvious way.

B. Reasoning with spider diagrams

We now give informal descriptions of the sound but not a complete set of reasoning rules for spider diagrams. For formal descriptions see [8].

Add contour. A new contour can be added to a unitary diagram. Each zone is split into two zones (one inside and one outside the new contour) and shading is preserved. Each spider foot is replaced by a connected pair of feet, one in each of the two new zones. For example, in Fig. 2, d_2 is obtained from d_1 by adding a contour. This rule is reversible and we will refer to its reverse as **Delete contour**.

Add shaded zone. A new, shaded zone can be added to a unitary diagram. This rule is reversible and we will refer to its reverse as **Delete shaded zone**. For example, in Fig. 2, diagram d_3 is obtained from d_2 by deleting a shaded zone.

Erase shading. Shading can be erased from any zone in a unitary diagram.

Delete spider. A spider whose habitat is completely non-shaded can be deleted from a unitary diagram.

Add spider foot. In a unitary diagram, a foot can be added to a spider in a zone it does not yet touch.

Fig. 2.   Applications of "add contour" and "delete shaded zone".

All of the remaining diagrammatic rules are reversible.

Split spider. A unitary diagram d containing a spider s whose habitat has a partition into non-empty regions r_1 and r_2 can be replaced by $d_1 \sqcup d_2$, where d_1 and d_2 are copies of d except that the habitat of s is reduced to r_1 in d_1 and r_2 in d_2. For instance, diagram d in Fig. 3 has a spider with two feet. We can split this spider into two parts, giving $d_1 \sqcup d_2$.

Fig. 3.   An application of rule "split spider".

Excluded Middle. A unitary diagram d with a non-shaded zone z can be replaced by $d_1 \sqcup d_2$, where d_1 and d_2 are copies of d except that z is shaded in d_1 and contains an additional spider in d_2. For instance, d in Fig. 4 has a non-shaded zone $B - C$. Applying excluded middle to this zone yields $d_1 \sqcup d_2$.

Fig. 4.   An application of rule "excluded middle".

An α**-diagram** is a diagram in which each spider has exactly one foot. Two unitary α-diagrams with (essentially) the same zone set are in **contradiction** if a zone is shaded in one diagram and contains more spiders in the other. For example, in figure 4, the diagrams d_1 and d_2 are in contradiction because $B - C$ is shaded and contains no spiders in d_1 but contains one spider in d_2.

Fig. 5.   An application of rule "Combining".

Combining. A compound diagram consisting of the conjunction of two non-contradictory α-diagrams d_1 and d_2 whose zone sets are the same can be combined into a single unitary diagram, d_3, with the same zone set. The number of spiders in any zone in d_3 is the maximum number of spiders in that zone in d_1 and d_2, and a zone is shaded in d_3 if it is shaded in either d_1 or d_2. For example, diagram $d_1 \sqcap d_2$ in Fig. 5 can be replaced by d_3. We note here that due to the non-deterministic nature of the reverse of this rule, we have not included the reverse in our implementation.

There are also reasoning rules that have analogies in logic. We include in our set of rules **Idempotency** (for example, $D \vdash D \sqcap D$), **De Morgan's laws**, **Involution** ($\neg\neg D \equiv_\vdash D$) and **Distributivity**. All of these rules are reversible. We also include the **Absorption** rules which state that $D_1 \sqcap (D_1 \sqcup D_2)$ can be replaced by D_1 and $D_1 \sqcup (D_1 \sqcap D_2)$ can be replaced by D_1. Whilst the reverses of the absorption rules are sound, due to the non-deterministic nature of the reverses these have not been included in our implementation.

If D_1 can be transformed into D_2 by a reversible rule, then any occurrence of D_1 in a compound diagram can be replaced by D_2. If D_1 can be transformed into D_2 by a non-reversible rule, then any occurrence of D_1 in a compound diagram can be replaced by D_2, provided the occurrence of D_1 being replaced is 'inside an even number of not's'. For example, in the diagram $\neg((D_1 \sqcap D_3) \sqcup (D_4 \sqcap \neg D_1))$ the first occurrence of D_1 is not inside an even number of not's, but the second is. We say that diagrams D_2 is **obtainable** from diagram D_1, denoted $D_1 \vdash D_2$, if and only if there is a sequence of diagrams $\langle D^1, D^2, ..., D^m \rangle$ such that $D^1 = D_1$, $D^m = D_2$ and, for each k where $1 \leq k < m$, D^k can be transformed into D^{k+1} by a single application of one of the reasoning rules. Such a sequence of diagrams is called a **proof** from **premise** D_1 to **conclusion** D_2.

III. A* APPLIED TO PROOF WRITING

To construct a proof, a rule needs to be applied to the premise diagram, followed by another rule to the resulting diagram, and so on, until the conclusion diagram is reached. At any stage, multiple rules might be applicable. The problem of deciding which rule to apply is an example of a more general class of so-called search problems, for which various algorithms have been developed (see [9] for an overview). A^* is a well known search algorithm [5].

A^* stores an ordered sequence of proof attempts. Initially, this sequence only contains a zero length proof attempt, namely the premise diagram. Repeatedly, A^* removes the first proof attempt from the sequence and considers it. If the last diagram of the proof attempt is the conclusion diagram, then an optimal proof has been found. Otherwise, it constructs additional proof attempts, by extending the proof attempt under consideration, applying rules wherever possible to the last diagram.

The effectiveness of A^* and the definition of "optimal" is dependent upon the ordering imposed on the proof attempt sequence. The ordering is derived from the sum of two functions. One function, called the **heuristic**, estimates how far the last diagram in the proof attempt is from the conclusion diagram. The other, called the **cost**, calculates how costly it has been to reach the last diagram from the premise diagram. The new proof attempts are inserted into the sequence, ordered according to the cost plus heuristic. A^* always finds the solution with the lowest cost, if one exists, provided the heuristic used is **admissible** [1]. A heuristic is admissible if it is **optimistic**, which means that it never overestimates the cost of getting from a premise diagram to a conclusion diagram. We define all rules to have a cost equal to one, which means that any admissible heuristic gives a lower bound on the number of proof steps needed in order to reach the conclusion diagram.

The amount of memory and time needed by A^* depends heavily on the quality of the heuristic used. For instance, a heuristic that is the constant function zero is admissible, but will result in a breadth-first search of the state space, giving long and impractical searches. The better the heuristic (in the sense of accurately predicting the lowest cost of a proof), the less memory and time are needed for the search.

IV. THE HEURISTIC FUNCTION

To define the heuristic function, we capture differences between the premise diagram and the conclusion diagram to give an estimate of the length of a shortest proof. In [3], we proposed several metrics to capture differences between two unitary diagrams, focussing on the difference in contour sets, zone sets, shaded zone sets, and spiders. These metrics were combined to provide a heuristic function for unitary diagrams. We will use similar metrics to judge the similarity between compound diagrams, in addition to new metrics to capture differences in the structure of a compound diagram. First, we must determine what we mean by the contour set of a compound diagram (and similarly for zones, etc). Perhaps surprisingly, the most useful approach is not the obvious one:

to take the union of the sets of contours of the unitary parts. We now illustrate why this naive approach is not useful. Suppose we were to take the union of the contour sets of a diagram's unitary components as a measure of the contours in that diagram and to define the cardinality of the symmetric difference of the contour sets for D_1 and D_2 as our contour difference metric between D_1 and D_2. Such a metric should be good at guiding applications of the Add Contour and Delete Contour rules. Assume $D_1 \sqcap (D_1 \sqcup D_2)$ is our premise diagram, and D_1 our conclusion diagram. The diagram D_1 could have a vastly different contour set to D_2 (and, therefore, to $D_1 \sqcap (D_1 \sqcup D_2)$). Using the absorption rule, the premise diagram can be changed into the conclusion diagram in one step. Hence, for admissibility to hold, the heuristic must be at most 1 but the cardinality of the symmetric difference of the contour sets may be much larger than 1. We would have to cap the metric at 1, and this would will lead to a weak heuristic when we need to apply the Add/Delete Contour rules many times. Therefore, we will not use a simple union.

Actually, we would prefer a contour measure to be invariant under all the logic rules, and only to reflect the need for applications of the Add and Delete Contour rules. Each measure is designed to be invariant under many rules (if we apply a rule then the measure remains unchanged) and variant under few rules (if we apply a rule then the measure changes). So, we would prefer a measure of the contour set of $D_1 \sqcap (D_1 \sqcup D_2)$ to be the same as that of D_1. We would like to define $Contours(D_1 \sqcap D_2)$ and $Contours(D_1 \sqcup D_2)$ in such a way as to achieve this. There are two obvious operations which can be done on sets: union and intersection. If we define $Contours(D_1 \sqcap D_2)$ as the union of $Contours(D_1)$ and $Contours(D_2)$, and $Contours(D_1 \sqcup D_2)$ as the intersection of $Contours(D_1)$ and $Contours(D_2)$, then:

$$Contours(D_1 \sqcap (D_1 \sqcup D_2)) =$$
$$Contours(D_1) \cup Contours(D_1 \sqcup D_2) =$$
$$Contours(D_1) \cup (Contours(D_1) \cap Contours(D_2)) =$$
$$Contours(D_1).$$

A similar result can be achieved by performing these operations the other way around (i.e., using intersection for conjunction, and union for disjunction). We will call the contour set obtained using the first definition m_1 (m for measure), and the contour set obtained using the second definition m_2. Both definitions are required because we have explicit negation of diagrams in our system. To make sure the contour sets are invariant under involution (and De Morgan's laws) we use $m_1(\neg D_1) = m_2(D_1)$ (and similarly, $m_2(\neg D_1) = m_1(D_1)$).

A similar approach can be taken to define measures for zones, shaded zones and spiders. To avoid repetition and show invariance under the logic rules, in the next section we will generalize this approach.

A. Building a set of independent metrics

To define our metrics, we first define various measures on diagrams. As discussed previously, a useful measure of the

contours in a diagram should be invariant under the logic rules but variant under the Add/Delete Contour rules in order to steer the proof writer towards applying the Add/Delete Contour rules when they are required.

In this section, we describe generic measures and show how these are invariant under logic rules. Intuition about these generic measures may be gained by comparing them with the specific example at the beginning of this section.

If D is a diagram, define a pair of measures $m_1(D)$ and $m_2(D)$ recursively using families of n-ary functions with domain X^n (X will be determined by the context) $g_{1,n} \colon X^n \to X$ and $g_{2,n} \colon X^n \to X$ (where $m_i(D) \in X$). For instance, in the example for contours given above, $g_{1,n}$ takes the union of n sets and $g_{2,n}$ takes the intersection of sets. We start by defining

$$m_i(\neg D) = m_j(D)$$

($i \neq j$) which ensures that m_1 and m_2 are invariant under involution. For example,

$$m_1(\neg\neg D) = m_2(\neg D) = m_1(D).$$

We extend the definition as follows:

$$m_i(D_1 \sqcap ... \sqcap D_n) = g_{i,n}(m_i(D_1), ..., m_i(D_n))$$

$$m_i(D_1 \sqcup ... \sqcup D_n) = g_{j,n}(m_i(D_1), ..., m_i(D_n))$$

where $j \neq i$.

By observing the subscripts i and j, we can see that these definitions already guarantee that the measures m_1 and m_2 are invariant under De Morgan's laws. For example

$$
\begin{aligned}
m_1(\neg(D_1 \sqcup D_2)) &= m_2(D_1 \sqcup D_2) \\
&= g_1(m_2(D_1), m_2(D_2)) \\
&= g_1(m_1(\neg D_1), m_1(\neg(D_2)) \\
&= m_1(\neg D_1 \sqcap \neg D_2).
\end{aligned}
$$

To ensure invariance under commutativity and associativity, we require functions which satisfy
$$g_{i,3}(x, y, z) = g_{i,2}(x, g_{i,2}(y, z))$$
and
$$g_{i,2}(x, y) = g_{i,2}(y, x),$$
for variables x, y and z. Provided that
$$g_{i,2}(x, g_{j,2}(y, z)) = g_{j,2}(g_{i,2}(x, y), g_{i,2}(x, z)),$$
we have invariance under distributivity laws. Finally, consider the absorption laws. For invariance, we need
$$g_{i,2}(x, g_{j,2}(x, y)) = g_{i,1}(x).$$

This set of conditions on the n-ary functions $g_{1,n}$ and $g_{2,n}$ are provided by the choices $g_{1,n} = max$, $g_{2,n} = min$ on numerical parameters and $g_{1,n} = \cup$, $g_{2,n} = \cap$ on set parameters. For example, for invariance under distributivity, we have $max(x, min(y, z)) = min(max(x, y), max(x, z))$, for integers x, y, z.

Of course, this recursive definition of measures m_1 and m_2 is incomplete without specification of a base case which defines $m_i(d)$ where d is a unitary diagram. Deriving $m_i(d)$ from the contours of the unitary diagram, for example, provides a pair of measures m_1 and m_2 which are invariant under all the logic rules, but which are variant under the reasoning rules Add Contour and Delete Contour.

In the following section, for each pair of measures, we will assume that the above recursive definition holds unless stated, and give only information about base cases. We also state how to combine the measures m_1 and m_2 to get a contribution to the heuristic function between D_1 and D_2.

B. Measure and metric definitions

1) Contours: Here we define two measures which are invariant under all logic rules but variant under the Add Contour and Delete Contour rules. These measures will be used to detect differences in the contour sets. The definition follows recursively as in section IV-A, with n-ary functions $g_{1,n} = \cup$ and $g_{2,n} = \cap$. The base cases are provided by $m_1(d) = m_2(d) = \{$the labels of the contours of $d\}$, where d is unitary. Note that $m_1(D) = m_2(D)$ holds for unitary D, but need not hold for compound D. That is, the measures m_1 and m_2 are not equal. For example, in Fig. 6, $m_1(d_1) = m_2(d_1) = \{A\}$, $m_1(\neg(d_2 \sqcap d_3)) = \emptyset$ and $m_2(\neg(d_2 \sqcap d_3)) = \{A, B, C\}$. If there is a contour label in $m_i(D_2)$ but not in $m_i(D_1)$ then

Fig. 6.   Illustrating the measures.

we need to apply a reasoning rule to add that contour to D_1. Moreover, for each $C \in m_i(D_2) - m_i(D_1)$ we need to apply a reasoning rule to add C when we transform D_1 into D_2. That is, we need to add at least $|m_i(D_2) - m_i(D_1)|$ contours to D_1. Define for diagrams D_1, D_2, $i \in \{1, 2\}$:

$$AddC_i(D_1, D_2) = |m_i(D_2) - m_i(D_1)|$$

$$RemC_i(D_1, D_2) = |m_i(D_1) - m_i(D_2)|.$$

Combine these to give

$$CDiff_i(D_1, D_2) = AddC_i(D_1, D_2) + RemC_i(D_1, D_2).$$

Finally, the **contour difference metric** between diagrams D_1 and D_2 is defined to be

$$CM(D_1, D_2) = max\{CDiff_1(D_1, D_2), CDiff_2(D_1, D_2)\}$$

For the diagrams in Fig. 6, $CM(d_1 \sqcap d_3, d_4) = max\{2, 2\} = 2$ and $CM(\neg d_1, d_1 \sqcap d_3 \sqcap d_4) = max\{2, 1\} = 2$. We take the maximum because, for example, one application of the Add Contour rule can contribute to both $CDiff_1$ and $CDiff_2$ (we cannot take the sum $CDiff_1 + CDiff_2$). For example, if we introduce B to diagram d_4 in Fig. 6, yielding diagram d'_4 in Fig. 7 then $CM(d_4, d'_4) = max\{1, 1\} = 1$. The sum $1 + 1$ would not provide a lower bound on the length of a shortest proof from d_4 to d'_4.

2) Zones: We will define metrics that detect differences in the zone sets, using $g_{1,n} = \cup$ and $g_{2,n} = \cap$. The base cases are provided by $m_1(d) = m_2(d) = \{$the zones of $d\}$ where d is unitary. Note, again, that $m_1(D) = m_2(D)$ holds for unitary D, but need not hold for compound D. This will also be the case for the remaining measures we define with one exception. Before calculating the zone metrics for the heuristic function, we need to ensure that the unitary components of the premise and conclusion diagrams have the same contour sets. It has been argued in [3] why this is needed for unitary diagrams, and the same reasoning applies for compound diagrams. We apply the Add Contour rule to all unitary components of D_1 to make a new diagram, $CForm(D_1, D_2)$, in which each unitary diagram includes all contour labels from D_1 and D_2 (this being the union of the sets of contour labels of their unitary components). Similarly, we make a new diagram $CForm(D_2, D_1)$ by applying the Add Contour rule to D_2. For example, in Fig. 6, $CForm(d_1 \sqcap \neg(d_2 \sqcup d_3), d_4)$ is $d_1' \sqcap \neg(d_2' \sqcup d_3')$, shown in Fig. 7, obtained by adding contours to each unitary component d_1, d_2 and d_3. Similarly, $CForm(d_4, d_1 \sqcap \neg(d_2 \sqcup d_3))$ is d_4'. Define for diagrams D_1, D_2, $i \in \{1, 2\}$:

$$AddZ_i(D_1, D_2) = \begin{cases} 1 & \text{if } m_i(CForm(D_2, D_1)) \nsubseteq \\ & m_i(CForm(D_1, D_2)) \\ 0 & \text{otherwise} \end{cases}$$

$$RemZ_i(D_1, D_2) = \begin{cases} 1 & \text{if } m_i(CForm(D_1, D_2)) \nsubseteq \\ & m_i(CForm(D_2, D_1)) \\ 0 & \text{otherwise.} \end{cases}$$

The capping of $AddZ_i$, and $RemZ_i$ is similar to the capping

Fig. 7.   Contour forms.

applied to $AddZone$ and $RemZone$ in the case of unitary diagrams [3]. This is due to the fact that a single application of either Add Shaded Zone or Delete Shaded Zone to D_1 can change the zone set in $CForm(D_1, D_2)$ by more than one zone. For example, in Fig. 8, we can add one zone to d_1, giving d_3 but $CForm(d_1, d_2)$ has two fewer zones than $CForm(d_3, d_2)$. We define two metrics (that we will use to define the zone difference metric) between diagrams D_1 and D_2 to be

$$AddZ(D_1, D_2) = max\{AddZ_1(D_1, D_2), AddZ_2(D_1, D_2)\}$$

$$RemZ(D_1, D_2) = max\{RemZ_1(D_1, D_2), RemZ_2(D_1, D_2)\}.$$

The reason for taking the maximum (as opposed to the sum) is that, for example, applying the rule Delete Shaded Zone can affect both $AddZ_1$ and $AddZ_2$ simultaneously (and similarly, $RemZ_1$ and $RemZ_2$). We define the **zone difference metric**

Fig. 8.   Illustrating the zone measure capping.

$$ZM(D_1, D_2) = AddZ(D_1, D_2) + RemZ(D_1, D_2).$$
As an example, in Fig. 6, $ZM(d_1 \sqcap \neg(d_2 \sqcup d_3), d_4) = 1+1 = 2$.

3) Shading: We will now define metrics that detect differences in the shading, using $g_{1,n} = \cup$ and $g_{2,n} = \cap$. The base cases are provided by $m_1(d) = m_2(d) = \{$the shaded zones of $d\}$ where d is unitary. Before calculating the shaded zone difference metric for the heuristic function, we need to ensure that the unitary components of the premise and conclusion diagrams have the same zone sets. It has been argued in [3] why this is needed for unitary diagrams, and the same reasoning applies for compound diagrams. We take the unitary components of diagram $CForm(D_i, D_j)$ and add shaded zones until they are in Venn form (every possible zone is present, given the contour label set), giving $Venn(CForm(D_i, D_j))$. Shown in Fig. 9 are the unitary components of $Venn(CForm(d_1 \sqcap \neg(d_2 \sqcup d_3), d_4)) = d_1'' \sqcap \neg(d_2'' \sqcup d_3)$ and $Venn(CForm(d_4, d_1 \sqcap \neg(d_2 \sqcup d_3))) = d_4''$, where d_1, d_2, d_3 and d_4 are in Fig. 6. Define for diagrams D_1, D_2,

Fig. 9.   Venn forms.

$i \in \{1, 2\}$:

$$AddSh_i(D_1, D_2) = \begin{cases} 1 & \text{if } m_i(Venn(CForm(D_2, D_1))) \\ & \nsubseteq m_i(Venn(CForm(D_1, D_2))) \\ 0 & \text{otherwise} \end{cases}$$

$$RemSh_i(D_1, D_2) = \begin{cases} 1 & \text{if } m_i(Venn(CForm(D_1, D_2))) \\ & \nsubseteq m_i(Venn(CForm(D_2, D_1))) \\ 0 & \text{otherwise.} \end{cases}$$

We define two metrics (that we will use to define the shading difference metric) between diagrams D_1 and D_2 to be:

$$AddSh(D_1, D_2) = max\{AddSh_1(D_1, D_2), AddSh_2(D_1, D_2)\}$$

$$RemSh(D_1, D_2) = \\ max\{RemSh_1(D_1, D_2), RemSh_2(D_1, D_2)\}.$$

We combine these to give the **shading difference metric**

$$ShM(D_1, D_2) = AddSh(D_1, D_2) + RemSh(D_1, D_2).$$

For example, $ShM(d_1 \sqcap \neg(d_2 \sqcup d_3), d_4) = 1 + 1 = 2$, where d_1, d_2, d_3 and d_4 are in Fig. 6.

4) Spiders: We now define metrics which detect differences in the spiders, using $g_{1,n} = \cup$ and $g_{2,n} = \cap$. The base cases are provided by $m_1(d) = m_2(d) = Sp(d)$ where

$$Sp(d) = \{(i,r) : 1 \leq i \leq n \text{ where } n \text{ is the number of}$$
$$\text{spiders whose habitat is the region } r \text{ in } d\},$$

and d is unitary. For example, in Fig. 6,

$$m_1(d_1) = \{(1, \{A\}), (2, \{A, U - A\}), (1, \{A, U - A\})\}.$$

Informally, then, a spider is a pair, (i,r) and i indicates that (i,r) is the ith spider inhabiting r. The set $m_1(d_1)$ includes one such pair for each spider in d_1. So, if there are no spiders inhabiting r in d_1 then this is represented by the absence of any element in $m_1(d_1)$. Define for diagrams $D_1, D_2, i \in \{1,2\}$:

$$AddSp_i(D_1, D_2) = |m_i(CForm(D_2, D_1)) -$$
$$m_i(CForm(D_1, D_2))|$$

$$RemSp_i(D_1, D_2) = |m_i(CForm(D_1, D_2)) -$$
$$m_i(CForm(D_2, D_1))|.$$

We define two metrics (that we will use to define the spider difference metric) between diagrams D_1 and D_2 to be:

$$AddSp(D_1, D_2) = max\{AddSp_1(D_1, D_2), AddSp_2(D_1, D_2)\}$$

$$RemSp(D_1, D_2) =$$
$$max\{RemSp_1(D_1, D_2), RemSp_2(D_1, D_2)\}.$$

Note here that, for example, a single application of the Excluded Middle rule can impact both $AddSp$ and $RemSp$. Moreover, the rule Split Spider can introduce two new spiders. Thus we define the **spider difference metric**

$$SpM(D_1, D_2) = max\{AddSp(D_1, D_2), RemSp(D_1, D_2)\}/2.$$

For example, $SpM(d_1 \sqcap \neg(d_2 \sqcup d_3), d_4) = max(0, 4)/2 = 2$, where d_1, d_2, d_3 and d_4 are in Fig. 6.

5) Connectives: In this section we define metrics which detect differences in the connectives, using $g_{1,n} = max$ and $g_{2,n} = min$. The base cases are provided by, for unitary d,

$$m_1(d) = m_2(d) = 0.$$

For these measures we over-ride part of the generic definition of the measures. As usual, use

$$m_i(D_1 \sqcap ... \sqcap D_n) = g_i(m_i(D_1), ..., m_i(D_n))$$

$$m_i(D_1 \sqcup ... \sqcup D_n) = g_j(m_i(D_1), ..., m_i(D_n))$$

(where $j \neq i$) but, where possible, over-ride this definition with, for $i = 1, 2$

$$m_i(d_1 \sqcap D_2 \sqcap ... \sqcap D_n) = 1 + m_i(D_2 ... \sqcap D_n)$$

where d_1 is a unitary diagram. One effect of this is to contrast $m_1(d_1) = 0$ with $m_1(d_1 \sqcap d_1) = 1$. Application of the idempotency rules can increase or decrease these measures, almost doubling or halving their value. For this reason, we use log_2 to create measures which count potential rule applications. Other rules, such as Excluded Middle, can increase the measures from 0 to 1.

The two metrics (that we will use to define the connective difference measure) between diagrams D_1 and D_2 are defined to be, for $i = 1, 2$, in the case when $m_i(D_1), m_i(D_2) > 0$

$$CnnM_i(D_1, D_2) = |log_2(m_i(D_1)) - log_2(m_i(D_2))|,$$

and in the case when $m_i(D_1) = 0$ and $m_i(D_2) > 0$

$$CnnM_i(D_1, D_2) = 1 + log_2(m_i(D_2)),$$

and in the case when $m_i(D_1) > 0$ and $m_i(D_2) = 0$

$$CnnM_i(D_1, D_2) = 1 + log_2(m_i(D_1)),$$

otherwise we define

$$CnnM_i(D_1, D_2) = 0.$$

A single application of an idempotency rule can affect both $CnnM_1$ and $CnnM_2$ simultaneously, so to prevent multiple-counting of these rule applications, the contribution to the heuristic function is the maximum of $CnnM_1$ and $CnnM_2$. We define the **connective difference metric** to be

$$CnnM(D_1, D_2) = max\{CnnM_1(D_1, D_2), CnnM_2(D_1, D_2)\}.$$

For example, $CnnM(d_1 \sqcap \neg(d_2 \sqcup d_3), d_4) = max(2, 0) = 2$.

6) Not metric: In this section we define measures which detect differences in the numbers of 'nots', using $g_{1,n} = max$ and $g_{2,n} = min$. The base cases are provided by, for unitary d, $m_1(d) = 0$ and $m_2(d) = 1$. For these measures we over-ride part of the generic definition of the measures. Instead of $m_i(\neg D) = m_j(D)$ we define, for non-unitary diagrams D

$$m_i(\neg D) = 1 + m_j(D).$$

One effect of this is to contrast $m_2(\neg\neg d) = 2$ with $m_2(d) = 0$. Application of the involution rule can increase or decrease these measures by 2. For this reason, we use half their value before we evaluate their contribution to the heuristic function.

Define the **not difference metric** between diagrams D_1 and D_2 to be

$$NM(D_1, D_2) = \frac{max\left\{ \begin{array}{c} |m_1(D_1) - m_1(D_2)|, \\ |m_2(D_1) - m_2(D_2)| \end{array} \right\}}{2}$$

For example, $NM(d_1 \sqcap \neg(d_2 \sqcup d_3), d_4) = 1$. The reason for taking the maximum is that, for example, the excluded middle rule can impact m_1 and m_2 simultaneously. The reason for dividing by two is that a single application of the involution rule can increase m_1 and m_2 by two (and its reverse subtract two).

C. Compound heuristic

Define the **compound diagram heuristic**, H, between D_1 and D_2 to be the sum

$$H(D_1, D_2) = CM(D_1, D_2) + ZM(D_1, D_2) + NM(D_1, D_2)$$
$$+ max\{ShM(D_1, D_2), SpM(D_1, D_2), CnnM(D_1, D_2)\}.$$

Note that we take the maximum of the shading metric, spider metric and connective metric because, for example, a single application of one of the rules Excluded Middle and Split Spider can affect all these measures simultaneously.

We generated a random sample of 500,000 pairs of diagrams for which the heuristic function was optimistic. We conjecture that the heuristic function is admissible.

V. IMPLEMENTATION AND EVALUATION

We have implemented this heuristic search as part of a spider diagram reasoning tool. The search can either stop when a proof is found, or seek the set of all optimal proofs. The application keeps a record of the number of proof attempts stored during the search. An initial comparison of the effectiveness of the heuristic was conducted by building random proofs (within small but arbitrary limits on complexity) and searching for the proofs using a breadth first search (zero heuristic) as compared to the heuristic outlined in this paper. The benefits gained are assessed by considering the data set of ratios (number of proof attempts with our heuristic)/(number of proof attempts with breath first search).

The number of proof attempts with the zero heuristic ranged from 34 to 443,000, and with our heuristic, ranged from 15 to 270,000. We collected data for 178 random proofs. The ratio of numbers of proof attempts ranged from 1 (where the zero heuristic searches the same space as our heuristic) to 0.004 (where our heuristic vastly reduces the search space size). The median ratio was 0.184, an 81.6% reduction in the size of the search space. More spectacular results were obtained for longer proofs. Further work is needed to establish why, in some cases, our heuristic gives no saving in the size of the search space.

VI. CONCLUSION

In this paper, we have demonstrated how a heuristic A^* approach can be used to automatically generate shortest proofs in a spider diagram reasoning system. We regard this as an important step towards generating readable proofs. Our work can be extended in a number of ways. The cost element of the evaluation function can be altered to incorporate factors that impact readability. For example:

- Comprehension of rules. There may be a difference in how difficult each rule is to understand. We can model a difference in the relative difficulty of rules by assigning different costs. Currently, we are conducting an experiment to determine the relative understandability of the rules.
- Drawability of diagrams. As discussed in [6] not all diagrams are drawable, subject to some well-formed

conditions. We can increase the cost of a rule application if the resulting diagram is not drawable.

Another extension of this work is to include further reasoning rules. The rule set in this paper forms part of a sound and complete set. However, enlarging the collection of reasoning rules available to the heuristic proof writer may affect the admissability of the heuristic function. Moreover, using additional rules enlarges the search space. Even if the heuristic function is admissible with the addition of a further reasoning rule, it may be the case that the heuristic function becomes less effective because the search space is larger. However, the benefit of adding further rules is that there will be more cases where proofs can be found: if $D_1 \vDash D_2$ and all proofs from D_1 to D_2 require a rule that we have excluded then, currently, no proof will be found.

In addition to its use for automatic theorem proving, our heuristic function can also be used to support interactive proof writing. It can advise the user on the probable implications of applying a rule (for example "Adding contour B will decrease the contour difference measure, so might be a good idea"). Possible applications of rules could be annotated with their impact on the heuristic value. The user could collaborate with the tool to solve complex problems.

REFERENCES

[1] R. Dechter, and J. Pearl. Generalized best-first search strategies and the optimality of A^*. *Journal of the Association for Computing Machinery, 32*, 3, pages 505-536. 1985.

[2] J. Flower, and G. Stapleton. Automated Theorem Proving with Spider Diagrams. In proceedings of Computing: The Australasian Theory Symposium (CATS 04), pages 116-132, ENTCS,Science Direct, 2004.

[3] J. Flower, and J. Masthoff, and G. Stapleton. Generating Readable Proofs: A Heuristic Approach to Theorem Proving with Spider Diagrams. In proceedings of Diagrams 2004, pages 166-181, Springer-Verlag, 2004.

[4] J. Gil, and J. Howse, and S. Kent. Formalising spider diagrams. In *Proceedings of IEEE Symposium on Visual Languages (VL99)*, pages 130–137. IEEE Computer Society Press, 1999.

[5] P.E. Hart, N.J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on System Science and Cybernetics*, 4, 2, pages 100-107, 1968.

[6] J. Flower, and J. Howse. Generating Euler Diagrams. Proceedings of Diagrams 2002, pages 61–75, Springer-Verlag, 2002

[7] J. Howse, F. Molina, and J. Taylor. SD2: A sound and complete diagrammatic reasoning system. In *Proc. IEEE Symposium on Visual Languages (VL2000)*, IEEE Computer Society Press, 2000.

[8] J. Howse, G. Stapleton, and J. Taylor. Reasoning with Spider diagrams. Avaiable from www.cmis.brighton.ac.uk/research/vmg, 2004.

[9] G.F. Luger. Artificial intelligence: Structures and strategies for complex problem solving. Fourth Edition. Addison Wesley: 2002.

[10] F. Molina. *Reasoning with extended Venn-Peirce diagrammatic systems*. PhD thesis, University of Brighton, 2001.

[11] S.-J. Shin. *The Logical Status of Diagrams*. Camb. Uni. Press, 1994.

Two-way exchange of knowledge through visual annotation

Daniela Fogli[1], Giuseppe Fresta[2], Andrea Marcante[3], Piero Mussio[3]

[1]*Dipartimento di Elettronica per l'Automazione, Università di Brescia, Brescia, Italy*
[2]*Istituto di Scienza e Tecnologia dell'Informazione (ISTI), CNR, Pisa, Italy*
[3]*Dipartimento di Scienze dell'Informazione, Università di Milano, Italy*
fogli@ing.unibs.it, giuseppe.fresta@isti.cnr.it, {marcante, mussio}@dico.unimi.it

Abstract

This paper discusses electronic annotation and its importance as a tool for two-way exchange of ideas among humans pursuing a common goal. It introduces a model of electronic document (e-document) and of electronic annotation (e-annotation) and the definition of a set of tools, which support an annotation strategy permitting two-way exchange of ideas between the document authors and users. It also illustrates an application of the model and of the definition highlighting the importance of two-way communication. The tools supporting electronic annotation are specified using a formal definition of characteristic pattern, virtual entity, and virtual entity behavior, and are implemented as a set of XML-based tools customized to the user culture, tailorable to the task and adaptable to the current situation.

1. Introduction

Document production and annotation are important tools in the traditional processes of knowledge communication and accumulation. They result from a long evolution: before the computer age, communities of experts developed in time documentation styles, notations and annotation procedures with the aim of recording the community's knowledge on a permanent physical media. This enabled the community's knowledge to be available to members when and where they require it and in the shape required to perform their current activities. Electronic documents (e-documents) evolve this scenario because they appear as new media, and complement the traditional documents in recording, annotating and making available community knowledge. The process of interaction between members of the community and e-documents appears as a new and complex field often constrained by present day design and implementation technologies. Most current web systems permit a one way exchange of ideas from authors to readers [1]. A one way exchange system limits the communication abilities and the sharing of common knowledge. For example, as Ka-Ping Yee, designer of the CritLink Mediator, observes: "an inaccuracy in a document that could have been corrected once by a reader must be noticed and re-corrected by each new reader..." [2], in the best situation in which each new reader is acquainted of the

inaccuracy! From our experience in many technical and scientific fields, as well as from the quoted literature [1][2][3][4], it emerged that two-way annotation is widely used as a tool which permits adequate knowledge communication and accumulation in problem solving. This paper introduces a model of electronic document (e-document) and of electronic annotation (e-annotation) and the definition of a set of tools, which a) give the user the same note-taking possibilities as paper documents; b) offer new possibilities due to their electronic nature, such as the capability of examining the history of a note and of its revisions; c) support an annotation strategy permitting two-way exchange of ideas between the document authors and users. It also illustrates an application of the model and of the definition highlighting the importance of two-way communication, discussing a scenario drawn from a first analysis of a photo-interpreter and a glaciologist collaborating in achieving the classification of a glacier. The tools take advantage of the new appealing capabilities offered by multimedia, hypermedia and multimodal systems to develop documentation and annotation techniques.

Two niceties of the approach are:

a) the model of annotation proposed is general enough to describe other existing annotation frameworks. The developed prototypes result compatible with the annotation frameworks which define annotation formats and protocols ruling its communication, such as Annotea [3][4];

b) in WIMP systems, an e-document, developed following the model, is presented to the user (materialized, in the following) as an image on the screen, whose pixels can be addressed singularly. This assumption has two consequences. First also texts are treated as images in the interaction process. Images, graphs, texts, and the mixes of them are treated in a uniform way. Second, it is possible to capture and check each action performed on each pixel on the screen. In this way both a mechanism for adequate control of the interaction and one for linking annotation to any element of the physical representation of the document can be realized.

The paper organization reflects our bottom up approach to system development: we first study what users need by analyzing how they currently perform their activities. Next we develop some evolutionary prototype with a participatory design approach [5]. In the next section is

therefore outlined our view on document annotation emerging from our experience and from the literature. In section 3 electronic document (e-document) and electronic annotation (e-annotation) are introduced. Section 4 frames our view of e-document and e-annotation in a recently proposed model of HCI. In Section 5 we analyze the process of e-annotation and the tools required to support it. In Section 6 it is shown an example of a two-way exchange system for knowledge accumulation and communication in the Earth science field. In the last section conclusions are derived.

2. Pre-electronic documents and annotation

In the pre-electronic world, a document is constituted by a physical support modified by some human activity, which humans use as a tool of study, communication, consultation or research. Humans interpret the document by applying their cognitive criteria and recognizing sets of elementary signs as functional or perceptual units, called *characteristic structures* (c*s*s) in [6]. Examples of cs*s* are letters in an alphabet, symbols or icons of technical languages, pictorial elements in a picture. Humans associate to each cs a meaning: the association of a cs with a meaning is called *characteristic pattern* (cp). Humans combine cs*s* into complex ones (letters into words, icons into plant maps etc.) and associate a meaning to the complex cs*s*, so defining complex cp*s*. The document itself is recursively recognized as a meaningful entity, a complex cs, and when interpreted, as a complex cp. An annotation is 'a note, added by way of comment or explanation' [7] to a document or to a part of a document. The entity being annotated is called the *base of the annotation*. The annotation is a complex cs interpreted by a human as a cp. In general, both pre-electronic documents and annotations are *multimedia* elements: they are formed by combining texts, images and graphs. Fig. 1 shows a subset of a document, a scientific paper enriched by annotations of different types [8]. Five annotations explain the structure of the Vedrette glacier image. Four of them are one word textual annotations; the last is a multimedia one, composed by a word (Watershed) and an histogram. In some case an annotation is idiosyncratic, that is not semantically explicit but represented by symbols, which cannot be understood by an occasional reader, but only by members of a community who agreed on its meaning -in the extreme case the community being a singleton formed by the only human who made the annotation for his own use-. This is the case of symbols '*' appearing in Fig.1, one annotating the beginning of example 2 and the second the end. The base of the annotation is often made evident by a *visual identifier*. Visual identifiers may also mark parts of a document for other uses. According to this definition, annotation is in general different from a visual identifier. An exception is the visual identifier which underlies the last three lines in Fig. 1, and acts as an idiosyncratic annotation (i.e. it signals the importance of a sentence). In Fig.1, five cs*s* are visually identified in the remote sensed image: they are surrounded

by visual identifiers in the form of continuous lines of different colors.

Fig. 1. An inscription resulting from a human scientific interpretation of a Landsat image of Vedrette Glacier

written in different languages, organized with different goals and according to different reading habits.

Example 2. The inscription of Fig. 1 is embedded in an e-document written in English for the glaciologist community in Fig. 3 and in Fig. 7, while it is embedded in an e-report for Italian clerks of an environmental protection agency in Fig. 4. The glaciologist and clerk communities share a common interest in glaciological data: the glaciologist interprets the image, and interpretations are used by clerks in their activities. The two communities can be seen as two sub-communities of the more general Earth Scientist and Technologist community, and their two notations as dialects of a more general notation of this larger community. ♦

Fig. 1. *"Within the document" multimedia annotation of a multimedia document.*

Visual identifiers are not always present, as it occurs in the case of the second idiosyncratic annotation of Fig.1. In some case, the human annotating a base makes explicit the link between annotation and base using *a visual link*. This is the case of the five annotations enriching the Landsat image in Fig.1, where each identified base -a cs in the remote sensed image- is visually linked to its corresponding annotation by an arrow. Visual links are not always created. The author of the idiosyncratic annotations has not made the links explicit, leaving the identification of the relation annotation-base to the reader. On the whole, a human creates an annotation using a set of graphical elements which s/he considers meaningful (letters, elements of an alphabet, icons). S/he uses visual identifiers and visual links whenever s/he considers necessary to make the link between annotation and base clearer. This stance is different from the one in [9], where for example visual linking is not explicitly considered. On the other side, we agree with these authors on classifying annotations as

'*within the document*', such as in-line, on the margin or on the top or bottom of the page, or '*stand alone*', such as written on a separate piece of paper. Communities of experts in different disciplines have developed and refined their documentation and annotation techniques to permanently record their concepts, thoughts, procedures and data on some physical support. Members of the community adapt their behaviors to better exploit the techniques: for example they develop in time specialized *reading patterns*, sequences of reading modes and events, which allow efficient reading of a document in relation to the current aims and tasks of the reader. Reading patterns characterize user reading activity and hence the user-document interaction while users navigate and manipulate documents. Reading patterns depend on user culture, skills and goals but also on document notation, activity performed and context [10]. The members of the community also develop strategies and guidelines to perform document annotation -how to identify the base, how to link annotation and base-. These strategies and guidelines evolve according to user culture and understanding of the rhetoric of the communication [11] as developed within the community. We start from the analysis here summarized to define electronic annotation and to implement tools to support e-annotation of e-documents.

3. Electronic document and annotation

In the electronic world, documents and annotations become '*electronic*', in that they are no more recorded on a permanent support, but exist '*virtually*' as the results of the interpretation of a program by a computer [5]. Users can perceive, access, manage and annotate e-documents and e-annotations because the computational process generates some physical representations perceivable by them, for example images on a screen. These physical representations only exist and are perceivable until the electronic machinery maintains them in existence. Because of this virtual existence, e-document and e-annotation are less persistent than paper-based one, but this dependence on a machine offers some advantages. The e-document may evolve to "a unit consisting of dynamic, flexible, non linear content, represented as a set of linked information items, stored in one or more physical media or networked sites; created and used by one or more individuals in the facilitation of some process or project" [12]. However, e-documents appear to users as single entities even when their content is distributed in different, geographically remote repositories. E-documents can be managed, annotated and adapted by their users, thus evolving during their usage and adapting to their users. Moreover, the physical representation of a document and of its annotations results from a mapping of the content of the document and annotations stored inside the machine into output events perceivable by users (e.g. the images on the screen). The process of creating the content of a document or of an annotation can be separated by the process of its physical representation, which may be multimodal and tailored to the single user. The materialization -i.e. the activity of translating an internal representation into a physical one-can be adapted to the culture, skills and abilities of the current user, without altering document or annotation content [13]. However, from the point of view of documentation, e-document and e-annotation maintain a different nature as in the pre-electronic world. E-annotation is a multimedia-multimodal comment which is added to a part of an e-document (its target document). E-annotations can only exist as associated to an e-document.

4. E-documents as virtual entities

According to the Pictorial Computing Laboratory model as described in [6], e-documents are virtual entities, which some humans use as a tool of study, communication, consultation or research. The specification of e-documents and e-annotations is based on the techniques adopted to specify virtual entities. For this reason we briefly introduce virtual entity definitions and specification techniques. A virtual entity (ve in the following) is a virtual dynamic open system. It is *virtual* in that it exists only as the results of the interpretation by a computer of a set of programs P; *dynamic* in that its behaviour evolves in time; *open* in that the evolution depends on its interaction with the environment. We are interested in *interactive* ve, i.e. ve in which a human user is part of the environment and interacts with the ve. A ve manifests its state to the users as a characteristic structure (cs) -in the current implementation a set of pixels on the screen-. The interaction begins when the user acts on some input device to manifest his/her requirements or commands to the ve. The ve captures input events generated by user actions and reacts to them generating output events toward users. Output events are characteristic structures materialized on the output devices of a computer to become perceptible by the users. The cs generated as a reaction to input events depends on the current state of the ve. The user perceives the cs generated in reaction to his previous action and decides what to do next performing a new action. P is a set of programs, some of which -called I (Input) programs- acquire the input events generated by the user actions, some -called AP (APplication) program- compute the ve reactions to these events, and some -called O (output)- output the results of this computation. At each step of this cycle, the ve state is defined as a *characteristic pattern* $cp=<cs, u, <int, mat>>$, where int (interpretation) is a function, mapping some subset of the current cs of the ve to elements in the current computational state u of the program AP and mat (materialization) a function mapping elements of u to subsets of pixels in cs.

Let us call *behaviour* of the ve a sequence of cycles <input action, computation of the reaction, output generation, user decision>. The set of admissible behaviours for a ve is defined by the quadruple $ve=<P, CS, AC, cp_0>$ where P is the program generating the ve, CS is the set of admissible perceptible output css, AC is the set of admissible input events which can be generated by user actions, and cp_0 is

the state of the ve at the initial time t_0. In the following the elements of AC are denoted by the name of the user action which generates them 'select icon x' denotes the set of one or more events generated by the user actions to select icon x -e.g. double click on it- and interpreted by the ve as a unique command. A simple example of ve is the "floppy disk" widget to save file, whose cs -a diskette icon- appears in the iconic toolbar of MS Word[TM]. This ve outputs different css to indicate different states of the computational process: for example, if at time t_0 the user selects (performs the action of selecting) the icon, the ve reacts by a three steps behavior: it a) changes its cs -i.e. highlighting the disk shape-; b) saves in a disk file the current version of the document; c) once the document is saved, the disk shape goes back to its usual materialization (not highlighted). The program P can be specified following several techniques. In our current implementation [14] the program P is specified as a set of fragments, codified in several W3C compliant languages and can be interpreted by every SVG-compliant browser. SVG is the web standard for vector graphics [15]. The implementation of our model of the ve required the definition of IM^2L (Interaction Multimodal Markup Language), an XML-based mark-up language defined to describe ve logical and layout structure. On the whole the languages involved are: SVG to manage the interactive visual part, IM^2L, DTD for structure, ECMAscript for specifying the ve dynamics. The program P, is implemented as a quadruple P=<IM^2L-doc, IM^2L-dtd, pro-lib, func-lib>, where IM^2L-doc is an IM^2L document describing the logical and layout structure of the ve at hand, IM^2L-dtd is the DTD of IM^2L, pro-lib is a library of SVG prototypes, func-lib is a library of ECMAscript functions determining the interaction and dynamics of the ve. In a IM^2L-doc, a set of ve-body elements contains those information that the designers want to appear on the screen to support the user in understanding the current state of the ve. A ve-body element is formed by data and metadata elements. Examples of data are the name of a button or the results of some observation to be displayed. The metadata element describes properties of the data -such as author, date, procedures followed to create the data-. Metadata may be materialized to be seen by the user, or used for ve management, such as storage, indexing, retrieving, sending to a different user. The different nature of e-document and e-annotation is reflected into the properties of their body. The body of an e-annotation (a-body element) can only exist depending on the existence of a body in a target e-document (d-body). In other words, given the IM^2L program P of an e-document, its IM^2L fragment contains one or more d-body elements. One or more a-bodys can be associated to a d-body; a-bodys can only exist associated to a d-body. In the following, we consider three kinds of e-documents:

1) the interactive environment, which is a virtual entity whose css are whole images on the screen;
2) the target e-document, that is a part of the interactive environment, which the user considers as the ve to study and to work on. It can be a text, an image, a graph, or a mix of them, equipped with a set of tools;
3) the annotation manager, which is an e-document that does not exist in isolation, but is always associated to a target e-document, and permits to create, materialize, access and manage the content of an annotation.

5. E-annotation and annotation kit

Users create e-annotations with reference to a cs of a target e-document. A user identifies the *base* of the annotation - i.e. a subset of pixels (some words in a text, a structure in an image) s/he wants to comment within the cs of the target e-document represented on the screen. The user can make evident the base by creating a *visual identifier* and/or make explicit the existence of the annotation by a *visual link*, as it was usual in the pre-electronic world. In any case, the user creates an e-annotation, which can be later retrieved and updated by the same or other users. A set of tools -the *annotation toolkit*- permits the creation and management of e-annotation, visual link and visual identifier. These tools are the *annotation manager, the visual link creator* and *the visual identifier creator*.

An *e-annotation* is a pair E-A=<a-body, a-cs> where the a-body is an element of the IM^2L d-body of the target document, while a-cs is a cs created by the annotation manager. The *visual identifier* (vi) is a button, i.e. a ve which activates a computational activity when selected. Its cs is a marker identifying the base of the annotation -e.g. a closed line surrounding it or a transparent segment superimposed to it-. The vi may be a stationary, active or proactive button. When selected, a stationary vi does not react, while an active vi reacts. A proactive vi is also capable of autonomous activity, such as pre-fetching data related to the content of the annotation being build. The *visual link* is a button. Its cs is an icon, materialized near or pointing to the base. The selection of its cs determines the activation of the annotation manager. The *annotation manager* is a complex ve, i.e. a ve composed by other ves. It is composed by a set of operators, a set of user or system writable text fields, and a set of labels. Operators appear on the screen as icons, such as buttons and menus. As said, the annotation manager permits to create, materialize, access and manage the annotation content. In analogy to what happens in traditional paper-based annotation, the annotation manager can be a part of the target e-document or exist as a "symbiotic" e-document, i.e. an e-document that only lives if an associated target e-document exists. The choice will depends on the usual practice adopted by the users. Its cs and its behaviour are defined according to the context of use: in any case it must display the annotation body, represented according to the user requirements and current needs. The *visual link creator* is a button. It is associated to the target e-document: the program Pvl generating it is a subprogram of the program Pt generating the target e-document and its cs appears on the screen as part of the target e-document cs. When selected it becomes

active and requires the users to select a point in the e-document CS, and creates in that point a visual link instance. The *visual identifier creator* is a button. It is associated to the target e-document in that the program Pvi generating it is a subprogram of the program Pt generating the target e-document and its CS appears on the screen as part of the target document CS. When selected it becomes active and requires the user to draw the visual identifier on the annotation base, and creates the corresponding visual identifier instance.

Document annotation tools exist embedded in current word processors and document readers, e.g. the MS Word and Adobe Acrobat Reader reviewing mechanism. These tools can be described by the proposed e-annotation model, highlighting their properties. For lack of space, we do not discuss this issue here, but we introduce the use of a stand-alone annotation toolkit designed for two-way exchange between technicians involved in an image interpretation task. The image to be interpreted belongs to the body of the target e-document and the structures of interest in it are the base of the annotation. The image interpretation arises as a final annotation resulting from the exchange of intermediate annotations between the two technicians. The annotation manager is a symbiotic e-document equipped with several tools for the creation and updating of the annotation body. It also proactively collects and displays metadata (e.g. the authors of the annotations, title of the document), to support user interpretation of the annotation. The annotation bodies are materialized as text, images, graphs, within the CS of an annotation manager. The whole history of the annotation is stored and managed through a menu, which, when opened, shows the sequence of update annotation identifiers and allows the selection of the desired one.

6. Two-way exchange of knowledge through e-annotation

A scenario is described to illustrate how knowledge can be accumulated and communicated by two-way exchange of stand-alone annotations. In the scenario, a photo-interpreter and a glaciologist, incrementally gain insight and reach the classification of a remote sensed image by a two-way exchange of annotations. The photo-interpreter and the glaciologist are supported by two prototypal interactive environments, B-Glaciologist and B-Photointerpreter, which share a knowledge repository. These environments constitute a two-way exchange system which supports the communication and accumulation of knowledge about glacier classification. The environments run under a web browser (Internet Explorer in this case), and therefore they may reside in (possibly) different places and can be used at (possibly) different time.

Fig. 2. *The photo-interpreter is seeing the annotation performed by the glaciologist and is adding her own annotation.*

In Fig. 2 a screenshot of B-Photointerpreter is presented, in which the Explorer tools CSs can be recognized at the top of the figure. Under them, a set of CSs form a header, which presents general information about the creators of the system. Three CSs lie under the header: an equipment area on the right with a title identifying B-Photointerpreter, a working area on the left, and a message area on the bottom. In the equipment area, four menus are present denoting repositories of entities to be worked (images and annotations) and equipments to work on entities. In the

working area, the cs s of two virtual entities, a target e-document and an annotation manager, can be perceived by the photo-interpreter. The target e-document includes a d-body and tools to work on it. The d-body is constituted by data, the raster image in this case, and metadata, such as the name of the geographical entity ("Gruppo Adamello") and the name of the photo-interpreter ("dr. Anna Rampini") analyzing the image of interest. Tools are represented in the target e-document as icons in a toolbar. The cs of the visual link creator (green button labelled 'a') and of the visual identifier creator (green button with the closed curve) appear among them. In Fig. 2, the photo-interpreter is studying an intermediate result obtained by a glaciologist who, by interacting with B-Glaciologist, recognized some cs s of interest in the glacier image, to be signaled to the photo-interpreter. The glaciologist recorded the results of this interpretation as an e-annotation. The existence of an e-annotation is signaled by two cs s, superimposed to the cs of the d-body (the raster image). They are the cs of a visual identifier -the opaque shield surrounded by a red line-, and the cs of a visual link -the pencil icon-. The visual identifier has been created by the glaciologist by interacting with the visual identifier creator (the tool for free hand drawing of closed curves) available among the tools of the target e-document. The visual link has been created by B-Glaciologist as a reaction to the user action of selecting the visual link creator and of clicking on a point to be the annotation anchor. The photo-interpreter reached the current state of the interaction in the following way: she has: 1) opened B-Photointerpreter; 2) accessed the target e-document previously annotated by the glaciologist; 2) recognized a visual identifier and a visual link in the target e-document; 3) selected the visual link. As a reaction, B-Photointerpreter presented the annotation manager with the glaciologist annotation. The photo-interpreter has seen the request by the glaciologist and has added her annotation in the part of the annotation manager devoted to new annotations. Figure 2 shows also the annotation manager whose cs s are organized as follows: it displays on the top a toolbar, under which the body of the annotation previously inserted by the glaciologist is materialized. This body includes data (the text "please examine and classify the marked area. Is it an accumulation area?") and metadata (the identifier of the annotation -"nota:1/1"-, the name of the prototype used by the author -"B-Glaciologist"-, and the title of the annotation -"classification request"-). These metadata, are metadata related to the target e-document, but are displayed as data in the body of the annotation manager. The bottom cs s of the annotation manager materialize the a-body that the photo-interpreter has just inserted to respond to the glaciologist's request. The a-body includes data (the text "I classified using XY method. The classification includes ablation and accumulation areas") and metadata (the title "Classification", and the name of a linked file "24-06-04"). Author and locality metadata are automatically managed and displayed by the annotation manager. At the end of this annotation activity, the photo-interpreter may store the annotation in the knowledge repository by selecting the 'Add Note' button. The glaciologist can successively use B-Glaciologist to study the updated annotation.

7. Conclusions

This paper discusses e-annotation and its importance as a tool for two-way exchange of ideas among humans pursuing a common goal. The discussion leads to a definition of e-annotation and of a set of tools required to support its creation and use. The discussed scenario shows how the traditional annotations (e. g. the classification of the structure) can be implemented as an e-document. The tools permit the two-way exchange of annotations, which supports a better understanding of the document at hand.

References

[1] Heck, R., M., Luebke, S. M., Obermark, C. H., A survey of Web Annotation Systems. *Digital Documents* 1999, Id. 31.

[2] Yee, K-P., CritLink: Better Hyperlinks for the WWW, [Online] 1998 <http://crit.org/~ping/ht98.html>.

[3] Annotea Project, http://www.w3.org/2001/Annotea/.

[4] Kahan, J., Koivunen, M-R., Prud'Hommeaux, E., Swick, R. R., Annotea: An Open RDF Infrastructure for Shared Web Annotations, *Proc. of the WWW10 International Conference*, Hong Kong, May 2001.

[5] Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T. Human-Computer Interaction, Addison-Wesley, Wokingham, UK, 1994.

[6] Bottoni, P., Costabile, M. F., Mussio, P., Specification and Dialog Control of Visual Interaction. *ACM TOPLAS* 21(6), 1999, 1077-1136.

[7] *Merrian-Webster Dictionary*, online <http://www.m-w.com>

[8] Carrara, P., Fogli, D., Fresta, G., Mussio, P., Toward overcoming culture, skill and situation hurdles in human-computer interaction, *Int. J. Universal Access in the Information Society*, 1(4), 2002, pp. 288-304.

[9] Ovsiannikov, I. A., Arbib, M. A., Mcneill, T. H., Annotation Technology, *Int. J. Human-Computer Studies*, 50, 1999, 329-362.

[10] Hornbæk, K., Frøkjær, E., Reading Patterns and Usability in Visualization of Electronic Documents, *ACM TOCHI*, Vol. 10, No. 2, June 2003.

[11] Bottoni, P., Levialdi, S., Rizzo, P., An Analysis and Case Study of Digital Annotation, *Proc. 3rd Int. Workshop DNIS 2003*, Aizu, Japan, *LNCS 2822*, 2003, pp. 216-231.

[12] Shamber, L. What is a document? Rethinking the concept in uneasy times. *Journal of the ASIS*, 47 (9), 1996, 669-671.

[13] Mussio, P. E-Documents as tools for the humanized management of community knowledge, *Keynote Address, ISD 2003*, Melbourne, Aug. 2003.

[14] Fogli, D., Fresta, G., Mussio, P., Salvi, D., Documents in Visual Reasoning: from paper-based to electronic ones, *Proc. DMS 03*, Miami, FL, September 2003, 277-282.

[15] W3C: Scalable Vector Graphics (SVG), [Online] 2001 <http://www.w3.org/Graphics/SVG/>

An XML Schema for Sharing Geometric Information Between Mobile Robots

Joseph J. Pfeiffer, Jr.
Department of Computer Science
New Mexico State University
Las Cruces, NM 88003 USA
`pfeiffer@cs.nmsu.edu`

Abstract

Cooperative action by ensembles of mobile robots requires a mechanism by which they can share the discoveries they have made regarding their environment. XMap is an XML schema providing a language by which uncertain information regarding the environment can be shared among such an ensemble.

XMap provides for a hierarchical representation of 2- and 3-dimensional objects, uncertainty regarding their presence, and explicit specification of a common origin and distance units.

This paper presents the use of XMap within the context of the Isaac visual programming environment, and describes the enhancement of the Isaac environment to include cooperative robots.

1 Introduction

A number of application domains exist in which it would be very helpful for ensembles of mobile robots to cooperate in accomplishing tasks, ranging from "play" such as robot soccer to military UAV (unmanned autonomous vehicle) tasks. In order to accomplish these tasks, it must be possible for the robots to communicate information regarding their locations and goals, and any information they have learned regarding the environment. In our application domain, all information is maintained as geometric information, so all of these tasks reduce to the sharing of geometric information.

Sharing geometric information, in turn, requires communicating information regarding the shape, location, and classification of objects, as well as measures of confidence regarding the objects which have been located.

In this paper, we describe an XML schema which has been developed to accomplish these goals. The schema describes objects using a three dimensional triangulated mesh, with the triangles marked with a degree of confidence in their presence. In addition, a simulation of the use of the schema has been developed, which permits robots to multicast geometric information to one another.

The remainder of this paper is organized as follows. Section 2 describes prior work which is relevant to this effort, including other robotic and geographic XML schemas and a brief description of the Isaac mobile robot environment. The description of XMap itself is in Section 3; Section 4 describes the integration of our inter-robot communication mechanism into Isaac. Finally, a summary of results and prospects for future research are given in Section 5.

2 Prior Work

This work is an extension of the author's prior work in rule-based mobile robots to a cooperative environment, and also draws on previous work in developing XML schemas for robotics and geographic information system. This section briefly reviews this background.

2.1 Isaac

Isaac is a rule-based visual language for uncertain geometric reasoning, specifically intended for controlling mobile robots[3, 6, 8, 5]. Some of its features include:

1. Visual representation of geometric concepts

2. Explicit representation of uncertain knowledge regarding the robot's surroundings

3. Evidential reasoning for model updating

4. Consistent handling of input, output, and inference rules[6]

A typical Isaac rule for modifying its world model is shown in Figure 1. In this rule, the robot is represented by the octagon that appears in both the left and right

hand sides of the rule. The robot's local "forward" direction is toward the top of the page. In the on-line version of the paper, the triangle at the top of the left hand side of the rule is blue, while the triangle at the bottom of the right hand side of the rule is green. This rule would be interpreted as, "*if there is a blue region ahead of me, leave a green region in my trail.*" Its use would be in a maze-solving or mapping algorithm: as the robot explores the maze, another rule would leave a blue trail behind the robot. If it were to encounter a blue area, it would "know" that it was the second time that region had been encountered, and it would leave green behind to mark that fact. It should be noted that while this example uses common color names (blue and green), internally the colors are simply referenced by a color number, and colors in a a hue-saturation-brightness space are assigned for the visualization. While a set of standard hues is available, the user is free to change these assignments to accommodate cultural variations or limitations in color perception.

Figure 1: Sample Isaac Rule

Other rules exist to perform input (translating sensor readings into objects) and output (rule firings based on objects in the robot's environment causing actuator responses).

Isaac uses Dempster-Shafer evidential reasoning to manage the uncertainty of its environment: the degree to which the system has a belief in the presence of an object, and a belief in its absence, are represented explicitly. The maximum sum of belief and disbelief is 1.0; if the sum is less than that, the remaining unallocated belief is a measure of the system's uncertainty regarding the presence of the object. Rules include a measure of object presence or absence, and new objects are entered into the environment using Dempster's Rule of Combination[7]. The system uses a three-dimensional color space to visualize objects in the environment, with hue being used to represent object type, saturation for belief, and brightness for disbelief.

The reader is directed to the references, in particular [5], for a more detailed description of Isaac, including its rule-firing mechanisms. The evidence visualization mechanism is described in more detail in [4].

2.2 XML and Robots

A number of other projects have investigated the use of XML schemas in various aspects of robot control and communication. One of the most developed such languages is RoboML (robot manipulation language)[2]. This language has aims somewhat at variance to those of XMap, as it focusses on the robotic actuators and sensors themselves, rather than on a geometric world model.

Though not targeted at robots, GML (Geography Markup Language) is emerging as a standard for interoperability of Geographic Information Systems. It is, however, both richer than XMap in providing features that are of limited use to the Isaac project, and poorer in not including an uncertainty model compatible with that used by Isaac[1].

3   XMap

This section describes the XMap schema in detail, including both the schema itself and an example of its use in defining an object.

3.1   Object Hierarchy

A simplified view of XMap's object hierarchy is shown in Figure 2.

Figure 2: XMap Object Hierarchy

In this hierarchy, a geometric object is composed of an arbitrary number of triangles. Each triangle is constructed from three edges, and an edge is defined by two vertices. This representation was chosen for XMap due to (1) the ease of manipulating the representation, (2) the fact that arbitrary polygons can be constructed of triangles, and (3) the fact that curved objects can be approximated with arbitrary precision using triangles.

Each vertex in the object may optionally be defined relative to a local origin, which is typically the location of the robot which has accumulated the information. Be-

fore sending the object to another robot, the vertices are transformed to a global coordinate space.

In the following subsections, the XMap schema is presented. Due to the limited space available, the parts of the schema will be presented as a series of examples. These examples combine to describe a simple two dimensional object represented using XMap. Figure 3 shows the object being defined. It is a rectangle constructed from two triangles; this figure shows the coordinates and the vertex, edge, and triangle indices. In the figure, the vertex indices and coordinates are in a roman font, edge indices are in *italic*, and triangle indices are in **bold**. The object's color and Dempster-Shafer belief values are not visualized in this figure in the interests of clarity in the printed Proceedings.

Figure 3: Object From XMap Example

3.2 Objects

An XMap object includes all of the vertices, edges, triangles, and subobjects of the object as sequences; the triangles and edges will use references to elements of these sequences in their definitions. Figure 4 shows an example of an object, as represented in XMap. The vertices, edges, and triangles making up the object will be shown later in Figures 5, 6, and 7.

```
<object>

    list of vertices (see Figure 7)
    list of edges (see Figure 6)
    list of triangles (see Figure 5)

</object>
```

Figure 4: Example of XMap Object

In a three dimensional model environment, an object can also explicitly include its own volume. The intent of this element is to reduce the need to continually recalculate an object's volume.

3.3 Triangles

The object in Figure 4 is made up of two triangles, shown in Figure 5. A triangle consists of a triple

```
<triangles>
    <numTriangles>2</numTriangles>
    <triangle>
      <index>1</index>
      <color>3</color>
      <indexTriple>1 3 4</indexTriple>
      <belief>0.9</belief>
      <disbelief>0.05</disbelief>
    </triangle>
    <triangle>
      <index>2</index>
      <color>5</color>
      <indexTriple>2 3 5</indexTriple>
      <belief>0.25</belief>
      <disbelief>0.45</disbelief>
    </triangle>
</triangles>
```

Figure 5: Triangle Example

of indices identifying edges. It may optionally also include Dempster-Shafer belief elements (`belief` and `disbelief`), a color, and the triangle's area. As the triangle uses references to its edges, it is possible for an edge to be explicitly shared between two triangles. Figure 5 shows the triangles making up the object in Figure 4.

Triangle **1** is assigned Color 3 with a **belief** of 0.9 and a **disbelief** of 0.05 (leaving an uncertainty of 0.05); it is made up of edges *1*, *3*, and *4*. Triangle **2** is assigned Color 5 with a **belief** of 0.25 and a **disbelief** of 0.45, leaving an uncertainty of 0.3; it is made up of edges *2*, *3*, and *5*. Edge 3 is shared between the two triangles.

3.4 Edges

An edge is defined by two vertices, and may also explicitly describe its own length. The five edges making up the object are shown in Figure 6. Edge *1* connects vertices 1 and 2; edge *2* connects vertices 1 and 3; edge *3* connects vertices 1 and 4; edge *4* connects vertices 2 and 4; and edge *5* connects vertices 3 and 4. Since references to vertices are used, a vertex may be shared by multiple edges (and indeed each of the vertices in the object is shared by two or three edges).

A number of additional elements are also available in edges, though they are not used at present. As with triangles, edges can be assigned colors and Dempster-Shafer

```
<edges>
    <numEdges>5</numEdges>
    <edge>
      <index>1</index>
      <indexPair>1 2</indexPair>
    </edge>
    <edge>
      <index>2</index>
      <indexPair>1 3</indexPair>
    </edge>
    <edge>
      <index>3</index>
      <indexPair>1 4</indexPair>
    </edge>
    <edge>
      <index>4</index>
      <indexPair>2 4</indexPair>
    </edge>
    <edge>
      <index>5</index>
      <indexPair>3 4</indexPair>
    </edge>
</edges>
```

Figure 6: Edge Example

```
<vertices>
  <numVertices>4</numVertices>
  <vertex>
    <index>1</index>
    <point3D>0 0 0</point3D>
  </vertex>
  <vertex>
    <index>2</index>
    <point3D>0 1 0</point3D>
  </vertex>
  <vertex>
    <index>3</index>
    <point3D>1 0 0</point3D>
  </vertex>
  <vertex>
    <index>4</index>
    <point3D>1 1 0</point3D>
  </vertex>
</vertices>
```

Figure 7: Vertex Example

belief values. This will prove useful if Isaac (and Isaac's inter-robot communication) is extended for operations on graphs in the future.

3.5 Vertices

Finally, a vertex is a point in 3-space. The vertices from the running example are shown in Figure 7 .

3.6 Additional Elements

XMap also defines a number of additional elements, though they are not used in the above example.

Origin Each object in an XMap document has an origin associated with it. The origin establishes a local coordinate space for the object, so it can be geometrically transformed as required without editing each of the elements within it. The origin establishes distance units for the object, and can be used to establish a global georeference for the object. An example of an XMap origin is shown in Figure 8.

Timestamp An object may also have a timestamp associated with it. These could be used in a cooperative environment in which robots not only communicate their own data, but also act as relays for data from

```
<origin>
    <index>1</index>
    <point3D>
        4027.027912
        08704.857070
        212.15
    </point3D>
    <distanceUnits>meter</distanceUnits>
    <datum>WGS84</datum>
    <directionOfLatitude>
        N
    </directionOfLatitude>
    <directionOfLongitude>
        E
    </directionOfLongitude>
</origin>
```

Figure 8: Origin Example

other robots. The timestamp can be used to identify either old or redundant data, reducing congestion.

Subobjects It is also possible for an object to include a hierarchy of subobjects; for instance, a person might be defined with a head, four limbs, and a torso as subobject. Each subobject's origin would then be defined relative to its parent object.

4 Inter-Robot Communication in Isaac

Isaac's execution engine proceeds in two phases: in the first phase, each rule is matched in the robot's local environment, and the degree of its activation is determined. In the second phase, Dempster's Rule of Combination is used to update the world model according to each of the active rules. For rules whose right hand sides correspond to actuators, commands are sent to actuators at the end of this phase.

For use with XMap, this has been enhanced. As each of the rule activations is calculated, its objects are translated into XMap documents and multicast to all of the robots in the cooperative ensemble. At the end of a robot's update phase, any objects which have been received from other robots are applied, as local rules would have been applied. The local robot is capable of weighting the incoming objects according to confidence in their senders.

For purposes of investigating communication of geometric information, Isaac simply uses multicast UDP datagrams. It is assumed the underlying network (most probably 802.11b wireless ethernet) is able to handle the details of delivering packets. A user-configurable multicast IP address and port number is used for each ensemble of robots. As the robot updates its world model, the updates are sent to all of the robots in the ensemble for fusion using the algorithms described earlier. The sockets have their `SO_REUSEADDR` option set, so multiple robot simulations on a single host are written identically to multiple simulations on several hosts. XMap documents are tagged with the hostnames and process IDs of the robots sending them, so a robot is able to filter and discard its own objects.

The use of datagrams maps very well to the update model described above. Datagrams provide only best-effort delivery and are not guaranteed to be delivered in order; however, when they are delivered they are guaranteed to be correct and not fragmented. Our model update operations are commutative, so out-of-order delivery is not an issue. Likewise, failure to receive a datagram is simply missing evidence in the construction of the model, much like an unreliable sensor that sometimes fails to return a value. As the datagrams (if delivered at all) are correct and complete, we don't need to concern ourselves with issues of data integrity.

We are explicitly not concerned with issues of routing of messages among ensembles of robots in ad hoc networks where not all of the robots are within range of one another. This is an interesting issue which is being studied by many research groups at present; presumably it will be possible to leverage this research in the future. Similarly, we are not concerned with security at this time.

An example of two robots exploring an arena is shown in Figure 9. Here, two robots have been assigned the task of exploring an arena. One was set at the bottom center of the arena; the other near the center. Both were started, and communicated their model updates to each other as they explored the environment. This figure shows the control screen for the first robot; it is visible near but somewhat below the center of the arena (the small black "Pac-Man" figure represents this robot). The second robot has mapped the other blue trail, but is not visible. On the screen, the maze is shown in black. Maze walls which have been found by the robots are red, and the trail following the robots is blue. The example also demonstrates the use of hue, saturation, and brightness: areas which have not been explored (and consequently have complete uncertainty) are unsaturated, and not bright: they appear as grey. Areas which have been probed with sonar with no return (and hence are believed to be empty) are unsaturated, but bright: they appear as white. Areas which are believed to contain obstacles are highly saturated red. Finally, the trails being laid down by the robots are also highly saturated, in blue.

Figure 9: Two Robots Exploring an Arena

The two simulations are entirely independent, except through their communications. In this case both

were on a single host, however, they could equally well have been on two separate hosts which could well have been two separate robot platforms. Similar experiments have been carried out with simulations on as many as four hosts using a combination of 100BaseT wired and 802.11B wireless networks.

5 Conclusions and Future Research

An XML schema has been defined to permit the communication of uncertain geometric information between mobile robots. This has been used to define a facility for communication between robots using the Isaac rule-based geometric reasoning language; the information has been successfully transferred between a number of robots being simulated between several hosts.

Far more interesting than the actual results of this work is the avenues it opens for future work. Some of the many possible topics which can be explored include:

- Cooperative rule-based robots.

- Integration of robots running the Isaac environment with other robotic platforms.

- Efficient communication of geometric information among ensembles of robots using wireless communications. This is particularly interesting since the information being communicated may itself be useful in establishing the routing algorithms.

The XMap schema can also be used in domains other than those currently contemplated by the Isaac project. While Isaac is a strictly two-dimensional environment, XMap can be used to define three dimensional objects, including both objects such as rough terrain and true objects in three dimensional space. Also, as edges can be assigned colors and belief values, the system can be used to define geometric graph structures such as region adjacency diagrams.

6 Acknowledgments

References

[1] Open GIS Consortium. Open gis web page. http://www.opengis.org/.

[2] Maxim Makatchev and S. K. Tso. Human-robot interface using agents communicating in an xml-based markup language. In *Proceedings of the 2000 IEEE International Workshop on Robot and Human Interactive Communications*, pages 270–275, Osaka, Japan, September 2000.

[3] Joseph J. Pfeiffer, Jr. A language for geometric reasoning in mobile robots. In *Proceedings of the IEEE Symposium on Visual Languages*, pages 164–171, Tokyo, Japan, September 1999.

[4] Joseph J. Pfeiffer, Jr. Using brightness and saturation to visualize belief and uncertainty. In Mary Hegarty, Bernd Meyer, and N. Hari Narayanan, editors, *Diagrammatic Representation and Inference, Second International Conference, Diagrams 2002*, pages 279–289, April 2002.

[5] Joseph J. Pfeiffer, Jr. A prototype inference engine for rule-based geometric reasoning. In Alan Blackwell, Kim Marriott, and Atsushi Shimojima, editors, *Diagrammatic Representation and Inference: Third International Conference, Diagrams 2004*, pages 216–226, Cambridge, UK, March 2004. Springer-Verlag.

[6] Joseph J. Pfeiffer, Jr., Rick L. Vinyard, Jr., and Bernardo Margolis. A common framework for input, processing, and output in a rule-based visual language. In *Proceedings of the IEEE Symposium on Visual Languages*, pages 217–224, Seattle, Washington, USA, September 2000.

[7] Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[8] Rick L. Vinyard, Jr., Joseph J. Pfeiffer, Jr., and Bernardo Margolis. Hardware abstraction in a visual programming environment. In *Proceedings of the IIIS World Multiconference on Systemics, Cybernetics and Informatics*, pages 94–98, Volume I. IIIS, July 2000. Orlando USA.

Content-based Image Retrieval Combining Shape Recognition, Color and Text

Donggang Yu and Wei Lai
School of Information Technology
Swinburne University of Technology,
PO Box 218, Hawthorn, VIC 3122, Australia

Jiro Tanaka
Institute of Information Sciences and Electronics
University of Tsukuba
Tsukuba, Ibaraki 305-8573, Japan

Abstract

More and more images are used in HTML documents and databases. Effective image information retrieval on the WWW and databases are required to locate relevant images. This paper presents a new method of shape recognition which can be used for image shape based retrieval. The color images are separated by color reduction. The series of structural features of image contour are extracted based on difference codes. The shapes of objects are described and recognized based on morphological structural patterns. The decision and classification of retrieved objects of websites or database are based on the description of shape structures, prior text and color information, which are associated each other.

1 Introduction

More and more images are used in HTML documents and databases. Effective image information retrieval on the WWW and databases are required to locate relevant images. Most of current approaches are text based, color based and texture based [1] [2]. There is few research on shape based. This paper presents a new method of shape recognition which can be used for image shape based retrieval. In Section 2, color separation of color images is introduced, linearization, description features of contours, and structural points are described. The decision and classification of retrieved object images of websites or database are based on the description of shape structures, prior text and

color information, which are associated each other. Finally, a conclusion is given. The diagram of processing is shown in Fig. 1 where the key text, colour and shape of retrieval objects are known as prior information.

Figure 1. The diagram of processing for information retrieval based on text, color, and shape recognition.

2 Color image separation, extraction of description features and shape recognition

In order to retrieve required information from websites and database, we have to use all possible information such as text, color and image shape to construct a special set in a database or websites. For example, if the set is "flower image", the following procedures should be done: (a) all flower images are collected; (b) shape of flower is recognized; (c) determine which flower the flower is based on its shape, color and text. Also, it is possible that the same shape is different object in different background which are described by text

contents and color. For example, the sample images in Fig. 2 could be recognized as Indo-Pacific star (starfish) in marine environment or a star in a painting etc.

(1)   (2)   (3)   (4)

Figure 2. Four sample color images of web searching based on the text of flower.

2.1 Color image separation based on color reduction

In most of cases, it is possible that the object images can be represented and understand with a limited number of colors [3] [4]. Especially, in many cases, the object images can be constructed by one color. For example, we can say the blue sky, white cloud, yellow lily flower, silver plan, red or gold sun and so on. That means the object image could consists of one color. However, the true type color images consist of more than 16 million different colors in a 24 bit full RGB color space. In oder to separate the object image which consists of one color, it is necessary to reduce the number of colors. One method of color reduction [3] [4] is used, which is multi-thresholding (adaptive color reduction) with SOFM neural network. Based on the above algorithm, the optimal number of thresholds of color reduction, c_n, is estimated. That is color set (represented by C_R) of color reduction:

$$C_R = \{c_r([r0, g0, b0]), ...c_r(r[c_n - 1], g[c_n - 1], b[c_n - 1])\} \quad (1)$$

where $c_r([ri, gi, bi])$ is i-th color after reducing the number of colors in an image. For example, the number of colors of images in Figs. 2(3-4) and Fig. 2(1) is reduced, and the processed results are shown in Figs. 3(1), 4(1) and 5(1) respectively. Furthermore, each layer image (object image having one color) can be got based on reduction colors. Let $I(c_r[i])$ is the layer image which is found based on the $c_r[i]$-th color, where the background color is supposed as white color in general cases. Suppose B(r,g,b) is background color, and B(r,g,b)=B(255,255,255). If the object image consists of white color based on prior information (text, color etc.), the background color is black color B(r,g,b)=B(0,0,0). Based on the above algorithm, the

(1)   (2)   (3)   (4)

Figure 3. The color reduction and layer images of the image in Fig. 2(3)

set of the layer images (I_{rc}) is

$$I_{rc} = \begin{cases} I_{rc0}[0] & I_{xy}[r, g, b] = c_r[r0, g0, b0] \\ \vdots & \\ I_{rci}[i] & I_{xy}[r, g, b] = c_r[ri, gi, bi] \\ \vdots & \\ I_{rci}[c_n - 1] & I_{xy}[r[c_n - 1], g[c_n - 1], b[c_n - 1]] \\ & = c_r[r[c_n - 1], g[c_n - 1], b[c_n - 1]] \end{cases}$$

$$(2)$$

For the images in Figs. 3(1), 4(1) and 5(1), their layer images are shown in Figs. 3(2-4), 4(2-5) and 5(2-5) respectively based on the above algorithm.

(1)   (2)   (3)   (4)   (5)

Figure 4. The color reduction and layer images of the image in Fig. 2(4).

(1)   (2)   (3)   (4)   (5)

Figure 5. The color reduction and layer images of the image in Fig. 2(1).

2.2 The linearization and description features

All layer images can be transformed into binary images based on two colors of object and background.

For example, if a yellow flower need to be retrieved or classified, the binary image is found from the yellow layer image in Fig. 3(3). Its binary image is shown in Fig. 6(1). Similarly, the binary image in Fig. 7(1) is found from red layer image in Fig. 5(5). line segment,

Figure 6. Binary image, contour and smooth following, linearization, and structural points of the image in Fig. 3(3).

curvature angle and bend angle between neighboring lines, and their convexity and concavity are useful to describe the shape of binary images. Many methods and algorithms [5], [6], [7], [8] are developed for the description of contours in the past. Let the starting point of an binary image be the upper-left corner. The chain code set of its contour k is represented as:

$$C_k = \{c_0, c_1...c_i, ...c_{n-1}, c_n\} \quad (3)$$

where C_k is the chain code set of contour k, and i is the index of the contour pixels. The contour following results of images in Figs. 6(1) and 7(1) are shown in Figs. 6(2) and 7(2) respectively. The difference code is defined as:

$$d_i = c_{i+1} - c_i. \quad (4)$$

In the smoothed contour, $|d_i|$ equals 0 or 1 [9] [10]. The smooth following results of two examples (Figs. 6(2) and 7(2)) are shown in Figs. 6(3) and 7(3) respectively.

2.3 The linearizing lines based on difference codes

Suppose that chain code set of a linearized line is

$$c_k^{ln} = \{c_k^{ln}[0], c_k^{ln}[1]...c_k^{ln}[i], ...c_k^{ln}[n_k^{ln} - 1]\}, \quad (5)$$

where k is represented as the contour k of an image, ln as the line ln of contour k and n_k^{ln} as the total number

Figure 7. Binary image, contour and smooth following, linearization, and structural points of the image in Fig. 5(5).

of pixels which are contained in the line ln. A linearized line has following property:
if

$$d_{ij} = c_k^{ln}[i] - c_k^{ln}[j] \quad (i = 0, ...[n_k^{ln}-1]), (j = 0, ..[n_k^{ln}-1]), \quad (6)$$

then

$$| \ d_{ij} \ | \leq 1 (mod 8) \quad (i = 0, ...[n_k^{ln}-1]), (j = 0, ...[n_k^{ln}-1]). \quad (7)$$

Therefore, a linearized line contains only two elements. They are called as the element code, and are represented by $cdir1$ and $cdir2$ respectively. A smoothed contour can be linearized based on the above algorithm, and its related chain code set c_k^{ln}, x and y coordinate sets of linearized lines can be found. The linearization results of two examples (Figs. 6(3) and 7(3)) are shown in Figs. 6(4) and 7(4) respectively, and the starting point of each line is represented by character "Y".

2.4 The set of curvature and bend angles of linearized lines

1. **The curvature angle of linearized lines**
   Let l_{se} is straight line between the starting and end points of each linearized line. The curvature angle is defined as the direction angle between the x coordinate axis and l_{se} of the linearized line, and the angle is formed with starting from the direction of the x coordinate axis to the direction

of linearized line, which is determined by the lin-
earized line's element codes, in anti-clock. It can

**Figure 8. Finding the curvature angle of lin-
earized lines.**

be found based on Fig. 8. Suppose $\angle curve$ is the
curvature angle, then $\angle curve$ can be found based
on four cases (see Fig. 8) which are determined by
two element codes of linearized line as following
(corresponding four quadrants):

**(1) $cdir1$ and $cdir2$ being chain code 0, 1 or
2 (the first quadrant)**

let $\angle se$ be the tangent angle of linearized line ln,
then

$$\angle se = tag^{-1}(\frac{\Delta y}{\Delta x}), \qquad (8)$$

where Δx and Δy are the absolute values of differ-
ences of x and y coordinate between starting and
end points of line, which are shown in Fig. 8.
If $cdir1$ and $cdir2$ of linearized line ln consist of
chain code 0, 1, or 2, then

$$\angle curve = (180°/\pi)\angle se \qquad (9)$$

**(2) $cdir1$ and $cdir2$ being chain code 4, 5 or
6 (the third quadrant)**

In this case, $\angle se$ is found based on Equation (8),
and

$$\angle curve = 180° + (180°/\pi)\angle se \qquad (10)$$

based on Fig. 8.

**(3) $cdir1$ and $cdir2$ being chain code 2, 3 or
4 (the second quadrant)**

In this case, $\angle se$ is

$$\angle se = tag^{-1}(\frac{\Delta x}{\Delta y}), \qquad (11)$$

and

$$\angle curve = 90° + (180°/\pi)\angle se \qquad (12)$$

based on Fig. 8.

**(4) $cdir1$ and $cdir2$ being chain code 6, 7 or
0 (the fourth quadrant)**

In this case, $\angle se$ is found based on Equation (11),
and

$$\angle curve = 270° + (180°/\pi)\angle se \qquad (13)$$

based on Fig. 8.

2. Bend angle and its property

The bend angle is defined as the angle between the
lines j and $(j+1)$. It can be calculated based on
Fig. 9.

**Figure 9. Finding the bend angle between two
neighboring linearized lines.**

(a) **The bend angle of linearized lines**

Let $\angle curve[j]$ and $\angle curve[j+1]$ be the cur-
vature angle of linearized lines j and $[j+1]$
respectively, and $\Delta[j, j+1]$ be their curvature
angle difference, then

$$\Delta[j, j+1] = \angle curve[j] - \angle curve[j+1], \quad (14)$$

where $\angle curve[j]$ is α, $\angle curve[j+1]$ is β, and
$\Delta[j, j+1]$ is $|\alpha - \beta|$ in Fig. 9 respectively. Let
$\angle angle[j]$ be the bend angle, then:

$$\angle angle[j] = 180° - |\Delta[j, j+1]| \qquad (15)$$

based on Fig. 9.

(b) **The property of bend angle**

The bend angle property, convex or concave,
can be determined based on the structure
patterns of the element codes between two
neighboring linearized lines. Let $l_{cdir1}[j]$ and
$l_{cdir2}[j]$ be the first and second element codes
of the line j, and $l_{cdir1}[j+1]$ and $l_{cdir2}[j+1]$
be the first and second element codes of
the line $(j+1)$ respectively, then there are
thirty two detection patterns of the bend an-
gle property (convex or concave). The detec-
tion patterns (1-4) and (7-10) are shown in
Fig. 9.
One detection rules can be described as fol-
lowing: If $l_{cdir1}[j]$ is chain code 0 and $l_{cdir2}[j]$
is chain code 1 (see Figs. 10(1) and 10(7)), or
$l_{cdir1}[j]$ is chain code 1 and $l_{cdir2}[j]$ is chain
code 0 (see Figs. 10(2) and 10(8)), then

- If $l_{cdir1}[j+1]$ is chain code 2 or $l_{cdir2}[j+1]$
  is chain code 2, then $\angle angle[j]$ is convex
  (see Figs. 10(1-2)).

- If $l_{cdir1}[j+1]$ is chain code 7 or $l_{cdir2}[j+1]$ is chain code 7, then $\angle angle[j]$ is concave (see Figs. 10(7-8)).

Similarly, other detection rules can be described based on other patterns.

Based on the above algorithms, the series set of the curvature angle, the series of bend angle (including their convexity or concavity) of two sample images, which are shown in Figs. 6(4) and 7(4), can be calculated based on the above algorithm can be found from starting line to end line. These series sets of description features can be used to describe and recognize the shape of contours.

Figure 10. Detecting patterns of bend angle convexity or concavity.

2.5 Structural points of smoothed contours

The structural points are used to represent convex or concave in the direction of chain codes between two neighboring lines along the contour. Assume that $line[ln]$ is the current line and that $line[ln-1]$ is the previous line.

Definition 1: The convex point in the direction of code 4 (represented with the character "∧")
If the element codes 3, 4 and 5 occur successively as a group of neighborhood linearized lines, then the point corresponding to code 4 is a convex point.
if $cdir1$ of $line[ln]$ is code 4, $cdir2$ is code 5 and the direction chain code of the last pixel of $line[ln-1]$ is code 3, then the first pixel of the current line $line[ln]$ is a convex point.

Definition 2: The concave point in the direction of code 4 (represented with the character "m")
If the element codes 5, 4 and 3 occur successively as a group of neighborhood linearized lines, then the point corresponding to code 4 is a concave point.
if $cdir1$ of $line[ln]$ is code 4, $cdir2$ is code 3 and the

direction chain code of the last pixel of $line[ln-1]$ is code 5, then the first pixel of the current line, $line[ln]$, is a concave point.

Similar to Definitions 1-2, other structural feature points can be defined and found. These points are convex points "v", "[", ")", "F", "o", "T", "s", and concave points "$", "]", "(", "f", "O", "t" and "S" which are shown in Fig. 11. These structural points describe the convex or concave change in different chain code directions along the contour, and they can therefore be used to represent the morphological structure of contour regions.

Figure 11. The pattern models of structural points.

2.6 Shape recognition based on structural points, curvature and bend angles

The sets of structural point, curvature and bend angles of contours can be found based on the above algorithms. They can be used to recognize the shape of contours. Based on the above method, all structural points of the contours in Figs. 6(4) and 7(4) can be found, and they are shown in Figs. 6(5) and 7(5).
For morphological structure of the outer contour of Fig. 6(5) there is a series of structural points:
"∧" → "F" → "[" (convex) → "]" → "f" (concave) → "F"→ "["→ "s" (convex) → "S" (concave) → "s"→ "v" (convex) → "$" (concave) → "v"→ "o"→ ")" (convex) → "("→ "O" (concave) → "o"→ ")"→ "T" (convex) → "t"→ "(" (concave) → ")"→ "T"→ "∧" (convex) → "m" (concave) →"∧" (convex).
For morphological structure of the outer contour of Fig. 7(5) there is a series of structural points:
"∧" → "F" → "[" (convex) → "f" → "m"→ "t" (con-

cave) → "T"→ "∧" →"F"→ "[" (convex) → "]"→ "f" (concave) → "F"→ "["→ "s"→ "v" (convex) → "S" (concave) → "s"→ "v"→ "o" (convex) → "O"→ "$" (concave) → "v"→ "o"→ "s" (convex) → "("→ "O"→ "$"→ "S" (concave) → "v"→ "o"→ ")"→ "T"→ "∧" (convex) → "m"→ "t" (concave) → "F" →"∧" (convex).

It is clear, both outer contours are six angles because of six pairs of convex and concave change. For most sorts of lily flower there are six petals which are constructed by sixangles. For the image in Fig. 4(4), there are eight petals which are constructed by eight-square. Each convex change consists of a group of structural points which can be used to describe convex change in the direction of a chain code. For example, the first group of structural points in Fig. 6(5) is structural points "∧", "F" and "[" which are all convex points to describe the shape of the first convex change, and the first concave change of the contour consists of structural points, "]" and "f" which are all concave points. In some cases, that the shape recognition of other layer images which are in the outer contour is needed. Their colors and shape can describe the property of object image. The series set of linearized lines, curvature and bend angles give the detail of contour description for each convex or concave change. Based on the above similar description, different object image structure patterns can be constructed. Based on these structural patterns, the shape of object images can be recognized. In this paper, the object image structure patterns of some flowers are constructed and used to retrieve on websites or classified. Therefore, if retrieved object image is lily flower, the images in Figs. 2(1-3) are selected. If red lily flower is searched, the images in Figs. 2(1-2) are got. Also, the image in Fig. 2(3) is yellow lily flower. The image in Fig. 2(4) is not lily flower based on the constructed structure patterns of the object images.

3 Conclusion

An efficient and new method is developed to retrieve on websites or classified object images. Color images are separated by color reduction. Layer images are extracted, the related binary images are found. A new method of contour description, linearization, curve and bend angles, and structural points, is introduced based on difference chain codes. The text, color and shape recognition are used. These algorithm can be used to recognize contour shape. Based on text and color information, and shape recognition, the object image which is retrieved or classified can be determined.

References

[1] G. Lu, "Design issues of Multimedia Information Indexing and Retrieval Systems", *Journal of Network and Computer Applications (Academic Press)*, **22**, (**3**), pp.175-198, 1999.

[2] A. Mojsilovic, J. Kovacevic, J. Hu, R.J. Safranck, S.K. Ganapathy, "Matching and retrieval based on the vocabulary and grammar of color patterns" *IEEE Trans. on Image Processing*, **9**, (**1**), pp.38-54, 2000.

[3] N. Papamarkos, A. E. Atsalakis and C. Strouthopoulos, "Adaptive Colour Color Reduction," *IEEE Trans. on System, Man, and Cybernetics* , **32** , pp. 44-56, 2002.

[4] N. Papamarkos, C. Strouthopoulos and I. Andreadis, "Multithresholding of Colour and Gray-level Image Through a Neural Network Technique," *Image and Vision Computing*, **18** , pp. 213-222, 2000.

[5] H. Freeman and L. S. Davis, "A Corner Finding Algorithm for Chain Code Curvature Angle," *IEEE Trans. Comput.*, **c-26** , pp. 297-303, 1974.

[6] F. Moktarian and A. K. Mackworth, "A Theory of Multiscale Curvature-Based Shape Representation for Planer Curvature Angles," *IEEE Trans. Pattern Analysis Mach. Intell.*,**14** , (**8**), pp. 789-805, 1992.

[7] N. Katzir, M. Lindenbaum and M. Porat, "Curvature Angle Segmentation under Partial occlusion," *IEEE Trans. Pattern Analysis Mach. Intell.*,c **16** , (**5**), pp. 513-519, 1994.

[8] Y. Lin, J. Dou and H. Wang, "Contour Shape Description Based on an Arch Height Function," *Patt. Recog.*, **25**, (**1**), pp. 17-23, 1992.

[9] D. Yu and H. Yan, "An efficient algorithm for smoothing binary image contours," *Pro. of ICPR'96.*, **2**, pp. 403-407, 1996.

[10] D. Yu and H. Yan, "Separation of Touching Handwritten Multi-Numeral Strings Based on Morphological Structural Features," **34**, (**3**), pp.587 -599, 2001.

Obstacles to the industrial use of visual programming

Philip T. Cox

Lei Dong

Faculty of Computer Science, Dalhousie University, Halifax,Canada

Abstract

Visual programming has yet to make significant inroads into the world of professional software development. Although professional developers use visual tools for some parts of the software construction process, they generally do not code algorithms visually, even though there is evidence to suggest that there would be significant advantages to doing so.

One of the major barriers to the industrial adoption of visual programming is that software companies and professional programmers invest heavily in the tools they use. This investment is partly monetary, but is to a much greater extent an investment in the time required to learn to use the tools effectively and to mould the development infrastructure to suit the tools. A professional software engineer is more likely to adopt a development tool if it conforms to a standard, official or de facto.

With the long term aim of creating a visual programming environment that allows fluid movement between program visual representation, and textual representation in some language acceptable to professional developers, we present a preliminary study of translating between Java and JGraph, highlighting the difficulties inherent in this task.

1 Introduction

Visual programming has been studied for almost forty years, rather intensively for the last twenty, the increased interest stimulated by various factors, such as the move to graphical operating systems, and graphical interfaces for applications.

Despite this significant activity and the benefits that visual representations have brought to end-users, the direct use of visualisation in software design and development still lags far behind its use in end-user applications, and tends to be limited to higher level organization. For example, there are visual formalisms for modelling software structures, such as Unified Modelling Language (UML), and Computer-Aided Software Engineering (CASE) tools employing visual representations such as UML to facilitate specification. At a lower level, GUI builders allow interfaces to be constructed by direct manipulation of interface objects. Integrated Development Environments (IDE) usually consist of an application framework built on a textual programming language, together with a visual GUI builder, and windows and panels containing scrolling lists and other kinds of controls, providing views of various aspects of a project, such as class hierarchies.

Many visual programming languages have been proposed, offering a variety of visual syntaxes for representing algorithms. At the lower levels of software development, however, professional programmers almost exclusively code algorithms in textual programming languages, despite the fact that studies and practical experience have shown that visual programming languages have much to offer the professional software developer.

In examining the reasons for this, it is instructive to compare two commercial visual languages which have much in common but markedly different levels of success, namely LabVIEW and Prograph.

LabVIEW, developed by National Instruments starting in the mid-80s, is a software development environment for developing process control and data acquisition applications [6]. It has enjoyed considerable commercial success in the engineering and scientific community for which it is intended [1], apparently at least partly due to its visual data flow programming language [13].

Prograph CPX, an IDE based on a visual data flow language [3,8], was aimed at professional software developers. Implemented on the Macintosh, it therefore competed with IDEs such as Codewarrior. Prograph was well-received from a critical point of view, and was used to build both "shrink-wrapped" products and critical in-house applications in a wide variety of application areas. In the mid-90s, Apple Computer commissioned several focus-group studies to determine the viability of Prograph CPX as an industrial development environment, and received a strongly positive response particularly from Windows developers [11]. Despite such positive results and feedback, Prograph has not been a commercial success.

A critical difference between these two visual programming language products is the contrasting requirements of their intended users. The users of a domain-specific tool such as LabVIEW need to solve particular kinds of problems in a very focussed domain where the range of different functionalities is relatively small. They do not require tools which are arbitrarily flexible or extensible. Professional developers, on the other hand, need tools which will allow them to build and maintain any kind of software, and work in conjunction with other tools from different vendors. In short, they need to base their work on tools that conform to software industry standards. This requirement was confirmed by the Apple

study, which noted *"It may be difficult to convert developers who are constrained by organizational standards, large base of existing code,…"* Prograph, despite its considerable benefits, did not have this crucial characteristic.

The dominance of textual programming has a lot to do with the history and evolution of computers, and the rich theoretical foundations, standards and tools that have been built around texual languages. visual programming languages face significant challenges in making inroads into industrial development.

A recently proposed visual language, JGraph, has similar structure to Prograph, but is also closely aligned with Java [9,10]. Proposed originally as a visualisation of the key concepts of Java, JGraph might be a vehicle for introducing the benefits of visual programming at the algorithmic level to industrial software developers. To realise this goal, however, a JGraph implementation would have to provide a variety of tools to allow the programmer to smoothly integrate visual programming into the development process.

It is well recognised that software development environments that provide multiple views of program structures must allow the programmer to choose a view and to move between views as required, while the system maintains consistency between views [5,7]. To achieve a useful integration of JGraph and Java in a development environment, therefore, it would be crucial for the programmer to be able to move freely between visual and textual representations of code.

We report here on a preliminary investigation into this requirement, presenting translations from JGraph to Java and *vice versa*. These do not achieve the desired goal, but serve to illustrate the difficulties of attaining transformations that produce "reasonable" code and are inverses of each other.

2 Introduction to JGraph

In this section we provide a brief description of JGraph, and urge the reader to consult [10] for full details. We will assume the reader is familiar with Prograph [3], and will concentrate our discussion on the differences between JGraph and Prograph and the similarities between JGraph and Java.

Figure 1 illustrates the upper level structures of JGraph, from *project* down to *method declaration*. Reading from the top left, a project consists of a number of packages. In this example **MyProj** contains a package, **MyPackage** consisting of a class MyClass and two interfaces Interface1 and Interface2.

Figure 1: Upper level JGraph structures

The inheritance relationships between these, and an imported class MySuper, are depicted in the top right window. The ornaments ■ and ▼ on the class and interface icons correspond to Java specifiers public and package respectively. The window second from the top on the left, and the one below that, list the imports and fields, respectively, of MyClass. The decoration ◆ on the int field named i denotes the specifier private.

The next window down shows that MyClass has two methods, a static void method BubbleSort with one parameter, and MyMethod which returns int, has two parameters and throws some exceptions, which are shown in detail in the remaining two windows.

To this level JGraph and Java are identicals.. They diverge at the level of expressing algorithms, where JGraph is structurally more similar to Prograph. For example, Figure 2 shows the method BubbleSort in MyClass, a static void method with one int[] parameter, and associated loops inner and outer, and local swap. Every method, loop and local con-

Figure 2: The method BubbleSort, associated repeats and locals

sists of a sequence of *normal* cases, followed by a possibly empty sequence of *catch* cases, and at most one *finally* case. Neither BubbleSort nor its associated loops and local have catch or finally cases.

JGraph does not have *list* as a primitive type. The terminal ⟦┄┄⟧ on the inner loop operation is an *array* terminal, which functions in a fashion analogous to the list terminal in Prograph, indicating sequential processing of array elements. The operations ⟦array set⟧ and ⟦array get⟧ are *array set* and *array get* respectively.

Figure 3: The method MyMethod

Like Java, JGraph is strongly typed, so each of the terminals and roots in the data flow diagram has an associated type, which can be displayed as required.

JGraph provides an exception-handling mechanism somewhat similar to Java's but adapted to the case structure of methods, inherited from Prograph. This is illustrated in Figure 3 which depicts the method MyMethod of MyClass. This method has one *normal* case, which may directly throw an exception by passing it to the throws terminal ⟦!⟧ on its output bar, and may indirectly throw an exception via the operation **GetIt**. MyMethod has one *catch* case, which catches any ArgException thrown in the normal case, and one *finally* case. The special root ⟦■⟧ on the input bar of the finally case transmits outputs from normal or catch cases, if any. The roles of normal, catch and finally cases are analogous to those of try, catch and finally clauses in Java.

3 Generating Java from JGraph

In this section, we describe a translation from JGraph to Java. Our description is necessarily superficial, relying heavily on an examples. Complete details can be found in [4].

Since the upper levels of the two languages are isomorphic, the translation to Java of the package, class and method-declaration levels of a JGraph application is straight-forward, and is left to the reader. The subtle and difficult issues occur at the method-body level, where one must reconcile the markedly different control structures of the two languages. We illustrate some of these issues by considering the translation to Java of the BubbleSort method and associated loops and locals in Figure 2 above.

The code in Figure 4 results from the translation of the BubbleSort method itself. Before we relate this code to the diagram in Figure 2, some definitions are required. First, as

in Prograph, the iteration of a repeat operation in JGraph may stop because a control inside it is activated, or because it has an array input, all the elements of which have been processed. Repeats are thus classified as *controlled* or *counted* respectively. In our example, the repeats outer and inner each contain a *terminate* control, so are controlled. In addition, inner has an array input, so is also counted.

Because BubbleSort contains a controlled repeat, line 5 declares a flag bf which occurs in the appropriate loop condition in the code for outer in Figure 5. Local variables are declared in lines 6-9, corresponding to the similarly labelled roots in Figure 2. Lines 10-12 correspond to the operations, length, 1 and <=. Line 13 results from translating the *terminate* control on the <= operation. Activating a terminate control causes execution of the case in which it occurs to stop immediately. The translated code achieves this behaviour by throwing the special exception ITermException.

Next we consider the translation of the repeat operation outer, provided in Figure 5 below. The repeat inner is both controlled and counted, hence a flag of is declared at line 2, and at line 3, index and upper-limit variables oi and ou are declared for managing the iteration of inner through input arrays. In line 4, the flag bf is initialised in preparation for its role in controlling the loop corresponding to outer beginning at line 8. In line 5, the local variable corresponding to the single simple terminal of outer is declared and initialised. Line 6 declares the local variable corresponding to the loop terminal of outer, and line 7 copies the value at this loop terminal to the corresponding loop root, providing it with a value in case zero iterations occur.

In the body of the loop, line 10 copies the current value of the looped variable to the correponding local variable, then, because a *terminate* control occurs inside outer, the code for the cases of outer are enclosed in a try clause. Note that unlike BubbleSort, outer is a repeat, so in addition to

```
1   public static void BubbleSort(int[] b1)
2   {
3     try
4     {
5       boolean bf;
6       int b2;
7       int b3;
8       boolean b4;
9       int b5;
10      b2 = b1.length;
11      b3 = 1;
12      b4 = b2 <= b3;
13      if( b4 ) throw new ITermException() ;

        _____
        code for outer - see Figure 5

14    }
15    catch(ITermException be)
16    {
17    }
18  }
```

Figure 4: The translation of BubbleSort

halting execution of the case in which it occurs, the *terminate* control will bring iteration to an end. Hence the catch clause sets the flag controlling the loop to `false`. The inside of the `try` clause has similar structure to the corresponding code in Figure 4, with the addition of the assignment in line 27 which updates the value of the loop root of *outer*.

```
1   {
2     boolean of;
3     int oi, ou ;
4     bf = true;
5     int[] o1 = b1 ;
6     int o2 ;
7     b5 = b2 ;
8     while ( bf )
9     {
10      o2 = b5 ;
11      try
12      {
13        int o3 ;
14        int o6 ;
15        int o4 ;
16        int o5 ;
17        boolean o7 ;
18        o6 = o2-1;
19        o7 = o6==0;
20        if( o7 ) throw new ITermException() ;
21        o3 = 1;
          _____
          code for inner - see Figure 6

22        b5 = o6 ;
23      }
24      catch(ITermException oe)
25      {
26        bf = false;
27      }
28    }
29  }
```

Figure 5: The translation of outer

Figure 6 shows the code corresponding to the repeat operation inner. Its overall structure is similar to that for outer, with the following additions. The variables oi and ou used for managing iteration through the input array are initialised in lines 4 and 5. The loop condition includes the control flag and appropriate array-indexing test. Lines 16 and 17 extract the next array element and increment the array index.

Finally, the code produced by translating swap is shown in Figure 7. The notable new feature here is the structure that results from a local (or method, or repeat) which has more than one case. Like *terminate*, a *next-case* control also causes execution of a case to be immediately halted, and is translated in a similar way. The code consists of a `try` clause containing the code generated from the first case, including statements that throw a special exception `ICaseException`, followed by a `catch` clause containing the code for the remaining cases. Hence translating a sequence of *n* cases produces a nested arrangement of *n* `try`-`catch` blocks.

If a local, method or repeat also has `catch` and `finally` cases as illustrated in Figure 3, the code for the normal cases

```
1   {
2     of = true ;
3     int[] i2 = o1 ;
4     oi = 0 ;
5     ou = o1.length ;
6     int i1 ;
7     int i3 ;
8     int i4 ;
9     o4 = o3 ;
10    o5 = o2 ;
11    while (of && oi < ou)
12    {
13      i3 = o4 ;
14      i4 = o5 ;
15      i1 = o1[oi] ;
16      oi++ ;
17      try
18      {
19        int i5;
20        int i6;
21        int i7;
22        boolean i8;
23        i5 = i4-1;
24        i8 = i5 == 0;
25        if( i8) throw new ITermException() ;
26        i7 = i2[i3] ;
          _____
          code for swap - see Figure 7

27        i6 = i3+1;
28        o4 = i6;
29        o5 = i5;
30      }
31      catch(ITermException ie)
32      {
33        of = false;
34      }
35    }
36  }
```

Figure 6: The translation of inner

is nested in a `try` clause, followed by appropriate `catch` and `finally` clauses.

4   Generating JGraph from Java

Now we will consider the opposite transformation, generating a JGraph program from an arbitrary Java program. Two important differences between JGraph and Java that heavily influence this translation are that JGraph is data flow and

```
1   {
2     int s1 = i1;
3     int[] s2 = i2;
4     int s3 = i7;
5     int s4 = i3;
6     try
7     { // first case
8       boolean s5;
9       s5 = s1<=s3;
10      if( !s5 ) throw new ICaseException();
11    }
12    catch(ICaseException se)
13    { // second case
14      int s6;
15      s2[s4]= s1;
16      s6 = s4-1;
17      s2[s6] = s3;
18    }
19  }
```

Figure 7: The translation of swap

different variables (roots) in JGraph have distinct representations. Hence JGraph has the *single-assignment* property, variables (roots) are uniquely defined, and the scope of each variable is limited to the case in which it occurs. Java variables are not subject to these condition.

The translation process consists mainly of a sequence of transformations of the Java program that remove these and other differences, as follows.

1. *Control structures:* All loops are translated to `while` loops, and `switches` to `ifs`.

2. *Multiple declarations:* variables are renamed to ensure that names are used only once, and declarations moved to the beginning of the method.

3. *Nested sequences* of statements are flattened.

4. *Returns:* If a method has return type `T` and body `<body>`, then `<body>` is replaced by

       try{<newbody>} catch (IRet_T e) {return result;}

   where `<newbody>` is obtained by replacing every statement of the form "`return  E;`" in `<body>` by "`throw  new IRet_T(E);`", and `IRet_T` is defined as follows:

       public class IRet_T extends Throwable {
         T result ;
         public IRet_T(T x) { result = x; }
       }

5. *Unary operators:* All occurrences of `++x` and `x++` are replaced by `(x=x+1)` and `((x=x+1)-1)` respectively. The operator `--` is dealt with similarly.

6. *Conditions:* Each statement of the form

       if (<condition>) Bt else Be

   is replaced by

       c=<condition>; if (c) BT else Be

   and a declaration for `c` added. Each statement of the form

       while (<condition>) B

   is replaced by

       c=<condition>; while (c) { BT c=<condition>; }

   and a declaration for `c` added.

7. *Embedded assignments:* Assignment expressions occurring in other expressions are removed and added as statements.

8. *Missing else* clauses are added to conditionals.

9. *Exceptions:* In Java, a value assigned to a variable before an exception is thrown will be available in the code executed after the exception, provided this code is in the scope of the variable. In JGraph, however, when control transfers out of the case where the exception is thrown, the only values computed in the case which will still be available are those passed as outputs from the case and these will be available only in the corresponding finally case. Hence we need to rearrange the Java code so that in the JGraph translation, a value assigned to a root before an exception is thrown will be available in the computation conducted after the exception. To accomplish this, each statement that may throw an exception is replaced by code that will generate an appropriate exception without actually throwing it, and avoid executing code that would not have been executed if an exception had been thrown in the original code. The exceptions created are assigned to a variable,

declared at the beginning of the method to be of class `Throwable`, and used to throw an exception, if necessary, only at the end of the method. In this way the various throws are reduced to assignments to this variable.

10. *Isolating variables:* As mentioned above, the scope of a variable (root) in JGraph is the case in which it occurs. To ensure that the Java program has this property, certain variables are added, as follows. If S is a conditional statement, for each variable x that occurs both inside and outside S, a new variable x' is declared, and

   • if x is assigned a value in S then `x = COPY(x');` is added to the end of the `then` and `else` bodies of S.

   • if x is used before it is assigned in S, or x is assigned in only one body of S, then `x' = COPY(x);` is added before S.

   Finally, all occurrences of x in S are replaced by x'. Similar transformations are applied if S is a try statement or a loop. In the latter case, a statement LP(x') will also be appended to the end of the loop body if x is both used and assigned in S, and used before it is assigned. These `COPY` and `LP` statements render the program uncompilable. They indicate how variables inside a structure correspond with variables outside, and identify loop variables.

11. *Removing multiple assignments:* Step 2 guarantees that all variables in a method are uniquely named so two assignments to variables of the same name must be assignments to the same variable. In looking for multiple assignments, we can restrict our attention to assignment statements since embedded assignment expressions were removed in step 7. We also note that because of step 8, every conditional statement has an else clause.

   We say that two assignments are *compatible* iff the variables being assigned are different, or the assignments are in different clauses of a try-catch statement or an if-then-else statement; otherwise the two assignments are incompatible.

   If x = A; and x = B; are two compatible assignments occuring in this order, every occurrence of x in and following the second assignment is replaced by a new variable x', except that if LP(x) occurs in this portion of the code, then it is replaced by LOOP(x,x'). A declaration for x' is added to the beginning of the method. This step is repeated until it no longer applies.

12. *Variable to variable assignments:* All assignments of the form x=y; are deleted, and each occurrence of x is replaced by y.

As an example of this process, consider the Java method `BubbleSort` in Figure 8 and the result of preprocessing it, shown in Figures 9 to 10. The reader is invited to relate it to the above steps. Note that some steps do not apply in this example, namely 3, 4, 7 and 9.

The preprocessed code has the single assignment property, and although the scope of each variable is the whole method body, all its occurrences lie at one level of the nested code, except for occurrences in the "fake" COPY and LOOP statements.

```
1   public static void BubbleSort(int [] a)
2   {
3     int i, j, n, tmp;
4     n = a.length;
5     for(i = 0; i < n-1; i++)
6     {
7       for(j = 0; j < n-1-i; j++)
8         if(a[ j + 1] < a[j])
9         {
10          tmp = a[j];
11          a[j] = a[j+1];
12          a[j+1] = tmp;
13        }
14    }
15  }
```

Figure 8: The method BubbleSort

```
1   public static void BubbleSort(int [] a)
2   {
3     int i, j, n, tmp;
4     int j1 ;
5     int i1 ;
6     boolean c1 ;
7     boolean c2 ;
8     boolean c3 ;
9     boolean c3p ;
10    int jp ;
11    int[] ap ;
12    boolean c2p ;
13    int jp1 ;
14    int[] ap1 ;
15    int np ;
16    int ip1 ;
17    boolean c1p ;
18    int ip ;
19    int[] ap2 ;
20    int np1 ;
21    boolean c1a ;
22    boolean c1pa ;
23    boolean c2a ;
24    boolean c2pa ;
25    int ia ;
26    int ja ;
27    n = a.length;

      ────────────────────────────────
      code for outer loop - see Figure 10

28  }
```

Figure 9: Preprocessed method

```
1    i = 0;
2    c1 = i < n-1 ;
3    c1p = COPY(c1) ;
4    ip = COPY(i) ;
5    ap2 = COPY(a) ;
6    np1 = COPY(n) ;
7    while( c1p )
8    {
9      j = 0;
10     c2 = j < np1-1-ip ;
1      c2p = COPY(c2) ;
2      jp1 = COPY(j) ;
3      ap1 = COPY(ap2) ;
4      np = COPY(np1) ;
5      ip1 = COPY(ip) ;
6      while( c2p )
7      {
8        c3 = ap1[ jp1 + 1] < ap1[jp1];
9        c3p = COPYb(c3) ;
10       jp = COPYi(jp1) ;
11       ap = COPYr(ap1) ;
12       if( c3p )
13       {
14         tmp = ap[jp];
15         ap[jp] = ap[jp+1];
16         ap[jp+1] = tmp;
17       }
18       else
19       {
20       }
21       j1 = jp1+1 ;
22       c2pa = j1 < np-1-ip1 ;
23       LOOP(c2p, c2pa) ;
24       c2a = COPY(c2pa) ;
25       LOOP(jp1, j1) ;
26       ja = COPY(j1) ;
27     }
28     i1 = ip+1 ;
29     c1pa = i1 < np1-1 ;
30     LOOP(c1p, c1pa) ;
31     c1a = COPY(c1pa) ;
32     LOOP(ip, i1) ;
33     ia = COPY(i1) ;
34   }
```

Figure 10: Outer loop

Figure 11: The resulting JGraph method and associated repeats and local

The preprocessed code is directly transliterated into JGraph in the obvious way, obtaining the method and associated repeats and local shown in Figure 11. Note that, although sorting an array is a prime application for the array processing feature of JGraph, as shown in Figure 2, the generated program does not use it. A translation that would include this feature would have to spot potentially quite complicated patterns.

Our example does not illustrate preprocessing step 9 which "reshapes" with the exception handling code of a Java program. This step addresses a crucial issue, namely that in a Java program, a value computed for a variable prior to an exception arising will be available to the exception handler, provided it is within the scope of the variable. In JGraph, the handler is in a separate case from where the exception originates, so to ensure that the JGraph code resulting from the handler receives the same information, we essentially need to replace the exception handling structure by nested conditionals.

5 Discussion

In this section we discuss the characteristics of the code generated by the two translations.

In JGraph, changes in the normal progression of execution through a data flow diagram are accomplished by controls, which cause abrupt termination of execution and transfer of control elsewhere. To mimic this in Java, we have had to resort to exception throwing. Hence the generated Java code relies heavily on the exception mechanism for ordinary control flow, and contains the special exception classes needed to accomplish this.

A Java program generated from JGraph will contain a large number of variables resulting from single assignment and the fact that in JGraph variables are local to cases.

During translation of a Java program to JGraph, try-catch-finally statements are largely eliminated and replaced by conditionals. Some try-catch structures are reintroduced to deal with exceptions in a very localised fashion. The end result is that in the JGraph code, there will be no finally cases, and each catch case simply outputs the caught exception. Any normal case that may throw an exception contains at least one operation which calls a method that may throw an exception, but has no exception root on its output bar. Finally, the only place an exception terminal may occur is on the output bar of the case of a method. Note that every method will have exactly one case.

To deal with the fact that a Java method can halt execution and return to its caller from any point, the generated JGraph uses special exception classes, instances of which are thrown to mimic the arbitrary Java return. Clearly a large number of these classes will be generated if a program includes many methods with many return types.

There are various other trivial characteristics of a JGraph program generated from Java. For example, the conditions of conditional statements and loops are just variables, and the repeat operations do not have any array inputs.

Although we have defined two transformations that produce correct equivalent programs, these translations are not simple inverses of one another. We have not shown an example of applying the two translations end-to-end, however, it should be obvious to the reader that doing so will produce code that is quite different from the original, regardless of whether we start with a Java program or a JGraph program. An example of this non-reversibility is as follows.

When a JGraph program is translated to Java, a sequence of cases in the original program corresponds to nested try-catch-finally statements in the resulting Java program. During the reverse translation, these nested try-catch-finally statements are replaced by nested if-then statements, each of which becomes a two-case local operation. If the original sequence consisted of more than two cases, then it will not be the same as the final JGraph obtained after two translations, since it has case sequences of length at most two. In fact, even if the sequence had only two cases, translating to Java produces three conditionals, each of which will become a two-case local when the reverse translation is applied.

One question which arises from the above discussion is the following. If we translate back and forth several times, does the process iterate towards some fixed point? The answer to the question is clearly negative, as illustrated by the above example, where each round of translations multiplies the number of two-case locals by three.

Because of the major differences between JGraph and Java, both translations introduce special mechanisms and structures, resulting in code that would not be written by a programmer proficient in the target language. For example, a Java programmer is unlikely to write code which embodies the single assignment rule of data flow and achieves conditional execution by throwing exceptions. Similarly, a proficient JGraph programmer is unlikely to create a program in which exception throwing and handling is replaced by deeply nested conditional statements.

6 Concluding remarks

As a step towards making a visual programming language acceptable to professional software developers, we have studied the issues involved in translating between Java and the visual data flow language JGraph. The translations we have devised, although correct, do not produce code which properly exploits the facilities of the target languages, or would be acceptable to a software developer. Neither are the translations inverses of one another.

Clearly, the translations we have presented are quite naïve in many respects, and significant improvements might be achieved via more sophisticated pattern matching, taking a more global view of an input program. However, although Java and JGraph are similar in many respects, their differences are sufficient to cast doubt on the feasibility of revers-

ible translation.

Translating programs from one representation to another is by not means a new idea; compilers and other language translators do this routinely. However, we are interested not in translation to facilitate execution, but in giving the programer the ability to view code in a way which is best suited to the task currently at hand. For example, a pictorial view of a particular arrangment of program structures may help the programmer spot a logic error, whereas a textual view of sequence of complex formulae may be more appropriate. Other projects have a similar goal, for example Simonyi's intentional programming [13], in which the programmers design intents are represented in the form of an *IP tree* in a database and can, given an appropriate translator, generate a corresponding program in a desired textual programming language. Visual programming is limited to manipulation of the IP tree. The issues of translating between visual programming languages and textual ones are not addressed.

In our continuing work in this direction, we are investigating changes to JGraph that will not only facilitate translation to and from Java, but also improve the usability of the language. In particular, we are presently revising the limited exception-handling capabilities of the language. A preliminary proposal for an exception handling model is presented in [2].

7 References

[1] E. Baroth and C. Hartsough, Visual Programming in the Real World, in *Visual Object-Oriented Programming: Concepts and Environments*, M.M. Burnett, A. Goldberg, T.G. Lewis (eds), Manning Publications, Greenwich CT (1995), 21-42.

[2] P.T. Cox and with S. Gauvin, Exceptions in Visual Data Flow Programming Languages, 2003 International Conference on Visual Languages and Computing, in *Proc. 9th Int. Conf. on Distributed Multimedia Systems*, Miami FLA (2003), 360-367.

[3] P.T. Cox, F.R. Giles and T. Pietrzykowski, Prograph: A Step toward Liberating the Programmer from Textual Conditioning, *Proc. IEEE Workshop in Visual Languages*, (1989), 150-156.

[4] L. Dong, *Transforming Visual Programs into Java and Vice Versa.* MCS Thesis, Faculty of Computer Science, Dalhousie University (2002).

[5] J. Grundy, J. Hosking, S. Fenwick and W. Mugridge, Connecting the Pieces, in *Visual Object-Oriented Programming: Concepts and Environments*, M.M. Burnett, A. Goldberg, T.G. Lewis (eds), Manning Publications, Greenwich CT (1995), 229-252.

[6] G.W. Johnson, R. Jennings, *LabVIEW Graphical Programming*, McGraw-Hill (2001)

[7] S. Meyers, Difficulties in integrating multiple view editing environments, *IEEE Software*, vol. 8, no. 1 (1991), 49-57.

[8] Pictorius Inc., *Prograph CPX Reference Manual*, (1996).

[9] C.C. Risley and T.J. Smedley, Visualization of Compile Time Errors in a Java Compatible Visual Language, *Proc. IEEE Symposium on Visual Languages*, Halifax, (2000), 22-29.

[10] C.C. Risley, *JGraph: A Java Compatible Visual Language*, MCS Thesis, Faculty of Computer Science, Dalhousie University, (2000).

[11] R. Rowe and J. Burns, Prograph CPX Qualitative Research, Riley Rowe and Associates, (1995)

[12] C. Simonyi, *The death of computer languages, the birth of Intentional Programming*, Tech. Rep. MSR-TR-95-52, Microsoft Research, (1995).

[13] K.N. Whitley and A. F. Blackwell, Visual Programming in the Wild: A Survey of LabVIEW Programmers, *Journal of Visual Languages and Computing*, Vol. 12, No. 4, (Aug 2001), 435-472.

Proceedings

The 7th International Conference on Visual Information Systems (VIS'2004)

Co-Editors

Jian Kang Wu, Institute for Infocomm Research, Singapore
Clement Leung, Victoria University, Australia

An Architecture for Interactive Query Refinement in Sensor-based Information Fusion Systems

G. CASELLA*, S. K. CHANG**, G. COSTAGLIOLA*, E. JUNGERT***, X. LI** *and* T. HORNEY**

*Dept. Informatica e Matematica, University of Salerno, ITALY {giocas, gencos}@unisa.it

**Dept. of Computer Science, University of Pittsburgh, USA {chang, flying}@cs.pitt.edu

*** Swedish Defence Research Agency (FOI), Sweden {jungert, tobho}@foi.se

Abstract

An architecture for query execution and interactive query refinement on heterogeneous data from spatially distributed sensors is proposed. The objective is to define a system in which it is always possible to add new data sources and new algorithms for data processing and user query execution. The query language that forms the basis for query formulation is called ΣQL. Ambigous and vague query can be refined as well. An ontological knowledge base to model fundamental architecture components and their relations is introduced. The method to choose algorithms that can execute user queries based on available data, the Pequliar interactive system and the Reasoner are described. The Reasoner automatically or semi automatically carries out an iterative information fusion step by means of rules that are either automatically launched by the Reasoner or defined by the user.

1. Introduction

In this paper we define an architecture for query execution and interactive query refinement on heterogeneous data from spatially distributed sensors.

The objective of this work is to define a system in which it is always possible to add new data sources and new algorithms for data processing and user query execution. Ambiguous and vague query can be interactively refined as well. The ΣQL query language forming the basis for this is described in [4],[5]. For example a possible query is to find an object (such as a car) in an area or detect particular events (such as a region covered by a flood).

Generally speaking, to formulate a query such as: "Find cars in a region 5 hours before the region is covered by a flood", the user needs to specify the objects to be recognized (cars and flood), the area of interest (the region), the time of interval (when), the relations of interest among the objects ("Find cars in a region" and "5 hours before the region is covered by a flood"), the output format.

To specify the area of interest the system provides the user with maps that are stored in a database. To let the user indicate predefined objects to be recognized these objects are described in the ontological knowledge base [1,7,8]. Furthermore the system should enable the user to define new objects.

The user can interact graphically with the objects and the maps, and provide a time interval (similarly to the case of the sentient map [2]) and all the other information needed to formulate the query.

A user interface should give the feedback to the user to let the user know whether the defined query is admissible at that moment or during some later time period, or it cannot be executed until other sensors and/or algorithms are added to the system.

Once the initial query has been specified, the system needs to complete it with the missing "technical" information such as: What are the characteristics of the objects needed to satisfy the relations (i.e. the query)? What sensors can be used in that area at that time to recognize these characteristics optimally? What algorithms can be used with the selected sensors for the particular use?

In the above mentioned query, the characteristics of interest to satisfy the relation: ("Find cars in a region" and "5 hours before the region is covered by a flood") are Position (area covered) and Time for both the car object and the flood object. Hence the relation becomes:
Position_car covered_by Position_flood AND
Timecar= Timeflood −5h

In the above mentioned query, the characteristics of interest to satisfy the relation: ("Find cars in a region" and "5 hours before the region is covered by a flood") are Position (area covered) and Time for both the car object and the flood object. Hence the relation becomes:
Position_car covered_by Position_flood AND
Timecar= Timeflood −5h

The question now is: will the system need to infer the characteristics of interest, or will the user need to specify the characteristics by expressing the relations in more detail? A possible intermediate solution is to add to the ontological knowledge base all the relations of interest in a certain domain. The user has the possibility of browsing

and defining the object characteristics from their ontological description while creating the query or, at a higher level, selecting predefined relations between objects from the ontological knowledge base, and the definition of the relation in OKB tells the system what are the values of the characteristics of interest.

Furthermore, once all the information has been gathered a complete ΣQL query can be built. The query processing phase proceeds then in two separate parts:

1. Low level query processing: to recognize all the objects and properties needed to run the high level query.
2. High level query processing: to provide the final result by extracting only the information satisfying the required relations among objects.

In the first phase, given the objects, their characteristics, their spatial and temporal position and all the knowledge coming from the ontological knowledge base about objects, sensors, algorithms and conditions, the system is now able to build an initial plan of sensors-algorithms applications. The aim of this first phase is to use, in the best way, all the data sources and algorithms available to execute a user query. Crucial points are the correct management of available data sources, the choice of the most suitable algorithms for query execution and a good presentation of the results.

The execution of the plan on the real data might modify the plan itself in order to get more precise results. The plan uses sensor data fusion [6] operations to make all the data converge to a complete final result. In the second part the query is run against the recognized items. An interactive system called Peculiar is used to resolve the ambiguity in the high level queries. According to this approach the ΣQL query is seen as logically divided into two distinct parts: the modifying plan and the relational constraints (WHERE clause). Alternatively, the ΣQL query processing phase might take care simultaneously of the sensor information and of the relational constraints. However the query will have to change during its execution because of the plan adaptation. This makes the whole system much more complex and less structured.

In our approach we consider two kinds of algorithms. The first ones are recognition algorithms and their goal is to detect something from available data. The second ones are fusion algorithms, their goal is to fuse recognition algorithm results or other fusion algorithm results to obtain more information. We consider the case in which more than one algorithm is suitable to execute the user query and we present a method to choose the best.

In section 2 we present an ontological knowledge base to model fundamental architecture components and their relation. In section 3 we describe a method to choose algorithms that can execute user queries based on available data. In section 4 we illustrate and briefly describe the system architecture. The Reasoner and reasoning approaches are discussed in Section 5. Section 6 presents some fusion query examples. Finally in Section 7 we give the conclusions.

2. The Ontological knowledge base

The ontology is used to model fundamental architecture components and their relations. A similar approach is presented in [7,8]. All concepts used in the architecture with their properties and relations are modelled in the ontology.

In this section all components are described:

Data Source. The concept "Data Source" models a generic entity that can provide data. A Data Source has a position, expressed in global coordinates (for example longitude and latitude), that points the place where the sensor is. Data Source has also an active range that represents the area where data are acquired and is expressed in relation of the data source position. Data source produces a sequence of Perception Source (see Figure 1), which corresponds to a certain sensor data type such as CCD, IR, LASER. An algorithm may input only a certain sensor data type, or a set of data type, and so can work only with data sources that provide such type of data.

Context Source. The background or proper context in which an object may occur and that is subject to a query in ΣQL can be seen as a data source used to enhance the outcome of the query. This was similarly discussed in [4] where the "local view" and the "global view" were introduced as means for query refinement. Here the two views are jointly called Context Source.

Query input. Query input includes information delivered by the users including such information as area-of-interest (AOI), (time) interval of interest and requested object types. In this source, AOI may be used to determine the extension of the context source. Second to this the Meta Data source is used to determine whether there are any data available from any source in Perception Source that correspond to AOI and where Perception Source corresponds to, e.g. data from a sensor. The latter type of sources can be seen as primary sources that are always used as query input.

Dependency Tree. The Dependency Tree that is generated internally to each query can also be used as a data source in cases where queries about query result are of concern.

Query Output. The result of any given query can be also used in an iterative query. Altogether, the concept of Data Source becomes much more complex when applied to iterative queries generated either in dialogue with or automatically by Pequliar. Although this may cause difficulties in composing iterative queries it also extends the number of possible queries, which in turn, makes ΣQL more adaptive to complex problems related to, for instance, higher orders of information fusion. That is, in applications like situation and impact analysis and for this reason it becomes possible to talk about context sensitive object assessment where context refers not just to the proper context but to all the data sources given in Figure 1.

Perception Source. This concept models the data type that a sensor produces. It is introduced to associate data sources and algorithms to process data; a data source outputs a particular sensor-data-type, and an algorithm

can work on a particular data type (or a set of them) in input. A Perception Source has a name and other additional information. As additional information we consider, for example minimal and maximal rate of data sources that provide this data type. Examples of Sensor Data Type are IR, CCD, LASER, TEXT, etc.

Data-Unit. This concept models a single data unit. A Data Unit has a type "Perception Source" that describes what kind of sensor has produced it, and a timestamp that shows when it has been produced. Timestamp can be expressed in relation to GMT.

Algorithm. The algorithms have a name that identify them unambiguously and are of two types: "recognition algorithm" or "fusion algorithm".

Recognition Algorithm. The aim of the recognition algorithms is to detect one or more 'characteristics' in the data types to be processed. The data type (or the set) in input is a Perception Source and is produced from a "Data Source". By the term "characteristic" we mean, for example, characteristic-shape (a car), characteristic-color (red object), characteristic-motion (objects that move), characteristic-crash (objects that crash), characteristic-text (a particular sentence in a document or in a image), etc. Every characteristic belongs to the concept "Recognizable Characteristic".

Fusion Algorithm. The aim of a fusion algorithm is to fuse information produced by other algorithms (fusion and / or recognition algorithms) to obtain better results. Fusion algorithm's input is then the output of a set of algorithms. As recognition algorithms, fusion algorithms have a set of "Recognizable Characteristics" that represent what the algorithm can recognize from the input data.

Recognizable Characteristics. The introduction of the extended set of data sources enables the system to answer queries about proper-background and similarity. Proper background is a construct, which can be used to ask the system if a certain object that has been found is present in a proper background, e.g. a bus is properly situated on a road but it is not proper to find it in the middle of a lake. Similarity, in this case, means similarity between objects, i.e. is a given object type similar to another object type. For example, is a car similar to a vehicle or a bus similar to a building?

This concept of recognizable characteristics models the set of characteristics (or events) that the data fusion and recognition algorithms can detect. A particular recognizable characteristic is the "object relationship". For example the system can detect "two objects that crash" or if some objects are proper in which backgrounds (proper-background) and which objects are similar to which other objects (similarity).

Objects that can be searched for and background types in the ontology, allow us to define object relations such as is-proper-in-background between "Recognizable Object" (the base concept for objects that can be searched for) and "Terrain" (the base concept for backgrounds). Now, for each "Recognizable Object" an instance of the proper-in-background relation is created to each Terrain concept describing a background where it is proper to find it. It is, of course, possible to do the reverse, i.e. define a relation is-NOT-proper-in-background instead and create instances of that relation between "Recognizable Object" instances and Terrain instances where the object is not proper in the background. With either definition, it is now easy for the system to check if it is proper to find a certain object in the background where it was found.

Template. This concept models a concrete example of something that a user can be interested in. It has a set of "Recognizable Characteristic" (for example if the template represents a car, the recognizable characteristic is the shape, if the template represents "a red car that is moving", the recognizable characteristics are the shape-color-motion). A source file or other information can be associated to a template, for example "car-shape.jpg" can be associated to a car shape or RGB(FF,00,FF) to a color. A User interface will use templates to help a user in formulating the queries.

Recognizable Object. This concept model the objects that can be recognized by the recognition algorithms. Recognizable objects further divided into mobile and immobile objects.

External Condition. This concept models the external conditions in which a sensor has collected data. External conditions are "Weather Condition" and "Light Condition" and have values belonging to the "Discrete Strength Value" concept.

Maps View. This concept models all the information required from a user interface to enable a user to select the territory of interest. Figure 1 shows a graphical representation of the ontology modeled in a hierarchical manner known as ontology tree. The hierarchy has the ultimately general concept called thing at the top. All other concepts inherit directly or indirectly from thing. This means that "everything is a thing". The concept thing has no properties and no relations; it just acts as the parent of all other concepts. The hierarchy is organized so that more specialized concepts appear further down the inheritance chain.

When the ontological structure has been created it is populated with instances becoming a knowledge base called ontological knowledge base.

3. The Query Execution Planner

The aim of the query execution planner is to choose the more suitable algorithms, available to the system, to execute user queries in relation to available data. When data-sources and algorithms for data processing have been chosen a representation of this choice is transferred to the ΣQL query builder.

To describe the query execution planner we use a wide hierarchical structure given in [4]. We represent the query execution process with a tree for every area with data. A single node is given in Figure 2a.

Figure 1. The ontological knowledge base

The term "Sensor Data Type" represents the type of data against which we execute the query. The term "Spatial Coordinates" and "Temporal Coordinates" refer to the area and time in which data are retrieved. The term "Object to Recognize" represents the object (or the event) that we want to recognize and the term "Characteristic Set to Recognize" represents the characteristics that we need to recognize to detect the "Object to Recognize". An arrow from node A to node B denotes that the output of A is the input of B.

We present the steps to build the tree, called sensor dependency tree, reminding that all the information about entities (algorithm, sensors, external conditions) come from the ontological knowledge base modelling and storing them.

Step 1: The system analyzes a user query to create tree leafs. Every leaf represents a Data-Source that can provide data about the area and the time interval to which the user is interested. In particular, every leaf represents a Data-Source with an active range that intersects the area of interest for the user and that can provide Data-Units with a timestamp that belongs to the time interval to which the user is interested.

Step 2: The system processes the user query and extracts the characteristics to be recognized to execute the query. In relation to these characteristics the system tries to find an available algorithm that is able to detect these

characteristics and so to execute the query. If there exists a unique algorithm to execute the query the system transfers all the necessary information to the Σ-Query Plan Builder to execute it. The aim of the Σ-Query Plan Builder is to build a ΣQL query in relation to the available data and algorithms as presented in [5]. If there exists more than a pair (algorithm, data source) available to execute the query, the system chooses the one from which the best results are attended on the base of the algorithm certainty range and external condition in which data are retrieved. An approach to make this choice is presented in [7]. If there is not any pair (algorithm, sensor) to execute the query the system executes step 3.

Step 3: The system considers all the possible user query sub-sets and for every subset it tries to find an algorithm to execute in relation to the available data. For example if the user is interested in finding "all red cars in an area" a subset of this query is to "find all cars", another subset is to find "all red objects". For every subset if there exists more than a pair (algorithm, data source) available to execute the query the system chooses the one from which the best results are attended. For every partial result obtained the system tries to find a fusion algorithm that is able to improve that partial result. Improving results means to execute the most part of the user query and then to recognize the largest number of characteristics.

The system tries to improve results until the user

Figure 2. (a) A node in sensor dependency tree (b) a sensor dependency tree

query is totally executed or there are not other available algorithms. Finally the system transfers all the necessary information to the Σ-Query Plan Builder to execute the query (or partial queries). Here we present a short example in which we suppose that the user is interested to find all "red cars that are moving" in a particular area and in a particular time interval. In this query "Recognizable Characteristic" are Shape (car) , Color (red) and Property – Movement (moving object). Furthermore we suppose that the user is interested in an area called "locality" and in a particular time interval called "period". Considering a given Data Sources and Algorithm sets available we have the sensor dependency tree in figure 2b.

Step 1: To find an object located in *locality* in the time interval *period* considered, there are available data from three perception sources, i.e. in this case sensor types. The perception sources are IR, CCD and Laser.

Step 2: To find a particular moving object of a given color in the system there are no algorithms that can work on the available data source types.

Step 3: In the system there exists an algorithm, "*alg 123*", that can detect a particular shape in data provided from a LASER Data Source. In the system there exists an algorithm, "*alg 185*", which can detect a colored shape in data provided from a CCD. Finally there exists a fusion algorithm, "*alg 13*", to execute user query fusing output of recognition algorithm 123 and 185.

If the system can not execute the user query totally the plan builder creates more than one tree. Every tree represents the execution of a query subset. In the previous example if it does not exist any fusion algorithm to improve the results of the recognition algorithms, the system can only find cars and red moving objects in *locality*. The Output Manager presents these partial results to the user.

4. The Architecture of the ΣQL System

In this section we present the architecture (see figure 3) to define and execute queries on heterogeneous data. Every piece of information about the sensors, available data sources and the algorithms are stored in the ontological knowledge base. The ontology also stores object instances

in which the user can be interested, the user interface presents these instances as templates. A user defines queries through this interface. The user interface uses an Input Manager and an Output Manager to help the user in query definition and to present his/her results. The Input Manager uses the Map Multilevel View Manager to show the user a representation of the territory in which he/she can define a query. When a query has been defined it is transferred to the Query Execution Planner. The Query Execution Planner creates the sensor dependency tree(s) as described above. If the query cannot be executed in its entirety the system calculates what is the best part of the query that can be executed and notifies this part to the user.

Using the Query Planner results the Σ-Query Plan Builder builds the Σ-query or queries using the ΣQL language and then the Query Executor execute/s it/them.

The result of a ΣQL-query is generally the object types requested by the user including also the recognizable characteristic such as attribute and status values of these objects. Example of attributes are color, size etc. while the status of an object generally corresponds to such characteristics as position, orientation or speed etc.

The difference between an attribute and a status value is basically that an attribute is not subject to change in the short range of time, that is, the color of a vehicle may change but not within the time frame of concern to the user. Status values may change within a very short time frame that may be less than seconds; consider for instance position and speed.

Common to all the information returned by ΣQL is that the type attribute is associated with a belief value. For the most part, such belief values are given to indicate to what extent the result of a query can be believed. In the most general case the belief values are just given for the object types and from each type of sensor data and eventually there is also a belief value given as a result of the fusion process that takes place for the majority of the queries; this is due to the use of multiple sensor data sources. Cases when fusion is excluded may occur just for simple and trivial queries.

Other information from the query processor that might

Figure 3. Architecture of the Sigma System

be of concern for the Reasoner are for instance the quality of the data in the area of interest given by the user. To determine the source data quality for a certain area of interest the corresponding meta-data will be required. This must, however, be subject to further research.

Updated, and hopefully more informative, belief values will be achieved through a reasoning step in Pequliar that includes the generation of new and more elaborate queries that will be executed subsequently. Input to this reasoning step is mainly the output, that may include the dependency tree information, from the query processor, the context information and metadata (see also the data sources in Figure 1). Secondary to this is the applicable ontological information. The meta-data is used to select the portion of the context information that corresponds to the area of interest (AOI). Once the Reasoner has come to a conclusion in its process a new and elaborate query is created and executed; thus quite often producing an adjusted belief value that better mirrors the situation in focus, that is a more informative belief value may have been achieved. The query results (partials or totals) and their quality are managed by the Output Manager that present them to the user. Result quality is given based on relation to the original user query, the certainty range of the algorithms used and the external conditions. The output manager gives also to the user the chance to view data from which query results have been obtained.

5. The Reasoner

The Reasoner accepts the output from the Query Processor, and either selects a reasoning rule by itself or by input from the user.

The output of the query processor is a collection of entities that are the results of query processing, such as "trucks" recognized by the Query Processor. The Reasoner selects an applicable rule from the following space S, which is the Cartesean product of the sub-spaces including sources, objects to be recognized, attributes of objects, time, location and spatial/temporal/semantic relations. In other words:

$S = Source$ **x** $Object$ **x** $Attributes$ **x** $Time$ **x** $Location$ **x** $Relations$

For example, the Reasoner may need to pick a rule that is applicable to a different source, to recognize a certain type of objects with attributes in a certain range, in a certain time interval, for objects in a certain spatial location, and satisfying certain relations. The Reasoner searchers the rule base to select an applicable rule. The rule could be a query template, which is then substantiated and sent back to the Query Processor. If the Reasoner cannot select a rule by itself, either because the rule base is not yet populated or because the space S is not properly defined, the Reasoner can accept input from the user. The next time, such rules constructed by the user is remembered, forming one part of the scenario.

Pequliar should be able to carry out a number of further operations related to a number of different applications that generally are of spatial and/or temporal nature. Examples of applications where the Reasoner need to be involved may include:

- tracking of objects
- solution of the association problem
- aggregation of objects
- prediction of future object behavior in space and over time
- determination of complex object relationships

Determination of the result of these operations is carried out by the Reasoner by means of the learned rules in combination with the metadata and the available context information i.e. information from all the data sources. In this way new and more comprehensive queries can be created from templates in the rule base. These queries are then executed by the ΣQL query processor. This may lead to a situation that requires a second invocation of the Reasoner that takes place after the comprehensive query have been processed. In this way the Reasoner will be able to learn from the generation of the elaborate queries. However, in the above more comprehensive applications it may not be sufficient to just run a comprehensive query but also to take a step further and perform higher level information fusion, e.g. situation analysis but this is outside the scoop of this work and must be subject to further research efforts.

6. Iterative Query Refinement for Information Fusion

The typical tasks for iterative information fusion, can be grouped into three types.

The reasoning process as carried out here depends on whether the belief values that are output from any user query has got a value that is uninformative or simply that the user wants to extend the original query in some way.

The first type of queries require the generation of new and elaborate ΣQL queries. An example of this type of query is *"are there any other objects in the proximity of the retrieved object that are similar or of equal type as the retrieved object?"* Proximity refers to the AOI or an extension of it and similarity to the ontology; the reasoner creates new elaborate queries from the templates.

A simple way of handling similarity would be to define two objects as similar if they have a common ancestor in the ontological hierarchy no more than i steps up the inheritance chain, where e.g. i=1 or i=2. Assuming that i=2 then a Tank and a Car would be similar according to the ontology in Figure 1, whereas a Truck and a River is not.

An example of ΣQL query of the first type is presented in the follow, while some other examples can be found in [3]

Select $object_k.type$, $object_k.position$, $object_j.type$
cluster * alias $object_k$
from PerceptionSource
where $relation(AOI, object_k)$ = 'inside'
    and $object_k.type$ = 'truck'
    and $object_j.t$ = $object_k.t$
    and $distance(object_k, object_j) < \delta$
    and $similar(object_k, object_j)$
    and $object_j$ **in**
    Select $objedct_i.type$, $object_i.position$
    cluster * alias $object_i$
    from PerceptionSource
    where $relation(AOI, object_i)$ = 'inside'
        and $object_i.t$ = t_{given}
        and $object_i.type$ = 'truck'

The second type of queries require generally the invocation of particular functions that basically are concerned with the resolution of the association problem that may occur in just a single case or as a part of a tracking task. Examples of such functions are the determination of whether topological relations are true or false.

An example of such queries is "Can the retrieved object be associated with an earlier single observation?" This query type requires the solution of the association problem.

The third type of queries requires only investigation of the result of the various sub-queries of the user defined query, that is in many cases an inspection of the content of the dependency tree. An example is : "Did any sensor (data sources) contribute to the result in any extreme way?". This refers to the single belief values from the various sensor related sub-queries and requires only a check of the dependency tree.

7. Conclusion and Future Work

In this paper an architecture for interactive query refinement for a sensor-based information fusion system has been described.

The architecture is based on the ΣQL query language and includes an elaborate ontology and its knowledge-base.

The ontological knowledge system supports the query processing by selecting the most appropriate sensors and sensor data analysis algorithms in a sensor data independent way, which means that the users are not involved in the sensor and algorithm selection process.

The ontology also include means for determination of the most relevant objects with respect to the applied queries, i.e. the existing object patterns that actually are used for the recognition of these objects.

The query process is basically controlled by the dependency tree that controls the different steps in the query execution.

The Sigma System also includes a reasoner called Pequliar that is referred to as a post query language reasoner. The purpose of this Reasoner is to automatically or semi automatically carry out an iterative information fusion step by means of rules that are either automatically launched by the Reasoner or defined by the user. In the latter case the Reasoner can learn the new rule and use it in later queries.

A rule corresponds to a refined query described in terms of a generic pattern. Through the refined queries a better support for the higher levels of information fusion can be achieved in particular for situation awareness and impact analysis.

References:
[1] D.L. McGuinness "Ontologies for Information Fusion" , Proceedings of the International Conference on Information Fusion 2003 (Fusion'03), Cairns, Australia, July 8-11, pp 650-657

[2] S.K. Chang, "The Sentient Map", Journal of Visual Languages and Computing, Academic Press, Vol. 11, 2000, 455-474.

[3] S.K. Chang, E. Jungert, "Iterative information fusion using a reasoner for objects with uninformative belief values" to appear in Seventh International Conference on Information Fusion, Stockholm, Sweden, 2004.

[4] S.K. Chang, E. Jungert and G. Costagliola, "Multi-sensor Information Fusion by Query Refinement", Proc. of 5th Int'l Conference on Visual Information Systems, Hsin Chu, Taiwan, March 2002, pp. 1-11.

[5] S.K. Chang, E. Jungert and G. Costagliola, "Queryng Distributed Multimedia Databases and Data Sources for Sensor Data Fusion", to appear in IEEE Transactions on Multimedia, 2004.

[6] Handbook of Multisensor Data Fusion, D.L. Hall & J. Llinas (Eds.) CRC Press, New York, 2001.

[7] T. Horney, "Design of an ontological knowledge structure for a query language for multiple data source", FOI-R-0498-SE, May, 2002, ISSN 1650.1942, Scientific Report.

[8] T. Horney, E. Jungert, M. Folkesson, "An Ontology Controlled Data Fusion Process for Query Language", Proceedings of the International Conference on Information Fusion 2003 (Fusion'03), Cairns, Australia, July 8-11.

[9] J. Nielsen, "Usability Engineering", Morgan Kaufman, New York, 2001

Using Relevance Feedback in Multimedia Databases

Chotirat Ann Ratanamahatana   Eamonn Keogh

Department of Computer Science and Engineering, University of California, Riverside, CA 92521

{*ratana, eamonn*)@cs.ucr.edu

ABSTRACT

Much of the world's data is in the form of time series, and many other types of data, such as video, image, and handwriting, can easily be transformed into time series. This fact has fueled enormous interest in time series retrieval in the database and data mining community. We argue, however, that much of this work's narrow focus on efficiency and scalability has come at the cost of usability and effectiveness. In this work, we introduce a general framework that learns a distance measure with arbitrary constraints on the warping path of the Dynamic Time Warping calculation. We demonstrate utility of our approach on both classification and query retrieval tasks for time series and other types of multimedia data including images, videos, and handwriting archives.

1. INTRODUCTION

Much of the world's data is in the form of time series, and many other types of data, such as video, image, and handwriting, can also be trivially transformed into time series. This fact has fuelled enormous interest in time series retrieval in the database and data mining community. We argue, however, that much of this work's narrow focus on efficiency and scalability has come at the cost of usability and effectiveness. For example, the lion's share of previous work has utilized the Euclidean distance metric, presumably because it is very amenable to indexing [2][5][7]. However, there is increasing evidence that the Euclidean metric's sensitivity to small differences in the time axis makes it unsuitable for most real world problems [1][4][6][11][22] [26]. This fact appears to have gone almost unnoticed because, unlike their counterparts in information retrieval, many researchers in the database and data mining community evaluate algorithms without considering precision/recall or accuracy [13].

In this work, we introduce a new distance measure and empirically show its utility with thorough experiments measuring the precision/recall and accuracy. While we will demonstrate that our measure is the best in literature, it has a potential weakness; It requires some training or human intervention to achieve its finer results. However, to achieve this end, we will show that the classic information retrieval technique of relevance feedback can be used.

The rest of the paper is organized as follows. The remainder of this section will familiarize readers with the time series and its tight connection with other types of multimedia data. Section 2 gives a review of Dynamic Time Warping (DTW), and related work. In Section 3, we introduce our approach which is based on learning domain specific (and possibly class specific) constraints on the popular DTW distance measure using a representation we call the *R-K Band*. Section 4 demonstrates how this framework is used for relevance feedback then reports the empirical evaluation on three real-world datasets. Lastly, Section 5 gives conclusions and direction for future work.

1.1 The Ubiquity of Time Series Data

In this section, we wish to expand the readers' appreciation for the ubiquity of time series data. Rather than simply list the traditional application domains, i.e. stock market data, electrocardiograms, weather data, etc., we will consider some less obvious applications that can benefit from efficient and effective retrieval.

Video Retrieval: Video retrieval is one of the most important issues in multimedia database management systems. Generally, research on content-based video retrieval represents the content of the video as a set of frames, leaving out the temporal features of frames in the shot. However, for some domains, including motion capture editing, gait analysis, and video surveillance, it may be fruitful to extract time series from the video, and index *just* the time series (with pointers back to the original video). Figure 1 shows an example of a video sequence that is transformed into a time series. There are several reasons why using the time series representation may be better than working with the original data. One obvious point is the massive reduction in dimensionality, which enhances the ease of storage, transmission, analysis, and indexing. In addition, it is much easier to make the time series representation invariant to distortions in the data, such as time scaling and time warping.

Figure 1 Stills from a video sequence; the right hand is tracked, and converted into a time series

Image Retrieval: Image Retrieval has become increasingly crucial in our information-based community. Large and distributed collections of scientific, artistic, technical, and commercial images have become more prevalent, thus requiring more sophisticated and precise methods for users to perform similarity or semantic based queries. For some specialized domains, it can be useful to convert the images into "pseudo time series". For example, consider Figure 2 below. Here, we have converted an image of a leaf into a time series by measuring the local angle of a trace of its perimeter. The utility of such a transform is similar to that for

video retrieval.

Figure 2. Many image indexing/classification tasks can be solved more effectively and efficiently after converting the image into a "time series"

Handwriting Retrieval: While the recognition of *online* handwriting [10] may be largely regarded as a solved problem, the problem of transcribing and indexing existing historical archives remains a challenge. The usefulness of such ability is obvious. For even such a major historical figure as Isaac Newton, there exists a body of unpublished, handwritten work exceeding one million words. For other historical figures, there are even larger collections of handwritten text. Such collections are potential goldmines for researchers/biographers.

Figure 3.A shows an example of text written by George Washington, which is all but illegible to modern readers with little experience with cursive writing.

Figure 3.A) An example of handwritten text by George Washington. B) A zoom-in on the word "Alexandria", after being processed to remove slant. C) Many techniques exist to convert 2-D handwriting into a time series; in this case, the projection profile is used (Fig. created by R. Manmatha)

Many off-line handwritten document image-processing algorithms have recently been proposed in the interest of word recognition and indexing [12]. While handwriting is not a time series, there exist several techniques to convert handwriting to (one or more) time series; many of these transformations were pioneered by Manmatha and students [17].

1.2 Existing Work on Time Series Retrieval

The explosion of interest in time series indexing in the last decade has been extraordinary, with well over a thousand papers devoted to the subject [13]. However, the vast majority of the work has focused on the Euclidean distance; recent work has demonstrated that this similarity model generally does not work well for many real-world problems since even very similar time series often demonstrate some variability in the time axis. The problem of distortion in the time axis can be addressed by Dynamic Time Warping (DTW), a distance measure that has long been known to the speech processing community [15][16][20][23]. This method allows for non-linear alignments between the two time series to accommodate sequences that are similar but out of phase, as shown in Figure 4.C.

Our approach takes this recent work on DTW as its starting point. In particular, DTW is currently viewed as a "one-size-fits-all" algorithm, which is applied to diverse domains in a black box fashion. We note, however, that we may be able to fine-tune the algorithm, for a particular domain and even a particular query, by selectively limiting the amount of warping we allow along various parts of the query. For example, in Figure 4, we can see that the first 1/5 and the last 1/6 of the time series do not require warping. This happens to be true for all instances in this particular domain. As we will demonstrate, by selectively limiting the amount of warping allowed, we can actually improve the accuracy of DTW, and as an important side effect, we can drastically improve the indexing performance. Before formally introducing our technique, we must review the basic DTW algorithm in some detail.

2. BACKGROUND

Suppose we have two time series, a query sequence $Q = q_1, q_2, ..., q_i, ..., q_n$ of length n and a candidate sequence $C = c_1, c_2, ..., c_j, ..., c_m$ of length m. The DTW algorithm finds the optimal time alignment between these two given time series. To align two sequences using DTW, an *n*-by-*m* matrix is constructed where the (i^{th}, j^{th}) element of the matrix corresponds to the cumulative squared distance, $d(q_i, c_j) = (q_i - c_j)^2$, the alignment between points q_i and c_j. To find the best match between these two sequences, the path through the matrix that minimizes the total cumulative distance between them is discovered. This is illustrated in Figure 4. A warping path, W, is a contiguous set of matrix elements that characterizes a mapping between Q and C. The kth element of W is defined as $w_k = (i,j)_k$. By definition, the optimal path W_o is the path that minimizes the warping cost:

$$DTW(Q,C) = \min \left\{ \sqrt{\sum_{k=1}^{K} w_k} \right\} \quad (1)$$

This path can be found using dynamic programming to evaluate the following recurrence which defines the cumulative distance $\gamma(i,j)$ as the distance $d(i,j)$ found in the current cell and the minimum of the cumulative distances of the adjacent elements:

$$\gamma(i,j) = d(q_i, c_j) + \min\{ \gamma(i-1,j-1), \gamma(i-1,j), \gamma(i,j-1) \} \quad (2)$$

Although the dynamic programming algorithm reduces the (potentially exponential) number of paths which we must consider to a "mere" $m * n$, this may still be prohibitively large for many problems. The following well known constraints further reduce the number of warping paths that must be considered:- *Boundary Conditions, Continuity Condition, Monotonic condition, and Adjustment Window Condition.*

Figure 4. A) Two similar sequences Q and C, but out of phase. B) To align the sequences, we construct a warping matrix, and search for the optimal warping path, shown with solid squares. Note that the "corners" of the matrix (shown in dark gray) are excluded from the search path as part of an Adjustment Window condition. C) The resulting alignment

By applying the above conditions, we can restrict the moves that can be made from any point in the path and so restrict the number of paths that need to be considered. Figure 4.B illustrates a particular example of the last condition with the Sakoe-Chiba Band. Since a good path is unlikely to wander very far from the diagonal, the distance that the path is allowed to wander is within the window of size *r,* above and to the right of the diagonal. As we will see in Section 3, it is this type of constraint that we will exploit to improve DTW.

2.1 Related Work
There has been relatively little work on relevance feedback for both time series and multimedia retrieval. However, relevance feedback in text-mining community has been the subject of much research since the 1970's [3][18][19] and still is an active area of research. It is only in recent years that the researchers started to expand relevance feedback into time series [14], image [25], and multimedia retrieval domains.

Before addressing the relevance feedback system with DTW, we first must introduce our representation, the *R-K Band*, which will be used for the DTW distance measure in the classification task and relevance feedback.

3. RATANAMAHATANA-KEOGH BAND
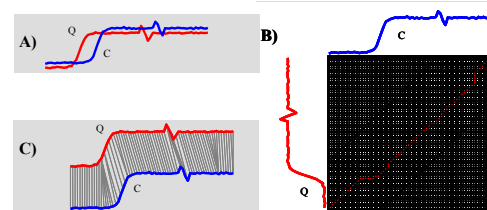The 'Adjustment Window Condition' discussed in Section 2 has been almost universally applied to DTW, primarily to prevent unreasonable warping and to speed up its computation. However, surprisingly little research has looked at discovering the best *shape* and *size* of the window. Most practitioners simply use one of the well-known bands, e.g. Sakoe-Chiba Band [20] or Itakura Parallelogram [9], proposed in the context of speech recognition several decades ago. In addition, there is a widespread but unwarranted belief that having wider bands improves accuracy, and having narrower bands decreases accuracy. The use of smaller-size band is seen as a compromise made to make the algorithm tractable. This belief has been proved to be false by our extensive experiments on wide variety of datasets; surprisingly, the accuracies often peak at smaller-size window, and degrade or become stable for wider window sizes. The motivation for our work has sparked from this discovery; we find that in general, the effect of the window size on accuracy is very substantial, and is strongly domain dependent. And if the *width* of the band can greatly affect accuracy, then the *shape* of the band could also have similarly large effects. Our ideal solution would be to find an optimal band (both shape and size) for a given problem that will potentially increase the accuracy. In the next section, we first introduce our representation, the *R-K Band,* which allows user to specify arbitrary shaped constraints.

3.1 A General Model of Global Constraints
We can represent any warping window as a vector *R*:

$$R_i = d \qquad 0 \le d \le m, \ 1 \le i \le m \qquad (3)$$

where R_i is the height above the diagonal in the *y* direction, as well as the width to the right of the diagonal in the *x* direction. Note that $|R| = m$, and the above definition forces *R* to be symmetric, i.e. the constraint above the diagonal is the mirror image of the one below the diagonal.

To represent a Sakoe-Chiba Band of overall width of 11 (width 5 strictly above and to the right of the diagonal) with

the definition:

$$R_i = \begin{cases} 5 & 1 \le i \le m-5 \\ m-i & m-5 < i \le m \end{cases} \qquad (4)$$

or an Itakura Parallelogram with the definition:

$$R_i = \begin{cases} \left\lfloor \frac{2}{3} i \right\rfloor & 1 \le i \le \left\lfloor \frac{3}{8} m \right\rfloor \\ \left\lfloor \frac{3}{8} m \right\rfloor - \left\lfloor \frac{2}{5} i \right\rfloor & \left\lfloor \frac{3}{8} m \right\rfloor < i \le m \end{cases} \qquad (5)$$

The classic Euclidean distance can also be defined in terms of $R_i = 0; \ 1 \le i \le m;$ only the diagonal path is allowed. More generally, we can define any arbitrary constraint with a suitable vector *R*. Figure 5 illustrates some examples of *R-K Bands*.

Figure 5. We can use *R* to create arbitrary global constraints. A) Note that the width of the band may increase or decrease. We can also use *R* to specify all existing global constraints, e.g. Sakoe-Chiba Band B) and Itakura Parallelogram C)

An interesting and useful property of our representation is that it also includes the ubiquitous Euclidean distance and classic DTW as special cases. We also can exploit the *R-K Bands* for both classification and indexing (query retrieval) problems, depending on the task at hand. In particular,

- for classification, we can use a different *R-K Band* for each class; we denote the band learned for c^{th} class as the *R-K$_c$ Band.*
- for indexing, we can use *one R-K Band* that maximizes the trade off between efficiency and precision/recall.

Having introduced an *R-K Band*, we can easily represent any arbitrary warping windows. However, we are left with the question of how to *discover* the optimal *R-K Band* for the task at hand. In some cases, it maybe is possible to manually construct the bands, based on domain knowledge. For example, a cardiologist may know from experience that the Romano-Ward syndrome may manifest itself with high variability in the length of one part of the heartbeat (the QT-wave), but little variability in the other section of a heartbeat (the UP-wave)[24]. We could explicitly attempt to encode this insight into an *R-K Band* for retrieving instances of the disease, allowing R_i to be large where variability is expected.

Unfortunately, our preliminary attempts to manually construct *R-K Bands* met with limited success, even for simple toy problems. Furthermore, since the number of possible *R-K Bands* is exponential, exhaustive search over all possibilities is clearly not an option. In the following sections, however, we will show how we can *learn* the high-quality bands automatically from the data.

3.2 Learning Multiple *R-K$_c$ Bands* for Classification
While it is generally not possible to handcraft accurate *R-K Bands*, it *is* possible to pose the problem as a search problem, and utilize classic search techniques from the artificial

intelligence community. Using generic heuristic search techniques, we can perform both forward and backward searches. The forward search starts with the initial Sakoe-Chiba band (uniform) of width 0 (Euclidean), and the backward search starts from the uniform band of the maximum width m, above and to the right of the diagonal.

Figure 6. An illustration of our forward search algorithm

Due to space limitations, we will only give a simple intuition behind this approach as illustrated in Figure 6. For a forward search, we start off with Euclidean Bands, one for each class. Then it takes turns, one class at a time, trying to expand or increment the whole section of the band before re-evaluating its overall accuracy for that particular band. If an improvement is made, we keep on expanding that section; otherwise, we undo the expansion, split that section in half, and then recursively expand each portion individually before another re-evaluation. Backward search is very similar; except that we start off with a very wider band then try to tighten the band instead of expanding. Here, we do not include a bi-directional search, a straightforward combination of the forward and backward search; this is omitted in this work for brevity.

The searches are complete when one of the following is true:- No improvement in accuracy can be made; the width of the band reaches m for the forward search and 0 (Euclidean) for the backward search or; each section of the band (after recursively cut the portion in half) reaches some threshold. We set a threshold to be some function of m.

Figure 7. The $R\text{-}K_c$ Bands learned from 6 different species

We can illustrate the utility of our $R\text{-}K_c$ Bands for classification by the following simple experiment. We tested various similarity measures (Euclidean, DTW with 10% warping, DTW with best uniform warping, and DTW with $R\text{-}K_c$ Bands) on the Leaf dataset (dataset details in 4.3.2) and measure their classification error rates. Euclidean is very fast but inaccurate, giving 34.16% error rate. DTW with 10% uniform warping gives a big improvement with only 4.52%. However, the best uniform warping size for this dataset is at 8.6% window size, giving 4.3%. With $R\text{-}K_c$ Bands, we

produce 6 different bands, one for each class shown in Figure 7. Classification using these bands gives us almost a perfect result, a mere 0.9% error rate. These promising results suggest that $R\text{-}K_c$ Bands are very effective in improving accuracy in classification.

3.3   Learning One *R-K Band* for Indexing

In addition to creating $R\text{-}K_c$ Bands for classification, we can learn one single *R-K Band* for indexing or query retrieval. The one-band learning algorithm is very similar to the multiple-band learning in the previous section, except that we only maintain one single band that represents the whole problem and that we measure the precision/recall instead of the accuracy.

We re-illustrate this approach by another simple experiment, measuring precision and recall for indexing. We take 10 examples of *Cylinders* from the Cylinder-Bell-Funnel dataset [6][11] and place them in a database containing another 10,000 *random-walk* sequences that are similar in shape but do not belong in the class. Another 30 examples of *Cylinders* with 470 *random-walk* sequences are used in the *R-K Band* training process. To evaluate our method, another 10 different *Cylinder* examples are used to make 10 iterations of k-nearest neighbor queries to the dataset, using various distance measures (Euclidean, DTW with 10% warping, and DTW with *R-K Band*).

We measure the precision from 1-object (10%) to 10-object (100%) recall levels. The results are shown in Figure 8. It is apparent that utilizing an *R-K Band* in this problem improves both precision and recall by a wide margin, compared to Euclidean and DTW with 10% warping. However, an *R-K Band* needs to be learned from a training data, which may not be practical or available in many circumstances. To resolve this problem, we can build a training data through relevance feedback system, with a little help from the user in identifying the positive and negative examples to the system. We will explain how this works in Section 4, but first we will attempt to develop the readers' intuition as to *why* the *R-K Bands* can produce superior performance

Figure 8. The Precision-Recall curves from 10% to 100% recall for various distance measures: Euclidean, DTW with 10% window size, and our proposed method – *R-K Band* that gives *perfect* precision for all recall levels

3.4   Intuition behind *R-K Bands* Learning

After seeing some examples of our *R-K* Band's utility, we would like to further convince readers by giving an intuition why *R-K Band* improve accuracy. Consider the following problem of face classification based on the head profile. We took a number of photos (20-35) of each individual with different expression on the face, e.g. talking, smiling, frowning, etc. We then use the similar method (see section 1.1) to extract each of the head profile into time series as shown in Figure 9.

We will show by experiment how *R-K Bands* may play an

important role in this problem. First, we consider a 2-class problem: a dataset that contains only the collection of profiles from 2 different individuals that look rather different (1 male, 1 female). The *R-K_c Bands* learned from our framework discover the bands both of size zero, the Euclidean distance measure, with 2% error rate. The result suggests that these two individuals are very distinguishable, i.e. the set of time series within each class is much different from another, just by looking at their Euclidean distances. Hence, no warping is necessary; in fact, too much warping could potentially hurt the accuracy because one person could be forced to match with another.

Figure 9. Starting from the neck area, the head profile is converted into a "pseudo time series"

We then extend our experiment by adding 2 more male individuals into our problem (112 instances total). The corresponding *R-K Bands* are learned which give very low error rate of 1.8% (vs. 6.25% for Euclidean).

4. RELEVANCE FEEDBACK

In text-mining community, relevance feedback is well known to be effective method to improve the query performance [3][18][19][21]. However, there has been relatively little research in non-text domains, such as images or multimedia data. In section 1.1, we have introduced time series as an alternative in representing certain types of multimedia data, including special cases of images and video. We will explain in this section how we utilize and incorporate the technique into the relevance feedback system using our proposed framework, *R-K Band*.

4.1 Query Refinement

Relevance feedback methods attempt to improve performance for a particular informational need by refining the query, based on the user's reaction to the initial retrieved documents or objects. In text retrieval in particular, the user's ranking of the document allows reweighing the query terms.

Working with time series retrieval is rather similar to the text retrieval; a user can draw or provide an example of a query and retrieve the set of best matches' retrieval of images/videos/time series. Once the user ranks each of the results, a query refinement is performed such that a *better-quality* query is produced for the next retrieval round. For real time series retrieval (i.e. electrocardiograms or stock market data), the querying interface can show the time series directly. For transformed data of images or video, the underlying time series representation is hidden from the user, and the user sees only thumbnails of the actual images or video snippets. In our system, the user is asked to rank each result in a 4-point scale as shown based on relevance to their informational needs. These rankings are converted into appropriate weights which are used in the query refinement process (averaging the weighted positive results with the current query).

Figure 10. An example of averaging 4 sequences with DTW. Pairs of sequences are hierarchically combined by DTW with their weights until the final averaged sequence is obtained

However, averaging a collection of time series that are not perfectly time-aligned is non-trivial and DTW is needed [8]. Each pair of time series are averaged according to their weights and warping alignment. The results from each pairs are hierarchically combined

Figure 10 illustrates this averaging process using equal weights for all sequences; in practice, the weights may all be different. In the next section, we will show how the relevance feedback system can benefit from our proposed *R-K Band* framework.

4.2 R-K Band in Relevance Feedback

We will empirically demonstrate that our proposed *R-K* Band combined with the query refinement can improve precision and recall of retrieval. Table 1 shows our relevance feedback algorithm.

Table 1: *R-K Band* learning with Relevance Feedback

```
Algorithm RelFeedback(initial_query)
1. Repeat until all rankings are positive.
2.    Show the 10 best matches to the current query to
      the user.
3.    Let the user rank how relevant each result is.
4.    According to the ranking, accumulatively build
      the training set; positive result → class 1,
      negative result → class 2.
5.    Learn a single envelope (R-K Band) that
      represents the given training data.
6.    Generate a new query, by averaging (with DTW)
      the positive results with the current query
      according to their weights (rankings).
7. end;
```

In the first iteration, given a query, the system uses the initial *R-K Band* (the special case of Euclidean distance) to retrieve the 10 nearest neighbors, and then shows them to the user (line 1). When the user finishes their ranking, the positive and negative responses are noted and collected as a training data (lines 3-4). The algorithm uses this training data to learn an *R-K Band* that best represents the positive objects in the training set while being able to correctly differentiate the positive from the negative instances (line 5). The training data will be accumulated during each round, developing a larger training set, thus producing progressively finer results. The process is complete when only positive feedbacks are given to the system or the user abandons the task.

In our experiments, we consider 3 multimedia datasets to be tested using the relevance feedback technique

4.3 Datasets

To evaluate our framework, we measure the precision and recall for each round of the relevance feedback retrieval. Since we only return the 10 best matches to the user and we would like to measure the precision at all recall levels, we purposely leave only 10 relevant objects of interest in all the databases.

4.3.1 Gun Problem

This dataset comes from the video surveillance domain (see Figure 1). The dataset has two classes, 100 examples each:

- **Gun-Draw:** The actors have their hands by their sides. They draw a replicate gun from a hip-mounted holster, point it at a target for approximately one second, and then return the gun to the holster, and their hands to their sides.
- **Point:** The actors have their hands by their sides. They point with their index fingers to a target for approximately one second, and then return their hands to their sides.

For both classes, the centroid of the right hand is tracked both in X- and Y-axes; however, in this experiment, we only consider the X-axis for simplicity. The dataset contains 200 instances, 100 for each class. Each instance has the same length of 150 data points. For the relevance feedback purpose, we only leave 10 Gun-Draw examples in the Point database, and randomly pick another example for an initial query.

4.3.2 Leaf Dataset
This dataset contains a collection of 6 different species of leaf images, including 2 genera of plant, i.e. oak and maple. Maple has 4 different species, and Oak has 2, with 442 instances in total (original images are available at http://web.engr.oregonstate.edu/~tgd/leaves/dataset/herbarium). Each instance is linearly interpolated to have the same length of 150 data points. In our experiment, we choose Circinatum maple as the specie of interest, i.e. only 10 images of such specie are left in the database, and we randomly selected another separate image as our initial query.

4.3.3 Handwritten Word Spotting Dataset
This is a subset of the WordSpotting Project dataset. In the full dataset, there are 2,381 words with four features that represent each word image's profiles or the background/ink transitions. For simplicity, we pick the "background/ink transitions" (feature 4) and use the word "*would*" which occurs in the dataset 11 times for our query. Hence, one is removed to be used as our initial query.

4.4 Experimental Results
We measure the performance of our relevance feedback system with the precision-recall plot from each round of iteration. Figure 11 below shows the precision-recall curves of the three datasets for the first five iterations of relevance feedback. Our experiments illustrates that each iteration gives significant improvement in both precision and recall.

Figure 11. The precision-recall plots for the Gun, Leaf and Word spotting datasets with 5 iterations of relevance feedback

5. DISCUSSION AND CONCLUSIONS
In this work, we have introduced a framework for both classification and time series retrieval. The *R-K Band* allows for any arbitrary shape of the warping window in DTW calculation. With our extensive evaluation, we have shown that our framework incorporated into relevance feedback can reduce the error rate in classification, and improve the

precision at all recall levels in video and image retrieval.

6. REFERENCES
[1] Aach, J. and Church, G. (2001). Aligning gene expression time series with time warping algorithms. Bioinformatics. Volume 17, pp. 495-508.

[2] Agrawal, R., Lin, K. I., Sawhney, H. S., & Shim, K. (1995). Fast similarity search in the presence of noise, scaling, and translation in times-series databases. VLDB, pp. 490-501.

[3] Attar, R. & Fraenkel, A.S. (1977). Local Feedback in Full-Text Retrieval Systems. JACM '77, Vol. 24, No. 3, July 1977, pp. 397-417.

[4] Bar-Joseph. Z., Gerber, G., Gifford, D., Jaakkola T & Simon. I. (2002). A new approach to analyzing gene expression time series data. In the 6th Annual Intl Conf. on Research in Comp Molecular Bio. pp. 39-48.

[5] Chan, K.P., Fu, A & Yu, C. (2003). Haar wavelets for efficient similarity search of time-series: with and without time warping. IEEE TKDE, Vol.15, No.3. pp. 686-705.

[6] Diez, J. J. R. & Gonzalez, C. A. (2000). Applying boosting to similarity literals for time series Classification. Multiple Classifier Systems, 1st Inter' Workshop. pp. 210-219.

[7] Faloutsos, C., Ranganathan, M., & Manolopoulos, Y. (1994). Fast subsequence matching in time-series databases. In Proc. ACM SIGMOD Conf., Minneapolis. pp. 419-429.

[8] Gupta, L., Molfese, D., Tammana, R., and Simos, P. (1996). Nonlinear Alignment and Averaging for Estimating the Evoked Potential. IEE Transactions on Biomedical Engineering, Vol. 43, No.4, April 1996.

[9] Itakura, F. (1975). Minimum prediction residual principle applied to speech recognition. IEEE Trans. ASSP-23, pp. 52-72

[10] Jain, A.K. & Namboodiri, A.M. (2003). Indexing and Retrieval of On-line Handwritten Documents. IEEE 7th ICDAR, Vol. II, pp. 655-659.

[11] Kadous, M. W. (1999). Learning comprehensible descriptions of multivariate time series. In Proc. of the 16th ICML pp. 454-463.

[12] Kavallieratou, E., Dromazou, N., Fakotakis, N., & Kokkinakis, G. (2003). An Integrated System for Handwritten Document Image Processing. Int'l J. of Pattern Recognition and AI, 17(4)pp. 617-636.

[13] Keogh, E. and Kasetty, S. (2002). On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration. In the 8th ACM SIGKDD. pp. 102-111.

[14] Keogh, E. & Pazzani,M (1999). Relevance feedback retrieval of time series data. In Proceedings of the 22nd ACM-SIGIR. pp 183-190.

[15] Kruskall, J. B. & Liberman, M. (1983). The symmetric time warping algorithm: From continuous to discrete. In Time Warps, String Edits and Macromolecules. Addison-Wesley.

[16] Myers, C., Rabiner, L & Roseneberg, A. (1980). Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. IEEE Trans. ASSP-28, pp. 623-635.

[17] Rath, T. & Manmatha, R. (2003). Word image matching using dynamic time warping. CVPR, Vol. II, pp. 521-527.

[18] Rickman, J.T. (1972). Design Considerations for a Boolean Search System with Automatic Relevance Feedback Processing. SIGIR2: August 1972. Proceedings of the ACM conference-Vol. 1, pp. 478-481.

[19] Rocchio, J.J. (1971). Relevance Feedback in Information Retrieval. The SMART Retrieval System: Experiments in Automatic Document Processing. pp. 313-323.

[20] Sakoe, H. & Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. IEEE ASSP-26. pp. 43-49.

[21] Salton, G., & Buckley, C. (1990). Improving retrieval performance by relevance feedback. JASIS41(4), pp.288-297.

[22] Schmill, M., Oates, T. & Cohen, P. (1999). Learned models for continuous planning. In 7th International Workshop on AI and Statistics.

[23] Tappert, C. & Das, S. (1978). Memory and time improvements in a dynamic programming algorithm for matching speech patterns. IEEE Trans. ASSP-26, pp. 583-586.

[24] Viskin, S. (2000). Cardiac Pacing in the Long QT Syndrome: review of available data and practical recommendations. Journal Cardiovasc Electrophysiol 11:593-600.

[25] Wu, P. & Manjunath, B.S. (2001). Adaptive Nearest Neighbor Search for Relevance Feedback in Large Image Databases. ACMMM, pp89-97.

[26] Yi, B, K. Jagadish, H & Faloutsos (1998). Efficient retrieval of similar time sequences under time warping. ICDE, pp. 23-27.

Incremental Benchmark Development and Administration

Michael Grubinger

School of Computer Science and Mathematics

Faculty of Engineering, Science & Technology

Victoria University of Technology

Melbourne City, PO Box 14428

Victoria 8001, Australia

{michael.grubinger@research.vu.edu.au}

Clement Leung

School of Computer Science and Mathematics

Faculty of Engineering, Science & Technology

Victoria University of Technology

Melbourne City, PO Box 14428

Victoria 8001, Australia

{clement@matilda.vu.edu.au}

ABSTRACT

A benchmark for performance calibration in visual information search has been developed and published by the International Association of Pattern Recognition (IAPR). As such a benchmark suite grows and improves, a need arises for a benchmark administration system to be developed. This paper presents the second version of such a standardised benchmark suite and reports on how it was extended, refined and improved from the first version by using a new benchmark administration system. Image, query and indexing rules are defined; the benchmark administration system and its specification and architecture are presented. Moreover, the underlying database system and corresponding linguistic indexing aspects are described.

Keywords – image benchmark, complex image contents, structured annotation, benchmark administration system

1. INTRODUCTION AND MOTIVATION

Searching visual information on the Internet presents a substantial challenge, even though text-based searching is reasonably well-developed and mature. At present, searching images on the Internet tends to result either in one of two extreme possibilities.

- **Too many images being returned.** This tends to be the case for relatively simple queries focusing on particular single objects (e.g. cars, violins etc.) These searches are obtained by basically using keywords applied to captions or other related textual materials (possibly including speech and narrative) from which the contents are inferred or extracted. Very often, hundreds of thousands of images may be returned. Some may meet the required search criteria, some would invariably fail to do so. If an inaccurate or "misleading" caption is used, then very often the wrong images are returned.

- **An empty set being returned.** This tends to be the case for more complex queries. For example, when a query involves several objects, and they are being

considered as conjunctions, then there may not be any correct images being returned. Most often, this failure is caused by the inability of the caption to incorporate all the objects rather than that there is no image that actually meeting the required search criteria. This highlights once more the inadequacy of keywords. Thus, there are images meeting the required query specification but they are not retrieved due to the indexing limitations. The incorporation of specific relationship between the given objects is generally reduced to keyword search based on the objects with the relationship between them completely ignored by the search engine.

As a consequence, a need for more research into the mechanisms of effective visual information search exists and an increasing number of content based image retrieval systems has become available [1]. Unfortunately, there is a tendency for many systems to use different data and queries to emphasise the advantages of the algorithms proposed, with the result that a system may seem to work much better than it really is. Objective comparison should be carried out on a common basis with the same image data, queries and performance measures, and in doing so the better techniques can be detected and research progress can be achieved.

In information retrieval, the measures of *recall* and *precision* are used, and these are generally related to the use of a finite collection of items. Measuring recall for Internet image search is generally not relevant as one is dealing with an "open" set of images rather than a finite closed set. Precision, however, is more meaningful. This is represented in the precision-recall graph in Figure 1, where the y-axis gives the precision and the x-axis represents recall.

Fig 1: Recall and Precision

All this points to the need for a more standardised description of images that is able to incorporate more semantics with greater flexibility for representing the objects, relationships together with the appropriate adjectives or modifiers. This will allow the results to gravitate towards the left end of the graph in Figure 1 when higher precision may be obtained. Such precision may be practically attainable if a standardised markup convention based on XML is used similar to the use of HTML for text-oriented documents. It is likely that even with a single sentence, the retrieval precision may be improved dramatically on Internet image search. Our benchmark administration system will facilitate such an extension.

The next section provides an overview of related works in benchmarking and how the new, second version of the IAPR benchmark relates to them. Section 3 summarizes the rules for images, queries and indexing. Sections 4 presents the new benchmark administration system and its specification, describes the underlying database architecture and associated linguistic structures.

2. RELATIONSHIP WITH OTHER WORKS

Benchmarking for visual information systems is a relatively young research area, in comparison to other standardised evaluation systems in similar areas such as text retrieval that have been in existence for many years. One of the first articles [2] was written in 1997, and the issue was also raised in [3] and [4]. Further articles like [5], [6], [7] or [8] were written, all of them pointing to the need of a standardised evaluation system for visual information systems.

Thereupon, a couple of benchmark systems have been implemented. In [9], provides useful links to a number of image benchmarks and image collections are provided. The problem is that most of these benchmark systems furnish images with either primitive image contents [10], [11] or dwell excessively on self similarities with hardly any relationships [5] [11]. In [12], a set of recommendations and specifications for a benchmark system based on images with complex image contents is presented, which seems to offer the most promising approach.

Fig 2: Position of IAPR benchmark

While most of the other benchmarks concentrate on primitive image contents and purely automatic image

retrieval (as indicated in Figure 2), [13] presents the outcome of building a benchmark based on complex image contents (first version) in following the IAPR specifications in [12]. Further information on the difference between primitive and complex image contents can be found in [14]. The first version of the IAPR benchmark has been presented in [15] and has now been extended to 5000 images.

This paper describes how the second version of the IAPR benchmark was extended, refined and improved from the first version with the help of a benchmark administration system.

3. BENCHMARK DEVELOPMENT

Benchmarks are never static and will need to be updated and evolve as the need for visual information search changes and develops. A particular useful way of viewing our benchmark is to draw an analogy between it and the TPC benchmarks [16].

The TPC process originated with the batch-oriented TP1, but evolved to include TPC-A, TPC-B, TPC-C, TPC-H, TPC-R, and TPC-W, which represent successive developments along the same theme. Such incremental development and refinement will appear to be unavoidable for our benchmark. In addition, the TPC benchmarks cover a range of processing paradigms; e.g. TP1 is geared to batch processing, TPC-A/C are geared to OLTP (Online Transaction Processing), TPC-D/H/R are geared to decision support OLAP (Online Analytic Processing), and the TPC-W is geared to Web-based order processing. Hence, we expect the present completed benchmark will need to cater for a variety of applications paradigms. Indeed, other established benchmarks (e.g. SPEC) also exhibit similar patterns. Thus, in adopting a long term view, it will be necessary to develop a flexible benchmark administration system to facilitate the ongoing refinement and development of the present benchmark.

Based on the criteria given in [12], a first version of a benchmark (consisting of 1000 images, 25 queries, and a collection of performance measures) was set up and published in 2003 [14]. In order to ensure a high level of quality of this first version of the benchmark, the following rules were strictly observed.

All the images in the benchmark

- are multi-object JPG images (photos)
- contain a good cross-section of a number of objects n,
- have at least two objects (at least one of the objects is an active object = subject),
- show a varying number of attributes and relations. In general, the number of relationships r will be

$$r \leq \binom{n}{2}$$

- provide a good cross-section of repetitiveness within their objects (repeated in different cardinality ranges with the same or different attributes)

- are indexed according to MPEG-7 with adequate precision, cardinality, terminology, complexity and use of sub-elements.

If images have the same subject (in other words, they are in one logical category), the subjects have

- same relations to different objects,
- different relations to the same objects,
- same relations with different modifiers or even
- different relations to different objects.

A good cross-section of representative queries with

- varying sizes of the target set,
- varying cardinality,
- different generalization levels and
- adequate complexity

and a set of performance measures is provided. A more detailed description of all these rules (including scope of the image contents and examples) can be found in [13].

As long as the size of such a benchmark is not too big, it can be administered by hand. But once it reaches a substantial number of images, it is very hard to keep the overview of all the objects, verbs, and attributes used. How would one know if a newly added image is appropriate for the benchmark, has the right number of objects, the perfect generalization level, the ideal complexity of relationships, the optimal amount of attributes, complexity, repetitiveness of the objects and varying settings in terms of location, time, purpose or manners? This new version of the benchmark contains 5000 images. This extension would not have been possible without the use of a benchmark administration system.

4. BENCHMARK ADMINISTRATION

To facilitate incremental development and ongoing maintenance of the benchmark, a benchmark administration system using PHP and mySQL was built.

4.1 Specification and Implementation

The main features of the new benchmark administration system include the following:

- It provides functions for adding, deleting and editing images (Figure 3).

Fig 3: Image Administration Page

- It facilitates the indexing tasks, providing the necessary functions for efficient and accurate indexing (see Figure 4) that would allow the generation of XML output according to the description schemes defined in MPEG-7 [17].

Fig 4: Image Annotation Page

- The benchmark administration system provides statistical tools to guarantee that all the image selection rules (stated in section 3) are observed. This allows further refinement and improvement of the benchmark, because all the images that do not fulfil these requirements can easily be detected and deleted. Moreover, this will also help to extend the benchmark correctly, because with every new image that is inserted, it can be checked whether the image is appropriate for the benchmark or not (Figure 5):

Fig 5: Administration System Statistics

- The query creator will facilitate the search for representative queries for the benchmark. A query interface will check if the created query, the result set and the performance measures are appropriate to the benchmark (see query selection rules in [13]). If so, the query can be added to the query database (or replace an obsolete query).

- The XML generator will write the grammatical information that is stored in the database to an XML file.

4.2. Database Architecture

In accordance with the benchmark requirements, benchmark rules and the specification, the following is the

database system that supports the operation of the benchmark administration system (see Figure 6).

Fig 6: ER-Diagram of the underlying database system

4.3 Table Structure and Contents

The most important tables of the benchmark administration database are the following:

4.3.1 Images

For every image, the filename, size, resolution, width and height, the date when the image was taken and copyright information is stored. This, of course, is metadata and is mainly stored for administrative purposes

4.3.2 Annotations

The annotations table stores information that is common for one FreetextAnnotation sentence of an image (corresponding XML Tag: `<FreetextAnnotation>`). Each image must have at least one annotation, otherwise it is not qualified for this benchmark. There are no limits as regards the number of annotations. IAPR recommends a maximum of three free text sentences for each image [18]. Some examples are given below (Figures 7 and 8):

Fig 7: Boy and girl walking into sea. Boy and girl holding hands. Boy holding ball.

Fig 8: Patient lying in bed. Doctor searching patient. Nurse writing into folder.

4.3.3 Nouns

Each annotation sentence contains at least one subject (XML: `<Who>`) and can contain direct and indirect objects (XML: `<Who>`, `<WhatObject>`). This table stores all the nouns, no matter what usage they have in an annotation. One and the same noun (for example "cat") can be a

- subject ("_cat_ eating mouse")
- a direct object ("dog chasing _cat_")
- or an indirect object ("man giving _cat_ a cookie")

Hence, this noun table would just have one entry "cat". Both _definite_ ("the") and _indefinite articles_ ("a") are omitted. Modifiers (adjectives), usage and count (to meet _cardinality_ requirements as stated in [13]) of each noun are assigned at the specific annotation. Some examples are given below (Figures 9 and 10).

Fig 9: _Austrian_ soccer referee showing _red card_ to _blond Portuguese_ soccer player.

Fig 10: _Red_ crane lifting _long, thick, snowy_ trunks.

The noun table is recursively related to itself, namely to a more general noun to which it belongs. This hierarchy is necessary to meet the requirements of the _generalization_ aspects stated in [13]. Some examples are:

- _Tennis Player_ → _Sportsman_ → _Man_ → _Person_
- _Kangaroo_ → _Marsupial_ → _Animal_
- _Tennis racket_ → _racket_ → _sports equipment_

In the benchmark administration database, the lowest possible noun in the hierarchy is stored, because the higher levels can easily be derived from the more specific term. For example, if the image shows a boy hitting a kangaroo with a tennis racket, the image should be indexed at the lowest level possible:

"_Boy_ hitting _kangaroo_ with _tennis racket_"

and not:

"_Person_ hitting _animal_ with _sports equipment_"

4.3.4 Verbs

Each annotation sentence must contain a verb. (XML: `<WhatAction>`, e.g. Figures 11 and 12):

Fig 11: Soccer player _kicking_ soccer ball.

Fig 12: Athlete _pointing_ at new world record time.

The benchmark just uses verbs that clearly describe the situation in an image (like _running, jumping, painting, hitting_, and so on). Verbs or composite verb groups that need some level of interpretation (e.g. _finding, forgetting, trying to hit, attempting to escape_, etc.) are not used.

Since the annotation describes an action that is happening in the image (at that moment), the grammatically correct form used in the benchmark is the *present continuous tense*, the *auxiliary verb* (to be) is omitted.

4.3.5 Adjectives

This table stores the modifiers for nouns and at the same time modifiers for the subelements (`<When>`, `<Where>`, `<How>`, `<Why>`). Adjectives can belong to different categories (like colour, speed, size, etc.), a classification that will be beneficial for XML generation.

4.3.6 Adverbs

Each verb can have attributes (adverbs) that are stored in this table and assigned to the specific annotation in the relation annotations_verbs_adverbs. Examples are:

"Soccer player *kicking* ball *hard and precisely*."
"Athlete *proudly* pointing at new world record time."

However, *adverbs of degree* are not considered for the benchmark administration database. Adverbs of degree tell us about the intensity or degree of an action, an adjective or another adverb. Examples for adverbs of degree are: *Almost, nearly, quite, just, too, enough, hardly, scarcely, completely, very, extremely*. For example:

"Soccer player kicking ball *very hard*."
"*Extremely* tall boy playing basketball."

The use of these adverbs of degree very much depends on the interpretation of the person who is indexing the image. It is felt that they are too subjective and are therefore not used in the administration database.

4.3.7 Places

Each annotation sentence can contain place adverbs that indicate the location *where* images are taken or which is shown in the image respectively (XML tag: `<Where>`).

Fig 13: Tourists skiing *in the mountains in Austria.*　**Fig 14:** Kids jumping into water *in front of Eiffel Tower.*

Place adverbs can be further described by the use of attributes which are stored in the relation annotation_manners_adjectives, for instance:

"People skiing in *high snowy* mountains in *beautiful* Austria"

Like nouns, places can be put into a hierarchical order, because being in one place implies being in a more general place. Examples are:

on Eiffel Tower → in Paris → in France → in Europe
on chairlift → in skiing region → in mountains

Places are stored with the lowest level possible and without the *prepositions* (that are assigned at the relation).

4.3.8 Times

Each annotation sentence can contain time adverbs that indicate *when* the action happened (XML: `<When>`, examples are shown in Figures 15 and 16).

Fig 15: Man pushing car *in winter.*　**Fig 16:** Cyclist riding bicycle *at night.*

Time adverbs can further be described by the use of attributes, like:

"Man pushing car *in early winter*"
"Cyclist riding bicycle *late at night*"

which are stored in annotations_times_adjectives. Time adverbs can neither be put in a hierarchical order nor can they be categorized. Dates are not to be entered in this table either, they are stored in the image table as the date is the same for each annotation of an image.

4.3.9 Reasons

Each annotation sentence can contain *reason* or *purpose clauses* which describe *why* actions are taken (XML: `<Why>`, see examples in Fig 13 and 14).

Fig 17: Referee showing red card to soccer player *to send him off the field* *because he touched the ball with his hand.*　**Fig 18:** Man scraping snow of car windows *to have a clear sight because it had snowed.*

A *clause* is a string of words that expresses a proposition and typically consists of at least a subject and a verb. The reason clause is therefore not just a single word, and as a consequence, modifiers cannot be used here. If modifiers are required, they should be directly built in the reason clause itself:

The referee showing red card to soccer player *to send him off wet field.*

Further, the XML tag `<Why>` in the benchmark does not only contain *purposes* (as suggested in [20]) but also *reasons* in order to meet the natural language requirements. The word "why" in the English language can ask not only for a purpose:

Why did the referee show the red card to the player?
To send the player off the field.

but also for a reason:

Why did the referee show the red card to the player?
Because he played the ball with his hands.

In many other languages, there are two different words for these two cases: if the XML tags were written in Spanish for example, we could have a `<PorQué>` Tag for *reasons* and a `<ParaQué>` Tag for *purposes*.

4.3.10 Manners

Each annotation sentence can contain *manner adverbs* (XML: `<How>`). Manner adverbs describe *how* actions are performed in an image. Since general adverbs are already used in the modifier table for verbs (adverbs), just *prepositional phrases* that describe how actions were performed are stored in this table (Figures 19 and 20):

Fig 19: *Tennis player hitting ball hard* <u>*with racket.*</u>   **Fig 20:** Woman leading horse <u>with reins.</u>

Both *hard* and *with a racket* describe how the tennis player hits the ball. "*Hard*" is an adverb and therefore goes into the adverbs table, whereas "*with a racket*" is a prepositional phrase (indicated by the preposition "with") and therefore goes into the manners table. Manner adverbs can be further described by the use of attributes (stored in the relation annotation_manners_adjectives). For example:

Tennis player hitting ball hard with <u>old, wooden</u> racket.

4.3.11 Queries

Stores the representative benchmark queries. The target set (and its size) for a specific query can be derived from the relation between images and queries. The performance measures that result from the queries (see in [13]) are derived values and therefore not stored in the database.

5. CONCLUSION AND FUTURE WORK

This paper describes the establishment of a new benchmark which is set up under the auspices of the Technical Committee 12 of the International Association for Pattern Recognition. Not only does it provide a substantial extension and improvement of the previous one (which was successfully established earlier), a new benchmark administration system has also been built which enables the ongoing refinement and development of the benchmark suite. The use of the MPEG-7 structured annotation standard has also been incorporated, which will help to facilitate high precision complex content search on the Internet based on XML.

The next step of the project will include further refinement, extension (up to 10.000 images) and improvement of the current version of the benchmark using the benchmark administration system. Functions like an improved XML generator (that can handle different approaches of producing XML code for further studies) and new statistical functions will be implemented.

The final goal will be the widespread adoption of this benchmark to be an international standard for visual information system performance evaluation for images with complex image contents. The development of such a benchmark is an incremental and ongoing process. Thus, input and participation from researchers in the international research community will be essential to achieving this final goal.

6. REFERENCES

[1]   Remco C. Veltkamp, Mirela Tanase: „Content Based Images Retrieval Systems: A Survey", Department of Computing Science, Utrecht University, The Netherlands, October 2002.

[2]   AD Narasimhalu, MS Kankanhalli, J. Wu: "Benchmarking Multimedia Databases". *Multimedia Tools and Applications*, 4:333-356, 1997.

[3]   D.P. Huijsmans and A. Smeulders (Eds.) *In the Proceedings of the Third International Conference On Visual Information Systems*, Amsterdam, The Netherlands, 2-4 June 1999, Springer-Verlag LNCS 1614, 1999.

[4]   Photonics West 2000 Conference: San Jose Conservation Centre, http://www.spie.org/web/meetings/programs/pw00/conferences.html, San Jose, Ca, USA, 2000.

[5]   Neil J Gunther, Giordano B Beretta: "A Benchmark For Image Retrieval Using Distributed Systems Over The Internet (BIRDS-I)", *HP Labs Technical Reports HPL 2000-162*, Palo Alto, CA, USA, 2000.

[6]   Clement Leung, Horace Ip: "Benchmarking For Content Based Visual Information Search", In *Advances In Visual Information Systems*, R. Laurini (Ed.), Springer-Verlag LNCS 1929.

[7]   Wolfgang Mueller, Stephane Marchand-Maillet, Henning Mueller, Thierry Pun: "Towards A Fair Benchmark For Image Browsers", *In SPIE Photonics East, Voice, Video And Data Communications,* Boston, MA, USA, November 5-8, 2000.

[8]   Henning Mueller, Wolfgang Mueller, Stephane Marchand-Maillet, Thierry Pun: "Automated Benchmark In Content Based Image Retrieval", *International Conference On Multimedia And Exposition, ICME 2001*, Tokyo, Japan, 2001.

[9]   Stephane Marchand-Maillet: Official Homepage Of VIPER (Visual Information Processing For Enhanced Retrieval), viper.unige.ch, University Of Geneve, Switzerland, October 2002.

[10]   Niels Nes, Martin Kersten: "The Acoi Algebra: A Query Algebra For Image Retrieval Systems", University Of Amsterdam, www.cwi.nl/~niels/pub/bncod/paper/paper.html, 1998.

[11]   Ground Truth Database, University Of Washington, Department of Computer Science and Engineering, www.cs.washington.edu/research/imagedatabase/groundtruth/, Seattle, WA, USA.

[12]   R. Laurini (Ed.) *Proceedings of the 4th International Conference On Visual Information Systems,* Lyon, France, November 2-4, 2000, Springer-Verlag LNCS 1929.

[13]   Michael Grubinger, Clement Leung: "A Benchmark for Performance Calibration in Visual Information Search", *In Proceedings of 2003 Conference of Visual Information Systems*, pages 414 – 419, Miami, Florida, USA, September 24-26, 2003.

[14]   Paul Over, Clement Leung, Horace Ip, Michael Grubinger: *"Multimedia Retrieval Benchmarks". Digital Multimedia on Demand, IEEE Multimedia April-June 2004,* pages 80 - 84.

[15]   Horace Ip (Ed.) *In Proceedings of the 2003 International Conference on Visual Information Systems*, Miami, Florida, USA, September 24-26, 2003

[16]   Transaction Processing Performance Council, TPC Benchmarks, http://www.tpc.org/information/benchmarks.asp, 2001 - 2004

[17]   International Organisation for Standardisation: Information technology -- Multimedia content description interface -- Part 5: Multimedia description schemes, *ISO/IEC 15938-5*, 2003

[18]   Michael Grubinger, Clement Leung: "Benchmark for Content Based Visual Information Search", http://sci.vu.edu.au/~clement/TC-12/images/, Melbourne, Australia, November 2002.

Benchmarking image retrieval applications

H. Müller, A. Geissbuhler
University Hospitals of Geneva
Rue Micheli-du-Crest 24,
CH-1211 Geneva 14

S. Marchand–Maillet
University of Geneva,
Rue du Général Dufour 24,
CH-1211 Geneva 4

P. Clough*
University of Sheffield
211 Portobello Street
Sheffield, S1 4DP UK

Abstract

Since the early 1990s, content–based visual information retrieval has been an important research topic in computer vision. A large number of systems have been developed as research prototypes, as well as commercial and open source systems. However, still no general breakthrough in performance has emerged and important real–world applications stay rare. The large amount of available multimedia information creates a need to develop new tools to explore and retrieve within mixed media databases. The replacement of analog films by digital cameras and the increasing digitisation in fields such as medicine will still increase this need.

One of the reasons for the impossibility to show an increase in performance is the fact that there is no standard for evaluating the performance of content-based retrieval systems. In the last years a rising number of proposals have been made on how to evaluate or not evaluate the performance of visual information retrieval systems which underlines the importance of the issue. Several benchmarking events such as the Benchathlon, TRECVID and Image-CLEF have been started, with varying success. This article describes work carried out by the University of Geneva on benchmarking visual information retrieval systems. A special emphasis will be on the Benchathlon and ImageCLEF evaluation events, their methodologies and outcomes.

1 Introduction

Ideas for content–based retrieval (CBR) in image or multimedia databases (DB) date back to the the early 1980s. Serious applications started in the early 1990s and the most well–known systems are maybe IBM's QBIC [5] and MIT's Photobook [20]. Content–based image retrieval (CBIR) became an extremely active research area with hundreds of systems and several hundred publications. A good overview article is [23]. Although active in research, only very little effort was put into comparing and evaluating the performance of systems. Small, copyrighted DBs were used that made comparisons betwen systems almost impossible and the shown graphs and measures problematic. The related field of text retrieval (TR) already did systematic evaluation and creation of datasets since the early 1960s with the Cranfield studies [1] and SMART [22].

The MIRA (Evaluation frameworks for interactive and multimedia information retrieval (IR) applications) project first focused on visual IR evaluation starting from 1996 [25]. A first article on benchmarking CBIR algorithms was published in 1997 [19]. New measures for evaluation were created but no example evaluation nor a DB was shown. In [24], the TR community and the TREC conference were first mentioned as a role model for visual retrieval evaluation. Leung and Ip [13] mention some minimum requirements with respect to the number of images and methodology used, but still no common DB or ground truth was used. In [12], the evaluation was reduced to one single performance measure which might be convenient for comparisons but will not be a good indicator to compare systems based on various aspects. Huijsmans [9] describes very interesting graphs that include measures such as the collection size and size of the ground truth into precision vs. recall graphs to eliminate the retrieval of relevant documents simply by chance. This is very good, but the comparison of retrieval results across DBs is still problematic. The Benchathlon network for evaluation is described in [7]. This includes concrete measures of performance effectiveness, justification for them and a literature review. Müller et al. [16] describes a more general framework for evaluation and includes a literature review as well as an example evaluation with an openly accessible DB. A more recent review is [10].

Many researchers have been critical of current benchmarking initiatives [6]. Part of the criticism is that current retrieval systems do not perform well enough to realistically benchmark them and that they are too separate from real user needs for results to be meaningful to end-user applications. This is not without reason. The current low level features correspond only sometimes to concepts that users are looking for. It is important, therefore, to evaluate systems based on real user needs, i.e. on what a real user is looking for. Only systematic evaluation can show system improvements. Not evaluating at all does not advance any system. The basic technologies for CBIR are available but now is the time to find out which technology works for what kind of queries.

2 Benchmarking components

An complete benchmark will include several components. The most important of these is the creation and availability of standard or common DBs, of typical search tasks, and ground truths for these tasks against which to compare and evaluate new systems. Following this, one can discuss and compare system performance at an organised evaluation campaign.

2.1. Data sets

Currently, the de–facto standard for image retrieval are still the Corel Photo CDs. However, there are problems with these including: they are fairly expensive, copyrighted and not available as a public resource, and they are now unavailable on the market-place. A request from our University to Corel for using lower–resolution images for benchmarking was not answered. A DB that is available free of charge and copyright and is used for evaluation is that of the Uni. of Washington. It contains around 1000 images that are clustered by regions. Other DBs are available for computer vision research but only rarely for image retrieval. The Benchathlon also created a test DB, but currently without search tasks and ground truths. In specialised domains such as medical imaging, there are DBs available. The National Institute for Health (NIH) publishes free of charge all the DBs gathered. A medical DB used for retrieval is that of casimage[1] [18]. In TR, the need for DBs was, again, identified very early on and test sets have been for years at the very core of evaluation [26]. For images, there is an effort to create annotated DBs [11] that can further on be used for system evaluation.

2.2. Query tasks and topics

The first question when evaluating a system should actually be *"What do we want to evaluate?"*. The goal for evaluation should be based on real user needs and not a computer vision expert's interest. Some studies have been performed on how real users query image DBs [14, 4] but too few and they are currently all based on users searching with text. Normally, there should be a selection of query tasks based on real–world user queries and then, images or textual formulations should be taken to select evaluation topics that can be used to compare systems. This will deliver results that correspond to what a user would expect from a system, and systems can consequently be optimised for these goals.

2.3. Ground truth

Of course, users can for simplicity be simulated to asses the system performance [27]. Like this, the system developer can define noise levels and as a consequence the system

performance. Real ground truth or a gold standard will need to include real users that assess the system performance for each query task and topic. This is expensive and involves much work. It has successfully been done in the major evaluation campaigns and much literature is available on statistical significance testing and problems when using pooling schemes to reduce the number of documents that the relevance assessors will have to watch [28].

2.4. Evaluation measures

A good review of performance measures used for image retrieval can be found in [17]. Although good descriptors that are easy to interpret are important for retrieval system evaluation, this is not the main problem at the moment. The measures can only be as good as the DB and ground truth available which is the current problem. Simple measures based on precision and recall, and especially precision vs. recall graphs seem to be the accepted standard for CBIR.

2.5. Benchmarking events

TR used to have several standard DBs that were used for evaluation since the 1960s [1]. Still, the single big event that showed a significant increase in performance was TREC[2] (Text REtrieval Conference) starting from 1992 [8]. TREC is a "friendly" benchmarking event for which large data sets and sets of seach tasks are generated, and systems compared based on this new data each year. Several subtasks have become independent conferences in the meantime as they grew bigger and more important (e.g. CLEF, TRECVID). Unfortunately a request to include CBIR into TREC was denied with the explication that there were no DBs available that could be distributed and were judged large enough.

Image retrieval does need a benchmarking event such as TREC to meet and discuss technologies based on a variety of DBs and specialised tasks (medical image retrieval, trademark retrieval, consumer pictures, ...)! This will allow having standard datasets, to identify good and less good techniques as well as performant interaction schemes. System improvements can be shown over time with such an event.

3. Events for visual information retrieval

3.1. TRECVID

TRECVID was introduced as a TREC task in 2001 with subtasks in shot–boundary detection and search tasks, mainly based on a textual description. Data sets in 2003 contain more than 130 hours of video in total. Video is different from images in that the speech and captions can be translated into text and thus, more that low–level visual descriptors can be used for semantic queries. The number of participants for TRECVID has grown steadily from 12 in 2001 to 24 in 2003.

---

[1]http://www.casimage.com/

[2]http://trec.nist.gov/

The number of subtasks has also grown and includes now story segmentation and classification as well as higher level feature extraction. This can be the recognition of a group of people etc. TRECVID is a success and has created a meeting point where technologies and their influences on retrieval can be discussed and compared based on the same datasets. Test collections have been created and can be used to optimise the system performance for future tasks.

3.2. The Benchathlon

The Benchathlon[3] was created in the context of the SPIE Photonics West conference, one of the important conferences for CBIR. The goal was to create a workshop where benchmarking and evaluation could be discussed among researchers and industry and where a benchmarking event for image retrieval was to be started. An evaluation methodology was developed [7] stating performance measures and their justification. An interactive evaluation methodology based on the Multimedia Retrieval Markup Language (MRML[4]) was presented [15] to allow interactive evaluation of systems. This was supposed to take into account the importance of relevance feedback (RF) for the evaluation of image retrieval systems. Based on real user ground truth, the behaviour on marking positive/negative feedback can be automised and used for evaluations.

2001 saw the first Benchathlon with basically a presentation of the outline document [7] and discussions among participants. In 2002 a first workshop with five presentations was held and this number raised to 8 in 2003. Unfortunately, the goal to really compare the systems' performance was not reached. Efforts included the generation of a DB containing a few thousand private pictures and a partly annotation of these [21]. Ground truth has not yet been generated for query topics to evaluate system performance. The proposed architecture for automatic evaluation was not accepted by many research groups either, although efforts were taken write tools for participants and help them to install an MRML–based system access.

3.3. ImageCLEF

The Cross Language Evaluation Forum (CLEF[5]) started as a subtask of TREC to allow IR over languages where for example the queries are in a different language than the documents. CLEF began in 2000 taking two days and listing over 25 papers in the proceedings. In 2003, one of the subtasks included was ImageCLEF[6], for the evaluation of cross language image retrieval systems [2]. ImageCLEF started with 4 participants using a DB of approximately 30,000 historic

---

[3]http://www.benchathlon.net/
[4]http://www.mrml.net
[5]http://www.clef-campaign.org/
[6]http://ir.shef.ac.uk/ImageCLEF2004/

photographs from St. Andrews University Images have English annotations and typical search requests were created in a variety of languages. The queries include one query image plus a textual description of the query.

Figure 1. Examples St. Andrews collection.

Figure 1 shows images of the DB. The fact that most images are in grey or brown scales also explains why, in 2003, there was no use of visual retrieval algorithms in the competition. The kind of query topics are very hard to answer visually as they are not based on the visual content but the semantics of the image. For this reason, in 2004, a more visual retrieval task will be added to ImageCLEF in the domain of medical images and an interactive task has also been added [3] to include some more user–centred evaluation. Figure 2 shows some example images from this DB that contains a total of almost 9000 medical images [18] of a medical teaching file including annotations in French and English.

Figure 2. Examples medical collection.

Query topics (26 in total) were chosen by a radiologist to represent the entire DB. Ground truthing is performed by radiologists. The search task is expressed as an image only, but within the DB, images are accompanied by texts describing medical conditions in French or English. This makes the task cross-language, but also gives particular potential to visual IR. Automatically extracted visual information is inherently insensitive to language and can thus be an important aid to cross–language IR. On the other hand, the combination of textual and visual cues can also deliver important results for the visual IR community as it adds semantics which are not easily derived from the image itself. With this, both the cross language and image retrieval communities can profit from the other to improve system performance for certain search tasks and obtain new insight into this particular type of IR. The 2004 competition has 10 participants for a set of search tasks based on the St. Andrews data, and 10 for tasks based on the medical data. This improvement from 4 in 2003 to 20 in 2004 shows the perceived importance of image retrieval within the context of cross–language IR. Entries vary widely from those using purely textual methods to those using purely visual ones. A large number of entries have also experimented with combining text and visual methods to increase performance. Further techniques such as automatic

query expansion and manual RF have been submitted by participants, as well as the use of various translation resources.

4. Conclusions

The CBIR community needs a common effort to create and make available datasets/query topics and ground truth to be able to compare the performance of various techniques. A benchmarking event is needed more than ever to give a discussion forum for researchers to compare techniques and identify promising approaches. Especially the use of multi–modal DBs and of cross–language IR on the evaluation of image retrieval algorithms is important as many real–world collections such as the Internet have exactly these characteristics. Strong participation in events such as TRECVID and ImageCLEF shows that there is a need to share data and results to advance visual IR.

References

[1] C. W. Cleverdon. Report on the testing and analysis of an investigation into the comparative efficiency of indexing systems. Technical report, Aslib Cranfield Research Project, Cranfield, USA, 1962.

[2] P. Clough and M. Sanderson. The CLEF 2003 cross language image retrieval task. In *Proceedings of the Cross Language Evaluation Forum*, 2004.

[3] P. Clough, M. Sanderson, and H. Müller. A proposal for the CLEF cross language image retrieval track 2004. *The Challenge of Image and Video Retrieval*, Dublin, Ireland, 2004.

[4] P. G. B. Enser. Pictorial information retrieval. *Journal of Documentation*, 51(2):126–170, 1995.

[5] M. Flickner, et al. Query by Image and Video Content: The QBIC system. *IEEE Computer*, 28(9):23–32, 1995.

[6] D. A. Forsyth. Benchmarks for storage and retrieval in multimedia databases. In *Storage and Retrieval for Media Databases*, vol 4676, pp 240–247, San Jose, USA, 2002.

[7] N. J. Gunther and G. Beretta. A benchmark for image retrieval using distributed systems over the internet: BIRDS–I. Technical report, HP Labs, Palo Alto, Technical Report HPL–2000–162, San Jose, 2001.

[8] D. Harman. Overview of the first Text REtrieval Conference. In *Proceedings of the first Text REtrieval Conf.*, pp 1–20, Washington DC, USA, 1992.

[9] D. P. Huijsmans and N. Sebe. Extended performance graphs for cluster retrieval. In *Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition*, pp 26–31, Kauai, Hawaii, USA, 2001.

[10] I. Jermyn, C. Shaffrey, and N. Kingsbury. The methodology and practice of the evaluation of image retrieval systems and segmentation methods. CNRS Rapport de recherche ISRN I3S/RR-2003-05-FR, Sophia antipolis, 2003.

[11] C. Jörgensen. Towards an image testbed for benchmarking image indexing and retrieval systems. In *Proceedings of the Int. Workshop on Multimedia Content–Based Indexing and Retrieval*, Rocquencourt, France, 2001.

[12] M. Koskela, J. Laaksonen, S. Laakso, and E. Oja. Evaluating the performance of content–based image retrieval systems. In *Int. Conf. On Visual Information Systems*, LNCS 1929, Lyon, France, 2000.

[13] C. Leung and H. Ip. Benchmarking for content–based visual information search. In *Int. Conf. On Visual Information Systems*, LNCS 1929, pp 442–456, Lyon, France, 2000.

[14] M. Markkula and E. Sormunen. Searching for photos – journalists' practices in pictorial IR. In *The Challenge of Image Retrieval, A Workshop and Symposium on Image Retrieval*, Newcastle upon Tyne, 1998.

[15] H. Müller, W. Müller, S. Marchand-Maillet, D. M. Squire, and T. Pun. A web–based evaluation system for content–based image retrieval. In *Proceedings of the Int. Conf. on Multimedia*, pp 50–54, Ottawa, Canada, October 2001.

[16] H. Müller, W. Müller, S. Marchand-Maillet, D. M. Squire, and T. Pun. A framework for benchmarking in visual information retrieval. *Int. Journal on Multimedia Tools and Applications*, 21:55–73, 2003.

[17] H. Müller, W. Müller, D. M. Squire, S. Marchand-Maillet, and T. Pun. Performance evaluation in content–based image retrieval: Overview and proposals. *Pattern Recognition Letters*, 22(5):593–601, 2001.

[18] H. Müller, A. Rosset, A. Geissbuhler, and F. Terrier. A reference data set for the evaluation of medical image retrieval systems. *Computerized Medical Imaging and Graphics*, 2004.

[19] A. D. Narasimhalu, M. S. Kankanhalli, and J. Wu. Benchmarking multimedia databases. *Multimedia Tools and Applications*, 4:333–356, 1997.

[20] A. Pentland, R. Picard, and S. Sclaroff. Photobook: Tools for content–based manipulation of image databases. *Int. Journal of Computer Vision*, 18(3):233–254, 1996.

[21] T. Pfund and S. Marchand-Maillet. Dynamic multimedia annotation tool. In *Internet Imaging III*, vol 4672, pp 216–224, San Jose, USA, 2002.

[22] G. Salton. *The SMART Retrieval System, Experiments in Automatic Document Processing*. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1971.

[23] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content–based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 No 12:1349–1380, 2000.

[24] J. R. Smith. Image retrieval evaluation. In *IEEE Workshop on Content–based Access of Image and Video Libraries (CBAIVL'98)*, pp 112–113, Santa Barbara, CA, USA, 1998.

[25] E. Sormunen, M. Markkula, and K. Järvelin. The perceived similarity of photos – seeking a solid basis for the evaluation of content–based retrieval algorithms. In *Final MIRA Conference*, Glasgow, 14–16 April 1999.

[26] K. Sparck Jones and C. van Rijsbergen. Report on the need for and provision of an ideal information retrieval test collection. British Library Research and Development Report 5266, Computer Laboratory, Uni. of Cambridge, 1975.

[27] J. Vendrig, M. Worring, and A. W. M. Smeulders. Filter image browsing: Exploiting interaction in image retrieval. In *Third Int. Conf. On Visual Information Systems*, LNCS 1614, pp 147–154, Amsterdam, The Netherlands, 1999.

[28] J. Zobel. How reliable are the results of large–scale information retrieval experiments? In *Proceedings of the 21st Annual Int. Conf. on Research and Development in Information Retrieval*, pp 307–314, Melbourne, Australia, August 1998.

Constructing 3D Point Correspondence for Vasculature using Approximate Parallelism

Jun Feng[1], Horace H. S. Ip[1, 2], and Shuk H. Cheng[3]

1 Image Computing Group, Department of Computer Science, City University of Hong Kong,
2 Centre for Innovative Applications of Internet And Mulitmedia Technologies (AIMtech)
City University of Hong Kong,
3 Department of Biology and Chemistry, City University of Hong Kong.

Abstract

This paper presents a method that uses approximate parallelism to construct the 3D point correspondence for vascular structures. Four types of approximated parallelism inherently present in the vasculature are investigated and their common characteristics are formalized into a set of force models. Then the fuzziness of the parallelism and the points coupling between a pair of vessels are achieved by a parallelism-based force-driven correspondence detection strategy. The resulting point correspondences are perceptually good and can be used for 3D model matching and visual information queries based on point distribution models and morphological analysis and quantification of the reconstructed vasculatures.

1. Introduction

Vasculature system is an organ system for oxygen and nutrient delivery as well as a conduit for communication between distant tissues. The importance of the vascular system was demonstrated by the lethality of animals with vascular defects [1-4]. Recent years, many medical reports and statistical analysis have indicated that vascular diseases have become one of the major sources of deaths for human beings [5]. To gain more understanding of the human vasculature development and diseases, angiography and microangiography techniques are used to observe circulatory patterns, both for human and model animals. A good survey about different imaging modalities of angiography could be found in [6,7].

However, it has been proven that automatically segmenting and reconstructing 3-D vascular models from angiographic images is very difficult [8], especially for the *in vivo* model animals due to the subtle effects of tissue characteristics and sample motion. In [9], we have designed and implemented a visual information system to extract 3D models of vasculature and to quantify and

compare the vascular structures obtained from microangiographic images series of zebrafish embryo, which is considered as one of the best model species to study human vascular development [10-12]. To obtain a more robust segmentation and reconstruction result, a relational-tubular deformable model (*Retu*) has been developed in [13]. Based on *Retu*, we are able to segment and obtain many variants of the 3D vasculatures of zebrafish. In order to understand the variability of the geometry and the quantitative morphology of these vasculature samples within a visual information system, computing the 3D correspondence of these 3D sample models is needed.

It is well known that solving the correspondence problem is a challenge and a prerequisite for many biomedical imaging applications such as organs growth measurement, aesthetic surgery evaluation, histological section registration etc. [14-18]. For angiographic image analysis in particular, point correspondences are established mainly for three purposes. Firstly, corresponding points are needed in the different projective angiographic images such as a pairs of Digital Subtraction Angiogram to reconstruct 3D vessel models [14,15]. Secondly, matching of the corresponding salient points to perform vessel quantification and growth measurement comparison [16]. Lastly, correspondence needs to be established for constructing point distribution models [17,18]. To solve the correspondence problems in the above tasks, most of the solutions reported are based on local and partial shape matching; typically by comparing the shape descriptor distances between salient points on the vessels [15-18].

However, in studying segmentation and reconstruction issues on the vasculature, we find that it is very difficult to identify or to define the corresponding points for the vessel segments using these shape descriptors. For example, in the caudal vasculature of zebrafish embryo, underdeveloped plexus are often conglutinated between the caudal artery (CA) and caudal vein (CV) (A vasculature atlas of the zebrafish is shown

in figure 1c. To visualize the structure of the caudal vasculature and quantify the well-grown vessels accurately, a multi-orientation dissections based r technique [19] is developed to segment and reconstruct the major vessels while filter out the plexus. One of the resulting caudal vascular models is shown in figure 1b. Although the connectivities between intersegmental vessels (Se) and CA are ideal to serve as the salient points for establishing correspondence, few meaningful feature points on CV could be identified using traditional approaches, which makes the vascular analysis and quantification (for example, the measurement of the distance between caudal vein and caudal artery) very difficult. Furthermore, local descriptors for partial shape matching such as curvature, and tangent are not helpful

Figure 1. Vasculature of zebrafish embryo. (a) One of the microangiographic images examples. (b) Reconstructed 3D model (c) Atlas of caudal vasculature

here to find useful salient points since tortuousness and local irregularity is an inherent characteristic of the vessels.

Interestingly, in this circumstance, humans are still able to make out relatively important points and to establish rough correspondence between two vessels based on their spatial geometric relationship. For example, for the model shown in figure 1b, many people may choose the blue marked points along CV as the landmark points and think these points have somehow correspondence with the connectivities on CA according to approximate parallelism between CA and CV. It has been shown that perceptual organization was a primitive level of inference that could operate on its own without high-level knowledge [20]. Psychologists believe that primitive grouping in our vision system is done on proximity, collinearity, curvilinearity, parallelism,

symmetry, closure and repetitive patterns [21]. Since for vasculature, approximate parallelisms frequently exist between a pair of vessels, it is reasonable to establish the point correspondence based on this kind of primitive knowledge.

In this paper, we present a method to define and to locate the corresponding points for a pair of vessels based on determining their approximate parallelism. To our knowledge, no literature has been reported to address this approach to the problem. Four classes of approximate parallelism frequently existed in a pair of vessels have been identified and our previous work on a force-driven optimization model has been adopted to incorporate the fuzzy parallelism information between the coupling point pairs. Then the final corresponding points could be obtained by a parallelism-based force-driven correspondence detection scheme.

The rest of the paper is organized as follows. Different types of approximate parallelism existed in the vasculature are analyzed in section 2, as well as a summarization of their common characteristics. Section 3 sketches the parallelism–based force-driven models and section 4 presents the correspondence-matching scheme for a pair of approximate parallel vessels. We show our experiment results in section 5 and conclude the paper in the end.

2. Existence of approximate parallelism in a pair of vessels and their characteristics

In [22], we investigated three kinds of perceptual parallel models for planar curves. However, for loose parallelism between two vessels, the morphology and properties are slightly different since basically, no rigorous mathematical definition on parallelism could be found for naturally occurring vasculature. Through observing many sample vasculatures, we can, instead, enumerate approximate parallelism that exists in a pair of vessels. Please note that the parallelism discussed here is basically a perceptual parallelism in 3D Euclidean space. We denote the vessel with a set of salient points as active vessel $V_A(s)$ and the vessel being matched as the passive vessel $V_P(s)$, s ranges from 0 to the length of the vessel. In this 3D vessel modeling, $V_*(s) = (x_*(s), y_*(s), z_*(s))$, where $* \in \{A, P\}$. We also use $V_*(s_k)$ to denote the discrete points along the medial axis of the vascular model, $1 \le k \le n$. Corresponding points on the passive vessel could also be represented as $V_P(f(s_k))$ since they are defined as a one-to-one mapping to the salient points on the active vessel. $V_A(s_k)$ and $V_P(f(s_k))$ are a conjugate pair in the approximate parallelism sense.

1. *Approximate Railroad Parallelism.*

In this type of approximate parallelism, two vessels together looks like a foreshorten railroad track as shown in figure 2a. Along the tracks, the shortest distance between any two points belong to either vessel is roughly geometric proportion. Mathematically, given two adjacent conjugate point-pairs, we have:

$$\frac{distance(V_A(s_k), V_P(s_k))}{distance(V_A(s_{k+1}), V_P(s_{k+1}))} \approx c \qquad (1)$$

2. Approximate translation parallelism.

The parallel vessels in this class are roughly achieved by translation (Figure 2b). It could be represented as:

- $V_P(s) \approx (x_A(s) + u, y_A(s) + v, z_A(s) + w)$ (2)

- $\theta_A(s) \approx \theta_P(s)$ where $\theta(s)$ is the tangent orientation at point s.

Figure 2. Diversiform approximate parallelism exhibits in vasculature

3. Approximate central similarity transformation parallelism.

Suppose o is the center of the central similar vessels with a positive similarity coefficient κ, the coupling point-pairs should be established by drawing lines from o and cutting both V_A and V_P, Then the approximate central similarity transformation parallelism (figure 2c) could be represented as: $\frac{distance(O, V_A(s_k))}{distance(O, V_P(s_k))} \approx \frac{1}{\kappa}$ (3)

4. "Bracket parallelism"

We could not say a pair of curves with bracket shape is parallel. However, the couple points on a pair of vessels that looks like a parenthesis would have similar relationship as equation (1). Furthermore, if one of the bracket vessels is projected with respect to its chord, it may be approximate parallel to the other bracket vessel. We make use of this property in section 3. This kind of

parallelism frequently appears in the vasculature, as shown in figure 2 d-f.

Based on the previous discussions, four common criteria could be extracted from the above cases of approximate parallelism between a pair of vessels. (1) **Sleepers Criterion.** We called the lines joining the corresponding couples of active and passive vessels sleepers. For approximate parallel vessels, sleepers should be either approximately parallel (parallelism case 1,2,4) or approximately concurrent (case 3). (2) **Tangent Alignment Criterion.** The deflection of the tangent at couple pairs on active and passive vessels should not be too large (parallelism case 2,3). Or, the tangents of corresponding pairs should be roughly symmetric equal about the chords (case 1,4). (3) **Similarity Criterion.** The two vessels should be similar (case 2,3) or symmetric similar in shape (case 1,4) (4) **Elasticity Criterion.** If the salient points in active vessels are basically evenly distributed, the distances between successive coupled points in the passive vessels should lengthened or shortened more or less uniformly (for all cases).

3. Force models for representing the approximate parallelism criteria

As we have mentioned previously, to get the corresponding points on the passive vessel by detecting the approximate parallelism of the active vessel, shape-matching approaches based on local descriptors could not work. Furthermore, for the objects like blood vessels which have highly irregular local geometric features, exact correspondence does not exist and does not make sense. The criteria described in previous section only suggest how the mapping scheme should be constructed and could not give us hard and fast rules to follow. What we need is a method to obtain the fuzzy correspondence within the framework of approximate parallelism to simulate human perceptual inference organization. This requirement limits the applicability of the traditional optimization techniques. In [23], we proposed a force-driven optimization solution for detecting perceptually parallel curves. The basic idea is formulating perceptual parallelism characteristics into a number of competing forces and letting them counteract each other until they arrive at equilibrium. In this way, the fuzziness built in human vision system could be coded into the force-driven scheme to allow matching of moderately deformed shapes. We believe that this approach is very suitable for finding the corresponding couples on the passive vessels based on the approximate parallelism of the active vessels. Instead of looking for the globally optimal solution that satisfies all of the criteria, we could obtain an acceptable coupling result by using the force-driven local optimization technique. To accommodate our vessel correspondence problem, the forces in [23] are slightly

modified and extended. Due to the limited space, we briefly list the force models as follows. . Please refer to the original paper for full details and the relevant mathematical theorems.

- Deflection Force (F_1 ----*DF*).

This force serves to constrain the direction of the coupling points on passive vessels to the expect orientation which makes the sleepers to be approximately parallel or concurrent each other (figure 3). The force is calculated as: (4)

$$DF(S_k) = \begin{cases} \left\langle (P_5 - P_0) \times (P_6 - P_5) \right\rangle & if & \left\langle (P_2 - P_1).(P_4 - P_3) \neq 1 \right\rangle \\ \left\langle (P_2 - P_1) \times (P_6 - P_5) \right\rangle & otherwise \end{cases}$$

where $\times$ is the cross product between two vectors.

- Tangent Alignment Force (F_2 ----*TAF*).

TAF adjusts misaligned tangent to ensure similar tangent directions of two coupling points (Figure 4a). For the approximate bracket parallelism described in section 2, tangent directions of the active vessels $V_A(s_k)$ are calculated along their mirror symmetric vessels $V_{A'}(s_k)$ with respect to its chord C_A (Figure 4b). This force could be represented as:

$$TAF(s_k) = \left\langle DDT(V_{A'}(s_k)) \times DDT(V_p(f(s_k))) \right\rangle \quad (5),$$

where *DDT* is the direction dependent tangent vectors given in [23].

Figure 3. Diagram of Deflection Force

Figure 5. Diagram of Approximation Error Force

Figure 4. Diagram of Tangent Alignment Force

- Approximation Error Force (F_3 ----*AEF*).

The aim of *AEF* is to generate a good approximation of passive vessels by the set of coupled points $V_P(f(s_k))$, which could make the two vessels similar in shape. The accuracy of this approximation could be indicated by the perpendicular distance of the investigated segment from the chord, as shown in figure 5.

Since the *AEF* comes from two sources, one is a force at $V_P(f(s_k))$ to push it towards $V_P(f(s_{k+1}))$ and the other is a force at $V_P(f(s_{k-1}))$ to push it towards $V_P(f(s_k))$, we define the *AEF* as:

$$AEF(s_k) = d(V_P(f(s_k))) - d(V_P(f(s_{k-1}))) \quad (6)$$

- Elasticity Force (F_4 ----*EF*)

EF is modeled to meet the requirement of uniform interval between successive corresponding points on the passive vessels. If we approximate the sleepers by the elastic strings, *EF* could be described as

$$EF(s_k) = \frac{L(f(s_k), f(s_{k+1}))}{L(s_k, s_{k+1})} - \frac{L(f(s_{k-1}), f(s_k))}{L(s_{k-1}, s_k)} \quad (7)$$

where $L(p_1, p_2)$ denotes arc length between point p_1 and p_2.

Figure 6. Diagram of Elasticity Force

4. Parallelism–based force-driven correspondence detection scheme

Given a pair of vessels, normally one artery and one vein, to get the point correspondence on the passive vessels based on approximate parallelism, first, we should locate the salient points on the active vessel. In most cases, the salient points include junctions and landmark points identified by biologists or other feature points [24]. More sample points on active vessel may be interpolated evenly between the identified salient points to approximate the vessel better. Then for each point s_k at active vessel, we need to find a corresponding point $f(s_k)$ on the passive vessel to meet the approximate parallelism criteria.

Similar to our previous work in [22] and based on the force models described previously, the main idea of our parallelism-based correspondence detection scheme is to search the corresponding point on the passive vessel,

which achieves $\tau(f(s_k)) = MIN(\sum_{i=1}^{4} \omega_i F_i(s_k))$ (8)

where ω_i are the weights assigned to different kinds of forces and F_i are the forces described in section 3. Since the approximate parallelism criteria have been formulated to forces, the minimum resultant of the combination forces could be achieved at the ideal corresponding points in rough parallelism sense.

In our implementation, the process is based on iterative greedy search. First, some candidate points are selected on the passive vessel randomly. At each iteration, for every candidate point $f(s_k)$ on the passive vessel, we calculate four forces and combine them as in equation (8) to get the resultant force $\tau(f(s_k))$. $\tau(f(s_k))$ actually describes the amount of the deviation of the candidate points. Suppose s_0 to s_n have been ordered from the start point to the end point of the active vessel. If $\tau(f(s_k))$ is not less than a small threshold, the moving direction $g(f(s_k))$ of the candidate $f(s_k)$ could be decided as:

$$-g(f(s_k)) = \begin{cases} DDT(f(s_k)) & if \quad \tau(f(s_k)) > threshold \\ DDT(f(s_k)) & if \quad \tau(f(s_k)) < -threshold \end{cases}$$
(9)

That is, the searching direction depends on the sign of $\tau(f(s_k))$. If $\tau(f(s_k)) > threshold$, $f(s_k)$ should move closer to $f(s_{k+1})$, otherwise, the search should be performed towards the direction of $f(s_{k-1})$. Since the deformation of the neighborhood points may update the amount of the forces upon one candidate point, the iterative process will continue unless all of $f(s_{k-1})$ satisfy $\tau(f(s_k))$ fells below a small threshold.

5. Experimental results on zebrafish embryo vasculature

We test our parallelism-based force-driven point correspondence searching scheme on the 3D model of the caudal vasculature of zebrafish embryo. The 3D vasculature is reconstructed from the microangiographic image series visualized by the confocal laser scanning microscopy [9]. One of the angiographic image examples has been shown in Figure 1a. In our experiment, the weights in equation (8) are set in the ration 150:50:5:30, which are the same as [23] for the normalization purpose. The correspondence lines between salient points on active vessels and resulting corresponding points on passive vessels for diversiform approximate parallel vessels are shown in figure 7. They are perceptually good and basically present the approximately parallel relationship between the vessels. Since in approximate parallelism there are in fact no exact rules to follow, therefore it is difficult to evaluate whether the resulting correspondence are indeed true correspondence. However, we attempt to validate our matching scheme on a special segment of vasculature of zebrafish. As shown in figure 8, intersegmental vessels (Se) appear in short intervals along the dorsal axis of the fish, connecting both DLAV (upper vessel in Figure 8) and CA (lower vessel), almost evenly. We try to locate the corresponding points for DLAV using our force-driven scheme by selecting CA as the active vessel, and junctions as the salient points. We could find that the resulting correspondences are closely adjacent to the junctions on DLAV, which does represent the uniformity property of the Se. These correspondence locations and matching are valuable for biologists and could be used as landmark for vascular quantification and measurements.

Figure 7. Experiment result on landmark points searching and correspondence matching in the caudal vasculature of zebrafish embryo. In all of the cases, upper or left vascular are active vessels.

Figure 8. The resulting corresponding matching on piece of vasculature segment of zebrafish.

6. Conclusion

In this paper, we propose a method to construct the point correspondence for a pair of vessels which do not possess obvious salient points. We observe that approximate parallelism frequently exists in a pair of vessels and we exploit the concept of perceptual grouping to obtain the corresponding points. Four types of approximate parallelism in a vasculature are introduced and the common criteria are extracted. These criteria are

adopted for formulating the forces models for a force-driven optimization process. The fuzziness of the approximate parallelism between vessels is incorporated into the force-driven correspondence detection scheme. The resulting corresponding points achieve equilibrium in the force field in which the competing forces are cancelled or counteracted each other. The established correspondence matching is visually acceptable. This method is applicable to building point distribution models and also valuable for vascular quantification and measurements in visual information systems. It also provides a good foundation for developing promising solutions that make use of primitive perceptual knowledge present in biological organs.

Acknowledgements

References

[1]   P. Carmeliet, V. Ferreira, G. Breier, S. Pollefeyt, L. Kieckens, M. Gertsenstein, M. Fahrig, A. Vandenhoeck, K. Harpal, C. Eberhardt, C. Declercq, J. Pawling, L. Moons, D. Collen, W. Risau, A. Nagy, Abnormal blood vessel development and lethality in embryos lacking a single VEGF allele, Nature 380, 435-439 ,1996.

[2]   N. Ferrara, K. Carver-Moore, H. Chen, M. Dowd, L. Lu, K.S. O'Shea, L. Powell-Braxton, K.J. Hillan, M.W. Moore, Heterozygous embryonic lethality induced by targeted inactivation of the VEGF gene, Nature 380, 439-442 , 1996.

[3]   F. Shalaby, J. Rossant, T.P. Yamaguchi, M. Gertsenstein, X.F. Wu, M.L. Breitman, A.C. Schuh, Failure of blood-island formation and vasculogenesis in Flk-1-deficient mice, Nature 376, 62-66, 1995.

[4]   F. Shalaby, J. Ho, W.L. Stanford, K.D. Fischer, A.C. Schuh, L. Schwartz, A. Bernstein, J. Rossant, J., A requirement for Flk1 in primitive and definitive hematopoiesis and vasculogenesis, Cell 89, 981-990 , 1997.

[5]   http://www.hopkinsmedicine.org/vascsurg/CEREBROVASC1A1.html#Anchor-47857

[6]   Jasjit S. Suri, Kecheng Liu, Laura Reden , "A Review on MR Vascular Image Processing Algorithms: Acquisition and Prefiltering: Part I" , IEEE Transactions on information technology in biomedicine, Vol 6 No 4 324-335 ,December 2002

[7]   Jasjit S. Suri, Kecheng Liu,Laura Reden , "A Review on MR Vascular Image Processing Skeleton Versus Nonskeleton Approaches: PartII , IEEE Transactions on information technology in biomedicine, Vol 6 No 4, 338-350 December 2002

[8]   Wang, K.C.; Dutton, R.W.; Taylor, C.A.; Improving geometric model construction for blood flow modeling, IEEE Engineering in Medicine and Biology Magazine, November/December 1999 33-39.

[9]   Jun Feng, Shuk Han Cheng, Horace Ip " Reconstruction and representation of caudal vasculature of zebrafish embryo from confocal scanning laser fluorescence microscope images ", Accept by Computers in Biology and Medicine.

[10]   http://www.nih.gov/science/models/zebrafish/

[11]   Lawson, N.D. and Weinstein, B.M., . Arteries and veins: making a difference with zebrafish. Nature Reviews in Genetics, 3, 674-682, 2002

[12]   Weinstein, B. M., Stemple, D. L., Driever, W. and Fishman, M. C.,   Gridlock, a localized heritable vascular patterning defect in the zebrafish. Nature Medicine, 1, 1143–1147, 1995.

[13]   Jun Feng, Horace Ip, Shuk Han Cheng, "A Relational-Tubular (ReTu) Deformable Model for Vasculature Quantification of Zebrafish Embryo from Microangiography Image Series", Accept by Computerized Medical Imaging and Graphics.

[14]   Elizabeth Bullitt et. al, , 3D Graph Description of the Intracerebral Vasculature from Segmented MRA and Tests of Accuracy by Comparison with X-Ray Angiograms, IPMI 99, Lectur Notes in Computer science 1613:1999 , 308-321

[15]   J. Y. Lee, C.H. Chen, J. M. Tasai, Y. N. Sun and C. W Mao, 3D Image Reconstruction of Brain Vessels from Angiograms, Proceedings of 1996 IEEE TENCON. Digital Signal Processing Applications ,Volume: 2 , 26-29, 547 - 552, Nov. 1996

[16]   Elizabeth Guest, Elizabeth Berry, Richard A.Baldock, Marta Fidrich, and Mike A. Smith, Robust Point Correspondence Applied to Two and Three-Dimensional Image Registration, IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 23. No. 2,165-179. Feb 2000.

[17]   T.F.Cootes, C.J.Taylor, D.H.Cooper,and J.Graham , "Active Shape Models---Their Training and Applications" , Computer Vision and Image Understanding Vol.61 No. 1, , page 38-59, January, 1995.

[18]   Horace H.S. Ip, Dinggang Shen , An affine-invariant active contour model (AI-Snake) for model-base segmentation , Image and Vision Computing Volume 16, Issue 2, 20, 135-146 , February 1998

[19]   Horace H. S. Ip, Judy J. Feng, and Shuk H. Cheng, Automatic Segmentation and Tracking of Vasculature from Confocal Scanning Laser Fluorescence Microscope Using Multi-Orientation Dissections, The First IEEE International Symposium on Biomedical Imaging, Washington DC, USA. July 7-10, 2002. 249-252

[20]   A. Witkin and J. Tenenbaum, On the role of structure in vision, Human and Machine Vision, eds. J. Beck, B. Hope et al, Academin Press, New York, 481-543,1983

[21]   M. Wertheimer, Principles of perceptual organization (translated), Readings in perception, Van Nostrand, Princeton, NJ, 1958.

[22]   W. H. Wong and Horace H.S. Ip, "On the Detection of parallel curves: Models and Representations", International Journal of Pattern Recognition and Artificial Intelligence Vol. 10 No. 7 813-827, 1996

[23]   Horace H.S. Ip and W.H. Wong, "Detecting Perceptually Parallel Curves: Criteria and Force-Driven Optimization", Computer Vision and Image Understanding Vol. 68, No.2. 190-208, Nov. 1997

[24]   B.Ray and K. Ray, An algorithm for detecting of dominant points and polygonal approximation of digital curves, Pattern Recognition Letters 13, 849-856,1992

CVPIC based uniform/non-uniform colour histograms for compressed domain image retrieval

Gerald Schaefer and Simon Lieutaud

School of Computing and Technology, The Nottingham Trent University
Nottingham, United Kingdom
gerald.schaefer@ntu.ac.uk

Abstract

We introduce a compressed domain image retrieval technique based on the Colour Visual Pattern Image Coding (CVPIC) compression method. The proposed algorithm allows the calculation of image features and hence content-based image retrieval (CBIR) to be performed directly on CVPIC compressed data without the need of decoding it beforehand. In particular, we make immediate use of the fact the uniform and non-uniform areas are coded separately in CVPIC and calculate two distinct histograms similar to the colour coherence vector approach. Retrieval results on the UCID dataset show good retrieval performance, outperforming methods such as colour histograms, colour coherence vectors, and colour correlograms.

1 Introduction

With the rise of the Internet and the availability of affordable digital imaging devices, the need for content-based image retrieval (CBIR) is ever increasing. While many methods have been suggested in the literature only few take into account the fact that - due to limited resources such as disk space and bandwidth - virtually all images are stored in compressed form. In order to process them for CBIR they first need to be uncompressed and the features calculated in the pixel domain. Often these features are stored alongside the images which seems counterintuitive to the original need for compression. The desire for techniques that operate directly in the compressed domain providing, so-called midstream content access, seems therefore evident [7].

Colour Visual Pattern Image Coding (CVPIC) is one of the first so-called 4-th criterion image compression algorithms [10, 9]. A 4-th criterion algorithm allows - in addition to the classic three image coding criteria of image quality, efficiency, and bitrate - the image data to be queried and processed directly in its compressed form, in other word

the image data is directly meaningful without the requirement of a decoding step. The data that is readily available in CVPIC compressed images is the colour information of each of the 4×4 blocks the image has been divided into, and information on the spatial characteristics of each block, in particular on whether a given block is identified as a uniform block (a block with no or little variation) or a pattern block (a block where an edge or gradient has been detected).

In this paper we make direct use of this information and propose an image retrieval algorithm similar to the colour coherence vectors introduced in [6] and the border/interior pixel approach in [12] which both show that dividing the pixels of an image into those that are part of a uniform area and those that are not can improve retrieval performance. In essence we create two colour histograms, one for uniform blocks and one for pattern blocks. Experimental results obtained from querying the UCID [11] dataset show that this approach not only allows retrieval directly in the compressed domain but also outperforms colour histogram, colour coherence vector and border/interior pixel techniques.

The rest of this paper is organised as follows: in Section 2 we give a brief introduction of image retrieval based on colour histograms, colour coherence vectors and border/interior pixel histograms. Also, we present the CVPIC compression algorithm used in this paper. Section 3 describes our novel method of image retrieval in the CVPIC domain while Section 4 presents experimental results. Section 5 concludes the paper.

2 Background

2.1 Colour-based image retrieval

2.1.1 Colour histograms

Swain and Ballard [13] were the first to introduce a useful object recognition algorithm based on colour features.

Surprisingly, their approach is very simple yet very powerful. They proposed to build a colour histogram of the image to describe its (colour) content. The histogram is built by (uniformly) quantising the colour space into a number of bins (often $16 \times 16 \times 16$) and counting how many pixels of the image fall into each bin. The colour histogram represents the index of the image; to compare two histograms and hence two images, they introduced (the complement of) histogram intersection defined as

$$d_{\text{HIS}}(I_1, I_2) = 1 - \sum_{k=1}^{N} \min(H_1(k), H_2(k)) \qquad (1)$$

where H_1 and H_2 are the colour histograms of images I_1 and I_2, and N is the number of bins used for representing the histogram. It can be shown that histogram intersection is equivalent to the L_1 norm and hence is a metric [13]. Histograms and histogram intersection are a successful and robust approach to image retrieval (in Swain's experiments the correct model was identified most of the time) and is thus at the heart of image search engines such as QBIC [5] and Virage [1].

2.1.2 Colour coherence vectors

Pass and Zabih [6] introduced colour coherence vectors as a method of introducing spatial information into the retrieval process. Colour coherence vectors consist of two histograms: one histogram of coherent and one of non-coherent pixels. Pixels are considered to be coherent if they are part of a continuous uniformly coloured area and the size of this area exceeds some threshold τ where τ is usually defined as 1% of the overall area of an image. The L_1 norm is used as the distance metric between two colour coherence vectors

$$d_{\text{CCV}}(I_1, I_2) =$$

$$\sum_{k=1}^{N} [|H_1^c(k) - H_2^c(k)| + |H_1^s(k) - H_2^s(k)|] \qquad (2)$$

where H_i^c and H_i^s and are the histograms of coherent and non-coherent (scattered) pixels respectively.

2.1.3 Border/interior pixel histograms

Stehling et al. [12] took a similar approach to that of coherence vectors. Pixels are classified as either interior or border pixels. A pixel is an interior pixel if (after a quantisation step) it has the same colour as its 4-neighbourhood, otherwise it is a border pixel. We see that in contrast to colour coherence vectors here the classification process is much simplified. Two histograms H^b and H^i for border and interior pixels are then created. Two images can be compared

by calculating the distance between those histograms as

$$d_{\text{BIP}}(I_1, I_2) =$$

$$\sum_{k=1}^{N} [|H_1^b(k) - H_2^b(k)| + |H_1^i(k) - H_2^i(k)|] \qquad (3)$$

2.2 Colour Visual Pattern Image Coding

The Colour Visual Pattern Image Coding (CVPIC) image compression algorithm introduced by Schaefer et al. [10] is an extension of the work by Chen and Bovic [2]. The underlying idea is that within a 4×4 image block only one discontinuity is visually perceptible.

CVPIC first performs a conversion to the CIEL*a*b* colour space [3] as a more appropriate image representation. As many other colour spaces, CIEL*a*b* comprises one luminance and two chrominance channels; CIEL*a*b* however, was designed to be a uniform representation, meaning that equal differences in the colour space correspond to equal perceptual differences. A quantitative measurement of these colour differences was defined using the Euclidian distance in the L*a*b* space and is given in ΔE units.

A set of 14 patterns of 4×4 pixels has been defined in [2]. All these patterns contain one edge at various orientations (vertical, horizontal, plus and minus $45°$) as can be seen in Figure 1 where + and - represent different intensities. In addition a uniform pattern where all intensities are equal is being used.

Figure 1. The 14 edge patterns used in CVPIC

The image is divided into 4x4 pixel blocks. Determining which visual pattern represents each block most accurately then follows. For each of the visual patterns the average L*a*b* values μ_+ and μ_- for the regions marked by + and - respectively (i.e. the mean values for the regions on each side of the pattern) are calculated according to

$$\mu_+ = \frac{\sum_{i \in +} p_i}{\sum_{i \in +} 1} \text{ and } \mu_- = \frac{\sum_{j \in -} p_j}{\sum_{j \in -} 1} \qquad (4)$$

where p_i and p_j represent the pixel vectors in L*a*b* colour space.

The colour difference of each actual pixel and the corresponding mean value is obtained and averaged over the

block according to

$$\epsilon = \frac{\sum_{i\in +} \|p_i - \mu_+\| + \sum_{j\in -} \|p_j - \mu_-\|}{16} \qquad (5)$$

The visual pattern leading to the lowest ϵ value (given in CIEL*a*b* ΔE units) is then chosen. In order to allow for the encoding of uniform blocks the average colour difference to the mean colour of the block is also determined according to

$$\sigma = \frac{\sum_{\forall i} \|p_i - \mu\|}{16} \text{ where } \mu = \frac{\sum_{\forall i} p_i}{16} \qquad (6)$$

A block is coded as uniform if either its variance in colour is very low, or if the resulting image quality will not suffer severely if coded as a uniform rather than as an edge block. To meet this requirement two thresholds are defined. The first threshold describes the upper bound for variations within a block, i.e. the average colour difference to the mean colour of the block. Every block with a variance below this value will be encoded as uniform. The second threshold is related to the difference between the average colour variation within a block and the average colour difference that would result if the block were coded as a pattern block (i.e. the lowest variance possible for an edge block) which is calculated by

$$\delta = \sigma - min_{\forall patterns}(\epsilon) \qquad (7)$$

If this difference is very low (or if the variance for a uniform pattern is below those of all edge patterns in which case σ is negative) coding the block as uniform will not introduce distortions much more perceptible than if the block is coded as a pattern block. Hence, a block is coded as a uniform block if at least one of the following criteria is met:

(i) $\sigma < 1.75$
(ii) $\delta < 1.25$

We adopted the values of 1.75 ΔE and 1.25 ΔE for the two thresholds from [10].

For each block, one bit is stored which states whether the block is uniform or a pattern block. In addition, for edge blocks an index identifying the visual pattern needs to be stored. Following this procedure results in a representation of each block as 5 bits ($1 + 4$ as we use 14 patterns) for an edge block and 1 bit for a uniform block describing the spatial component, and the full colour information for one or two colours (for uniform and pattern blocks respectively).

In contrast to [10] where each image is colour quantised individually, the colour components are quantised to 64 universally pre-defined colours (we adopted those of [8]). Each colour can hence be encoded using 6 bits. Therefore, in total a uniform block takes 7 ($= 1 + 6$) bits, whereas a pattern block is stored in 17 ($=5 + 2 * 6$) bits. We found that this yielded an average compression ratio of about 1:30.

We note, that the information could be further encoded to achieve lower bitrates. Both the pattern and the colour information could be entropy coded. In this paper however, we refrain from this step as we are primarily interested in a synthesis of coding and retrieval.

3 CVPIC uniform/non-uniform colour histograms

We note from above that the colour coherence vector (CCV) and border/interior pixel (BIP) approaches both calculate two types of histograms: one for pixels that belong to a uniform region and a second one for those pixels that do not fall into the first category.

In this paper we present an approach similar to those of CCV and BIP but exploit the information that is readily available in the CVPIC compressed format. We see that the encoded CVPIC information is directly visually meaningful. While on the one hand, the colour information is readily available, on the other hand information on the spatial content, i.e. shape-based information is also precalculated in the form of uniform and edge blocks.

The division into uniform and pattern blocks creates an automatic classification of image pixels. Pixels that are part of a uniform area (i.e. 'coherent' or 'interior' pixels) will more likely be contained within a uniform block. On the other hand pixels that form part of an edge (i.e. 'border' pixels) will fall into pattern blocks. We can therefore immediately distinguish between these two types of pixels without any further calculation (as would need to be done for CCV or BIP calculation). We hence create two colour histograms: a uniform histogram H^u by considering only uniform blocks and a non-uniform histogram H^n calculated solely from edge blocks. While exact histograms could be calculated by simply adding the appropriate number of pixels to the relevant colour bins while scanning through the image we suggest a simpler, less computationally intensive, method. Instead of weighing the histogram increments by the relative pixel proportions we simply increment the affected colour bins (two for an edge block, one for a uniform block) by 1^1. We also wish to point out that the resulting histograms are *not* normalised as is often the case with histogram based descriptors. The reason for this is that by not normalising we preserve the original ratio between uniform and pattern blocks - an image feature that should prove important for distinguishing between images with a similar colour content.

Having calculated H^u and H^n which can be done efficiently enough on-line, two images can then be compared

---

[1] We note that this puts more weight on the non-uniform histogram than on the uniform one.

by calculating the L_1 norm between their histograms

$$d_{\text{CVPIC}}(I_1, I_2) =$$

$$\sum_{k=1}^{N} [|H_1^u(k) - H_2^u(k)| + |H_1^n(k) - H_2^n(k)|] \quad (8)$$

4 Experimental Results

We evaluated our method using the recently released UCID dataset [11]. UCID, an Uncompressed Colour Image Database[2], consists of 1338 colour images all preserved in their uncompressed form which makes it ideal for the testing of compressed domain techniques. UCID also provides a ground truth of 262 assigned query images each with a number of predefined corresponding matches that an ideal image retrieval system would return.

We compressed the database using the CVPIC coding technique and performed image retrieval using the algorithm detailed in Section 3 based on the queries defined in the UCID set. As performance measure we use the modified average match percentile (AMP) from [11] and the retrieval effectiveness from [4]. The modified AMP is defined as

$$\text{MP}_Q = \frac{100}{S_Q} \sum_{i=1}^{S_Q} \frac{N - R_i}{N - i} \quad (9)$$

with $R_i < R_{i+1}$ and

$$\text{AMP} = \frac{1}{Q} \sum \text{MP}_Q \quad (10)$$

where R_i is the rank the i-th match to query image Q was returned, S_Q is the number of corresponding matches for Q, and N is the total number of images in the database. A perfect retrieval system would achieve an AMP of 100 whereas an AMP of 50 would mean the system performs as well as one that returns the images in a random order.

The retrieval effectiveness is given by

$$\text{RE}_Q = \frac{\sum_{i=1}^{S_Q} R_i}{\sum_{i=1}^{S_Q} I_i} \quad (11)$$

where R_i is the rank of the i-th matching image and I_i is the ideal rank of the i-th match (i.e. $I = \{1, 2, ..., S_Q\}$). The average retrieval effectiveness ARE is then taken as the mean of RE over all query images. An ideal CBIR algorithm would return an ARE of 1, the closer the ARE to that value (i.e. the lower the ARE) the better the algorithm.

In order to relate the results obtained we also implemented colour histogram based image retrieval ($8 \times 8 \times 8$

	AMP	ARE
Colour histograms	90.47	90.83
Colour coherence vectors	91.03	85.88
Border/interior pixel histograms	91.27	82.49
Colour correlograms	89.96	95.61
CVPIC uniform/non-uniform hist.	**93.28**	**62.40**

Table 1. Results obtained on the UCID dataset.

RGB histograms), colour coherence vectors, and the border/interior pixel approach. Results for all methods can be found in Table 1. From there we see that our novel approach is not only capable of achieving good retrieval performance, but that it actually clearly outperforms all other methods! While the border/interior pixel approach achieves an AMP of 91.27 and all other methods perform worse, CVPIC colour/shape histograms provide an average match percentile of 93.28, that is more than 2 higher than the best of the other methods. This is indeed a significant difference as a drop in match percentile of 2 will mean that 2% more of the whole image database need to be returned in order to find the images that are relevant; as typical image database nowadays can contain tens of thousands to hundreds of thousands images this would literally mean additionally thousands of images. The superiority of the CVPIC approach is especially remarkable so as it is based on images compressed to a medium compression ratio, i.e. images with a significantly lower image quality that uncompressed images whereas for all other methods the original uncompressed versions of the images were used. Furthermore, methods such as colour histograms, colour coherence vectors and colour correlograms are known to work fairly well for image retrieval and are hence among those techniques that are widely used in this field. An example of the difference in retrieval performance is illustrated in Figure 2 which shows one of the query images of the UCID database together with the five top ranked images returned by all method. Only the CVPIC techniques manages to retrieve four correct model images in the top five while colour correlograms retrieve three and all other methods only two.

5 Conclusions

In this paper we present a novel image retrieval technique that operates directly in the compressed domain of CVPIC compressed images. By exploiting the fact that CVPIC distinguishes between uniform and pattern blocks we are able to calculate histograms of pixels that fall in a uniformly coloured regions and those that do not. The resulting histograms are then simply compared by using the

Figure 2. Sample query together with 5 top ranked images returned by (from top to bottom) colour histograms, colour coherence vectors, border/interior pixel histograms, colour correlograms, CVPIC retrieval.

L_1 norm in order to perform image retrieval. Experimental results on a medium-sized colour image database show that the suggested method performs well, outperforming techniques such as colour histograms, colour coherence vectors, and border/interior pixel histograms.

6 Acknowledgements

References

[1] J. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, and R. Jain. The Virage image search engine: An open framework for image management. In *Storage and Retrieval for Image and Video Databases*, volume 2670 of *Proceedings of SPIE*, pages 76–87, 1996.

[2] D. Chen and A. Bovik. Visual pattern image coding. *IEEE Trans. Communications*, 38:2137–2146, 1990.

[3] CIE. *Colorimetry*. CIE Publications 15.2, Commission International de L'Eclairage, 2nd edition, 1986.

[4] C. Faloutsos, W. Equitz, M. Flickner, W. Niblack, D. Petkovic, and R. Barber. Efficient and effective querying by image content. *Journal of Intelligent Information Retrieval*, 3(3/4):231–262, 1994.

[5] W. Niblack, R. Barber, W. Equitz, M. Flickner, D. Glasman, D. Petkovic, and P. Yanker. The QBIC project: Querying images by content using color, texture and shape. In *Conf. on Storage and Retrieval for Image and Video Databases*, volume 1908 of *Proceedings of SPIE*, pages 173–187, 1993.

[6] G. Pass and R. Zabih. Histogram refinement for content-based image retrieval. In *3rd IEEE Workshop on Applications of Computer Vision*, pages 96–102, 1996.

[7] R. Picard. Content access for image/video coding: The fourth criterion. Technical Report 195, MIT Media Lab, 1994.

[8] G. Qiu. Colour image indexing using BTC. *IEEE Trans. Image Processing*, 12(1):93–101, 2003.

[9] G. Schaefer and G. Qiu. Midstream content access based on colour visual pattern coding. In *Storage and Retrieval for Image and Video Databases VIII*, volume 3972 of *Proceedings of SPIE*, pages 284–292, 2000.

[10] G. Schaefer, G. Qiu, and M. Luo. Visual pattern based colour image compression. In *Visual Communication and Image Processing 1999*, volume 3653 of *Proceedings of SPIE*, pages 989–997, 1999.

[11] G. Schaefer and M. Stich. UCID - An Uncompressed Colour Image Database. In *Storage and Retrieval Methods and Applications for Multimedia 2004*, volume 5307 of *Proceedings of SPIE*, pages 472–480, 2004.

[12] R. Stehling, M. Nascimento, and A. Falcao. A compact and efficient image retrieval approach based on border/interior pixel classification. In *Proc. 11th Int. Conf. on Information and Knowledge Management*, pages 102–109, 2002.

[13] M. Swain and D. Ballard. Color indexing. *Int. Journal Computer Vision*, 7(11):11–32, 1991.

Toward semantics level indexing and retrieval of images and video

Jian Kang Wu, Joohwee Lim and Dezhong Hong
Institute for Infocomm Research

Abstract

Users'indexing and retrieval of any information is always at semantic level. On other hand, existing technologies for indexing and retrieval of visual information are at "feature" level: the feature measures (such as color and texture) are extracted, self-organizing and retrieval are performed based on those feature measures. This paper reports two research attempts, which support indexing and retrieval of visual information at semantic level. One is to categorize visual objects to create conceptual categories. Via learning capabilities, user can build his/her own categories for storage, navigation and retrieval of his/her images and video. The other is to achieve and retrieve images and video using "visual keywords". "Visual keywords" are something like keyword and n-grams in text retrieval. Based on that, vector space can be created, semantic level indexing and retrieval is facilitated.

1. Introduction

Indexing and retrieval of visual content such as images and video has been at feature level for many years. Unfortunately, there has been always a demand from users' side for semantic level access to images and video. Here, we present our attempts using object cataloguing and visual keyword. Object cataloguing is a natural means to add semantics to objects within images and video. Here, the challenges are robustness during the cataloguing process, and the user's interaction when the catalogues are defined. Visual keyword is an analogue to vector space retrieval method in text retrieval domain. The challenge here is to derive visual "keyword" for a given class of images and video. In the following two sections, we will discuss these two methods and show experimental results.

2. Visual Keyword

We have developed a novel methodology to index and retrieve digital images. *Visual keywords* are intuitive and flexible visual prototypes extracted from a visual content domain with relevant semantics labels. An image is indexed as a spatial distribution of visual keywords.

Visual keywords can be further abstracted to form equivalence classes of visual synonyms. This concept-oriented visual thesaurus is different from the visual relations proposed by R.W.Picard [Pica95], which are founded on similarities between low-level visual features. Moreover, spatial localities of visual keywords are not discovered through segmentation, but aggregated by multi-scale view-based detection. We have obtained very promising results in retrieval and classification of natural scene stock photographs and family photographs. More details can be found in [Lim99a,Lim99b,Lim99c].

2.1 Methodology
2.1 Methodology

The indexing process has 4 key components. First a visual vocabulary and thesaurus is constructed from samples of a visual content domain. Then an image to be indexed is compared against the visual vocabulary to detect visual keywords automatically. Thirdly, the fuzzy detection results are registered in a *Type Evaluation Map* (TEM) and further aggregated spatially into a *Spatial Aggregation Map* (SAM). Last but not least, with visual thesaurus, the SAM can be further abstracted and reduced to a simpler representation, *Concept Aggregation Map* (CAM).

Visual keywords are visual prototypes specified by a user. Using an appropriate visual tool, the user crops domain-relevant regions from sample images and assigns sub-labels and labels to form vocabulary and thesaurus respectively. Suitable visual features (e.g. color, texture) are computed for each cropped region into a feature vector. i.e.

$$c_i : (s_{i1}, v_{i1}), (s_{i2}, v_{i2}), \ldots, (s_{ij}, v_{ij}) \ldots$$

where c_i are concept labels, s_{ij} are sub-labels for specific instances of concept i, and v_{ij} are feature vectors for regions ij. For instance, the sub-labels for the *sky* (label) visual keywords could be cloudy,

blue, bright, etc.

An image to be indexed is scanned with windows of different scales. Each scanned window is a visual token reduced to a feature vector t compatible (i.e. same feature types and dimension) to those of the visual keywords v_{ij} in previously constructed.

More precisely, given an image I with resolution M x N, a TEM T has a lower resolution of P x Q, $P \leq M$, $Q \leq N$. Each pixel $T(p,q)$ corresponds to a two-dimensional region of size w_x x w_y in I. Within this region, visual tokens corresponding to several scales k are extracted into feature vectors t_k and compared against the visual keywords v_{ij} to compute the fuzzy membership vectors $\mu_k\ (t_k, v_{ij})$. The most confident scale (i.e. $z = argmax_k\ max_{ij}\ \mu_k\ (t_k, v_{ij})$) is taken as the final candidate for detection,
$T(p,q,i,j) = \mu_z(t_z, v_{ij})$.

Likewise, SAM S tessellates over TEM with A x B, $A \leq P$, $B \leq Q$ pixels. Each SAM pixel (a,b) aggregates the fuzzy memberships for visual keyword ij over those TEM pixels (p,q) covered by (a,b),
$S(a,b,i,j) = \sum_{(p,q) \in (a,b)}\ T(p,q,i,j)$.

As the sub-label s_{ij} of a visual keyword v_{ij} describes a specific appearance of the concept labelled by c_i, they are visual synonyms that allow abstraction by further aggregation over visual keywords sharing identical concept labels,
$C(a,b,i) = \sum_j S(a,b,i,j)$.
This is useful when more general semantical concepts take precedence over specificity captured in different visual keywords. This point will be demonstrated in our experimental results.

The similarity matching adopted in our experiments for two images x,y is the city block distance, which performs better than other measures (e.g. Euclidean distance, cosine) experimented,
$s(x,y) = \sum_{(a,b)} \sum_{i,j} | S^x(a,b,i,j) - S^y(a,b,i,j) |$, or
$s(x,y) = \sum_{(a,b)} \sum_i | C^x(a,b,i) - C^y(a,b,i) |$.

2.2 Experimental Results

In this paper, we report retrieval results on natural scene photographs. Seven semantical categories of 111 visual keywords are cropped from samples of 500 professional photographs.

The categories (and quantitites) are: sky (10), water (20), mountain/beach (12), snowy mountain (10), field (26), tree (24), and shadow/dont-care (9).

Fig. 1-4 show the appearances of some visual keywords for selected categories. Each visual keyword has a semantical label and sub-label.

Fig. 1. Selected visual keywords for *sky*

Fig. 2. Selected visual keywords for *water*

Fig.3. Selected visual keywords for *trees*

Fig.4. Selected visual keywords for *snowy mountains*

The images are size-normalized to 256x256. Three scales (31x31, 41x41, 51x51) are adopted for visual token scanning. The features used to characterize a visual keyword and token are based on color and texture. For color, each region is down-sampled into 4x4 YIQ channels, which work better than simple means and local color histogram in our experiments. For texture, we adopted Gabor filters-based features [MaMa96] with 5 scales and 6 orientations. Both TEM and SAM have 6x6 resolutions.

In our experiments, retrieval based on visual keywords and thesaurus outperforms methods that rely on global histograms of color and texture significantly. As the global measures of color and texture do not care about semantics at spatial locations, the retrieved images could be far from

expectation. For example, in Fig. 5, the query image is a snowy mountain (top left) and only retrieved images (ranked in decreasing similarities from top to bottom, left to right) 3, 5, 6, 9, and 13 are of the same class. For the same query, the visual keyword approach has returned all except 14 (stream below mountains) relevant (i.e. snowy mountain) images (Fig. 6).

In the same token, only retrieved images 5, 10, 11 are trees (top-left image as query). In fact, the query contains coconut trees which are rare in our test data. Nevertheless, using visual thesaurus which only considers the abstract concept of trees, images (2-6,10-11,13-14) which exhibit different types (color, texture, shape) of trees are returned.

Fig. 5. Retrieving *Snowy Mountains* by global colors and textures

Fig. 6. Retrieving *Snowy Mountains* by visual keywords

3. Model-based object cataloguing and retrieval

The basic unit to be indexed or retrieved for images or videos is either an image or an image with highlighted objects. The acquisition of image contents at the semantic level is the main goal for this approach. A brief introduction will be given to object modeling, system training, and image cataloguing and retrieval.

3.1 Object modeling

The objective of the object model design is to extract effective features to distinguish and describe interested objects. Since home photo is the focus of our research domain, natural scene and people in images are taken as the majority of interests from users. Therefore, in this paper, two object models are proposed to extract and describe image contents:

Textured model: It is mainly for the description of natural objects, such as sky, tree, water, field, mountain, building, etc. It is mainly composed of three feature components: LUV color component, texture energy component, and texture direction component. The original image is initially segmented into homogeneous regions (based on the LUV color component) by means of the MAP segmentation algorithm. The homogeneous regions are treated as the basic visual units that are represented by a feature vector consisting of the three feature components.

Face model: It is mainly for the detection and recognition of human faces presenting in images. The hypothetical face regions are firstly segmented by means of skin-color modeling and prior shape knowledge of faces. They are further verified by a set of face templates that can be customized during the processing. The PCA-based method is finally employed to label the identity on each detected face.

We introduce the concept of *object category*. An object category is defined as a collection of the objects that possess similar properties and are described by the same object models. The object model assigned to each object category is employed to compute the features for each member object and separate them from each other.

3.2 System training

We noticed that with one object, (e.g. sky), multiple appearances of the object may be associated (e.g. clear sky, cloudy sky, sunset sky, etc.). Therefore, we have considered that, the system should be designed to accommodate a large number of object categories that can be customized by the users based on their application requirements, and support the multiple appearances of the objects.

To incorporate the human knowledge about interested objects into the object models, a system training scheme is addressed. In this approach,

we try to minimize the users' interactions by employing the normalized fuzzy C-means clustering algorithm.

Assume the distance of two feature vectors v_a and v_b on the first feature component f is expressed as $D_f(v_a, v_b)$. The cluster computation is applied to all the training data sets and it is carried on iterately. The cluster center for a data set X_i is computed by:

$$v_i = \left(\sum_{k=1}^{n_i} (u_{ik})^2 x_{ik} \right) \bigg/ \left(\sum_{k=1}^{n_i} (u_{ik})^2 \right) \qquad (2.1)$$

The fuzzy membership of the data members in the data set X_i can be computer by:

$$u_{ik} = \frac{C}{C + \sum_{j=1}^{C} \left(D_f(x_{ik}, v_i) / D_f(x_{ik}, v_j) \right)^2} \qquad (2.2)$$

where v_i is the cluster center of X_i, and v_j represents the cluster centers of other clusters at the current iteration step. Please note that (2.2) is normalize by the current cluster number C. The new cluster center is then updated by (2.1). The iteration of (2.1) and (2.2) is carried on until:

$$\max_{i,k} \{ | u_{ik} - u_{ik}^* | \} \leq \varepsilon \qquad (2.3)$$

A threshold T_u is set to evaluate the data member in each data cluster. If:

$$u_{ik} < T_u \qquad (2.4)$$

the data member x_{ik} will be taken out of the current object cluster to compose a new cluster under the same object category. The iteration of (2.1), (2.2), (2.3) and (2.4) is performed until converging (No new shifting of data member happens between clusters). Finally, a number of clusters are formed for the object definitions of the object categories. Each cluster represents a specific appearance of an object.

3.3 Image cataloguing and retrieval
Filing images into semantic categories can be very useful in browsing and retrieving images in the database. Because of the complexity and the vast amount of data, manual content indexing is very time consuming. A technique for automatic cataloguing and indexing of images is, therefore, proposed in this approach.

We introduce the concept of *image store* that can be used to hold a big collection of images with any common features, semantic contents, or higher level event descriptions. Its creation is based on one or multiple object categories that contain a number of object definitions. While filing images into the image store, the images are semantically segmented and labeled based on the object clusters in the object category. Based on the semantic image contents, the images are catalogued under related object titles. They are also indexed according to the significance of the objects contained in the images.

The system also supports searching for semantics. Three types of image queries are:

Query by visual objects: The related ranking number is set at the first column. The matching number for the column I can be obtained by:

$$\mu_{mI} = \left[\sum_{j}^{J} \mu_{Qj} \left(\sum_{k}^{K} \max_{h}^{H} \mu_{Ojkh} \right) \bigg/ K \right] \bigg/ \sum_{j}^{J} \mu_{Qj} \qquad (2.5)$$

where μ_{Ojkh} represents the matchness for the object k at j query, and H indicates the total number of the query object k in the analysed image. μ_{Qj} is denoted as the query state:

$$\mu_{Qj} = \min \left(\mu_{Tj}, \mu_{Lj} \right) \qquad (2.6)$$

where μ_{Tj} and μ_{Lj} are the fuzzy membership related to the query time and query levels respectively.

Query by images: The related ranking number is set at the second column. The matching number can

$$\mu_{mII} = \left[\sum_{j}^{J} \mu_{Qj} \left(\sum_{k}^{K} \mu_{Sjk} \max_{h}^{H} \mu_{Ojkh} \right) \bigg/ \sum_{k}^{K} \mu_{Sjk} \right] \bigg/ \sum_{j}^{J} \mu_{Qj}$$

be computed by:

$$(2.7)$$

where μ_{Sjk} represents the significance of the object in the image.

Query by semantic descriptions: The related ranking number is set at the third column. The computation for the ranking number at this colum n is made by:

$$\mu_{mIII} = \left[\sum_{j}^{J} \mu_{Qj} \left(\sum_{k}^{K} \max_{h}^{H} \mu_{Sjkh} \right) \bigg/ K \right] \bigg/ \sum_{j}^{J} \mu_{Qj}$$

$$(2.8)$$

where μ_{Sjkh} represents the significance of the object to be analyzed in the image.

By combining these three query methods, a ranking table can, therefore, be created automatically for the images according to the ranking numbers computed based on the query methods at the present query state, as shown in Table 2.1. The matching numbers in the first column are, of course, given the highest priority, followed by the second column and the third column. The more details the user specifies for the desired images, the more relevant the images obtained are to match user's requirements.

Table 2.1. The ranking table for image retrieval.

Images	Ranking numbers		
	First column	Second column	Third column
Image 1	μ_{mI}^1	μ_{mII}^1	μ_{mIII}^1
Image 2	μ_{mI}^2	μ_{mII}^2	μ_{mIII}^2
…	…	…	…

3.4 Experimental results

In the experiments, six natural objects are defined: *sky, tree, water, field, mountain,* and *building*. There are 2400 photos in the database. To train the system, we select 160 photos. In Table 2.2, we list the statistic figures and the clusters generated on each object. The number of the clusters generated under each object title mainly depends on the variation of the appearances of the object and the similarity to other categorised objects.

Table 2.2. The statistics for system training.

Objects	Training data	Number of clusters
Tree	97	23
Sky	82	16
Water	69	31
Field	74	38
Mountain	48	19
Building	83	44

To determine the effectiveness of the algorithms, we randomly select 160 images in the database. The results of the precision and recall for the six objects defined in this experiments, as well as a special object "people" defined based on the face model, are listed in Table 2.3.

Table 2.3. The precision and recall for the objects.

Objects	Precision	Recall
Tree	0.74	0.84
Sky	0.69	0.91
Water	0.63	0.73
Field	0.48	0.55
Mountain	0.69	0.77
Building	0.56	0.70
People	0.63	0.69

The images are automatically catalogued into seven categories. They are indexed in each category according to the relevant size of the corresponding object presented in the images. The first pages of images under three different categories are displayed in Fig. 2.1, 2.2, 2.3.

Fig. 2.1 Cataloguing results of "tree".

Fig. 2.2 Cataloguing results of "water".

Fig. 2.3 Cataloguing results of "people".

We take an example to show the process of the image retrieval and how the system refines the retrieval results. When a query is made by a semantic description "building", 46 images out of the 50 first ranked images retrieved contain "buildings". We display the first 6 images in Fig. 2.4. Then a query is made by submitting the fourth image in Fig. 2.4, the first 6 retrieved images are presented in Fig. 2.5. Among the first 50 retrieved images, 31 images contain *building*, *sky*, *water* and *tree*. And then another query is further made by specifying the building in the middle of the first image in Fig. 2.5. The first 6 retrieved images are displayed in Fig. 2.6. Among the first 50 retrieved images, 28 images contain the buildings that are visually similar to the one specified.

Fig. 2.4. Query results by "Building".

Fig. 2.5. Query results by an image.

Fig. 2.6. Query results by a specified visual object.

4. Conclusion and remarks

We have presented two methods to achieve semantic indexing and retrieval of images and video. Very good results have been obtained in the indexing and retrieval of digital home photos. Future work is to develop a complete technological framework so that the methods can be generalized to other applications.

References

[Lim99b] Lim, J.H. (1999). Visual keywords: from text IR to multimedia IR. In *Soft Computing in Information Retrieval: Techniques and Applications* (F.Crestani & G.Pasi ed.), Physica-Verlag, Springer Verlag, Germany, 1999.

[Lim99c] Lim, J.H. (1999). Learnable Visual Keywords for Image Classification. In *Proc. of ACM Digital Libraries'99, Berkeley, CA, USA, Aug. 11-14, 1999*, pp. 139-145. ACM Press.

[MaMa96] Manjunath, B.S., and Ma, W.Y. (1996). Texture Features for browsing and retrieval of image data. *IEEE Trans. PAMI, 18, 8*, pp. 837-842, 1996.

[Pica95] Picard, R.W. (1995). Toward a visual thesaurus. In *Proc. of Springer-Verlag Workshops in Computing, MIRO'95, Glasgow, Sep. 1995*.

Proceedings

The 2004 International Workshop on Mobile Systems, E-commerce and Agent Technology (MSEAT'2004)

Editor

Anthony Y. Chang, Oversea Chinese Inst. Of Tech., Taiwan

Routing-Profitable MAC Protocol for Mobile Ad Hoc Networks

San-Yuan Wang+, Lain-Chyr Hwang#, Chia-Hsu Kuo* and Tzu-Chiang Ma+

+ Department of Information Engineering, # Department of Electrical Engineering,

* Department of Communication Engineering, I-Shou University, Taiwan

{sywang, Lain, kuoch}@isu.edu.tw

Abstract

Due to the rapidly deployment and flexible configuration characteristics, Mobile Ad Hoc Networks (MANETs) have many applications in civilian environment, law enforcement and military arena. Among these applications, data packets have to dissemination from nodes to nodes hop by hop. The crucial operation of MANET is routing. In this paper, we investigate the probability to enhance the performance of routing protocol by modifying the MAC protocol. The basic idea is increasing the transmission probability for those nodes want to relay data. Through exhaust simulation, our navel MAC protocol enhance the on-demand routing protocols than that of the IEEE 802.11 DCF in various system parameters.

1. Introduction

A Mobile Ad Hoc networks (MANETs) is a complex autonomous system comprising wireless mobile nodes (act as an router) connected by wireless links, which can dynamically be organized into arbitrary topology. Due to the rapidly deployment and flexible configuration characteristics, MANETs have many applications in civilian environment, law enforcement and military arena. Among these applications, data packets have to dissemination from nodes to nodes hop by hop. The crucial operation of MANET is routing. The IEEE 802.11x MAC protocol [6], the de facto standard for wireless LANs, is carrier sense multiple access with collision avoidance (CSMA/CA) using binary exponential backoff (BEB).

Recently, a large number of random access MAC protocols, such as MACA [1,10,12,13], MACAW [2,15], FAMA[4,16], have been proposed to resolve the multi-access problem over wireless channel. Some papers [3,5] investigate the interactive between TCP and MAC. Authors in [5] indicate that the IEEE 802.11 DCF favor those nodes just complete transmission and eventually causes unfairness media access. The researchers [9] study the characteristics of the interaction between routing and MAC under different mobility model. The authors in [17] proposed an enhanced Routing-aware adaptive MAC with traffic differentiation and smoothed contention widow in MANETs.

In this paper, we investigate the probability to enhance the performance of routing protocol by modifying the BEB mechanism. The basic idea is increasing the transmission probability for those nodes want to relay data. Through exhaust simulation, our navel MAC protocol enhance the on-demand routing protocols than that of the IEEE 802.11 DCF in various system parameters.

2. Routing-Profitable MAC Description

The paper [5] indicates that the CSMA/CA with binary exponential backoff favors those nodes just completed transmission in MANETs. Eventually, while TCP traffic from source to destination, links have different throughput. Since routing is a critical communication operation in MANETs, we

investigate the possibility to enhance the routing performance by modifying MAC protocol and, eventually, improve the performance of the whole MANET.

The basic idea is to increase the channel capture chance of those nodes want to relay data. We modify the backoff mechanism as follows. Initially, the backoff time is

$$backofftime = \begin{cases} \left\lfloor 2^2 \times random() \right\rfloor & for\ nodes\ want\ to\ relay\ data \\ \left\lfloor 2^2 \times random() + 2^2 \right\rfloor & otherwise \\ 2^3 & for\ nodes\ just\ finished\ transfer \end{cases}.$$

When the node encounters collision c times, the backoff time is

$$backofftime = \begin{cases} \left\lfloor 2^{2+c} \times random() \right\rfloor & for\ nodes\ want\ to\ relay\ data \\ \left\lfloor 2^{2+c} \times random() + 2^{2+c} \right\rfloor & othrwise \end{cases}.$$

Fig. 1 illustrates backoff times of various nodes.

Fig.1: backoff time for various nodes

3. Simulation Configuration and Results

In this paper, we investigate the IP/MAC layer interactive via simulation. The simulation platform used is Network Simulator-2[18]. It includes several wireless protocols in its library (radio, propagation, MAC, network, transport and applications). We first study how our new MAC affects the inactive and proactive routing protocols. Then we evaluate the routing performance under various system situations including node density, mobility, and traffic load.

3.1 Effects on inactive and proactive routing protocols

We consider the grid topology with up to 25 nodes as

shown in Fig.2. Each node and its neighbor stands 200m apart. Nodes are static. Eight source destination pairs simultaneously transmit 8 packets. When using DSDV, the goodput of IEEE 802.11 outperforms ours. However, as running DSR, the average end-to-end delay of ours is better than BEB scheme as shown in Table 1. The reason is stated as follows. DSDV needs to maintain routing table. Our scheme is prefer those node want to relay data.

Fig. 2: Grid topology

Table 1:DSR complete time and DSDV goodput

Node		DSR Complete Time		Node		DSDVgoodput			
S	D	BEB	Ma2	S	D	BEB	Ma4N	Ma5N	Ma6N
1	7	1.59	3.34	1	7	10440	10440	10440	10440
14	17	7.65	1.70	1	17	11520	80	40	40
16	12	2.19	4.47	1	12	9400	9400	9400	9400
2	21	8.66	7.89	2	21	12680	0	200	40
22	4	1.99	4.15	2	4	13560	0	0	0
5	17	2.54	3.29	5	17	40	0	0	0
8	23	8.27	2.18	8	23	10440	0	0	0
average		4.70	3.86	total		68080	19920	20080	19920

In the following, we focus on the effects of node density, mobility and traffic load on the on-demand routing protocol (DSR).

3.2 Effects of Traffic Load

In the 700m*700m area, we randomly deploy 30 nodes. Nodes are static. Randomly choose 11, 13, 15 and 17 source-destination pairs and evaluate the end-to-end delay. As shown in Fig. 3, our schemes outperform BEB as the number of connections increasing.

Fig.3: End-to-end delay of various number of connections

3.3 Effects of Node Density

In this simulation, number of connections is fixed on 7. As shown in Fig. 4, node density is larger, the delay is longer. Our schemes are better than BEB.

Fig.4: Effects of node density.

3.4 Effects of Mobility

In this simulation, we randomly deploy 50 nodes in 700m *700m area. The number of connections is 7. Node mobility speed is randomly up to 40m/sec. We investigate the performance on various percentage of mobility. The notation 26%C means there are 13 of 50 nodes in mobile state with a randomly speed up to 40 m/sec. As shown in Fig.5, our schemes perform better in less mobility situation.

Fig. 5: Effects of mobility

4. Conclusion

In this paper, we propose a novel MAC protocol which can enhance the performance of on-demand routing protocols of MANET. The basic idea is to modify the back-off mechanism of IEEE 802.11 for increasing the transfer probability of nodes which want to relay data. Through exhaust simulation on NS2, our back-off mechanism outperforms Binary exponential back-off of IEEE 802.11 on various system situations.

5. References

1. Acharya, A.; Misra, A.; Bansal, S.; "MACA-P : A MAC for Concurrent Transmissions in Multi-hop Wireless Networks." Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. Page(s): 505 –508

2. Vaduvur Bharghavan, Alan Demers, Scott Shenker, Lixia Zhang. "MACAW: A Media Access Protocol for Wireless LAN's" ACM SIGCOMM Computer Communication Review , Proceedings of the conference on Communications architectures, protocols and applications, October 1994.

3. Mario Geria, Ken Tang, Rajive Bagrodia, "TCP Performance in Wireless Multi-hop Networks" Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on , 25-26 Feb. 1999 Page(s): 41 –50

4. Chane L. Fullmer and J.J. Garcia-Luna-Aceves, "Floor Acquisition Multiple Access (FAMA) for packet radio networks", Computer Communication Review, vol. 25, (no. 4), (ACM S1GCOMM '95, Cambridge, MA, USA, 28 Aug.-l Sept. 1995.) ACM, Oct. 1995

5. Shugong Xu and Tarek Saadawi, "Does the IEEE 802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Network?" IEEE Communications Magazine * June 2001

6. IEEE, "802.11 Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) specifications", November 1997.

7. N.H. Vaidya, P. Bahl, and S. Gupta, "Distributed Fair Scheduling in Wireless LAN", Proc. Mobicom'2000, Boston, MA, USA, Aug. 2000

8. Arup Acharya, Archan Misra, Sorav Bansal. "Mobile Ad Hoc Networks: A label-switching packet forwarding architecture for multi-hop wireless LANs", the 5th ACM international workshop on Wireless mobile multimedia Proceedings, September 2002.

9. Chris Barrett, Achla Marathe, Madhav V. Marathe, Martin Drozda. "Characterizing the interaction between routing and MAC protocols in ad-hoc networks", Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing, June 2002.

10. Matsuno, H.; Ishinaka, H.; Hamanaga, A.; "A simple modification for the drastic improvement of MACA in large propagation delay situation", IEEE Wireless Communications and Networking Conference, 2000. WCNC. 2000 , Volume: 2 , 23-28 Sept. 2000, Pages:865 - 869 vol.2

11. Kantawong, S.; Sivamok, N.; Wuttisittikulkij, L.; "Investigation of contention resolution algorithms for wireless communication systems", IEEE Vehicular Technology Conference, 2001. VTC 2001 Fall. VTS 54th , Volume: 1 , 2001 Pages:222 - 226 vol.1

12. Xiaoxin Wu; Mukherjee, B.; Chan, S.-H.G.; "MACA-an efficient channel allocation scheme

in cellular networks", IEEE Global Telecommunications Conference, 2000. GLOBECOM '00., Volume: 3 , 27 Nov.-1 Dec. 2000 Pages:1385 - 1389 vol.3

13. Talucci, F.; Gerla, M.; "MACA-BI (MACA by invitation). A wireless MAC protocol for high speed ad hoc networking", Universal Personal Communications Record, 1997. Conference Record., 1997 IEEE 6th International Conference on ,Volume: 2 ,12-16 Oct. 1997 Pages:913 - 917 vol.2

14. Tickoo, O.; Sikdar, B.; "Modeling and analysis of traffic characteristics in IEEE 802.11 MAC based networks", IEEE Global Telecommunications Conference. Volume: 1 , 17-21 Nov. 2002 Pages:67 - 71 vol.1

15. Lundy, G.M.; Almquist, M. and Oruk, T.; "Specification, verification and simulation of a wireless LAN protocol: MACAW", IEEE Military Communications Conference, 1998. Volume: 2 , 18-21 Oct. 1998 Pages:565 - 569 vol.2

16. Garcia-Luna-Aceves, J.J.; Fullmer, C.L.; "Performance of floor acquisition multiple access in ad-hoc networks", Computers and Communications, 1998. ISCC '98. Proceedings. Third IEEE Symposium on ,30 June-2 July 1998 Pages:63 – 68

17. Farid Nait-Abdesselam and Head Koubaa; "Enhance Routing-Aware Adaptive MAC with Traffic Differentiation and Smoothed Contention Window in Wireless Ad Hoc Networks", Proceedings of the 24th international conference on Distributed Computing systems Workshops (ICDCSW'04).

18. NS2. Network Simulator. http://www.isi.edu/ nsnam.

A Framework of Online Chain Store Integrated with Recommendation Mechanism

Jason C. Hung

Dept. of Information Management
Kuang Wu Institute of Technology
Peitou, Taipei, Taiwan 112, R.O.C.
E-mail: jhung@cs.tku.edu.tw

Schummi Yang

Dept. of CS&IE
Tamkang University
Tamsui, Taipei, 251, Taiwan, ROC
E-mail: yang@schummi.com

Abstract

With the rapid growth of the Internet and communication technologies, many real-world activities were applied on the Internet, including education, entertainment, academic activities, and commerce, etc. We proposed the nomadic media server (mobile media server) to achieve the maximum use of network bandwidth. The distributed load of central server will be retrenched appropriately. At the same time, the security protection degree will be strengthened via the changeable server. Actually, we use the mobile server/storage pre-broadcasting technique, as well as a communication network optimization algorithm, which is based on a graph compute mechanism. The mobile server will play an important role in the future virtual society, both on narrow band and broadband environment. Chain store over the Internet is a rare management model in electronic commerce. In this paper, we construct many chain stores via using nomadic media server.

Besides, with the advent of the World Wide Web, online merchants must know what customers want and make them buying something from their sites, so recommendation process becomes an important strategy for the merchants. In this paper, we analyze customers' behavior and interests, and recommend something useful to them based on the correlation among customers, product items and product features. And we propose a recommendation system for the e-commerce portal/site, which will help merchants to make suitable business decision and delivery personalized information to the customers.

Keywords: Nomadic Media Server, Virtual Society, Broadband Communication, Chain Store, Electronic Commerce, Recommendation.

1. Introduction

In recent years, electronic commerce brings some very hot research issues. These issues include security, recommendation system, payment mechanism, and so on. According to the idea of our mobile media server, we proposed a new trend in electronic commerce. We still not found any chain store over the Internet with recommendation because there is no good mechanism to support the environment. Our mobile server provides the solution for this idea.

As we know, the Internet is very popular in this decade. Many activities were applied on the Internet already. Commercial trade seems very exciting with this medium because it provides a more convenient method for business. Of course, in the flow of business, a complete trade includes cost assessment, payment evaluation, channel management …, etc. And there are many roles in the commerce, including business, provider, consumer, agent, and so on. So, they perform a supply chain. Actually, these components still exist in the Electronic Commerce (EC) over the Internet.

EC describes the manner in which transactions take place over networks, mostly the Internet. It is the process of electronically buying and selling goods, services, and information. There are many benefits to manage an electronic store over the Internet. Managers can decrease their cost, response time and make more profit. It is possible for consumer to be a manager since the benefits of managing an electronic store.

However, information overload is a critical problem, especially for the applications in the e-commerce arena. Moreover, product brokerage is essential in e-commerce. In traditional business behavior, customers get recommendations or suggestions from clerks or other people, on the contrary, customers buying something by themselves without others' assistance in the Internet. So recommendation system has become an important application area and academic research topic of the e–commerce.

2. Related Works

Generally speaking, there are many business models over the Internet. Some researches [1,2] mentioned the illustration of business model.

Some papers discussed the mobile environment. Actually, Mobile Peer to Peer (MP2P) [3] takes a historical view on the subject of mobile networks and underscores some of the main architectural principles enabling MP2P. This concept is further expanded to incorporate a mobile architecture including a ubiquitous end user device, where business model related policies and stored. In order to archive the high utilization of distributed resource, the Mobile Distributed Web Server System [4] provides a good solution.

Recommendation system has been introduced by Resnick and Varian in [5]. The idea is that users cannot

make a choice without sufficient personal experience or others' suggestion, thus, automated recommendation by the computer system would be very helpful in filtering product information. In this paper, we also focus on the recommendation part of the e-commerce applications.

Balabanovic and Shoham proposed a recommendation system in Feb. 1997 [6]. They combined content-based filtering and collaborative filtering to design their system. Content-based recommendation recommends product items to the customers who have purchase records or web-browsing activities records. On the other hand, collaborative recommendation classifies users into groups where users in the same group are of similar behavior or interests on the specific product items.

3. Traditional Chain Store

In the real world, there are many chain stores on the street, such as 7-11, starbucks coffee shop, McDonald's …, etc. The branch store joins an alliance to the original company and pays the fees of patent. This is another form of business to business.

The branch store serves consumers and owns their store like the original store. The outward appearance is always very similar to other branch stores. These branches then form chain stores. Their services are always similar, too. Since this is a kind of management in the real world, it should be able to be completely imitated on the Internet. If this way is available, there are many guaranteed benefits as following:

   (1) Load Sharing
   (2) Heighten Security
   (3) Full usefulness of Resource

Load Sharing means that the load of the original store (headquarter) could be shared by other branch stores. As to the security issue, many hackers are always interested in destroying a web site. If we have a mechanism about mobile server for distributing the content to other sites, the security will be heightened and guaranteed. In aspect of Full usefulness of resource, the same host can open different stores because it can become different branch stores after closing a store.

The simple model of chain store is adapted to both the real world and the virtual world. In the virtual world, the idea can be realized by our mobile media server strategy, which we will discuss in next section.

The role transition may occur at any time. The consumer can become as provider such as branch store while he registers on the original store. Then, the original store will transfer the content to the client for remotely building a server via our mobile media server technique.

4. Nomadic Media Server

We need the mobile media server provider to provide the service of content. Then, through the transferring medium, including wired media, or wireless ones, even satellite to transfer the content to the received host. After the host received the command from the mobile media server provider, it will be set as a server to serve all users over the Internet. This is illustrated as figure 1.

In the mobile environment, we propose a simple protocol to support the idea. In fact, the migrated data is not limited to only files, but also all continuous media. According to the simple protocol, we have some steps between the mobile media server provider and mobile server.

I. Mobile server end request for service of content.
II. Mobile Media Server Provider receives the request, then searches the goal and detect the hardware and software configuration.
III. Wait for the condition complete.
IV. After the condition complete, a mobile agent is sent to the mobile server end.
V. Mobile server opens the right to let the agent to write.
VI. Setting as server and migrating the content.

This protocol is illustrated as figure 2.

Figure 1: The Illustration of Mobile Server

Figure 2: The Migration Protocol

4.1 The Lifecycle of Nomadic (Mobile) Media Server

Actually, a mobile server/storage has its life. The mobile server/storage always follows the directions to any possible next states after it is born and before it is killed or died of natural. Figure 3 illustrates the following states.

♦ Start: the mobile server/storage was created.
♦ Searching: the mobile server/storage is searching the identified host.
♦ Suspending: the agent which is embedded in the

mobile server/storage is waiting for enough resource in order to enter the host.

♦ Dangling: the agent which is embedded in the mobile server/storage loses its goal of surviving, it is waiting for a new goal to mutate.
♦ Mutating: the mobile server/storage is changed to another host.
♦ Migrating: the contents of mobile server/storage were transferred.
♦ Restoring: the mobile server/storage will die or mutate, it restores its related data to the original host.
♦ Dying: the agent which is embedded in the mobile server/storage is died and the mobile server is killed.

Figure 3: Mobile Server/Storage Living State Diagram

4.2 The Algorithms for Mobile Server Living State

In the previous section, the living state and simple protocol of mobile server/storage were proposed. In order to justify the description, we provide algorithms to prove it. We have a global variable named *state*, to record the living state of mobile server. The developed algorithms are adapted to the migration protocol illustrated in figure 2.

The Searching algorithm describes that if a host request for service from mobile media server provider, then mobile media server provider should make sure whether the host's IP exists or not. The *Result* is a variable in boolean type.

```
State = Start
// A Global Variable for recording the living state

Algorithm: Searching
Given:
        State is the state of mobile server
        The IP of the host
        Result is boolean type (TRUE| FALSE)
Constraints:
        For each IP, it should accord with IPv4 or IPv6
Input:
        An IP
Output:
        For each IP, find the Result
Steps:
        IF State == Start or Suspending THEN {
                State = Searching
                For each IP, mapping it to IPv4 or IPv6
                IF the IP is qualified, THEN {
```

```
                        Set the Result =TRUE
                        State = Migrating
                }
                ELSE {
                        Result = FALSE and State = Dangling
                        Changing the transition
                }
        }
        ELSE
                Stop           // Nothing to do
```

If the IP is correct, the mobile media server provider sends a mobile agent to detect the configurations of the target as the following algorithm, including the operating platform, software configuration, and hardware configuration. The *state* will be set to Migrating until the complete of the detecting, otherwise, it will be set to Suspending. In software configuration, the program can transfer the suitable agent according to the operating system. Then, setting the remote host as server via the appropriate tool, including IIS (Internet Information Service), APACHE, … and so on for server setting, and SQL, MySQL, ORACLE…etc. for database setting.

```
Algorithm: Detecting the Configuration
Given:
        MA is for a Mobile Agent
        OS is for Operating System
        IP is for Internet Protocol
        SW_Con is for software configuration
        HW_Con is for hardware configuration
        Q = sequence-of ( OS, IP, SW_Con, HW_Con)
        Goal_Result is for the Result from the Searching
Algorithm, it is boolean type
Constraints:
        As to the OS, it may be Windows series, Linux, Unix, …etc.
        As to the SW_Con, it will be decided according to OS. It may be
IIS, APACHE, SQL, MySQL, ORACLE…etc.
        As to the HW_Con, it may be the list-of (video card, sound card,
network card, storage)
Input:
        An IP
Output:
        A sequence Q = ( OS, IP, SW_Con, HW_Con)
Steps:
        IF the Goal_Result == TRUE THEN {
                MA collects the configuration of the host through connecting
                Set OS = current platform
                Set IP = the host's IP
                Set SW_Con = one of (IIS, APACHE, …)
                Set HW_Con = list-of (video card, sound card, network card,
storage)
                Q = (OS, IP, SW_Con, HW_Con)
                State = Migrating
        }
        ELSE   {
                Q = Unknown and State = Suspending
                Changing the transition
        }
```
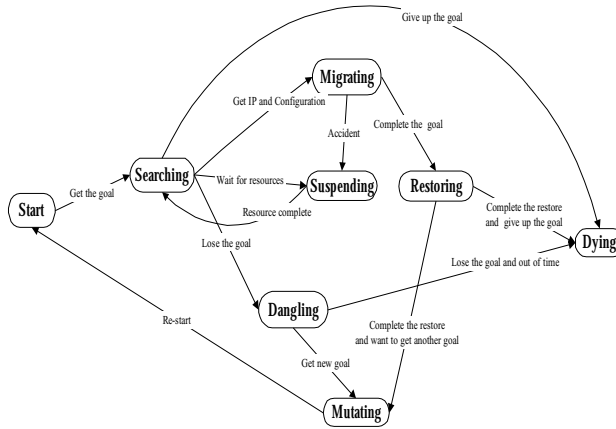
If the *state* is at Suspending, Dangling, Restoring, or Mutating, it means that the living state should be changed. The algorithm named changing the transition is to deal with this situation.

```
Algorithm: Changing the transition.
Given:
        Current_state is for the current state of mobile server
        IP_new is for new Internet Protocol
        Waiting_Time is for the time to wait for another goal
        Restore_Flag is a boolean type for deciding the restoring status
Constraints:
        Current_state = (Suspending, Dangling, Restoring, Mutating)
```

```
Input:
        Current state
Output:
        State = one of new state
Steps:
        IF Current_state == Suspending THEN
                State = Searching
        ELSE IF Current_state == Dangling THEN {
                IF the host got a new goal and within the Waiting_Time
THEN
                {       State = Mutating
                        Changing the transition.
                }
                ELSE
                        State = Dying
        }
        ELSE IF Current_state == Restoring THEN {
                IF the Restore_Flag == TRUE THEN
                        State = Dying
                ELSE
                {       State = Mutating
                        Changing the transition.
                }
        }
        ELSE IF Current_state == Mutating THEN  {
                IF Restore_Flag == FALSE THEN   {
                        Restore data to the callee
                        Set Restore_Flag = TRUE
                        State = Start
                }
                ELSE
                        State = Start
        }
```

The migration takes place in the Migrating and Restoring states. If in Migrating state, it means the mobile storage or mobile server are moved from the mobile media server provider to the indicated host. On the other hand, the direction is reverse. The following algorithm describes the work.

```
Algorithm: Migration
Given:
        MMSP stands for Mobile Media Server Provider
        Move_Flag is a boolean type for presenting the status of
migration
Constraints:
        State = one of (Migrating, Restoring)
Input: An IP and Current state
Output:
        Move_Flag = (TRUE, FALSE)
Steps:
        Move_Flag = FALSE
        IF State == Migrating THEN  {
                Migrate content from the MMSP to the host
                Move_Flag = TRUE
        }
        ELSE IF State == Restoring THEN   {
                Migrate the data from the host (Mobile Server) to the
MMSP
                Move_Flag = TRUE
                State = Mutating
                Changing the transition
        }
```

Finally, the idea could be adapted to many environments, including the Internet, wired network environment, wireless ones, etc. This is a fundamental model for supporting the mobile environment.

5. Collaborative Recommendation

In traditional business activities, we get recommendation or suggestions from other people; while in Internet environment, customers buy something by themselves and no body assists them. So a recommendation system became an important application area and academic research topic. Merchants can use the application to recommend items to customers based on their behavior, and how to keep consumer's interesting became an important thing in the environment. We use mobile agent technology to design the recommendation system. Agent technology has benefits that include automation, customization, notification, learning, tutoring and messaging. So in the paper we propose an agent-based recommendation system architecture, we use browsing-oriented approach to discovery the customer's behavior. The approach is a useful tool in the WWW environment and the application would be useful in electronic commerce environment.

5.1 Customer Behavior analysis approach

In the electronic commerce environment customer navigate what they wanted on World Wide Web. Customer's behavior will be captured according the navigation processes. Customer's navigation process includes several activities. People will buy several products or services at a time, and these bought products have relations between them and this information is useful for recommendation process. When customers navigate online chain stores, they leave their navigation sequence in the process. The navigation sequence will be useful for us to find why they bought or why they interest. This information implicitly contains some relation or meaning that will be useful for merchants.

We proposed a graphic model to analyze customer behavior for recommendation. We use mobile agent technology to capture customer's behavior. The technology can avoid some disadvantages like log analysis in the proxy environment. Furthermore, mobile agent technology can help to across several chain stores to capture customer behavior. We can capture customer's behavior in real time and will not loose information from anyone of the chain store. The agent will capture customer's behavior and build a customer buying and navigation history (CBNR) according to the user behavior.

According to the CBNR records we can build a graphic model CPF model. The model has three views that include customer-product view, customer-feature view and product-feature view. We use the user profile CBNR to extract useful information. So we build a three views analysis model – CPF model. Each view expresses an analysis metric. Figure 4 shows the basic CPF model.
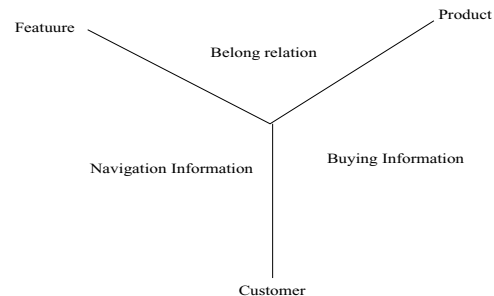
Figure 4: CPF model

- **Customer-Product Matrix**

Table 1 is the logical view of the Customer-Product Feature. If a customer i bought product j, then the C_iP_j will be filled with '1', otherwise, filled with '0'.

Table 1: Customer-Product matrix

					Product Identification			
		P_1	P_2	P_3			P_n	
	C_1							CCI_1
	C_2							CCI_2
				C_iP_j				
	C_m						C_mP_n	CCI_m
		PCI_1	PCI_2	PCI_3			PCI_n	

(left side label: Customer Identification)

About customers, we can use the *CCI* to find which customers are strong buyers; about products, we can use the *PCI* to find which products are potential products.

- **Customer Caution Index (CCI)**

$$CCIi = \frac{\sum_{j=1}^{n} CiPj = 1}{n}$$

, where C_iP_j is the matrix's content

and n is the total number of the products.

- **Product Caution Index (PCI)**

$$PCIj = \frac{\sum_{i=1}^{m} CiPj = 1}{m}$$

, where C_iP_j is the matrix's content

and m is the total number of the customers.

- **Product-Feature Matrix**

Table 2 is the logical view of the Product-Feature matrix.

Table 2: Product-Feature matrix

				Feature Identification			
		F_1	F_2	F_3			F_n
	P_1	$P_1\,F_1$	P_1F_2				$P_1\,F_n$
					P_iF_j		
	P_m	$P_m\,F_1$					P_mF_n

(left side label: Product Identification)

This matrix shows that some products have some features that express the value of the products. If product i has the feature j, then the P_iF_j will be filled with '1', otherwise, it would be filled with '0'. The content of this matrix is established by the merchants who sale the products. The content of this matrix also consists of the statistical information. $P_iF_j = \{\,V_{ij}\,,\,S_{ij}\,\}$, V_{ij} is a Boolean value we have just described above, S_{ij} is a statistical information generated with the following formula.

$$S_{ij} = \frac{total\_navigation\_time\_of\_the\_feature}{total\_navigation\_time\_of\_the\_product}$$

- **Customer-Feature Matrix**

The matrix means someone is interested in a product and focuses on the product's feature. In the previous section we built CBNR for the customers; now, we extract the information into the *navigation vector*. Each customer has a navigation vector that hide in the CBNR respect to the product. According to the navigation vector, we compare the customers with each other to find their relation and then classify them into heterogeneous groups. A navigation vector of a product for the customer i looks like: $NVi = (\,f_1\,,\,f_2\,,f_3\,,...,\,f_n\,)$, where the product has n features. We compute the elements of the navigation vector by the following equation:

$$navigation\_vector\_element\_k = \frac{\sum f_k}{K} \times avg(f_k)$$,

$$avg(f_k) = \frac{time\_of\,(f_k)}{total\_time(p_j)}$$

where , K is the number of features which the customer navigated, $time\_of(f_k)$ is the total time the customer navigated the feature f_k, $total\_time(pj)$ is the total time the customer navigated the product j. And we use *Final Navigation Vector* to update the user's latest preference:

$FNVi = old(\,FNVi\,) \times W_1 + (f1, f2, f3, ...fn) \times W_2$,,

where $W_1 + W2 = 1$, $(f_1, f_2, f_3, ...f_n)$ is the new FNV value of the customer i. W_1, W_2 is a set of factors which is adjusted according to the user's response to the recommendation. If the user accepts our recommendation, we will enhance the factor W_2; if the user doesn't accept our recommendation, we will enhance the factor W_1. The approach is the basic evaluation of our system and the fundamental method to update the user profile. Table 3 is the logical view of the Customer-Feature matrix.

Table 3: Customer-Feature matrix

				Feature Identification			
		F_1	F_2	F_3			F_n
	C_1	C_1F_1	C_1F_2				C_1F_n
					C_iF_j		
	C_m	C_mF_1					C_mF_n

(left side label: Customer Identification)

We use *algorithm 1* to cluster users into groups according to their behavior on a specific product. And we use *algorithm 2* to find users' buying pattern.

Algorithm 1: Generate User Groups for Products
Input: CF Matrix, k as the cluster number
Output: User Groups

```
{
    for i = 1 to k
        Gi ← Ci
    do {
        for i = 1 to n {
            most ← ∞
            for j = 1 to k {
```

$$temp = \sqrt{\sum_{z=1}^{m} |C_iF_z - G_jF_z|^2}$$

```
            if  temp < most
                index = j
            }
            Gindex ← Ci
        }
    }
```

```
         for   i = 1 to k {
                temp  ← 0
                for   j = 1 to Gi.length
                        temp  ← temp + Gi, j
                Gi  ← temp / Gi.length
         }
      } while set G changes
}
```

Algorithm 2: Discover Buying Pattern

Input: CP Matrix, minimum support count
Output: Buying Patterns

```
{
      for each Pj {
            count  ← 0
            for each Ci
                  if   CiPj = 1
                        count  ← count + 1
            S  ← j
            Is  ← count
      }
      do {
            for i =1 to I.length
                  for j = i to I.length {
                        count <- 0
                        for each Ck {
                              S <- i  ∪  j
                              if CkPi = 1
                                    count = count + 1
                        }
                        if count >= MSC
                              Is <- count
                  }
      } while I.length = 0
      output I
}
```

6. The Architecture and Implementation of our Chain Store

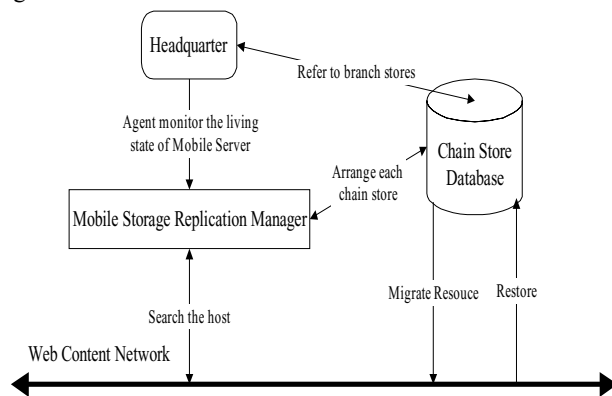The architecture of Chain Store is illustrated as figure 5.

Figure 5: The Architecture of Chain Store

According to the previous section, the model of chain store could be implemented on the Internet. As to the viewpoint of electronic commerce and virtual society, it is very useful for the users of the Internet. It is very difficult for Internet Content Providers to build an electronic chain store because they have to complete many complicated steps for building a web server first. In order to solve this complex problem, our mobile media server provides an automatic, convenience solution. The detailed descriptions were presented in section 4.

7. Conclusions

Due to the huge utilization of Internet, electronic commerce is the trend of future business. The chain store is very rare over the Internet. According to our mobile server, the electronic chain store can be built. The contributions about the electronic chain store are as following:

(1) We proposed the idea of electronic chain store over the Internet, and it is very easy for a consumer to become a business or vendor.
(2) The load of original store will be decreased due to shared by other branch stores.
(3) The problem of security can be solved partially because of the mobile server.
(4) The bandwidth can be reserved adaptively.
(5) Shorter response time can increase income because consumers always browse the nearest branch stores rather than the distant original ones.

Furthermore, we proposed a recommendation system based on user behavior. Not only did we propose a framework for recommendation system, but also practical algorithm for user classification and buying pattern discovery. We believe that personalization is an important issue for the recommendation system in e-commerce. We improved the platform that it is able to host a chain of retail stores within which users can shop around and communicate with each other. We hope the system we proposed would become a useful application for e-commerce and gained the advantages of the mobile agent technology in the e-commerce environment.

References

[1] Merrill Warkentin, Vijayan Sugumaran, and Ravi Bapna, "Intelligent Agents for Electronic Commerce: Trends and Future Impact on Business Models and Markets." Chapter 7, Internet Commerce and Software Agents: Cases, Technologies and Opportunities, IDEA GROUP PUBLISHING, 2001.

[2] Gordijn, J.; Akkermans, H. "Designing and evaluating e-business models." IEEE Intelligent Systems, Volume: 16 Issue: 4 , July-Aug. 2001.

[3] Charas, P. "Peer-to-peer mobile network architecture." in Proceedings of the First International Conference on Peer-to-Peer Computing, 2002, Page(s): 55 –61

[4] Timothy K. Shih, Yun-Chung Lu, and Jason C. Hung, " The Design and Implementation of a Mobile Distributed Web Server System" in Proceedings of National Computer Symposium 2001, December 20-21, 2001.

[5] Paul Resnick and Hal R. Varian. "Recommendation System." Communication of the ACM, 40 (3), pp. 56-58, 1997.

[6] Marko Balabanovic and Yoav Shoham. "Fab: Content-based collaborative recommendation." Communication of the ACM, 40 (3), pp. 64-72, 1997.

A Minimum Delay Routing Protocol for Bluetooth Radio Networks

Gwo-Jong Yu and Wei-Chen Hwang
Deptment of Computer and Information Science,
Aletheia University, Taiwan
yugj@email.au.edu.tw

Abstract— **Bluetooth is a low power and short range wireless communication technology. Device equipped with Bluetooth device can operate in the unlicensed 2.4G ISM band. Bluetooth adopts FHSS (frequency hopping spread spectrum) technique to resist radio interference and TDD (time division duplex) to simulate duplex communication. In previous Bluetooth routing protocols, the routing path with minimum number of hop count is chosen as the target route. However, the shortest routing path may pass through hot-spot nodes with heavy traffic which increase the delay time of packet delivery. According to queueing theory, there is a relation between traffic flow and delay time. In wireless network, heavy traffic flow implies long delay time. In this paper, we proposed a short delay routing protocol for Bluetooth radio system. The delay time is estimated in each link using query theory. Due to the operations of devices with roles of master, relay and slave are different, we also discuss three delay time estimation models based on query theory. Experimental results reveal that the proposed short delay routing protocol outperforms DSR-based Bluetooth routing protocol in terms of delay time.**

Keywords- Bluetooth, querying theory and routing protocol.

I. INTRODUCTION

Bluetooth is a short range wireless communication technology which can be used for cable replacement in portable or fixed electrical equipments. It operates on un-licenced 2.4G industry, science and medical (ISM) band and adopts frequency hopping spread spectrum (FHSS) technology to reduce radio interference. When two devices need to communicate with each other, intermediate nodes are needed to forward data packets. In Ad Hoc networks, lots of routing protocol have been proposed to solve the communication problem between two devices. Ad Hoc network consists of mobile devices which form a communication network dynamically. In the following, we will compare the delay times and related problem of previous proposed routing protocol.

1. Routing path selection factors: The rules to select optimal pathes vary for each routing protocols. The commonly used route selection rules include (1) shortest path, (2) minimum delay time, (3) QoS requirement, (4) load balance and (5) network lifetime. The complexity of routing protocol increased as more rules are taken into considerations.

2. Routing Strategy: According to the variation of routing strategies, routing protocols can be categorized into table driven, on-demand and hybrid classes. (1) *Table Driven Routing Protocol*: In table driven routing protocols, each device maintains a routing table which contains informations of how to reach destination nodes. (2) *On-Demand Routing*

Protocol: It is not necessary to maintain routing table in on-demand routing protocol. When a route request is initiated, a route is established in a dynamic manner using flooding approach. (3) Hybrid Routing Protocol: The advantages of table driven and on-demand routing protocols are combined to improve performance of routing protocol.

In literature, lots of researchers have proposed many bluetooth routing protocols. Among them, Lamport and Lynch [1] used flooding mechanism to forward page packets to find routing path. As soon as the first packet received by destination device, it will reply immediately to source node to determine the shortest routing path. When routing path is established, the architecture of scatternet is changed such that the efficiency of data transmission can be further improved. The information of the discovered routing path will be recorded in those devices which participats the route process. This historical routing information will benefit the route discovery phase when the next route request with the same destination is initiated. However, the drawbacks are that the information of historical path will expire if the topology of scatternet changes frequently. In this circumstance, using historical routing information will take large risk. Lindsey [2] had proposed a routing protocol to search the shortest routing path in Bluetooth radio system. In DSR-based routing protocol, the same routing path will be used repeatly. In this circumstance, those intermediated nodes will have large accumulated traffic because many routing pathes pass through these nodes simultaneously. To solve this problem, we will introduce a short delay routing protocol which discovery the routing path with minimum accumulated and estimated queueing delays. The advantages of the proposed short delay routing protocol include (1) small delay time, (2) load balance, (3) long network lifetime and (4) support of QoS request in upper network layer.

Based on previous discussion, the minimum delay routing protocol can find the path with minimum delay times. We will estimate the delay time of each master and relay devices through queueing theory. We also propose a method to compute the amount of extra delay cause by new route. In route search phase of on-deman routing protocol, the delay information can be used to find the route with minimum accumulated delay times. Characteristics of the computed route include minimum delay, load balance and long lifetime such that the lifetime of entire network can be increased.

The remaining sections are organized as follows. In Section II, we will introduce basic queueing models. Section III is the proposed minimum delay routing protocol.

Experimental results are demonstrated in Section IV. Finally, Section VI concludes this paper.

II. PRELIMINARY

In Section III, a minimum delay routing protocol is proposed to find the route with minimum accumulated delay. The performance of the proposed protocol depends on the accuracy of the estimated delay time. In this section, we will review basic characterstics of queueing theory such that the task of delay time estimation can be achieved easily.

2.1 Queueing Model

Generally Speaking, a queueing model includes the following components. An input source which generate service request (client) in a random manner is required. After the request is generated, the client will enter the queueing system and become a member in the queue. When the server is available, a member from the queue is chosen to be served according with some service discipline. When the service mechanism is complete, the client will leave the queueing system. The basic queueing process is shown in Figure 1.
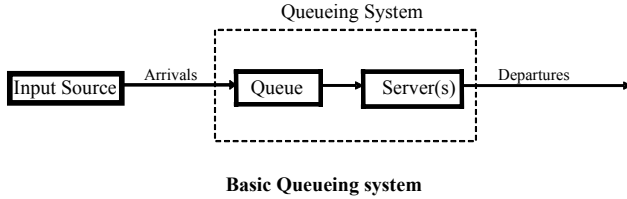

Basic Queueing system

Figure 1: Basic queueing process

2.2 M/M/1 Model

M/M/1 model represents a queueing system which comprises a single server. Both service request inter-arrival time and service time are exponent times. Becase M/M/1 only adopts the prefix three symbols of Kendall notation [8], it is assumed that the capacity is infinite and the service discipline is first in first out (FIFO). Due to the capacity is infinite, the inter-arrival rate is constant λ. Similarily, there is only one server in system, as long as at least one client in system, the service rate will be constant μ. According to the infinite summation formula $\sum_{n=0}^{\infty} \beta^n = \frac{1}{1-\beta}, \beta < 1$ and let$\rho = \frac{\mu}{\lambda}$, then the number of clients L in system are $L = \frac{1}{1-\rho} = \frac{\lambda}{\mu - \lambda}$. According to Little rules, we can compute the average waiting time in queue is $W = \frac{L}{\lambda} = \frac{1}{\mu - \lambda}$.

2.3 M/G/1 Model with vacation

The analysis of M/M/1 queueing model is based on the birth-death process. This implies that both the inter-arrival time and the average service time must obey exponent distribution. The assumption provides convenient of analysis and desired property. However, this assumption limits the application of real life problem. Especially, the service time is exponent distribution implies the input source is a passion process. In most case, this is a reasonable assumption. But in case that the

arrival process is arranged, the assumption is invalid. Furthermore, in pratice, the service rate is far from a exponent distribution. Especially, the clients require similar service. So, it is important to adopt a reasonable queueing model. In the following, we will introduce M/G/1 model.

For better system utilization, a server will give service to multiple clients. Thus each client will encounter service interval and waiting interval. When server take service to other clients, it is reasonable to treate the server as if it is in vacation untile it is served next time. In M/G/1 queueing with vacation model, the residual life approach [8] is commonly used to analysis the waiting time of client in queue. Let $X_k(t)$ and $V_k(t)$ denote the tth service time and the tth vacation interval of client k, respectively. Then the average service time after T services is $\overline{X}_k = \frac{1}{T}\sum_{t=S}^{S+T} X_k(t)$. Similarilty, the average vacation time after T times of vacations is $\overline{V}_k = \frac{1}{T}\sum_{t=S}^{S+T} V_k(t)$. Let λ_k denote the data arrival rate of client k, then according to [8], the average queueing delay will be $W_k = \frac{\lambda_k \overline{X}_k^2}{2(1-\rho)} + \frac{\overline{V}_2^k}{2\overline{V}_k}$, where $\overline{X}_k^2 = \frac{1}{T}\sum_{t=S}^{S+T} X_k^2(t)$, $\overline{V}_k^2 = \frac{1}{T}\sum_{t=S}^{S+T} V_k^2(t)$ and $\rho = \lambda_k E(X_k) = \lambda_k \overline{X}_k$. So, when $X_k(t)$ and $V_k(t)$ are known for any systems with polling mechanism, the average queueing delay time W_k can be computed.

III. THE MINIMUM DELAY ROUTING PROTOCOL

In this section, the proposed minimum delay time routing will be described using a numerical example as shown in Figure 2. Then, the routing protocol and packet format will be presented. Finally, the meaning associated with minimum delay route will be discussed.

Figure 2: An numerical example to illustrate the minimum delay routing protocol.

In Figure 2, there is a symbol a/b associated with each link, where a denoted the number of transmited slots associated with this link in a time interval and b denotes the estimated delay time. The device which initiate route discovery is device S. The target device is destination D. When source initiate route discovery through flooding, the

possible routes associated with route request packets reveived by destination *D* are as follows:

(a) Path 1:
S→M1→R1→M2→R3→M5→R6→M6→R8→M7→D, with delay time＝4.0(sec), traffic flow = 47(slot) and hop count＝10.

(b) Path 2:
S→M1→R1→M2→R3→M5→R7→M8→R9→M9→R10→M7→D, with delay time＝3.7(sec), traffic flow = 45(slot) and hop count＝12.

(c) Path 3:
S→M1→R2→M3→R4→M4→R5→M5→R7→M8→R9→M9→R10→M7→D, with delay time＝3.0(sec), traffic flow ＝47(slot) and hop count＝14.

(d) Path 4:
S→M1→R2→M3→R4→M4→R5→M5→R6→M6→R8→M7→D, with delay time＝3.3(sec), traffic flow ＝49(slot) and hop count＝12.

When destination D received route request packets within a time interval, it can choose the route with preferred criteria. The route with minimum hop coute is path 1 while the route with minimum delay is path 3. The estimated delay time associated with path 1 and 3 are 4.0 and 3.0 second respectively. The route with minimum hop count may not achieve best performance. But the minimum delay routing protocol can find the route with minimum delay. When the estimated delay associated with all links are the same, the minimum delay routing protocol reduce to shortest path routing protocol.

3.1 Minimum Delay Routing Protocol

The minimum delay routing protocol is described as follows.

(1) Due to the communication process within a piconet must pass through the master device, the amount of traffic associate with each link in a time interval can be computed by master. Using the traffic flow information and basic queueing model, the estimated average delay time associated with each link can be computed accurately. The formula to compute the delay time associated with each link has been described in previous sections.

(2) When the source device initiates route request, it will broadcast a route request packet to its neighbor. The packet will be rebroadcasted by masters and relays through entire scatternet in a flooding manner. The information embedded in the route request packet includes source address and the accumulated delay time of links which this packed has been pass through. When this packet is further transmitted to the next node, the delay time of the new link will be added into the route request packet. When the route request packet reach destination node, the total accumulated delay time can be extracted from packet, so the destination can decide which route will encounter as little delay as possible.

(3) When destination node receive packets coming from various path, it can choose the best path by selecting the route with minimum delay. The delay information can be obtained from packets accordingly.

From the above mechanism, the route with minimum expected delay can be obtained.

3.2 The Relation between DelayTtime and Traffic Lload

Although the focus of minimum delay time routing protocol is delay time, the relation among delay time, lifetime and traffic load is obvious. A small delay time implies that the traffic load is light, the power consumption is small, the throughput is good and traffic load is balanced. Thus, the above objectives can be achieved through delay time analysis. The following sections discuss the detail analysis.

IV. EXPERIMENTAL RESULTS

4.1 Experiments Setup

To evaluate the performance of the proposed minimum delay routing protocol, we design a simulation environment to observe and measure the system performance in various conditions. Besides the parameters of the proposed algorithm, the remaining parameters of simulations includes (1) the size of the environment, (2) the number of devices in a piconet, (3) the number of constructed routing path between two devices and (4) data arrival rates. Experiments are conducted in a connected scatternet. All devices are within communication range. The size of simulation environment is 50x50 in which 50 devices are placed.

4.2 Performance Evalutaion

The proposed minimum delay routing protocol (MDP) will be compared with the state of the art, RVM method (Bhagwat, P.; Segall, in"A routing vector method (RVM) for Routing in Bluetooth scatternets") through the following comparison.

- Comparison of the relation between delay time and traffic load.
- Comparison of the relation between throughput and traffic load.
- Comparisioin of the relation between network life time and traffic load.

A. The relation between delay time and traffic load

There are many factors which effect data transmission in Bluetooth network. One of the important factors is the delay which implies traffic load. In literature, lots of proposed routing protocol using hop count to decide the suitable route. However, small hop count does not imply short delay time. Especially in hot spot area, large number of routing pathes pass through the same device which may incure large communication latency due to heavy traffic.

Figure 3 shows the relation between traffic load and delay time. The X-axis of Figure 3 represents traffic load which is measured by number of packets (slots). It is assumed that both master and slave use DH1 packet to transmit data. So the number of packets will equal to the number of slots. As shown in Figure 3, the delay time increases for all methods

when the traffic load increase. But the delay time of MDP is smaller than RVM. So the proposed MDP is superior than RVM in terms of delay time.
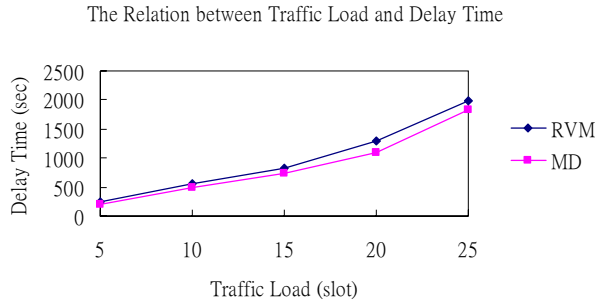
The Relation between Traffic Load and Delay Time

Figure 3: Comparision of the MDP and RVM in terms of delay time and traffic load.

B. The relation between number of devices and delay times

The number of devices in a scatternet will also affect the delay time of routing pathes. As the number of devices increase, the delay time increases. The comparison of relations between number of devices and delay times are shown in Figure 4. The X-axis and Y-axis of Figure 4 are number of devices and delay times, respectively. Figure 4 corresponds to traffic load 10. From these figures, it is observed that the delay times increase in both RVM and MDP. However, the proposed MDP has smaller delay time than RVM in all cases. The MDP has better performance because information of delay time in each link is provided.
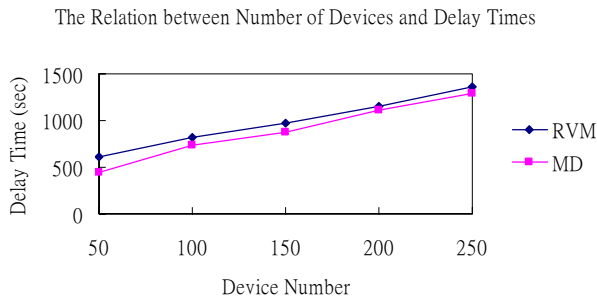
The Relation between Number of Devices and Delay Times

Figure 4: Comparision of the relations between number of devices and delay time under the condition that the traffic load is 10.

C. The relation between delay time and average number of relay

Through proper route selection, the troughput of data transmission can be increased. But in a scatternet, if the number of relay is too small in a piconet, master will have few choices to select good path. It is possible that the hot spot area can not be avoided such that the performance of the system is poor. Thus, a scatternet with more relay device will benefit the route selection process.

To observe the relation between number of relays and delay time, experiment is conducted in various traffic loads.

Figure 5 shows the relations between average number of relays and delay times in traffic load 10, 20 and 30, respevtively. When the number of relay node is increased in a piconet, the construct topology will have smaller delay time. From Figure 5, it is observed that the delay time of MDP is smaller than RVM when the number of relays is fixed. It is also observed that increase the number of relays will decrease the delay times.
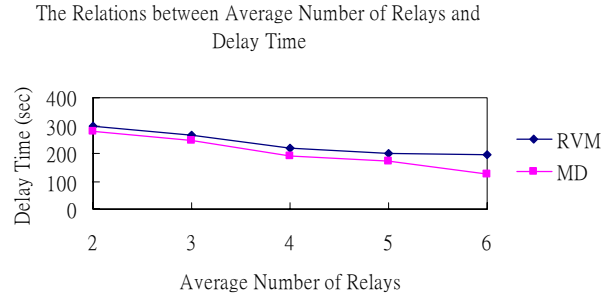
The Relations between Average Number of Relays and Delay Time

Figure 5: Comparison of the relations between average numbers of relays and delay times when traffic loads 10 packets per second.

V. CONCLUSION

In most Bluetooth routing protocol, the route with minimum hop count is chosen as the final route. However, large delay times occurred when the traffic load is heavy in some intermediate node of the route. In this paper, we have proposed a minimum delay routing protocol for Bluetooth radio network. The delay time associated with each link is estimated according the roles of devices in the two ends of a link through proper queueing modeling. Based on the estimated delay times in each link, a route with minimum accumulated estimated delay time is chosen as a suitable route. Experimental results demonstrate that the proposed MDP has smaller delay times in comparision with RVM. In the future, the proposed delay time estimation framework will be extend to 802.11 WLAN and Ad Hoc networks.

REFERENCE

[1] L. Lamport and N. Lynch., "Chapter on distributed computing.", Technical Report MIT/LCS/TM-384, Massachusetts Instituted of Technology, Cambridge, Massachusetts, '89

[2] Lindsey, S., Raghavendra, "Power-Efficient Gathering in Sensor Information Systems.", Aerospace Conference Proceedings, '02, IEEE, vol.3, '02, pp. 1125 –1130

[3] M. L. Neilsen and M. Mizuno., "Decentralized consensus protocols.", In IEEE 10th International Phoenix Conference on Computers and Communications, '91 pp. 257-262

[4] M.R. Pearlman Z.J. Haas., "The zone routing protocol (zrp) for ad-hoc networks.", Technical report, Internet Draft RFC, November '97

[5] Mitchell L. Neilsen, "Quorum Structure in Distributed System", '92

[6] S. Lindsey, C. Raghavendra, and K. Sivalingam., "Data gathering in sensor networks using the energy*delay metric.", IEEE Transactions on Parallel and Distributed Systems, vol.13, no.9, September '02

[7] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan., "Energy-Efficient Communication Protocol for Wireless Microsensor Networks.", In Proceedings of the Hawaii Conference on System Sciences, January '00

[8] Sanjay K. Bose, *An Introduction to Queueing Systems*, Kluwer Academic/Plenum Publishers, New York.

Stock Price Prediction Agent Based on Gray Theory

Kuan-Cheng Lin, Jeff T. C. Lee, Min-Tzu Wang and Shu-Huey Yang*

Department of Information Management
Kuang Wu Institute of Technology
Peitou, Taipei, Taiwan 112, R. O. C.
kuanchenglin@kwit.edu.tw

*Da-Long Primary School
Da-tong, Taipei, Taiwan 113, R. O. C.
nnfuzzy@tp.edu.tw

ABSTRACT

In this study, we describe a stock price prediction agent system based on gray theory. This system is constructed on a web by using JavaServer Page and the system is user-friendly.

Keywords: Stock price, gray theory, JavaServer Page

1. Introduction

The stock price prediction plays an important role on the business prediction. The adequate stock price prediction can help personal investor and fund manager to obtain most profit. Usually, technical forecasting and fundamental forecasting are used to predict the stock price. However, for the reasons of insufficient data or uncertainty, the traditional forecasting method is generally hard to predict the optimal price. In this study, we describe the stock price prediction agent based upon the gray theory. The prediction agent directly uses the gray accumulation operation (AGO) and GM(1,1) [1][2] to forecast the stock price.

The web-based stock price prediction agent system is constructed on http://203.64.218.5/stock. And, the web-based stock price prediction agent system is constructed on a Tomcat 4.1.27 web server in the Windows 2003 server operating system. The system must authenticate its users for security. The system helps the users to predict the stock price. The system uses the JSP program with the Java Database Connectivity (JDBC) driver to enable access to a database of the historical stock price. The JavaServer Pages (JSP) technology has been designed to provide a simplified, fast means of creating web pages that display dynamically-generated content [3].

Section 2 describes the theory and algorithm of gray prediction model and offers an example. Section 3 describes the functions of the web-based stock price prediction agent system. Finally, section 4 concludes.

2. The theory and algorithm of gray prediction model

Deng introduces the gray theory in 1982 [1][2]. The gray theory can use a few known data by the way of accumulation generation operation (AGO) to build a gray prediction model. The gray prediction model [1][2] is introduced in subsection 2.1. The computing algorithm of stock price prediction agent system is described in subsection 2.2. And, subsection offers an example of the computing algorithm.

2.1 Gray Model GM(1,1)

Let $x^{(0)}=(x^{(0)}(1), x^{(0)}(2), ..., x^{(0)}(n))$ is the raw data and $x^{(1)}=AGO(x^{(0)})'=(x^{(1)}(1), x^{(1)}(2), ..., x^{(1)}(n))$.

where $x^{(1)}(k)=\sum_{i=1}^{k} x^{(1)}(i)$. And, the following equation is the gray differential equation:

$$x^{(0)}(k)+aZ^{(1)}(k)=b, \ k=1,...,n \qquad (1)$$

The gray model described by the above equation is GM(1,1) since it consists of only one variable $x^{(0)}$.

In Eq. (1), $Z^{(1)}(k) = 0.5x^{(1)}(k)+ 0.5x^{(1)}(k-1), k=1,2,...,n$, a is the developing coefficient, b is the gray input, and $x^{(0)}$ is a gray derivative which maximizes the information density for a given series to be modeled. According the least square method, we have

where

$$\hat{a} = \begin{bmatrix} a \\ b \end{bmatrix} = \left(B^T B\right)^{-1} B^T y_n$$

$$B = \begin{bmatrix} -Z^{(1)}(2) & 1 \\ -Z^{(1)}(3) & 1 \\ ... & ... \\ -Z^{(1)}(n) & 1 \end{bmatrix}, \ y_n = \begin{bmatrix} x^{(0)}(2) \\ x^{(0)}(3) \\ ... \\ x^{(0)}(n) \end{bmatrix}$$

Expanding the above equations, we have

$a=(CD-(n-1)E)/((n-1)F-C^2)$

$b=(DF-CE)/ ((n-1)F-C^2)$

where

$C=\sum_{k=2}^{n} Z^{(1)}(k),$

$D=\sum_{k=2}^{n} x^{(0)}(k),$

$E=\sum_{k=2}^{n} Z^{(1)}(k) x^{(0)}(k),$

$F=\sum_{k=2}^{n} Z^{(1)}(k)^2 .$

By referring the first-order differential equation of sequence $x^{(1)}$ as follows:

$d x^{(1)}(k)/dk+ aZ^{(1)}(k)=b,$

and we have the following equation:

$x^{(1)}(k+1)=(x^{(0)}(1))e^{-ak}+b/a(1-)e^{-ak})$

$x^{(0)}(k+1)= x^{(1)}(k+1)- x^{(1)}(k).$

Finally, by substitute a and b, we can establish the gray prediction equation.

2.2 Gray Prediction Algorithm by GM(1,1)

The prediction algorithm based on gray theory are describe as follows:

Step 1: Build the observed exchange rate sequence

$X^{(0)}=(X^{(0)}(1), X^{(0)}(2), ...,, X^{(0)}(n))$

where $X^{(0)}(k)$ represents the exchange rate at time k.

Step 2: Generate the accumulation generating operation (AGO) sequence.

$X^{(1)}=AGO(X^{(0)})'=(X^{(1)}(1), X^{(1)}(2), ..., X^{(1)}(n))$

where $X^{(1)}(k)=\sum_{i=1}^{k} X^{(1)}(i)$

Step 3: Calculate the mean value of the AGO sequence.

$Z^{(1)}(k) = 0.5X^{(1)}(k)+ 0.5X^{(1)}(k-1), k=1,2,...,n$

Step 4: Compute the parameter a and b.

$a=(CD-(n-1)E)/((n-1)F-C^2)$

$b=(DF-CE)/ ((n-1)F-C^2)$

where

$C=\sum_{k=2}^{n} Z^{(1)}(k),$

$D=\sum_{k=2}^{n} x^{(0)}(k),$

$E=\sum_{k=2}^{n} Z^{(1)}(k) x^{(0)}(k),$

$F= \sum {}^{n}_{k=2} Z^{(1)}(k)^2$.

Step 5. Establish the prediction equation.

$x^{(1)}(k+1)=(x^{(0)}(1))e^{-ak}+b/a(1-)e^{-ak})$

$x^{(0)}(k+1)= x^{(1)}(k+1)- x^{(1)}(k)$.

2.3 An Example

In this section, we offer an example to illustrate the algorithm describes in the above subsection.

Step 1: Choose a stock and establish the price sequence with respect to day.

$X^{(0)}=(52 ,50 ,49.9 ,51 ,50.5 ,49.4 ,47.1)$

Step 2: Generate the accumulation generating operation (AGO) sequence.

$X^{(1)}=(52 ,102 ,151.9 ,202.9 ,253.4 ,302.8 ,349.9)$

Step 3: Calculate the mean value of the AGO sequence.

$Z^{(1)} =(77 ,126.95 ,177.4 ,228.5 ,278.15 ,326.35)$

Step 4:Compute the parameter *a* and *b*.

$C= \sum {}^{n}_{k=2} Z^{(1)}(k)$

$= 77+126.95+177.4+228.5+278.15+326.35$

$=1213.95$

$D= \sum {}^{n}_{k=2} x^{(0)}(k)$

$=50 +49.9 +51 +50.5 +49.4+47.1$

$=297.9$

$E=\sum^{n}_{k=2} Z^{(1)}(k) x^{(0)}(k)$

$=77*50+126.95*49.9+177.4*51+228.15*50.5$

$+278.1*49.4+326.35*47.1$

$=59863.005$

$F=\sum^{n}_{k=2} Z^{(1)}(k)^2$

$= 77*77+126.95*126.95+177.4*177.4$

$+228.5*228.5+278.15*278.15+326.35*326.35$

$=289412.4175$

$a=(CD-(n-1)E)/((n-1)F-C^2)$

$=0.009351887$

$b=(DF-CE)/ ((n-1)F-C^2)$

$=51.54212054$

Step 5: Establish the prediction equation.

$x^{(1)}(k+1)= -5459.413939e^{-0.009351887k}+5511.413939$

where

$x(1)=52, b/a=135697$

Step 6: Verify the prediction results.

$k=1, x^{(1)}(k+1)=X^{(1)}(2)= 102.8178, X^{(0)}(2)=50.8178$

$k=2, x^{(1)}(k+1)=X^{(1)}(3)= 153.1626, X^{(0)}(3)=50.3448$

$k=3, x^{(1)}(k+1)=X^{(1)}(4)= 203.0388, X^{(0)}(4)=49.8761$

$k=4, x^{(1)}(k+1)=X^{(1)}(5)= 252.4507, X^{(0)}(5)=49.4119$

$k=5, x^{(1)}(k+1)=X^{(1)}(6)= 301.4027, X^{(0)}(6)=48.9519$

$k=6, x^{(1)}(k+1)=X^{(1)}(7)= 349.8990, X^{(0)}(7)=48.4963$

Hence, we have the predicted results in the following table：

Original Stock Price	Predicted Stock Price	Error(%)
52	52	0
50	50.8178	-1.63566
49.9	50.3448	-0.89139
51	49.8761	2.20357
50.5	49.4119	2.15462
49.4	48.9519	0.90693
47.1	48.4953	-2.96458

And, the root mean square error (RMS) is 0.018378822.

3. An Stock Price Prediction Agent System

The web-based stock price prediction agent system is constructed on http://203.64.218.5/stock. And we describe the structure and function of the proposed system.

3.1 System Overview

The web-based stock price prediction agent system is constructed on a Tomcat 4.1.27 web server in the Windows 2003 server operating system. The system

must authenticate its users for security. The system helps the users to predict the stock price. The system uses the JSP program with the Java Database Connectivity （JDBC) driver to enable access to a database of the historical stock price.

3.2 The function of the Stock Price Prediction Agent System

The system has the following functions:

(1) System administration: system administrator can establish the accounts and passwords of users.

(2) Stock selection: users can choose the stock to predict the future stock price.

(3) Parameter selection: users can freely choose the prediction parameter, like the number of initial sequence.

(4) Period Selection: users can freely choose the prediction period: day, week or year.

4. Conclusions

In this paper, a web-based stock price prediction agent system based on gray theory was proposed. The stock price prediction can be completed on-line because the algorithm of gray prediction model is simple and fast. Finally, the on-lined prediction function of web-based system are not limited by time, it can be used all day. So, the proposed system is user-friendly.

REFERENCES

[1] J.L. Deng, "Control problem of Grey system," *System Control Letter,* Vol.1 No.1, 1982, pp.288–294.

[2] J.L. Deng, "Introduction to Grey system," *The Journal of Grey System,* Vol.1, No.1, 1989, pp.1–24.

[3] Java Server Pages (**JSP**) Specifications http://java.sun.com/**jsp**

The Use of Accumulated User Mobility for Supporting Quality of Service in Mobile Cellular Systems

[†]Liang-Teh Lee, [‡]Chen-Feng Wu, [†]Ching-Ren Wei, and [†]Der-Fu Tao
[†]Department of Computer Science and Engineering, Tatung University Taipei, Taiwan
[†]ltlee@cse.ttu.edu.tw +886-2-2592-5252 ext 3263
[‡]Department of Information Management, Yu-Da College of Business Miao-li, Taiwan
[‡]cfwu@ms1.ydu.edu.tw

Abstract

As the number of mobile users in cellular systems increases, the user mobility becomes the dominating factor of guaranteeing quality of services (QoS). In order to support QoS for mobiles in cellular systems, the concept of accumulated user mobility is combined with the handoff algorithm to reduce the amount of reserving channels (codes) for handoff calls and increase the total carried traffic load. The relational function of extenics is introduced to formulate the velocity of mobile in the proposed method. With the cell approaches, the information of user profile is applied to the proposed method to calculate the handoff attempts of each adjacent cell in current cell. By the accumulated handoff attempts of the six adjacent cells, the appropriate channels (codes) can be reserved for handoff beforehand, and the required QoS is preserved during handoffs occurring. Both of QoS and channels (codes) are the considered issues in the proposed scheme, so better system utilization can be achieved.

Keywords: user mobility, QoS, accumulated, handoff, extenics

1. Introduction

Due to the rapid evolution of mobile communications, the applications are also accompanied to vary from voice data which is low demand of bandwidth to multimedia services that are high demand of bandwidth. In such a large number of bandwidth demand for multimedia services in mobile cellular systems, the most important issue is to support quality of service (QoS) for subscribers. As a result of the increase in mobile velocity and limited radio spectra, it is difficult to allocate suitable bandwidth for mobiles before handoff to the appropriate cell. In order to increase the system capacity, the cell size will be mutated from micro-cell to pico-cell in future cellular systems, thus the dwell time in a certain cell for all of the call duration will become smaller than before. Because the handoff frequencies that are affected by the cell size increase progressively, mobiles maybe come through performance degradations. Due to the variable user mobility, it became more complex to predict the

appropriate cell for handoff. The past research [1] showed the impact of mobility on cellular network and provided a modeling method for configuring cellular networks to study the dynamics of mobility. Therefore, the user mobility should be considered in an effective handoff algorithm for supporting QoS.

The blocking probability of new call and resource utilization should be also mentioned simultaneously when the QoS is considered for an effective handoff algorithm. There are so many studies [2][3] which focused on the issue of QoS guarantee, and the effective user mobility and resource management is mentioned in [4]. The improvement of radio bandwidth is always thought as a dynamic channel (code) allocation problem in past literatures. In order to reduce the call dropping rate (CDR), the reservation scheme was proposed in the study [5]. However, seldom literature has developed to satisfy both issues of the above at the same time. Although prioritized channel assignment schemes for accommodating handoff attempts and optimization studies for assigning channels to priority classed have been proposed in the reference study [6], the user mobility has been considered a crucial factor for affecting traffic performance in the cellular networks.

The emergency problem on demand of extensive bandwidth for multimedia services and the increase velocity of mobiles is appropriate bandwidth allocation to guarantee QoS and progress the resource utilization obviously. Thus if the prediction of user mobility is applied suitably for channel (code) allocation in mobile cellular networks, the resource utilization will be improve clearly.

However, in order to solve QoS problem, a proper handoff algorithm that could reserve the required bandwidth for the mobile in the predicted cell should be proposed. There two approaches, which are cell approaches making use of global knowledge of users traffic flows in a particular cell and user approaches relying on the knowledge users mobility in short or long term, focuses on the subject of handoff prediction, but the resource utilization is not so satisfied. In this paper, therefore, accumulated user mobility is applied to handoff the algorithm which formulates the user velocity by extenics [7] and is combined with the profile concept of cell approaches for solving this problem. The proposed

scheme does not intend to predict the accuracy movement of mobile but calculate the accumulated attempts of the adjacent cell which the current is going to handoff to by using the user profiles, user regular routines information, and the speed of mobile movement.

2. Modeling by Accumulated User Mobility

In order to analyze user mobility, the mobile speed is viewed as the most important factor that should be considered firstly. Thus we make efforts in the acquisition of user velocity and the formation of user velocity. According to the free space propagation of radios [8], the distance between the base station and the mobile can be calculated by the received power of mobile. The speed can be obtained roughly by dividing the displacement in a time period.

2.1 Formulating of User Velocity

Due to the complexity of modeling the mobile velocity into an equation expression, however, we reference to the method of modeling the relational function in the Extenics [7] for reducing the complexity.

Wu adjust the original relational function for defining the speed of mobile movement in wireless cellular networks. There are three kinds of movement speed defined in Table 1. The probability of handoff is very low except only just along the edge of two cells when a pedestrian take a walk around no matter urban area or suburbs. Because the handoff probability is very low, the definition value of relational function is modified to be 0.

In contrast to pedestrian, vehicles mobility can be classified two classes; one is that vehicles move in the speed between 5km/hr and 50 km/hr around urban area, and the other is that vehicles move in the speed of faster than 50 km/hr around suburbs. The first case can be regard as extension part for Extenics, because the speed of vehicles may be slower than pedestrian for a traffic jam and faster than 50 km/hr when the traffic is smooth. In the other case, because the vehicles move in the speed of faster than 50 km/hr, the probability of handoff occur will be very high and the definition value of relation function will always be greater than 1.

Table 1 Definition of relational function

	Potential category of traffic	Definition of velocity function $k(v)$
Speed $\leq$ 5 km/hr	Pedestrians	$k(v) = 0$
5 km/hr < Speed $\leq$ 50 km/hr	Vehicles in urban area	$k(v) = (v - 5) / (50 - 5)$
>50 km/hr	Vehicles in suburbs	$k(v) = 1 + (v - 50) / V_{limited}$

2.2 Formulating of Accumulated User Mobility

The proposed scheme is suitable for the traditional honeycomb, and the area of geometry for each cell is equal. Based on the honeycomb structure, the potential cells of handoff have only six adjacent ones. For cell 'a', illustrated in Figure 2.1, may handoff to the cell which is adjacent with cell 'a' such as 'b', 'c', 'd', 'e', 'f', or 'g'.
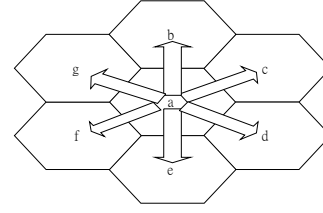

Figure 2.1 The possible handoff cell for cell 'a'

In the proposed method, cell 'a' is as an initial point to calculate the potential handoff attempts which are defined by the accumulation function of weight for each adjacent cell. In order to define the accumulation function of weight, we find out that the affecting factors for handoff are movement speed and user regular routes. The movement speed is defined by velocity function, and the potential probability of movement for each cell can be obtained by analyzing user regular routes.

Due to obtain the movement probability for each mobile in every cell, a profile-based method of user approaches [9] is applied to observe and record the mobile habits for a periodic time in each cell. By the profile-based method, two phases are required: the phase one learning the user's mobility and every move of the user is recorded, and the second phase selects a cell which the algorithm looks for recorded transitions in its base. The proposed scheme reduces the recording information of cell by modifying data structure of the reference [6], and the necessary information of cell for each mobile is described as followings:

- i: the current cell number.
- T_{dwell_avg}: the average dwell time in a cell for a mobile call.
- T_{dwell_min}: the minimum dwell time in a cell for a mobile call.
- $P_{move(j)}$: the probability of handoff to cell j, and j is the adjacent cell.

To calculate the accumulation attempts of handoff in each cell, an accumulation function is defined and formularized as the equation (eq.2-1), where the weight factors w_of_v, w_of_t and w_of_mp are represented weight of velocity function, weight of the average dwell time and weight of move probability, respectively. The portion of each weight factor stands for the dominant portion for the handoff and can be adjusted dynamically by imf_v, imf_t, and imf_mp which summation is equal to 1 according to the real situation.

$$A(w) = k(v) * w\_of\_v(w) *imf\_v + w\_of\_t(w) *imf\_t + w\_of\_mp(w) *imf\_mp \quad (eq. 2-1)$$

3. Handoff Algorithm and Simulation

Based on the past literatures, many channel allocation schemes have been proposed for handoff such as fully shared scheme (FSS), guard channel scheme (GCS) and so on. In order to guarantee the call dropping probability (or called QoS) below a certain level, the guard channel scheme is usually applied for handoff.

3.1 The Proposed Handoff Algorithm

The proposed method which applies accumulated user mobility can be composed of two phases: Phase 1 is a learning phase which is responsible for collecting user profile by user approaches and recording to the corresponding data structure; and the major duty of phase 2 is to calculate the total accumulation attempts of the adjacent cell according to the defined accumulations function of weight. In order to simplify, the proposed method considers recording the data of user profile of each cell for each mobile by modifying the data structure of the reference research [10]. And the two phase's handoff algorithm (TPHA) is illustrated in Figure 3.1.

TPHA (Two Phase's Handoff Algorithm)
Begin
Phase 1:
Begin
   Learn user profile;
   Record user profile to database;
End Phase 1;
Phase 2:
Begin
   Choose mobile information from database;
   Select the current cell number (i) and get the information;
   If $T_{dwell\_avg} >= T_{call\_avg}$
     $w\_of\_t(w)_i = HIGH$; /*Set the value of $w\_of\_t(w)_i$ to HIGH*/
   Else
     $w\_of\_t(w)_i = LOW$; /*Set the value of $w\_of\_t(w)_i$ to LOW*/
   EndIf;
   While ($j \in A$, A is the set of adjacent cells of cell i)
   Begin
     $w\_of\_m(w)_j = P_{move(j)}$; /* Set the $w\_of\_m(w)_j$ to be the value of move probability from cell (i) to cell(j); */
     $A(w)_j = k(v) * w\_of\_v(w)_i * imf\_v + w\_of\_t(w)_i * imf\_t + w\_of\_mp(w)_j * imf\_mp$;
     /* Calculate the accumulation attempts from cell (i) to cell (j)*/
     $\tilde{A}(w)_j = \sum A(w)_k$; /*Sum up $A(w)_k$ (k is the adjacent cells of cell(j)) in cell(j)*/
     $GCS\_num = Total\ channels - \tilde{A}(w)_j * Predict\_basis$;
     /*Reserve the necessary resource and adjust the number of guard channel in cell(j) by total accumulation attempts of mobility $\tilde{A}(w)_j$ */
     Prepare the handoff to the cell(j);
   EndWhile;

Figure 3.1 Two phase's handoff algorithm

In the phase 1, user profile is learning by applying user approach in [10]. During the learning phase, every move of user is recorded. For a certain cell, therefore, the movement probability of the adjacent cells can be obtained with its own handoff times divided by the total handoff times in current cell.

In phase 2, the value of *w_of_t* is set by comparing T_{dwell_avg} with T_{call_avg}, and the value will be set either "**LOW**" or "**HIGH**", where "**LOW**" and "**HIGH**" are adjustable definition constants. Moreover, the calculation of accumulated mobility attempts in current cell triggers off the calculations of total accumulated mobility attempts in adjacent cell which is illustrated in Figure 3.2, and the total accumulated mobility attempts of weight can be as the factor for adjusting the number of channels in the six adjacent cells, where "**Predict_basis**" is the critical value which can be adjusted according to the quality of service. The most difference among the proposed method and the past literatures is the adaptive threshold for the number of guard channels by using the total accumulated mobility attempt "$\tilde{A}(w)_j$"

Figure 3.2 The illustration of calculation for total accumulated mobility attempts

3.2 Performance Evaluation

In performance evaluation, the terms of dropping probability of handoff calls, blocking probability of a new call, and channel utilization are compared between predicted handoff and unpredicted handoff. Besides, the number of guard channels for predicted handoff calls are dynamically adapted and also shown in the experimental results.

The number of mobiles in our simulation is bigger than the channels in cell so that the traffic model to the cell can be approximated as a Poisson process [10] [11]. The related parameters are shown as follows.

- λ: call arrival rate according to Poisson process of rate λ.
- μ: the mean completion time assumed to exponential distribution with mean of $1/\mu$ [8].
- θ: the portable mobility and the dwell time is assumed to be exponential distribution with mean

of 1/θ.

In our experiments, the mean of call holding time $1/\mu$ is 6 min, and the mean of dwell time $1/\theta$ is 3 min. According to the assumptions of experiments, the handoff calls are 50% of total calls, and predicted handoff calls are 50% of handoff calls. The simulation results are shown in Figure 3.3, and it shows lower dropping probability for proposed method, but almost the same blocking probability and channel utilization. The number of guard channels is adjusted dynamically within the simulation process, thus the system can maintain the best utilization.
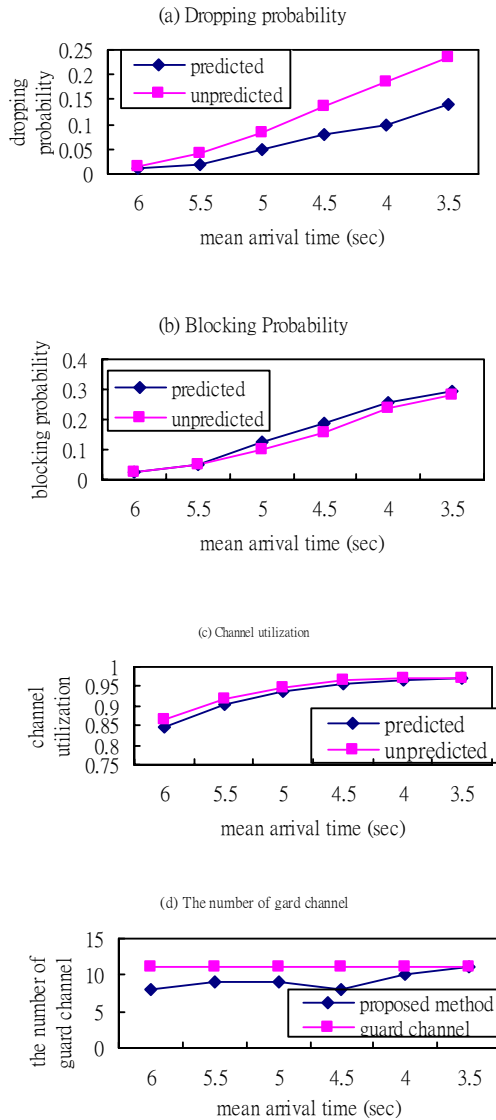

(a) Dropping probability

(b) Blocking Probability

(c) Channel utilization

(d) The number of gard channel

Figure 3.3 Simulation results

4. Conclusions and Future Works

In this paper, a new concept that uses accumulated user mobility is applied for prediction of handoff attempts, and the extenics is also introduced and applied to formulate the mobile velocity to the relational function. The accumulated user mobility that is derived from the speed of movement, dwell time, and move probability, is applied for adjusting the number of guard channels dynamically. The proposed scheme reserves the necessary number of channels for handoff calls and reduces the number of guard channels which can increase the utilization, hence, the call dropping probability can be decrease and QoS is also guaranteed. Furthermore, we will try to compare the QoS factors which use the different number of guard channels and observe the variations of the number of guard channels with others in future.

5. References

[1] Bezalel Gavish and Sepresh Sridar, "The impact of mobility on cellular network configuration," Wireless Network, Vol. 7, pp-173-pp-185, 2001.

[2] J. Misic, and T.Y. Bun, "On call level QoS guarantees under heterogeneous user mobilities in wireless multimedia networks," Proc. GLOBCOM'99, vol. 5, 1999, pp.2730-2736

[3] K.L. Yeung, and S. Nanda, "Channel management in micro/macrocell cellular radio systems," IEEE Tran. Veh. Tech., nol. 45(4), pp-601-612, 1996.

[4] Y. Fang and I. Chlamtac, "Traffic analysis and mobility modeling in PCS networks," IEEE Trans. Commun., nol. 47 (7), pp-1062-1072, 1999.

[5] Tajima and Imamura, K., "A Strategy for Flexible Channel Assignment in Mobile Communication Systems," *IEEE Vehicular Technology Transactions*, Vol. 37, May 1988.

[6] Hoo-ki Kim and Jae-il Jung, "The handover algorithm that considers the user's mobility pattern in wireless ATM," IEICE Trans. Fund., vol.E84-A, No.6, pp-1362-1369, June 2001.

[7] Cai, W., "Extension theory and its application", *Chinese Science Bulletin*, Vol. 44 No. 17, pp1538-1548, Sept. 1999.

[8] Rappaport, T. S., Wireless Communications Principles and Practice 2ed, Prentice Hall, Upper Sadle River NJ, 2000.

[9] Laurent Perato and Khaldoun Al Agha, "Handover prediction: User Approach versus Cell Approach,"

[10] Y.C. Kim, D.E. Lee, B.J. Lee, and Y.S. Kim, "Dynamic channel reservation based on mobility in wireless ATM networks," IEEE Commun. Mag., pp-47-51, Nov. 1999.

[11] Y. B. Lin, S. Mohan, and A. Noerpel, "PCS channel assignment strategies for handoff and initial access," IEEE Personal Commun., Third Quarter, pp-47-56, 1994.

A Design of Cluster Computing Environment for Multimedia Processing

[1]Liang-Teh Lee, [2]Ching-Ren Wei and [1]Chen-Feng Wu
[1]Department of Computer Science and Engineering, Tatung University
40 Chungshan N. Rd., Sec. 3, Taipei, Taiwan, R.O.C.
ltlee@cse.ttu.edu.tw
[2]Department of Applied Foreign Languages, Kuang Wu Institute of Technology
No. 151, I-teh Street, Peitou, 112 Taipei, Taiwan, R.O.C.
cjwei@mail.kwit.edu.tw

Abstract

With the rapid improvements in processors and network, the design of cluster has proven the potentials in low cost and high performance. In general, a high quality multimedia processing requires the huge computing power to deal with large amount of data within a certain period of time. In this paper, we propose the design of using a low cost, low power consumption and small-sized microprocessor as computing nodes to form a cluster system to increase the computing power and to accelerate the H.264 video processing. We analyze and distribute the computation in H.264 video processing, especially the part of heavy-load computation of decoder scheme, to each node of the cluster system in parallel. System analysis shows that, with the proposed design of cluster computing, the efficient real-time processing and high performance is achieved.

1. Introduction

With the widespread adoption of technologies such as streaming video in internet, DVD-Video and even digital television, the video compression has become an important issue and component in media transmission. In late 2001, ISO/IEC MPEG and ITU-T VCEG decided on a joint venture to wards enhancing standard video coding performance – specifically in the areas where bandwidth and/or storage capacity are limited. This Joint team of both standard organizations is called Join Video Team (JVT). The standard formed is called H.264/MPEG-4 part 10 and is presently referred to as JVT/H.26L/Advanced Video Coding (AVC) [1, 2]. H.264 can be widely used in video communication servers in IP network and wireless environment. The H.264 is an upcoming international video coding standard with superior objective and subjective image quality [3, 4]. The main goals of JVT are significant coding efficiency, simple syntax specifications and seamless integration of video coding into all current protocols and multiplex architectures (network friendliness). Requirements for H.264 arise from the various video applications that aims at supporting video streaming, video conferencing, over fixed and wireless networks and over different transport protocols, etc. H.264 features thus aim at meeting the requirements evolving from such applications. H.264 grouped its capabilities into profiles and levels – Baseline, Main and Extended profile. A "profile" is a subset of the entire bit stream of syntax that is specified by the international standard. Within each profile, there are a number of levels designed for a wide range of applications, bit rates, resolutions, qualities and services. A "level" has a specified set of constraints imposed on parameters in a bit stream. It is easier to design a decoder if the profile, level and hence the capabilities are known in advance. Baseline profile can be applied to video conferencing and video telephony, main profile can be applied to broadcast video, and extended profile can be applied to streaming media.

The video coding standard H.264/MPEG-4 part 10 is aimed to code video sequences at approximately half the bit rate in comparison with MPEG-2 at the same quality. The emerging standard also aims at obtaining significant improvements in coding efficiency, error robustness and network friendliness. Large computing power is required in multimedia processing due to large amount of data which are to be processed within a certain response to reach the quality. For the concern of increasing the effectiveness of the design, a full-scale H.264 system decoder with a low cost, low power consumption and small-sized microprocessor to construct the cluster for working collectively is proposed in this paper. Instead of the traditional PC cluster, we also propose the design of using a small-sized microprocessor as the node of the cluster computing system. With the proposed design and increased computing power in the system, the performing of highly complicated processing in multimedia applications within a low-end processor can be solved. Furthermore, by applying the SoC technology, the cost, power consumption and the size of the cluster computing system can be reduced significantly. For video coding, the latest H.264 standard is selected to implement the video decoder for accelerating the decoding speed. This paper is organized as follows: Section 2 introduces the adopted microprocessor. Section 3 describes the detailed design of the system architecture. Section 4 discusses how the video decoding works in the system. Finally, a conclusion is made in section 5 with more future work.

2. Microprocessor in cluster node

The microprocessor adopted in our proposed design is ARM7TDMI. It is a 32-bit microprocessor [5]. With the properties of good performance and low power consumption, it is very attractive for embedded
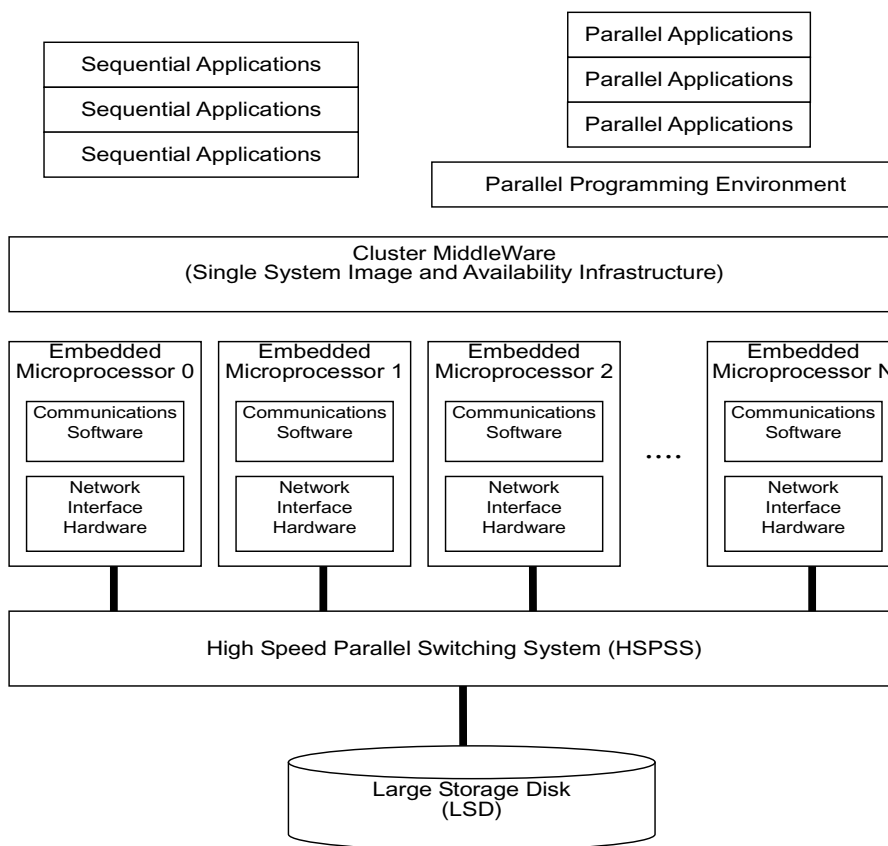
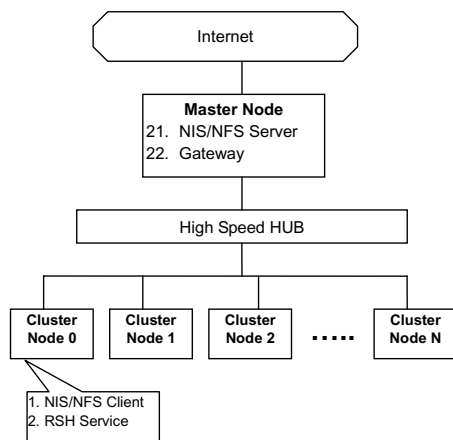Figure 1. High Speed Parallel Processing System Architecture

Figure 2. Proposed scheme of the cluster

applications. The microprocessor adopted in our system is a RISC architecture, the instruction set and related decode mechanism are much simpler than those shown in CISC architectures. The feature of a RISC architecture results in a high instruction throughput and impressive real-time interrupt response from a small and

cost-effective chip. This microprocessor, ARM7TDMI, has a three-stage pipeline, all parts of the processing and memory systems can operate continuously. The ARM memory interface can allow performance potential and without incurring high costs in the memory system. However, ARM7TDMI does not have an instruction or data cache; thus, it is mostly used as a controller core rather than for data processing.

3. System Architecture

Figure 1 presents the High Speed Parallel Processing System (HSPPS) architecture. We use the embedded microprocessor as a node to form a cluster. A high-speed parallel switching system is also provided to connect each node of the cluster. Because the embedded system is insufficient in storage facilities, a Large Storage Disk (LSD) is setup for each node in the system to share and store the immediate or final data. Moreover, we can setup a cluster middleware for supporting single system image and availability infrastructure. The proposed system provides a parallel programming environment for either sequential or parallel applications.

Each node of the cluster used in the current system is an ARM evaluation board, which is developed by MiceTek Company [6]. The operating system uClinux is ported in each board and the message-passing interface
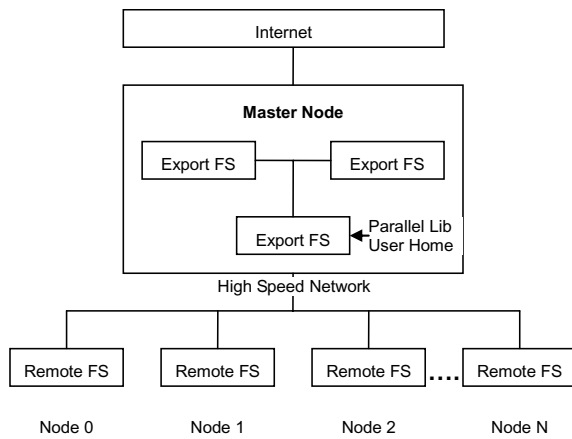
Figure 3. Software environment of cluster

(MPI) is applied for performing the distributed computing of the H.264 video decoding. Figure 2 shows the experimental scheme of the proposed system [7]. A personal computer is served as a master node to connect to the Internet for outward communication and manages the load balancing among all slave nodes. The master node must act as a NIS/NFS server, and the slave nodes are NIS/NFS clients with RSH service.

4. System Analysis

An abstract software environment of the proposed cluster is shown in Figure 3. The master server exports its "user home" shared with all slave nodes.

As described in the previous section, the H.264 standard consists of Profiles and Levels. The Profile specifies a set of algorithmic features and limits, which shall be supported by all decoders conforming to that profile. However, users may not require all features provided by H.264. The encoders are not required to make use of any particular set of features supported in a profile. For any given profile, levels generally correspond to processing power and memory capability on a codec. Each level may support a different picture size – QCIF, CIF, ITU-R 601 (SDTV), HDTV, S-HDTV, D-Cinema and data rate varying from a few tens of kilobits per second (Kbps) to

hundreds of megabits per second (Mbps). The non-integer level numbers are referred as "intermediate levels." All levels have the same status, but note that some applications may choose to use only the integer-numbered levels.

The H.264 standard codec, JM 8.0, is adopted for our wallet-size cluster [8]. Figure 4 shows the H.264 decoder scheme [9]. The basic functional elements (prediction, transform, quantization, entropy encoding) are slightly different from previous standards (MPEG1, MPEG2, MPEG4, H.261 and H.263); the important changes in H.264 occur in the details of each functional element. The decoder receives a compressed bit stream from the NAL. The data elements are entropy decoded and reordered to produce a set of quantized coefficients X. These are rescaled and inverse transformed to give D'n (this identical to the D' n shown in the Encoder). Using the header information decoded from the bit stream, the decoder creates a prediction macro block P, identical to the original prediction P formed in the encoder. P is added to D'n to produce uF'n which is filtered to create the decoded macro block F'n.

According to the scheme described above, the processing complexity of each module in the decoder can further be estimated. By using the "Intel VTune Performance Analyzer" to analyze the percentage of execution time of each module in the decoder when the baseline profile is considered, the experimental result is presented in Figure 5 [10, 11]. The result shown in Figure 5 is obtained by decoding a 4CIF resolution video with Qstep 30. The most critical time modules are in sequence motion compensation (41.88%), entropy coding (28.25%), deblocking (10.48%), and integer transform (10.33%). Obviously, by applying the proposed scheme for parallel and distributed processing can speedup the video decoding for fulfilling the real time requirement. For evaluating the performance of the system, H.264 standard codec JM 8.0 is adopted for our experiment to measure the following data: PSNR, bit rate, complexity reduction of the testing sequences, and decoding time. Furthermore, Intel Vtune Performance Analyzer will be used to measure the computational cost in terms of number of clock cycles used.
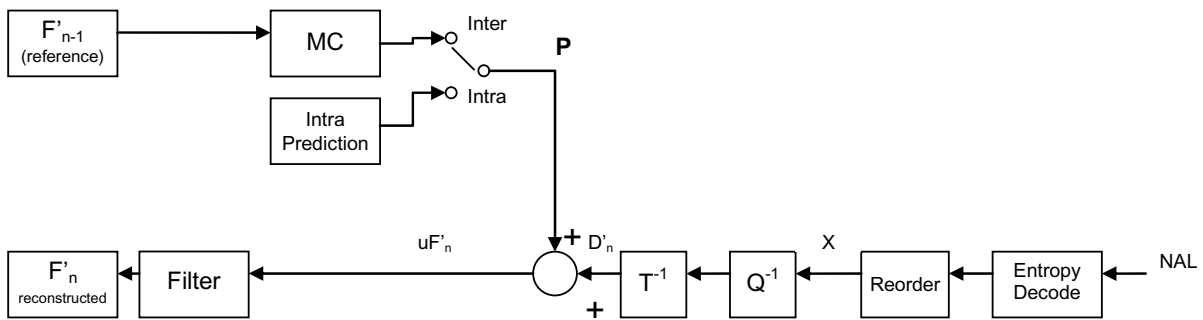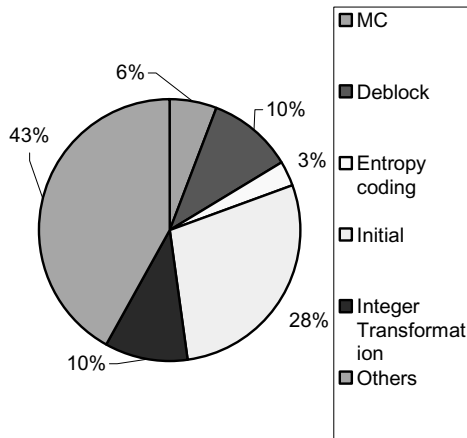

Figure 4. The H.264 decoder

Figure 5. The complexity of each module in H.264 decoder

5. Conclusions

Comparing with the previous standards, H.264 is more flexible and can improve the coding efficiency. However, it should be noted that this is the expense of added complexity to the encoder/decoder. Also, it is not backward compatible to the previous standards. The level of complexity of the decoder can be reduced by designing the specific profile and level. In this paper, we propose a scalable wallet-size cluster computing system with the low cost and low power consumption. By clustering the embedded system to increase the computing power, the system can be applied to multimedia applications to speedup the video decoding for fulfilling the real time requirements. In the future, we will explore H.264 decoder and the methodology for supporting multimedia communication and buffer management between server and clients. More works on evaluating the performance of the system will be made.

References

[1.] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264/ISO/IEC 14 496-10 AVC), Mar. 2003.

[2.] A. Tamhankar and K. R. Rao, "An overview of H.264/MPEG-4 Part 10," The 4th EURASIP Conference focused on Video/Image Processing and Multimedia Communications, Vol. 1, pp.1-51 July 2003.

[3.] Chang, O. C. An, and Y. M. Yeung, "A novel approach to fast multi-frame selection for H.264 video coding," Proceedings of 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 3, pp. 413-416, April 2003

[4.] Tamhankar and K. R. Rao, "An overview of H.264/MPEG-4 Part 10," The 4th EURASIP Conference focused on Video/Image Processing and Multimedia Communications, Vol. 1, pp.1-51 July 2003.

[5.] "ARM7TDMI Data Sheet," ARM DDI 0029E, August 1995.

[6.] http://micetek.com/

[7.] http://web.csie.chu.edu.tw/~cs87668/cluster.htm

[8.] http://bs.hhi.de/~suehring/tml/download/

[9.] http://www.vcodex.fsnet.co.uk/h264_overview.pdf

[10.] Song-Wen Wang and Chia-Lin Wu, "Design and Implementation of H.264/AVC Digital Image Decoder System," Jun 2002.

[11.] http://www.intel.com/software/products/vtune/

[12.] K. Ramkishor and V. Gunashree, "Real time implementation of MPEG-4 video decoder on ARM7TDMI," Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing, May 2001

Formation and Routing Protocols for Bluetooth Scatternets with Heap Structures*

Tzung-Shi Chen♣ and Hsin-Yi Ke

Department of e-Learning Technology
National Tainan Teachers College
Tainan 700, Taiwan
chents@ipx.ntntc.edu.tw

Abstract

In this paper we propose a new structure for Bluetooth scatternet formation and its routing protocol. We adopt heap tree structure as the backbone to form the Bluetooth scatternet, called BlueHeap. We assume the Bluetooth devices unique address as the key for formation BlueHeap. BlueHeap allows nodes joining or leaving the network arbitrarily, and it can be based on the heap property to rapidly recovery the scatternet. In routing aspect, we assume the source node knows the destination node's position in the BlueHeap. We use the position to compute the shortest path between source node and destination node. We can say the BlueHeap has the capability of self-routing.
Keywords: Bluetooth, Heap tree, Routing, Scatternet, Wireless Networks.

1. Introduction

Ubiquitous computing is emphasized in recent years. There are many types of wireless network have been proposed in recent years, such as WLAN, ad hoc network, sensor network, Bluetooth and so on. Bluetooth [1] is one of wireless technology and it is an emerging Personal Area Network (PAN) technology. Bluetooth is a single chip, which can adds in many kinds of mobile devices like PDAs, laptops, mobile phones or digital cameras, etc. It emphasized on low-power, low-complexity, and low cost. According to the specification design, Bluetooth technology can apply to any mobile devices.

Bluetooth scatternet formation and routing protocol are popular issues for research. There are several criteria should be noticed. In a scatternet, the number of piconets should be minimized that can provide quickly routing. In order to reduce switching overheads, a node participate in at most two piconets. The scatternet should provide good mobility and fault tolerant capability. Two piconets connect by just one bridge to reduce redundant inter piconet. The degree of the node should be limited and it can avoid parking any nodes. In [2], the authors proposed BlueStar for multi-hop Bluetooth network. Chang [4] utilizes Hypercube to construct the Bluetooth scatternet and called BlueCube. BlueCube let routing and communication become easy, but if some nodes want to join or leave the BlueCube may let the network

failed. In [11] and [12] are proposed tree based scatternet construction protocols. In [5] and [7], the authors both construct Bluetooth scatternet with ring structure. The advantages of ring are reliability and routing easily. Nevertheless, if there are too many nodes join to the scatternet of ring, it will let the routing path become too long.

In our study, we focus on the subjects of scatternet formation and routing protocol. We propose a new scatternet structure prone to establish, maintenance and recovery. We take advantage of heap tree to form the Bluetooth scatternet and called BlueHeap (BH). Heap tree is a simple data structure and can tolerate nodes arbitrarily joining or leaving. The structure of scatternet is distributed. We assume Bluetooth unique BD_ADDR as the key to form the BlueHeap. Except original links of the tree, we also build sibling and non-sibling node links for routing. The nodes of BlueHeap backbone are all masters. It means every node of BlueHeap is also a piconet. In order to avoid too frequently role switching, we assign bridge node between two piconets [10]. The bridges can reduce power consuming and transmitting time, which happen in role switch.

The rest of the paper is organized as follow. In session 2, we will refer to some scatternet formation and routing protocols that has been proposed. In session 3, we propose the new scatternet formation structure and in session 4 we detail the routing protocol of BlueHeap. In session 5 shows the result comparing with related work. In last session conclusion and discuss our future work.

2. Related Work

There are many kinds of topology have been proposed for scatternet formation. In this section, we review some scatternet formation scheme and routing protocol.

In order to enhance the Bluetooth structure reliability, some scholars adopt ring structure to form the Bluetooth scatternet. Ring is a simple structure and it can reduce the scatternet formation complexity. In [5], the target is using ring to evaluate the scatternet it would to enhance the network reliability. But nodes of the Bluering structure all play two roles that are master and slave.

According to the specification [1], one node can be active state only in one piconet. If the node plays two roles it has to switch the hopping frequency sequence from one piconet to the other. This motion may extend the transmit time and overhead. In [7] the nodes of the ring are also all masters and they through the bridges that they designed S/S to connect every piconet. Both two Bluerings provide easily routing scheme for routing. The routing of Bluering is one direction and not holds any routing table. But Bluering has a potential problem. If there are too large numbers of nodes join to the ring structure, the ring may become very long. It means the Bluering length will be long and it could lead to increase packet delays ratio.

Hypercube is a parallel computing structure and it supports disjoint paths and tolerates faults. In [4], the authors make use of the hypercube structure to form the scatternet and called BlueCube. The BlueCube is a distributed scatternet that does not need to select one node as the coordinator. The BlueCube construction process is simple. In routing, the BlueCube provides more than one path for routing and routing in the BlueCube is faster. There are still some drawbacks in the BlueCube. The BlueCube structure is robust and if nodes would join or leave the network may cause the network failure. Song *et al*. [10] proposed that use de Bruijn graph as the backbone to form the Bluetooth scatternet. The dBBlue provides the diameter of the graph is O (log *n*) and find a path with at most O (log *n*) hops for every pair of nodes. According to the de Bruijn graph property it let the dBBlue have self-routing capability.

Tree structure is popular for people to adopt forming Bluetooth scatternet. In [11] [12], they both use tree structure to construct the scatternet. The initially phase of construction Bluetree is choosing a single node to be the root, that called blueroot. Because every master node can only have 7 active nodes, the Bluetree should limit the node of slaves. They use simple geometric to reconfigure and limit the number of slaves greater than 5. In routing aspect, very path finding must through the blueroot; it should increase the load of the blueroot. The blueroot becomes the bottleneck of the communication. In [11], in order to let the scatternet have self-routing capability, it makes use of 7-ways search tree as the Bluetooth scatternet. A self-routing scatternet can keep routing overhead minimum. The topology that proposed can deal with networks of large sizes but it need relies on a complex merger procedure. They design a locking mechanism for node joining. In search tree structure, they cannot guarantee the final Bluetree is balances.

Salonidis *et al*. [9] proposed a construction algorithm for Bluetooth scatternet formation. The first construction phase is coordinator election. When the coordinator has been elected, it should responsible for collect whole scatternet information and then to execute the phase of role determination. The method is centralized, and the coordinator need bear large load. For multihop ad hoc network based on Bluetooth scatternet, the papers [2] [6] proposed that use star structure as the Bluetooth scatternet. In [2], the BlueStar constructed by three phase, first is topology discovery, second is BlueStar formation, and the final is selection of gateways devices to connect multiple BlueStar. The BlueStar is also a centralized Bluetooth scatternet. The master of the BlueStar has a complete knowledge of its neighbors' role and of the ID of their master. The paper [6] is used a hierarchical Bluetooth scatternet architecture for wireless home networks. They design two levels for forming a Bluetooth scatternet and the level 1 are devices communication among themselves in the piconet, level 2 piconet is primarily used for providing interconnectivity for routing. In [6], one piconet is in the center and its slaves are the bridges to connect other masters. Non-central piconets do not extend to more piconets. So this is a two level hierarchy. All inter-piconet traffic must go through the central piconet, and this may be the bottleneck of the traffic. But in other aspect, using this structure has less average number of hops for routing. In both centralized methods, if the coordinator or the master of the BlueStar occurs failure, the whole s may be failure.

Routing is another important subject for Bluetooth. A prefect scatternet not only provide quickly construct scatternet but also can routing faster. A famous routing in Bluetooth is routing vector method (RVM) [3]. It provides fast route discovery but it also has some limits like it only use in unicast service. RVM is a centralized method and using flooding scheme to find the destination for routing. Using flooding scheme will lead the packet overhead become large and raises power and bandwidth consuming.

3.　BlueHeap Formation

In this section, we propose a new Bluetooth scatternet formation method. We adopt the heap tree structure to be the backbone [10] of the Bluetooth scatternet. In order avoiding the root as the bottleneck, besides the original link of the tree structure, we build the link of the sibling and non-sibling nodes for routing. There are three roles play in the BlueHeap: master, pure slave and bridge. Every backbone node is master and the bridge node is assigned by the master to connect two masters. The slaves that are not be assigned as the bridges are pure slaves, and the pure slaves cannot be inquired by other master. In the BlueHeap, the level is defined from up to low and it means the root present the top level of the BlueHeap. The BlueHeap topology shows in Figure1.
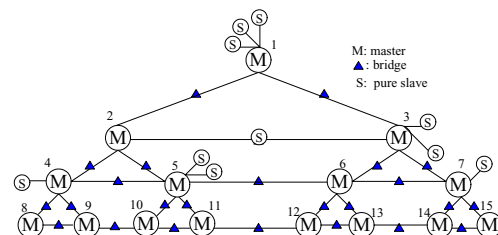

Figure1: BlueHeap Topology.

3.2 BlueHeap Construction

In this subsection, we shall detail our method of forming the Bluetooth scatternet. We adopt the max-heap tree as our BlueHeap. Before we describe the BlueHeap construction process, we propose some assumptions for our BlueHeap:
(1) When a node is assigned the role of the bridge, it can not switch back to inquiry state and one bridge just can serve between two piconets.
(2) In the BlueHeap, every node knows the positions about its parent, leaf and neighboring nodes.
(3) Nodes are all in the transmission ranges.

We use a centralized formation mechanism to form our BlueHeap, like [9]. Firstly, we assign a single node as the root of the BlueHeap, see the Figure 2, but this root is uncertain a root for the final BlueHeap. According to the heap tree's structure, they sometimes shall base on the key to adjust position and find the fit position. We assume the unique Bluetooth device address (BD_ADDR) as the key and adopt it to form the BlueHeap and according the key to adjust the position. We give the token to the node which is constructing the BlueHeap. There are two purposes for assigning token; one is to know the node whether has two children nodes or not, the other is to know the last node of the BlueHeap. If the node has two children nodes, it must pass the token to the next node and continues to construct the BlueHeap. The new node that wants to join to the BlueHeap must inquiry the bridge node and exchanges the information through the bridge node for comparing the key. Beside the bridge node, the pure slaves of the piconet cannot be inquired. If there are no new nodes want to join to the BlueHeap, the bridge stopping be inquired till time out and it will be retracted by the master and return to pure slave. If there no new nodes join to the BlueHeap till time out, the node with the token is the last node's parent node. Every node of the BlueHeap keeps its level mark k and position symbol x to remember its level and position. Through the level we can easily compute the position range of every level. If node n_z in level k we can compute the range level:

$$\begin{cases} k \geq 1 \\ 2^{k-1} \leq z \leq 2^k - 1 \end{cases}$$

The position symbol lets node joining or leaving the BlueHeap can be adjusted easily. Base on the symbol of position, nodes can easily to know which one is their parent node. When a node joins to the BlueHeap, then they will compare the information of BD_ADDR. There are two position adjusting conditions may occur during the scatternet construct procedure.

Case I: If the new node's BD_ADDR is smaller then the original backbone node of the BlueHeap, the node will directly join into the BlueHeap. See the Figure 2, we firstly assign a root node and the new node with key 97 and 95 are smaller than the 98, so they can join to the BlueHeap directly.
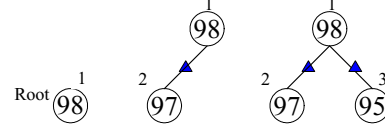

Figure 2: The construction of the BlueHeap.

Case II: During the procedure of constructing the BlueHeap, if the new node's key is bigger than the original backbone node, they must adjust the position. When nodes begin exchange the position, the masters need notice the bridge nodes disconnect the link with other piconet and do DIAC inquiry [1]. Bluetooth specification reserved 63 IACs (DIACs) for dedicated inquiry operations. See the Figure 3. It shows node n_7 with key 100 that wants to join to the BlueHeap and it inquiries the bridge of n_3 to join. In this condition, the node with key 100 is bigger than parent node, so it must perform position adjusting. The node, which holds the token shall send the token to the node that replace the original position. And continue to construct the BlueHeap.

Figure 3: Node n_7 with key 100 joins to the BlueHeap.

If the BlueHeap has been constructed completely, then they implement the link construction procedure. Beside the original links of the BlueHeap, we establish the links with siblings and non-sibling nodes. In building sibling nodes links, the one of leaf node can through its parent node to know whether it has sibling node or not. If it has sibling node, then assigns a pure slave as the bridge and execute the inquiry procedure to construct the link. During building the non-sibling nodes link, the node shall through their parent nodes to notice the non-sibling node for establishing the link. Figure 4 shows the BlueHeap that have been constructed the sibling and non-sibling links.

Figure 4: Construct sibling and non-sibling links.

3.3 BlueHeap Maintenance

Bluetooth has the property as ad hoc network, so the devices may join or leave the communication range eternally. Heap tree provides a good property to recovery the link of the structure when nodes joining or leaving.

3.3.1 Node Leaving

In node leaving procedure, the position adjusting is

not executed absolutely. If the last node's key is bigger than the failure node's key and smaller than the parent node's key, it can replace directly. We can see the Figure 5.

Figure 5: The example of node leaving.

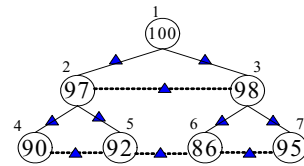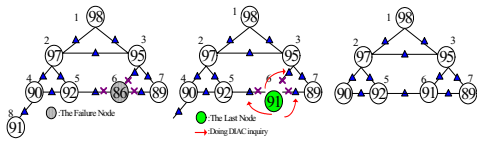When the last node's is bigger than the failure node's key, they shall adjust the position. The Figure 6 illustrates an example of a node failure and position adjusting. Node n_6 leaves the network and the last node n_{15} replace the node n_6's position. Then node n_6 executes the node changing procedure and comparing the key with its child nodes. Then it changes the position with the node that has the bigger key till every node in the fit position. The position adjusting is similar to the above article that we proposed when constructing the BlueHeap. They use DIAC inquiry to adjusting position and rebuilding the links.
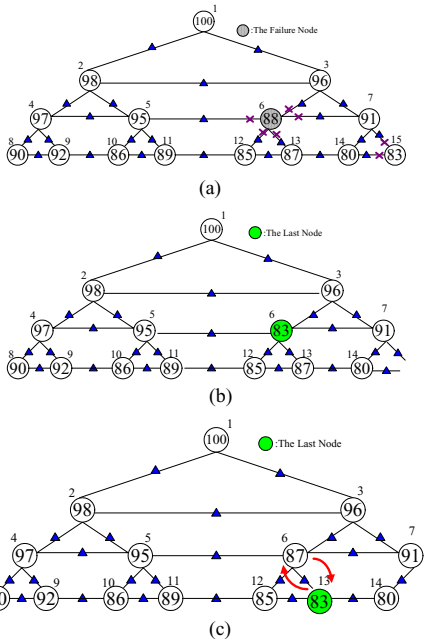

Figure 6: The example of node leaving: (a) node n_6 leaves the network, (b) node n_{15} replaces the position of failure node n_6 and (c) node n_{13} changes the position with the node n_5.

3.3.2 Node Joining

The node joining procedure is similar to the constructing process. They firstly compare the keys, if the new joining node's key is bigger than its present node, they must begin node position adjusting process. Given an example, if new node with key 99 wants to join to the BlueHeap, it rests on the token to find the last position of the scatternet. See the Figure 7. The new node with key 99 inquires the bridge node, which belongs to the node that holds the token and receives the

inquiry response. Its parent node with key 90 is smaller than the new node key 99, so it needs to execute position adjusting. The nodes compare their key till the node with the key is bigger than the new node with key 99. Than they disconnect the original links and execute DIAC inquiry. The position adjusting is not simultaneous. It executes step by step.

(a)

(b)

Figure 7: (a) New node n_2 joins to the BlueHeap, and position exchanges process and (c) new node n_2 has been joins to the BlueHeap.

3.3.3 Bridge Failure

Sometimes, the failure may occur in bridge nodes. In this situation, the bridge node must be reassigned by its original master. If the original master does not have any other pure slaves to reassign, it will implement DIAC inquiry. The other master, which connect with the same failure bridge assign a pure slave to be discovered and go into DIAC inquiry scan.

4 BlueHeap Routing Protocol

Routing is an important issue for Bluetooth. In this chapter, we shall detail the routing protocol in BlueHeap. The routing protocol is similar to AODV [8]. The routing path establishs on-demand.

4.1 BlueHeap Routing Protocol

Before we detail the routing protocol, we firstly assume that the source node knows the destination node's position in the BlueHeap.

Here, we utilize the position computing to construction the routing path. In the BlueHeap, nodes hold position and level mark, so we can through these two values to compute and find the optimal routing path. When the destination node receives the relay packet, it can base on it position to know whether the source node in the same level or not. The count of the range is based on the formula $2^{k-1} \leq z \leq 2^k - 1$. If the source node's position is smaller than the destination node's and not in the same level, the destination node replies to its parent node and repeat the motion source node's level and find it. We can establish the path for routing. During the routing path construction process, every node holds a

routing table toe record the next hop. The routing has three fields are: P (position of itself), key (BD_ADDR) and N-hop (next hop's position) Bellow, we bring up three cases to detail the BlueHeap routing protocol.

Case I: The destination and source nodes are in the same level. We use the example in Figure 8 to illustrate; node n_{18} wants to communicate with node n_{26}. The destination n_{26} is in the same level with the source node n_{18}. In order to find the optimal routing path, they can use the formula that we proposed to compute the optimal routing path. The parameters are in the Table 1.

Table 1: The parameters.

Paramter	Defination
l	The hop of the level
w	The number of hops between two nodes
t	The optimal hops for routing

In this case n_{18} and n_{26} are between 8 hops (=26-18), and it not the optimal routing path. So n_{26} need transmit the reply packet to the upper level and find the optimal routing path. We can use the formula bellow:

$$\begin{cases} l = \lceil \lg w \rceil - 2 \\ t = 2l + \left\lceil \dfrac{w}{2^l} \right\rceil \end{cases}$$

Firstly, we need to know how many levels so the l is $l = \lceil \lg 8 \rceil - 2 = 1$. We just need transmit to one upper level. Than we can find the total optimal hops for routing is $t = 2*1 + \left\lceil 8/2^1 \right\rceil = 6$.
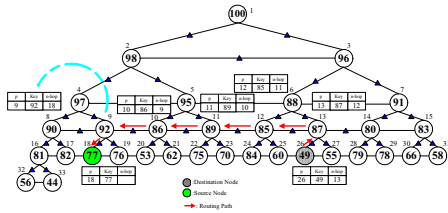

Figure 8: The destination node in the same level with the source node.

Case II: The second condition is the destination node in the upper level of the BlueHeap. In the beginning, the source node can base on the destination node's position to know the destination node in the upper level of it. The source node sends the request packets to parent node and repeat this motion till the node that in the same level with the destination node. When the request packets be transmitted to the node that in the same level with destination node, its shell continue to be transmitted to the destination node through the link of sibling and non-sibling nodes. See the Figure 9.

Figure 9: The destination node in the upper level of the source node.

Case III: The third situation is the destination node is in the lower level of the source node. This situation is the destination node's key is smaller than the source node's. We assume that n_2 wants to transmit data to node n_{14} with key 80. Because the destination node in the low level of the source node so it needs to compute the range to find the next hop node for transmit the request packets. The Figure 10 shows this condition.

Figure 10: The destination node in the lower level of the source node

In BlueHeap, we use the position to find the optimal routing path so we can say the BlueHeap has the self-routing capability.

5 Comparisons with Related Work

In this section, we compare the routing path length with Bluering and BlueStar. The Figure 11 shows the result of the comparison, we can see the BlueHeap routing path has shorter than Bluering but longer than BlueStar. The reason is in the BlueHeap exists one special case in routing that is the source node and the destination node are situated in the same level. In the tree structure, when the levels increase, the nodes number also increase. Although we can find the optimal path length, but the path length still increase a little.

Figure 11: The routing path length comparing.

A well scatternet structure needs support not only one disjoint path for routing. When this condition happened, the scatternet occur failure or the routing path may disconnect. Although the BlueStar provides the shortest path for routing but it exists the potential problem. The BlueStar has only one path for routing. In BlueHeap, it constructed base the heap tree and we build the sibling and non-sibling links so it let the scatternet exists more than one disjoint path. In the bellow Figure 12, we can clear to see the disjoint paths of the BlueHeap, in Figure 12 we ignore the bridge in it..

Figure 12: Node disjoint paths.

The ring structure only provides at most two disjoint paths for routing and the BlueStar provides only one path for routing. We count the disjoint paths in Table 2.

Table 2: Node disjoint paths comparison.

Scatternet structure	Number of node disjoint path
BlueHeap	1~5
Bluering	1~2
BlueStar	1

Beside the comparison that we show the advantages of the BlueHeap, there are many benefits to use the BlueHeap to form the Bluetooth scatternet. According to the specification, the Bluetooth devices communication must in the range of each other. In the BlueHeap, just need two nodes, which are neighboring nodes in the communication range and not all nodes in the communication of each other. The BlueHeap can enlarger and accommodate more devices.

6 Conclusion and Future Works

In this paper, we proposed a new Bluetooth scatternet formation and routing protocol. We adopt the heap tree structure to form the scatternet and called BlueHeap. BlueHeap is based on the standard heap structure to establish Bluetooth device is emphasized on mobility so the scatternet may change frequently. Heap tree has a property is it can dynamic adjust it structure and recovery to the original state. According this, when some nodes want to join or leave the BlueHeap, we can base on the key and token to execute the position exchanging and recovery the scatternet to the normal state. During the construction process, every backbone node of the BlueHeap needs to record its position, which side and level mark for routing.

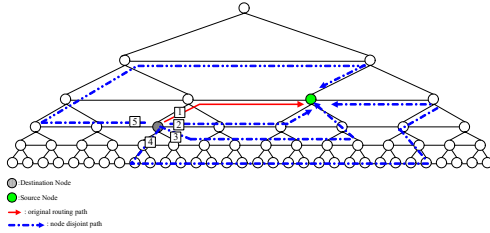In routing aspect, the tree structure has a bottleneck is every routing path must through the root node. In order to enhance the routing efficiency and disperse this bottleneck problem so we established sibling nodes and non-sibling nodes links. We assume the source node knows the destination node's position and through the computing to find the optimal routing path.

In the future work, we will detail focus on the BlueHeap variation and we hope we can measure the routing preference in ns-2.

References

[1] Bluetooth Special Interest Group, "Specification of the Bluetooth System", Core, version 1.2, http://www.bluetooth.com/, November 2003.

[2] S. Basagni and C. Petrioli, "Multihop Scatternet Formation for Bluetooth Networks," *in Proceedings of Vehicular Technology Conference* (VTC 2002), Vol. 1, pp. 424-428, Birmingham, Alabama, USA, May 2002.

[3] P. Bhagwat and A. Segall, "A Routing Vector Method (RVM) for Routing in Bluetooth Scatternets," *in Proceedings of International Workshop on IEEE Mobile Multimedia Communications International Workshop* (MoMuC 1999), pp. 375-379, San Diego, California, USA, November 1999.

[4] C.-T. Chang, C.-Y. Chang, and J.-P. Sheu, "BlueCube: Constructing a Hypercube Parallel Computing and Communication Environment over Bluetooth Radio System," *in Proceedings of International Conference on Parallel Processing* (ICPP 2003), pp. 447-454, Kaoshiung, Taiwan, October 2003.

[5] C.-C. Foo and K.-C. Chua, "Bluerings-Bluetooth Scatternets with Ring Structures," *in Proceedings of IASTED International Conference on Wireless and Optical Communications* (WOC 2002), Banff, Alberta, Canada, July 2002.

[6] W. Lilakiatsakun and A. Seneviratne, "Wireless Home Networks based on a Herarchical Bluetooth Scatternet Architecture, " *in Proceedings of International Conference on IEEE Int'l Conference on Networks* (ICON 2001), pp. 481-485, Bangkok, Thailand, October 2001.

[7] T.-Y. Lin, Y.-C. Tseng, K.-M. Chang, and C.-L. Tu, "Formation, Routing, and Maintenance Protocols for the Bluering Scatternet of Bluetooths," *in Proceedings of* International *Conference on System Sciences* (ICSS 2003), pp. 313-322, Hawaii, USA, January 2003.

[8] C.E. Perkins and E.M. Royer, "Ad-hoc on-demand distance vector routing" *in Proceedings of Mobile Computing Systems and Applications Workshop* (WMCSA 1999), pp. 90-100, New Orleans, Louisiana, USA, February 1999.

[9] T. Salonidis, P. Bhagwat, L. Tassiulas, and R. LaMaire, "Distributed Topology Construction of Bluetooth Personal Area Networks," *in Proceedings of Conference on the IEEE Computer and Communications Societies* (**I**NFOCOM 2001), Vol. 3, pp. 1577-1586, Anchorage, Alaska, USA, April 2001.

[10] W.-Z. Song, X.-Y. Li, Y. Wang, and W. Wang, "dBBlue：Low Diameter and Self-routing Bluetooth Scatternet," *in Proceedings of The Joint Workshop on Foundations of Mobile* Computing (DIALMPOMC 2003), pp. 22-31, San Diego, California, USA, September 2003.

[11] M.-T. Sun, C.-K. Chang, and T.-H. Lai, "A Self-Routing Topology for Bluetooth Scatternets," *in Proceedings of International Symposium on Parallel Architectures, Algorithms and Networks* (ISPAN 2002), pp. 13-18, Makati City, Metro Manila, Philippines, May 2002.

[12] G. V. Zaruba, S. Basagni, and I. Chlamtac, "Bluetrees-Scatternet Formation to Enable Bluetooth–Based Ad Hoc Networks," *in Proceedings of International Conference on Communications* (ICC 2001), Vol. 1, pp. 273-277, St.-Petersburg, Russia, June 2001.

Adaptive Layer-Based Scheduling for Real-Time Transmission on Scalable Multimedia Stream

Chia-Ying Tseng, Liang-Teh Lee, Yu-Lan Shih, and Kang-Yuan Liu
Department of Computer Science and Engineering, Tatung University
40 Chungshan N. Rd., Sec. 3, Taipei, Taiwan
cytseng@cse.ttu.edu.tw

ABSTRACT

With the recent development in compression and network technology, stream media has adopted in internet and intranet. Application of stream media technology will play a key role in future development of fast network. Original video/audio data will be stored in the devices for storage after having been pre-compressed by video/audio compression algorithm. When clients have requirements, stream server retrieves the video/audio data from the storage devices through the network. In this paper, we proposed and implemented a simple and effective real-time scheduling algorithm, adaptive Layer-Based Least-Laxity -First (LB-LLF) scheduling algorithm, to improve the output quality of video/audio on network and to achieve synchronized playback effect. The proposed algorithm considered real-time constraint, unequal priorities of scalable media stream in different layers, and a good trade-off between coding efficiency and drifting error. This guarantees the effective usage of available channel bandwidth and the better quality of playback in client.

1. INTRODUCTION

With the steady increase in the access bandwidth, more and more internet applications start to use the streaming audio and video contents. In response to the increasing demand on streaming video applications over the best-effort internet, the coding objective for streaming video is changed to optimize the video quality for a wide range of bit rates[1]. Generally, video compression can be divided into two categories, scalable compression and unscalable compression. Unscalable compression generates only one bit-stream, but scalable compression can generate multiple substreams, including one basic bit-stream. The basic bit-stream can do decoding independently and output poor-quality video streams, and the other substreams help improving the output quality. All the streams work together to output the best-quality video streams. Similarly, if only parts of the substreams (the basic bit-stream must be included) work together, the output video streams will be relatively poor in terms of the image quality. It is called back to ten years ago about researching of the layered scalable coding, and has adopted many international compression standard at present, such as MPEG-4 and H.263. Generally, the layered scalable coding can be divided into three categories: temporal scalable, spatial

scalable, and quality scalable. These three aspects result to SNR (Signal to Noise Ratio) and scalable encoding in temporal and spatial. One or more than one of the above scalable compressions can be adopted to serve the purpose of scalability. In order to provide better flexibility and meet different demands of delay, a new compression technique called FGS has been submitted to MPEG-4.

The Fine Granularity Scalable (FGS) [2] video coding adopted in MPEG-4 standard is a technique, only the base layer is predicted from the reconstructed base layer of a reference frame in the proposed scheme, and all enhancement layers are predicted from an enhancement layer of the reference frame. Because of the fine granularity scalability provided by the bit-plane coding technique in the enhancement layer, the FGS scheme can easily adapt to channel bandwidth fluctuations. However, since its motion prediction is always based on the lowest quality base layer, the coding efficiency of the FGS is not so good as, and sometimes worse than, the traditional SNR scalable coding. Compared with the non-scalable video coding schemes, the PSNR of the FGS may drop 2.0dB or more at the same bit rate.

The Progressive Fine Granularity Scalable (PFGS) coding scheme is an improvement over the FGS scheme. Similarly to FGS, PFGS coding also encodes video data frames into multiple layers, including a base layer of relatively lower quality video and multiple enhancement layers of increasingly higher quality video (some also refer to all the enhancement layers as a single enhancement layer with multiple bit-planes, but we prefer to refer to them as multiple layers). A typical framework of the PFGS is shown in Figure 1. The base layer, the first, the second, and the third enhancement layer of frame 2 are predicted from the base layer of frame 1, and the other higher quality enhancement layers of frame 2 are predicted from the fourth enhancement layer of frame 1, and so on. The coding efficiency of the PFGS can be up to 1.0dB higher in average PSNR than that of the FGS at moderate or high bit rates because of the use of high quality references in the enhancement layer coding. In general, it is reasonable to assume that the base layer is always available in the decoder. However, when network bandwidth drops, the decoder may partially or completely lose the high quality references. [3]
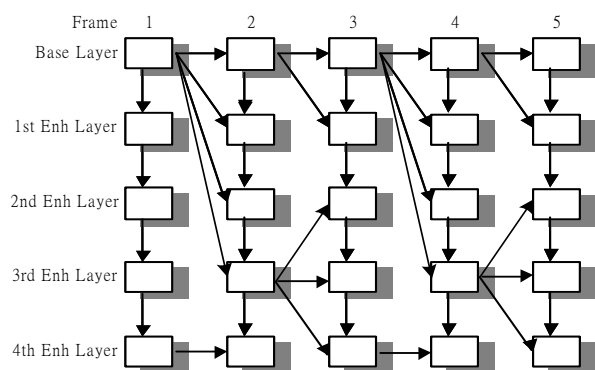
Figure 1: The framework of PFGS

In this paper, we proposed and implemented a simple and effective real-time scheduling algorithm, adaptive Layer-Based Least-Laxity-First (LB-LLF) scheduling algorithm, for delivery of scalable streaming media over lossy channel. It is a priority-based real-time scheduling scheme to select the packets to be transmitted at a given time. Furthermore, in real-time constraint, unequal priorities of scalable media stream in different layers to achieve a good trade-off between coding efficiency and drifting error are considered in the proposed algorithm. The algorithm can greatly improve the playback quality in the client.

2. THE SYSTEM ARCHITECTURE

To increase the traffic quality for streaming media in the internet, a server transmits multimedia stream packets to a receiver that buffers these packets for playback. A typical streaming system consists of clients and servers on a network. Figure 2 shows the architecture of the scalable streaming media system. The client requests are sent to the server via network connections which also serve for transmission of media data. The buffers in each client are used to provide some tolerance on variations in network delay as well as data consumption rates. The scheduler in the server controls the packet size and sequence, manages the server transmitted buffer and packets via the network to the clients' buffers. The media sequence consists of many frames, which are compressed into several layers. The layers are packed as *packet* and fed into the server's transmission buffer. These are the packets waiting to be scheduled for transmission. The server's scheduler selects one candidate packet at a time from those buffers and sends it to the network channel. [4] [5]

3. REAL-TIME SCHEDULING ALGORITHMS

3.1 Basic Technologies
Earliest Deadline First (EDF) and Least-Laxity-First algorithm (LLF) have been proven to be optimal dynamic scheduling algorithms [6] [7]. A main advantage of these dynamic scheduling algorithms can achieve a theoretically

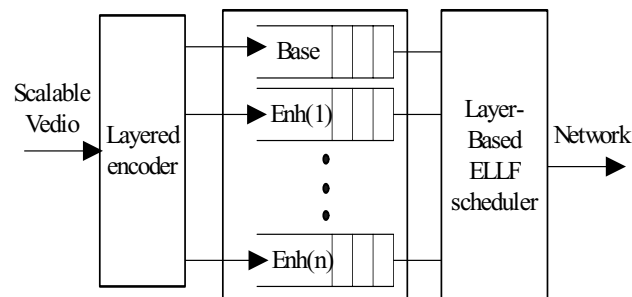possible 100% processor utilization without deadline misses.

Figure 2: The architecture of PFGS video streaming system

From the implementation point of view, it is easier to realize the EDF than the LLF algorithm. But both algorithms have also some drawbacks. In an overload situation all tasks of the given task sets miss their deadline in the EDF algorithms (known as the domino effect). Another drawback is that the EDF algorithm is possible to detect the violation of a deadline only after it has happened. The LLF algorithm is a dynamic scheduling method, i.e., it makes the decision for which task to execute next at scheduling time. For every task ready to run at the given moment the difference L between the time of deadline D and the end of processing time P is computed. This difference is called as laxity or slack which can be seen as an inverted priority value. The task with the smallest L-value is the one to be executed next. Whenever a task other than the currently running one has the smallest laxity, a context switch will occur. LLF algorithm is an optimal scheduling method. That means, if a given set of tasks is schedulable, then it can be scheduled by Least-Laxity-First. Another great advantage of the LLF algorithm is that no further assignment for fixed priorities to the tasks at development time except scheduling test. Furthermore, a task going to miss its deadline is recognized at the same moment when its laxity turns to zero with the task currently not being executed. At that time the deadline is not yet reached and emergency measures can be taken to cover the miss of a deadline. With LLF it is possible for the scheduler to detect an impending deadline miss during the execution of tasks. [8] [9]

3.2 Transmission Policy Model for Dynamic Schedulers
Two basic metrics to decide the sending order are the deadline D and the laxity L of the packets. It is important to select and schedule packet delivery of scalable streaming media over a lossy network. We assume that different layers in a frame have the same deadline. Each task (packet) T is characterized by the following parameters :

$T_x(y)$: the task (packet) of the yth layer in frame x. The tasks are put into the transmission buffers according to the decoding order.

$E_x(y)$: the earliest time at which the task $T_x(y)$ becomes ready for scheduling in transmission buffer.

$P_x(y)$: the processing time of a task $T_x(y)$.

$L_x(y)$: the laxity value of the task $T_x(y)$. $L_x(y) = D(x) - P_x(y)$.

$LT_x(y)$: the laxity value of the task $T_x(y)$ at a given time.

$D(x)$: the latest time at which all packets of frame x should be sent to the client, otherwise it is too late for playback.

$S_x(y)$: the scheduling time .

$B_x(y)$: the size of the packet $T_x(y)$.

B_w : the current channel bandwidth.

$R_x(y)$: the transmission time of the task $T_x(y)$.
$$R_x(y) = B_x(y) / B_w.$$

The processing time of a transmit pack is in Figure 3. If the following conditions are satisfied, the task $T_x(y)$ is ready for scheduling:

1. The current time t_{cur} is later than its earliest time $E_x(y)$, i.e. $E_x(y) <= t_{cur}$
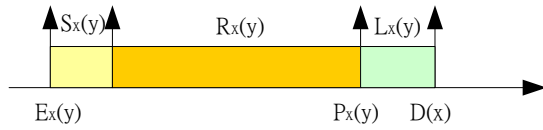2. The processing time $P_x(y)$ is earlier than its deadline, i.e. $t_{cur} + R_x(y) <= D(x)$

Figure 3: The processing time of a transmit pack

3.3 Layer-Based LLF Scheduling Algorithm

It is selected and scheduled packet by Least Laxity First scheduling algorithm delivery of scalable streaming media over a lossy network. We proposed the adaptive Layer-Based Least-Laxity-First (LB-LLF) scheduling algorithm, which combines importance and priority of the layer. As packets at different layers have different effects on the playback quality, we set the higher priority to the lower (more important) layer packets and set the lower priority to the higher (less important) layer packets. Thus, important layer packets should be transmitted earlier, with more chances to be retransmitted in case of loss. Packets in the same layer are served according to LLF scheduler. The detailed description of the algorithm is given as follows.

Layer-based LLF real-time scheduling algorithm :

Step 1: let R = frame set of ready packets with the lowest layer in the server transmission buffers.

Step 2: compare the current time t_{cur} with the deadline $D(x)$ of all the packets in R.
If $t_{cur} > D(x)$, remove the packet from R.

Step 3: calculate the least-laxity value $L_x(y)$ for each set from R = 1 to N (where N is the buffer size in server.)

Step 4: select the smallest least-laxity value $LT_x(y)$ packets from R.

Step 5: Send out the set of packets out, and let $t_{cur} = t_{cur} + P_x(y)$.   Go to Step 1.

The adaptive LB-LLF scheduling algorithm has two key points. First, it selects the packets from the lowest layer in ready state, the important packet to be sent by the server will transmit much earlier than its playback time, and this important packet will have more chances to be transmitted to the client buffer for displaying. Second, it calculates the least-laxity value for each set from R = 1 to N and services the packet of the smallest laxity set. In real-time constraint, unequal priorities of scalable media stream in different layers and achieve a good trade-off between coding efficiency and drifting error are considered in the proposed algorithm. This guarantees the better usage of available channel bandwidth and the smoother playback in client.

4. EXPERIMENTAL RESULTS

We use the Microsoft H.26L-PFGS encoder/decoder to generate the simulation data. It is an efficient scalable coding scheme with fine granularity scalability, where the base layer is encoded with H.26L, and the enhancement layer is encoded with PFGS coding. For comparison performance of the scheduling algorithms in the given channel bandwidth, we set the base layer as the highest priority that can get all the base layer data without any packet losses. In the experiment, the sequence Foreman in QCIF format is used. It is encoded with 25 frames per second and 200 frames are encoded and transmitted by the proposed scheduler. Because the maximum level of bit-plane is 6 in the Foreman sequence, there are 6 layers in the Enhancement layer. Different enhancement layer bit-plane has different frame size. The sizes of the enhancement layers in the sequence are shown in Figure 4. The enhancement layer 1 (Enh1) is the smallest in size, but it is the most significant layer. The enhancement layer 6 (Enh6) is the largest in size, but it is the least significant layer. The average rate of video data with all enhancement layers is 3.83 Mbps. The transmission buffer size is set to 25. The playback frame rate is 25 frames per second. The scheduling time is 0.2ms. The channel bandwidth is various from 0.5 to 3.5 Mbps.
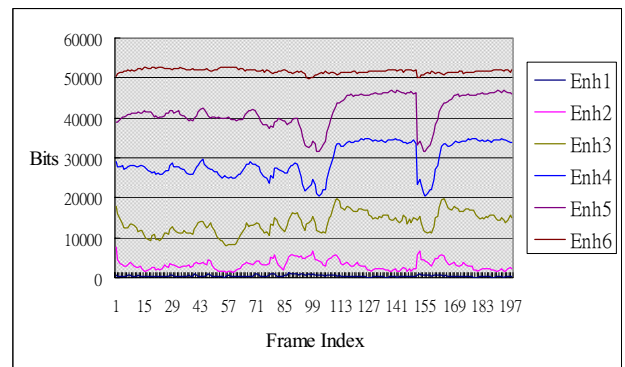

Figure 4: The bit size of the enhancement layers

YASA [10] is used as an experimental tool for scheduling ability analysis. Figure 5 shows comparisons among Frame-Based EDF, Layer-Based EDF, and Layer-Based LLF scheduling algorithms with various bandwidths from 0.5 to 3.5 Mbps. With the Layer-Based LLF scheduling algorithm, the experimental results of the channel utilization is 99.01%. It improves about 26.4% over a Frame-Based EDF and 0.24% or more over a Layer-Based EDF delivery algorithm. In Figure 6, we compare Layer-Based EDF and Layer-Based LLF scheduling algorithms with various buffer sizes from 5 to 30. It shows that the larger buffer size can get the better channel utilization, and our proposed algorithm (LB-LLF) is better utilized than the LB-EDF with slight margin. As mentioned above this improvement guarantees the effective usage of available channel bandwidth and the better quality of playback in client.
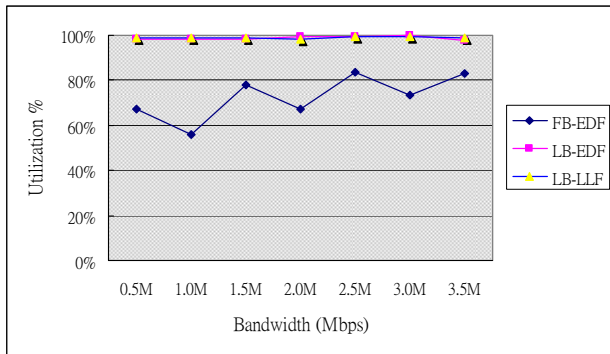

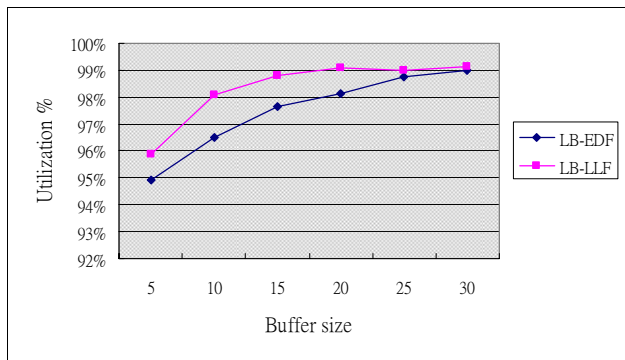Figure 5: The channel utilization with various bandwidths

Figure 6: The channel utilization with various buffer sizes

5. CONCLUSION

In this paper, we proposed and implemented a simple and effective real-time scheduling algorithm, adaptive Layer-Based Least-Laxity-First (LB-LLF) scheduling algorithm, to improve the output quality of video/audio on network to achieve synchronism effect. It is improving the playback quality for the packets of scalable streaming media by real-time scheduling algorithm in the server transmission buffers.

REFERENCES

[1] Feng Wu, Shipeng Li, and Ya-Qin Zhang, "A Framework for Efficient Progressive Fine Granularity Scalable Video Coding", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 3, March 2001, pp. 332-344.

[2] Qi Wang, Feng Wu, Shipeng Li, Yuzhuo Zhong, Ya-Qin Zhang, "Fine-granularity spatially scalable video coding", Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001, (ICASSP '01), vol.3, pp.1801–1804.

[3] Feng Wu, Shipeng Li AND Ya-Qin Zhang, "Progressive Fine Granular Scalable (PFGS) video using Advance-Predicted bitplane coding (APBIC)", The 2001 IEEE International Symposium on Circuits and Systems, 2001(ISCAS 2001), vol. 5, pp.97-100.

[4] Kui Gao, Wen Gao, Simin He, Peng Gao, Yuan Zhang, "Real-Time Scheduling on scalable media stream delivery", Proceedings of the 2003 International Symposium on Circuits and Systems, 2003(ISCAS '03), vol.2, pp. II-824 - II-827.

[5] Kui Gao, Yuan Zhang , Wen Gao, Simin He, "Real-Time Scheduling Supporting VCR Functionality For Scalable Video Streaming", 14th IEEE Proceedings on Personal, Indoor and Mobile Radio Communications, 2003(PIMRC 2003), vol.3, pp.2711–2715.

[6] Frank Golatowski, Jens Hildebrandt, Jan Blumenthal, Dirk Timmermann, "Framework for Validation, Test and Analysis of Real-time Scheduling Algorithms and Scheduler Implementations", Proceedings of the 13th IEEE International Workshop on Rapid System Prototyping, 2002, pp.146-152.

[7] Frank Golatowski, Jens Hildebrandt, Dirk Timmermann, "Rapid Prototyping with Reconfigurable Hardware for Embedded Hard Real-Time Systems", 19th IEEE Real-Time Systems Symposium, WIP-Proc., Madrid, Spain ,1998.

[8] Jens Hildebrandt, Frank Golatowski, Dirk Timmermann, "Scheduling Coproessor for Enhanced Least-Laxity-First Scheduling in Hard Real-Time Systems", Proceedings of the11th Euromicro Conference on Real-Time Systems, June 1999, pp.208-215.

[9] Sung-Heun Oh, Seung-Min Yang, "A Modified Least-Laxity-First Scheduling Algorithm for Real-Time Tasks", Proceedings of the 5th International Conference on Real-Time Computing Systems and Applications, Hiroshima, Japan, 1998, pp. 31-36.

[10] YASA – Yet Another Schedulability Analyzer, University of Rostock, 1998, http://www-md.e-technik.unirostock.de/ma/gol/yasa/

Design and Implementation of a Bluetooth Ad-Hoc Presentation System

Shih-Yao Lee, Kuei-Ping Shih, Hung-Chang Chen, and Chien-Min Chou

Department of Computer Science and Information Engineering
Tamkang University
Tamshui 251, Taipei, Taiwan
Email: kpshih@mail.tku.edu.tw

Abstract—**The paper presents the development of the Bluetooth ad-hoc presentation system. The presenter can make a good presentation by using our system and constrains of location will not be worried. Therefore, the system can be used in school instructions, conference presentations, research publications, and any other scenarios where a presentation system is needed. The system holds true market values due to the cooperation of both the business and the academic fields. It is expected to create a new generation of mobile presentation system.**

Keywords- Bluetooth, presentation, wireless networks, ad-hoc networks.

I. INTRODUCTION

Bluetooth is a standard for a small, cheap radio chip to be plugged into computers, printers, mobile phones and so on. A Bluetooth chip is designed to replace cables by transmitting the information at a special frequency to the receiver. Based on this character, our research plan utilizes Bluetooth and ad-hoc networking as the core technology along with Remote Control technology to design and develop a mobile presentation system.

Generally speaking, PowerPoint is used to perform a presentation in the following scenarios:

1. Schools teaching environment.
2. Business presentation.
3. Conference proceeding.

Figure 1 illustrates the Bluetooth ad-hoc presentation system. The laptop and the industrial PC are connected by Bluetooth dongles. The presenter's laptop works as the client while the industrial PC works as the server respectively. Firstly, the server runs at the standby stage and waits for the PowerPoint file sent by the client. A file is then transmitted from the client to the server. Finally, when the client sends remote command to the server, the server will execute pre-defined functions such as "open file",

"previous page", "next page" and "exit". These functions could help the presenter do presentation from a remote terminal. All the functions are achieved by way of Bluetooth technology.

Figure 1   System Diagram

The rest of the paper is organized as follows. Background information is introduced in Section II. Section III describes the structure and principle of our system. The experiment results are presented in Section IV. Finally, Section 0 concludes the contribution of our presentation system and our future works.

II. BACKGROUND

Bluetooth is a short range, low cost and low power wireless access technology used to replace the cable(s). The 2.4 GHz unlicensed *Industrial-Scientific-Medical*(ISM) band is split into 79 channels spaced 1 MHz apart to employ a *frequency hopping spread spectrum* (FHSS) technique. The Bluetooth system provides the packet delivery based on slotted *Time Division Duplex* (TDD) scheme where each slot is 625 μs [4].

A Bluetooth profile specification lists the requirements for both Bluetooth protocol software and application software. All Bluetooth devices must support one or more Bluetooth profiles in order to interoperate with each other. The current Bluetooth specification has thirteen profiles. Among them, Generic Access Profile, Service Discovery Profile, Dial-up Networking Profile, Generic Object Exchange Profile, Object Push Profile and File Transfer Profile are supported in our system. In other words, our presentation system fully conforms to the Bluetooth

specifications and can interoperate with other Bluetooth devices.

III. SYSTEM DISCUSSION

We use the solution which is on the top of the ESI's and WildComm's Bluetooth suites to build our remote presentation system's prototype [1][2][3].

The system is designed on the top of the Bluetooth Protocol and the Bluetooth Object Push Profile is in the beginning. Unfortunately, delay problem comes out because of frequent connection establishments. Each time when sending remote commands from the local side to the remote side, a temporary ACL connection should be established. Therefore, the following two methods have been taken in order to improve the performance of our system:

1. Using "Create a folder and refresh" method to simulate connectionless environment. However, it has a connection refreshment delay problem.
2. Design another application program running on the top of PPP and the Bluetooth Protocol. This solution does work and the details will be described latter.

Two remote control presentation programs were designed. They are running on the top of the Bluetooth Protocol and TCP/IP over the Bluetooth Protocol respectively. The two systems are going to be discussed in the following subsections.

A. Solution 1

1) Networked structure - Figure 2 shows the network structure of the system.

Applications		Applications	
OBEX		OBEX	
Generic Object Exchange Profile		Generic Object Exchange Profile	
Serial Port Profile		Serial Port Profile	
Generic Access Profile		Generic Access Profile	
SDP	RFCOMM	RFCOMM	SDP
L2CAP		L2CAP	
	LMP	LMP	
Baseband		Baseband	

Figure 2   Bluetooth Structure (Solution 1)

2) System's working theories – Figure 3 shows the flow chart of our system.

3) System's working theories – Pseudo Code

Figure 4 shows the pseudo code of the solution 1. The first two lines of the code are used to initialize the Object Push Profile. Among them, the second line sets the directory path to save receiving specific file. When other Bluetooth devices use the Object Push Profile to send a file to the local device, the op.ObjReceiveRequest event will be triggered and then the function ongetfiles will be executed.

For function "GetObject()", when the return value is TRUE means that the server has received the file. On the contrary, the FALSE value means that the server has rejected to receive the file.

Figure 3   Program Flow Chart (Solution 1)

```
op.Initialize(False)
        op.Iop.BTInboxPath = "c:\bt"
    AddHandler op.ObjReceiveRequest, AddressOf ongetfiles
    Sub ongetfiles(ByVal objname As String)
    Dim plg As New Form7()
        Dim dlg As New Form9()
        dlg.Label1.Text = "file is sending" + objname
    If dlg.ShowDialog = DialogResult.OK Then
        Console.WriteLine("!!!!!!!!!!!!!" + objname)
        op.GetObject(True)
        plg.Show()
    Else
        op.GetObject(False)
    End If
        End Sub
```

Figure 4   Pseudo Code (Solution 1)

B. Solution 2

1) Networked structure - The following diagram shows the network structure of the system.

Figure 5   TCP/IP Over Bluetooth Structure (Solution 2)

2) System's working theories – Flow Chart

The flow chart of solution 2 is similar to Figure 3 . The difference is on the connection method. In solution 2, TCP/IP protocol is used to support object exchange. The object here means the PowerPoint files and the corresponding commands. At the end of file receiving, the server waits for the client to send a predefined function call to execute specific functionalities such as "Next – next page", "Previous - previous page" and "Exit – exit presentation".

3) System's working theories – Pseudo Code

```
Function NextPPT()
    ppt.SlideShowWindows(Index:=1).View.Next
    If(ppt.SlideShowWindows(Index:=1).View.State=
        ppSlideShowDone) Then
            wsTCP(1).SendData "over" & vbCrLf
    End If
End Function
```

Figure 6   Pseudo Code (Solution 2)

In the pseudo code, a function defined by us is called the "NextPPT" function. The sever program can execute the function by using the method "SlideShowWindows(Index:=1).View.Next" when receiving a remote command from the client.

IV.   PERFORMANCE VERIFICATION

The paper aims at providing a presentation system. Therefore, in this section, Bluetooth connection's performance will be analyzed in the following two scenarios. We use numbers of SDKs during the experiment process:
1.   Piconet connection: point to point and point to multipoint master and slave connection scenarios.
2.   Scatternet connection: the master of a piconet also works as a slave of other Piconets.

The followings are the experiment results.

A.   Point to Point Piconet Connection

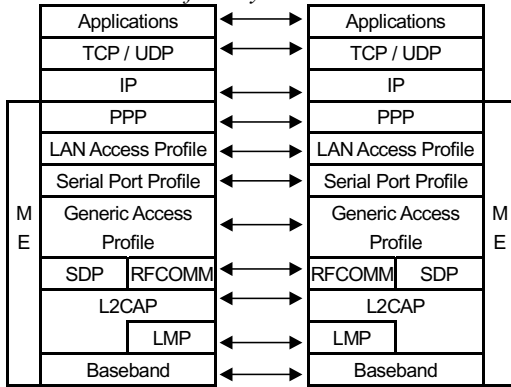The transferring speeds were tested non-simultaneously here. The Affix was tested by the DH1, DH3, and DH5 packet types manually. On the contrary, other SDKs were tested under a dynamic assigning packet type algorithm.

Figure 7   POINT TO POINT PICONET Connection

TABLE I.   Master-Slave Point to Point Connection Bandwidth Result

Scenarios / SDK	Master pushes Slave Baud Rate (Kbps)	Slave pushes Master Baud Rate (Kbps)
ESI v1.3.4	179 ~ 182	182 ~ 192
ESI v1.3.6	314 ~ 373	322 ~ 378
CC&C v1.1.2.8	319 ~ 387	318 ~ 383
CC&C v1.2.2.9	430 ~ 461	418 ~ 468
Affix v1.2.1 DH1	86.4 ~ 89.3	86.9 ~ 90.6
Affix v1.2.1 DH3	140.4 ~ 143	140.8 ~ 144.5
Affix v1.2.1 DH5	114.4 ~ 122.4	115.2 ~ 121.8

B.   Point to Point Piconet Bi-direction Connection

The bi-directional transferring speeds were tested simultaneously. However, the connection was not established at the same time.

Figure 8   Point to Point Piconet Bi-direction Connection

TABLE II.   Master-Slave Point to Point Bi-direction Connection Bandwidth Result

Scenarios / SDK	Master A pushes Slave A and Master B pushes Slave B simultaneously-Baud Rate (Kbps)
ESI v1.3.4	N/A
ESI v1.3.6	206 ~ 249
CC&C v1.1.2.8	N/A
CC&C v1.2.2.9	282 ~ 311

C.   Point to Multipoint Piconet Connection

The tri-directional transferring speeds were also tested simultaneously. Figure 9 shows the simulation environment.
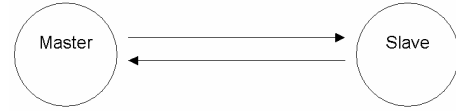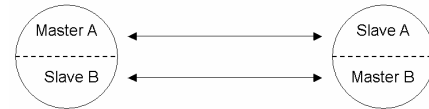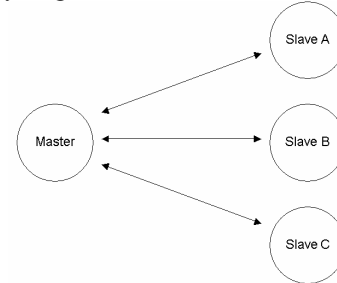

Figure 9   Point to Multipoint Piconet Connection

TABLE III.   Master-Slave Point to Multipoint Connection Bandwidth Result

Scenarios SDK	Master pushes Slave A Baud Rate (Kbps)	Master pushes Slave B Baud Rate (Kbps)	Master pushes Slave C Baud Rate (Kbps)
ESI v1.3.4	N/A		
ESI v1.3.6	155 ~ 182		
CC&C v1.1.2.8	N/A		
CC&C v1.2.2.9	201 ~ 227		

D. Scatternet Connection

Scatternet is also tested. Figure 10   illustrates the simulation environment.
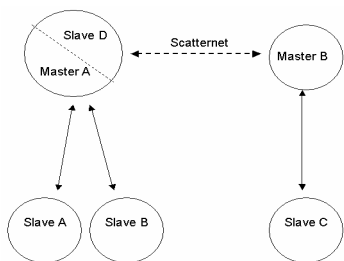

Figure 10 Scatternet Connection

The Scatternet connection shown in Figure 10   has absolute high priority to occupy 83% of the total bandwidth. The piconet of master A has a similar performance like the scenario in subsection B after this transmission is finished.

1) Phase 1

TABLE IV.   Scatterent Connection Bandwidth Result (Phase 1)

Scenarios SDK	Master A pushes Slave A & B Baud Rate (Kbps)	Master B pushes Slave C Baud Rate (Kbps)	Master B pushes Slave D Baud Rate (Kbps)
ESI v1.3.4	N/A		
ESI v1.3.6	8.2 ~ 15.8	232 ~ 267	
CC&C v1.1.2.8	N/A		
CC&C v1.2.2.9	11 ~ 19.2	256 ~ 290	

2) Phase 2

TABLE V.   Scatterent Connection Bandwidth Result (Phase 2)

Scenarios SDK	Master A pushes Slave A & B Baud Rate (Kbps)	Master B pushes Slave C Baud Rate (Kbps)	Master B pushes Slave D Baud Rate (Kbps)
ESI v1.3.4	N/A		
ESI v1.3.6	211 ~ 253	Transmission Complete	
CC&C v1.1.2.8	N/A		
CC&C v1.2.2.9	289 ~ 325	Transmission Complete	

From the above simulations, SDK CC&C works better than other SDKs in all scenarios. Therefore, CC&C is a suitable SDK for the implement of our presentation system.

V.   CONCLUSIONS

Our Bluetooth ad-hoc presentation system uses Bluetooth technology to connect the client and the server. The invention of our system takes advantage of the Bluetooth technology to make the presentation more convenient than before. Our system could develop learning interest of presenters. On the other hand, it could encourage the presenters to speak out during presentations, because the presenters do not stand in front of all audiences but stay behind at his seat. Based on this point, it could also improve presenters' learning conveniences.

After having a look at the difference between present presentation scenarios and our Bluetooth presentation system, we might find that there is no much distinction when it is compared to the 802.11 technologies at the moment. It is because that Bluetooth technology is still expensive today. Bluetooth technology has not been widely deployed today, and therefore we do not have a lot of products to support the Bluetooth technology. However, Bluetooth module will soon be embedded in most of the mobile computing devices like PDAs, Mobile phones and laptops. For this reason, our proposed presentation system will become more distinct in the near future.

We did a trial on the Bluetooth connectivity performance and derive a result from the formation of piconet and sactternet, even the scatternet could be only formed and tested in one way today [4][5]. We lead the way of using up-to-date wireless technology to support educational activity, and this will be widely applied in the near future.

Although our system is working well under point-to-point basis, the presenter has to disconnect the existing connection for the next presenter's connection. In another word, it does not support multi-client scenario as well as the switching function between clients to handover control rights. This point could be a technical development issue in the future. On the other hand, we could embed the whole system into a small box with Plug-and-Play nature to create business fortune and patent from the business point of view in the future.

REFERENCES

[1] ESI XTNDConnect Blue SDK Guide, v1.3.4, 2001
[2] ESI XTNDConnect Blue SDK Guide, v1.3.6, 2002
[3] WildComm BlueTake Guide, v1.2.2.9, 2001
[4] Specification of the Bluetooth System, volume I, v1.1, Feb 2001
[5] Specification of the Bluetooth System, volume II, v1.1, Feb,2001

Eliminating Conflicts of a Coordination System over Construction of Plans

Anthony Y. Chang
Department of Information Technology
the Overseas Chinese Institute of Technology
Taichung, Taiwan, R.O.C.
email: achang@ocit.edu.tw

Abstract

In this paper we develop a general, theoretical computational model for discussing synchronization and specification scheme. We analyze the domains of relationships between agents in a coordination system. A set of algorithms is proposed to derive reasonable relations between agents. Possible conflicts in the agent specification are firstly detected and eliminated. Our mechanism then constructs partial order relations among actions. We also construct a temporal algebra to deal with qualitative and quantitative temporal relationships and to reason about definite and indefinite time. The algebra can also integrate instant points and even intervals. Several computation tables are proposed, and each table include a set of complete logics. The mechanism is efficiently to eliminate conflict specification and to generate synchronization scenarios. The contributions of this paper are using the generic relationship framework to handle synchronization specifications, and to eliminate conflicts in order to satisfy agent request.

Keywords: Agent Synchronization, Multi-Agent, Scheduling, Coordination Systems, Interval Algebra

1. Introduction

When several agents work together, it is necessary to communicate between the master agent and the other worker agent. To synchronize the various types of agents is the major challenge for a coordination system. The coordination of actions is the set of supplementary activities which need to be carried out in a multi-agent environment.[1] When accessing common resources, in order to guarantee the system remains in a coherent state, actions have to be synchronized by computing procedures[2] Biniaris present key issues related to the distributed implementation of a Finite-Difference Time Domain (FDTD) code using java mobile agents, and special agent communication and synchronization aspects related to FDTD are presented in [3] . Conservative and optimistic approaches for resolving distributed simulation of multi-agent hybrid systems is presented in [4] Karim Hussein[7] provided a collaborative agent interaction and synchronization system for insuring effective structuring and control of a distributed conference. Shivakant Mishra and Peng Xie[5] design an interagent communication and synchronization model in the DaAgent mobile agent-based computing system.

There are some temporal frameworks to specify temporal data model of synchronization. Little and Ghafoor[6] proposed the object composition Petri Net(OCPN). However it does not easily capture the distributed nature of application or react to the real world, just indicated in [13] The time-flow graph (TFG) is proposed in [14] but the graphs fail to represent interactive semantics and to reduce ambiguous situations. Time-line diagram is an general and intuitive model, such as QuickTime[15] and MAEStro[16] . In a time-line, it could compute the absolute and relative time precisely, but it fails to specify uncertain and indefinite temporal information. A. Safavi and S. F. Smith[10] proposed a search-based approaches to measure the global impact of a commitment on the entire schedule.

Based on our early research result about synchronization mechanisms [2] we analysis the relationship between agents and define a temporal algebra system for unifying and scheduling. A set of algorithms is proposed to derive consistent relations between agents. The mechanism is efficiently to eliminate conflict specification and to generate

synchronization constraints.

2. Conflicts of a Coordination System

A coordination system usually contains several agents must be coordinated and a number of resources must be shared and synchronized with actions. These resources need to be arranged as layout.

Modeling an agent temporal scenario often requires synchronizing the distributed resources. In following sections, we propose an integrated temporal computational model to deal with the following inconsistencies.

- **Qualitative inconsistency**: the conflicts occur in the semantics or logics of temporal relations. Such as a scenario plays dependently with other objects which can not be satisfied with temporal relations. Moreover events with unpredictable times of occurrence and durations may violate some specification constraints. For example, media A, B plays before media C in a <seq> container followed a specification <C begin= 'A.begin' …/>, there exist inconsistency between temporal specifications.

- **Quantitative inconsistency**: the conflicts occur in the scheduling on syncbase-value, event-value, or offset-value.

- **Resources limitation**: resources are needed to accomplish agent actions, they are limited and vehicles are obliged to coordinate their action to avoid each other. Resources have to be shared with eliminating pointless actions, reducing costs and avoiding any possible conflicts.

- **Mutual Constraints**: the conflicts occur in the interdependencies between actions of agents.

3. Multi-Agent Synchronization

J. Feber[2] indicated that rapidity, adaptability and predictiveness are the temporal characteristics of the coordination system. Also, to synchronize several actions have to define the manner in which actions are time-related. Coordination is a matter of positioning actions in time and space. As soon as several agents have to move together, their movements have to be synchronized in time and location. Since the agent synchronization base on time with a dynamic variation, the inconsistency often occurs in both qualitative semantics and quantitative values. We analyze conflicts between actions of a set of agents, and propose a methodology based on temporal algebra to deal with qualitative and quantitative inconsistency. It not only eliminates conflicts relations but also reasons about indefinite, uncertain, and incomplete temporal relationships.

4. Relational Domain

This Section describes the symbolic constraint propagation. In a coordination system, it is necessary to manage the inter-agent communications between the actions of a set of agents. Three agents have to take mutual dependencies and constraints. The general idea is to use the existing information about the relations among time intervals or instants to derive the composition relations. For example, there exists three agents X, Y, and Z, have to coordinate their action, it means that not only must X coordinate with Y, but also coordinate with Z. With specifying interdependencies, if "X before Y" and "Y before Z", it plains action of agent X has to be before Z.

The composition may result in a *multiple derivation*. For example, if "X before Y " and "Y during Z ", the composed relation for X and Z could be "before", "overlaps", "meets", "during", or "starts". If the composed relation could be any one of some relations, these derived relations are called *reasonable relations* in our discussion. A reasonable set is a set of reasonable relations according to our definition. A reasonable set can not be empty, since there must exist at least one relation between any two events, assuming that they are in the same one dimension (i.e. the time line).

In some cases, relation compositions may result in a *conflict specification* due to the user specification or involved events synchronously. For example, if specifications "X before Y", "Y before Z", and "X after Z" are declared by the user, there exists a conflict between X and Z. When the specific relations are not found in derived reasonable set, the specification may cause conflicts.

We analyze the domain of temporal relations and use a directed graph to compute the relations of agents. In the computation, we consider all possibilities: the unknown derivations, the multiple derivations, and the conflict derivations.

Definition : An *user edge* denotes a relation between a pair of objects defined by the user. The relation may be reasonable or non-reasonable.

Definition : A *derived edge* holds a non-empty set of reasonable relations derived by our algorithm. The relation of the two objects connected by the derived edge can be any reasonable relation in the set.

For each pair of objects in the time line, there exists a set of possible binary relations held between the pair of objects. For an arbitrary number of objects (denoted by nodes), some of the relations (denoted by

edges) are specified by the user while others are derived. If there exists a cycle in the directed relation graph, a conflict derivation may occur. If there exists no cycle, there is no conflict. Based on the above considerations, we suggest that the computation domain reveals four types, as discussed below.

The *complete relation domain* is a complete graph which contains possible conflicts. We want to find a *reasonable relation domain* containing no conflict derivation. Note that, in these two domains (i.e., the complete and the reasonable), both user edges and derived edges exist. If there is a conflict among a set of user edges, one of the user edge must be removed from the cycle, or the relation of that user edge must be re-assigned. If there is no conflict, the two domains are equal.

The *reduced relation domain* contains relations specified by the specification only. It is possible that the user issues a conflict situation. To avoid the occurrence of conflicts, we place a restriction on the user's interaction. Instead of allowing the user to add an arbitrary relation to the relation graph, we only allow the user to add objects to a *restricted relation domain*, which is a tree and a sub-domain of the reduced relation domain. That is, when the user is about to add a new edge, the user either adds a new node connected to an existing node via an user edge, or joins two sub-trees via the user edge. No cycle is created in the restricted relation domain. Thus, the conflict situation does not exist. When deleting an user edge, the user has to maintain the connectivity of the tree. If all nodes are connected, the user specification is complete. Otherwise, the coordination system should alert the user to complete the specification. The above domains can be summarized as the following:

- The **complete relation domain** (a complete graph) contains user edges and derived edges, with possible cycles and possible conflicts.

- The **reasonable relation domain** (a graph) contains user edges and derived edges, with possible cycles but no conflict.

- The **reduced relation domain** (a graph) contains only user edges, with possible cycles and possible conflicts.

- The **restricted relation domain** (a tree) contains only user edges, without cycle.

The four domains are used in the analysis and computation of object relations. In Section 5 and, we propose two algorithms computing the reasonable

relation domain.

Figure 1: Relational Domain

5. Maintaining Reasonable Relationship

The purpose of the first algorithm is to add derived edges to the reduced relation domain. If there is a conflict cycle in the original reduced relation domain, the algorithm eliminates that conflict first by altering the user to select a reasonable relation to replace the original one (i.e., *UE*, also see algorithm *EliminateConflicts*). This is why the resulting graph may contain some new user edges (i.e. *UE'*). Thus, the resulting reasonable relation domain is a complete graph, which is equal to the complete relation domain. This conflict elimination is achieved by invoking the *EliminateConflicts* algorithm. Suppose G is a graph of the reduce relation domain, and GV and GE are the vertex set and edge set of G. Initially the reasonable relation domain is set to the reduced relation domain. The algorithm computes derived edges based on user edges. The reason of using the user edges is that these edges contain the minimal and sufficient information of what the user wants. If the algorithm computes derived edges from other derived edges, eventually, the algorithm has to compute the set intersection of all possible derivations for the reasonable set of the new derived edge.

Algorithm : *ComputeRD1*
Input : $G = (GV, GE)$
Output: $K_n = (K_nV, K_nE)$
Preconditions : true
Postconditions : $GV = K_nV \wedge GE \setminus UE \cup UE' \subseteq K_nE$
Steps :
1 : $G = EliminateConflicts (G)$
2 : $K_n = G \wedge pl = 2$
3 : repeat until $| K_nE | = | K_nV | * (| K_nV | -1) / 2$
  3.1 : for each $e = (a, b) \wedge e \notin K_nE \wedge a \in K_nV \wedge b \in K_nV \bullet$
    there is a path of user edges from a to b , with path length $= pl$
  3.2 : suppose $((n_1, n_2), (n_2, n_3),\ldots, (n_{k-1}, n_k))$
  is a path with $a = n \wedge b = n \wedge k = pl + 1$

3.3 : set $e.rs = Table29$ ((a, n_{k-1}).rs, (n_{k-1}, b).rs)
     3.4 : $K_nE = K_nE \cup \{ e \}$
     3.5 : $pl = pl + 1$

The first algorithm, *ComputeRD1*, starts from taking each path of user edges of length 2, and computes a derived edge from that path. The insertion of edge $e = (a, b)$ results a cycle, but no conflict. The reasonable set of edge e (i.e., *e.rs*) is computed from two edges, (a, n_{k-1}) and (n_{k-1}, b), which are user edges or derived edges. Since we increase the path length, *pl*, of the path of user edges one by one. The derived edge (a, n_{k-1}) (or user edge, if $pl = 2$) must have been computed in a previous interaction. The algorithm repeats until all edges are added to the complete graph K_n, which contains $n* (n$-1$)/2$ edges.

Algorithm : *EliminateConflicts*

Input : $G = (GV, GE)$

Output : $G' = (G'V, G'E)$

Preconditions : G contains only user edges $\wedge$ $G' = G$

Postconditions : $G' = G$

Steps :

1. for each $P = ((n_1, n_2), (n_2, n_3),\ldots, (n_{k-1}, n_k))$ in G'
with $n_1 = n_k$ $\wedge$ $k > 3$

   1.1 : for each i, $1 \leq i \leq k$-2

        1.1.1 : set $(n_i, n_{i+2}).rs = Table29$ ((n_i, n_{i+1}).rs, (n_{i+1}, n_{i+2}).rs)

   1.2 : $rs = Table29$ ((n_k, n_{k-2}).rs, (n_{k-2}, n_{k-1}).rs)

   1.3 : if $(n_k, n_{k-1}).r \notin rs$ then

        1.3.1 : ask user to choose a $r' \in rs$

        1.3.2 : set $(n_k, n_{k-1}).r = r'$

In *ComputeRD1*, we use the conflict elimination algorithm. A conflict occurs only if there is a cycle. For each cycle in the reduced relation domain, the *EliminateConflicts* algorithm finds a derived edge between any two consecutive edges, namely, (n_i , n_{i+1}) and (n_{i+1} , n_{i+2}). The algorithm then checks if the last user edge making the cycle represents a relation (i.e., $(n_k , n_{k-1}).r$) belongs to the reasonable set computed for the user edge (i.e., *rs*). If not so, the algorithm asks the user to choose an arbitrary relation r' belongs to the reasonable set and use the relation to replace the original one.

In the actual implementation of an application, depending on the user's specification, directions of user edges are easily decided and represented in the implementation. Based on the composition of object relations has a direction, as illustrated in Figure 5.2. To compose r_1 and r_2 , if the direction of one of the edge is in an opposite direction, we must firstly find the inverse relation before the composition proceeds.
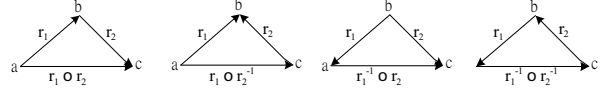

Figure 2: Object relation composition with directions

6. Reasoning Mechanisms

We also propose a generic temporal constraint model to compose consistent scenarios without conflicts, to integrate event with point and interval temporal specification. This model can deal with both accurate and indefinite scenarios.

From the point view of any single process, events are ordered uniquely by times that shown on the local clock. Qualitative calculus is calculus of intervals instead of real numbers. To deal with qualitative representation, the real number in timeline is subdivided into three intervals: $[-\infty, 0]$, $[0, 0]$ and $[0, +\infty]$, and are denoted as $\{<\}$, $\{=\}$ and $\{>\}$ (*before, simultaneously*, and *after*) for representing relations between two points. The notation R1 and R2 denotes the point relations over three points A, B, and C which A R1 B and B R2 C.

Uncertain or imprecise knowledge is expressed in terms of qualitative variables and qualitative functions. The value of a qualitative variable can represent an element as well as a set of elements. On the other hand, the value of a quantitative variable specifies only one element in the set of possible values. If the set is the entire real number line, a value represents a specific point. This Section shows quantitative-qualitative mechanisms to reason about qualitative and quantitative temporal knowledge. The knowledge is not only maintaining quantitative-qualitative relations but also checking consistency and eliminating conflict relations. An powerful temporal algebra system translates point and interval relations and integrates quantitative and qualitative relations.

In order to express more precise relations without losing qualitative information, the temporal relations extended with qualitative mechanisms for handling quantitative information. To give a concrete form to the topic of temporal representation, consider the following variable and equations with quantitative and qualitative information.

Definition 6.1: A *quantitative-qualitative variable* is defined by a 2-tuple

$$Q = (Q_R, V)$$

where Q_R, is a qualitative relation, and V is a quantity which indicates degrees of Q_R. ■

Definition 6.2: Formal Endpoint Relations

A formal endpoint relation $Q_E = (E_R, V_E)$ is a quantitative-qualitative valuable.

where E_R, is an endpoint relation based on the point space $\{<, =, >\}$, and V_E is a quantity which expresses a quantitative value associated with E_R between two endpoints. ■

Definition 6.3: quantitative-qualitative functions

Addition, subtraction and unary minus are functions that take two *quantitative-qualitative variables* and return a third:

$+ : Q \times Q \to Q$
$- : Q \times Q \to Q$
$[\cdot] : Q \times Q \to Q$ ■

The meaning of some quantitative-qualitative calculus operators and equality are defined as follows.

Table 1: quantitative-qualitative addition: $+$

(R2,v2) / (R1,v1)	$<_{v2}$	$=$	$>_{v2}$
$<_{v1}$	$<_{v1+v2}$	$<_{v1}$	$<_{v1-v2}$, if (v1>v2) =, if (v1=v2) $>_{v2-v1}$, if (v1<v2)
$=$	$<_{v2}$	$=$	$>_{v2}$
$>_{v1}$	$>_{v1-v2}$, if (v1>v2) =, if(v1=v2) $<_{v2-v1}$, if(v1<v2)	$>_{v1}$	$>_{v1+v2}$

Table 2: quantitative-qualitative subtraction: $-$

(R2,v2) / (R1,v1)	$<_{v2}$	$=$	$>_{v2}$
$<_{v1}$	$<_{v1-v2}$, if (v1>v2) =, if (v1=v2) $>_{v2-v1}$, if (v1<v2)	$<_{v2}$	$<_{v1+v2}$
$=$	$<_{v2}$	$=$	$>_{v2}$
$>_{v1}$	$>_{v1+v2}$	$>_{v1}$	$>_{v1-v2}$, if (v1>v2) =, if(v1=v2) $<_{v2-v1}$, if(v1<v2)

Table 3: quantitative-qualitative unary minus: [-]

(R1, v1)	[-](R1, v1)
$<_{v1}$	$>_{v1}$
$=$	$=$
$>_{v1}$	$<_{v1}$

In addition, a quantitative-qualitative equation correctly expresses both qualitative equation and quantitative equations by formal endpoint variables and operators. We give a set of equations: for an example

- Given $[d_A]$, $[d_B]$ and $[AsBs]$
  $[AsBe] = [AsBs] + [d_B]$
  $[AeBs] = -[d_A] + [AsBs]$
  $[AeBe] = -[d_A] + [AsBs] + [d_B]$
- Given $[d_A]$, $[d_B]$ and $[AsBe]$
  $[AsBs] = [AsBe] - [d_B]$
  $[AeBs] = -[d_A] + [AsBe] - [d_B]$
  $[AeBe] = -[d_A] + [AsBe]$
- Given $[d_A]$, $[d_B]$ and $[AeBs]$
  $[AsBs] = [d_A] + [AeBs]$
  $[AsBe] = [d_A] + [AeBs] + [d_B]$
  $[AeBe] = [AeBs] + [d_B]$
- Given $[d_A]$, $[d_B]$ and $[AeBe]$
  $[AsBs] = [d_A] + [AeBe] - [d_B]$
  $[AsBe] = [d_A] + [AeBe]$
  $[AeBs] = [AeBe] - [d_B]$

where $[d_A]$, $[d_B]$, $[Ab]$, $[Bb]$, $[Ae]$, and $[Be]$ are expressing duration of A, duration of B, begin of A, begin of B, end of A, and end of B respectively. ■

Example 6.1: Considering three temporal interval of actions, A, B, and C, following requirements are just be known:

- The duration of A is 20-units length.
- The duration of B is 10-units length.
- The duration of C is 16-units length.
- Beginning of A is before beginning of B for 30 units.
- End of B is after beginning of C for 13 units.

In integrated temporal algebra, the information could be denoted as:

$[d_A] = <_{20}$
$[d_B] = <_{10}$
$[d_C] = <_{16}$
$[AsBs] = <_{30}$
$[BeCs] = >_{13}$

We can derive the complete temporal knowledge after following derivation.

Deriving formal endpoint relations:

$[AsBe] = [AsBs] + [d_B] = <_{30} + <_{10} = <_{40}$
$[AeBs] = -[d_A] + [AsBs] = >_{20} + <_{30} = <_{10}$
$[AeBe] = -[d_A] + [AsBs] + [d_B] = >_{20} + <_{30} + <_{10} = <_{20}$
$[BsCs] = [d_B] + [BeCs] = <_{10} + >_{13} = <_{3}$
$[BsCe] = [d_B] + [BeCs] + [d_C] = <_{10} + >_{13} + <_{16} = <_{13}$
$[BeCe] = [BeCs] + [d_C] = >_{13} + <_{16} = <_{3}$
$[AsCs] = [AsBs] + [BsCs] = [AsBe] + [BeCs] = <_{27}$
$[AsCe] = [AsBs] + [BsCe] = [AsBe] + [BeCe] = <_{43}$
$[AeCs] = [AeBs] + [BsCs] = [AeBe] + [BeCs] = <_{7}$

$$[AeCe] = [AeBs] + [BsCe] = [AeBe] + [BeCe] = <_{23}$$

The temporal algebra system is proved as an algebraic group, with associative and transitive relations. We could compute timing from serial specifications.

7. Conclusions

In this paper, we develop an efficient methodology to analyze and eliminate the conflicts between actions of a set of agents in coordination systems. Our mechanism then constructs partial order relations among actions. We develop a temporal algebra system as synchronization data model to unify interaction event and time specification of agent actions and. The synchronization models are generalized by composing point temporal relations with qualitative and quantitative functions. We also construct an automatic reasoning framework in which to help automatic actions of agents design more easily. Temporal semantics can be extracted from the solutions. Authors could easily describe the synchronization behavior of actions with consistence.

The main contribution of this paper is to provide a mechanism for maintaining synchronization constraints with detecting and eliminating conflicts. We hope this system could benefit other interesting researches and applications such as mobile computing and e-commerce applications etc.

References

[1]   Thomas W. Malone, "What is coordination theory?" in Proceedings of National Science Foundation Coordination Theory Workshop, MIT, 1988.

[2]   Jacques Ferber, "Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence," Addison Wesley Longman Inc., 1999.

[3]   Christos G. Biniaris, Antonis I. K., Dimitra I. K., and Iakovos S. V., "Implementing Distributed FDTD Codes with Java Mobile Agents," IEEE Antennas and Propagation Magazine, Volume: 44 , Issue: 6 , Dec. 2002, pp.115 - 119

[4]   Yerang Hur and Insup Lee, "Distributed Simulation of Multi-Agent Hybrid Systems,", In Proceedings. Fifth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing 2002. (ISORC 2002), 29 April-1 May 2002, pp.356 – 364.

[5]   Shivakant Mishra and Peng Xie, "Interagent Communication and Synchronization in DaAgent,"in Proceedings of 21st International Conference on Distributed Computing Systems, 16-19 April 2001, pp.699 – 702.

[6]   Thomas D. C. Little and Arif Ghafoor, "Interval-Based Conceptual Models for Time-Dependent Multimedia Data," IEEE transactions on knowledge and data engineering, Vol. 5, No. 4, 1993, pp. 551- 563.

[7]   Karim Hussein, Feniosky Pena-Mora and Ram D. S., "CAIRO: A System for Facilitating Communication in a Distributed Collaborative Engineering Environment,", in proceedings of the Fourth Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, 20-22 April 1995, pp.154 – 162.

[8]   Timothy K. Shih and Anthony Y. Chang, "The Algebra of Spatio-Temporal Intervals," in Proceedings of the 12th International Conference on Information Networking, Japan, January 21-23, 1998, pp.116-121.

[9]   Maria Jose Perez-Luque and Thomas D. C. Little, "A Temporal Reference Framework for Multimedia Synchronization," in the IEEE Journal on Selected Areas in Communications, Vol. 14 No. 1, January 1996, pp.36-51.

[10]   A. Safavi and S. F. Smith, "An Evaluation Function to Compare Alternative Commitments During Manufacturing Planning and Scheduling," in Proceedings of First International Conference on Expert Planning Systems, 27-29 Jun 1990, pp.22 – 27.

[11]   Andrea Gozzi, Massimo P., and Antonio B., "A Multi-Agent Approach to Support Dynamic Scheduling Decisions," in Proceedings. ISCC 2002. Seventh International Symposium on Computers and Communications, 1-4 July 2002 , pp.983 – 988.

[12]   Lisa Cingister DiPippo, and Victor Fay-Wolfe, "A Real-Time Multi-Agent System Architecture for E-Commerce applications," in Proceedings of 5th International Symposium on Autonomous Decentralized Systems, 26-28 March 2001, pp.357 – 364.

[13]   Yahya Y. Al-Salqan and Carl K. Chang, "Temporal Relations and Synchronization Agents," IEEE Multimedia, Vol.3, No.2, pp.30-39.

[14]   L. Li, A. Karmouch and N.D. Georganas, "Multimedia Teleorchestra with Independent Sources: Part I - Temporal Modeling of Collaborative Multimedia Scenarios," in Multimedia Systems, ACM Press and Spring Int′l, Berlin, Feb. 1994, pp.143-153.

[15]   Apple Computer, Inc., QuickTime Developer's Guide, 1991.

[16]   G. D. Drapeau and H. Greenfield, "Maestro- a Distributed Multimedia authoring environment," in proceedings of Summer 1991USENIX Conference, 1991, pp.473.

A Signature Verification Based User Interface Agent for Learning Portfolio System

Wei-Hsien Wu[1], Jeng-Horng Chang[2], Ping-Lin Fan[3], Min-Tai Chou[1,*], and Hui-Fen Chiang[1]

1. Department of Information Management, Kuang Wu Institute of Technology, Taipei, Taiwan, R.O.C. * Email: kwangwue@ms2.hinet.net
2. Department of Information Management, Takming College, Taipei, Taiwan, R.O.C.
3. Graduate School of Toy and Game Design, National Taipei Teachers College, Taipei, Taiwan, R.O.C.

Abstract

A biological features based identification scheme that used in a user interface agent (UIA) is proposed. Two features embedded in signature are extracted, then, Statistic measurement and similarity test are used to verify user identity.

1. Introduction

This paper proposes a user identification method for learning portfolio in an e-learning system. To ensure the user who should complete the designated lesson via world wide web, randomly asking the user to input his/her private information is necessary[1]. However, the information could be disclosed to others, and thus if the user complete the lesson by himself is still unknown. Biologic features are suitable for using in such identification application because it is hard to imitate.

Biological features that are suitable for identification in e-learning application should satisfy the following two conditions:

(1) easy to input;

(2) input device is not expensive.

Frequently used biologic features include signature and fingerprint and iris. The latter two are often used in security system. However, the input device is much expensive relative to that of the former. For a signature verification system, a cheaper graphic tablet is enough to serve as the input device.

In this paper, "signature" is not restricted to "name", but all writing words.

2. System overview

Shown in Figure 1 is a learning system with the proposed UIA. Each client computer is equipped with a graphic tablet. One primary job of UIA of the system is to acquire signature information to verify user identity. The flow chart is illustrated in Figure 2. First, while a student selects a curriculum and registers, the UIA displays an article and asks the student to write down that article word by word several times on the tablet. The UIA then extracts futures embedded in each word and pass the information back to the learning server. Every time when the student logins the curriculum, the UIA will randomly choice several words from the article in a

period and ask the student to write down. The input signature is matched with the saved feature templates. If a specified percentage of correction rates are reached, the student passes the verification, and the lesson continues. Otherwise, the UIA will stop the lesson and mail a message to the registered student to ask for explanation, or simply record in the portfolio.

3. The verification scheme

An on-line signature verification system[2,3] is generally divided into 3 stages. The first stage acquires signature information including: coordinate and pressure of each point, pen down and pen up locations, pen inclination, curvature, speed of chirography, signing time, and so on. The second stage converts the input information into other domains, extracts features from the converted data and forms feature vector. Finally, in the last stage, verification techniques are conducted on the feature vector and generate result. The verification techniques include neural network, Hidden Markov Model, statistical classifier, dynamic programming, genetic algorithm, etc. More comprehensive survey could be found in [2,3].

In this paper, we propose an on-line Chinese signature verification scheme in which information embedded in strokes is extracted. The successful verification rate is about 86% for a total of 30 persons' testing samples. Since the verification rate is not necessary being very high in the proposed UIA, the proposed signature scheme is feasible.

3-1 Feature extraction

Features to be extracted should satisfy two conditions: (1) stable characteristics to the signature owner; (2) easy to discriminate from fake signature. Roughly, there are two categories of signature features: one exposes to the public, the other hides deeply as secret. The former refers to the characteristics which can be analyzed from the signature such as number of stroke, angle between strokes, point distribution, direction/angle deviation of point/stroke, outline, histogram, power spectrum, etc; the latter is more detail concerning one's writing custom, including: total signing time, velocity of pen moving, pen-up and pen-down location, pressure, tilt and azimuth of pen (pen inclination), and so on. In this paper, we adopt two features: total signing time and directional feature vector of an outline of a special 2D signature image.

3-2 verification

Two values are generally used to evaluate the performance of a signature verification system. One refers to a ratio of genuine signature rejected by the system, and the other is that of false signature accepted by the system. The former is in term of false rejection rate (FRR, Type I error), the latter is in term of false acceptance rate (FAR, Type II error).

3-2-1 Total signing time

The upper bound and lower bound of the feature are formulated as:

$$ST_{upper} = Mean_{ST} + \beta \bullet \sigma_{ST}$$

$$ST_{lower} = Mean_{ST} - \beta \bullet \sigma_{ST}$$

where: $Mean_{ST}$ and σ_{ST} are mean value and standard deviation of total signing time of all training samples, respectively. β is a constant to adjust the range.

If the feature of an input signature is located

between lower and upper bound, then it passes the test. When β increases, it means easier to pass the test. Consequently, FRR declines and FAR rises.

3-2-2 Directional feature vector

The directional feature vector X is defined as $\{X/, X\text{-},$ $X/, X\backslash\}$, in which each element represents a feature in one direction. To extract directional feature, the original signature information is first converted into a 2D image (refer to Figure 3.) according to the point distribution and writing speed. We first vertically divide the image into several partitions so that there is the same number of black pixels in each vertical partition. Then, the image is partitioned horizontally in the same way. The next step scans the pixel from left to right and from top to bottom. For every pixel belongs to outline of the signature in an area of 2×2 grids, find and count its neighbor (also an outline point) from 4 directions: south, east, northeast, and southeast. Four variables denoted as $X/, X\text{-}, X/,$ and $X\backslash$ are used to record the ratio of each directional neighbor appearing in all the areas.

Let there be N training samples for each person, and denote as $X_1, X_2,..., X_N$. We first find X_n to be the reference template which is highly similar to the other. Correlation test When $R(X,Y)$ is used to do the similarity measure. When $R(X,Y)$ is closer to 1 means higher similarity between X and Y.

First, find $S(i)$, i = 1, 2, …, N, $S(i)$ is defined as:

$$S(i) = \frac{1}{N-1} \sum_{k=1, k\neq i}^{N} R(X_i, X_k).$$

Next, find $S(n) = \max(S(1), S(2),..., S(N))$, then the n-th sample is the one that owns the highest similarity with the other. Hence, X_n is chosen to be the reference template.

Last, mean ($Mean_X$) and standard deviation (σ_X) of set $\{R(X_n,X_i)|i=1, 2, …,N; i \neq n\}$ are calculated.

When a test template Y is input, find $d = R(X_n,Y)$. If d satisfies the condition: $Mean_X - \alpha \bullet \sigma_X \leq d \leq Mean_X + \alpha \bullet \sigma_X$, then it passes the test. α is a constant to adjust the acceptance range.

4. Conclusions

Since the UIA plays a role of assistant, biological features used in identification application in learning system should be as simple as possible. In such application, high verification rate is not necessary, but the efficiency is more concerned.

One disadvantage is that user who wants to learn from the proposed system should install a graphic tablet at the client. However, the hardware is cheaper and cheaper, the negative influences will be decrease in term of time.

References

[1] Newmarch, J. *"Courseware on the Web:An Analysis of Student Use"*, AusWebp7 The Third Australian World Wide Web Conference, 1997.

[2] V. S. Nalwa, "Automatic On-Line Signature Verification," IEEE Proceedings, Vol. 85, No. 2, pp. 215-239, 1997.

[3] Gupta, J and McCabe, A. "*A Review of Dynamic Handwritten Signature Verification*," James Cook

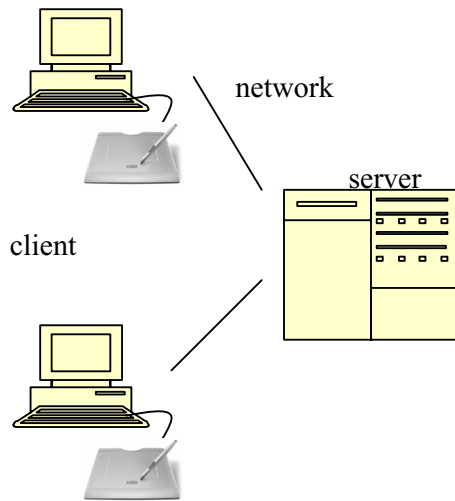University, Australia (1997).
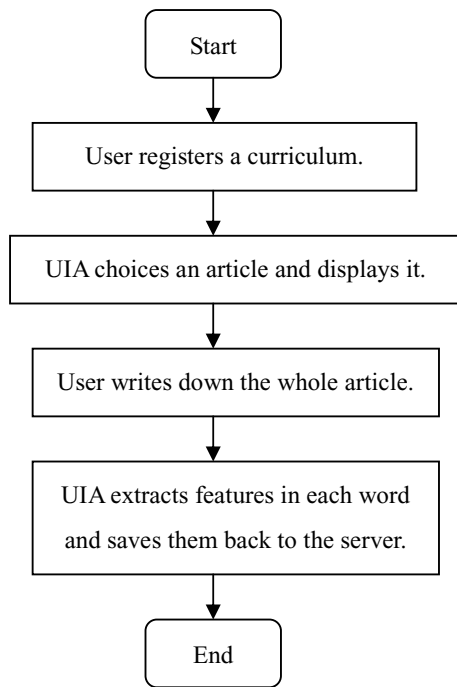

Figure 1. The proposed learning system

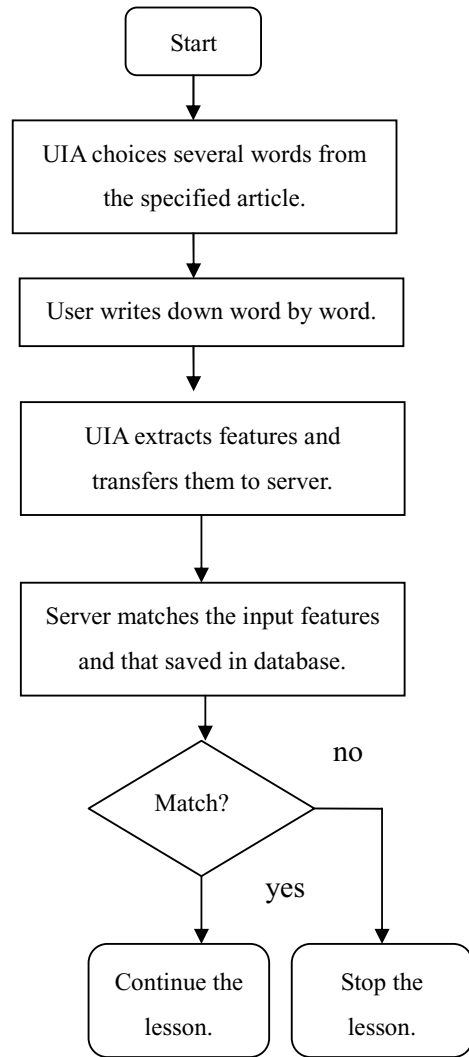Figure 2(a) stage of extracting features for each student.

Figure 2(b) stage of identification.
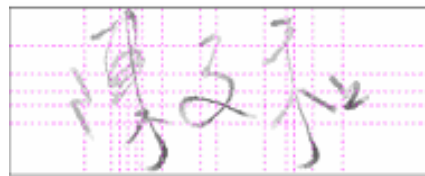

Figure 3. Non-uniform partition example.

Real-time Person authentication in E-learning Environments by FAFVS

Jeng-Horng Chang [a*], Wei-Hsien Wu [b], Hui-Fen Chiang [c]

[a] Takming College, Department of Information Management, Taipei, Taiwan, R.O.C.

[b] Department of Mathematics and Information Education, National Taipei Teachers College, Taipei, Taiwan, R.O.C.

[c] Department of Information Management, Kuang Wu Institute of Technology, Taipei, R.O.C.

[*] Corresponding author. Tel: +886-2-658-5801 ext 5229

email address: jhchang@mail.takming.edu.tw (J.-H. Chang)

Abstract: Along with the development of Internet and information technology, there has been a huge revolution on the sort of learning, since we have involved in a new century of e-learning. However, it also shows the problem about the difficulties of the management and integration of learning resources. Especially, the person who seated behind the cable-line is always unknown. We proposed the possible solution to the person authentication by integrating the optical fingerprint-scanning mouse in the market and the new developed FAFVS to resolve this difficulty.

1. Introduction.

Along with the development of Internet and information technology, there has been a huge revolution on the sort of learning, since we have involved in a new century of e-learning. At the current time, the learning resources accumulating in the Internet has been abounding gradually [1]. Following the rapid development of Internet, there exist various Internet activities among which the E-learning has been unprecedented emerging due to the championship of Life-long learning and recognition of asynchronous learning by government in addition that the E-learning is not constrained by specific space, location, objects and time.

However, it also shows the problem about the difficulties of the management and integration of learning resources [2]. Plenty of problems should be resolved, such as the difficulty on seeking resources and tests, low reusability, incapability of the transformation on different learning platforms, etc. Especially, the person who seated behind the cable-line is always unknown. For the person identification problem involved in the E-learning topic, we are incapable to

identify the real class member.

Thus, we proposed the possible solution to the person authentication for E-learning environments by integrating the optical fingerprint-scanning mouse in the market and the new developed Fast Automatic Fingerprint Verification System, FAFVS, to resolve this difficulty. In this paper, the major topic is then transferred to the design of a sophisticate system of fingerprint verification technique.

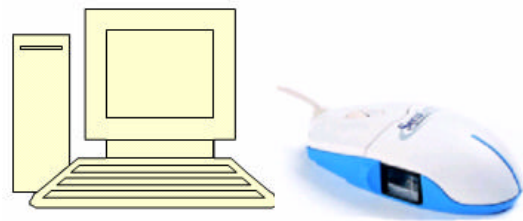## 2. Overview of the System

Figure1. Real-time fingerprint capturing mouse.

For the person authentication on the Internet, biometrics verification, such as fingerprint, human face, voice, provide the most reliable problem solving approaches. Currently, the optical real-time fingerprint capturing mouse, illustrating in figure 1, provides the best performance for our application in the E-learning

environments.

In our system, the fingerprint are pre-scanned and registered in a third-party server. When a participator of an E-Learning class attempted to login, a two phase person identification process will start: First, the users should key-in their personal ID and password for the enrolled class, and the application program sent a time-stamp with the id to acquire the password to verify. The roll of the time-stamp in the password checking stage is to avoid a malicious password generating program. If the password can not be typed in a certain period, or the password has been typed more than three times, the user should be abandoned.

The second step is to scan the fingerprint of the user by the optical fingerprint-scanning mouse if the password checking process is successes. The fingerprint scanning and verifying process should be process repeat every several minutes until the class participator log-off the E-learning class. Since the fingerprint scanning, processing and verification steps will be processed many times. We need a methodology that can perform the fast fingerprint processing and feature extraction processes.

3. The Fast Automatic Fingerprint Verification System (FAFVS)

In order to provide the fast fingerprint verification process, the pre-processing of fingerprint images should be reduced. In this paper, we propose a novel method to segment gray-level fingerprint images with nonuniform illuminanced regions. This proposed method is a local and point-dependent approach. In our method, the smoothed gray-level histogram is formulated as a *mixture Gaussian distribution* that has been proved by Zhuang [3]. The only assumption in our method is that the gray-level histogram of a certain fingerprint image is constructed by two or three Gaussian distributions. We use statistical approach and some heuristic parameters to decompose this histogram into non-overlapping distributions. Then, the mean and variance values of

each distribution can be optimally determined. We must emphasize that our method does not employ conventional parameter estimation procedure by iterative parameter refinement. The segmentation time is thus greatly speedup.

Figure 2. The gray-level histogram decomposition results.

Let the image size be x by y and the window size for partition the entire image be w by w. If both x and y are not a multiple of the window size w, some rows and columns should be added onto the image for the windowing operation. Let the increasing number of columns and rows be ax and ay, respectively. Thus,

$$ax = w - x \bmod w$$

$$ay = w - y \bmod w$$ Then, perform the image resizing algorithm to enlarge the image size to [$x+ax$, $y+ay$], see Fig. 3. After finishing the binarization process, the image should be reduced to its original size.

Figure 3. Partition image with *w by w* overlapping windows to reduce blocky effect.

In the proposed method, a w by w pixels window is

created and sliding by *w/2* pixels from left to right, then from top to bottom to partition a gray-level image into many square subimages as illustrated in Figure 6. Since our segmentation method is based on local thresholding, the overlapping windows can prevent blocky effect.

The threshold value in each window is designed as the mean value of the pixels in this sub-image. If the standard deviation of the block is less than the value *S*, we use the estimated value T_B as the threshold value to eliminate the image background. After finishing the binarization process for each block, two possible methods can be used to accomplish the recombination process for the overlapping windows. They are the AND operation and the OR operation. As shown in Figure 4 (a), the AND operation actually an intersect operation. That is, the resulting binarization will be black if both the corresponding pixels in the overlapping block are all appear to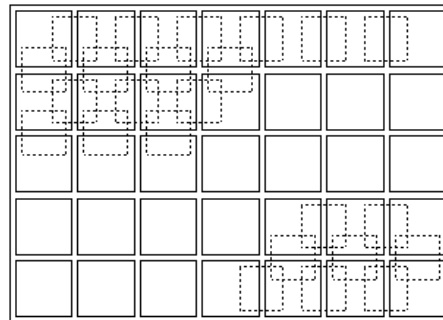 be black. On the other hand, as shown in Figure 4 (b), the OR operation will produce a black pixel if either one pixel of the corresponding image is black. The segmentation algorithm is summarized as follows:

w = window size
image ←augment image according to *w* T_B and T_R ←
         global histogram threshold values
         selection
[*row, col*] ← size of ***image***
for ***i*** ← 1 to *row-w+1* by *w/2*
  for ***j*** ← 1 to *col-w+1* by *w/2*
    window ← *image(i:i+w-1,j:j+w-1)*

    if (standard deviation of pixels in ***window*** < ***S***)
        T ← T_B
    else
        T ← mean value of pixels in ***window***
    end

    window_bw ← thresholding *window* according to
        T
    image_bw(i:i+w-1,j:j+w-1) ←
     AND*(image_bw(i:i+w-1,j:j+w-1), window_bw*)
  end
end
binary image ←reduce ***image_bw*** to original size

Figure 4. (a) The AND operation of two overlapping blocks, (b) the OR operation.

4. Experimental Results

Some experiments have been conducted to evaluate the performance of the proposed fingerprint image segmentation algorithm with *NIST Special Database 4* fingerprint images [4]. The fingerprint images were acquired and quantized into 512×512 with 256 gray levels in the test data set. The window size *w* and the variation measure *S* are two parameters that are not automatically given in out method. For the variation measure *S*, which are used to detect the background region, are set to a constant value 10 because the variation of the background is always very low. We perform our fingerprint segmentation algorithm with different window size and different overlapping operation.

Original image   AND operation   OR operation

Global mean   AND operation + Median filter   OR operation + Median filter

Figure 5. A sub-image with ink deficiency and the segmentation results.

Figure 5 shows a sub-image with ink deficiency region. The segmentation result by applying AND and OR operation both produce better results than a single threshold. For the spur points in the OR operation, perform median filtering can alleviate this situation. Figure 6 demonstrates the feature extraction results.

Figure 6. The feature extraction results of ridge ending (square) and the ridge bifurcation (circle).

5. Summary and future works

In this paper, we proposed a novel approach of internet person authentication problem for E-learning environments by fingerprint verification. Experiments show the FAFVS can work faster than other fingerprint processing system because we reduced the preprocessing stages that are necessity of the traditional fingerprint verification system.

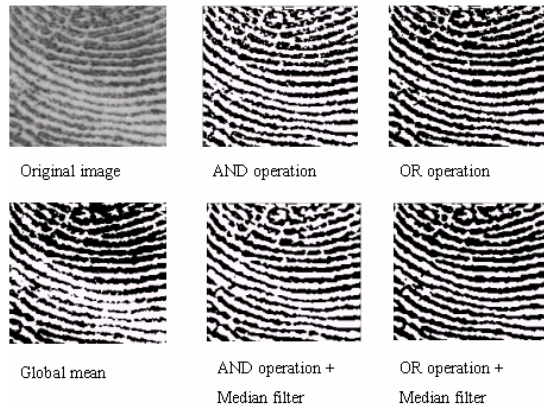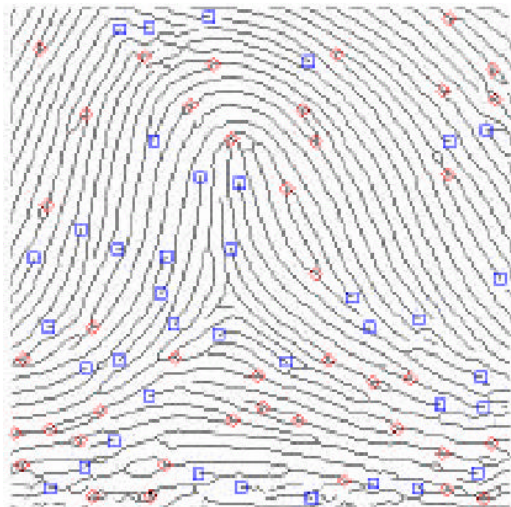However, the disadvantages of our system incorporated only fingerprints probably reject some accidental injured fingers. The person authentication system can be more sophisticate if our system can consider other biometrics. Figure 7 is a conceptual idea that use three biometrics, fingerprint, human face and voice, as the cues of a person verification system. In this system, user say the password and then perform speaker verification. If the speaker can be identified correctly, the password verification will be processed to ensure the user. However, if the speaker can not be identified, the system retrieve n most faces which are the most similar person of the voice. Then perform fingerprint verification process to the n faces to identify one. This system can promote our system performance for lower false-rejection rate(FRR) and is more suitable for a E-learning environments.

Figuer 7. A low FRR system using fingerprints, faces and voice of the participator.

Acknowledgements

Refereces

[1]. Ashraf A. Kassiml, Kazi Sabbir Ahmed and Surendra Ranganath (2001). A Web-Based Intelligent Approach To Tutoring. International Conference on Engineering Education, August 6-10, Oslo, Norway.

[2]. Brusilovsky, P. and Vassileva, J (2003), Course sequencing techniques for large-scale web-based education. International Journal of Continuing Engineering Education and Lifelong Learning 13 (1-2).

[3]. Zhuang X., Huang Y., Palaniappan K. and Zhao Y., (1996), Gaussian Mixture Density Modeling, Decomposition, and Applications. IEEE Trans. On Image Processing, 5(9), 1293-1302.

[4]. Watson C. I. and Wilson C. L.(1992), Fingerprint Database, National Institute of Standards and Technology, Special Database 4, FPDB.

Exchange Rate Prediction Agent Based on Gray Theory

Kuan-Cheng Lin, Min-Tzu Wang, Jeff T. C. Lee and Shu-Huey Yang*

Department of Information Management
Kuang Wu Institute of Technology
Peitou, Taipei, Taiwan 112, R. O. C.
kuanchenglin@kwit.edu.tw

*Da-Long Primary School
Da-tong, Taipei, Taiwan 113, R. O. C.
nnfuzzy@tp.edu.tw

Abstract:

In this study, we proposed an exchange rate prediction agent based upon the gray theory and neural network. The prediction agent uses the gray relation analysis to obtain the input variable of the neural network model and forecast the exchange rate.

Keywords: Price Prediction, Grey Prediction Model, Agent

1. Introduction

The exchange rate prediction plays an important role on the business prediction. The adequate exchange rate prediction can help enterprise to make correct decision in hedge, financing, investment and budget. Usually, technical forecasting and fundamental forecasting are used to predict the exchange rate. The technical forecasting uses the historical date of exchange rate to predict the future exchange rate. In fundamental forecasting, the exchange rate is determined by the following theory [1][2][3]:

(1) Purchasing power parity theory.
(2) Balance theory of international payment.
(3) Theory of monetary approach.
(4) Interest rate parity.
(5) Portfolio balance approach.

However, for the reasons of insufficient data or uncertainty, the traditional forecasting method is generally hard to predict the optimal price. In this study, we describe two types of exchange rate prediction agent based upon the gray theory and neural network. The prediction agent uses the gray relation analysis to obtain the input variable of the neural network model and forecast the exchange rate.

The remainder of this paper is organized as follows: Sections introduced the gray theory and neural network that the exchange rate prediction agent used. Section 3 describes the exchange rate prediction algorithm. Section 4 makes conclusions and future research.

2. Gray Theory and Neural Network

2.1 Gray Relation Analysis

Deng introduces the gray theory in 1982 [4][5]. And, gray relation analysis is one of the most important theory of gray theory. The gray relation was originally proposed to relate the main factor to the other reference factors in a given system.

Definition 1: Gray relational coefficient $r((x_i(k), x_j(k))$

In the gray relational space $\{P(X); \Gamma\}$, there is a sequence x_i $(x_i(1), x_i(2),…, x_i(k))$ *belong to X, for i=0,1,2,…,m, k=1,2,3, …,n.*

The gray relational coefficient $r((x_i(k), x_j(k))$ is defined as follows:

$$r((x_i(k), x_j(k)) = \frac{\Delta \min . + \varsigma \Delta \max .}{\Delta_{oi}(k) + \varsigma \Delta \max .}$$

for i=1,2,…,m, k=1,2,3, …,n.

x_0 is the reference sequence, x_i is the specified comparison sequence.

$$\Delta_{oi} = \| x_0(k) - x_j(k) \|$$

$$\Delta \min . = \overset{\min.}{\forall j} \in i \overset{\min.}{\forall k} \| x_0(k) - x_j(k) \|$$

$$\Delta \max . = \overset{\max.}{\forall j} \in i \overset{\max.}{\forall k} \| x_0(k) - x_j(k) \|$$

$\varsigma \in [0,1]$ is the distinguishing coefficient.

Definition 2 Gray relational grade $r((x_i, x_j)$

$$r((x_i, x_j) = \frac{1}{n} \sum_{k=1}^{n} r(x_i(k), x_j(k))$$

Definition 3 Gray relational Ordinal

For reference sequence x_0, and the specified comparison sequence x_i, let $x_0 = (x_0(k))$, $x_j = (x_j(k))$ *for i=0,1,2,…,m, k=1,2,3, …,n.*

If $r((x_0, x_i) \geq r((x_0, x_j)$, the gray relational ordinal is defined as $x_i \succ x_j$.

And, the gray relational grade $r((x_0, x_i)$ is greater than gray relational grade $r((x_0, x_j)$.

2.2 Neural Network

Owing to that proposed scheme is based on artificial neural network model, this section introduces the computation of artificial neural network (ANN) model, referred to as two-layer functional-link neural network model [7]. The ANN model is used primary since the Ann model has rapid computational ability owing to its' massive parallel structure [10]. The input unit is connected with the output unit and the functional-link unit, which is formed by the modified perceptron algorithm we have presented in [8][9].

3. Exchange Rate Prediction Agent System

The web-based stock price prediction agent system is constructed on http://203.64.218.5/exchangerate by using JavaServer Page [6][15]. The prediction agent system has two functions: technical forecasting and fundamental forecasting.

3.1 Technical forecasting

There are many technical forecasting factors introduced, such as relative strength index (RSI), Moving Average Convergence and Divergence (MACD), Stochastic Line (KD), On Balance Volume (OBV), BIAS, and so on. We first establish the raw sequence for the technical forecasting factors and the exchange rate. Then, we compute the gray relational grade for the technical forecasting factors and the exchange rate to obtain the gray relational ordinal. Therefore, we choose the more important technical forecasting factors as the input

variable and the exchange rate as the output. Finally, we the neural network is trained to forecast the exchange rate.

3.2 Fundamental forecasting

There are many fundamental forecasting factors introduced in the literatures [1][2][3], such as foreign exchange reserve, relative inflation rate, nominal money supply, current account, interest rate differential, and so on. We first establish the raw sequence for the fundamental forecasting factors and the exchange rate. Then, we compute the gray relational grade for the fundamental forecasting factors and the exchange rate to obtain the gray relational ordinal. Therefore, we choose the more important fundamental forecasting factors as the input variable and the exchange rate as the output. Finally, we the neural network is trained to forecast the exchange rate.

4. Conclusions

In this paper, a web-based exchange rate prediction agent system based on gray theory and neural network was proposed. The he prediction agent system has two functions: technical forecasting and fundamental forecasting. The stock price prediction can be completed on-line because the algorithm of gray relational and neural network is simple and fast.

REFERENCES

[1] Frenkel, "Purchasing Power Parity:Doctrinal Perspective andEvidence from the 1920s", Journal of International Economics8,no.2,pp.169-191,1978.

[2] Frenkel Jeffrey,"On the Market:A Theory of Floating Exchange Rates Based on Real Interest Rate Differentials",American Economic Review 69,pp.610-622,September 1979.

[3] MacDonald,Ronald and Mark P. Taylor,"The Monetary Approach to the Exchange Rate:Rational Expectations, Long-RunEquilibrium and Forecasting",IMF staff Papers 40,pp.89-107,1993.

[4] J.L. Deng, "Control problem of Grey system," *System Control Letter,* Vol.1 No.1, 1982, pp.288–294.

[5] J.L. Deng, "Introduction to Grey system," *The Journal of Grey System,* Vol.1, No.1, 1989, pp.1–24.

[6] JavaServer Pages (**JSP**) Specifications http://java.sun.com/**jsp**

[7] D. Sobajic, Artificial neural network for transient stability assessment of electric power systems, Ph. D. Thesis, Computer Science Dept., Case Western ReserveUniv. Cleveland, OH.,(1988) 55-69.

[8] Y. P. Chu, P. C. Chang and C. M. Hsiech, A learning algorithm for generating minimal perfect hashing function, Journal of Electrical Engineering, Vol.34, No.4, Aug. (1991) 271-277.

[9] Y. P. Chu and C. M. Hsieh, An Artificial Neural Network Model with Modified Perceptron Algorithm, Parallel Computing 18, (1992) 983-996.

[10] David Bailey, Donna Thompson, How to develop neural network, AIEXPERT, June, (1990) 38-47.

[11] Philip D. Wasserman, Neural Computing Theory and Practice, (ANZA Research, 1989).

[12] J. A. Feldman and D. H. Ballard, Connectionist models and their properties, Cognitive Science 6, (1982) 205-254.

[13] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning internal representations by error propagation, Parallel Distributed Processing: Explorations in the [14]

[14] Microstructures of Cognition, Vol.1, (1988) 318-362. D. E. Rumelhart, Geofferey E. Hinton, and Ronald J. Williams, Learning representations by

back-propagating errors, Nature 323, (1986) 533-536.

[15] Kuan-Cheng Lin, Jeff T. C. Lee, Min-Tzu Wang and Shu-Huey Yang "Stock Price Prediction Agent Based on Gray Theory", in proceedings of the 2004 International Workshop on Mobile Systems, E-commerce, Agent Technology (MSEAT'2004), San Francisco, California, USA, Sep. 8-10, 2004.

An Agent-Based Admission Control for the Bounded Delay Services

*Kuan-Cheng Lin, Ren-Jing Huang and Yen-Ping Chu

*Department of Information Management
Kuang Wu Institute of Technology
Peitou, Taipei, Taiwan 112, R. O. C.
kuanchenglin@kwit.edu.tw

Department of Applied Mathmatics
National Chung-Hsing University
Tai-Chung, Taiwan, R.O. C.

Abstract

In this study, we propose an call admission control agent improve the network utilization by using $\sigma(\rho)$ (burstiness curve) traffic specification instead of a single set of (σ,ρ) while the complexity of the network is still the same as the network of the single leaky bucket traffic model. Keywords: Call admission, bounded delay service, agent

1. INTRODUCTION

The performance of a bounded delay service is largely influenced by three factors: (1) the specification which describes the worst case traffic from a connection, (2) the scheduling discipline the network switches use, and (3) the accuracy of the admission control functions. The main design goal is to maximize the performance of the network; that is, to maximize the number of connections that can be supported without violating any bounded delay guarantees. In this study, we focus on issue (1) and (3). We improve the network utilization by using burstiness curve traffic specification $\sigma(\rho)$ instead of single (σ,ρ). We define the burstiness curve $\sigma(\rho)$ of a message as the maximum number of bits that must be buffered at a node if messages are allocated at a fixed rate ρ bps, while the complexity of the network is still same as

the network of a single leaky bucket model. The network employs a DJ regulator to reconstruct the traffic pattern and first-come-first-serviced (FCFS) to schedule the service order of each input packet. By using the call admission formula of [1] which is based on a single leaky bucket traffic, we derived the close form admission control formula for burstiness curve traffic model.

The user needs to propose a traffic specification for resource allocation. By analyzing the packet stream of input data, the user gets a burstiness curve $\sigma(\rho)$, which can be used as the traffic specification. If the allocation rate (ρ) is smaller than the average rate of input data, the queue will build infinitely. If the allocation rate is larger than the peak rate of input data, the extra rate is wasted. So the allocation rate is between the average rate and peak rate of input data. We can increase the value of the burst (σ) and lower the value of the rate (ρ), all these sets of (σ,ρ) values can still conform the input data stream as shown in figure 1. According to [2], the knee of burstiness curve is good choice for getting maximum number of connections. A knee in the burstiness curve is a distinct indication that for descriptors that are slightly away from the knee, either the σ or the ρ parameter rapidly increases. In our study, this is not always correct, since the optimal class depends on the environment of the network.

2. Algorithm of call admission control agent

2.1 Network Architecture and Traffic Model

In [1], we proposed a network architecture that permits N kinds of (σ_i, ρ_i) classes of input traffics. Class j conforms to leaky bucket (σ_j, ρ_j). The architecture of switch nodes of the network is shown in figure 2. At the first node of each connection, we add a leaky bucket regulator to policy the input data. At each hop, we add a delay jitter regulator to reconstruct the traffic pattern. The characteristics of the source traffic are modified as the source traffic passes through the network. We add a Delay Jitter (DJ) regulator [3] for each connection at the switch to reshape the traffic pattern to conform to the same traffic pattern at the first node before the traffic enters the FCFS queue. By using burstiness curve traffic specification, users send piecewise (σ_j, ρ_j) parameters to the network as shown in figure 2.

2.2 Joint Selection

The input data considered in this paper can be audio, MPEG, or JPEG etc. Of the different data types, MPEG Video is most highly burst, so we use MPEG as an example here. According to [4], for an input data, such as MPEG video stream, there is an unique $\sigma(\rho)$ function. According to [2], the knee of burstiness curve is a good choice. Since the goal of the network is to maximize the number of connections, the optimal class is dependent on the call admission policy. We propose a method to jointly select the (σ, ρ) class according to the admission control policy. The user provides the burstiness curve in piecewise form. The call admission policy will select the class with maximum function value in order to get maximum number of connections. In this study, the schedule policy can be either OPT (optimal) or EQ (Equal Allocation).

2.3 Call Admission Formula

For the proposed network architecture, call admission policy can be either optimal distribution (OPT) or equal distribution (EQ). We neglect propagation and node processing delays. In [1], two formulas are derived for OPT and EQ policy. We use these formulas as our cost functions. The user can put parameters of each class into this cost function. The class with the maximum value is the optimum.

Theorem 1 For the proposed network with OPT policy, consider a connection passing through n switches connected in cascade and the bandwidth of node i is l_i. The delay requirement is Q. Assumed current number of connections of class $1, 2, \dots, N$ at node i is $N_1^i, N_2^i, \dots, N_N^i$, respectably, then the number of connections of class k that is permitted to enter again is N_{OPT}. We define $N_{OPT}(k)$ as the cost function of OPT policy.

$$N_{OPT}(k) = min\{N_\sigma(k), N_\rho(k)\} \quad (1)$$

Where N_σ is the number of connections depend on σ and N_ρ is the number of connections depends on ρ.

$$N_\sigma(k) = \left\lfloor \frac{Q - \sum_{i=1}^{n} \frac{\sum_{j=1}^{N} N_j^i \sigma_j}{l_i}}{\sigma_k \sum_{i=1}^{n} \frac{1}{l_i}} \right\rfloor \quad (2)$$

$$N_\rho(k) = min\left\{ \left\lfloor \frac{l_i - \sum_{j=1}^{N} N_j^i \rho_j}{\rho_k} \right\rfloor \quad i = 1, 2, \cdots, n \right\} \quad (3)$$

Theorem 2 For the proposed network with EQ policy, considers a connection passing through n switches connected in cascade and the bandwidth of node i is l_i. The delay requirement is Q. Assumed current number of connections of class $1, 2, \dots, N$ at node i is $N_1^i, N_2^i, \dots, N_N^i$, respectably, then the number of connections of class k that is permitted to enter again is N_{EQ}. We define $N_{EQ}(k)$ as the cost function of EQ policy.

$$N_{EQ}(k) = \min\{N_\sigma(k), N_\rho(k)\} \quad (4)$$

Where N_σ is the number of connections depends on σ and N_ρ is the number of connections depends on ρ.

$$N_\sigma(k) = \min\{\frac{\frac{Ql_i}{n} - \sum_{j=1}^{N} N_j^i \sigma_j}{\sigma_k}, i=1,2,\cdots,n\} \quad (5)$$

$$N_\rho(k) =$$

$$\min\{\lfloor \frac{l_i - \sum_{j=1}^{N} N_j^i \rho_j}{\rho_k} \rfloor, \ i=1,2,\cdots,n \} \quad (6)$$

Definition For the proposed network, it is assumed that there are n sets of (σ,ρ) class making up the input data stream. The n classes are class $C_1, C_2, \cdots, C_n$. Without lose generality, we assume that $\sigma_{C1} > \sigma_{C2} > ... > \sigma_{Cn}$ and $\rho_{C1} < \rho_{C2} < ... < \rho_{Cn}$. The choice for the scheduling policy can be either OPT or EQ. We define the optimal class as

Optimal class(C_i) for OPT policy

$=C_k$ such that $N_{opt}(C_k)=Max\{N_{opt}(C_i), i=1,2,...,n\}$

Optimal class(C_i) for EQ policy

$=C_k$ such that $N_{EQ}(C_k)=Max\{N_{EQ}(C_i), i=1,2,...,n\}$

With the equation (1) to (6), we can develop the formula to select the optimal class in the next section.

2.4 Optimal Class Selection Formula

To determine how the optimal class varies with the delay requirement, we use OPT policy as an example and analyze the formulas (1), (2) and (3) as follows. Suppose the network supports N classes (σ_1,ρ_1), (σ_2,ρ_2), … , and (σ_N,ρ_N), respectively. Consider a connection passing through M switches connected in cascade and the bandwidth of node i is l_i. The minimum bandwidth is $l_{min} = \min\{l_1, l_2, \cdots, l_M\}$. Supposed the current number of connections is N_i for class i. By analyzing the packet stream of input video, we get a burstiness curve. The user matches this σ-ρ curve with the support classes of the network and gets n sets of (σ,ρ) classes that can be used for the traffic specification. Suppose these n classes are class $C_1, C_2, \cdots, C_n$, and $\sigma_{C1} > \sigma_{C2} > ... > \sigma_{Cn}$ and $\rho_{C1} < \rho_{C2} < ... < \rho_{Cn}$. The delay requirement of input connection is Q. We analyze the relation of the admission region with a ____ y requirement for tandem network as below:

Case 1: In the case of a large delay requirement we assume Q is larger than $Q_{C_1}^*$ and we select class C_1.

When the delay requirement Q is larger, we know from equation (2) that N_σ is also larger. In this case, from equation (1), we know the number of connections permitted to enter is restricted by the N_ρ. From equation (3), we know the larger the ρ value, the smaller N_ρ will become. In order to have larger N_ρ we will select the class with the smaller ρ value. Since the ρ value of class C_1 is the smallest one among the ρ_i values of the classes C_i, $i=1,2,..., n$, the $N_\rho^{C_1}$ is the largest among the $N_\rho^{C_i}$.

Therefore, we select class C_1. When we follow the curve of class C_1 in the figure 3, we are able to get a delay requirement $Q_{C_1}^*$, such that $N_\sigma^{C_1}$ is equal to $N_\rho^{C_2}$.

$$N_\sigma^{C_1} = \frac{Q_{C_1}^* - \sum_{i=1}^{M} \frac{\sum_{j=1}^{N} N_j^i \sigma_j}{l_i}}{\sigma_{C_1} \sum_{i=1}^{M} \frac{1}{l_i}}$$

$$= N_\rho^{C_2} = \frac{l_{min} - \sum_{j=1}^{N} N_j^i \rho_j}{\rho_{C_2}}$$

$$Q_{C_1}^* = (\frac{\sigma_{C_1}}{\rho_{C_2}} \sum_{i=1}^{M} \frac{1}{l_i})(l_{min} - \sum_{j=1}^{N} N_j^i \rho_j)$$

$$\cdot \sum_{i=1}^{i=M} \frac{\sum_{j=1}^{N} N_j^i \sigma_j}{l_i}$$

Case 2: In the case of smaller delay requirement we assume Q is smaller than $Q^*_{C_{n-1}}$ and we select class C_n.

When the delay requirement Q is smaller, we know from equation (2) that N_σ is also smaller. In this case, from equation (1), the number of connections permitted to enter is restricted by the N_σ. From equation (2), we know the larger the σ value, the smaller N_σ will become. In order to have larger N_σ we will select the class with smaller σ value. Since the σ value of class C_n is smallest one among the σ_i values of the classes C_i, $i=1,2,...,n$, the $N_\sigma^{C_n}$ is the largest among $N_\sigma^{C_i}$. Therefore, we select class C_n. When we follow the curve of class C_n in the figure 3, we are able to get a delay requirement $Q^*_{C_{n-1}}$, such that $N_\sigma^{C_{n-1}}$ is equal to N_ρ^{Cn}.

$$N_\sigma^{C_{n-1}} = \frac{Q^*_{C_{n-1}} - \sum_{i=1}^{M} \frac{\sum_{j=1}^{N} N_j^i \sigma_j}{l_i}}{\sigma_{C_{n-1}} \sum_{i=1}^{M} \frac{1}{l_i}}$$

$$= N_\rho^{C_n} = \frac{l_{min} - \sum_{j=1}^{N} N_j^i \rho_j}{\rho_{C_n}}$$

$$Q^*_{C_{n-1}}$$
$$= (\frac{\sigma_{C_{n-1}}}{\rho_{C_n}} \sum_{i=1}^{M} \frac{1}{l_i})(l_{min} - \sum_{j=1}^{N} N_j^i \rho_j)$$
$$+ \sum_{i=1}^{M} \frac{\sum_{j=1}^{N} N_j^i \sigma_j}{l_i}$$

Case 3: For delay requirement Q is between $Q^*_{C_k}$ and $Q^*_{C_{k-1}}$, then we select class C_k.

When we follow the curve of class C_k in figure 3, then we are able to get a delay requirement $Q^*_{C_k}$ such that $N_\sigma^{C_k} = N_\rho^{C_{k+1}}$, i.e.

$$N_\sigma^{C_k} = \frac{Q^*_{C_k} - \sum_{i=1}^{M} \frac{\sum_{j=1}^{N} N_j^i \sigma_j}{l_i}}{\sigma_{C_k} \sum_{i=1}^{M} \frac{1}{l_i}}$$

$$= N_\rho^{C_{k+1}} = \frac{l_{min} - \sum_{j=1}^{N} N_j^i \rho_j}{\rho_{C_{k-1}}}$$

$$Q^*_{C_k} = (\frac{\sigma_{C_k}}{\rho_{C_{k+1}}} \sum_{i=1}^{M} \frac{1}{l_i})(l_{min} - \sum_{j=1}^{N} N_j^i \rho_j)$$
$$+ \sum_{i=1}^{M} \frac{\sum_{j=1}^{N} N_j^i \sigma_j}{l_i}$$

Since

$$\frac{\sigma_{Cn}}{\rho_{Cn-1}} \leq \frac{\sigma_{Cn-1}}{\rho_{Cn-2}} \leq \cdots \leq \frac{\sigma_{C2}}{\rho_{C3}} \leq \frac{\sigma_{C1}}{\rho_{C2}},$$

then $Q^*_{C_{n-1}} \leq Q^*_{C_{n-2}} \leq \cdots \leq Q^*_{C_2} \leq Q^*_{C_1}$, we summarize the relation of class selection and delay requirements as follows:

Case 1: $Q \geq Q^*_{C_1}$ Select class C_1

Case 2: $Q^*_{C_k} \leq Q \leq Q^*_{C_{k-1}}$ Select class C_k, for

                       $k=2,3, ... , n-1$

Case 3: $Q \leq Q^*_{C_{n-1}}$ Select class C_n

3. Numerical Results

In this section, with tandem network, no initial connection, and then evaluate the admissible region with a delay for each new class, and then evaluate admissible region with delay for each new class.

From [5], we use a 10 minutes segment of "star wars" compressed video traces as shown in figure 4. The format, resolution, and frame rate are IBBPBBPBBPBB, 384x288, and 25 frame/sec, respectively. By analyzing the packet stream of the input video, the user matches this σ-ρ curve with the support classes of the network and gets 7 sets of (σ,ρ) classes that can be used for traffic specification as listed below:

The network structure is five hop tandem network with no initial connection, shown in figure 4. The bandwidth between each node was 622.08 Mbits/sec except for l_3, which is 51.84 Mbits/sec. The number of connections of each class (class 1 to 7) is as figure 5 and 6 for OPT policy, and figure 7 and 8 for EQ policy, respectively. We compute all the Q_k^* are listed in table 2. According to the delay requirement, we list the optimal class in table 3. We focus on the admission region analysis between 1ms to 3sec to cover all the above time points. From equation (1) and (2), the relation of the number of connections of each class with delay is shown in figure 5. From figure 5, we see that each curve of a class has two sections, one is a linear increase section and the other is constant section. For example, the linear section of class 1 is from delay 0 sec to 2.5 sec and the constant section is from delay 2.5 sec to 3sec. From figure 5, we will see the class 1 is suitable for large delay requirement, while the class 7 is good for smaller delay requirement. The performance is over 1.16 times for best case to peak rate case and 1.11 times for best case to fix class case in experiment 1 for OPT policy. For EQ policy, performance is over 1.06 times for best case to peak case and 1.12 times for best case to fix class case. From figure 6 and 8, we see that the peak rate allocation was a better performance in low delay requirement.

4. CONCLUSIONS

We proposed a call admission control agent and solved the class selection problem by joining burstiness curve with the call admission function. The method of optimal class selection is to put each class into the call admission formula and get a function value. The larger the function value, the higher the number of connections a class can get.

5. REFERENCES

[1]   Yen-Ping Chu, Ren-Jing Huang, and Kuan-Cheng Lin. "Optimal Schedulable Region Analysis of FCFS Network Model Supports End-to-end Delay Requirement". Proceeding of 1999 National Computer Symposium, Tamkang University, Taipei, Taiwan, pp. B-133-138.

[2]   S. Keshav, "An Engineering Approach to computer Networking," Addison-Wesley, pp.402-406, 1997.

[3] H. Zhang and D. Ferrari, "Rate-controlled service disciplines," Journal of High Speed Networks, vol.3, no.4, PP.389-412, 1994.

[4] Rahul Garg "Characterization of video traffic," Technical report of international computer science institute, 1995.

[5] Oliver Rose. "Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems". Proceedings of the 20th Annual Conference on Local Computer Networks, Minneapolis, MN, 1995, pp. 397-406.
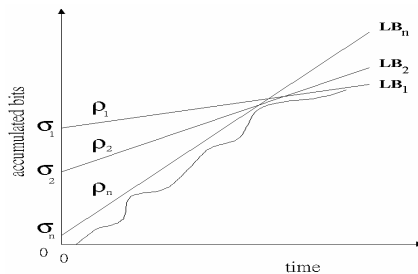
Figure 1 Variant sets of leaky bucket parameters

Figure 2   The Architecture of Switching Node

Figure 3   The points of $Q^*_{C_{n-1}}$ and $Q^*_{C_1}$

Figure 4 The tandem network

Figure 5 The admission region of OPT policy

Figure 6 Admission region of different strategy for OPT

policy

Figure 7 The admission region of EQ policy

Figure 8 Comparison of admission region of EQ policy

Table 1   The classes for input MPEG stream

Class	σ(bits)	ρ(Kbit/sec)
1	1332104	710.
2	899816	740.
3	546400	770.
4	345792	800.
5	216704	830.
6	144792	860.
7	124816	890.

Table 2   The Q_k^* of experiment 1

	Q_1^*	Q_2^*	Q_3^*	Q_4^*	Q_5^*	Q_6^*
OPT	2.40	1.55	0.91	0.55	0.33	0.21

Table 3 The class selection for OPT policy

	Delay Requirement	Optimal class
	Above 2.40sec	1
	1.55sec~2.40sec	2
	0.91sec~1.55sec	3
OPT	0.55sec~0.91sec	4
	0.33sec~0.55sec	5
	0.21sec~0.33sec	6
	Below 0.21sec	7

An Efficient Multipath Routing Protocol for Mobile Ad Hoc Networks

Shih-Lin Wu[†],Shih-Jen Wu[†],Shing-kai Wang[†]

[†]Department of Computer Science and Information Engineering
Chang Gung University, Kwei-Shan Tao-Yuan, Taiwan, R.O.C
E-mail:slwu@mail.cgu.edu.tw; m9021022@stmail.cgu.edu.tw; m9129020@stmail.cgu.edu.tw

Abstract— In mobile ad hoc networks (MANET), one of the most important issues is routing. Many researchers have worked on this issue and proposed solutions from various view points such as short path, energy conservation, load balance, stability, etc. We think that a good path should possess two characters: (i) longer Path Life Time (PLT), and (ii) shorter Path Transmission Time (PTT). A path with longer PLT has the less control overhead to recover route error during transmitting packets and with shorter PTT means the more end-to-end packets can be transmitted within a given time. From the view of conveyed the maximum packets, the path with the largest value of the product of PLT and PTT is the best choice. However, single path may incur a significant cost to wait long time for path recovery. Multipath routing is one of the techniques which have been proposed to alleviate the cost. In addition, multipath routing also provides some benefits: end-to-end throughput, fault tolerance, and load balance. However, a efficient multipath routing strategy is based on a good scheduling strategy to decide which path conveys which packet. We proposed a novel scheduling strategy that evaluates the transmission condition of multiple paths to find out a path that has the minimal transmitting time. Through simulations, we demonstrate the advantage of our new protocol.

I. Introduction

The flourish development of wireless networks has influenced human's daily life profoundly and widely. Thanks to the advancement in wireless communications and lightweight, small-size, and portable/wearable computing devices have made the dream of "communication anytime and anywhere" possible. A *mobile ad hoc network (MANET)* consists of a set of mobile hosts operating without the aid of established infrastructure of centralized administration (e.g. base stations or access points). Communication is done through wireless links among mobile hosts by their antennas. Due to concerns such as radio power limitation and channel utilization, a mobile host may not be able to communicate directly with other hosts in a *single-hop* fashion. In this case, a *multihop* scenario occurs, where the packets sent by the source host must be relayed by several intermediate hosts before reaching the destination host. The advantage of it is rapidly to establish the connection, saving trouble-deploying wires; it is very useful under the condition - the areas without wires, such as war, salvage, or the based station sabotaged by natural disasters. The disadvantage of it is that each mobile host moves freely making the network topology changeable and unpredictable. Moreover, due to the absence of fixed network topology and central control, to design a suitable routing protocol is one of the most challengeable works in MANETs.

Many routing protocols [12], [14], [15] have been developed in MANETs and can be categorized as two parts: proactive and reactive. Proactive routing protocols, for instance DSDV [11], take advantage of broadcast to collect mobile host and routing protocol information of mobile host, and always need to be updated. Their merit is that when source node transmits data packet, a path to transmit can be expediently found. The drawback is that in order to gain path information more efficiently, it must be more frequently updating routing table - when employed in a high mobility of the Internet, which will generate over-demanding overheads. On the contrary, reactive routing protocols, such as AODV [10], DSR [7], only establish a path only when a source node has to-be-transmitted data packets to its destination. The source node seeks paths while transmitting data packet, though it can immediately transmit it; however, compared with routing protocol of proactive, it can save the network source that pays for regularly updating information of path, and less consumption of bandwidth as well.

II. Relative Work

Most of traditional routing protocols only choose a single path to transmit data packets. However, due to the topology of MANETs changeable apt to break links frequently, we have to look for another path to transmit the data packets again when link breakage. It will waste a lot of time in looking for the path again in the single routing protocol, causing the transmission of data packet prolonging too long. Therefore, TCP will cost time out, and the efficiency will reduce rapidly. In order to reduce the time to search another path again, many methods of promoting its efficiency is proposed successively as considering with stability of paths. One of these methods [6] considers the link stability which is defined as the times of received signal greater than a threshold and then use the more stable links to construct a path.

Another method use the mobility prediction scheme stated in [13]. When a mobile host receives a data packet from previous mobile host, it will know the speed, position, direction of previous mobile host by the information puts in the data packet. And utilizing the global positioning system (GPS) to know the absolute coordinate position of the mobile host in the space of path, and to use the end time of link as the route selection. However, using GPS to get the information of other mobile host still have a lot of shortcomings as list below. First, Although it can obtain the information of other mobile hosts exactly when install GPS, in a position also increased the calculation of each mobile host. Second, Installed GPS in every mobile host are extra cost. Last, GPS is only suitable

for using outdoors, whenever the user loses the function once getting to the room.

Multipath transmission is an efficient method to take care the reroute problem. This method can be divided into two ways. One way is that only pick one path used as primary route and the others as backup routes. When the main route can't be used, use backup rout to continue transmit the data packet, such as AODV-BR [1], [3]. But used this way will have following problems–when breakage in the main route, because doesn't using the backup path to transmit packet until main path broken, and so on, we don't the path is still useful, and do not guarantee the backup route can be used certainly to aid the transmission , so waste more control overhead and extension packet and transmit time instead. The other way is mainly using the round robin manner letting the selected paths take turns to transmit data packets. Transmitted the data via different paths, not only can avoid the network congestion, but use resources of the network effectively. Meanwhile, it also can observe the situation of each path, such as ODMRP [5]. But using every path blindly will make the end-to-end transmission delay too long.

III. Design Multiple Path Routing Protocol with parallel transmission capability

There are two types of Multipath routing protocols: one is node-disjoint paths, and the other is link-disjoint paths. In node-disjoint paths, any two paths don't have any node in common, expect their source and destination nodes while in link-disjoint paths, any two paths don't have the same link in their paths. Note that link-disjoint paths may have common nodes. Of course, the condition of node-disjoint actually guarantees that links fail independently, However the condition of node-disjoint is a much stricter than link-disjoint and thus presents a much less number of disjoint paths or may select the longer hop counts as well as delay paths. This makes node-disjoint less effective. But Source node can use the higher other spatial to parallel to transmission packets efficiency. However, in the varied MANETs, every mobile host have a ability to forward data, then We believe that not only considered node disjoint may achieve the better transmission efficiency. But we thought the above way as the transmission strategy all have its flaws, so we adopt the way takes a main basis by way PTT, dynamic to use Multiple paths strives for the effectiveness, adopts the above movement to be able to have the advantages, Because is using PTT to chooses path, therefore we certainly can use shortest delay path, But after the algorithm estimate, we possibly can discard other longer delay paths to carry on the transmission, perhaps only use it once for a very long time, like this we may avoid using the path with heaver traffic load or a longer hop counts' path to transmit packets. Also may the effective use different paths, and prompt realized whether above each path topology does have the change, whether continues to use. To avoid when the path expired, but Source node still wants to continue using that path and leading packet lose.
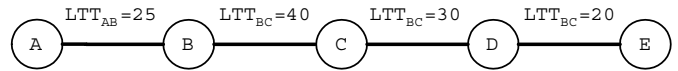

Fig. 1.   Path A-B-C-D-E and LTT

Moreover when Destination node discovery to nose out the packet's PTT has fierce change, Should needs to produce information and feedback to Source node, That's because the Ad hoc wireless networks network topology and the current capacity often changes, so Source node must be able to know immediately and renewal PTT in every paths, that we can avoid Scheduling policy to occur error results.

TABLE I
DEFINITION

Symbol	Description
Time[Data]	The time when source received a Data
PLT	Path Life Time
PTT	Path Transmission Time
LTT	Link Life Time
FLT	First Link Transmission Time
MLT	Max Link Transmission Time
Path[m]	Source has number of m paths
Path(Data[n])	The Path which source uses it to transmit data n
DTCP	The Difference Time for two consecutive packets arrive at destination

IV. What's LTT FLT MLT and PTT

LTT is a time interval that one mobile host receives a packet and successful to transmit the packet to others mobile hosts, LTT involve the MAC layer contention's time wastage plus network layer buffer queuing delay time final and plus transmission delay. Because in a Ad hoc wireless networks mobile host receives packet, if in buffer also has other packets not yet to transmit, then this packet must wait till after it the early come packets is all transmitted, and waited for the time which transmits is network layer buffer queuing delay; Each mobile host transmits packet must content for the channel firstly , which the time is called MAC layer delay; Transmission Delay is refers by the different transmitted rate the time which packet needs; Then we can measured LTT by the way, Summarizes as follows, because in Ad hoc networks is a non- synchronized environment, we cannot the supposition each Mobile host's clock all be synchronization. When each mobile host produces or receives forward packet, immediately covers a time stamp, when turns to this packet transmission, again covers a time stamp, gets the time difference is Network Layer Buffer Queuing delay adds and MAC level Delay, therefore when receiver receives packet, in according to packet size as well as the transmission speed figures out Transmission Delay, receives the end then to calculate it with to deliver the end LTT.

In Fig. 1, node A is source, node E is Destination. Source A has a path A-B-C-D-E to destination E. Each link has it's own LTT, ex. LTTAB=25: The time interval that Source node send a
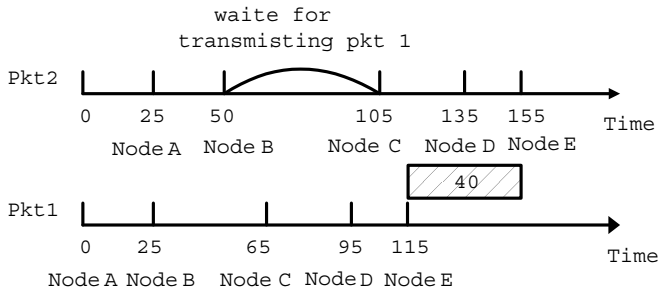
waite for
transmiting pkt 1

Pkt2

| | | | | | | | |
0   25   50   105   135 155   Time

Node A   Node B   Node C   Node D Node E

40

Pkt1

0   25   65   95   115   Time

Node A Node B   Node C Node D Node E

Fig. 2. an example for the effect of MLT

RREQ or data packet to his neighbor, and so on, if we measure the LTTBC =40BLTTCD =30BLTTDE =20. When destination E receiver this packet, that we can get the MLT that largest of the path's LTT from those LTT. PTT (path transmission time) is the sum of path' total LLT, so we get PTT = LTTAB + LTTBC + LTTCD + LTTCD = 25 + 40 + 30 + 20 = 115.

Found from our research observations, the time interval of two packets to reach destination will be equal to this path of MLT - if using the same path to consecutively transmit two packets. Exploring the cause, it is found that because MLT is the bottleneck of path, in the course of transmitting the two packets, the second packet will be delayed in the area of MLT. Only under the situation that the first packet has finished the transmission will the second packet start. And, that's why the time interval of reaching destination will be equivalent to MLT. As Fig. 2 demonstrates, the numbers below time axis represent the time Packet arriving Mobile Host. As a result of this cause, we discover that among multi-paths, it doesn't mean that all packets using the minimal path of PTT to transmit results in less time needed for transmission.

The reason why FLT will influence the time of whole transmission is that Source node fails to constantly transmit packets. Besides, the surrounding objects of Source node may possibly be affected by objects' own surroundings to further disturb the area of LTT, causing each Path of FLT various. Source node, moreover, needs to help transmit Packets as well.

Hence, we propose a mathematical calculation, consisting of PTT and Multiple Paths. Prior to transmitting packets, going through mathematical calculation first so as to arrange the best suitable path to transmit packet out of multi-paths, it can not only gradually cut down the time of transmission, but also highly upgrade the efficiency of Internet bandwidth.

V. METHODS OF FINDING PATHS

Source node pursues multipath by using similar AODV or DSR in the mode of reactive to get the path. In fig.3 each mobile host of networks has one routing table recording the information of paths. The establishment of path can be divided into two steps simply put as stated below:

1   Route discovery: by the time source node A needs the path to transmit packet, the first move is to precede route discovery, discovering a path of RREQ from destination D. The intermediate node is the

first time to receive RREQ, founding reverse path in source node at first, and then RREQ flooding. Before flooding, it needs to attach itself, the LLT (Link Life Time) of last Mobile host, and LTT (Link Transmission Time) to this RREQ. If repeatedly receiving this RREQ, it would be abandoned and stop forwarding. When the last destination receives this RREQ, according to reverse path, it may reply a RREP to source node in a time interval. In the wake of the intermediate node receiving RREP, the reverse path also requires to be established to the destination. In other words, while source node receives a RREP, a path to destination is founded as well. Source node receives RREP out of multi-paths, depicting individual path of PTT, RTT and MLT.

2   Route maintenance: Once a intermediate node of the path detects the disconnection of link, the route error (RERR) packet immediately generated to inform the source node of disconnection. Because of multipath, right after source node receives RERR, the source node won't restart flooding RREQ to find other paths right away. It is not until all of paths disconnected that source node will once again have flooding RREQ to search paths.

VI. METHODS OF TRANSMITTING DATA PACKET

After the source node receives different paths of RREP from destination node, the obtaining information of PTT and MLT establish DTCP table. And, the following transmission of packets should according to the mathematical calculation designed by us find out the most suitable path to transmit packets. Providing that the destination node detects that the information of LTT attached to packet large enough to change the sequence of packet's assembly schedule arranged by original DTCP table, it will need to reply a Control Message, the mobile updating Source node to the information of networks environment.

In figure 3, Mobile host A is source node, and Mobile host D is destination; source node has three paths to transmit packets. At the Route Discovery Time, we should detect each path of PLT via signals; Path [1], Path [2], and Path [3] of PLT are 1500, 2000, and 1700 respectively. Before transmitting Packet each time, we need to assess if remnant PLT of the path is larger than PTT. Only under such circumstance can we select this path to undertake transmission. If the source node has received Control Message, established in DTCP table, replied by each path from destination, the first packet select the path with the smallest PTT or Path [3], to transmit. Figure 3, there are three paths, path [1] = A, B, C, G, D, Path [2] = A, E, F, G, D, Path [3] = A, H, I, J, D. From table 2 (DTCP table), we can learn that if source node A orderly transmit two packets, destination D receives the possible time interval from this two packets.
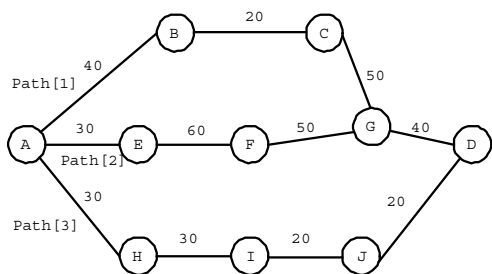
Fig. 3. source node A has three paths to transmit packets to destination D. The numbers below the line stand for PLT, and the numbers above the line represent LTT of a link.

Current path	PTT	FLT	MLT	Next path	DTCP
1	150	40	50	1	50
				2	70
				3	-10
2	180	30	60	1	0
				2	60
				3	-50
3	100	30	30	1	80
				2	110
				3	30

TABLE II

THE DATA STRUCTURE OF DTCP TABLE.

VII. THE MATHEMATICAL CALCULATION OF DTCP

Case 1. If Packet N takes advantage of Path [n] to precede transmission and Packet N + 1 intends to use Path [n] to transmit, DTCP(Path[n],Path[n]) = MLT(Path[n]) EX:DTCP(Path[1],Path[1])=50 Case 2. If Packet N takes advantage of Path [n] to precede transmission and Packet N + 1 intends to use other Path [m] to transmit, DTCP(Path[n],Path[m]) = PTT(Path[m]) - PTT(Path[n]) - FLT(Path[n]) EX:DTCP(Path[1],Path[2])=180-(150-40)=70

But, we discover that if only using mentioned-above two regular rules to engage in Assemble Schedule, it fails to achieve the best using condition. The situation details as stated below: If according to Table 2, the first Packet chooses the minimal PTT to transmit, using Path [3], and then the second Packet according to DTCP table, selecting Path [3] as well, and problems will thus occur. As a result of Packet 1's failure to reach destination node, if attempting to use Path [3] to transmit, the DTCP table of Path [3] equivalent to the field of path [3], we need to add up MLT (30) of Path [3], leading to reasonable estimating figures. We thus adopt a new rule.

Case 2-1. If Packet N uses Path [n] to undertake the transmission, and Packet N +1 intends to use other Path [m] to transmit, and still having previous packets of Packet N +1 are transmitting which haven't reached Destination node, we should add up desirable MLT of this path first, after which the paths from DTCP table should be selected.
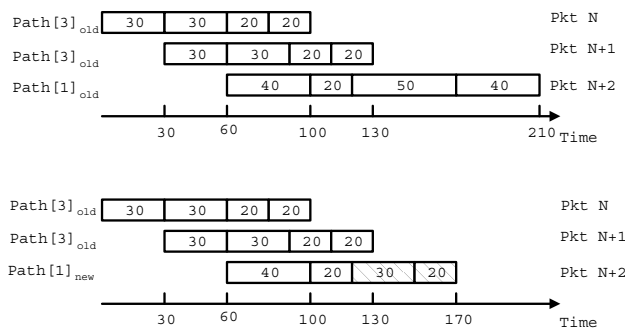

Fig. 4. In the passage of 500 time units, the figure of showing LTT situation changes.

DTCP(Path[n],Path[m]) = PTT(Path[n]) - PTT(Path[m]) - FLT(Path[m])+MLT(Path[m]) EX: Under situation of transmitting packets above of Path [3] DTCP(Path[3],Path[3])= 100-100-30+30 = 60

However, if the re-calculated figures are still the smallest (60) in DTCP, Path [3] still has to be used. But, if the following transmitting Packet 3 detects that Packet 1 and Packet 2 are still engaging in transmission by searching DTCP table, DTCP (Path [3]) will need to add up the last MLT of Path [3] that should be 60+30=90. At this moment, we should select Path [1] to precede transmission (because the value of DTCP (Path [3], Path [1]) is the smallest.) Therefore, as long as we follow the rules above, orderly selecting appropriate Paths to transmit packets, each Packet may orderly reach destination node and makes the delay of End-to-End minimal.

VIII. METHODS OF UPDATING PATH INFORMATION

The network topology of MANETs is always changing.When Destination node receive measures of dispersion of LTT of any path of transmitting packets adequate enough to influence the outcome of mathematical calculation,Fig. 4 replying a Control Message containing of information of dispersion to inform source node is necessary. By doing so, source node can use after-changed FLT, MLT and PTT to accurately arrange the path of transmitting packets. As Fig. 4 demonstrated, if Destination detects the LTT above Path [1] has changed and thus restart calculation DTCP Table III, Destination node discover that the sequence of transmitting packets have changed from Path [3], Path [3], Path [1] to Path [3], Path [1], Path [3], replying a Control message to inform Source node.

IX. SIMULATION

The model in our simulation is a MANET in a square area of 100m100m with 70 mobile hosts in it. The propagation range of each mobile host is 150. Mobile hosts in the network are set randomly with the traffic rate of 50 packets/sec. The mobility model here is random waypoint. The movement patterns are generated for 5 different speeds: 0, 5, 10, 15, 20 seconds. There are 4 different traffics follows environment in my simulation model. Our simulations involve three kind of routing protocol,

Current path	PTT	FLT	MLT	Next path	DTCP
1	110	40	40	1	40
				2	110
				3	30
2	180	30	60	1	-40
				2	60
				3	-50
3	100	30	30	1	40
				2	110
				3	30

TABLE III

DCTP TABLE OF RE-CALCULATION IN DESTINATION NODE

Fig. 6.   Packet arrival rate V.S. Mobile Speed

Fig. 7.   Packet average End-to-End delay V.S. connection pair
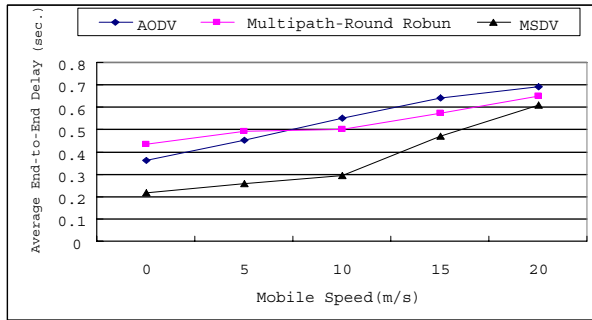

Fig. 5.   Packet average End-to-End delay V.S. Mobile Speed

the first two respectively is AODV and Multipath - Round Robin routing protocol, last routing protocol which is we proposed, hereafter refers to as Multipath - Schedule. The affiliation by above three kind of routing protocol simulation results, further discusses our routing protocol and AODV, the Multipath - Round Robin routing protocol performance difference; Under is comparison performance metrics.

- Average packets End-to-End delay
- Total arrival packets

We use the different mobility speeds on simulation, surveys under the different mobility speeds, networks topology change regarding network whole packets arrival rate and Average End-to-End Delay change. The affiliation discusses our routing protocol from this the experiment in the different network environment time the serviceability.

First we by Mobile Host mobility speed took a variable observes average packet End-to-End delay to have what change. Like the Fig. 5 shows, we discovered network in a environment change not high situation, Multipath-Round Robin's average delay is highest, this is because under stable environment, Round-Robin possibly can choose a delay longer path, but AODV all is chooses shortest delay path to transmit packet. But if all uses identical strip path in the transmission time, actually can create buffer delay lengthening. But works as when the network environmental variation is quicker, backup
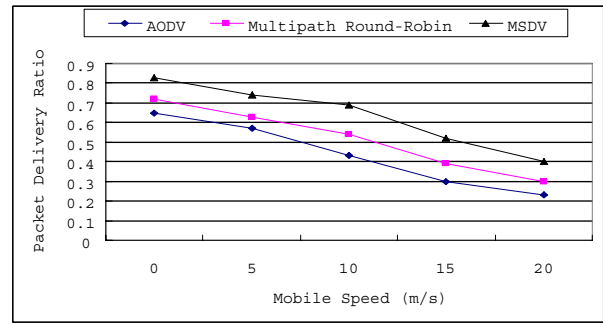
path only then can appear the benefit. But we proposed developed algorithm simultaneously to give dual attention to backup path with to consider path's delay two kinds of factors. Therefore we can obtain better average delay under the mobility speed quite low environment. But when mobility speed arrives under each second 20 meter environment, because environment changes too fast, causes the path which may select to transmit is extremely scarce, causes to three kind of routing protocol's average delay quite to approach.

Fig. 6 is regulates mobile speed to observe the change of packet arrival rate, we may discover, in mobility speed under low environment, if only pure to use multipath, the performance is not certainly high, but in the situation which mobility speed slowly increases, Multipath-Round Robin and our routing protocol can have the obvious disparity production, to show our routing protocol may more effective use Multipath carry on a row of regulation, therefore our performance surpasses Multipath-Round Robin, achieves better performance.

Experiment 2: We first vary the number of sessions form connection follows. We keep the max. speeds constant at 10 m/s in this set of experiments. To observe our routing protocol whether can have the suitable accent to achieve the better performance utilization.

Mobile Host mobility speed is 10 (m/s) by the random way point moving, the accent changes the current capacity to observe. May see by Fig. 7, when the network environment is not busier, we don't use Multipath blindly , our routing protocol may use other comparatively not busy Mobile Host

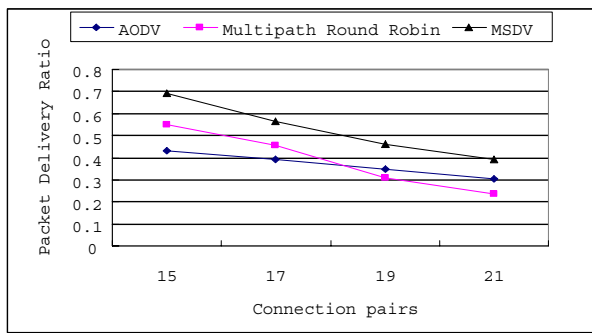Fig. 8.   Packet average End-to-End delay V.S. connection pair

to transmit packets, moreover makes an effective platoon regulation to packet the movement, causes us End-to-End delay far to be smaller than Multipath-Round Robin. But AODV is all transmits all packet by way of identical strip Shortest delay Path, therefore works as when current capacity change, its delay change certainly cannot have reveals the change, however Multipath-Round Robin actually wants simultaneously to use all paths for the parallel transmission, when network is in lighter loading, End-to-End delay truly has compared to AODV comes well, when network loading changes busily, this method actually meets extra creates the clogged up phenomenon, increases End-to-End delay. Therefore because we inducted PTT as well as the Multipath two factors, the dynamic measured the network the change, the suitable accent changes the transmission route, may reduce average End-to-End delay.

By Fig. 8, we can observe our routing protocol may use other mobile host to transmits packets, in addition the original backup path mechanism, causes our routing protocol to be allowed to have the better potency performance, but is more and more heavy when network loading, our routing protocol can have the possibility to degenerate with single path routing protocol. This is because our routing protocol can act according to each path delay to regulate the transmission path. But discovered by the Multipath-Round Robin curve, when traffic load is heavier, uses other path to transmit, instead can have the worse potency performance.

X. CONCLUSION

In this paper, we have presented the new multipath routing protocol that is based on the idea of path selection criteria, such as PLT and PTT, to choose the better paths and the new scheduling strategy to arrange to-be-transmitted packets to suitable paths for transmission. Our main idea of scheduling strategy is to find out a path that has the minimal transmitting time among those available paths. Through simulations, we demonstrate the advantage of our new protocol.

XI. ACKNOWLEDGEMENTS

REFERENCES

[1] Song Guo and Yang, O.W, "Performance of backup source routing in mobile Ad Hoc wireless networks," *in Proc. IEEE WCNC'02,*

[2] Z. Hass and M. Pearlman, "The performance of Query Control Schemes for the Zone Routing Protocol," *ACM/IEEE Trans.*Net., vol. 9, no. 4, Aug. 2001, pp. 427-38

[3] S. J. Lee and M. Gerla, "AODV-BR: Backup Routing in Ad Hoc wireless networks," *Proceedings of IEEE WCNC 2000,*Chicago, IL, Sept. 2000.

[4] Lee, S.-J. and Gerla, M., "Split multipath routing with maximally disjoint paths in ad hoc networks," *Communications, ICC 2001. IEEE International Conference on*, Volume: 10 Page(s): 3201 -3205 vol.10

[5] Nasipuri, A.and Das, S.R., "On-Demand multipath Routing for Mobile Ad Hoc wireless networks," *Proceedings of the 8 th Annual IEEE Internation Conference on Computer Communications and networks (ICCCN),* Boston, MA, October 1999, pp. 64-70.

[6] N. Imai, T. Nakagawa, H. Morikawa and T. Aoyama, "Stability Adaptive Routing Algorithm in High-Density Networks," October 22-28, iecom 2000, The Nagoya Congress Center, Nagoya, JAPAN

[7] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad hoc Wireless Networks," *Mobile Computing,* ,edited by T. Imielinski and H. Korth, Chapter 5, Kluwer Publishing Company, 1996, pp. 153-181.

[8] E. Papapetrou, and F.-N. Pavlidou, "Disjoint routes for on demand routing protocols in ad hoc networks" *Global Telecommunications Conference,* 2001. Page(s): 2804 -2808 vol.5

[9] Paul K., Bandyopadhyay S., Mukherjee A.and Saha D., "Communication-aware mobile hosts in ad-hoc wireless network," *Personal Wireless Communication, IEEE International Conference on,*Wednesday, February 17, 1999, Orlando, FL USA. Page(s): 83 -87

[10] Perkins, C.E and Royer, E.M "Ad hoc on demand distance vector (aodv) routing," *Proceedings of MobiCom '99,* ,Seattle, WA, August 1999, pp. 207-218.

[11] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *In Proc. of the ACM SIGCOMM,* ,October 1994, London. vol. 24, no. 4, pp. 234-244.

[12] E.M. Royer and C-K Toh, "A Review of Current Routing Protocols for Ad-hoc mobile wireless networks," *IEEE Personal Communications,* , CA. April 1999

[13] S. H. Shah, and K. Nahrstedt, "Predictive Location-Based QoS Routing in Mobile Ad Hoc wireless networks," *In Proceedings of IEEE International Conference on Communications (ICC 2002),*New YorkANYAApril 2002.

[14] Y.-C. Tseng, S.-L. Wu, W.-H. Liao, and C.-M. Chao, "Location Awareness in Ad Hoc Wireless Mobile Networks", IEEE Computer, June, 2001, pp. 46-52.

[15] Y.-C. Tseng, W.-H. Liao, and S.-L. Wu, "Mobile Ad Hoc Networks and Routing Protocols" (published as a book chapter in Handbook of Wireless Networks and Mobile Computing, John Wiley & Sons, edited by I. Stojmenovic, publication date: 2003). Chaper 17, Page(s):371-392.

[16] Lei Wang, Yantai Shu, Miao Dong, Lianfang Zhang and Yang, O.W.W., "Adaptive multipath source routing in ad hoc networks," *CommunicationsA 2001. ICC 2001. IEEE International Conference on* A Volume: 3 A 2001 Page(s): 867 -871 vol.3

[17] Ming-Hong Jiang and Rong-Hong Jan, "An efficient multiple paths routing protocol for ad hoc network,"*Information NetworkingA 2001. Proceedings,* 15th International Conference on A 2001 Page(s): 544 - 549

A Mobile Recommender via Location Agent

Chuan-Feng Chiu, *Jason C. Hung, and *Yi-Ping Huang
Multimedia and Information Network Lab (MINE Lab)
Department of Computer Science and Information Engineering
Tamkang University
Email : cfchiu@cs.tku.edu.tw

*Department of Information Management
Kuang Wu Institute of Technology
Peitou, Taipei, Taiwan 112, R.O.C.
E-mail: jhung@cs.tku.edu.tw

***Abstrac**t*
Internet has become a popular platform for exchange information. No spatial is the major advantages of the architecture. And several network environment, GSM/GPRS or wireless network have been widely used. Personalized information recommendation is a useful application area for heterogeneous environment. Through the recommendation mechanism users can obtain the better choice to maximum their needs and the merchants can obtain more profit also. In the paper we develop a recommendation mechanism that use SOM algorithm and Perarson Correlation as the user distance measure to find the longest common sequence that shows the potential and useful suggestion list. We believed that the recommender mechanism would personalize the user's request and drop un-related information.

Keywords : Internet, ecommerce, Location Recommender, Agent

1. Introduction

With the advent of Internet computing environment, several people can communicate with each other in different locations and would not know each other. Based on the reason the different communication mechanism would appear and changed our daily life. People would travel to any places even the place infrastructure is different. So a personalized agent would be useful for persons who would be roamed in different places. On the other hand, when the user arrived at some where, he/she might not understand the products or service in the stranger locations. The user would ask his personal agent to produce request for requesting useful information. The agent could process and accept the recommendation or

suggestion based on the tastes of the user. Therefore personalized recommendation would be applied to users based on the location information and user tastes.

In this paper we address the location recommender based on the cluster approach and longest common sequence as the recommendation mechanism in the electronic commerce environment. And through a personalized assistant user could get useful information, product suggestion based on the user's tastes. We believed that the personalized information application would be the trend in the Internet environment, especially for the ecommerce application area.

2. Related Work

In this section we describe some researcher have done in the recommendation research area. Paul Resnis K and Hal R. Varian propose a recommendation definition and direction in 1997 [2]. The idea is that users cannot make a choice without sufficient personal experience or other people's suggestion. Thus, automated recommendation by the computer system would be very helpful in filtering product information. Markom Balabanovic and Yoav Shoham proposed a recommendation system --- Fab in 1997 [1]. They combined content-based filtering and collaborative filtering approaches to design their system. Content-based recommendation recommends product items to users who have purchase records or showed interest from browsing on the web. On the other hand, collaborative recommendation classifies users into several groups. Users in the same group are of similar behavior or interest on product items. These researchers only discuss the recommendation mechanism on one site or system. In the paper we

address the recommendation system in the community and proposed the community recommendation mechanism.

On the other hand, intelligent software agent is a popular technology for solving problems. The software agent can be defined that the agent is a computing component to perform some tasks according to the knowledge. The attributes that include autonomy, proactive, and reactive are the advantage of the software agent and these are the attributes of agent based negotiation system also. Several researchers have discussed the agent technology in the past. In [6] the authors presented the concept of the agent-based software architecture. The agent communication language and the agent architecture are two important issues. In order to achieve information and message sharing, the agent communication language is the basic mechanism. In order to build the agent computing environment, the agent architecture is another critical problem.

BASAR is a personalized agent system that keeps the web links based on the user bookmarks [7]. The system is able to support information updating and reduce the number of links by deleting less used items. In [8], a web-based information browse agent is proposed. The system uses the KQML as the agent communication language and reduces networking load. And in order to reduce the complexity of browsing, they use the structure meta-information mechanism.

With the advent of the Internet computing architecture, the mobility is the new ability of the agent. Hence the agent with the mobile capability is appeared and become the new research topic. The mobile agent technology overview can be found in [9,10,11]. Aglets[13] is a mobile agent platform based on the Java technology. The system uses Agent Transfer Protocol as the agent communication infrastructure and the architecture is able to support persistence, security, and agent collaboration. Voyager [14] and Concordia[15] are another two mobile agent systems that also support the agent communication and agent computing environment. The mobile agent platform, MAGNA, and its architecture are proposed in [12].

3. Recommender Mechanism

In this section, the design of recommender mechanism would be proposed. In the recommender system, the basic idea is to discover similar group or tastes from the amount of usage data or information. User voting would be the basic information for discover user taste similarity. At first, user would be asked to fill the voting options as the user's favorites. Based on the voting value, in this paper a recommerder mechanism is proposed to analyze the voting raw data that revealed the user's tastes.

In the proposed mechanism, the basic idea is to divide users into several groups and the users in the same group would have the similar interest regarding the user's voting. Self-Organization Map[16, 17, 18] is a method to cluster data to several groups. Expect for the cluster method, Self-Organization Map, the Pearson correlation would be used as the measurement of similarity between users.

In the following, the user classification algorithm is described for discover similar users and cluster groups.
Step 1 : Choose weight vector W
Step 2 : Select x from input vector V
Step 3 : Calculate the user similarity by the Pearson Correlation equation
Step 4 : Find the minimun distance of input as the winner. *winner = arg min corr(x, w$_i$)*
Step 5 : Update weight vector
Step 6 : Repeat step 2 until the group would not change
Step 7 : Output groups

From the output of the result of group, the longest common sequence would be found in each group as the common similar tastes of users in the group. So the potential products or service could be recommended to users according to the recommendation list of the longest common suggestion sequence.

4. Location Recommernder Agent

In the previous sections the recommender algorithm have been presented. In this section a Location Recommender Agent is proposed. The Location Recommender Agent would compute the recommendation list and the agent also accept the user's request that input by users. Figure 1 shows the software architecture of Location Recommender Agent. Basically, the agent contains three basic layers including Agent User Interface, Agent Computing Logic and Agent Communication Interface.

Agent User Interface : The interface would be the middleware between user and the agent entity. User would input the request to the interface and the interface would send the request to the Agent Computing Logic to produce the correct actions. And the response would be output to user from the Agent User Interface.

Agent Computing Logic : The middleware would translate the user request and process the request. And the Computing Logic layer will generate recommendation sequence based on the location information of the user and tastes from user voting value.

Agent Communication Interface : The layer will be responsible for the communication capability to communicate with other Location Reommender Agent and the central raw database that including location information and the user voting value. On the other hand, there exists several agent communication languages for the agents and we decide the KQML as the agent communication language. The agent communication protocol is based on the TCP/IP network transmission mechanism. Through the general agent communication language, the unify communication interface will be between agent to agent or agent to central database.

Figure 1 : Location Recommender Agent

And the Location Recommender Agent would be an assistant of user. The user would be on the web site or roaming between mobile network (GSM/GPRS etc.) or wireless network. The agent would act the role to accept request from user and response or direct recommendation from server. The design of such assistant would be useful for users who would travel to somewhere and roaming between different network infrastructures. Figure 2 shows the relationship between agent, user and different network infrastructure.

Figure 2 : Relationship of Location Recommender Agent

6. Implementation

The implementation system is based on the Aglet API. The aglet is the pure java mobile agent platform and defines the basic functionalities of the mobile agents. Figure 3 showed the Aglet system architecture that include aglet runtime layer and communication layer. The aglet runtime layer implement the Aglet interfaces and communication layer address the communication mechanism between the mobile agents. According to the aglet architecture we extend these basic functionalities to implement the community recommendation system. The site in the community would record the other sites information and the agent status that has migrated to other site. Therefore, the site is not only the execution places for the mobile agent but also a client to receive the recommendation from others in the community.

Figure 3 : Aglet System Architecture

7. Conclusion

We take the advantages of agent technology and develop a recommender agent based on the location to recommend interests to users. We believed that the personalized recommendation would be useful application in the ecommerce environment. In this paper the recommender mechanism that combined SOM and Pearson Correlation approach is proposed. In the future we will add the security protection mechanism to the architecture to avoid the mobile agents or messages hacked by untrust units because unexpected data would not be allowed for the personalized agent.

8. Reference

[1]. Marko Balabanovic and Yoav shoham. "Fab : Content-based collaborative recommendation." Communication of the ACM, 40(3), p64-72, 1997.
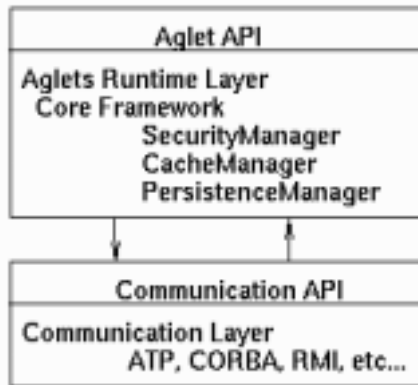
[2]. Paul Resnick, and Hal R. Varian. "Recommendation systems." Communication of the ACM, 40(3), p56-58, 1997.

[3]. Peter R. Wurman, Michael P. Wellman, William E. Walsh, "The Michigan Internet AuctionBot: a configurable auction server for human and software agents", in Proceedings of the second international conference on Autonomous agents, p301– 308, 1998.

[4]. P. Maes, R. Guttman, and A. Moukas. "Agents that Buy and Sell: Transforming Commerce as We Know It." Communications of the ACM, Vol 42, No 3, 1999, p81-91.

[5]. Anthony Chavez, Pattie Maes, "Kasbah: An Agent Marketplace for Buying and Selling Goods", in Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, London, UK, April 1996.

[6]. Michael R. Genesereth, "Software Agents." Communication of the ACM, Vol. 37, No. 7, July, p48-54, 1994.

[7]. Christoph G. Thomas and Gerhard Fischer, "Using agents to Personalize the Web." In Proceedings of the 1997 ACM IUI conference(IUI'97), Orlando Florida, U.S.A., pp53-60, 1997.

[8]. Chanda Dharap and Martin Freeman, "Information agents for Automated Browsing," in Proceedings of the 1996 ACM CIKM conference(CIKM'96), Rockville, MD, U.S.A., pp296-305, 1996.

[9]. J. Kiniry, D. Zimmerman, "A hand-on look at Java mobile agents." IEEE Internet Computing. 1 (4), 1997.

[10]. V.A. Pham, a. Karmouch, "Mobile software agents : an overview." IEEE Communication Manazine, july, 1998.

[11]. H.S. Nwana, "Software agents : an overview." Knowledge Engineering Review, 11(3), 1996.

[12]. Sven Krause and T. Magedanz, "Mobile service agents enabling intelligence on demand in telecommunications." In Proceedings of the 1996 IEEE Global Telecommunications Conference, London, UK, p78-84, 1996.

[13]. http://www.trl.ibm.com/aglets/

[14].http://www.objectspace.com/products/voyager

[15]. http://www.merl.com/projects/concordia/

[16].T. Kohonen, Self-organizing Maps, Second Edition, Springer-Verlag, 1997.

[17]. S. Kaski, K. Lagus, T. Honkela, and T. Kohonen, "Statistical aspects of the WEBSOM System in Organizing Document Collections", Computing Science and Statistics, 29, 1998, 281-290.

[18]. T. Kohonen, "Self-organization of very large document collections: State of the art", Proc. of the 8th Int. Conf. On Artificial Neural Networks, 1, Springer, 1998, 65-74.

A Mobile Learning Platform for Supporting Outclass English Learning Activities

Chih-Yung Chang[+], Kuei-Ping Shih[+], Gwo-Jong Yu[*],Shih-Chieh Lee[+], Hsu-Ruey Chang[+]

[+]Department of Computer Science and Information Engineering, Tamkang University

cychang@cs.tku.edu.tw

[*] Department of Computer and Information Science, Aletheia University

yugi@wireless.mcs.au.edu.tw

Abstract—**This paper illustrates the design and implementation of outdoor learning project. The research in the paper exploits the planning of outclass learning activities in high school curriculum, the implementation of a facilitating and e-mobile-enabled software platform on portable computers in outdoor activity and the evaluation of the outclass learning's influences on students' learning performance. The goal of outdoor learning activities is to integrate learning technology and diversity into school curriculum. The designed mobile learning platform assists learners in learning with useful functions such as workflows building, workflows commentary and retrieving of the learning sheets through wireless network. Thus, construction and implementation of learning activities become easier, and students gain more opportunity of interaction from learning activities through the functions as chatting in wireless manner and data exchange provided by the platform. The activity record-keeping functions through video-taping and voice recording on the mobile learning platform increases students' spontaneity in information gathering as well as motives for spontaneous learning. Participating in the outdoor activity and study in an unburdened, freely, and lively ambience, students can have better learning performance as well as broaden scope of learning.**

I. INTRODUCTION

This Mobile learning refers to "learning through mobile computational devices" [1]. Sariola mentioned the advantages of mobile learning, "The mobile environment integrates studies that take place on campus, at home or outside university facilities into shared and flexible learning environment"[2]. Mobile learning shows more flexible and efficient than inclass learning and e-learning owing to time-boundless and space-boundless learning space. Kynaslahti [3] mentioned that mobility, from educational viewpoint, is composed of three elements: convenience, expediency, and immediacy. Chang *et al.* [4] mentioned that mobile learning contains three essential elements: mobile equipments, communications technology, and user interfaces. The three essential elements are indispensable in the process of outdoor activity.

Y. S. Chen *et al.* [5] propose bird watching learning system using scaffolding techniques. The system constructs an ad hoc bird watching learning classroom using handhelds and laptop among instructor and learners. In the process of activity, learners can use their own PDA search bird knowledge in the database using data mining technique. Depending on learners' performance in the activity, the system offers each learner different level of support. M. Okada *et al.* [6] propose environmental learning system. The system builds a networked environmental learning in real and virtual worlds. The aims of the activity are to raise participants' interests in environment and awareness to nature.

The aforementioned activities all apply the mobile learning in outdoor learning and show their positive advantages in mobile learning. This paper describes the design and implementation of a mobile outclass learning technology project. The project contains the planning of outclass learning activities in high school curriculum, the implementation of e-mobile-enabled software platform on portable computers in outdoor activity and the evaluation of the outclass learning's influences on students' learning performance. The outdoor learning activities focus attentions on learners and technology-support learning, giving possibilities to experience-based and team-based collaborative learning. Three learning activities "Making an Interview with Foreigner", "Eating Out" and "Foreigners visits" are designed and conducted in senior high school. A software platform is implemented to satisfy the learners' demands in outdoor learning. With the use of developed platform during outdoor learning, learners are enabled to keep record of the learning progress in a facilitating way and share and discuss with other learners their own experiential knowledge independent of the time and location.

The rest of this paper is organized as follows: Section 2 give the design of mobile learning platform. Section3 describe the conduct of outclass learning activities and experimental study in section 4. Finally, the paper concludes in Section 5.

II. THE DESIGN OF MOBILE LEARNING PLATFORM

A good outdoor mobile learning platform should has the following factors: mobility, instructors-learners interaction, knowledge accessibility and so forth. As shown in Fig. 1, our system offers many useful functions which have been packaged into different modules. The "Video-Management" module and "Voice-Management" module allow students to record activity-related films, pictures, and conversations during the participation of the activity and share the record with other learners after activity. The "Wireless Transmission" module supports download learning-sheet and activity-related materials from server and upload the activity record and reports to server in a wireless manner. The "Presentation-Management" module helps student to prepare a presentation with integration of various formats including video, audio, picture and text after the activity. From the use of "Help" module, students may get help from teaching materials during the participation of activity. With the support of above mentioned modules, the system enables students to keep the track of their learning process,

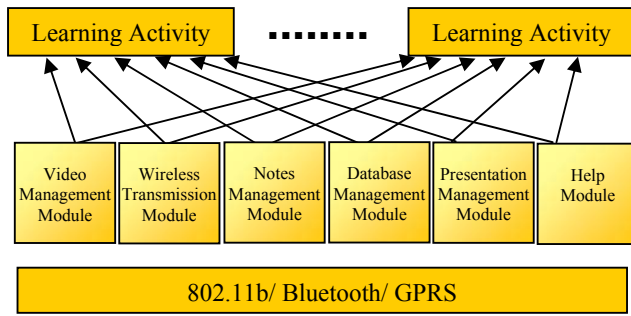understand their learning efficiency, and overcome their learning disability.

Figure 1: Modules designed for outclass learning activities.

III. THE CONDUCT OF OUTCLASS LEARNING ACTIVITIES

Outdoor learning offers opportunities of learners another experience-based learning that cannot be derived from classroom teaching. Learners can develop the academic skills as well as the social skills by physically attending in outdoor learning. Mobile outdoor learning, utilizing mobile computing technology in learners' outdoor learning, facilitates and enhances the learners' outdoor learning with the aids of wireless learning equipments.

Three scenarios of learning activities have been conducted using the developed learning system and introduced below.

■ *Activities : Outclass English Learning & Eating Out*

In classroom style English learning, reading and writing skills are usually more heavily emphasized. As a result, students are more confident about their reading and writing skills, but even after extensive study, they are fearful of conversing with a foreigner. Practice English with native English speaker will not only help student exercise their skill in English listening and speaking, but also increase their experiences in interaction with a foreigner. We design outclass English learning activity to satisfy both the desire of interaction with native English speaker and the demand of extending the scope of lesson study in textbook.

Two learning activities, "Making an Interview with Foreigner" and "Eating Out", are designed and conducted in senior high school. The two learning activities are designed according to lessens in classroom study of English course. As show in Fig 2a, the design "Making an Interview with Foreigner" aims at creating a practical environment for situated learning in English and providing opportunity of interaction with native English speaker. Through the interactive dynamic conversation with foreigner, students may practice their English skill in listening and writing. Each student is equipped with a tablet PC embedded with video & voice recording hardware and the developed outclass learning activity system. Students are partitioned into groups with each group containing three students.

In preparing, the teacher prepares a learning sheet that lists several questions for students to ask the foreigner about the similarly and difference in culture of Taiwan and his or her

country. The teacher also designs the learning sheet including the learning flow, the rules, and the frequently used vocabularies, phrases, and sentences as the preliminary guidance in "help" module, through which students may prepare their interview or get help when they visit the foreigner. In the activity, each group then visits the foreigner to obtain the answer of questions indicated in the learning sheet. During the interview, students may use the designed system to keep the track of interview process in their tablet PC. After that, students in a group will cooperatively prepare the presentation material in classroom and share their experience with the others. Finally, a web-based scoring system is provided for each group to score the presentation of the other groups, motivating them in English learning based on competitions. The second activity, "Eating Out" is similar to the first one as show in Fig 2b. Students are requested to order western-style food in a restaurant, record the activity process and share their experiences with the others.

(a) "Making an interview with Foreigner" activity.

(b) A learning sheet that indicates the learning flow, rules, and questions for answer.

Figure 2: A snapshot of interview and eating out.

The two learning activities, "Making an interview with foreigner" and "Eating out", offer opportunities for students to interact with foreigner. Students are motivated to prepare and practice the topics and the vocabularies, phrases, sentences that might be used in the interview. Practice English with native English speaker not only help student exercise their skill in English listening and speaking, but also extend their learning scope in class lesson. The developed system also helps them record, review and share with the others the learning process. With the experience of interaction with foreigner, they are

more confident in conversation and are not fearful of conversing with a foreigner.

■ *Activities: Foreigners visits*

In the activity, students are asked to interview the foreigners. By using Tablet PC with developed system, the student can make easy foreigner visits with the functions that system provides. For instance, through the CCD video taping recording and video-playing module provided by system, the students can easily videotape the procedure of the foreigner visit. All the procedure of foreigner visit will be stored in the database for future reference.

The user interface of the system is quite convenient for each learner. Leaner just follows step-by-step approach provided by the system and use it with ease right away. Learners can briefly review the whole procedure of the activity by the preliminary of the activity module in developed. The teacher posts the related document in which teacher can remind the students of the notice events in the activity before the activity proceeds.

All the procedure of foreigner visits will be videotaped in the database and saved as files. Learners can select the desired film files and review the video film after the activity. By using the note-taking and file sharing module, learners are able to keep notes in the process of the activity. Besides, learners can write their own opinions about the experience of visiting with foreigners in a text file and upload the file to the database for other learners' observation through file sharing module. Learners can give each other a score depending on the content of text file uploaded to the database.

The developed system also offer photography module which enable learners take any photo related to the activity. If learners have taken any photos in the activity and want to share with the other learners, they can upload the photo to the database. Other learners can download these photos from the system and share different other's experience of the activity.

After the activity, learners can download other learners' works from database, appreciate the work, and give each of them a score through the grading module in developed system. Teacher can realize the student's grades from the database.

IV. TRIAL TEST AND ASSESSMENT

To understand the effect of the developed system to the students, a series of investigation are done during the activities including design and progress of learning activities, student questionnaire, and analysis from the questionnaire. The questionnaires help to understand the analysis the necessity of the developed system which is the main concern in our research. Ranging from the function of the developed system, the content in the developed system, the application in the developed system in the learning technology, the questions in questionnaire is specifically designed such that it can fairly reveal the viewpoints of the questionnaire respondents. The questionnaire is issued to the student respondents before the activity and after the activity. Through the statistical analysis and quantization of the result s of students respondents, the

respondents' opinion and viewpoints can be revealed and help in the future design of the teaching activities.

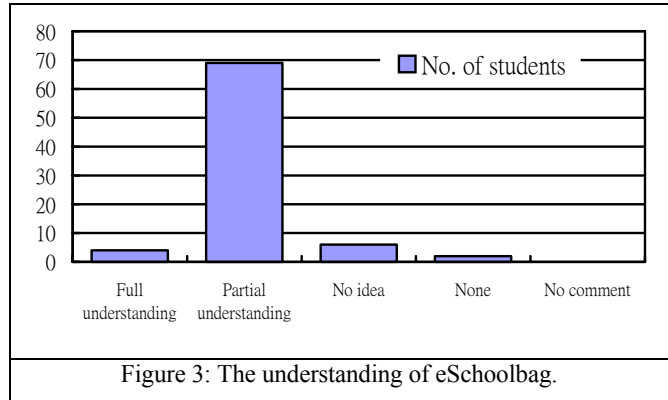Figure 3: The understanding of eSchoolbag.

Figure 3 show how much students know about usage of the Tablet PC with the developed system before the activity. From the chart, it seems much of the student respondents partially know about the usage of the Tablet PC with the developed system. It is not difficult to understand the result. The Tablet PC with the developed system is more like the computer. Some of the respondents think the Tablet PC with the developed system is a small-size computer. In fact, the Tablet PC with the developed system is not merely a small-size computer. It has still something different from small-size computer as notebook; the Tablet PC with the developed system put more emphasis on the electronification of teaching material and it also has some basic function that a computer can provide. The electronification of teaching material is different from e-textbook. In the Fig. 4, student respondents show their expectation of the function that the Tablet PC with the developed system can provide. The majority of students think the Tablet PC with the developed system should be at least quipped with the following 3 functions: "surfing the net", "providing basic programs like drawing, word processing, reading, and etc.", "e-mail reception and delivery".

The Fig. 5 shows the example of the result. It seems student think that it is the most adequate to use the Tablet PC with the developed system as a learning-aided tools in the English class. The reason they held is English is practical in life. The tablet PC with the developed system provides a platform and good interface that allows student to talk with native speakers. The Tablet PC with the developed system can help students in note-taking and reviewing the lesson. Through the repeated practice, students can recognize the accurate pronunciation and their listening ability can also be improved. Besides, English requires some video and audio information as helpful aid to learning. The Tablet PC with the developed system also provides the function of electronic English/Chinese dictionary and allows students to go on the internet to find some information about English and send messages through it. All the aforementioned properties and functions make the Tablet PC with the developed system a suitable learning-aid tool in English. Parts of students think the Tablet PC with the developed system is hard to be used in the learning in some course like math, music, and so forth. They think the subject like math focus on practice in exercises which need pen and
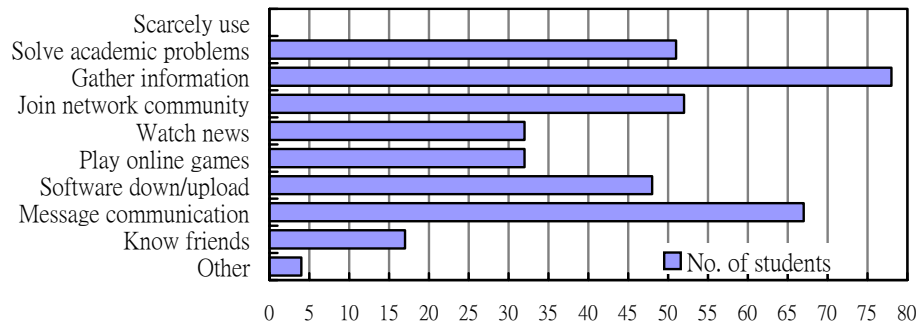
Figure 4: The expectation of the function that the learning system can provide.
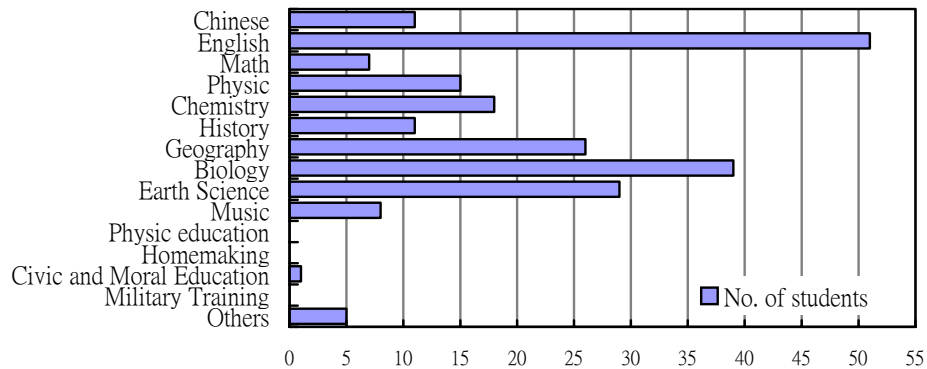

Figure 5: The expected usage of eSchoolbag.

white paper to practice and computer peripheral like mouse cannot replace pen because it's not convenient as pen. Much time will be spent in doing exercises. Other courses like music and physical education don't have necessity to use the Tablet PC with the developed system as a learning-aided tool.

In brief, the project design in teaching with using the Tablet PC with the developed system is successful. Many students give more positive response to the activity and they did gain more knowledge and some professional skill using. The Tablet PC with the developed system in the class. The designed activities helps them more thing more than what they can gain from textbook such as peer cooperation, knowledge of the Tablet PC with the developed system, and so forth.

V. CONCLUSIONS

This paper describes the learning activities and platform design for mobile learning. Ad Hoc and mobile classrooms and eSchoolbag systems are designed and implemented. The development of information and communication technology changes how, what, who, when, where and why we learn. Unfortunately, little is known of the exact impact that these changes will have on education. However, we are certain that several new learning and teaching styles (or new learning models) will emerge to cope with the changes in the near future.

With the Ad Hoc and mobile classroom and eShoolbag systems, the authors believe that the new learning models are easy to establish both indoors and outdoors.

REFERENCES

[1] Quinn, C. "*mLearning: Mobile, Wireless and In-Your-Pocket Learning*". Line Zine Magazine 2000.
http://www.linezine.com/2.1/features/cqmmwiyp.htm

[2] Sariola, J., Rönkä, A., Tella, S. & Kynäslahti, H.," *From Weak Signals to the Concept of mLearning: The LIVE Project Revisited*", ICT in the Primary School: Changes and Challenges,pp. 48–65, 2001.

[3] Kynäslahti, H.," In Search of Elements of Mobility in the Context of Education",in Mobile Learning (eds. H. Kynäslahti & P. Seppälä), pp. 41–48, 2003.

[4] C. Y. Chang, J. P. Sheu,"*Design and Implementation of Ad Hoc Classroom and eSchoolbag System for Ubiquitous Learning"*, in IEEE International Workshop on Mobile and Wireless Technologies in Education(WMTE 2002), pp. 8-14, Sweden, Aug. 2002.

[5] Y.-S. Chen, M. T.-C.Kao, and J.-P. Sheu," *A Mobile Scaffolding-Aid-Based Bird-Watching Learning System*", Journal of Computer Assisted Learning, Vol. 19, No. 3, pp.347-359, Sep. 2003.

[6] Masaya Okada, Akimichi Yamada, Hiroyuki Tarumi, Mizuki Yoshida, and Kazuyuki Moriya: "*DigitalEE II: RV-Augmented Interface Design for Networked Collaborative Environmental Learning*", Proceedings of the International Conference on Computer Support for Collaborative Learning , pp. 265 - 274, 2003.

Identity-Based Instant Broadcast in Mobile Ad-Hoc Networks

Yi Mu and Willy Susilo

School of Information Technology and Computer Science
University of Wollongong, Wollongong
NSW 2522, AUSTRALIA
Email: {ymu,wsusilo}@uow.edu.au

Abstract

In a mobile ad hoc network, security of communications amongst mobile hosts cannot be based on an online trusted authority due to randomness in forming the network. This paper proposes a novel scheme that allows any mobile host in a randomly formed ad hoc network to broadcast a confidential instant message to a subgroup of mobile hosts in the network so that all associated hosts can receive the message concurrently. The merit of the scheme is that broadcasting is that the encryption of messages relies upon only the identifiers of the associated hosts such as BD_ADDRs or domain names. There is no need for any trusted online authority

Keywords: Ad-hoc Network, Secure Protocol.

1 Introduction

A mobile ad hoc network can randomly be formed with requiring any fixed infrastructure. The basic idea of ad-hoc networks has been known since 1970s with the introduction of *packet radio networks* which was used for military applications PRnet, developed by the American Defense Advanced Research Project Agency (DARPA). In the process of designing the standard for Wireless Local Area Networks (WLAN), which is also known as the 802.11 standard, the IEEE replaced the term packet radio network by ad-hoc network, which is frequently associated with self-organization. To enable this feature, a dynamic model must be employed, by allowing mobile hosts to enter and leave the network without interfering the current communication. This feature is really attractive and this allows the ad-hoc networks to be used in many applications.

Along with the required property of ad-hoc networks, the security assurance must also be considered. In a dynamic network with limited resources such as

a mobile ad-hoc network, a communication protocol has to address several important aspects, including: *availability:* the protocol must guarantee that the network will survive even with the presence of some malicious users; *confidentiality:* the protocol should ensure that certain information is never disclosed to unauthorized users; *integrity:* the protocol must ensure that a message being delivered is not corrupted; *authenticity:* the protocol should ensure that a message originates from the claimed host that the message identifies as its source; and *cost effectiveness:* the protocol should make sure that ensuring all the above properties will not significantly more expensive than communicating without these requirements.

The above important features of mobile ad-hoc networks are desirable. There have been already several solutions proposed for securing the communication among hosts in ad-hoc networks, but all of these solutions failed to address full security issues mentioned above. On one hand, some solutions provide the security of mobile ad-hoc networks without allowing its dynamic feature (for example, [9]) and on the other hand, some solutions allow the dynamics that happen in the group with some limited security assumptions.

In this paper, we propose a construction of *instant broadcast* for a freely formed mobile ad hoc network, where mobile hosts can join or leave the ad hoc network at random. In our scheme, any mobile host who has joined the network can broadcast an instant message to a subgroup of mobile hosts so that they can receive the message concurrently. The merit of our scheme lies in the method that allows a broadcast encryption relies upon only the identifiers of the associated the hosts and there is no need for any online trusted authority.

The rest of this paper is organized as follows. Section 2 reviews some related works and describes our contributions. Section 3 Presents the model of our ad hoc network. Section 4 gives the initial setup of the ad hoc network. Section 5 describes the secure protocol in

detail. Section 6 discusses the security of our protocol. The final section is the conclusion.

2 Related Work

The concept of secure broadcasting was introduced by Fiat-Naor[5] for solving the problem of multi-message encryption, which is known as the *broadcast encryption*. Conceptually, the broadcast encryption is based on a single symmetric cipher equipped with a number of affine substitution boxes, where n messages can be converted into n ciphertexts that are broadcast to the other end of a communication channel. The ciphertexts are then decrypted with the same key(s). We must point out that broadcast encryption is completely different from our broadcasting concept that implies one-to-many mappings accommodating dynamic user/key management.

In general, the merit of a secure broadcast system relies on the following properties, which are also the basic requirements for our systems. (1) Security: a broadcast system should be secure in a polynomial time frame. As a result, only a legitimate user can receive a message sent by a broadcaster or broadcasters. (2) Dynamic user/key management: a broadcaster can arbitrarily add or remove a subscriber to or from a receiver group without changing the decryption keys of the other subscribers. (3) Efficiency: the computation and communication overhead associated with a broadcasting must be kept to a minimum.

The traditional ways in handing key distribution in a mobile ad-hoc network are always problematic, since the general assumption of existence of a trusted authority (`TA`) is not sound in a freely formed mobile ad-hoc network. However, it should be reasonable to assume the existence of some sort of trustiness to enable key distribution. We refer this assumption to as two scenarios: *Strong Trust* and *Weak Trust*.

- **Strong Trust**. Mobile hosts rely on an online `TA` for key distribution. This assumption is not valid in a freely formed ad-hoc network; therefore it is out of our interest.

- **Weak Trust**. Mobile hosts are associated with a `TA` such as their home domain, but they do not require an online `TA` in key distribution. There have been a quite lot of work based on this assumption, for example, [9, 13, 8, 12].

Based on the strong trust model, any existing key distribution protocols could be applied (for example [4, 7]); therefore it is out of our interests. One of interesting key distribution models that satisfy the weak

trust assumption is the dynamic trust model was initiated by Kong *et al* [9] based on threshold cryptography. In their model, mobile hosts can dynamically construct a trusted group consisting of predefined number of members in an ad-hoc network. It requires to have a Trusted Authority (`TA`) at beginning to construct shares for an authorized secret key. The trusted group can play a role of online trusted authority for future key distribution. The major drawback of this scheme is that the share construction is *static*. It means that it is *not* suitable for a freely formed ad-hoc network where the mobile hosts can *join* and *leave* freely.

There are some other attempts, for example, making use of the Diffie-Hellman key exchange algorithm with the assumption that there exists a `TA` who can issue public key certificates [1]. However, the computational overhead of such system is proportional to the number of users; therefore it will be considerably high when the number of mobile hosts is large. Some other related work in this area includes the security policy [11, 10] which is suitable to small devices with limited computational power.

2.1 Our contributions

In this paper, we introduce the concept of *instant broadcast* for freely formed mobile ad hoc networks. Broadcasting in our scheme relies upon only the identities of the associated mobile hosts or receivers. Our scheme allows broadcasting to be *dynamic* which means that any mobile hosts or receivers can be dynamically selected by the broadcaster. In other words, the size of the randomly selected subgroup can vary. Moreover, our scheme does not require any online trusted authority that manages key distribution in the network; therefore, there is no any need for us to set up the network with secret sharing scheme given by Kong *et al* [9].

We make use of the recent work by Boneh and Franklin in identity based cryptography and our formulism on a polynomial setting. In a broadcast, the broadcaster selects identities of intended receivers and utilize these identities and a system public key to encrypt a message, the each mobile host in the group can decrypt the message with his/her private key. The idea for such broadcasting is entirely novel and we found that it is especially suitable for small mobile ad hoc networks.

3 Model and security requirements

3.1 Model

We now define our model and provide some notations required throughout this paper.

An ad-hoc network, $\mathcal{A}_n$, as illustrated in Figure 1 consists of n mobile hosts. Every mobile host can broadcast an instant message to a group of hosts in terms of their identities. We call such groups as instant groups or groups.
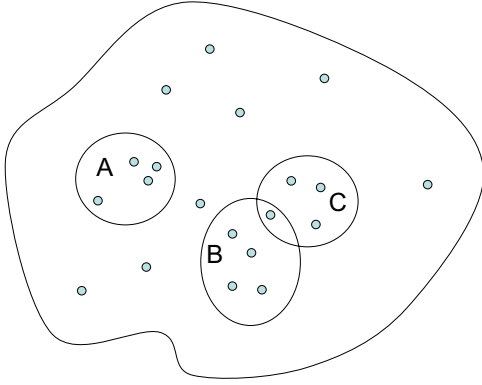

Figure 1: Illustration of an ad-hoc network which contains three ad hoc groups A, B, and C which are freely formed in terms of selected identities of mobile hosts by the broadcaster. Groups B and C have an overlap in which one mobile host has a membership in both B and C.

Definition 1 *Given an ad hoc network $\mathcal{A}_n$, we say that $\mathcal{G}_m$ of size m is an instant ad hoc group with respect to an encrypted message M, if all identities of mobile hosts $\mathcal{H}_i \in \mathcal{G}_m$ are associated with M and all hosts $\{\mathcal{H}_i\}_{i=1,\cdots,m}$ are able to extract M.*

Our protocol consists of several *events*: REGISTRATION, JOIN$_{\text{network}}$, BROADCAST.

- In a REGISTRATION event, a mobile host $\mathcal{H}_i$ registers itself with its home domain. Upon the completion of REGISTRATION, the host $\mathcal{H}_i$ obtains the public key P_{pub} of the system and its private key $S_{\mathcal{H}_i}$.

- The JOIN$_{\text{network}}$ event is referred to as a construction event. In a JOIN$_{\text{network}}$ event, a mobile host $\mathcal{H}_i \not\subset \mathcal{A}_n$ can join the ad-hoc network $\mathcal{A}_n$ that consists of existing mobile hosts and available spaces

to be filled by new mobile hosts. As a result of JOIN$_{\text{network}}$, we have $\mathcal{A}_{n+1}$. After JOIN$_{\text{network}}$, $\mathcal{H}_i$ can invoke a BROADCAST event.

- In a BROADCAST event, a host $\mathcal{H}_i \in \mathcal{A}_n$ broadcasts an encrypted message to an instant ad-hoc group $\mathcal{G}_m$, $m \leq n$ and all hosts in $\mathcal{G}_m$ can extract the message.

To construct a secure and dynamic ad-hoc mobile network, we require the following security properties. The scheme must be secure against collusion attack. Namely, a collusion of some mobile hosts in a group $\mathcal{G}_m$ cannot find out others private decryption keys. We require our scheme is as secure as the Bilinear Diffie-Hellman problem.

4 Initialization

In the following, we describe a secure scheme that fits into our model introduced in Section 3. In this scheme, we assume that there is a home server, which only exists in the initialization phase. His role is only to setup the system. We note that this is applicable in practice since normally a system administration needs to allocate some identification to each mobile device.

4.1 Bilinear pairing

The basic setup of our schemes are based on the pairing [3]. The basic system parameters for our systems are described as follows. Let E denote an elliptic curve over a field K with characteristic > 0, and $E[n]$ be its group of n-torsion points.

Definition 2 *Let $n \in \mathbb{Z}_{\geq 2}$ denote an integer, coprime to the characteristic of K with characteristic > 0. The Weil pairing is a mapping*

$$\hat{e} : E[n] \times E[n] \to \mu_n$$

where μ_n is the group of nth roots of unity in $\bar{K}$.

Under the definition of the Weil pairing, if $\hat{e}(P,Q)$ is not the unit in μ_n, then $\hat{e}(aP, bQ) = \hat{e}(P,Q)^{ab}$ for $P, Q \in E[n]$ and all $a, b \in \mathbb{Z}$. Please refer to Page 43 of [2] for details of the Weil pairing.

Group $E[n]$ is a cyclic additive group, now denoted $\mathbb{G}_1$, which maps to a cyclic multiplicative group $\mathbb{G}_2$ by the Weil pairing. If n is prime, then both $\mathbb{G}_1$ and $\mathbb{G}_2$ have a prime order.

4.2 Setup

The home domain $\mathcal{H}$ needs to set up the system such that all necessary parameters can be used during ad hoc network constructions. $\mathcal{H}$ selects the following parameters:

- a large prime $p = 2q + 1$ where q is also prime,

- an additive group $\mathbb{G}_1$ and a multiplicative group $\mathbb{G}_2$ (both have order p),

- a master secret key $s \in \mathbb{Z}$, and

- a number $P \in \mathbb{G}_1$.

$\mathcal{H}$ then computes the system public key $P_{pub} = sP$ which is then made available for all mobile hosts who have registered with $\mathcal{H}$. $\mathcal{H}$ also selects two strong public one-way functions $H_1 : \{0,1\}^* \to \mathbb{G}_1$ and $H_2 : \mathbb{G}_1 \to \{0,1\}^*$.

5 ID-based instant broadcast

5.1 Registration (REGISTRATION)

REGISTRATION event occurs between a mobile host and the home server. When a mobile host $\mathcal{H}_i$ to join the system. The home server assigns $\mathcal{H}_i$ with a private key $S_{ID} = sQ_{ID}$, where $Q_{ID} = H_1(ID)$. After REGISTRATION, $\mathcal{H}_i$ becomes a member in the system. Any subset of mobile hosts in the system can form an ad hoc network.

5.2 Construction (JOIN$_{\text{network}}$)

A subset $\mathcal{A}_n$ can setup an ad-hoc network with n hosts, $\mathcal{H}_i \in \mathcal{A}_n$, $i = 1, 2, \cdots, n$. After the JOIN$_{\text{network}}$ event, all BD_ADDR's in $\mathcal{A}_n$ can be observed from all mobile hosts in $\mathcal{A}_n$. Secure communications amongst these mobile hosts can now be "ID-based".

5.3 Instant broadcast (BROADCAST)

The BROADCAST event is performed as follows. Any host $\mathcal{H}_j \in \mathcal{A}_n$ can broadcast an instant message to an arbitrary subgroup $\mathcal{G}_m \subseteq \mathcal{A}_n$.

To broadcast an instant message to $\mathcal{G}_m$, the sender prepares the following parameters. Assume the identities of mobile hosts are denoted by ID_i, $i = 1, \cdots, m$.

- Select a random $r \in \mathbb{Z}$.

- Compute $R = rP$.

- Compute $x_i = \hat{e}(rQ_{ID_i}, P_{pub})$.

- Compute the polynomial function $f(x) = \prod_{i=1}^{m}(x - x_i) \bmod p$. From $\prod_{i=1}^{m}(x - x_i) = \sum_{i=0}^{m} a_i x^i \bmod p$, we obtain

$$
\begin{aligned}
a_0 &= \prod_{j=1}^{m}(-x_j), \\
a_1 &= \sum_{i=1}^{m}\prod_{j \neq i}^{n}(-x_j), \\
&\cdots \\
a_{m-2} &= \sum_{i \neq j, j > i}^{n}(-x_i)(-x_j), \\
a_{m-1} &= \sum_{j=1}^{m}(-x_j), \\
a_m &= 1.
\end{aligned}
$$

$\{a_i\}$ satisfy $\sum_{i=0}^{m} a_i x_j^i = 0 \bmod p$, $j = 1, \cdots, m$. From the set $\{a_i\}$, we can construct the corresponding exponential functions,

$$\{a_0 P, a_1 P, a_2 P, ..., a_n P\} \equiv \{P_0, P_1, P_2, ..., P_m\}.$$

$\{x_i\}$ form an authorized set $\mathcal{X}_m$ for mobile hosts in $\mathcal{G}_m$.

Let $M \in \{0,1\}^*$ be the message. The encryption protocol on M is as follows.

- Pick a random $k \in \mathbb{Z}$ and a random $D \in \mathbb{G}_1$.

- Compute $(m+2)$-tuple $T \leftarrow (R, M \oplus H_2(D), D + kP_0, kP_1, ..., kP_m)$.

- Broadcast T to $\mathcal{G}_m$.

To decrypt T, $\mathcal{H}_i \in \mathcal{G}_m$ computes

$$\hat{e}(S_{ID_i}, R) = x_i,$$

$$C_0 \prod_{j=1}^{m} x_i^j C_j = D,$$

$$C \oplus D = M.$$

The correctness of the decryption is proved as follows.

$$\hat{e}(S_{ID_i}, R) = \hat{e}(sQ_{ID_i}, rP) = \hat{e}(rQ_{ID_i}, P_{pub}) = x_i,$$

$$
\begin{aligned}
C_0 \prod_{j=1}^{m} x_i^j C_j &= D + k(a_0 + a_1 x_i + ... + a_m x_i^m)P \\
&= D + 0P = D.
\end{aligned}
$$

The soundness of the protocol is that only the mobile hosts in $\mathcal{G}_m$ can decrypt the message. This is because for any $x_j \notin \mathcal{X}_m$, $\sum_{i=0}^{m} a_i x_j^i \neq 0 \bmod p$.

5.4 Construction of a short-term group

If $\mathcal{H}_i \in \mathcal{A}_i$ broadcasts a session key k_s to $\mathcal{A}_m$ instead of a message, then an instant ad hoc group $\mathcal{G}_{m+1} = \mathcal{A}_m \cup \mathcal{H}_i$ will be constructed. k_s is then used as the group secret key that can be used for a session of communication amongst group members in $\mathcal{G}_{m+1}$. To ensure that all members in the group know who are indeed in the group, $\mathcal{H}_i$ broadcasts k_s concatenated with all ID's belonging to $\mathcal{A}_m$ using the scheme described in the preceding section to construct $\mathcal{G}_{m+1}$.

This construction requires that the initiator or broadcaster be trusted by all group members in terms of group memberships. All members in the group should know who are confined in the group. When such trustiness does not exit, the instant broadcast of messages will be a better solution.

As long as the life of $\mathcal{G}_{m+1}$ lasts, k_s can be used as the group key. The new session key can be initialized by any member in the group when felt necessary.

5.5 Computational complexity

Our scheme is suitable for an ad hoc network since most ad hoc networks have a small size of less than 100 mobile hosts. The computational complexity should be able to be coped by normal laptop computers. In particular, when broadcasting is used for session key distribution, the communications in the group can then be based on a symmetric key cipher such as AES.

Also, observe that when a session key needs to be replaced in a group, the broadcaster, the host who initiated the previous session key can reuse $\{P_0, P_1, P_2, ..., P_m\}$ to encrypt a new session key for the next session of group communications.

6 Security

Theorem 1 *The security of the encryption refers to the Bilinear Diffie-Hellman Problem (BDHP).*

Proof: In bilinear pairings, the Decisional Diffie-Hellman Problem (DDHP) is defined as follows: given $P, aP, bP, cP \in \mathbb{G}_1$ and $a, b, c \in \mathbb{Z}_q$, decide whether $c \overset{?}{=} ab \in \mathbb{Z}_q$. The bilinear DDHP is easy [6], since $e(aP, bP) = e(P, P)^{ab}$.

The security of a pairing based algorithm can be based on the computational Diffie-Hellman problem (CDHP): Let a, b be chosen from $\mathbb{Z}_q$ at random and P be a generator chosen from $\mathbb{G}_1$ at random. Given (P, aP, bP), compute $abP \in \mathbb{G}_1$. $\mathbb{G}_1$ is referred to as a Gap Diffie-Hellman (GDH) group and CDHP can be referred to as a Gap Diffie-Hellman problem, if DDHP can be solved in polynomial time and no polynomial algorithm can solve CDHP with non-negligible advantage within polynomial time.

There is a new type of hard problem [3], which is referred to as Bilinear Diffie-Hellman Problem (BDHP). It states that given P, aP, bP, cP, find $\hat{e}(P, P)^{abc}$ In our scheme, the secret decryption key is computed from $x_i = \hat{e}(rQ_{ID_i}, P_{pub})$ and R is assumed public. Matching it with BDHP, we have that given $R = rP$, $Q_{ID_i} = aP$, and $P_{pub} = sP$, find $\hat{e}(P, P)^{ras}$, where we have assumed that $Q_{ID_i} = aP$; namely, Q_{ID_i} can be generated from the generator P of $\mathbb{G}_1$. Boneh and Franklin [3] proved the security of this kind of crypt systems by using a random oracle model. $\square$

The security of the system can also refer to the factorization of $D + kP_0$ in order for finding D. Given P as a generator of $\mathbb{G}_1$, the probability of finding D is $1/p$, which is negligible when p is large. The factorization can also be carried out using public values, $kP_1, ..., kP_m$), i.e., finding a correct x_a such that $D + kP_0 + \sum_{i=0}^{m} x_a^i kP_i = D$. This success probability is also $1/p$. Clear, these attacks are based on exhaustive search, which is hard when the data fields are large.

7 Conclusion

We have presented a novel scheme for secure instant broadcasting in mobile ad hoc networks. In our scheme, a message can instantly be broadcasted to a group of mobile hosts, where the broadcaster (a mobile host) utilizes the identities such as BD_ADDR numbers of the receiving hosts only. We have also described the case that a group of mobile hosts want to form a instant ad hoc group for one or a few sessions of communications, by allowing the broadcaster broadcasts a session key to the specific group. The communications within the group can then be based on a symmetric cipher such as AES. We have also analyzed security of our scheme and showed that it is equivalent to the Bilinear Diffie-Hellman problem.

References

[1] N. Asokan and P. Ginzhoorg. Key agreement in ad-hoc networks. *Computer Communication Review*, 2000.

[2] I. Blake, G. Seroussi, and N. Smart. *Elliptic Curves in Cryptography.* Cambridge Unversity Press, 2001.

[3] D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In J. Kilian, editor, *Advances in Cryptology, Proc. CRYPTO 2001,* LNCS 2139, pages 213–229. Springer Verlag, 2001.

[4] M. Burmester and Y. Desmedt. A secure and efficient conference key distribution system. *Advances in Cryptology - Eurocrypt '94*, pages 275–286, 1994.

[5] A. Fiat and M. Naor. Broadcast encryption. In D. R. Stinson, editor, *Advances in Cryptology, Proc. CRYPTO 93,* LNCS 773, pages 480–491. Springer Verlag, 1994.

[6] A. Joux and K. Nguyen. Separate decision deffie-hellman from deffie-hellman in cryptographic groups. available from eprint.iacr.org.

[7] M. Just and S. Vaudenay. Authenticated multi-party key agreement. *Advances in Cryptology - Asiacrypt '96*, pages 36–49, 1996.

[8] J. Kong, H. Luo, K. Xu, D. L. Gu, M. Geria, and S. Lu. Adaptive security for multilevel ad hoc networks. *Wireless Communications and Mobile Computing*, 2:533–547, 2002.

[9] J. Kong, P. Zerfos, H. Lao, S. Lu, and L. Zhang. Providing robust and ubiquitous security support for mobile ad-hoc networks. In *Proceedings of the 9th International Conference on Network Protocols*. IEEE Computer Society Press, 2001.

[10] F. Stajano. The Resurrecting Duckling - what next? *The 8th International Workshop on Security Protocols, Lecture Notes in Computer Science, 2133*, 2000.

[11] F. Stajano and R. Anderson. The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. *The 7th International Workshop on Security protocols, Lecture Notes in Computer Science 1796, Springer-Verlag*, 1999.

[12] A. Weimershirch and G. Thonet. A distributed light-weight authentication model for ad-hoc networks. In *Lecture Notes in Computer Science*. Springer Verlag, 2001.

[13] L. Zhou and Z. J. Haas. Securing ad hoc networks. *IEEE Networks*, 13(6):24–30, 1990.

Proceedings

The 2004 International Workshop on Distance Education Technologies (DET'2004)

Co-Editors

Chein-Shing Wang, St. John's and St. Mary's Institute of Technology, Taiwan
Ruei-Xi Chen, St. John's and St. Mary's Institute of Technology, Taiwan

CHRONOBOT: A TIME AND KNOWLEDGE EXCHANGE SYSTEM FOR E-LEARNING AND DISTANCE EDUCATION

Shi-Kuo Chang and Ganesh Santhanakrishnan
Department of Computer Science, University of Pittsburgh
Pittsburgh, PA 15260 USA (chang, ganesh@cs.pitt.edu)

Abstract

The chronobot is a time/knowledge exchange system. One of the objectives of the system is to improve the effectiveness of e-learning and distance education. The chronobot has the following components: the knowledge manager, the time manager, the time/knowledge exchanger and, for e-learning and distance education applications, a powerful and versatile virtual classroom. In this paper we present a time/knowledge exchange protocol. We propose a constraint satisfaction algorithm to solve the time/knowledge exchange problem, which interoperates with the various chronobots and co-ordinates the settlement of a bid in a distributed manner and selects the best bid using a metric known as the QoB (Quality of Bid). We have also begun to extract various patterns for time/knowledge exchange for incorporation in the protocol.

1. Introduction

The chronobot is basically a time and knowledge manager. It performs its functionality through time/knowledge exchange protocols. The uniqueness in the concept of a Chronobot is the exchangeability of time and knowledge. The protocol for time/knowledge exchange is key in determining the success of the system envisioned. We discuss the protocol in the following sections. Typically when a Chronobot enters a bid for time/knowledge exchange, other Chronobots respond and the time/knowledge exchange protocol is used to settle the bid.

In the two examples described in [1] the user's close supervision of the chronobot is essential for the chronobot to accomplish **timely knowledge generation**. Only the user knows what he/she really wants. If there is no close supervision, the chronobot may collect a lot of garbage, making subsequent information fusion more difficult. That is why the user must spend time to supervise the chronobot to gather information, and give appropriate labels to the collected information items. Timely knowledge generation depends upon the chronobot's ability for **information fusion**. The chronobot may also be very useful for enhancing human interactions as well.

The rest of the paper is structured as follows. In section 2, the functions that must be performed by the Time/Knowledge Exchange protocol are discussed. In section 3, a complete description of the proposed algorithm is given. In section 4, related work is discussed and finally we conclude in section 5 also outlining directions of future research work.

2. Functions of the Time/Knowledge Exchange Protocol

The Time/Knowledge exchange protocol needs to specify exactly what needs to be done under different conditions. The functions of the exchange protocol are as follows.

I. The protocol must decide whether to allow the Chronobot that wants to bid may proceed to do so, based on prior history obtained from User Profile information.

II. The protocol must decide which Chronobots can take part in the bidding process.

III. The protocol must specify the order in which the Chronobots can take part in the bidding process. In other words, the protocol must specify a global ordering of how the various bids will appear.

IV. The protocol must validate the bids of the various Chronobots.

V. The protocol must specify the order in which the bids of the various Chronobots are presented to the Chronobot that is initiating the bidding process.

VI. The protocol must specify the order in which the bids of the various Chronobots are presented to the Chronobot that is initiating the bidding process.

VII. The protocol must notify the Chronobot that is initiating the bidding process when one of the Chronobots that had bid earlier wants to retract its bid.

VIII. The protocol must decide online, using a set of criteria presented to it, which Chronobot(s) will be assigned the task required by the initiator of the bid.

IX. The protocol must notify the Chronobot(s) that is selected of the results of the bidding process. If the Chronobot(s) refuse to honor the bid, then the protocol should update the user profile appropriately and start the bidding process all over again by repeating steps I. through VIII.

X. If the Chronobot(s) agree to the bid, then the protocol must notify the Chronobot that initiated the bid that the bid has been resolved.

XI. The protocol must also notify other Chronobots that took part in the bidding process that the bid has been resolved.

XII. If the Chronobot(s) do not satisfy their agreement the protocol must update the user profile. If the Chronobot that initiated the bid still needs the job to be done, then the protocol must start the bidding process all over again by repeating steps I. through VIII.

XIII. When the Chronobot(s) performs the requested task, the protocol must update the user profile appropriately.

XIV. The protocol must notify the Chronobot that initiated the bid that the job has been accomplished.

In order to perform the aforementioned tasks, there needs to be a detailed algorithm that maps the requirements of the Chronobot that initiated the bid and the Chronobot(s) that proposes to perform the task. I go on to discuss the algorithm in the next section.

3. Algorithm for Time/Knowledge Exchange

The Algorithm for Time/Knowledge Exchange must first take into account all the initial specifications of the protocol and devise a way in order to settle the bid among the Chronobot that initiated the bid and the Chronobot(s) that proposes to perform the task. The Algorithm is basically a constraint satisfaction algorithm that interoperates with the various Chronobots and co-ordinates the settlement of a bid in a distributed manner selecting the best bid using a metric known as the QoB (Quality of Bid).

Firstly, the algorithm performs validation as described in section 2. Then it utilizes a technique based on Multi-Expert weight updation to select the best possible Chronobot(s) to perform the task. This procedure is described as follows.

The algorithm for time/knowledge exchange must consider criteria for deciding the

Quality of Bid (QoB). We now go on to describe the complete set of criteria.

3.1 The Criteria for Time/Knowledge Exchange

The complete set of criteria that the algorithm must take into account in order to decide the Chronobot(s) that win(s) the bid, are as follows.

1. The user profile history of the Chronobot:

   The User profile history will be maintained for each Chronobot entering the bid and it will be checked by a routine before the bid is being finalized. The Chronobot initiating the bid must annotate the weight that it wishes to assign to this in the skill set specification.

2. Detail whether the Chronobot has defaulted before or not:

   The information regarding whether the Chronobot that is entering the bid has defaulted or not must be maintained for each Chronobot and the appropriate weight that the Chronobot that is initiating the bid gives to it must also be retrieved from the database.

3. The Skill set required by the Chronobot initiating the bid:

   The Chronobot initiating the bid can specify how much weight it wants to assign to each of the skill sets respectively.

4. Prior experience with handling similar problems:

   This can be annotated in the skill set requirement and the weight that the Chronobot initiating the bid wants to assign to this can also be specified.
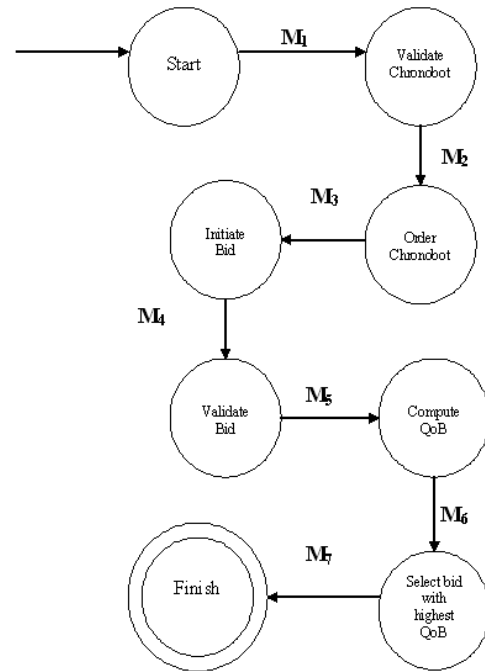
5. Whether the Chronobot initiating the bid has a preference or not in multiple Chronobot(s) solving the problem:

In certain cases, it is better for multiple Chronobots to work on solving the problem. This can be specified by the Chronobot, initiating the bid, in the weight that it wants to assign for this criterion.

6. The feedback from previous Chronobot(s):

   The Chronobot(s) that have utilized the Chronobot entering the bid must leave behind feedback, either positive or negative. The Chronobot, initiating the bid, can specify the weight that it wants to assign for this criterion by annotation in the skill set.

Now we go on to pictorially describe the schematic for the Time/Knowledge exchange protocol as follows.

3.2 The propagation of bids among the various Chronobot(s)

At this point, it appears that the algorithm for Time/Knowledge Exchange must propagate the bids among the various Chronobots using a centralized update approach. In other words, the bids of the various Chronobots will appear on the document that hosts the initiation of the bid.

The Chronobots are free to update their bids and the updation should appear with minimum time delay on the document that hosts the initiation of the bid.

The actual Automata-based formulation of the protocol is continued in the following sections starting with the definitions.

3.3 Definitions

Definition 1. The self-model [2] of a Chronobot includes the current set of values for the various criteria and the weights that the Chronobot assigns for each of those criteria.

Definition 2. The alien-model [2] of a Chronobot includes the current set of values that the various Chronobots taking part in the bidding process assign to the multitude of criteria specified by the Chronobot initiating the bid.

Definition 3. A Chronobot is a 6-tuple (M, S, f, s_0, F, DB)

Where

M is a non-empty set of messages (the message space)

S is a non-empty set of states (the state space), $S = S_c * D_s * D_a$

Where

S_c is a nonempty set of states of the Chronobot

D_s is a nonempty set of self-models of the Chronobot, and

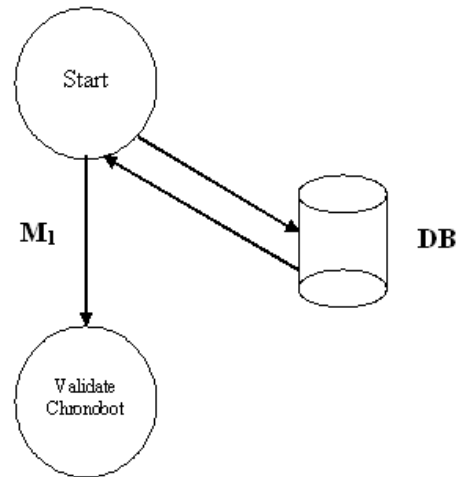D_a is a nonempty set of alien-models of the Chronobot

$f : S_c * (M * M)^* \to S_c$ is the state transition function. Given the current state of the Chronobot and the history of messages exchanged between the Chronobot initiating the bid and the Chronobots participating in the bidding process, f specifies the next state

s_0 in S is the initial state of the Chronobot

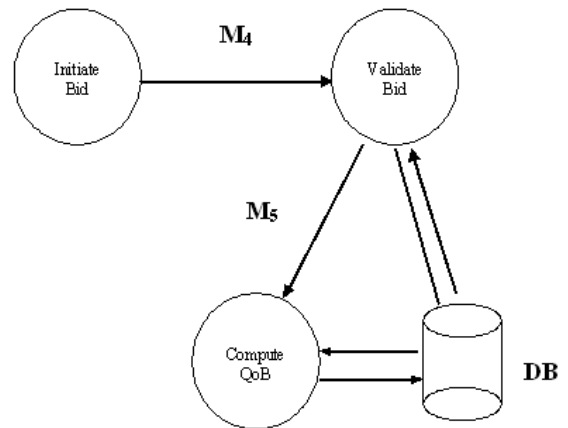F, a subset of S, is the nonempty set of final states

DB is the Database that stores the history of all the Chronobots involved in the Time/Knowledge Exchange.

We now go on to describe the messages that are generated for state change.

Initially the Database of User Profile History is accessed and then based on the state of the Chronobot specified in the database, message M_1 is used to validate the Chronobot.

Message M_2 is used to order the Chronobots to enable them to enter the bids in a distributed manner.

M₂

M₄

M₅

Message M₃ is used to signal to the Chronobots that are participating in the bidding process to start entering their bids.

M₃

M₆
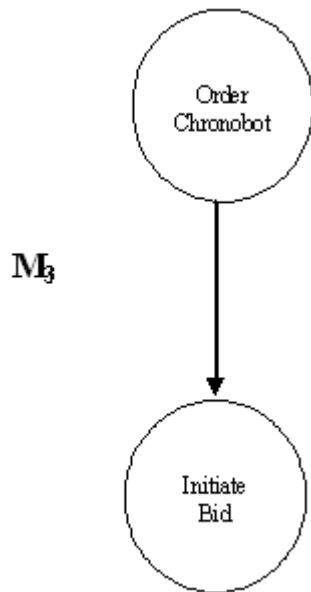
M₇

DB

Message M₄ is used to validate the bids that are presented by the Chronobots that are participating in the bidding process. Message M₅ is used to compute the QoB based on information from the Database.

Message M₆ is used to inform which Chronobot has the highest QoB among the bids that are presented by the Chronobots that are participating in the bidding process. Message M₇ is used to announce the

winner(s) of the bidding process. Then the database is updated appropriately.

3.4 The Time/Knowledge Exchange Protocol

The Time/Knowledge exchange protocol specifies exactly what needs to be done under different conditions. The Automata-based formulation of the different steps that the protocol must take is as follows.

The protocol for Time/Knowledge Exchange must first take into account all the initial specifications of the protocol and devise a way in order to settle the bid among the Chronobot that initiated the bid and the Chronobot(s) that proposes to perform the task. The Algorithm is basically a constraint satisfaction algorithm that interoperates with the various Chronobots and co-ordinates the settlement of a bid in a distributed manner selecting the best bid using a metric known as the QoB (Quality of Bid).

Firstly, the algorithm performs validation as described earlier. Then it utilizes a technique based on Multi-Expert weight updation to select the best possible Chronobot(s) to perform the task. This procedure is described as follows.

Our terminology and approach for the problem is as follows.

➢ The "experts" are criteria that make predictions; denoted by vector (x_t).

➢ Weights of experts (w_t) represent their quality of predictions.

➢ *The Master Algorithm* predicts with a weighted average of the experts' predictions
   • $y_t = w_t \cdot x_t$

➢ Depending on the outcome ($\acute{y}_t$) the weights of the experts are updated.

   • $\acute{y}_t = w_{t+1} \cdot x_{t+1}$

➢ In this way the system improves itself in the form of a feedback based approach.

The updation of weights can be done using either of two approaches. The first approach

is called Loss Update (Weighted Majority Algorithm [3]). It uses the following formula for weight updation.

$$w_{t+1},i := w_t,i \cdot e^{-\eta Lt,i} \quad for\ i = 1..n$$

Normalization ($\sum w_t,i$)

where :
   η (eta)   = the learning rate
   $W0$ = (1/n, 1/n, … , 1/n) → weights initialized equally

But the problem is that Loss Update learns too fast, but does not recover fast enough (*e −$\eta Lt,I = e$ −$1 \approx 1/2.7$*)

There is a better approach for Weight updation known as "Share" [4] Updates. The motivation is essentially biological for Share. For instance imagine all cache policies as species competing for food (credits in the equation) in habitat (the criteria that are involved for selection).

The *fitness* of specie is based on how well it eats
   – In our context fitness of a species is its weight assigned accordingly by the user.

The population share (or frequency) of a specie depends on its fitness. Highly fit species may starve the others and if conditions change the whole system collapses. Predators (probabilistically) prey on the most frequent (or easy) species. Predators protect diversity (mixing) among species. By avoiding the most fit specie from starving the others. Our predator implementation in terms of the credits assigns the credits to the various criteria. The problem is that we cannot allow duplications, in other words, assign probabilistically and also lucky draws (unfairness) can cause bad assignment of credits. Hence the appropriate implementation works as follows.

➢ A pool is created from the shared weights

➢ The algorithms are forced to share in proportion to their loss

➤ Weights are redistributed from the pool to make sure all policies have some minimal weight (w_{t+1},i)

Incorporating the aforementioned weight updation algorithm employing Share Updates, the general algorithm for Time/Knowledge exchange is as follows.

The following are the procedures used in the algorithm.

Let UPDATE_USER_PROFILE() be the procedure that updates the user profile of Chronobots participating in the system.

Let ORDER_CHRONOBOTS() be the procedure that globally orders the various Chronobots participating in the system.

Let VALIDATE_BID(k) be the procedure that validates the bids of the various Chronobots.

Let VALIDATE_CHRONOBOT(k) be the procedure that validates each and every Chronobot that wishes to participate in the bidding process.

Let Chronobot (i) denote the one initiating the bid.

The step by step algorithm is as follows.

Step 1. Chronobot (i) initiates the bid for a task.

Step 2. For each Chronobot (j) wishing to participate in the bidding process call VALIDATE_CHRONOBOT(j) to validate each of the Chronobots that wishes to participate in the bidding process.

Step 3. Call ORDER_CHRONOBOTS() to order the Chronobots wishing to participate in the bidding process.

Step 4. For each bid k initiated, call VALIDATE_BID(k) to validate the bid originating from the various Chronobots.

Step 5. Compute the following for each bid that has been initiated in order to compute the QoB (Quality of Bid).

$$QoB\ (i) = W_1 * C_1 +$$

$$+W_2 * C_2 + \ldots$$

$$+W_n * C_n$$

Where:

C_1, C_2, … C_n are the various criteria for deciding the bid such as user profile history of the Chronobot, whether the Chronobot has defaulted before or not, and so on.
W_1, W_2, … W_n are the weights respectively for the various criteria.

Step 6. Select the bid with the highest QoB (Quality of Bid) and present it to the Chronobot initiating the bid. Resolve ties appropriately.

Step 7. If the Chronobot initiating the bid agrees to the bid, mark the bid as resolved and inform all other Chronobots appropriately.

Step 8: After the task has been completed, update weights appropriately.

Step 9. If the Chronobot(s) that had proposed the bid defaults or withdraws from the task, repeat steps 1 through 8.

4. Related Work

Although there has been prior art in quantifying knowledge exchange in the context of Natural Language processing (NLP) there has been no prior work on the exchangeability of knowledge and time and in the advocation of such knowledge/time exchange protocols. I summarize the work of three of the best attempts at quantifying knowledge exchange.

In [5], the authors describe a project known as SHADE (SHAred Dependency Engineering). SHADE's approach has three main components: a shared knowledge representation (language and domain-specific vocabulary), protocols supporting information exchange for change notification and subscription, and facilitation services for content-directed routing and intelligent matching of information consumers and producers.

In [6], a protocol called Generic Frame Protocol (GFP) is discussed. It provides a set of Common Lisp functions that provide a generic interface to underlying frame knowledge representation systems (FRSs). The GFP protocol is complementary to language specifications developed to support knowledge sharing. KIF, the Knowledge Interchange Format, provides a declarative language for describing knowledge. The Generic Frame Protocol focuses on operations that are efficiently supported by most FRSs (e.g., operations on frames, slots, facets; inheritance and slot constraint checking).

In [7], the authors discuss the design and use of a shared representation of knowledge (language and vocabulary) to facilitate communication among specialists and their tools and its relationship to current and emerging data exchange standards.

5. Conclusion and Future Work

We have outlined an automata-based formulation of a Time/Knowledge exchange protocol. The protocol interoperates with the various chronobots and co-ordinates the settlement of a bid in a distributed manner and selects the best bid using a metric known as the QoB (Quality of Bid).

Our contributions are twofold. Firstly, we have proposed a time/knowledge exchange system which is unique by itself in terms of its characteristics. Secondly, we have outlined a detailed protocol to perform time/knowledge exchange efficiently in a distributed and coordinated manner.

We are currently in the process of implementing the protocol in a run time

system and testing it in a real-time bidding scenario. A good area of future work or a topic that requires further inspection is that of propagation of bids among the Chronobots themselves. Another issue is that of capturing patterns in the process of time/knowledge exchange. We are currently in the process of investigating it.

References

[1] Shi-Kuo Chang, "A Chronobot for Time and Knowledge Exchange," Proceedings of 2004 Int'l Conference on Distributed Multimedia Systems, September 8-10, Hotel Sofitel, San Francisco Bay, 2004.

[2] Shi-Kuo Chang, "A model for information Exchange," in International Journal of Policy Analysis and Information Systems, Vol. 5, No. 2, 1981.

[3] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," in Proceedings of the IEEE Symposium on Foundations of Computer Science, pp. 256-261, 1989.

[4] M. Herbster and M. Warmuth, "Tracking the best expert," in Proceedings of the Twelfth International Conference on Machine Learning, pp. 286-294, 1995.

[5] James G. McGuire, Daniel R. Kuokka, Jay C. Weber, Jay M. Tenenbaum, Thomas R. Gruber, Gregory R. Olsen, "SHADE: Technology for Knowledge-Based Collaborative Engineering,", Journal of Concurrent Engineering: Applications and Research (CERA), 1(2), September 1993.

[6] Peter D. Karp and Thomas R. Gruber, "A Generic Knowledge-Base Access Protocol," SRI International Technical Report, 1994.

[7] Gregory R. Olsen, Mark Cutkosky, Jay M. Tenenbaum, and Thomas R. Gruber, "Collaborative Engineering based on Knowledge Sharing Agreements," Procedings of the 1994 ASME Database Symposium, September 11-14, 1994, Minneapolis, MN.

An Intelligent Web Information Aggregation System Based upon Intelligent Retrieval, Filtering and Integration

Sheng-Yuan Yang

Department of Electronic Engineering,
St. John's and St. Mary's Institute of Technology
Department of Electronic Engineering,
National Taiwan University of Science and Technology
TAIWAN

Cheng-Seen Ho

Department of Computer Science and Information
Engineering,
National Taiwan University of Science and Technology
TAIWAN

Abstract

This paper introduces FAQ-master as an intelligent Web information aggregation system, which employs intelligent retrieval, filtering and integration techniques to provide high-quality FAQ answers from the Web to meet the user information request. We focus on the issues of how to faithfully capture user intention, facilitate knowledge sharing and reusing, and effectively discover and aggregate unstructured Web information. This design can tackle the following three aspects of Web search activities at the same time: user intention, document processing, and website search. The techniques involved in the design include domain ontology, user models, website models, and data aggregation and proxy mechanisms. Our experiments show: 1) The Interface Agent can correctly understand user intention and focus of up to 80% of the user's queries. 2) The Proxy Agent can share up to 70% of the query loading from the backend process, which can effectively improve the overall query performance. 3) The Answerer Agent has 5 to 20% improvement in precision rate and produces better ranking results. 4) The Search Agent performs very well in obtaining accurate and stable webpages classification, which in turn supports correct annotation of domain semantics to the webpages and helps fast and precise query search with a high degree of user satisfaction.

1. Introduction

The user in the Web age expects to spend shortest time in retrieving really useful information from the Web rather than spending lots of time and ending up with lots of garbage information. Three major problems are behind the bad quality of searching on the Web: no databases of a single search engine can cover the entire Web, the pre-defined database indices are complex, inflexible, and hard-to-use, and the keyword-based interface cannot faithfully capture true user's intention. Three major schools of solutions have appeared in the literature. First, OntoBroker [7], WebKB [4], and SHOE [10] provide content-based search by annotating a document with proper semantics. Second, systems, such as Letizia [16], Amalthea [18], Syskill&Webert [21], Basar [25], Avanti [8], WebWatcher [13], ifWeb

[1], Profile [24], and Personal Webwatcher [17], employ a user-specificity-based filtering system to work as the front end for the search process. Finally, meta-search is used to replace the single search engine approach; examples include GIOSS [9], Metacrawler [23], Savvysearch [6], Profusion [5], NECRI meta-search engine [14] and OySTER [19]. These solutions essentially try to improve the search result from only one of the following three aspects of Web search: *user intention*, *document processing*, or *website search*. In this paper, we propose FAQ-master as an intelligent Web information query system, which can improve search quality from all the whole three aspects.
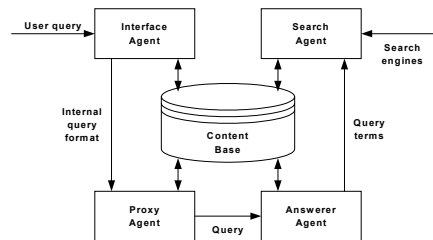

Fig. 1 System architecture of FAQ-master

FAQ-master [11,12] possesses intelligent retrieval, filtering and integration capabilities and can provide high-quality FAQ answers from the Web to meet the user information request. By a high-quality, integrated answer we mean an answer which is profound, up-to-date, and relevant to the user's question. Fig. 1 illustrates the architecture of FAQ-master. It contains four agents supported by a Content Base, which in turn contains a User Model Base, Template Base, Domain Ontology, Website Model Base, Ontological Database, Solution Library, and Rule Base. The Interface Agent captures user intention through an adaptive human-machine interaction interface with the help of ontology-directed and template-based user models [2,31]. It also handles user feedback on the suitability of proposed responses. The Search Agent performs in-time, user-oriented, and domain-related Web information retrieval with the help of ontology-supported website models [26,27,28]. The Answerer Agent works as a back end process to perform ontology-directed information aggregation

from the webpages collected by the Search Agent [3,29]. Finally, the Proxy Agent works as an ontology-enhanced intelligent proxy mechanism, which shares most query loading with the Answerer Agent [15,30]. From the viewpoint of FAQ service, this architecture can collect most useful FAQ files, preprocess the files into consistent data, aggregate FAQ answers according to the user query and his user model, and provide user-oriented FAQ answers through a quick proxy mechanism.
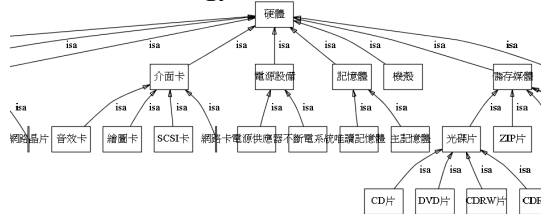
2. Domain Ontology

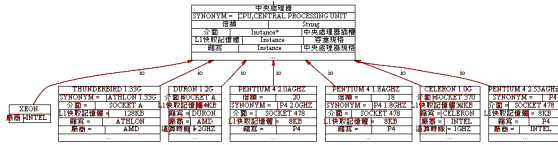

Fig. 2 Part of PC ontology taxonomy

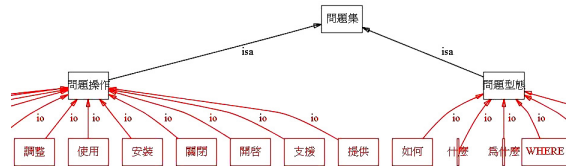Fig. 3 Ontology for the concept of "中央處理器" (CPU)

Fig. 4 Part of problem ontology taxonomy

The most key background knowledge of the system is domain ontology, which is developed for the PC domain using Protégé 2000 [20]. Fig. 2 shows part of the ontology taxonomy. The taxonomy represents relevant PC concepts as classes and their parent-child relationships as *isa* links, which allow inheritance of features from parent classes to child classes. We carefully selected the properties that are most related to our application from each concept and defined them as the detailed ontology for the corresponding class. Fig. 3 exemplifies the detailed ontology for the concept of "中央處理器" (CPU). In the figure, the root node uses various fields to define the semantics of the CPU class, each field representing an attribute of "CPU", e.g., interface, provider, synonym, etc. The nodes at the lower level represent various CPU instances, which capture real world data. The complete PC ontology can be referenced from the Protégé Ontology Library at Stanford Website (http://protege.stanford.edu/ontologies.html). We also developed a problem ontology to deal with query questions. Fig. 4 illustrates part of the Problem ontology, which contains "問題型態" (question type) and "問題操作" (question operation). Together they imply the semantics of a question. Finally, we use Protégé's APIs to develop a set of ontology services,

which provide primitive functions to support the application of the ontologies. The ontology services currently available include transforming query terms into canonical ontology terms, finding definitions of specific terms in ontology, finding relationships among terms, finding compatible or conflicting terms against a specific term, etc.

3. Interface Agent with Ontology-Supported User Models

A user model contains *interaction preference*, *solution presentation*, *domain proficiency*, *terminology table*, *query history*, *selection history*, and *user feedback* (Fig. 5). The interaction preference is responsible for recording user's preferred interface, e.g., favorite query mode, favorite recommendation mode, etc. When the user logs on the system, the system can select a proper user interface according to his preference. We provide two modes, either through keywords or natural language input. We provide three recommendation modes according to hit rates, hot topics, or collaborative learning. We record recent user's preferences in a time window, and accordingly determine the next interaction style. The solution presentation is responsible for recording solution ranking preferences of the user. We provide two types of ranking, either according to the degree of similarity between the proposed solutions and the user query, or according to user's proficiency about the solutions. In addition, we use a Show_Rate parameter to control how many items of solutions for display each time, in order to reduce information overloading problem. The domain proficiency factor describes how familiar the user is with the domain. By associating a proficiency degree with each ontology concept, we can construct a table, which contains a set of <concept proficiency-degree> pairs, as his domain proficiency. Thus, during the decision of solution representation, we can calculate the user's proficiency degree on solutions using the table, and accordingly only show to the user his most familiar part of solutions and hide the rest for advanced requests. To solve the problem of different terminologies to be used by different users, we include a terminology table to record this terminology difference. We then use the table to replace the terms used in the proposed solutions with user favorite terms during solutions representation to help the user better comprehend the solutions. Finally, we record the user's query history as well as FAQ selection history with corresponding user feedback in each query session in the interaction history, in order to support collaborative recommendation. The user feedback is a complicated factor. We remember both explicit user feedback in the selection history and implicit user feedback, which includes query time, time of FAQ click, sequence of FAQ clicks, sequence of clicked hyperlinks, etc.

In order to quickly build an initial user model for a new user, we pre-defined five user stereotypes [22], namely, expert, senior, junior, novice, and amateur [11,32], to represent different user group's characteristics. This approach is based on the idea that

the same group of user tends to exhibit the same behavior and requires the same information. Fig. 5 illustrates an example user stereotype. When a new user enters the system, he is asked to complete a questionnaire, which is used by the system to determine his domain proficiency, and accordingly select a user stereotype to generate an initial user model for him. However, the initial user model constructed from the stereotype may be too generic or imprecise. It is then subjected to refinement to reflect the specific user's real intent after the system has experiences with his query history, FAQ-selection history and feedback, and implicit feedback.
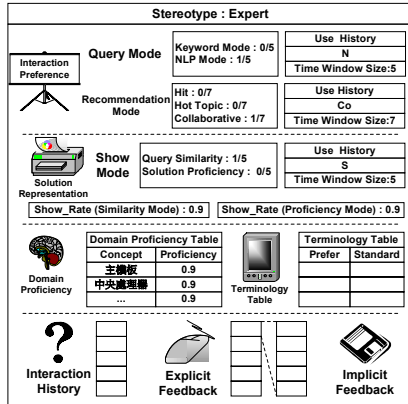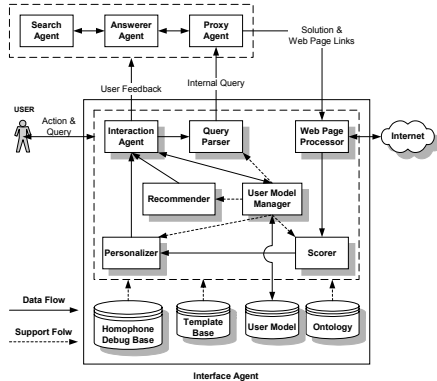

Fig. 5 Example of expert stereotype

Fig. 6 Interface Agent architecture

Fig. 6 shows the architecture of the Interface Agent. The Interaction Agent provides a personalized interaction, assistance, and recommendation interface for the user according to his user model, records interaction information and related feedback in the user model, and helps the User Model Manager and Proxy Agent to update the user model. The Query Parser processes the user queries by segmenting words, removing conflicting words, and standardizing terms, followed by recording user's terminologies in the terminology table of the user model. It finally applies the template matching technique [31] to select best-matched query templates, and accordingly transforms the query into an internal query for the Proxy Agent. The Proxy Agent then searches for solutions and collects them into a list of FAQs, each containing a corresponding URL. The Web Page Processor pre-downloads FAQ-relevant webpages and

performs such pre-processing tasks as labeling keywords for subsequent processing. The Scorer calculates the user's proficiency degree for each FAQ in the FAQ list according to the terminology table in the user model. The Personalizer then produces personalized query solutions according to the terminology table. The User Model Manager is responsible for quickly building an initial user model for a new user using the technique of user stereotyping as well as updating the user models and stereotypes to dynamically reflect the changes of user behavior. The Recommender is responsible for recommending information for the user based on hit count, hot topics, or group's interests when a similar interaction history is detected.
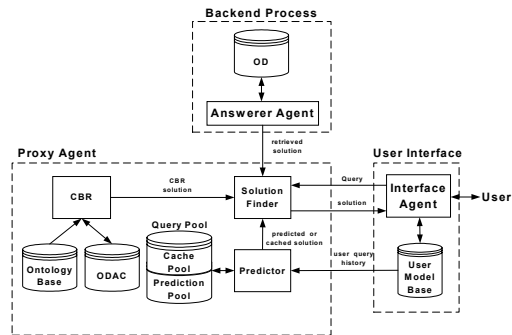
4. Ontology-Supported Proxy Agent

Fig. 7 Proxy agent architecture

The Proxy Agent performs solution finding by a two-tier architecture [30]. The first tier predicts queries and caches possible solutions. The second tier employs case-based reasoning to derive solutions for the given user queries. Fig. 7 illustrates the architecture of the Proxy Agent and shows how it interacts with the User Interface and Backend Process. First, the Interface Agent collects queries from the user according to his user model. The user queries history can be used by the Predictor to do query prediction and query cache services. The Answerer Agent manages OD (Ontological Database), which stores preprocessed Q-A pairs collected from the Web, and retrieves proper Q-A pairs to respond to user queries. Inside the Proxy Agent, the Ontological Database Access Cases (ODAC) is the case base for CBR, which stored all of user query cases. From the viewpoint of cognitive science, ODAC accumulates the system experience of user query responses. And it will incrementally extend its coverage of the domain knowledge after the system accumulates more past cases. The sources of the ODAC are the new cases aggregated by the Answerer Agent or the adapted cases through case adaptation. The Predictor is responsible for tracking user query history for storing most often occurring queries in the Cache Pool, and predicting next possible queries for storage in the Prediction Pool. With these two pools, together called Query Pool, it can provide services of query cache and query prediction as the first-tier proxy mechanism to reduce query response time. The CBR serves as the second-tier proxy by employing the

case-based reasoning technique to reason about solutions for a given query from the ODAC. If there exists a case in the ODAC that is exactly the same as the user query, the CBR directly outputs it as the solution to the user. If only similar cases to the user query exist in the ODAC, the CBR adapts the solutions for the user through a case adaptation process. The CBR is also responsible for the maintenance of cases in the ODAC by evaluating and determining whether the adapted Q-A pair deserves storage in the ODAC to serve as a new case according to the user's feedback on the solution. The Solution Finder is the central control in finding solutions. Given a user query, it first checks with the Predictor for any possible cached or predicted solutions. If none exits, it invokes CBR to retrieve or adapt old solutions. If still no solution produced, it finally passes the query to the Backend Process, asking the Answerer Agent to aggregate a solution from OD.

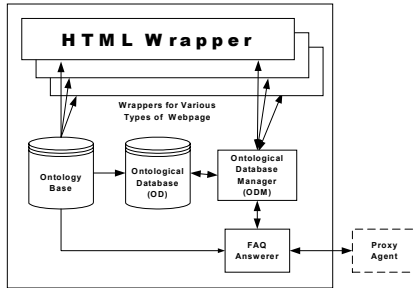## 5. Ontology-Supported Answerer Agent

Fig. 8 Answerer Agent architecture

Fig. 8 illustrates the architecture of Answerer Agent. The Ontological Database (OD) is a stored structure designed according to the ontology structure, serving as an canonical format for storing FAQ information. The HTML wrapper performs parsing, extracting and transforming of Q-A pairs on each Web page into the canonical format for the Ontological Database Manager (ODM) to store in OD. Finally, the FAQ Answerer is responsible for retrieving the best matched Q-A pairs from OD, deleting any conflicting Q-A pairs, and ranking the results according to the match degrees for the user.

Given an internal representation of a user query, the FAQ Answerer can easily transform it into a SQL statement. Fig. 9 shows the transformed SQL statement. Note that we include all the keywords in the query conditions. This is called the full keywords match method. In this method, the FAQ Answerer retrieves only those Q-A pairs whose question part contains all the user query keywords from OD as candidate outputs. If none of Q-A pairs can be located, the Answerer then turns to the partial keywords match method to find solutions. In this method, we select the best half number of query keywords according to their TFIDF values and use them to retrieve a set of FAQs from OD. We then check the retrieved FAQs for any conflict with the user query keywords by submitting the unmatched keywords to proper ontology services, which check for any semantic conflict. Only those FAQs that are proved consistent with the user intention by the ontology are

retained for ranking. We finally apply different ranking methods to rank the retrieval results according to whether full keywords match or partial keywords match is applied. Factors involved in the ranking processes contain keyword's appearance probability, user's satisfaction value, FAQ's statistic similarity value, and keyword's compatibility value [29].

```
Select      *
From   是否
Where  動作＝'支援' AND
       問句關鍵詞 like '% 1GHZ % K7V % 中央處理器%'
```
Fig. 9 Example of transformed SQL statement

6. Search Agent with Ontology-Supported Website Models

Fig. 10(a) illustrates the format of a website model, which contains a website profile and a set of webpage profiles. A website profile contains statistics information about a website. The webpage profile contains three sections, namely, basic information, statistics information, and ontology information. The first two sections profile a webpage and the last annotates domain semantics to the webpage.

(a) Format of a website model

(b) Conceptual structure of a website model
Fig. 10 Website model format and structure

Each field in the basic information section is explained below. *DocNo* is automatically generated by the system for identifying a webpage in the structure index. *Location* remembers the path of the stored version of the Web page in the website model; we can use it to answer user queries. *URL* is the path of the webpage on the Internet, the same as the returned URL index in the user query result; it helps hyperlinks

analysis. *WebType* identifies one of the following six Web types: com (1), net (2), edu (3), gov (4), org (5), and other (0), each encoded as an integer in the parentheses. *WebNo* identifies the website which contains this webpage. It is set to zero if we cannot decide what website the webpage comes from. *Update_Time/Date* remembers when the webpage was modified last time. The statistics information section stores statistics about HTML tag properties, e.g., *#Frame* for the number of frames, *#Tag* for the number of different tags, and various texts enclosed in tags. Specifically, we remember the texts associated with *Titles*, *Anchor*s, and *Headings* for webpage analysis; we also record *Outbound_URLs* for user-oriented webpage expansion. Finally, the ontology information section remembers how the webpage is interpreted by the domain ontology. It shows that a webpage can be classified into several classes with different scores of belief according to the ontology. It also remembers the ontology features of each class that appear in the webpage along with their term frequencies (i.e., number of appearance in the webpage). *Domain_Mark* is used to remember whether the webpage belongs to a specific domain; it is set to "true" if the webpage belongs to the domain, and "false" otherwise. This section annotates how a webpage is related to the domain and can serve as its semantics, which helps a lot in correct retrieval of webpages.

Let's turn to the website profile. *WebNo* identifies a website, the same as used in the webpage profile. Through this number, we can access those webpage profiles which describe the webpages belonging to this website. *Website_Title* remembers the text between tags <TITLE> of the homepage of the website. *Start_URL* stores the starting address of the website. It may be a domain name or a directory URL under the domain address. *WebType* identifies one of the six Web types as used in the webpage profile. *Tree_Level_Limit* remembers how the website is structured, which can keep the search agent from exploring too deeply; e.g., a value of 5 means it explores down to level 5 of the website structure. *Update_Time/Date* remembers when the website was modified last time. This model structure helps interpret the semantics of a website through the gathered information; it also helps fast retrieval of webpage information and autonomous Web resources search, as shown in Fig. 10(b).

Fig. 11 shows the architecture of Search Agent. Focused Crawler is responsible for gathering webpages into DocPool according to user interests and website model weakness. Model Constructor extracts important information from a webpage stored in DocPool and annotates proper ontology information to make a webpage profile for it. It also constructs a website profile for each website in due time according to what webpages it contain [28]. Webpage Retrieval uses ontology features in a given user query to fast locate and rank a set of most needed webpages in the website models and displays it to the user. Finally, Query Interface receives the user query, expands the query using ontology, and sends it to Webpage Retrieval,

which will return a list of ranked webpages. Note that the query is transformed into a list of keywords by Interface Agent for query expansion by this component.
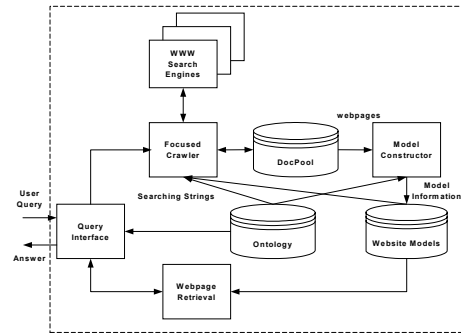

Fig. 11 Search Agent architecture

7. Conclusions

We have described FAQ-master as an intelligent Web information integration system. It is equipped with techniques of domain ontology, user models, website models, solution aggregation and solution proxy to support effective discovery, retrieval, and integration of high-quality information from a distributed environment to meet the user information request. It was properly designed to tackle the following three aspects of Web search activities at the same time: user intention, document processing, and website search.

We have completed the implementation of a prototype of FAQ-master. It includes four agents. First, The Interface Agent work as an assistant between the users and the system for capturing true user's intention and accordingly providing high-quality FAQ answers. Second, the Proxy Agent works as a two-tier mediator between the Interface Agent and the backend Answerer Agent. It employs an ontology-enhanced intelligent proxy mechanism to effectively alleviate the overloading problem usually associated with a backend server. Third, the Answerer Agent employs the wrapper technique to help clean, retrieve, and transform FAQ information collected from a heterogeneous environment, such as the Web, and stores it in an ontological database. It works as a back end process to perform ontology-directed information storage and aggregation from the webpages collected by the Search Agent. Finally, the Search Agent performs in-time and domain-related Web information retrieval with the help of ontology-supported website models to provide a semantic level solution so that it can provide fast, precise and stable search results with a high degree of user satisfaction.

Our experiments show: 1) The Interface Agent can correctly understand user intention and focus of up to 80% of the user's queries. 2) The Proxy Agent can share up to 70% of the query loading from the backend process, which can effectively improve the overall query performance. 3) The Answerer Agent has 5 to 20% improvement in precision rate and produces better ranking results. 4) The Search Agent performs very well in obtaining accurate and stable webpages classification, which in turn supports correct annotation

of domain semantics to the webpages and helps fast and precise query search with a high degree of user satisfaction.

Acknowledgment

References

[1] F.A. Asnicar and C. Tasso, "if Web: A Prototype of User Model-Based Intelligent Agent for Document Filtering and Navigation in World Wide Web," *Proc. of the 6th International Conference on User Modeling (UM-97)*, Chia Laguna, Sardinia, Italy, 1997, 3-12.

[2] Y.H. Chiu, An Interface Agent with Ontology-Supported User Models, Master Thesis, Dept. of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, 2003.

[3] F.C. Chuang, Ontology-Supported and Ranking Technique-Enhanced FAQ Systems, Master Thesis, Dept. of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, 2003.

[4] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigram and S. Slattery, "Learning to Extract Symbolic Knowledge from the World Wide Web," *Proc. of the 15th National Conference on Artificial Intelligence (AAAI-98)*, Madison, Wisconsin, 1998, 509-516.

[5] DesingLab at University of Kansas, "ProFusion meta-search," Available at http://www.designlab.ukans.edu/profusion/, 1997.

[6] D. Dreilinger and A.E. Howe, "Experiences with Selecting Search Engines Using Meta-Search," Available at http://www.media.mit.edu/~daniel, 1996.

[7] D. Fensel, M. Erdmann, and R. Studer, "OntoBroker: How to Make the WWW Intelligent," *Proc. of the 11th Banff Knowledge Acquisition for Knowledge-Based Systems, Workshop (KAW-98)*, Banff, Canada, April 1998, Also available at http://ksi.cpsc.ucalgary.ca/KAW/KAW98/KAW98Proc.html.

[8] J. Fink, A. Kobsa, and A. Nill, "User-oriented Adaptivity and Adaptability in the AVANTI Project," *Proc. of the International Conference on Design for the Web: Empirical Studies*, Microsoft Usability Group, Redmond, WA, 1996, Also available at http://www.microsoft.com/usability/webconf.htm.

[9] L. Gravano, H. García-Molina, and A. Tomasic, "Precision and Recall of GIOSS Estimators for Database Discovery," *Proc. of the 3rd International Conference on Parallel and Distributed Information Systems (PDIS'94)*, Austin, Texas, USA, 1994, 103-106.

[10] J. Heflin, J. Hendler, and S. Luke, "Applying Ontology to the Web: A Case Study," *Proc. of the 5th International Work-Conference on Artificial and Natural Neural Networks (IWANN'99)*, Alicante, Spain, 1999, 715-724.

[11] C.S. Ho, Research on FAQ-master as an Intelligent high-quality Information Integration Agent for PC-DIY, National Science Council Project, NSC-89-2213-E-011-059, Taipei, Taiwan, 2000.

[12] C.S. Ho, An Intelligent Web Information Integration System Based upon Intelligent Retrieval, Filtering, and Integration, National Science Council Project, NSC-89-2218-E-011-014, Taipei, Taiwan, 2001.

[13] T. Joachims, D. Freitag, and T. Mitchell, "WebWatcher: A Tour Guide for the World Wide Web," *Proc. of the 1997 International Joint Conference on Artificial Intelligence (IJCAI'97)*, Nagoya, Japan, 1997, 770-775.

[14] S. Lawrence and L. Giles, "Context and Page Analysis for Improved Web Search," *IEEE Internet Computing*, 2(4), 1998, 38-46.

[15] P.C. Liao, An Intelligent Proxy Agent for FAQ, Master Thesis, Dept. of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, 2003.

[16] H. Lieberman, "Letizia: An Agent that Assist Web Browsing," *Proc. of the 1995 International Joint Conference on Artificial Intelligence (IJCAI'95)*, Montreal, Canada, 1995, 924-929.

[17] D. Mladenic, "Machine Learning Used by Personal WebWatcher," *Proc. of ACAI-99 Workshop on Machine Learning and Intelligent Agents*, Chania, Crete, 1999, Also available at http://www.cs.cmu.edu/~TextLearning/pww/pww.html.

[18] A. Moukas, "Amalthea: Information Discovery and Filtering Using a Multiagent Evolving Ecosystem," *Proc. of the 1st International Conference on the Practical Applications of Intelligent Agents and MultiAgent Technology (PAAM)*, London, April 1996 (Available at http://moux.www.media.mit.edu/people/moux/papers/PAAM96/)

[19] M. Müller, "An Intelligent Multi-Agent Architecture for Information Retrieval from the Internet," Technical report, U. of Osnabrüeck, Germany, Available at http://mir.cl-ki,uni-osnabrueck.de/~martin/index.publications.2.h tml, 1999.

[20] N.F. Noy and D. L. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology," Available at http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial -noy-mcguinness.pdf, 2000.

[21] M. Pazzani, J. Muramatsu, and D. Billsus, "Syskill&Webert: Identifying Interesting Web Sites," *Proc. of the 13th National Conference on Artificial Intelligence and 8th Innovative Applications of Artificial Intelligence Conference (AAAI 96, IAAI 96)*, Portland, OR, 1996, 54-61.

[22] E. Rich, "Stereotypes and User Modeling," in User Models in Dialogue Systems, A. Kobsa and W. Wshlster (eds.), Springer-Verlag, 1989, 35-51.

[23] E. Selberg and O. Etzioni, "Multi-service Search and Comparison Using the MetaCrawler," *Proc. of the 4th International World Wide Web Conference*, Boston, MA, USA, 1995, 169-173.

[24] J. Simons, "Using a Semantic User Model to Filter the World Wide Web Proactively," *The 6th International Conference on User Modeling (UM'97)*, Chia Laguna, Sardinia, Italy, 1997, 455-456.

[25] C.G. Thomas and G. Fischer, "Using Agents to Improve the Usability and Usefulness of the World-Wide Web," *Proc. of the 5th International Conference in User Modeling (UM'96)*, West Newton, 1996, 5-12.

[26] C..M. Wang, Web Search with Ontology-Supported Technology, Master Thesis, Dept. of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, 2003.

[27] S.Y. Yang and C.S. Ho, "A Website-Model-Supported New Search Agent," *The 2nd International Workshop on Mobile Systems, E-Commerce, and Agent Technology (MSEAT'03)*, Miami, Florida, USA, 2003, 563-568.

[28] S.Y. Yang, C.M. Wang, and C.S. Ho, "How Do Ontology-Supported Website Models Help Web Search?" Submitted to International Journal on Artificial Intelligence Communications, 2003.

[29] S.Y. Yang, F.C. Chuang, and C.S. Ho, "An FAQ System Empowered by Ontology and Improved Ranking Techniques," Submitted to International Journal of Intelligent Information Systems, 2004.

[30] S.Y. Yang, P.C. Liao, and C.S. Ho, "An Intelligent Two-Tier Proxy Agent for FAQ Service," Submitted to International Journal on Data and Knowledge Engineering, 2004.

[31] S.Y. Yang, Y.H. Chiu, and C.S. Ho, "Ontology-Supported and Query Template-Based User Modeling in Interface Agents," Submitted to International Journal of Human-Computer studies, 2004.

[32] S.Y. Yang and C.S. Ho, "Ontology-Supported User Models for Interface Agents," *Proc. of the 4th Conference on Artificial Intelligence and Applications (TAAI'99)*, ChangHua, Taiwan, 1999, 248-253.

An Efficient Process and Strategy to Upgrade Traditional Copy Center

Huay Chang[1][2], Shieh-Shing Lin[3], Yong Su[1], Feng-Jih Wu[3]

[1]: Department of Business Administration, Management School, Fu Dan University
[2]: Department of Information Management, Chihlee Institute of Technology
[3]: Department of Electrical Engineering St. John's & St. Mary's Institute of Technology
cecilia2chang@yahoo.com.tw[1][2]

ABSTRACT

The business division of Ministry of Economics Affairs(MOE) has submitted "Business Automation Ten-Year Plan" in 1991 and "Business Automation and Electric Promotion Plan（07/1999-12/2004)"in 1999. In America, a "Quick Response Plan" was submitted in 1986 and "Efficient Consumer Response Plan" was submitted in 1992. Based on new technical management method and business modern management mode, the traditional copy centers in Taiwan should realize to upgrade now.

In this paper we made an analysis in five phases: "Requirements Phase、Supply Phase、System Phase、Integration Phase and Staff Phase" The results of this research are four upgrading models: i. The Model of Modern Management Operation ii. The Model of Customer Service iii. The Model of Integration Goods Distribution iv. The Model of Employees' High-Tech Training

Key words: Business Automation, Business Modernization, Business Upgrade, Consumer Service Quality

1. INTRODUCTION

Facing the e-commerce, many consumers operate computers as their daily tools. The owners of the traditional copy centers feel that there are many new competitors arise from the industries [1].

We have a high interesting in how to upgrade the traditional copy centers. We plan to make a series of stages to help the traditional copy centers achieving the goal of upgrade [2].

The MOE has planned the 'investment deduction' and 'upgrade grading' plan within two series Business Modernization Plans. The Business Division is in charge of the education training and skill instruction issues. Indeed, this is a good chance for the business to implement reorganization and upgrade.

2. RESEARCH MOTIVE AND PURPOSE

2.1 Research Motive

The motive of this study is concluded in the following phases: Requirement Phase, Supply Phase, System Phase, Integration Phase and Personnel Phase.

2.1.1 Requirement Phase

How to increase the quality of consumer service? How to increase the suppliers' service speed and quality of the traditional copy center?

2.1.2 Supply Phase

In order to establish the fast and convenient service, the industry of copy center should establish its competidence strength [3].

2.1.3 System Phase

The internal standardization paper forms flow should be designed first. Then, the information management and statistical analysis would make the increase in management quality [4].

2.1.4 Integration Phase

Due to the lack of integration planning of all the suppliers around the traditional copy center, it appears the less control of information transferring and goods delivery.

2.1.5 Personnel Phase

There is an increase in regulating consumer protection rules annually. Therefore, the consumers' emphasis on their personal right runs day by day.

2.2 Research Purpose

The purpose of this search is:

i. A research in the upgrade of the traditional copy center's flow of the overall industry environment?

ii. A research in the upgrade of the traditional copy center's flow of the consumers' service operation?

iii. A research in the upgrade of the traditional copy center's flow of the suppers' operation?

iv. A research in the upgrade of the internal information management operation of the traditional copy center's ?

v. A research in the upgrade of the traditional copy center's flow of the traditional operation management model?

3. LITERATURE REVIEW

The MOE has created the "The Industry Automation Plan of the Republic of China" [5]. "Business Automation Promotion Plan (7/1991- 6/2000)" is one of the sub-plans. The five major sub-plans are: Information Flow Standardi- zation, Product Sale Automation, Product Selection Automation, Product Flow Automation, and Accountancy Booking Standardization. The research targets are: selected retailers, distribution centers and consumers. The planned goals are: Economic Benefit, Social Benefit and Environmental Benefit.

There is a good evaluation record while promotion business automation under the construction of the Business Division of the MOE. For example, there is a consumer's service training in producing product bar code. In order to protect the right of purchasing good products for the consumers, the distribution centers are designed. It transforms various types of ordering forms through computer network technique within companies. And the EDI (Electronic Data Interchange) and EOS (Elec- tronic Ordering System) ordering standards created by the government are submitted to the business owner.

At the end of "Business Automation Promotion Plan", the Business Division of the MOE implemented the second stage's 'Business Automation and Electrical Promotion Plan (07/1999-12/2004)'. This plan inherits all the strengths from the former ten-year plan. This plan aims at the following business: 3C product, grocery product, drug & cosmetic, book & video entertainment, customized clothes and tourism service.

The traditional copy center is not included in the second stage. There is no any research in the field of upgrading the traditional copy center. Based on the results of the former Business Automation Plan, we would like to make a research in upgrading the traditional copy center [6].

In order to integrate the overall information around business, the product channel between traditional copy center and the supplier should be planned cautiously. Therefore, the 'EOS [7]' and 'EDI [8]' ordering stan - dardization models resulted from the first stage 'Business Automation Ten-Year Plan' could be involved in our study. The field of e-commerce is one of the categories in our study, too. And the owners of the traditional copy center should pay more attention in the security of the e-commerce payment method [9].

4. RESEARCH METHODOLOGY

4.1 Research Method

4.1.1 Group Interview Method

4.1.2 Questionnaire Investigation Method

4.2 Research Design

4.2.1 Select Sample

The selection of the samples is determined by the urgent degree of the upgrading need.

4.2.2 Research Scope

i. An understanding of the industry environment of the traditional copy center.

ii. An understanding of the social requirements of the traditional copy center.

iii. An understanding of the operation management status of the traditional copy center.

iv. An understanding of the goals and potential difficulties of the traditional copy center.

v. An understanding of the owners' background and the operation items of the traditional copy center.

vi. An understanding of the status of the information development of the traditional copy center.

vii. An understanding of the status of the employees' information techniques of the traditional copy center.

viii. An understanding of the possibilities of the multi-types e-commerce of the traditional copy center.

5. RESEARCH CONCLUSIONS

After further research, an efficient process model and strategy are submitted. This efficient model consists of four sub-models. The detailed introduction is listed below.

5.1 The Model of Modern Management Operation

5.1.1 'Functional Area' of Internal Environment

While the traditional copy center upgrades, there is a function distinction in this office (Fig. 5-1). The detailed descrip- tion of each function area is:

i. Copying Area:

Different types of copy machines are located for the multiple-use of the consumers. These machines can be used by the consumers themselves.

ii. Delivery Channel:

While at the 'service -free' time, the consumers put their documents on the delivery channel. The documents will be delivered to the back-desk, the copy schedule and the number of the pick-up desk will be made next day.

iii. Pick-Up Box:

The pick-up box is just like the mail-box in the post office. The consumers can pick up their original docu-ments and the copied papers in this box.

iv. Computer Console Room:

v. Storage Room:

The storage room is used to storage all kinds of the goods and materials used in the store.

vi. Binder Area:

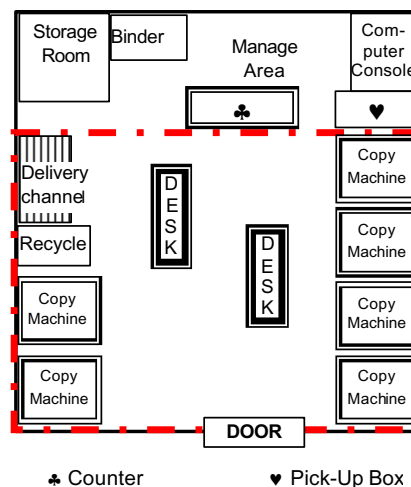The various types of binder are needed by the consumers. All the binder equipments are placed here.

Fig. 5-1、The Functional Area of the Upgrading Modern Copy Center

vii. Counter:

The cashier is needed in commercial stores.

viii. Customer Area:

The consumers are allowed to use all the equipments and the service in this area.

5.1.2 'Factory System' Concept

The concept of the 'Factory System' is: in order not to invest huge amount in each branch store [10], there is the max-function equipments planning in the master store. The general-function copy machines are located in the branch stores. In order to provide high-quality service in the master store, a 'high-quality copy channel' is designed in each branch.

The digital files of the consumers' can be transferred to the master store via the network. The papers are delivered to the consumers by the distri- bution system. There is fair quality service for the consumers.

5.1.3 'Digital Network Copy Center' Positioning

After finished upgrading, the traditional copy centers turns to be a 'Digital Network Copy Center'.

The consumers can transfer their digital files to the copy center wherever they are at home or at the office. After finished copying, the papers are delivered to the consumers by the distribution system [11].

5.1.4 'Service-Free Copy Center' Operation Model

People are more and more busy today. We would like to transform the copy center into the operation model of 'Service-Free Copy Center' at night. There is no any employee in this copy center at night. All the copy equipments are used by the consumers themselves. In order to protect the consumers' safety and the properties of the copy center, the overall safety management system is established at night.

Under this kind of operation model, the following items should be re-organized and re-evaluated: internal flow, employees, information techniques, inventory control, payment method, overall security control.

5.1.5 'POS (Point of Sales) System' Payment Model

The 'POS System' Payment Model is consisted of the goods bar-code, cash machines and the related computer hardware and software. All the information collected by the POS system will be turned to the manager for adjusting the operation management policy.

While using the 'POS System', the hardware and software used in the front-desk and back-desk are:

i. Front-desk: POS cashiers, bard-code scanners, POS front-desk saling system.

ii. Back-desk: Computer equipments, printer, bar-code printer, network equipments, modems, phones, fax machines, POS back-desk sale system.

5.1.6 'Overall Security' Management Model

In order to protect the consumers' safety and the properties of the copy center, the overall safety management system is established at night.

First, the door-security system is established. The video-cameras are located at the suitable place. There is

an alarm system that is connected to the police station.

5.1.7 'Overall Computer Framework' of Copy Center

The computer assisted equipments are used among the copy centers, the suppliers and the consumers. 'The Overall Business Environment Computerization Diagram' is designed (Fig. 5-2).

Fig. 5-2 Overall Computer Framework
of Copy Center

5.2 The Model of Customer Service

5.2.1 'ID Code' Service Model

After the upgrade, the '24 Hour Automatic Copy Center' is created. It is quite necessary to design an 'ID Code' to verify the consumer's identifications [12]. The function of the 'ID Code' is used while transferring the network information and the various kinds of use of the IC card. There is a different 'ID Code' for each consumer.

5.2.2 'E-Store' Operation Model

After upgrade, the consumers can transfer the digital files, query the copy schedule and receive the discount service notification via the 'e-Store' operation model.

5.2.3 'IC Card' Operation

The 'IC Card' is an important ID verification for the consumers while using the modernization equipments in the copy center [13]. The functions of the IC Cards are:

i. Door-entry control card

ii. Customer personal data

iii. Customer transaction records

iv. Entry records at 'service-free' time

v. Copied paper amount records

vi. Binder times records at 'service-free' time

5.3 The Model of Integration Goods Ordering Distribution

The daily ordering operation and inventory control are not processed by the computer in the traditional copy center. A lot of mistakes occurred, such as: the shortage

of inventory, the over loading of goods, the mistake of goods ordering, etc.

Based on the 'EDI' program and the 'EOS' program, presented by the Business Division of the Ministry of Economics Affairs, the ordering forms can be made through the network. This type of ordering system not only speeds up the distributing system, but also reduces the goods storage cost. (Fig. 5-3)
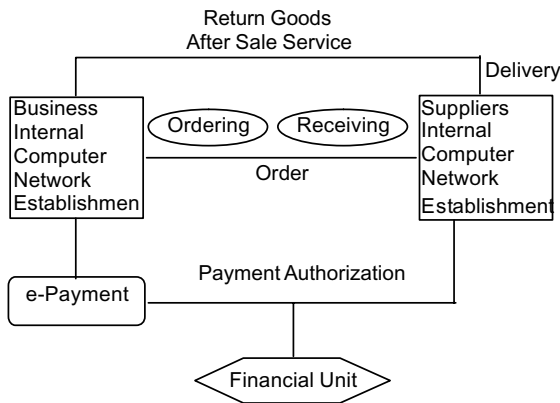

Fig. 5-3 The Diagram of Ordering Flow

5.4. The Model of Employees'High-Tech Training

5.4.1 Basic Skills Training

After upgrade, the employees need to be involved in the operation management with the information technique equipments. Therefore, there are new basic skill require-ments: basic computer operation, web-site planning and copy machine operation.

i. Basic Computer Operation and Web-Site Plan

The related copy equipments are operated by the information digital machines after upgrade. The increase of the result and quality of the internal operation management issues within the copy centers also rely on the computer equipments.

ii. Operating Skills of Copy Equipments

The copy machines own not only the digital copy function, but also the computer and fax function. Therefore, the employees must learn to operate the new digital copy machines.

5.4.2 Employees' Duty Assignments

The employees' duty assignments should be arranged while upgrading the copy centers. These assign-ments are:

i. Front-Desk Operation

The main mission of the front-desk employee is to provide the fast and complete service. The employees must have a clear understanding of all kinds of the operation in the copy center.

ii. Back-Desk Operation

The mission of the back-desk employees is to provide the support of the front-desk operation. They must use computers to have a fully control of the inventory in time.

6. RESULT BENEFITS

The befits of the upgrade of the copy center are:

i. Increase the consumer service quality.

ii. Establish related industries promotion chains.

iii. Increase the labor productivity of this industry by using automatic technical equipments.

iii. Lower the labor cost of this industry by using automatic technical equipments.

iv. Establish "Copy Center" new image by multi-service policy.

7. REFERENCE

[1] The Service Web of Uniersal Information Web of the Business Division of the Ministry of Economics Affairs，"http://www.moea.gov.tw"

[2] Chou-Rung Jwung, "Emphasizing the Problems of the Upgrade of Traditional Industries", *Monthly Journal of Taiwan Economic Research*, pp.73-76, June, 1996.

[3] Ren-Jey Liu, "Re-create New Industry Competition Strength-The Upgrade and Change of Taiwan Indus-tries", *The Journal of Jung-Wei Information*, pp.82-89.

[4] Ming-Ling Shywe and Jung-Ming Gwo, "The Analysis and Business Strategies Toward New Advised Industry Upgrade Promotion Principles", Monthly Journal of Accounting, pp.72-79, Feb. 2000.

[5] Ming-Bon Chen, "The Problems of the Protection of the Developing e-Commerce Wisdon Property", *The Journal of Wisdom Property*, pp.1-15, March 2000.

[6] Chung-Shung Fu, "The Introduction of the Tax of the e-Commerce Transaction", *The Journal of Tax*, pp.14-17, July 1999.

[7] Yung-Long Chen, "Hot to Involve the e-Commerce by Low-Cost", *The Journal of Information and Computers*, pp.52-55, April 1999.

[8] The Business Division of the Ministry of Economics Affairs, "*The ROC Industry Automation Plan and the Business Automation Promotion Plan (July 1991~ June. 2000)*", April 1995.

[9] Wen-Shyung Wu and Jung-Pu Ho, "The Customer Satisfaction Model in Internal Network-The Practical Research in Shopping Web", *the 1st 2000 year National Business Modernization Conference*, pp.1-10, March 2000.

[10] The Statistics Division of the Ministry of Economics Affairs, "*The Index of Economics*", Jan. 2000.

[11] The Business Division of the MOE, "*The Standard of Business Information Dis-tributing*"

[12] The Business Division of the Ministry of Economics Affairs, "*The National Electric Ordering and Net-work System Implementing Skills*", Oct. 1995.

[13] The Business Division of the Ministry of Economics Affairs, "*The National business Electric Data Inter-change Value-Added Network System*", Oct. 1995.

The Applications of IA for the Emergency Systems

Dong-liang Lee*, Chin-Yung Yu**, Ching Hsu Chan*, and, Lawrence Y. Deng*

*St. John's & St. Mary's Institute of Technology, Taipei, Taiwan,

**Department of Information Engineering, Tamkang University, Taipei, Taiwan

lianglee@mail.sjsmit.edu.tw

Abstract

The information appliance (IA) is used to control the electronic consumer products. But the specifications of the digital devices are defined to the individual company. That causes to the majority of IA devices do not interconnect the owner functions of each other. The main purposes of this paper will issue an efficient architecture to integrate the difference IA devices. The proposed architecture is exploited the MIB tree structure to define the IA devices. In order to implement the purposes of data exchanging and integration, we also use the Agents software to construct a group-relation to solve the event of IA devices.

Keywords: Information appliances, agent, integrate, and LONWORK.

1. Introduction

For a usable IA device in which the functions are not only suitable to individual person need, but also efficient integer the information each other. In the recently time, the width-band is applied to the network, the more data and information, such as text, image, voice, graph, etc., is access via Internet. Thus, the IA devices are not only used as equipments of household, but also offer all kinds of information services.

In order to improve the performance of the IA devices, we must integrate the difference specifications of IA of the individual companies. Thus, we need to construct a platform to exchange the needed data and information of each IA device. In this paper, we will offer an integrating architecture to integrate all data and information. This proposed architecture is used MIB tree structure to define all information of IA devices. We also apply this proposed architecture to the emergency system to verify it is feasible.

In this paper, a concept of IA BOX will be proposed. The IA BOX and Internet are linked to access to all information of the IA devices. The more interactive and cooperative of software among the linked IA devices, which are linked to become a group to implement the purposes of integration and share sources and data.

2. Literature Review

The main application of entrance system (ES) is to support for recognizing the image, which is accessed from the sensor. Through the ES, it will tell whether the person is allowed to enter the building or not? The configuration of the ES is shown in Figure 1 [4] in which is combination of several CCD devices, sensors, and operation the software of Agents. When guests want to get into the office, the ES will progress the recognizable face image. The location of the guest is monitored by the sensors. Both the two information are communicated through the platform of Agents. Referring to Figure 1, we find the ES is a combination of four agents, they are monitoring agent, recognizing agent, interface agent, and the device control agent.

The monitor agent is constructed by several CCD devices and sensors for monitoring whether guests enter or not?

The recognizing agent is used to recognize the image information of face, which is transmitted from the monitor gent, to recognize whether the entrance person is a guest or not?

Fig. 1. Entrance system configuration.

The Interface Agent is used to manage the security. When the control agent agrees the person enter to office, a message of "open" is delivered to this interface agent.

The Device Control Agent is used to control and monitor all IA devices and sensor respectively. The information offers from the interface agent to the control agent, and the control agent sends the control signal, which is the processed information, to the interface agent for controlling the IA devices and using the Lon system [3].

The Lon system is a connection card or chip. We can create

any kinds of connector and control the connected equipments via this Lon system. The communication protocol of LONTALK is exploited in this Lon system.

As for the application of ES, the Lon system is used to integrated the information of IA devices, which connected to the Lon system chip or connection card. That causes the cost increasing and to connect the IA deceives deficiently each other.

3. The Applications and Structure of IA

The most popular household equipments, such as TV, computer, radio and so on, are the positive devices to receive the information. Nowadays, the Internet techniques are improved, thus we can through the Internet to research and access the needed data quickly and accurately in any place and any time. The main purpose of this paper is to expand the application field. Thus we will integrate the technologies of IA and   Internet to share the resource and data.

In this paper, an example of the fire alarm is used to prove the proposed platform of IA. Upon the sensor detecting fire alarm, the IA system either sends the alarm information to the security service firm, or starts the fire flighting system via the internet system.

The purpose of the integrated IA system, not only offers a facility operation Internet system, but also exchanges the needed data and information. Each device will implement the defined job while it receives exchanged data or information. The integrated IA devices can share the data of each other conveniently.

3.1 The Structure of IA

The structure of the integrated IA system is shown in Figure 2, which consists of three parts. They are: Control Section controls and harmonizes the platform for the integrated information, Application Section uses to transmit the data, command, and status words of the information of IA devices, and the Device Section is used to integrate the different kinds of the interfaces of the IA.
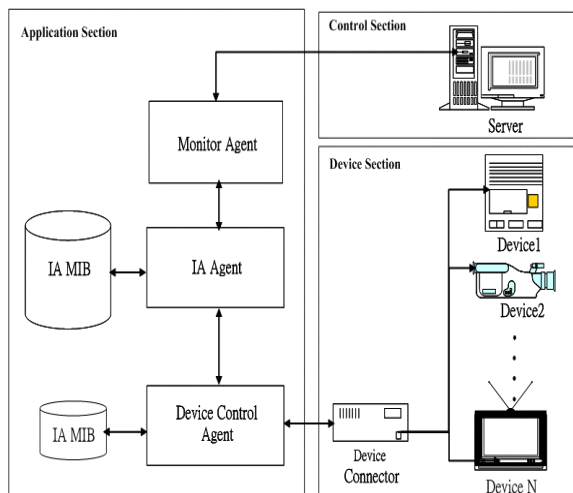

Fig. 2. The structure of IA.

3.2 Control Section

Control Section (CS) is the main part of IA system. The CS is a operation platform, database platform, monitor platform, and data-exchanging platform of this IA system. A IA BOX, which consisted of Device Control Agent, IA MIB database, and Device Connector Interface, is based on TCP/IP protocol to be installed into computer to exchange the needed data and status work of IA. The host will follow the received message, which sent from the IA devices, then send a control signal to fire and monitor the active IA devices.

In order to integrate all active IA devices and share the resource and data, we exploit the IA BOX to integrate all devices and to link the servers. The method of linking is shown in Figure 3, which the information equipments are linked to the device connector of the IA BOX. That is all information equipments must via the IA BOX to transmit the message, and through the TCP/IP protocol sends the message of the information equipment to the server, which will process and recognize the received message.

Fig. 3. Linking between IA and computers.

3.3 Application Section

The Application Section is a combination of Monitor Agent, IA agent, Device Control Agent, and the IA MIB database. The purpose of the Application Section is used to set the process status and events of the integrated IA. Thus, we can integrate the IA devices more convenience and efficient. Furthermore, supporting a optimal decision under a relational and helpful each other.

The specifications of each agent of the Application Section are illustrated as follows:

The Monitor Agent is used to check the message and the executed event. We use the Monitor Agent to monitor each IA device, recognize the executing events, and transmit the message of IA devices.

The IA Agent is to verify and notice the executed event whether associates the event group or not? The functions of the IA Agent are used to upload the message of IA devices, download the commands of Monitor Agent, and verify the set data of IA MIB database.

The Device Control Agent is used to start the IA devices of the event group while a message of execution is received. The main functions of the Device Control Agent are used to check the connection status of the integrated IA devices, and to pre-process the requirements of the IA devices and the data of the local IA MIB. For the registering of the IA devices is process by the Device Control Agent, which follows the set value of the IP of the IA and describes in Figure 4.

Figure 4 devices of IA IP values

The set value of the IP of the IA is a combination of seven parts, where were shown as follows:
Layer1: set the entry value of the IP of the IA.
Layer 2: define the classifications of the IA
Layer 3: indicate the manufactories the IA
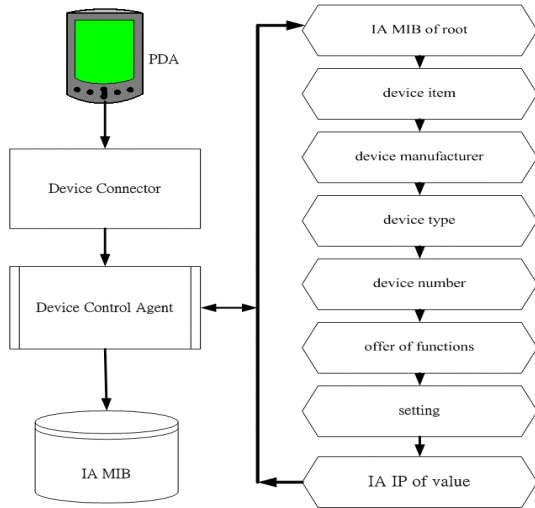Layer 4: indicate the model of the IA
Layer 5: indicate the number of the same IA
Layer 6: indicate the functions of the IA
Layer 7: set the functions of the IA

Finally, the IA MIB database is used to store the registered data of all IA devices. The IA MIB is consisted of three elements, event, group, and IA. The relation is shown in Figure 5.

Fig. 5. The relation between event, group, and IA.

When the IA device is linked to the IA BOX, the IA device will be registered and an IA IP is set. The registered IA devices can share the resources etch other in a group. Several groups can be implemented the same job via the event. The basic function of event is action.

Fig. 6. The example of setting the IA IP

Figure 6 is an example of seeting process of the IA IP. In this example, we use the SAMPO A 2000 alarm sensor. Firstly, in Layer 1, set the IA BOX entry, Root, to 192.168.1.1 (the virtual IP address). In Layer 2, set the code of the goods, alarm, to 3. Thus, the IA IP is 192.168.1.1.3. In Layer 3, we set the code of the company, SAMPO, to 1. Thus the IA IP is 192.168.1.1.3.1, in Layer 4, setting the second IA device to ready. Thus, the IA IP is 192. 168. 1.1.3.1.2. In Layer 5, setting the model 1, A2000 of the IA device to 2. Thus the IA IP is 192.168.1.1.3.1.2.2. In Layer 6, we set the function, upgrade to 1. The IA IP is 192.168.1.1.3.1.2.2.1. Lastly, in Layer 7, setting the function of the IA to 4. Thus the final IA IP is 192.168.1.1.3.1.2.2.1.4.

The integrated IA device is processed by an event and the IA MIB. Referring to the Figure 7, which illustrates the process of the trigged sensor, IA BOX, and IA MIB for message responding and event processing. The system is executed according to the IA IP set value.

Fig. 7. The process of the IA.

In this paper, we define the form of the IA IP as follows:

Each field of the defined IA IP are described as shown in Table 1.

Table 1. The Characteristics of fields of the IA IP

bit	Description
0-11	IP value of IA device
12	The goods of IA
13	The Company of IA goods
14	The type of IA
15	The number of the same IA
16	Error message and basic function of IA
17	Function defination

The Device Section is consisted of IA and Device Connector interface. All IA devices are connected by the Device connector interface. The Device Connector interface is also used as a transmitting/receiveing channel to start the execution command of the IA device.

4 Main Findings

4.1 The process of fire accident

We can apply the integrated IA to the fire accident of a multi-storey building to real-time produce the fire emergency accident. Exploiting the integrated IA system, we can execute the set event and handle the resource of this building, such as starting the camer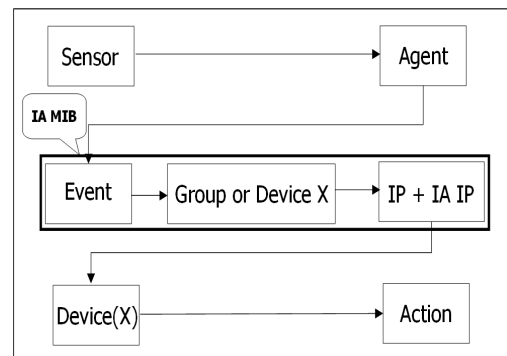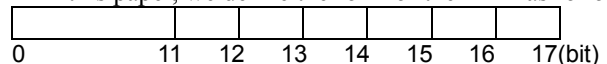a and the security system. Figure 8 illustrates the produce of the fire accident. Upon a conflagration detecting by the fire sensor, this IA system will send a emergency message immediately. The bell alarming, message transmitting and broadcasting, and noticing the management of building to arrive at the scene of fire accident. If the accident is handled and the scene is cleaned by the management of this building. An accident record is listed in the IA system automatically.

If the fire accident is un-handled by the management of this building, the IA system changes to the emergency status automatically, such as evacuating the resident people, putting out a fire, and noticing the fire bureau. Until the emergency accident cleared, the investigations of the accident are recorded in the IA system.

Figure 8 servers detects the process

Referring to the Figure 8, which is consisted of several events. The characteristics of those events are described as follows:

Fire alarm sensor is used to detect and sensor the temperature of the scene. If the fire trigs the fire alarm sensor, then the alarm message will be issued to IA agent.

The agent is used to integrate the IA device to construct a group event system, which all IA devices can support and share each other.

Management is the handler of the accident. When the accident occurred, the management must produce the scene.

Scene is indicated the place of accident. When a accident occurred, the management must arrive at the sceneimmediately.

Conflagration is indicated whether the accident is handled or not? If the accident had been handled then the scene returns to normal, or the scene changes to emergency.

Server is used as a host of this IA system. All information of the fire alarm sensor and the smoke detector are sent to the server, which will check the emergency message. If the emergency message is confirmed by the Server, then the system will change the status to emergency, or the Server will produce a message to the management to reset the system.

4.2 Emergency Machine

If the Server recognizes the emergency is tru, the emergency machine will be started, the following measures will proceed: evacuate peoples, put out a fire, and notice the fire department.

Alarm Recover is used to release the emergency status and recover the system to normal state while the conflagration is under control. After the alarm recovering, the management start to clear the scene of the accident.

Event report has the following functions after the calamity is recovered, such as investigation and criticism of the clamity, and bring the improvability strategy to avoid this kind of case take place again.

Accomplishment Event indicates the event is complemented and releases the trigger event by the system.

4.3 The Relation of Event Group

While the conflagration occurred, the relation event and event group are to set and execute. For example, the relation event groups are shown in Figure 9, which is a combination of fire alarm group (temperature and smoke sensor), give the alarm group (alarm and broadcast equipments), sprinkler group, smoke eliminator group, lighting group (emergency power and illumination equipments), and monitor group (camera and video recorder).

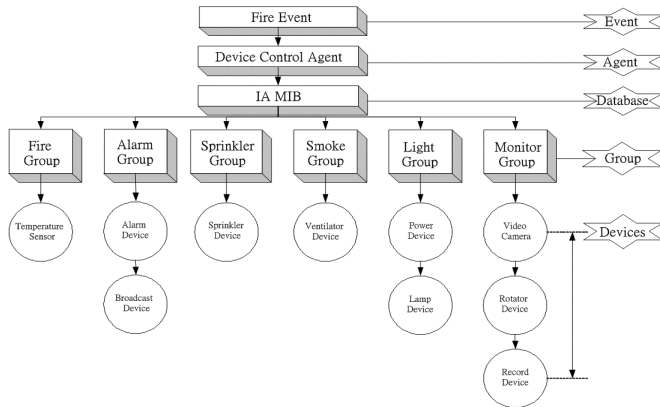Figure 9 trigger the group relation for fire disaster

The operation of IA is decided to the relation of Agent. Thus, the process flow of the Agents corresponds to the effects of integrated system. The interaction of Agents, which shown in Figure 10, is illustrated to the following steps:

Step 1: sending the message of the detected temperature of the sensor to the Device Control Agent via the Device Connector

Step 2: Comparing the message of the Device Control Agent to that of the IA MIB database. In order to obtain the set value, which is corresponded to the event.

Step 3: responding the respond correspondence value from IA MIB to the Device Control Agent.

Step 4: sending the message from the Device Control Agent to the host, and initialing the set IA devices, which is according to the value of the IA IP.

Step 5: re-comparing the massage of the IA agent to that of the IA MIB database of the host to obtain the correspondence value of the trigged event.

Step 6: responding the correspondence value from the IA MIB database of the host to the IA agent.

Step 7: deciding the receiving correspondence value by the Monitor Agent

Step 8: requesting a fire fighting of the Monitor Agent, when the status is emergency.

Figure 10. agents of interactive

4.4 Transmitting and Receuving Message

Figure 11 describes the transmitting flow chart of the message among the Device and the Server of the integrated IA system. In this paper, we discuss with the following two cases:

- Delivering from the Device to the Server
  The message of the device is delivered to the Device Control Agent to process. The Device Control Agent compares the received massage to the IA MIB database to obtain the IA IP. Furthermore, send the correspondence value to the host.

- Delivering from the Server to the Device
  A command is decided to the host, which obtains the set value of the IA IP from IA MIB database. Furthermore, the host sends the IA IP value to the Device Control Agent to start the correspondence events, group event and IA devices.

Fig. 11. Flow chart of transmitting and receiving message among Devices and Servers

5 Conclusions and the Further Works

In this paper, we propose a new structure, which is based on the IA MIB to solve the problems when the IA devices is integrated. Integrating the IA devices on the IA MIB, the IA devices are more flexible, convenient, and grouping to solve the event efficiently. For example, when a conflagration occurred, the message will be sent from the integrated IA system to management center. According to the received massage, the integrated IA system will decode it and decide to take emergency measures, which will minimize the damage. On the other hand, the real-time monitor system is started to monitor the electricity monitor system, spinkler system, security control system, and fire alarm system to build a automatic emergency monitor system.

The contributions of this paper are summarized as follows:

- Create the group model
  According to the IA MIB, we can integrate not only the IA devices, but also create the group model of IA devices. Through this model, all IA devices can share the resources and data.

- Create event model
  For the group model, other groups and equipments can be created a group based on the same event.

5.1 Further Works

The technologies of wireless communication are researched and applied widely. The wireless communication are embedded in the devices, especially the IA devices are required the functions of wireless communication. Applying the wireless techniques to the IA devices is the tendency in the future.

References

1. S.Clippingdale and I.Takayuki, "A Unified Approach to Video Face Detection, Tracking and Recognition," The Institute of Electronics, Information and communication engineers, 1999, Page : 99-109.
2. Neng-Shiang Liang, Li-Chen Fu, Chao-Lin Wu, "An integrated, flexible, and Internet-based control", Proceedings of the 2002 IEEE International Conference on Robotics & Automation, Department of Computer Science & Information Engineering, National Taiwan University, ROC Washington, DC, May 2002.
3. Haakenstad, L.K., "The open protocol standard for computerized building systems: BACnet", Control Applications, 1999. Proceedings of the 1999 IEEE International Conference on, Volume: 2, 1999, Page: 1585-1590.
4. Masaaki Ohgishi and Fumio Mizoguchi Faculty of Sci. and Tech., "Design of Entrance System Using Robotic Agents in Smart Office, "Science University of Tokyo Noda, Chiba, Japan, May 2002.
5. Perkins, David, "Understanding SNMP MIBs White Paper", July 1992.
6. Want, R.; Borriello, G., "Survey on information appliances, "IEEE Computer Graphics and Applications, Volume:20 3, May-June 2000 , Page: 24-31.
7. L.Wiskott, J.m.Fellous, N.Kruger, and C.Von der Malsburg, "Face Recognition by Elastic Bunch Graph Matching," IEEE Trans. Pattern Analysis and Machine Intelligence, 1997, Page : 775-779.

The Study of the VoiceXML-Compliant Interactive Voice Navigation System

Jenn Tang[1], Chin-Yung Chen[2], Tin J. Gou[1]
Department of Information management, Chihlee Institute of Technology 1
Department of Information management St. John's & St. Mary's Institute of Technolo[2]
tjenn@mail.chihlee.edu.tw[1], msi-head@mis.sjsmit.edu.tw[2], Gou@mail.chihlee.edu.tw[2]

Abstract--This work studies and designs an interactive voice-enabled web browsing system. The system complies with VoiceXML standards and has a three-tier architecture, involving front, middle and back tiers. Users accessing the front tier can retrieve web page information by telephone. The proxy server in the middle tier parses the VoiceXML web pages and performs the text-to-speech transformation. Meanwhile, web page data is gathered in the back tier and stored in the database. The prototype system is used to implement three types of web pages for different applications: weather forecasting; real time traffic information for freeways; and English learning. For the first and second applications, the designed proxy server automatically and periodically connects to the Internet to obtain the real time web page data. Tags complied with VoiceXML then are inserted into these pages. Subsequently, the voice module transforms text in these web pages into voice. Users dialing into the server by telephone can thus hear the real time information. Additionally, the Hidden Markov Model is used to design a speech recognition module for keyword recognition. This system allows users to input key commands vocally rather than via the keyboard. We believe that by integrating Internet and telephone networks such a system is extremely useful and worthy of further study.

Key Words--VoiceXML, 3 tiers, speech recognition, text to speech

1. Introduction

Searching for information on the Internet has become a part of daily life, and exchanging information via email and online chat is now widespread. However, the telephone remains the most popular and convenient medium for communication. Integrating the rich content available on the Internet with the Public Switching Telephone Network (PSTN) can enable users to obtained the diverse information on the Internet by telephone.

The integration between the Internet and telephone networks is increasingly being recognized. For example, the World Wide Web Consortium (W3C) [21] launched the Voice Extended Markup Language (VoiceXML), which can provide the development application of Interactive Voice Response (IVR). VoiceXML can generate interactive dialog in digital synthesized voice, speech reorganization, press button entry and voice recording, enabling users to obtain Internet information with telephone as the access interface. VoiceXML is an optimal implementation of voice network.

A telephone network that incorporates the VoiceXML web page and voice-enabled web browsing system can exchange data with the Internet. For example, Voice Portals can provide value-added services for example stock query, ticket booking, weather inquiry and hospital registration

[5,9]. Automatic voice service helps reduce call center costs and improve customer satisfaction. In banking, cost savings can be achieved on large quantities of repeated transactions via press-buttons and automatic voice services. The resulting Voice User Interface (VUI) provides wide access mechanisms and functions, and improves the value of voice applications. However, in China the applications of Chinese phylum, such as speech recognition, cadence and polyphone, remain imperfectly integrated with the VoiceXML system. Commercialization remains in the research phase.

This study designed a rudimentary VoiceXML-based voice information system for connecting Internet and telephone networks. This system enables the user to access Internet information on the telephone, and thus to gain new knowledge without temporal and spatial restrictions. This system helps in examining in the VoiceXML voice label mechanism, and also in familiarizing the user interface with the VUI build program and call flow design principal. The agent server was installed using Validation XML Parser and VoiceXML Document Type Declaration to parse the VoiceXML syntax. Moreover, Text To Speech (TTS) technology was used to convert the words in the web pages into voice and forward them to the user via the telephone network. The user could send the control information using Dual Tone Multiple Frequency (DTMF) and Automation Speech Recognition (ASR). For feasibility testing, three web pages were created using the VoiceXML language, dealing with weather forecasts, freeway traffic broadcasts and English learning, respectively, and the voice-enabled web browsing system was tested using these web sites.

2. Background and Objectives
2.1 Background

VoiceXML is a voice extensible markup language for designing voice interactive interfaces. The VoiceXML forum, involving IBM, AT&T, Lucent and Motorola, cooperates in interactive voice Internet access technology, and includes voice synthesis, digital voice recording, speech recognition, press-button entry, telephone control and additional functions. For the syntax structure, a series of call flows form the whole program. The technology is compatible with existing services based on the existing telecom and Internet. The information prompt and response are provided through interactive voice access. Consequently, users take a telephone set to operate the voice-enabled web browsing system and thus enjoy the rich applications and content available on the Internet. The VoiceXML syntax has the following distinguishing features [3-5,8,9,21]:
(1) A language that describes the interaction mode between

user and computer, with voice synthesis technology or pre-recorded voice file as the output, and DTMF or natural speech recognition as the input.

(2) Telephone set as the major input and output device.

(3) Compliant with XML architecture, with a telephone rather than a PC used as the communication media, and using VoiceXML rather than HTML, Voice Browser instead of a Web Browser, and a system designed by developers familiar with XML syntax.

(4) It is a high-level language with standardization and OS independent.

(5) A development language designed for domain-specific applications.

Like the HTML, the VoiceXML web page can be stored on any web server, and is compliant with the XML language in terms of syntax. Numerous tone elements are defined such that the voice function can be easily written into the VoiceXML file.
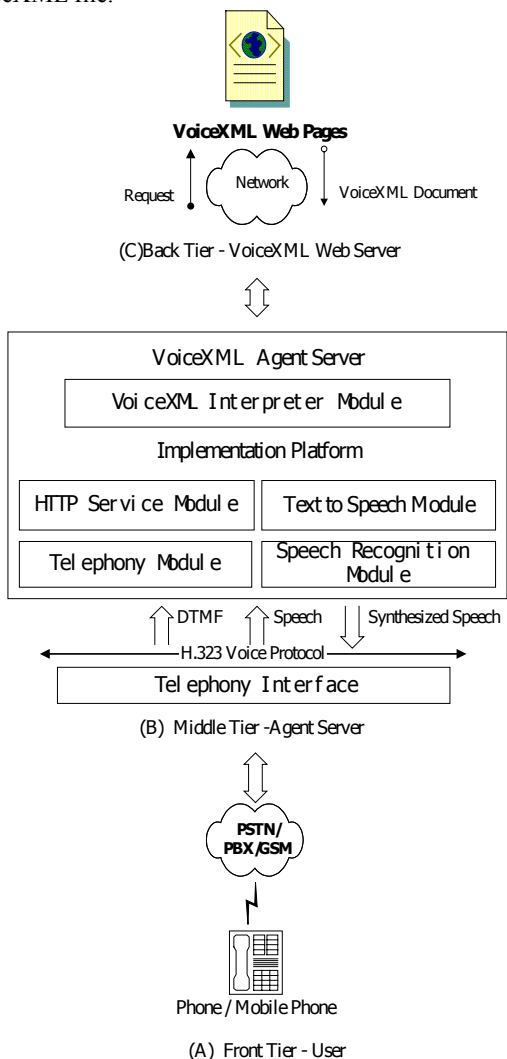

Figure 1. Architecture of the three-tier voice-enabled web browsing system
(A) Front tier – user
(B) Meddle tier – agent server
(C) Back tier - VoiceXML Web Pages

2.2 Objectives

In the VoiceXML standard [3,21] by W3C, a voice web browsing system has the following modules: (1) VoiceXML Interpreter Module, (2) Telephony Module, (3) Speech Recognition Module, (4) HTTP Service Module and (5) Text to Speech Module. These modules construct the functions required for the basic VoiceXML web browsing system. The following section introduces how this study follows the W3C specifications and designs the above modules to realize the system objectives.

3. Implementation

For system design, this study uses the voice card and Microsoft components, including TAPI 3.0 (Telephony Application Programming Interface), MS Speech 5.1 SDK (Software Development Kits) and Microsoft Internet Controls, to reduce the hardware cost. These software data are public standard specifications, and are free, enabling improved system design compatibility and expandability and reduced cost. Based on experience and technological results [6,7] in voice system development, this study refines the pure server architecture to produce a 3-tier architecture, as illustrated in Figure 1. The system is flexible for designing better integrated functions and provides web page selection and consolidation services. The hardware in the designed system includes voice card, telephone switch and two servers that are the agent server and web page server. The agent server is installed using five modules, as displayed in Figure 1(B): (1) VoiceXML Interpreter Module, (2) Telephony Module, (3) Speech Recognition Module, (4) HTTP Service Module and (5) Text to Speech Module. The VoiceXML web pages in the web page server provide weather forecast, freeway traffic broadcast and English learning services, doing so in the form of interactive web pages compliant with VoiceXML standards, as illustrated in Figure 1(C). The designs of the above modules are described below, along with the discussion and solution of related problems.

3.1 VoiceXML Interpreter Module

The VoiceXML is essentially an XML-based file. Consequently, the browser server requires XML Parse and VoiceXML Document Type Declaration [13], to parse the VoiceXML syntax correctly by verifying the grammar and file structure. The parsed results are processed using the Form Interpretation Algorithm [13]. The resultant VoiceXML elements then execute the relative actions individually. During execution, if the file has the grammars defined such as Speech Recognition element or Text To Speech (TTS) element, then that file is forwarded to the lower-layer Speech Recognition or Text to Speech functional modules for execution.

The VoiceXML file is parsed through Microsoft XML Parse4.0[17]. MSXML is a COM object and can be used in an ActiveX environment. If in VB and VC++ programming language, the dynamic link library msxml.dll of Microsoft XML Parse 4.0 stored in the system files are called upon to utilize.

The VoiceXML file defines interactive elements that are required by the file itself, but the user may be lost in the internal call flow. Some key words with global functions thus must return to their initial status. Alternatively, variable communication is defined between different call layers in the programming, so that the variables can be transferred directly or alternatively global keywords (such as home, help and exit) can be defined on the application

layer. These functions are active in all layers in the call flow.

3.2 Telephony Module

This module is defined as the input end interface, which is the protocol for the VoiceXML Interpreter Module to communicate with the telephone interface. Generally, manufacturers of voice cards provide software design programs for their products. However, such programs are incompatible with one another, creating problems in their use and limiting their expansion. Consequently, the system design based on the TAPI 3.0 protocols by Microsoft and Intel for maximizing the compatibility.

The user logs into the system and the location where the DTMF grammar is defined in the VoiceXML, or selects the required service using the telephone press-button, as illustrated in Figure 2, in which the VoiceXML is awaiting the entry of DTMF Grammars "1", "2", "3", "#". Additionally, the waiting time is defined in the file for the timeout event of DTMF, to prevent the system from waiting indefinite system waiting.

```
<grammar mode="dtmf" version="1.0" root="root">
   <rule id="root" scope="public">
      <one-of>
         <item> 1 2 3 </item>
         <item> # </item>
      </one-of>
   </rule>
</grammar>
```

Figure 2. Example of DTMF Grammars [19]

3.3 Speech Recognition Module

Speech Recognition involves sampling the features of the voice signal of the user, then converting them into words and working with the defined VoiceXML grammar to analyze the meaning of the voice. Although MS Speech SDK involves Speech Recognition API, it is designed for recognizing continuous strings. MS Speech SDK thus is poor at recognizing short words, and requires multiple repetitions and corrections to function. Consequently, this study examines how to design a system based on a Speech Recognition model.

This study uses the Hidden Markov Model; HMM [14,16] to design the Speech Recognition model. In our practice, corresponding recognition models first are made for the keywords required for recognition, to allow the sequential users to input the voice comparison. A fast algorithm [10] then is used to classify the keywords based on Figure 3. The tree searching structure significantly reduces recognition time and increases accuracy.

The voices entered by the user are recorded and compressed using the MciSendString API function. The recording format is PCM, 8KHZ, 16Bit, and Simple Channel. Short time energy and Zero crossing rate are used for the final detection to select the voice signal's portion and divide the signal into tone fames. High frequency voice loss occurs during the telephone line transmission and voice recording. To compensate for this loss, the voice signals in each tone frame are processed with pre emphasis through the first order high pass filter (Formula 1). The signal generally must be converted before processing. The short time energy conversion requires multiplying the original tone frame using the Hamming Window (Formula 2).

$$y[0] = x[0]$$
$$x[n] - 0.95\,x[n-1]$$

L denotes the sampling size of tone frame.
(Formula 1)

$$h[0] = 0.54 - 0.46\cos\left[\frac{2\pi n}{n-1}\right] \quad 0 \le n \le N-1$$

(Formula 2)

The feature parameter vector of the voice data is calculated using Linear Prediction Coefficients (LPC). $S(n)$ represents the sampled voice signal. Furthermore, $\hat{S}(n)$ is the discreet value (Formula 3), where α_κ denotes the linear discreet coding coefficient, and p is the order of the discreet filter. The discreet error thus is represented as illustrated in Formula 4.

$$\hat{S}(n) = \sum_{k=1}^{p} \alpha_k \; S(n-k)$$

(Formula 3)

$$e(n) = S(n) - \hat{S}(n) = \hat{S}(n) - \sum_{k=1}^{p} \alpha_k S(n-k)$$

(Formula 4)

Minimizing the square error in Formula 4 allows a group of optimal linear discreet coding coefficients to be obtained. The feature parameters finally used are the LPC linear scrambler coefficients, and are defined as the inverse Fourier conversion of the spectrum alogarithm, as indicated in Formula 5.

$$c_1 = \alpha_1$$

$$c_n = \begin{cases} \alpha_n + \sum_{m=1}^{n-1}\left(1 - \frac{m}{n}\right).\alpha_m.c_{n-m} & (1 \le n \le p) \\ & (n \ge p) \\ \sum_{m=1}^{p}\left(1 - \frac{m}{n}\right).\alpha_m.c_{n-m} & \end{cases}$$

(Formula 5)

Generally, voice is subject to the noise effect, which causes voice spectrum distortion or interference. However, noise does not significantly influence transfer scrambler. Since it is a dynamic parameter, voice can be combined with Formula 5, to determine the feature parameter for recognition in Formula 6.

$$\frac{\partial c_n(t)}{\partial t} = \Delta c_n(t) = \frac{\sum_{k=-k}^{k} k - c_n(t+k)}{\sum_{k=-k}^{k} k^2}$$

(Formula 6)

Finally, the Viterbi Algorithm in the Dynamic Programming mode is used to compute the similarity between the input voice feature and voice data model

feature values, to maximize the Joint Probability between the observation and status sequences, and to determine the optimal status sequence. The final calculation result is defined as the voice variable in the VoiceXML Document, and is then executed by the VoiceXML Interpreter.
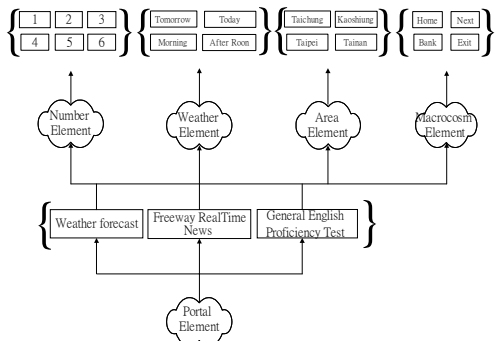

Figure 3. Classifications of keywords

3.4 HTTP Service Module

The design of this module uses the Web browser Object in Visual Basic to obtain the web page database and VoiceXML Document in the remote server, which then is submitted to the VoiceXML Interpreter Module for parsing, and a response is made based on the file-required function.

The VoiceXML web page browsing provided by the system must be updated in real time to reserve its availability, because both weather forecasts and freeway traffic broadcast involve real-time information. Updating this information manually can become extremely complex. The information sources in this study are the web pages of the central meteorological bureau [1] and the freeway real-time traffic information system [2] of the Ministry of Communication. The original codes of the web pages are selected to determine specific fields and generate the required strings. For example, in the web pages of the freeway real-time traffic information system of the Ministry of Communication, the values in the fields <td> and <p align="center"> are chosen and combined into the tone element fields of the VoiceXML Document, to provide users with the newest voice information, as illustrated in Figure 4.
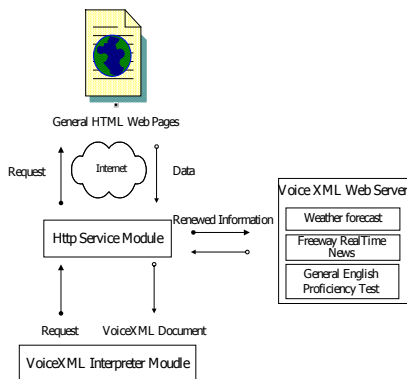

Figure 4. Fetching information and combining into VoiceXML web page

HTTP Service Modules essentially provide identical functions to Netscape Browser and Internet Explorer. However, the VUI cooperates with the VoiceXML Interpreter for whole-domain grammar design. The functions, including home, previous page, stop and help, are all voice-controlled whole-domain commands. The user can use these voice commands on any web page.

3.5 Text to Speech Module

To support multiple user environments, reduce system workload and enhance execution efficiency, the strings are removed from VoiceXML and the words are placed in a Froms2.0 Textbox Object, after which the Speech Engine synthesizes the voice, and finally the voice information is forwarded to the user via the voice card and telephone interface. This is the whole voice interactive session. However, MS Speech5.1 SDK [18] supports only simplified Chinese and can only be attached to a simplified Chinese operating system. A module is designed to convert traditional Chinese into simplified Chinese (Big5 to GB2312).The Unicode-supported Froms2.0 Object then is used to read the converted simplified Chinese, and the Speech Engine synthesizes the Chinese voice.

For the word string in the file, the VoiceXML Interpreter Module interprets the element-required voice engine and then reads it smoothly. Some elements for describing voice features can be defined in the file, for example expression speed and volume, as illustrated in Figure 5.

```
<vxml version = "1.0"?>
<vxml version = "2.0" encoding = "iso-8859-1">
<Form>
<Voice Required="Name=Microsoft Mike">
  Hello, Good luck.
</Voice>
<Voice Required="Name=Microsoft Simplified Chinese">
你好,歡迎進入語音入口網站
</Voice>
<Voice Required="Name=Microsoft Mary">
<Rate Speed="3">Taiwan</Rate>
<Spell>Good</Spell>
<Pitch Middle>System </Pitch>
</Voice>
</Form>
</vxml>
```

Figure 5. Voice description feature elements

4. System Achievements

The current system platform set up comprises two hosts (one agent server and one web page server), one voice card, one telephone set and one telephone switch. To demonstrate the achievements of this study, the results are summarized below.

4.1 Voice portal architecture

To verify the effectiveness of the subject voice-enabled web browsing system, this study designed a voice portal web site, as illustrated in Figure 6, providing weather forecast, freeway traffic broadcast and English learning services. Three different sites comprised VoiceXML web pages and CGI programs. The system updates itself from the central meteorological bureau and freeway real-time traffic information system of the Ministry of

Communication at 30 minute intervals, thus obtaining the latest information for updating the VoiceXML web pages and preserving the real-time nature of the information offered on the web pages, as illustrated below.

Figure 6. Architecture of VoiceXML portal

4.2 System demonstrations

When the user dials by telephone to access the voice-enabled web browsing system, the system first provides aural instructions. The user either uses the telephone buttons or simply uses spoken commands to select from among various services, including weather report, traffic broadcast and English learning, as shown in Figure 7(A). The telephone buttons are used as follows: Buttons 1 to 6 are used for selection, according to the prompts of specific web pages. Meanwhile, buttons 7 ~ # are global system commands and can be used on any web page, as below: 7: Volume up; 8: Volume down; 9: Previous page; *: Next page; 0: Home page; and #: Exit. The system loads the required web page depending on user commands, and loads different speech recognition element for different web pages, such as weather elements for weather forecast web pages, and e-learning elements for English learning web pages. Using different recognition models for different web page can effectively enhance the recognition effect, as illustrated in Figs. 7 (B) (C) and (D). The user can also use "home" and other verbal global commands on any web page. The representation of the study achievements demonstrates that the whole voice browsing process resembles browsing a HTML web page. The user can easily learn to use the system, and thus can access web pages without limitations of time and space. The ease of use verifies the practicality of the VoiceXML voice system designed in this study.

(A)

(B)

(C)

(D)

Figure 7. System demonstrations

5. Conclusion

Integration of Internet and telephone networks is a clear trend. This study attempted to design a system for VoiceXML voice-enabled web browsing, thus integrating web information with telephone networks. The problems associated with such integration of Internet and telephone networks are studied in detail based on the system development and application. The related voice processing technologies and applications discussed here, for example

VUI, call flow, text to speech and speech recognition, are practical yet quite challenging, and so deserve further development. This study created an interactive service web site, designed modules for text to speech and speech recognition and integrated the Internet and telephone voice control. Such systems allow users to easily obtain Internet information via the telephone.

The contributions of this study are as follows: The fact that a computerized information service platform is developed only on Internet represents a breakthrough. The information service platform integrated with the telephone network developed here is easier to use than similar systems developed elsewhere. Future studies will pursue further integration between Internet and telephone networks, as well as the technologies and problems related to developing VoiceXML interactive voice response.

Acknowledgment

References

[1] http://www.cwb.gov.tw/V4/index.htm.

[2] http://211.79.135.69/n1.htm.

[3] Lee, J.-Y., (2000), An Implementation of a Simple VoiceXML Browser, master thesis of National Chung Cheng University.

[4] Cho, M.-H., (2001), The Establishment of VoiceXML Service Environment and Design of the Development Tools, master thesis of Yuan Ze University.

[5] Lin, Z. –H., (2002), The Voice Spec. and Application - VoiceXML, Journal of Network Communicatino,130, 14-17.

[6] Chen, Y. –S. & Hong, S. -W., (2000), An Agent System for Integrating Messages on Internet and PSTN, Proc. Conf. on the Communication Technology Topic in Ministry of Education, 291-300.

[7] Chen, Y. –S., (2001), A Web Information Integration Service System with Telephone Voice Browsing, Proc. Conf. on the Communication Technology Contest for College Student, 383-395.

[8] Xu, Z. –Z., (2002), The Voice Browser Compliant with VoiceXML1.0 Spec, Journal of Computer and Communication, 100, 80-87.

[9] Zeng, Z. –W., (2001), The New Trend of Voice Commerce – the Explores of Voice Portal , Journal of Computer Technology, 61, 4-7.

[10] Yang, Z. -K., (2000), The Application of Rapid Algorithm in Large Vocabulary Keyword Retrieve, master thesis of National Center University.

[11] http://www.find.org.tw/0105/news/0105_news_disp.asp?news_id=2421.

[12] http://www-4.ibm.com/software/speech/dev/.

[13] http://www.w3.org/TR/voice-intro/.

[14] Jay G. Wilpon et. al., (1990), A Automation Recognition of keywords in Unconstrained Speech Using Hidden Markov Models, IEEE Trans. on Assp., 38(11), 1870-1878.

[15] Kenneth G. -R, (2000), VoiceXML and the Voice Enable Web, SpeechTek.

[16] Rabiner, L. -R. & R. W. Schafer, (1978), Digital Processing of Speech Recognition Signals, Prentice-Hall Co. Ltd.

[17] http://www.microsoft.com/downloads/search.aspx?displaylang=zh-tw.

[18] Microsoft Speech SDK5.1, http://www.microsoft.com/speech/speechsdk/sdkinfo.asp.

[19] Chen, M. –Y. & H. -R. Wu, (2002), The Establishment of Web Based Mandarin Dialog System with VoiceXM, TANET2002, 207-212.

[20] Rose, R.C., et al., (2001), A Hidden Markov Models Based Keyword Recognition System, ICASSP'90, 129-132.

[21] W3C (World Wide Web Consortium), http://www.w3.org/TR/voicexml20/.

An Efficient Method to Analyze the Relationship Between Music and Brand Positioning ~ A Practical Analysis in Fast-Food Product And Cosmetic Product

Yong Su[1], Huay Chang[1 2], Shieh-Shing Lin [3]

[1]: Department of Business Administration, Management School, Fu Dan University
[2]: Department of Information Management, Chihlee Institute of Technology
[3]: Department of Electrical Engineering St. John's & St. Mary's Institute of Technology
cecilia2chang@yahoo.com.tw[1 2]

ABSTRACT

Music owns the characteristics to inspirit human emotion. In order to increase the consumers' positive impression on product brand positioning, many brand designers use the core element 'music' while broad-casting the brand positioning information. Therefore, we can hear beautiful music accompanied by the various kinds of product brand positioning strategy. Music catches the attraction of the consumers. Music also establishes the specific magic in brand positioning by its structured elements. Therefore, how to design an appropriate brand music would become one of the most important goals while implementing universal brand positioning policy.

In our research, we believe that there is a quantity relationship between music and brand positioning. Therefore, based on the following stages: analyzing the music structured elements (rhythm, melody, timbre) of the fast-food product and cosmetic product, analyzing the four niches phases (brand space theory, brand asset theory, brand value theory, brand circle theory), collecting the samples, designing the questionnaire, establishing the statistical model. We found out the brand-positioning-three-dimensional coordinate. The three axis indexes are: 'overall index: consumer index: product index. After analyzing the structured elements of the music, we also found out the music-three-dimensional coordinate. And an appropriate vocabulary is generalized by the coordinate. We believe that the result will help the music designers creating appropriate music. It would also help achieving the goals of brand positioning policy.
Key words: Brand, Brand Positioning, Music, Emotion

1. INTRODUCTION

1.1 Research Background

The reaction of human while hearing the music is nature. Music creates an imagination space for consumers. Music establishes special relationship for brand and product. Music appears high influence on the brand attitude of the consumers. McInnis and Park [1] found, in spite of the high-involvement or low-involvement, music appears influence in inspiriting consumers' emotion and brand attitude. The influence is significant, especially under the low-involvement.

1.2 Research Motive

Due to the widely use of multi-type and multi-function broadcasting media, music places an important role in information promoting. Product brand means a feeling, reliance, a service for the consumers. While the brand music is broadcasting, the consumers feel the emotion inspiration. This kind of feeling deepen the consumers' realization of the product brand.

The creator of the music must realize that music creation means the creation of communication. The creator of music must also realize that the music must fully represent the importance of product brand. Therefore, seeking the relationship between music and brand positioning and using the relationship to deepen the realization of the consumers is the motive of the research.

1.3 Research Purpose

The purpose of the research is implemented by the following stages: first, focus on the characteristics of the brand positioning, and list the index values of the four niches phases, second, analyze the structured elements of the brand music and list the quantity values of the music structured elements, third, select appropriate sample, design questionnaire and establish statistical model.

1.4 Research Scope

The scope of this research is:

1.4.1 Brand Sample

The research sample includes the McDonald's fast-food product brand and SKII cosmetic product brand.

1.4.1 Music Sample

The research of the music includes the brand music of McDonald's and SKII.

1.5 Operational Definition

There is a few operation definition in the research:

1.5.1 Music

In the research, the 'music' means that can be broadcasted by multi-media equipments. The music doesn't include the lyric. Not only the various kinds of instruments but the natural sound is allowed to be used while designing the music.

1.5.2 Brand Positioning

After reviewing the past literatures, we submit four

niches phases theory. The four niches phase theory includes brand positioning elements those could be used in establishing the coordinate.

1.1 Endless Positioning
1.2 Universal Base
1.3 Business Environment
1.4 Target Market
1.5 Consumer Market

BRAND SPACE
NICHE-1

2.1 Tangible Asset
2.2 Reverse Marketing
2.3 Simple Message
2.4 Organization Management
2.5 People Asset
2.6 Word Power

BRAND ASSET
NICHE-2

BRAND POSITIONING

BRAND VALUE
NICHE-3

3.1 Business Life
3.2 Business Culture
3.3 Feeling Value
3.4 Network Asset
3.5 Green Brand
3.6 Trust Logo

BRAND CYCLE
NICHE-4

4.1 Brand Differential
4.2 Brand Impression
4.3 Brand Renewal
4.4 Re-Positioning
4.5 Life Cycle

Fig. 1-1 Four Niches Phases of Brand Positioning

2. LITERATURE REVIEW

The collected literature includes three parts: First,, the literature review of music structure elements; second, the relationship between music, consumer brand attitude and consumer psychology; third, the brand positioning.

2.1 Music Structured Elements

The famous description of music: 'The content of music is the exercising type of music' is said by Hanslick Eduar [2] in nineteen century. He thought the origin element of music is the regular and good listening voice, the principle of the pleasant music is using rhythm, the elements of writing songs are tone, harmony and rhythm. And the continuing melody is combined with the basic elements those create the basic structure of the music.

Manfred Clynes [3] pointed out the suitable structured music can efficiently help human's brain reaction and emotion reaction. Mei-Nyu Gwo [4] pointed out music is the art and science, is created by human nature development. Mei-Nyu Gwo [4] also pointed out all the music's basic principles are structured by rhythm, melody, harmony and timbre. The following literature review focuses on rhythm, melody and timbre.

2.1.1 Rhythm

Zuckerkandl Victo [5] pointed out music broadcasts via the time, music also changes via the time. He pointed out rhythm is the time factor of the music. There are two types of rhythm: the widely rhythm and the narrow rhythm. The widely rhythm includes the nature aspects of the universe, the physical appearance of human body, human action and human language. The narrow rhythm means the regulatory movement, the repetition of music's length, speed, strength under the limited time. This kind of moving includes the following three factors: strength, speed and repetition. Hevner [6] pointed out the slow movement easily creates the circumstances of silence, sad and lonely. The fast movement easily creates the circumstances of exciting and merry.

i. Time：The time includes three factors: time, speed and notes. Jing-Hwung Chang [7] pointed out while the music is playing, the strong sound and the weak sound is played repeatedly, is called time. The classification of time is in Table 2-1 and Fig. 2-1.

Fig. 2-1 The System of Double Times and Triple Times

Data Source: Jin-Hwung Chang, 1982

Table 2-1 The Classification of Time

Double Times	Triple Times
Two Time Four Time Six Time Twelve Time	Three Time Nine Time

Table 2-2 The Classification Of Speed

Rhythm	Slow Time	Medium- Slow Time	Medium Speed	Medium- Fast Speed	Fast Speed	Faster Speed
Time Speed Per Minute	40-56	56-69	69-88	88-108	108-132	132 -184

Data Source: Jin-Hwung Chang, 1982

Table 2-3   24 Styles of Music

Jazz Swing	Shuffle Rock	Bouncy
Jazz Waltz	Lite Rock	Ethnic
Waltz	Medium Rock	Funk Bossa
Country 12/8	Rock	Cha Cha
Country 4/4	Miami Rock	Bossa Nova
Country 12/8	Pop Ballad	Irish
Blues Shuffle	Pop Ballad 12/8	Rhumba
Blues Straight	Milly Pop	Reggae

Data Source：Jau Liu, 2001

ii. Speed：Holbrook and Anand [8] concluded that the music factors' influence represented as the sign of '∩'. They pointed out the appropriate speed of music influences human most. Jing-Hwung Chang [7] pointed out different types of speed is used to express different concept. The classification of speed is as Table. 2-2.

2.1.2 Melody

Gundlach [9] found that high-volume music acts happy and exciting emotion, low-volume music makes people feel silence emotion. Alpert and Alpert [10] [11] also found that the commercial music influences the mood and purchase desire of the consumer. The lite

melody of music leads to happy mood of the consumer. The sad melody of music increases the purchase desire of the consumer. We will make a discussion in the fields of style, interval and timbre.

i. Style: The style strongly represents the type of melody and rhythm. Jau Liu [12] listed twenty-four styles of music.

ii. Interval: The interval means the height of the relationship between two notes. Jing-Hwung Chang [7] pointed out the wider interval is called 'full interval', the narrow interval is called 'half-interval'. The 'half-interval' is half of the 'full-interval'.

iii. Key: Jing-Hwung Chang [7] pointed out the key is created while the relationship of the interval is established and the height of the notes is listed. The key means the accurate height of the notes.

2.1.3 Timbre

The timbre is created while using various kinds of sound. It includes not only the sound that is played by various kinds of instruments, but also the nature sound played by human beings and animals. Each timbre represents different sound curve. Each sound curve represents specific characteristics and various degrees of influence to the human beings.

2.2 The Relationship between Music, Consumer Brand Attitude and Consumer Psychology

Alpert [10] [11] has made a research in the influence of the consumer purchase desire about the playing of music. He found that the happy music made the consumers represent a joyful emotion about the product, the sad music made the consumers represent higher purchase desire.

Park and Young [13] made a research of the influence in the usage of music about consumers' psychology. They tried to find out the difference of the advertisement attitude, brand attitude and behavior ambitious with or without the background music and three different-level involvement (high-knowledge involvement, high-feeling involvement and low involvement). They pointed out three results: first, the consumers couldn't concentrate while playing the music under high--knowledge involvement; second, the result couldn't be appealed apparently while playing the music under high-feeling involvement; third, it appeals benefit while playing music under low-involvement.

Bozman, Mueling and Pettit-O'Malley [3] made an research in music commendation. They found out that there is an interactive influence on music commendation and the degree of involvement toward brand attitude. We see the shorter distance between negative-music and positive-music toward brand attitude under the high-involvement in Fig. 2-2. Bozman [3] and his group found out the music commendation results from the joyful feeling under the low-involvement. This reflects the emotion to brand itself.

Holbrook and Anand [8] made an research in the relationship between music rhythm and consumer purchase attitude. They found out while the rhythm strengthens the consumer purchase attention increases, vs. But the consumer purchase attention decreases while the
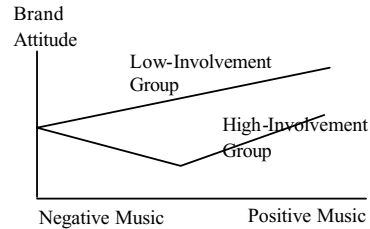

Fig. 2-2 The Cross-effect of Music Commendation and Involvement-Type Toward the Influence of Brand Attitude

Data Source：Bozman, Mueling and Pettit-O'Malley, 1993

rhythm speeds up.

The music influences the internal aspect of human beings. It re-creates the external behavior of human beings. While we are discussing the influ- ence function of the music, we must understand the music that is researched by the psychologists. Music owns the strong inspiration model. While the inspiration accu- mulates a series of repeatedly desire, the result raises apparently.

Posch, Palmer, Cantor and Mischel [14] researched the creation of brand attitude. The result of their research pointed out the consumers use the concept of prototypes to justify the benefit of the product. The concept of prototype represents the impression of the series of product nature or product characteristics. They thought that the creation of brand attitude is structured by three elements: feeling, concept and attitude. These elements are tightly combined with consumer' motivation. Gorn, Mitchell and Olson [15] researched the creation of brand attitude. They found that the creation of brand attitude is concerned with the leading of feeling.

2.3 Brand Positioning

Philip Kirtler [16] pointed out 'Brand is a name, a mark, a symbol or a graph; or the combination of them. Brand is used to identify the sale product or service. Brand is used to be distinguished with the competitor's product and service.

Ming-Hai Ye [17] thought that 'Brand positioning is a process and a result of establishing a target that includes all the brand impression. Brand positioning is the outcome of the marketing development. The core of the brand positioning is separated to the detailed market, selective target market, the brand significant positioning.'

Table 2-4 The Core of Brand Positioning

Detailed Market	1. Ensure the detailed variables and detailed market. 2. Outline the description of detailed market.	* The three-stage of the detailed market: Investigation stage, analysis stage, separation stage. * The basis of the detailed consumer market: Geography, people, psychology, behavior.
Selective Target Market	3. Evaluate the market attraction of each detailed market. 4. Select target market.	* Concerning factors: The attraction of the size and development tendency of detailed market, the internal structure of detailed market.
Brand Significant Positioning	5. Assure brand ability image for each market. 6. Select and plan the brand ability image and transfer to digital model.	

Data source: Ming-Hai Ye, Brand Renewal and Brand Marketing, Hebei People Publisher, Feb. 2001

Bing Chang [18] thought the general method of the brand positioning places in: Continuing satisfy the new needs of the consumer, inspiriting the purchase desire, introducing product function toward consumer' psychology, amplifying the product's new function to achieve sale goal and creating the representation symbol of the product. Yong Su and Shin-Ming Jin [19] pointed out the views of the brand positioning concern the following fields: the view of product function, the view of product quality, the view of target market, the view of price, the view of concept, the view of benefit, the view of direction.

Kevin Lane Keller, Brian Sternthal and Alice Tybout [20] pointed out the traditional method of brand positioning is to focus on the 'point of difference' that distinguishes each brand from its competitors. But this method might neglect other important issues. Under some circumstances, it is required for each brand to be placed in same base. But the efficient brand positioning requires not only the cautious concern about the 'point of difference', but also the 'point of parity'. The right and suitable product positioning competition needs an appropriate reference-coordinate that combines the 'point of difference' and 'point of parity' of the product.

3. RESEARCH METHODOLOGY

3.1 Research Framework

In this research, we use an efficient method to analyze the relationship between music and brand positioning. First, we selected the appropriate brand sample and music sample. Second, we established the brand positioning-three-dimensional coordinate. The three axis indexes are: 'overall index: consumer index: product index'. Third, we analyzed the music structured elements and calculate the music-three-dimensional coordinate. Then, an appropriate vocabulary was generalized by this coordinate. Furthermore, the testers were invited to the lab experiment and filled out the questionnaire, then the three-dimension-model relationship between music and brand positioning was established. Finally, an evaluation inspection process was made by the T-test analysis. If the relationship between music and brand positioning appears normally, the quantity model could be processed practically.
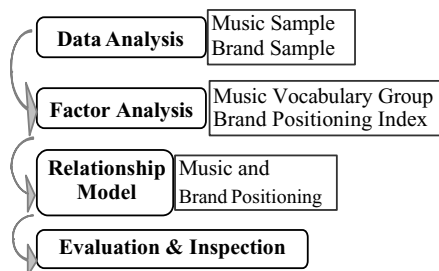

Fig. 3-1 Research Framework of the Relationship Between Music and Brand Positioning

3.2 Research Design

There are five stages in this research.

3.2.1 Data Collecting

The sample focuses on the universal brand.

i. Fast-Food Brand: McDonald's Company.

ii. Cosmetic Brand: SK II Company.

3.2.2 Data Analysis

After data was collected, we made further analysis in both music and brand positioning.

i. Music Analysis: Based on three music structured elements---rhythm, melody and style, we analyzed the music by music digital software. Finally, the music-three-dimensional coordinate is established.

ii. Brand Positioning Analysis: Based on the four niches phase, the brand-positioning-three-dimensional coordinate was established. The three axis indexes are: 'overall index: consumer index: product index'.

$$Y = w_1 X_1 + w_2 X_2 + w_3 X_3 + w_4 X_4$$

Y = Overall Index

X_k = Dimensional Variables

W_k = Principal Weight

Table 3-1 Brand Positioning Niches Variables

	X_1	X_2
Brand Positioning Niches Variables	**NICHE 1 Brand Space Theory**	**NICHE 2 Brand Asset Theory**
	X_3	X_4
	NICHE 3 Brand Value Theory	**NICHE 4 Brand Cycle Theory**
	X_1	X_2
Brand Positioning Niches Factors	1.1 Endless Positioning 1.2 Universal Base 1.3 Business Environment 1.4 Target Market 1.5 *Consumer Market*	2.1 Tangible Asset 2.2 Reverse Marketing 2.3 Simple Message 2.4 Organization Management 2.5 People Asset 2.6 Word Power
	X_3	X_4
	3.1 Business Life 3.2 Business Culture 3.3 Feeling Value 3.4 Network Asset 3.5 Green Brand 3.6 Trust Logo	4.1 Brand Differential 4.2 Brand Impression 4.3 *Brand Renewal* 4.4 Re-Positioning 4.5 Life Cycle

3.2.3 Establish Vocabulary

Based on the Farnswort's Correction Table Toward Hevner Adjective Circle, we collect the music description vocabularies.

Table 3-2 Farnswort's Correction Table Toward Hevner Adjective Circle

A	B	C	D	E
Cheerful Gay Happy Bright Bright Joyous Merry Playful Sprightly	Fanciful Light Quaint Whimsical	Delicate Graceful Lyrical	Dreamy Leisurely Sentimental Serene Smoothing Tender Tranquil Quiet	Longing Pathetic Plaintive Pleading Yearning

	F	G	H	I	J
	Dark Depressing Dolefu Gloomy Melancholic Tragic Mournful Pathetic Sad Serious Sober Solemn	Sacred Spiritual	Dramatic Emphatic Majestic Triumph-ant	Agitated Exalting Exciting Exhilarated Vigorous	Frus-trate

3.2.4 Experiment Design

In the lab test, we used the Semantic Differential Method (SD Method) to establish the evaluation questionnaire. Based on the music-three-dimensional coordinate, the testers were asked to fill out the questionnaire while hearing the music. After that, we used the factor analysis to find out the appropriate relationship between music and brand positioning.

3.2.5 Evaluation and Inspection

At last, the testers were invited to the lab. There were various types of music playing toward one brand. The testers were asked to fill out the questionnaire while hearing the music. The result would be used as the correction process of the evaluation and inspection of the model of the relationship between music and brand positioning.

4. CONCLUSION AND SUGGESTION

In this section, we would like to list the conclusion and the contribution we made in this research. We also list the limits and the future research direction of this research.

4.1 Research Conclusion

4.1.1 Analysis Result of Fast-Food Product

i. Music Analysis Result of Fast-Food Product

By analyzing the McDonald's music's rhythm, melody and style, we calculated the music-three-dimension-coordinate: "Change-Definite".

A vocabulary was generalized: 'Merry'.

ii. Brand Positioning Analysis Result of Fast-Food Product

By using the factor analysis, the brand positioning-three-dimension-coordinate was established. The "overall index: consumer index: product index" were calculated.

4.1.2 Analysis Result of Cosmetic Product

i. Music Analysis Result of Cosmetic Product

By analyzing the SKII music's rhythm, melody and style, we calculated the music-three-dimension-coordinate: "Bright-Dark".

A vocabulary was generalized: 'Graceful'.

ii. Brand Positioning Analysis Result of Cosmetic Product

By using the factor analysis, the brand positioning-three-dimension-coordinate is established. The "overall index: consumer index: product index" were calculated.

4.1.3 The Relationship between Music and Brand Positioning Of Fast-Food Product and Cosmetic Product

According to the "overall index: consumer index: product index" and the music-three-dimension-coordinate, the three-dimension-model of the relationship between music and brand positioning is as Fig. 4-1.
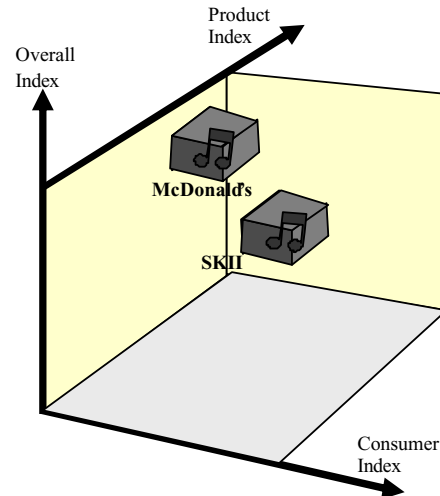

Fig. 4.1 Three-Dimension-Coordinate-model of the relationship between music and brand positioning

4.2 Research Contribution

Through this research, we learned more knowledge about music and brand positioning. We also realized that the relationship between music and brand positioning can be defined by the model of description vocabulary. This model would become the great help while the brand designer selecting the appropriate music for the brand. The appropriate music would also become the concept promotion of the brand positioning process for the consumers. The contribution of this research includes:

i. Clearly classification of the music type of the fast-food brand and cosmetic brand.

ii. Clearly definition of the description vocabularies of the fast-food brand and cosmetic brand.

iii. Clearly definition of the factors of the brand positioning and music.

vi. Completely establishment of the brand positioning three-dimension-coordinate and the music-three-dimension- coordinate.

v. Establishment of the three-dimension-coordinate model of the relationship between music and brand positioning.

4.3 Research Limits

In this research, we focused on the fast-food product brand and cosmetic product brand only. The limits of this research are:

i. In this research, we collected the data from McDonald's and SKII only. The samples are limited in this research.

ii. In this research, the lab tests and the questionnaires focused on McDonald's and SKII only. The samples are limited in this research.

4.4 Suggestion and Future Study

In this research, we focus on the fast-food product brand and cosmetic product brand only. Therefore, the research of the relationship between music and brand

positioning of other industries can be researched in the future. The research methodology of this research could become the primary basis of the future further research. The future further study could include:

i. Expanding the brand's samples: The expanding samples of the universal various brands could find out the regulatory relationship between music and brand positioning.

ii. Expanding the consumers' samples: According to the expanding of the research samples, the consumers' samples expand, too.

ii. Expanding the physical area of the consumers: According to the study of the universal various brands, we could compare the culture variance in different physical area.

5. REFERENCE

[1] Macinnis, D. J. and C. W. Park , "The Difference Role of Characteristics of Music on High-Involvement Consumers, Processing of Ads.", *Journal of Consumer Research*, Vol. 18, pp.161-173, September 1991.

[2] Hanslick Gorn Eduar, Gerald J.., "The Effects of Music in Advertising on Choice Behavior: A Classical Conditioning Approach", *Journal of Marketing*, Vol. 46, Winter 1982, pp.94-101, 1982.

[3] Mueling Bozman, Pettit-O'Malley, Clynes, Manfred, "*Communication and Generation of Emotion Through Essentic Form*", Emotions- Their Parameters and Measurement, Lennart Levi, ed. New York Raven Press, pp. 561-601, 1993.

[4] Mei-Nyu Gwo, "The Basic of Music Education-Voice Concept Teaching", *Course and Teaching Journal*, Vol.20, 1999.

[5] Zuckerkandl Victo, "Conceptualizing Involvement", *Journal of Customer Research*, Vol. 47, pp.4-13, 1986.

[6] Hevner, Kate, "The Affective Character of the Major and Minor Modes in Music", *Americal Journal of Psychology*, Vol. 47, January 1935, pp. 103-118, 1935.

[7] Jing-Hwung Chang, "*New-Edition of Music Knowledge*", Wunan Publisher, 1982.

[8] Holbrook and Punam Anand, "Aims, Concepts, and Methods in Marketing Reserch on Consumer Esthetics: The Effect of Tempo on Perceptual and Affective Response to Musics", *unpublished paper*, 1988.

[9] Ralph H. Gundlach, "Factors Determining the Characterization of Musical Phrases", Americal Journal of Psychology, Vol. 47, pp.624-643, October 1935.

[10] Judy I. Alpert and Mark I. Alpert, "The Effects of music in Advertising On Mood And Purchase Intentions", *unpublished working paper*, 85/86, 5-4, University of Texas, 1986.

[11] Judy I. Alpert and Mark I. Alpert, "Background Music as an Influence in Consumer Mood and Advertising Responses", *Advances in Consumer Research*, Vol. 16, Thomas K. Scrull, ed Honolulu, HI: Association of Consumer Research, pp.485-491,1988.

[12] Jau Liu, "*The Magic Band (Brand In A Box Pro)*", The Third Curve Information Company, 2001.

[13] Park, C. Whan and S. Mark Young, "Consumer Response to Television Commercials: The Impact of Involvement and Background Music on Brand Attitude Foemation", *Journal of Marketing Research*, Vol. 23, pp.11-24, February 1986.

[14] Posch, Palmer, Cantor and Mischel, 1978.

[15] Mitchell Gorn and Olson, "The Effects of Music in Advertising on Choice Behaviror: A Classical Conditioning Approach.", *Journal of Marketing*, Vol.46, pp.94-101, Winter 1982.

[16] KirtLer, "Establishing the Promotion Ability In China", *Business Research*, pp.4-7, Jan. 2003.

[17] Ming-Hai Ye, "*Brand Renewal and Brand Marketing*", HeiBei People Publisher, Feb. 2001.

[18] Bing Chang, "*Will Brand Die?*", GwangTung Economic Publisher, June 2001.

[19] Yong Su and Shin-Ming Jin, "*Modern Company Top Brand Strategy*", ShangTung People Publisher, 1999.

[20] Kevin Lane Keller、Brian Sternthal、Alice Tybout, "Three-Questions of Brand Positioning", *Harvard Business Review*, pp.91-97, Oct. 2002.

Study on Application of GIS to the Cadastral Data Management

Kune-Yao Chen
Department of MIS, St.
John's & St. Mary's
Institute of Technology,
Taiwan 251, Taiwan,
R.O.C.

Chih-yung Chen
Department of MIS, St.
John's & St. Mary's
Institute of Technology,
Taiwan 251, Taiwan,
R.O.C.

Ginn-Yein Chen
Department of Civil
and Water Resources
Engineering, National
Chia-Yi University,
Chia-Yi, Taiwan,
R.O.C.

Ming-Zong Zheng
Taiwan Water Supply
Corporation, Chia-Yi,
Taiwan, R.O.C

kychen@mis.sjsmit.edu.tw

yung@ mis.sjsmit.edu.tw

annie@mail.ncyu.edu.tw

cmd82110@yahoo.com.tw

Abstract

Owing to unsuitable land-development of the upstream of Ba-Zhang river and improper land-management of the catchment (ex. land expropriation has not yet completed, buffer zone of reservoir has not yet delimited etc.), the trapped deposits of Ren-Yi reservoir have become serious. It has been the fifth highest among the twelve off-stream reservoirs, since it operated in 1987. To solve above problems, the management of the cadastral data around the reservoir catchment becomes more important. The conventional operation of the authority treats the graphic database and attribute database separately. It is time consuming and can't provide the information immediately and precisely.

The Geographic Information System (GIS) is widely used to tackle spatial data. This study focuses on an application of GIS to the Ren-Yi reservoir watershed for establishing cadastral database of this area. It is good for reservoir watershed management and for the references of the decision maker.

Keyword: GIS, Buffer zone, Cadastral data

1. Introduction

The Ren-Yi Reservoir sets up in 1980, complete work in June of 1987, is a public water supply in one target to leave the slot reservoir, gather the water area about 3.6 square kilometers of, gather the area of water and protections to bring to all plant the high and economic farm crop and few miscellaneous woods wood inside most lands, the agrochemical usage rate is high, because the topsoil often turns over to move, meeting the pouring rain to clip to bring a great deal of sediment and agrochemicals to flow into the reservoir namely each time, plusing the Ba-Zhang river upper stream lands exploitation, making lead the water it contains the amount of sand high, make Ren-Yi Reservoir sludge high to reside the whole province the fifth, therefore the valid land management method is one of the initial works of the current management unit.

For managing the reservoir to gather the land exploitation method of the area of water effectively, safeguard the reservoir fluid matter and extension reservoir lifes, gather reservoir to watershed's cadastral data electronically, in really current urgent key job.

This research regards Ren-Yi Reservoir as to study the district, regarding geography information system as the tool, expecting the result that reaches as follows:

a. Establishing the cadastral data around the reservoir and related geography environment databases of the watershed.

b. Carry on the search, analysis and statistics and display the cadastral data of watershed through the geography information system, solve at the present time in handling by artificial, to raise the efficiency of cadastral data on the management.

c. For reducing the mire sand or pesticide direct afflux reservoirs, Make use of the spatial analytical function of GIS to delimited the buffer zone of reservoir.

2. Environment of Study Area

The Ren-Yi Reservoir is resided in southeast square in Chia-Yi City about 6 kilometers of, the whole area belongs to the Nai-On village inside the Fan-Ru country of Chia-Yi County.

The Ren-Yi Reservoir is located in valley marsh land of the Ba-Zhang river right bankses, the geography is low to sink the similar basin.

The geography of the reservoir then tilt to one sides to the south by the north, equally slope about 0.21, become the circular slightly, is to belong to the mound geography.

Figure 1 is satellite image and DTM data set after folding it as a result, can know to observe the district hinterland form to rise and fall the variety.

3. Study Method

Section 3.1 will introduce the study tools, and Section 3.2 will introduce the procedure of this study.

3.1 Study Tools

This research makes use of the geography environment database, cadastral database and spatial analysis by the geography information system software ArcView and ARC/ INFO software to establish the cadastral data of reservoirs, with the understanding and valid the management gathers the cadastral data of the land inside the watershed, to prolong the reservoir life and raise the precious resource of water.

3.2 Procedure

Adhere to the process of this study, the detailed implement step is as follows:

a. Establishment the geography environment database.

Making use of the geography information system software digitizes 1/5000 it be like a basic diagram, and obtain the information that basic diagram layer data need when extracting by the parties concerned, digitizing to establish the geography environment database, including the geography position diagram of watershed and the watershed geography diagram, geology diagram, soil diagram, land's owner diagram, land usage distinction diagram, land to make use of the limit diagram, land to make use of the current conditions diagram etc..

b. Establishment the spatial and property of cadastral data in watershed.

Current cadastral data, be divided into the cadastral paper diagram and writing datas two kinds of, but because the cadastral data quantity of this research demand is hugein addition to purchasing whole Fan-Ru country cadastral paper diagram from the application of Tai-Chung Land Survey Bureau, needing to face again Ju-Chi land office application to purchase the writing data. Data after compile, the establishment of spatial and property data and the process of the mergers are as follows:

(a)Digital cadastral diagram

The cadastral diagram that application purchase, after tidy up, by the software of ARC/ INFO, digital the scope of each data, the establishment becomes the electronics cadastral diagram, and input the serial number in cadastral data column, with the conduct and actions cadastral data to merge hour it common column.

(b) Tidy up the property of cadastral data

Spatial and property of cadastral data after merged, fold the analysis with the set of the GIS software, can immediately acquire the Ren-Yi reservoir to gather the cadastral diagram of watershed.

4. Result and Discussion

4.1 Cadastral database System of Ren-Yi reservoir

The cadastral data of Ren-Yi reservoir is located the Fan-Ru country of Chia-Yi County inside, as the figure 2 showed, the green part in diagram is a cadastral data of Nai-On village, the pink is part of Ren-Yi reservoir cadastral data in the scope.

It including in the property form of cadastral data diagram: the area, ground number, country name, district name, land type, ownership person's name, ownership person's ID NO., ownership person's address, telephone number, price of imposition, announces price, whether the imposition, property right belongs to the category etc. 13 columns of data.

4.2 The reservoir drowns the land imposition situation of the area

The reservoir drowns area is a full water level in reservoir district in the line, have land 1054 totally in this district, area about 2.46 square kilometers of, usually the reservoir drown area and should carry out the imposition all.

But data in manifestation, already by there is totally 869 in the land of imposition by water supply corporation, total area about 2.23 square kilometers of, have the 90.17% in total area, not yet imposition land then contain 159, total 0.19 square kilometers of, have the 7.84% in total area, the land that did not issue accounts 26, the area have 0.049 square kilometers of, account for 1.99% of total area, above and various area it distribute and content are shown in figure 3.

4.3 The leasing and the indemnity of Ren-Yi reservoir buffer zone

When reservoir full water level, water would by the capillary function rising break the root-zone of crops, causing the fruit tree to lose plant, make water supply cooperation every year all with leasing and indemnity method, subsidize the loss of the landholding person, let the water supply cooperation stir up a few harassments.

From 1987 to 2003, the reservoir drowns the land rental capital sum of the area neighborhood as NT$ 225,732 dollars. The building injures to compensate total amount of money as NT$ 1,288,929 dollars.

Total 16 years water supply cooperation is within the scope of full water level in reservoir 0.5 meters of to the leasing of the landholding person and compensate total amount of money as NT$ 1,514,661 dollars.

This research with a reference that fold analyze, distinguish calculate 30 meters of and 50 meters of imposition in scopes area (as figure 4 show) outside the full water level line and amount of moneys with provide manage unit in delimitation reservoir protection bring and carry out the related affair that land collect of the geography information system according to.

4.4 The valuation of protection bring zone

With the GIS analytical function, delimit 30 meters of reservoir protections to bring, have land 293 totally inside this district, the area is total 473,136 square meters of, need to be imposed private to own the land total 202, the area is

373,026.28 square meters of, the amount of money of imposition is NT$176,052,765 dollars, detailed as table 1 shown.

Same with the GIS analytical function, delimit 50 meters of reservoir protections to bring, have land 299 totally inside this district, the area is total 650,603 square meters of, need to be imposed private to own the land total 206, the area is 526,161 square meters of, the amount of money of imposition is NT$248,749,898 dollars, detailed as table 2 shown.

5. Conclusion and Future Work

The geography information system is the best tool of cadastral data management, the reservoir cadastral database that build to complete can be provided as search, statistics, demonstration and analysis etc. function, can solve the current management unit by artificial of homework, promote the cadastral data to manage the effect of the work significantly to also help normal operation of the reservoir.

Cadastral data through fold function of GIS, can search the other geography environment information of the cadastral record's data, for example fold with soil diagram set behind, then search that cadastral record.

The reservoir not yet delimits the scope for protect bring, only with the full water level is on-line 0.5 meters of scope for the buffer zone, because the water will break by the capillary function rising privately of root-zone in crops, cause the crops to lose plant. So the water supply corporation from 1987 to 2003 with leasing and indemnity of way spent NT$ 1,514,661 dollars totally. If delimit 30 meters of protections to bring the scope, need to cost amount of money as NT$176,052,765 dollars, if bring the scope with 50 meters of delimitation protections, need to cost amount of money as NT$248,749,898 dollars.

For letting the cadastral data and current conditions of the data to agree with, the follow-up research will make use of the satellite image and GPS systems, increasing the exploitation current conditions of set up each land, toing reach to gather the land inside the area of water to make use of it supervise and control, in order to prevent break the law the matter of develop the exploitation, to insure normal operation of the reservoir.

References

[1]G. Y. Chen, Integration of GIs, Remote sensing and Hydrological model for Runoff Simulation, Asian Institute Technology, Bangkok, Apr. 1996.

[2]Burrough P. A. Principles of GIS for land Resources Assessment, Clarendon press, Oxford, 1986.

[3]Bill Johnson and Marina Oliver, Digital Resources for Managing Wetlands: The Case of Playa Lake research - A Preliminary Report, 1997, Texas Tech. University.

[4]G. E. Hollis and J. R. Thompson, Hydrological Data for Wetland Management, Journal of the Chartered institution of Water and Environment Management 12(1),PP 9-17, Feb. 1995.

[5]Lemly, D. A. and Richardson, Guideline for Risk Assessment in Wetlands, Environmental Monitoring and Assessment, PP 117-134, 1997.
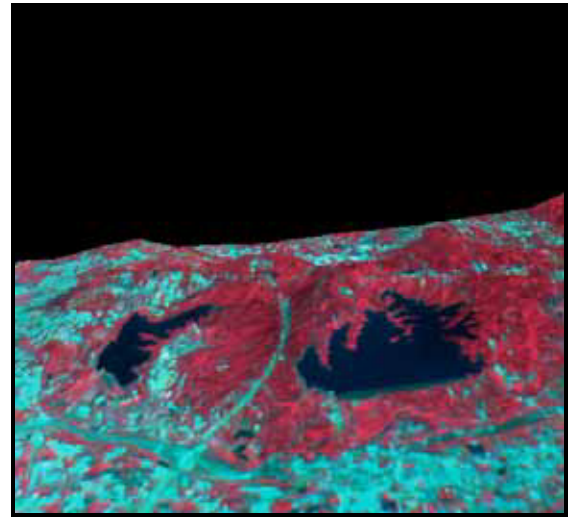
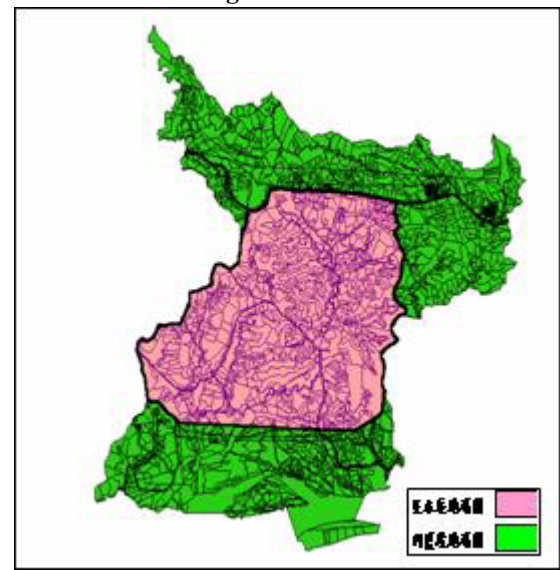**Fig. 1 The satellite image and DTMs turn stereoscopic diagram that draw**

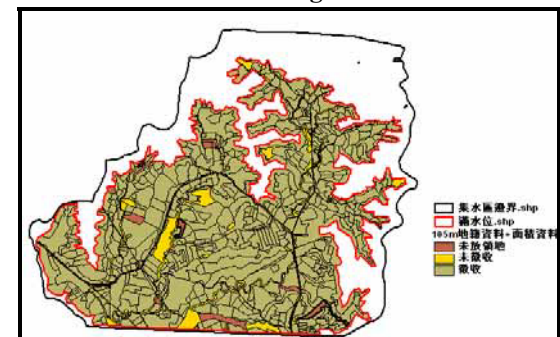**Fig. 2 cadastral data of Ren-Yi reservoir and Nai-On village**

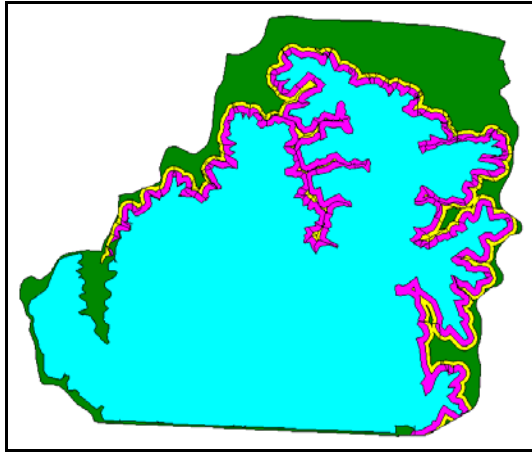**Fig. 3 The diagram of reservoir drowns the land imposition situation**

Fig. 4 The Ren-Yi reservoir protection brings the scope diagram

Table 1 The imposition general situation form of various land of 30 meters of protections bring zone in Ren-Yi reservoir

Item	No. of item	Total area(m²)	Number of imposition		Area(m²) of imposition		Cost(NT$ dollar) of imposition (Not yet imposition)
			imposition	Not yet imposition	Imposition	Not yet imposition	
Water	16	2958.12	2	14	101.38	2856.74	490,711.0
Not used	5	4291.8	5	0	4,291.80	0.00	0.0
Paddy Field	51	23,494.83	25	26	11,085.59	12409.24	5,892,626.1
Upland Field	64	165,299.17	19	45	28,066.7	137232.47	64,530,064.2
Forest	133	270,930.65	34	99	55,994.85	214935.80	100,969,354.0
Temples	1	332.11	0	1	0.00	332.11	36,532.1
Building	10	4,840.72	2	8	316.51	4524.21	3,908,510.0
Road	13	987.15	4	9	251.44	735.71	224,968.2
Sum	293	473,134.55	91	202	100,108.27	373026.28	176,052,765.6

Table 2 The imposition general situation form of various land of 50 meters of protections bring zone in Ren-Yi reservoir

Item	No. of item	Total area(m²)	Number of imposition		Area(m²) of imposition		Cost(NT$ dollar) of imposition (Not yet imposition)
			imposition	Not yet imposition	imposition	Not yet imposition	
Water	17	3514.54	2	15	189.42	3325.12	681,722.6
Not used	5	5643.85	5	0	5643.85	0.00	0.0
Paddy Field	51	30569.51	25	25	11627.22	18942.29	8,963,159.6
Upland Field	63	233967.70	19	44	41352.71	192614.99	90,559,848.6
Forest	140	368732.27	35	105	64620.78	304111.49	142,830,696.7
Temples	1	332.11	0	1	332.11	0.00	36,532.1
Building	9	6025.88	2	7	316.51	5709.37	5,153,741.0
Road	13	1817.40	4	9	359.32	1458.08	524,198.1
Sum	299	650603.26	92	206	124441.92	526161.34	248,749,898.7

A Preliminary Study of an Intelligent Self Diagnostics System

Chang-Ching Lin, Bruce Huang, Tien-Lun Liu

Department of Industrial Engineering and Management
St. John's and St. Mary's Institute of Technology
No. 499, Sec 4, Tamking Rd., Tamshui, Taipei, Taiwan
tclin@mail.sjsmit.edu.tw

Abstract—*In this paper artificial neural network (ANN) technologies and analytical models have been investigated and incorporated to increase the effectiveness and efficiency of machinery self diagnostics system. Several advanced vibration trending methods have been studied and used to quantify machine operating conditions. An on-line, multi-channel condition monitoring procedure has been developed and coded. The major technique used for self diagnostics is a modified ARTMAP neural network. The objective is to provide a possible solution for condition-based intelligent self diagnostics system.*

Index Terms—*Intelligent System, Self Diagnostics, Artificial Neural Network, Condition Based Maintenance.*

I. Introduction

In this paper, an integrated Intelligent Self Diagnostics System (ISDS) based on real-time, multi-channel and neural network technologies is introduced. It involves intermittent or continuous collection of vibration data related to the operating condition of critical machine components, predicting its fault from a vibration symptom, and identifying the cause of the fault. Referring to Figure 1, ISDS contains two major parts: the condition monitoring system and the self diagnostic system. The fault diagnostic system is based on the ARTMAP fault diagnostic network developed by Knapp and Haung [1, 2, 3]. The ARTMAP network is an enhanced model of the ART2 neural network [4, 5].

II. Condition Monitoring System

The condition monitoring system developed contains four modules (see Figure 1): data acquisition, Parameters Estimation (PE), Performance Monitoring (PM), and Information Display and Control (IDC). The entire system was coded using C programming language. We have developed a user friendly graphic interface that allows for easy access and control in monitoring an operating machine. The system has been tested and verified on an experimental lab settings. The detailed procedure of ISDS and programming logic is discussed in the following sections.

A. Data Acquisition Module

The data acquisition module is more hardware related than the other modules. Vibration signals were acquired through accelerometers connected to a DASMUX-64 multiplexer board and a HSDAS-16 data acquisition board installed in a PC compatible computer. The multi-channel data acquisition program controlling the hardware equipment has been coded.

B. Programming Logic for Parameter Estimation (PE) Module

The parameter estimation module is designed to estimate the parameters of the normal condition of a machine. It provides a procedure to set up the machine positions considered to be critical locations of the machine. The PE module must be executed before running the PM module. The information to be calculated in the PM module needs to be compared to the base-line information generated in the PE module.
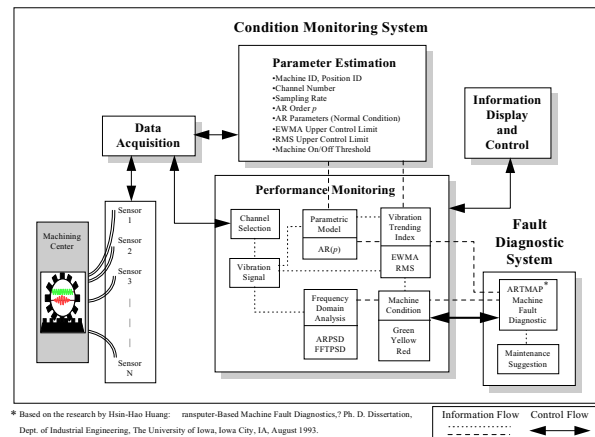

Figure 1. Overview of ISDS

The normal operating condition of a machine position is usually defined by experience or from empirical data. However, this normal condition is not unchangeable. Any adjustment to the machine, such as overhaul or other minor repairs, would change its internal mechanisms. In this case, the normal condition must be redefined, and all the base-line data of the monitored positions on the machine need to be reset.

The PE procedure starts with specifying the ID of a machine, its location ID, and several other parameters related to each position, such as channel number and sampling rate. Then the upper control limits of the Exponentially Weighted Moving Average (EWMA) [6] and Root Mean Square (RMS) [7, 8] vibration trending indices are determined and an adequate Autoregressive (AR) order is computed. The AR time series modeling method is the most popular parametric spectral estimation method which translates a time signal into both frequency domain and parameter domain [9]. Once the AR order is determined, the AR parameters can be

estimated through several normal condition signals collected from the particular position. A major issue with the parametric method is determining the AR order for a given signal. It is usually a trade-off between resolution and unnecessary peaks. Many criteria have been proposed as objective functions for selecting a "good" AR model order. Akaike has developed two criteria, the Final Prediction Error (FPE) [10] and Akaike Information Criterion (AIC) [11]. The criteria presented here may be simply used as guidelines for initial order selection, which are known to work well for true AR signals; but may not work well with real data, depending on how well such data set is modeled by an AR model. Therefore, both FPE and AIC have been adapted in this research for the AR order suggestion.

A setup file is then generated after the PE procedure is completed. This file, given a name that combines the machine ID and the position ID, consists of all the parameters associated with the specific position. The number of setup files created depends on the number of positions to be monitored in the PM mode, that is, each monitored position is accompanied by a setup file.
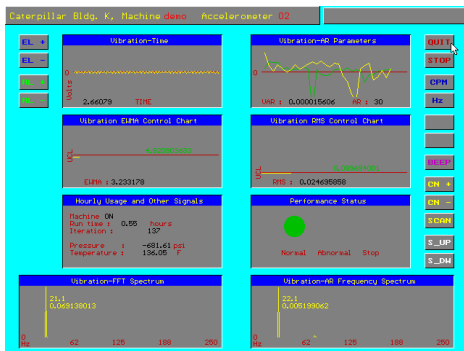

Figure 2. Condition Monitoring Information Display and Control (IDC) Screen Layout

In order to perform a multi-channel monitoring scheme a setup log file is also generated. This file contains all the names of setup files created in the PE mode. Every time a new position is added its setup file name is appended to the setup log file. The setup log file is very important. It not only determines the channels needing to be scanned when the PM program is executed, it also provides the PM program with paths to locate all the necessary information contained in the setup files. After the PE procedure is completed, on-line performance monitoring of the machine (the PM mode) begins.

C. Programming Logic for Performance Monitoring (PM) Module

In the PM module, vibration data arrive through the data acquisition hardware and are processed by AR, EWMA, ARPSD, RMS, FFT spectrum, and hourly usage calculation subroutines. After each calculation the current result is displayed on the computer screen through the Information Display and Control (IDC) module.

IDC is in charge of functions such as current information displaying, monitoring control, and machine status reasoning.

D. Information Display and Control (IDC) Module

As shown in Figure 2, eight separate, small windows appear on the computer screen when the IDC module is activated. Each

window is designed to show the current reading and information related to each calculation subroutine (e.g. AR, EWMA, ARPSD, RMS, and FFT spectrum) for the current position being monitored.

Window 1 is designed to plot the current time domain data collected from the data acquisition equipment. Window 2 displays both the AR parameter pattern of the current signal and the normal condition AR parameter pattern stored in the setup file generated in the PE module. Window 3 plots the current EWMA reading on a EWMA control chart and its upper control limit. Window 4 plots the current RMS value and its upper control limit on a RMS control chart. Both the RMS and EWMA upper control limits are calculated in the PE module. Window 5 displays the hourly usage and other information of the position. Window 6 indicates the current performance status of the position. Three different levels of performance status: normal, abnormal, and stop, are designed. Each status is represented by a different color: a green light signals a normal condition; a yellow light represents an abnormal condition; and a red light indicates an emergency stop situation. The determination of the status of a position based on the current readings is discussed in the next section. Window 7 gives the current ARPSD spectrum, which is calculated based on the AR parameters from Window 2. Finally, Window 8 displays the current FFT spectrum by using the time domain data from Window 1.

E. Vibration Condition Status Reasoning

Based on the criteria stored in the setup file and the current readings, the EWMA and RMS control charts show whether the current readings are under or above their respective upper control limit. If both readings are under their corresponding control limits, then the position is in a normal condition. However, if either one of the control readings exceeds its upper control limit, the performance status reasoning program would turn on the yellow light to indicate the abnormality of the position. In this case, the fault diagnostic system is activated.

F. Condition Monitoring Sample Session

Data collection, in the form of vibration signals, was conducted using the following test rig (see Figure 3): a 1/2 hp DC motor connected to a shaft by a drive belt, two sleeve bearings mounted on each end of the shaft and secured to a steel plate, an amplifier to magnify signals, a DASMUX-64 multiplexer board, and a HSDAS-16 data acquisition board installed in a personal computer. Vibration signals were collected from the bearing using 328C04 PCB accelerometers mounted on the bearing housings. Using the test rig, the following sample session was conducted.

Assume that when the motor was turned on initially, it was running in normal condition. Later, a small piece of clay was attached to the rotational element of the test rig to generate an imbalance condition. This was used as an abnormal condition in the experiment. In the beginning, the setup procedure (PE) needed to be performed in order to obtain the base-line information. The sampling rate used was 1000 Hz and the sampling time was one second. The PE program first acquired eight samples and then took their average. Using the average normal signal, the AIC and FPE criteria were calculated. An AR order suggestion for the normal condition of the test rig was made. The AR order was fixed throughout the entire experiment. Once the AR order was known, the program started estimating the AR parameters and upper

control limits of RMS and EWMA by collecting another eight data sets, calculating eight sets of AR parameters, and then averaging them. Finally, all parameters were stored in the setup file which would be used in the PM stage. An example of the normal condition parameters from a setup file are listed below:

- Machine ID: TESTRG
- Position ID: CHN1
- Channel number: 1
- Sampling rate: 1000
- AR order: 32
- AR parameters: ....
- $EWMA_{UCL}$: 0.8912
- RMS_{UCL}: 0.0367

When the machine was running in normal condition the readings of EWMA were approximately -0.486 far below the $EWMA_{UCL}$ of 0.8912. The readings of RMS were about 0.01895, and therefore, they were below the RMS_{UCL}. As soon as an imbalance condition was generated the EWMA and RMS readings jumped to values of 3.3259 and 0.0504, respectively. The EWMA and RMS readings indicated the test rig was in an abnormal condition since both readings exceeded their respective control limits.

The machine condition monitoring mode switches to self diagnostic mode when at least one index exceeds its control limit. Once the system is in the diagnostic system, a detailed automatic analysis begins to identify the machine abnormality occurred. The next section explains the fault diagnostic system designed for this research.
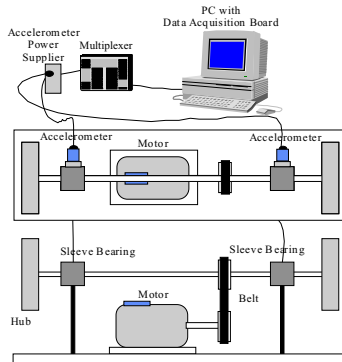

Figure 3. The Test Rig for ISDS Experiment

III. ARTMAP-Based Self Diagnostic System

A. Introduction to ARTMAP Neural Network

The self diagnostic system in this paper employs a neural network architecture, called Adaptive Resonance Theory with Map Field (ARTMAP). The ARTMAP network is developed as an extension of the ART neural network series [5]. The ARTMAP learning system is built from a pair of ART modules, which is capable of self-organizing stable recognition categories in response to arbitrary sequences of input patterns [3].

A modified ARTMAP architecture has been adopted in this paper in order to perform the supervised learning. The modified ARTMAP architecture is based on the research by Knapp and Huang, which replaces the second ART module by a target output pattern provided by the user [1, 3]. The major difference between

the modified ARTMAP network and the ART2 network is the modified ARTMAP permits supervised learning while ART2 is an unsupervised neural network classifier.

B. Performance Analysis of ARTMAP-Based Self Diagnostic System

The performance of the ARTMAP-based self diagnostic system was validated by employing vibration signals from test bearings. The two sleeve bearings were replaced by two ball bearings with steel housings. The new setup allows easy detachment of the ball bearing from the housing for exchanging different bearings.

Six bearings with different defect conditions were made. Table 1 describes these defective ball bearings. A two-stage vibration data collection was conducted for each bearing. Five sets of vibration signals were collected in the first batch, three sets in the second batch. A total of eight sets of vibration signals were collected under each defect. Therefore, there were a total of 48 data sets. All time domain vibration signals were transformed and parameterized through the ARPSD algorithm. The AR order used was 30. Thus, the dimension number for each AR parameter pattern was 31 (i.e., 30 AR parameters plus one variance). These 48 AR parameter patterns were used to train and test the ARTMAP-based self diagnostic system.

Table 1. Test Ball Bearings

Bearing #	Defect
1	Good bearing
2	White sand in bearing
3	Over-greased in raceway
4	One scratch in inner race
5	One scratch in one ball
6	No grease in raceway

Note that the 512 frequency components in each ARPSD spectrum were compressed to only 31 parameters in each AR model indicating the system dealt with a significantly reduced amount of data; this is extremely beneficial in real-time applications.

Figure 4 shows the plots of AR parameter patterns from the six defective bearings. The first column displays the six training patterns, which is the first one of the eight data sets from each bearing type. The second column illustrates some of the other seven test patterns, where the solid lines represent data from the first collection batch and the dotted lines are from the second batch. As can be seen from Figure 4, the profiles of the AR parameter patterns within each group are very similar. Only a few deviations can be seen between the first and second batches. The deviations come from the very sensitive but inevitable internal structure changes of the setup during the bearing attachment and detachment operations between the two data collections.

The experimental procedure began with using the first pattern of all the conditions for training and then randomly testing the other seven patterns. In addition, the modified ARTMAP network was designed to provide two suggested fault patterns (i.e., the outputs of the first two activated nodes from the F_2 field). Table 2 summarizes the test results on diagnosing the 42 test patterns. The

first column of Table 2 for each bearing type is the first identified fault from the network. It shows only 3 of the 42 test cases were mismatched in the first guess but they were then picked up correctly by the network in the second guess (see bold-face numbers in Table 2). Interestingly, these three mismatched patterns were from the second batch. If the profiles of Bearings 4 and 5 in the second batch (the dotted profiles in the second column of Figure 4) were compared, then one could see the test patterns of Bearing 4 from the second batch were much closer to the training pattern of Bearing 5 than that of Bearing 4. This is why the network recognized the test patterns of Bearing 4 as Bearing 5 in its first guess. These test results clearly display the capability and reliability of the ARTMAP-based self diagnostic system and the robustness of using AR parameter patterns to represent vibration signals. For the efficiency of the ARTMAP training, the training time of one 31-point AR parameter pattern was less than one second on a PC.

Table 2. Bearing Test Results of ARTMAP-Based ISDS

Pattern Number		Bearing Number											
		1		2		3		4		5		6	
Batch 1	1	Train		Train		Train		Train		Train		Train	
	2	1	3	2	6	3	1	4	2	5	6	6	2
	3	1	6	2	6	3	1	4	2	5	4	6	1
	4	1	6	2	6	3	1	4	2	5	4	6	2
	5	1	6	2	6	3	1	4	2	5	6	6	1
Batch 2	1	1	3	2	6	3	1	**5**	**4**	5	4	6	5
	2	1	3	2	6	3	1	**5**	**4**	5	4	6	5
	3	1	3	2	6	3	1	**5**	**4**	5	4	6	5

IV. Summary and Conclusions

This paper presents an integrated Intelligent Self Diagnostics System (ISDS). Several unique features have been added to ISDS, including the advanced vibration trending techniques, the data reduction and features extraction through AR parametric model, the multi-channel and on-line capabilities, the user-friendly graphical display and control interface, and a unique machine self diagnostic scheme through the modified ARTMAP neural network.

Based on the ART2 architecture, a modified ARTMAP network is introduced. The modified ARTMAP network is capable of supervised learning. In order to test the performance and robustness of the modified ARTMAP network in ISDS, an extensive bearing fault experiment has been conducted. The experimental results show ISDS is able to detect and identify several machine faults correctly (e.g., ball bearing defects in our case).
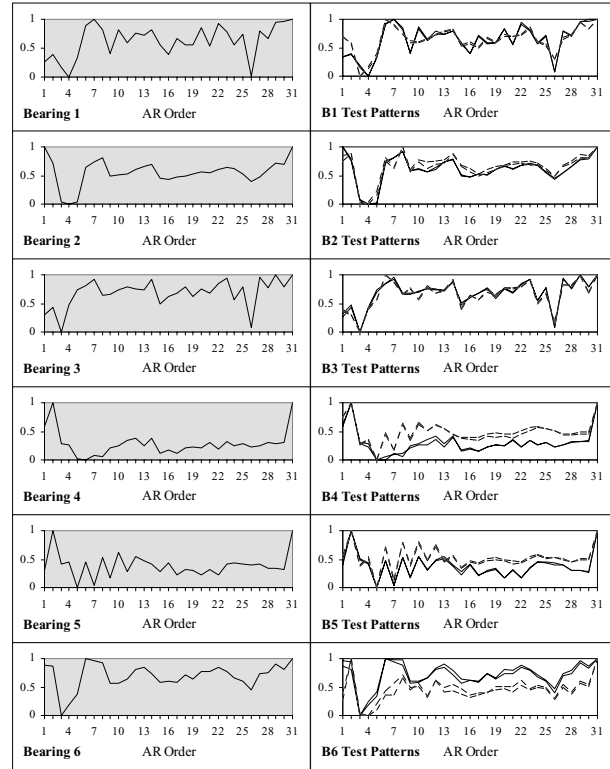

Figure 4. AR Parameters Patterns of Defective Bearings

REFERENCES

[1] Huang, H. -H., "Transputer-Based Machine Fault Diagnostics," *Ph.D. Dissertation, Department of Industrial Engineering, The University of Iowa*, Iowa City, Iowa, August 1993.

[2] Knapp, G. M. and Wang, H. -P., "Machine Fault Classification: A Neural Network Approach," *International Journal of Production Research*, Vol. 30, No. 4, 1992, pp. 811-823.

[3] Knapp, G. M., "Hierarchical Integrated Maintenance Planning for Automated Manufacturing Systems," *Ph.D. Dissertation, Department of Industrial Engineering, The University of Iowa*, Iowa City, Iowa, August 1992.

[4] Carpenter, G. A., & Grossberg, S., "ART2: Self-Organization of Stable Category Recognition Codes for Analog Input Patterns," *Applied Optics*, Vol. 26, No. 23, December 1987b, pp. 4919-4930.

[5] Carpenter, G. A., Grossberg, S., and Reynolds, J. H., "ARTMAP: Supervised Real-Time Learning and Classification of Non-stationary Data by a Self-Organizing Neural Network," *Neural Networks*, Vol. 4, 1991, pp. 565-588.

[6] Spoerre, J. K., "Machine Performance Monitoring and Fault Classification Using an Exponentially Weighted Moving Average Scheme," *Master's Thesis, Department of Industrial Engineering, The University of Iowa*, Iowa City, IA, May, 1993.

[7] Monk, R., "Vibration Measurement Gives Early Warning of Mechanical Faults," *Process Engineering*, November 1972, pp. 135-137.

[8] Wheeler, P. G., "Bearing Analysis Equipment Keeps Downtime Down," *Plant Engineering*, Vol. 25, 1968, pp. 87-89.

[9] Gersch, W. and Liu, T. S., "Time Series Methods for the Synthesis of Random Vibration Systems," *ASME Journal of Applied Mechanical*, Vol. 43, No. 1, 1976, pp. 159-165.

[10] Akaike, H., "Power Spectrum Estimation through Autoregression Model Fitting," *Ann. Inst. Stat. Math.*, Vol. 21, 1969, pp. 407-419.

[11] Akaike, H., "A New Look at the Statistical Model Identification," *IEEE Transactions on Automation Control*, Vol. AC-19, December 1974, pp. 716-723.

An Intelligent Fuzzy Controller for Perturbed Time-Delay Systems with Nonlinear Input

Tsai-Cheng Li[1], Fang-Ming Yu[2], and Huan-Wen Wu[3]

[1]*St. John's & St. Mary's Institute of Technology*
Department of Mechanical Engineering
li@mail.sjsmit.edu.tw

[2]*Kuang-Wu Institute of Technology*
Department of Electronic Engineering
[3]*Nan Jeon Institute of Technology*
Department of Mechanical Engineering

Abstract

In this article, an intelligent fuzzy controller, with single-input fuzzy logic control and self-tuning scheme, is proposed to cope with the problem of perturbed time-delay systems with nonlinear input efficiently. This method provides a simple way to achieve asymptotic stability. Other attractive features include the fuzzy rules been greatly reduced, as well as the insensitivity to the perturbation, especially the behavior of control input is dramatically improved. In addition, the method can handle the chattering problem inherent to the sliding mode control easily and effectively. Simulation results are presented to demonstrate the power of the method.

Key words: Fuzzy rules, asymptotic stability, perturbed time-delay systems

1. Introduction

In this paper, we propose an intelligent fuzzy controller, the self-tuning single-input fuzzy logic controller (ST-SIFLC) which satisfies the reaching condition $s\dot{s} < 0$, and only use the D_s, a signed distance to the sliding surface, as a fuzzy input variable. Thereby, the total number of rules of the system could be greatly reduced. Simultaneously, in light of the Lyapunov stability theorem, the system can thus be ensured to be asymptotically stable. The rule table is then constructed in a one-dimensional space and will be quickly organized and easily calculated. Furthermore, in order to get a better response of the system than that of pervious work [2, 5], a new kind of self-tuning algorithm for improving system performance by adjusting the scaling factor is proposed. The input scaling factor in a fuzzy control system is commonly used to conduct proper transformations between the real input data and pre-specified universe of discourses of the fuzzy input variables in the system. Theoretically, the scaling factor is a constant parameter. We derived a tuning scheme with only two rules of fuzzy inference and an ultimate equation from the result of defuzzification, this tuning algorithm is simple and convenient to use.

Moreover, most work of robust control has concentrated on the systems with uncertainties and/or external disturbances. The assumption of linear input is that the system model is indeed linearizable. However, in practice, due to physical limitation, there do exist

nonlinearities in the control input and their effect can not be ignored in realization. In addition, to obtain a more practical system model, the information of delay time should not be discarded. Thereby, it is necessary to develop a new robust control method to deal with the perturbed time-delay systems with input nonlinearity.

In this paper, an intelligent controller is derived through variable structure control (VSC) [2]. Although the method of variable structure control can solve the above-mentioned system, the drawback of the variable structure control is their chattering owing to the sliding control law that has to be discontinuous across the sliding surface. Moreover, if it is filtered at the output of the process, it may excite unmodeled high frequency modes, and may degrade the performance of the system and may even lead to instability. Although the FSMC method can be used to control nonlinear systems very well, they are with a linear input, not a nonlinear input and it can not be ignored in reality. Therefore, in this paper, we will adopt the FSMC with self-tuning single fuzzy input variable to design a robust controller for the perturbed time-delay system with a nonlinear input such that the behavior of the control input can be dramatically improved.

2. System Description

A general description of perturbed time-delay dynamical systems with nonlinear input [2] is

$$\dot{x}(t) = (A + \Delta A)x(t) + b\,\Phi(u) + A_d x(t - \tau_d) + f(t)$$
$$x(t) = \theta(t), \qquad -\tau_d \le t < 0 \tag{1}$$

where $x(t)$, b, and $f(t) \in R^n$ is the state variables, the input vector, and the disturbance vector, respectively. $u \in R$ is the control input of the system, $A \in R^{n \times n}$ is the state matrix, A_d is the delay term matrix including the uncertainty, and $\Phi(u): R \to R$ is a continuous function of nonlinear input and $\Phi(0) = 0$. ΔA is the bounded uncertainty matrix of A, τ_d represents a nonzero time delay, and $\theta(t)$ is a continuous vector-valued initial function. For dealing with the study of system (1), the following assumptions are taken:

Assumption 1. The matrix A_d and uncertainty matrices ΔA, f satisfy the following rank conditions [1].

$$\text{rank}[\Delta A] = \text{rank}[\Delta A \vdots b]$$
$$\text{rank}[\Delta A_d] = \text{rank}[\Delta A_d \vdots b]$$
$$\text{rank}[f] = \text{rank}[f \vdots b] \tag{2}$$

Based on **Assumption 1**, there exist vectors h, g, and a scalar d such that the following matching conditions hold [3].

$$\Delta A = bh \qquad \|h\| \le \beta_1$$
$$A_d = bg \qquad \|g\| \le \beta_2$$
$$f = bd \qquad \|d\| \le \beta_3 \tag{3}$$

The above matching conditions (3) are for simplicity of derivation.

Assumption 2. The nonlinear input $\Phi(u)$ applied to the system satisfies the following property [2]:

$$u \cdot \Phi(u) \ge \alpha \cdot u^2 \tag{4}$$

where $\alpha > 0$ and $\Phi(0) = 0$.

3. Variable Structure Control Design [2]

First, the following lemmas will be employed to derive the variable structure controller.

LEMMA 1 [2]. If the nonlinear input satisfies the property as indicated in eq. (4), there exists a continuous function $\phi(\cdot) : R_+ \to R_+$, $\phi(0) = 0$, and $\phi(p) > 0$ for $p > 0$. Therefore, if $|u(t)| = \phi(q)$, then

$$\alpha \cdot u^2 \ge q \cdot \phi(q), \quad \forall q \ge 0 \tag{5}$$

For system (1), the switching surface is defined as

$$s(t) = c^T x(t) = 0 \tag{6}$$

where $c \in R^n$ is a constant vector.

LEMMA 2 [2]. The motion of the sliding mode (6) is asymptotically stable, if the following condition holds

$$s\dot{s} < 0, \quad \forall t > 0 \tag{7}$$

To fulfill the condition stated in eq. (7), the desired variable structure control is suggested by

$$u(t) = -\frac{sc^T b}{\|sc^T b\|} \phi(x,t) \tag{8}$$

where $\phi(x,t) = \frac{r}{\alpha}\{[\|(c^T b)^{-1} c^T A\| + \beta_1]\|x\| + \beta_2\|x_d\| + \beta_3\}$

$,r > 1 \tag{9}$

4. The Intelligent Fuzzy Controller

A. Single- input fuzzy logic control (SIFLC)

The switching line for a second-order system (1) is defined by:

$$s : \dot{x} + c_1 x = 0 \tag{10}$$

First, we introduce a new variable called the signed distance. In Fig. 1, the distance between $A(x,\dot{x})$ and $B(x_1, \dot{x}_1)$ can be expressed by the following equation:

$$d_1 = [(x - x_1)^2 + (\dot{x} - \dot{x}_1)^2]^{1/2} = \frac{|\dot{x}_1 + c_1 x_1|}{\sqrt{1 + c_1^2}} \tag{11}$$

The signed distance d_s is then defined as follows:

$$d_s = \text{sgn}(s)\frac{|\dot{x} + c_1 x|}{\sqrt{1 + c_1^2}} = \frac{\dot{x} + c_1 x}{\sqrt{1 + c_1^2}} = \frac{s}{\sqrt{1 + c_1^2}} \tag{12}$$

where

$$\text{sgn}(s) = \begin{cases} 1 & \text{for } s > 0 \\ -1 & \text{for } s < 0 \end{cases} \tag{13}$$
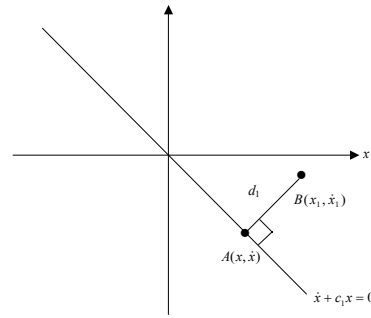

Fig.1. Derivation of a signed distance

For the nth-order system (1), eq. (6) can be represented as:

$$s = x^{(n-1)} + c_{n-1}x^{n-2} + \cdots + c_2\dot{x} + c_1 x = 0. \tag{14}$$

Hence, the general signed distance D_s is changed to be:

$$D_s = \frac{x^{(n-1)} + c_{n-1}x^{n-2} + \cdots + c_2\dot{x} + c_1 x}{\sqrt{1 + c_{n-1}^2 + \cdots + c_2^2 + c_1^2}}$$
$$= \frac{s}{\sqrt{1 + c_{n-1}^2 + \cdots + c_2^2 + c_1^2}}. \tag{15}$$

By the time derivative of both sides of (6), we obtain

$$\dot{s} = c^T \dot{x} = c^T A x + c^T b \Phi(u) + c^T b(hx + gx_d + d) \tag{16}$$

Then yielding

$$s\dot{s} = sc^T A x + sc^T b \Phi(u) + sc^T b(hx + gx_d + d) \tag{17}$$

Here, we consider that $c^T b > 0$. In (16), it is seen that $\dot{s}$ increases as $\Phi(u)$ increases and vice versa. In (17), it is seen that if $s > 0$, then decreasing $\Phi(u)$ will make $s\dot{s}$ decrease and that if $s < 0$, then increasing $\Phi(u)$ will make $s\dot{s}$ decrease. From the **Assumption 2**, we know that $u \propto \Phi(u)$. So, it is seen that if $s > 0$, then decreasing u will make $s\dot{s}$ decrease and that if $s < 0$, then increasing u will make $s\dot{s}$ decrease. Now, we choose a Lyapunov-like function $V = \frac{1}{2}D_s^2$

Then

$$\dot{V} = D_s \dot{D}_s = \frac{s\dot{s}}{1 + c_{n-1}^2 + \cdots + c_2^2 + c_1^2} \qquad (18)$$

Hence, it is seen that if $s > 0$, then $D_s > 0$, decreasing u will make $s\dot{s}$ decrease so that $\dot{V} < 0$ and that if $s < 0$, then $D_s < 0$, increasing u will make $s\dot{s}$ decrease so that $\dot{V} < 0$. So we can ensure that the system is asymptotically stable. From the above relation, we can conclude that:

$$u \propto -D_s \qquad (19)$$

Hence, the fuzzy rule table can be established as shown in Table 1. The triangular type membership function is chosen for the aforementioned fuzzy variables, as shown in Fig. 2.
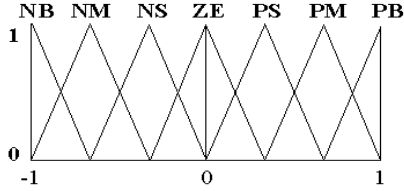

Fig. 2. Fuzzy variable of triangular type

Table 1. Rule table for SIFLC

D_s	NB	NM	NS	ZE	PS	PM	PB
u	PB	PM	PS	ZE	NS	NM	NB

The SIFLC is shown in Fig. 3. where D_s and u are the input and output of the single-input fuzzy logic control, respectively. The input of the proposed fuzzy controller is $\tilde{D}_s$, which is a fuzzified variable corresponding to D_s. The output of the fuzzy controller is U, which is the fuzzified variable corresponding to u. All the universes of discourse of $\tilde{D}_s$ and U range from -1 to 1. Thus, the range of nonfuzzy variables D_s and u must be scaled to fit the universe of discourse of a fuzzified variable $\tilde{D}_s$ and U with scaling factors K_1 and K_2 respectively, namely,

$$\tilde{D}_s = K_1 \cdot D_s(t) \qquad (20)$$
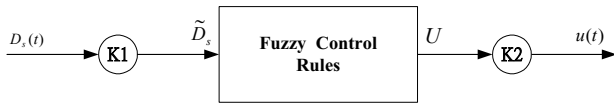$$u(t) = K_2 \cdot U \qquad (21)$$

Fig. 3. The block diagram of the SIFLC

B. *Self-tuning scheme and scaling factor*

In this section we will introduce the self-tuning scheme because the fuzzy system performance is sensitive to the scaling factors K_1. If large values of K_1 are available the system steady tracking error will be reduced but the overshoot may increase because large K_1 provide large control gain. Conversely, if small values of K_1 are chosen, the tracking accuracy will be degraded, leading to the small

control gain. So we propose to decrease K_1 when error is large, and vice versa. The fuzzy rule for tuning K_1 can be formalized as

• If error is large (EL) then scaling factor is small (CS)
• If error is small (ES) then scaling factor is large (CL).

By defining the following membership function:

$$\mu_{ES} = 2 / [\exp(-D_s) + \exp(D_s)] = 1 / \cosh(D_s),$$
$$\mu_{EL} = 1 - \mu_{ES},$$
$$\mu_{CL} = \begin{cases} 1, & K_1 = K_{1,\max}, \\ 0, & K_1 \neq K_{1,\max}, \end{cases} \quad \mu_{CS} = \begin{cases} 1, & K_1 = K_{1,\min}, \\ 0, & k_1 \neq K_{1,\min}, \end{cases} \qquad (22)$$

and using the max-min inference for defuzzification the value of K_1 can be found as

$$K_1 = K_{1,min} + \Delta K_1 / \cosh(D_s) \qquad (23)$$

where $\Delta K_1 = K_{1,\max} - K_{1,\min}$.
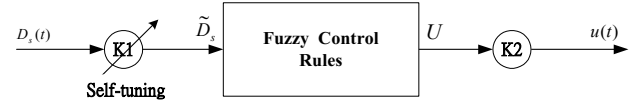
Fig. 4 shows ST-SIFLC controller.

Fig. 4 The block diagram of the ST-SIFLC

5. Simulation

Consider a perturbed time-delay system with nonlinearity shown below

$$\dot{x}(t) = (A + \Delta A)x(t) + b\Phi(u) + A_d x(t - \tau_d) + f(t)$$
$$= Ax(t) + b\Phi(u) + b[hx(t) + gx_d + d]$$

where $x_d = x(t - \tau_d)$, $\|h\| \leq \beta_1$, $\|g\| \leq \beta_2$ and $\|d\| \leq \beta_3$. From eq. (3), we also obtain $\Delta A = bh$, $A_d = bg$, and $f = bd$. The corresponding parameters are

$$A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}, b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, x_d = \begin{bmatrix} x_1(t - \tau_d) \\ x_2(t - \tau_d) \end{bmatrix},$$

$$\Phi(u) = (\delta\, e^{\lfloor \sin u \rfloor} + \gamma \cos u)u, \ \delta > \gamma > 0, \text{ and}$$

$$hx + gx_d + d = l_1 e^{(1+\sin x_1)}\sqrt{x_1^2 + x_2^2} + l_2 \cos x_{2d}\sqrt{x_{1d}^2 + x_{2d}^2} + l_3 \cos x_2$$

The switching surface is taken as $s(t) = 2x_1 + x_2$. Obviously, $c^T b > 0$. This meets the condition $c^T b > 0$ in (17). The initial values are arbitrarily chosen as $x_1(0) = -1$, $x_2(0) = 1$ and the numerical values are $\delta = 1.0$, $l_1 = 0.04$, $l_2 = 0.3$, $l_3 = -0.6$, and $\gamma = 0.3$.

A. VSC method [2]

For the computer simulation, the corresponding parameters of VSC method are $r = 1.1$, $\alpha = 0.7$, $\beta_1 = 0.3$, $\beta_2 = 0.3$, $\beta_3 = 0.3$, respectively. The delay time constants are taken as $\tau_d = 0.2$.

B. FSMC method [5]

Fig. 5 shows the block diagram of the FSMC [5]. The

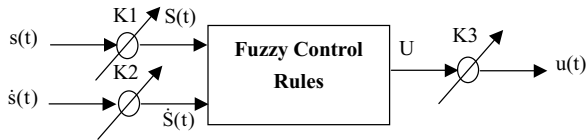scaling factors are chosen as K_1=0.5, K_2=0.5 and K_3=50.

Fig. 5   The block diagram of the FSMC [5]

C. Intelligent fuzzy controller, ST-SIFLC method

For ST-SIFLC method, the rule base is a one-dimension table as in Table 1. The scaling factor K_1 is variable due to the self-tuning scheme. Furthermore, $k_{1\min} = 0.1, k_{1\max} = 1.1$ and the other scaling factor is chosen as K_2=25. The computer simulation sampling steps for the perturbed time-delay system with nonlinearity in above three methods are 0.01 seconds.

Fig. 6 shows the phase plane of x_1 and x_2, it is observed that the sliding surface of ST-SIFLC is modified successively. The control input u is shown in Fig. 7. It is clear that the control signal of ST-SIFLC is dramatically reduced and all of the chattering has been eliminated. This demonstration shows that the proposed technique serves as a perfect control input behavior with high efficiency.
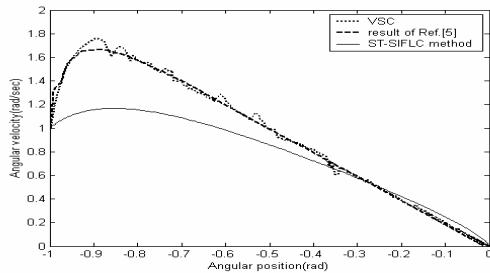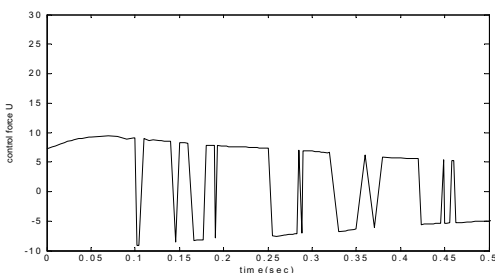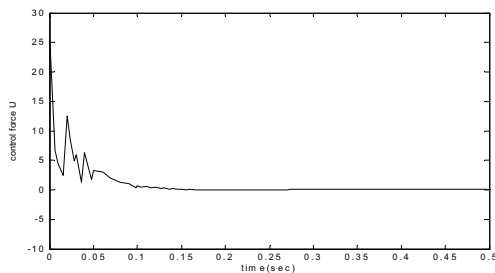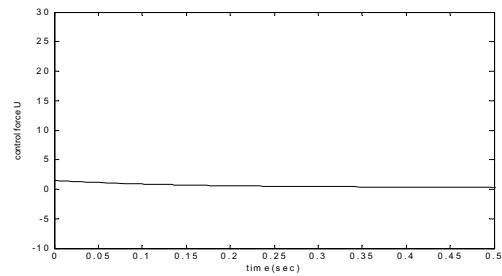

Fig. 6 The phase plane between x_1 **and** x_2

(a)

(b)

(c)

Fig. 7 The control force *u*: (a)VSC, (b)FSMC [5], and (c) The proposed method of ST-SIFLC

6. Conclusions

Several important results of the proposed technique are summarized in the following:
(1)   We don't require an exact mathematical model of the nonlinear systems for the controller design. The only information needed to design the controller is the influences of *u* with respect to D_s.
(2)   An inherent chattering problem of sliding mode control can be eliminated effectively without involving sophisticated mathematics.
(3)   This new kinds of self-tuning scheme in fuzzy logic controller, which improves the performance of chattering elimination and increases the efficiency of the control input by adjusting the scaling factor, and the asymptotic system stability can be guaranteed..

References

[1]   N. Luo and M. De La Sen, "State feedback sliding mode control of a class of uncertain time delay systems", *IEE Proceeding part D,* vol. 140, pp. 261-274, 1993.

[2]   K. C. Hsu, "Adaptive variable structure control design for uncertain time-delayed systems with nonlinear input," *Dynamics and Control*, vol. 8, pp. 341-354, 1998.

[3]   K. Shyu and J. Yan, "Variable-structure model following adaptive control for systems with time-varying delay", *Control theory and Advanced Technology,* vol. 10, pp. 513-521, 1994.

[4]   B. J. Choi, S. W. Kwak, and B. K. Kim, "Design of a single-input fuzzy logic controller and its properties," *Fuzzy Sets and Systems*, vol. 106, pp. 299-308, 1999.

[5]   F. M. Yu, H. Y. Chung, and S. Y. Chen, "Fuzzy sliding mode controller design for uncertain time-delayed system with nonlinear input," *Fuzzy Sets and Systems*, vol. 140, pp. 359-374, 2003.

MIDI-Based Audio Recording/Playing System with Smaller Data Quantity

Hsin-Chuan Chen and Chen-Chien Hsu

Abstract—**Conventional music CDs are produced by storing digital audio data files (wave files). High quality sound effect can be obtained when they are played at the cost of a huge data quantity. Alternate well-known digital music files, MIDI (Musical Instrument Digital Interface) files, have very small data size due to the fact that only the musical performance information is recorded. By separating vocal from music and using of a MIDI file that replace the wave data file of music, we propose an audio recording/playing system to allow more songs stored in a single CD than ever before, while maintaining satisfactory sound quality. The total data size of a song can be reduced significantly compared with that of the conventional music CDs.**

Index Terms— **MIDI**, **wave data**, **MIDI**, **synthesizer**, **audio recording/playing system.**

I. INTRODUCTION

In the production of conventional music CDs, the background music of a song is produced and recorded first, and then singer inputs his/her vocal associated with the background music, so as to record the vocal. Finally, two recorded sound sources including the background music and the vocal are mixed together to perform a complete song. For achieving a better sound quality during the recording of digital music CD, according to Nyquist Theorem, 44K sampling rate and 16-bit data resolution are required. However, higher sampling rate and data resolution mean larger digital data size (i.e. wave data) need to be generated. Consequently, there are few songs that can be stored in one compact disk, and these songs with wave data are not suitable for transmission on the Internet.

In the past, several researches are conducted to extract the information of notes such as pitch, length, and velocity from the music wave data, and then convert them to the MIDI data format [5][7]. However, due to complicated processes, such audio systems using extraction method are difficult to implement for the music played by multiple instruments or the audio combining music with vocal. Actually, most of the background music of songs is directly completed recording on the MIDI instrument devices, which can directly generate the corresponding MIDI data, thus it only use 11K sampling rate to record the vocal as a wave data file without re-recording the background music. In fact, using 11K sampling rate is good

Hsin-Chuan Chen and Chen-Chien Hsu are with the Department of Electronic Engineering, St. John's & St. Mary's Institute of Technology (phone: 886-02-2801-3131; fax: 886-02-2801-3413; e-mail: robin@mail.sjsmit.edu.tw).

enough to record the vocal, and it is also a benefit to reduce the total wave data size. Based on this idea, a new MIDI-based audio system is proposed in this paper. In the recording process, we only record the vocal wave data, and then combine this vocal wave data with the MIDI data generated by the MIDI instrument as a MIDI-wave file. In the playing process, the MIDI data and the vocal wave data are synchronously processed, respectively. Finally, their analog signals after DAC conversion will be mixed as an audio song. Due to a significant reduction on the data size of a song, a compact disc will store more songs than that of the conventional music CD.

II. BASIC CONCEPT OF MIDI

The Musical Instrument Digital Interface (MIDI) protocol, a standard specification, has been widely accepted and utilized by musicians and composers. Because it contains only the MIDI data needed by a synthesizer to play the sounds, the MIDI data is very efficient for representing musical performance information, and this makes MIDI an attractive protocol not only for composers or performers, but also for computer applications which produce sound, such as multimedia presentations or computer games [3]. The main advantage to use MIDI data is to achieve less storage space than the digitally sampled audio data [5]. Furthermore, better transmission efficiency for multimedia data on Internet can also be obtained.

A. MIDI Data Format

When we play music on the MIDI instrument, this MIDI instrument device sends a sequence of MIDI data in serial bit stream at 31.25K bps from its MIDI OUT port. Each MIDI data includes one STATUS byte followed by 0 to more DATA bytes to construct a MIDI message, in which the MSB of the STATUS byte set to 1 specifies various MIDI commands such as note-on, note-off, or program change, and the MSB of the DATA bytes set to 0 represents the number and range corresponding to the operation specified by the STATUS byte such as note-on velocity.

B. MIDI File Structure

A standard MIDI file is composed by single Header chunk followed by one or more Track chunks, and each chunk consists of type, length, and data [2]. Fig. 1 shows the MIDI file structure, where the data part of the Header chunk specifies the format, number of tracks, and timing for the MIDI file, and data part of Track chunk contains many MIDI messages and their corresponding Delta-Time bytes. The Delta-Time placed in front of each message is a variable-length quantity and

represents the amount of time before the following message [4], and the unit of the Delta-Time is clock tick depending on the tempo specified by in the header chunk of the MIDI file.

C. MIDI Sequencer and Synthesizer

For a MIDI-based audio player, sequencer and synthesizer are two important components [3]. The sequencer is used as a MIDI file interpreter, and it can generally manage multiple MIDI data streams, or tracks. As MIDI data is read, the sequencer decides how long each note play according to the Delta-Time bytes for each channel voice message in the MIDI file, and it also knows which instrument should be used to play these notes according to the STATUS bytes for each channel voice message in the MIDI file. The synthesizer is used to practically produce the various instrument sounds, which means the sound quality of MIDI devices is mainly dependent on their synthesizers. Two widely used techniques are Frequency Modulation (FM) synthesis and Wavetable synthesis. The FM technique can generate various sound waves with different timbres by the approach that uses one periodic modulator signal to modulate the frequency of another carrier signal. Alternate technique, Wavetable synthesizer, and uses a memory table to store a large number of sampled sound segments (i.e. sound waveforms) that may be looked up and utilized when needed. According to amplitude envelope of the attack and the sustain sections of various instrument sounds during playback, the Wavetable synthesizer will loop the samples in the corresponding segments. This synthesis approach has been the majority of professional synthesizers for achieving high sound quality.
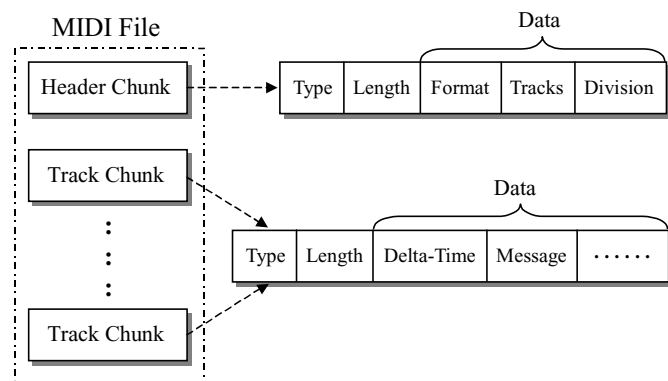

Fig. 1. MIDI File Structure

III. PROPOSED AUDIO RECORDING/PLAYING SYSTEM

To achieving a small digital data size for audio, we propose an audio recording/playing system using the MIDI file to replace the wave data file of music. Based on this ideal, recording and playing of audio all need two processing parts including the MIDI data for music and the wave data for vocal, which differ from the conventional music CD system. Besides, the MP3 compressor and decompressor are also considered to employ if expect to achieve a much smaller data size. In this section, the processes of recording and playing of the proposed

MIDI-based audio system will be described in detail, respectively.

A. Process of Recording

In the process of recording shown in Fig. 2, vocal input must be synchronized with music played, but they can separately generate their own data. Firstly, we use a MIDI instrument to play the desired music. At mean while; the corresponded MIDI data will be generated immediately. However, the generated MIDI data are quite much smaller compared with the conventional music wave data. When the music is played, the vocal input of singer also synchronously generates the wave data by PCM (Pulse Code Modulation) technique with 11 KHz sampling rate and 16-bit, 2-channel data resolution. Because the voice band is 4 KHz, only 11 KHz sampling rate for vocal is required, unlike 44 KHz for the conventional audio. Thus the vocal data only needs 1/4 times by the original wave data. Finally, we combine the MIDI data of music and the wave data of vocal to perform a MID-Wave file for using in the proposed audio record/play system. Fig. 3 shows the detailed format of a MIDI-wave file, where SYN data is used to indicate two starting time for music playing and vocal playing, respectively.

Fig. 2. Block diagram of recording process

B. Process of Playing

Fig. 4 shows the process of playing for the proposed audio system. When a MIDI-wave file is read, according to the format of the MIDI-wave file, the MIDI data and the vocal wave data are synchronously processed, respectively. By the SYN data in a MIDI-wave file, the processed starting time of the vocal wave data and the MIDI data can be decided. Following the Delta-Time bytes of MIDI data, the MIDI sequencer interprets the recorded MIDI data as a sequence of MIDI messages at appropriate time. The tone generator using FM modulation or wave table techniques will synthesize various wave data of tones corresponded to different MIDI messages, such as piano, saxophone, trumpet, etc. In the end of process, two D/A converters are used to separately convert two wave data into the practical analog waves, and these analog waves will be mixed to recover an original audio song. However, the standard MIDI data do not specify how the sound of each instrument is produced in details, which means different synthesizers may

generate different sound performance for the same instruments. In the future, only adding the accurate definition about sound performance in the MIDI data, any MIDI synthesizers can produce the same sound as that of the original music.
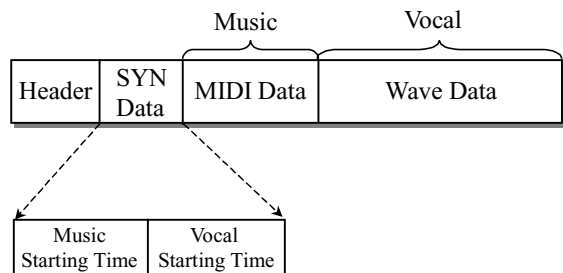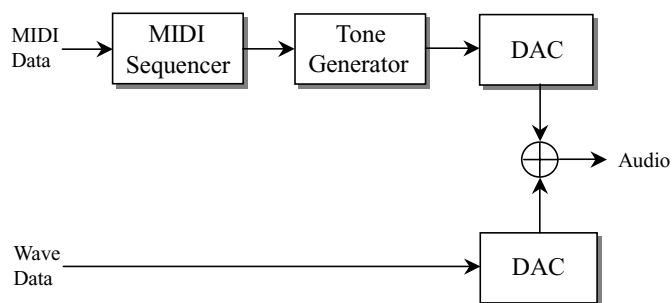

Fig. 3. Format of a MIDI-wave file

Fig. 4. Block diagram of playing process

C. Implementation of Player

According to the process of playing described in section 3.2, we use two processors to implement the MIDI-oriented audio player. In the architecture of player shown in Fig. 5, the data bits of all digital components are16 bits, and the functions of all parts are shown as the following descriptions.

(1) CD-ROM Controller: used to control the CD-ROM device in order to read the MIDI-wave files stored in a CD, and directly move the MIDI-wave file into the local memory of CPU for each playing.

(2) CPU: a general-purpose processor, used to drive the CD-ROM controller and separate the MIDI data from the wave data.

(3) DSP Processor: used as a MIDI sequencer to interpret the MIDI data and a tone generator to synthesize various wave data of different instruments by using the lookup of the wave table [6].

(4) Local Memory: used to store the program/data of system and the MIDI-wave file read from the CD-ROM Controller. After the CPU splits the MIDI-wave file, only the wave data resides in the local memory.

(5) Shared Memory: used to store the MIDI data separated from the MIDI-wave file for the DSP processor.

(6) DAC: Two D/A converters are used to separately convert two wave data into practical analog waves.

(7) Mixer: used to mix the music generated from the MIDI data with the vocal generated from the wave data.

In the design of the proposed player, the synchronization between music playing and vocal playing is an important issue. Although the starting time for vocal playing and music playing are specified by "SYN" data in the MIDI-wave file, however, the delay time incurred by the processing of the MIDI data including the interpretation of MIDI messages and wave data synthesis need to be considered. Here, the fast processing of a DSP processor can reduce the delay time. Furthermore, the effect of the tone generator is also a key factor to affect the performance of player, which means that the played sound quality depends on the resolution of wave synthesis. Because the shared memory must be accessed by the CPU and the DSP processor, thus the DSP processor that supports DMA function can directly access the share memory when it needs to process the MIDI data, and no performance degradation due to stealing the cycle that the CPU do not use the share memory.

Fig. 5. Architecture of the MIDI-based audio player

IV. EXPERIMENTAL RESULTS

Before implementing the practical circuit of the proposed player, we use the sound effect card based on PC and a MIDI instrument to emulate this system. In recording process (shown in Fig. 6), we input a music MIDI file into the MIDI instrument, and then the output of the MIDI instrument is connected to the line input of sound effect card of PC1. The vocal input will be connected to the microphone inputs of sound effect cards of PC1 and PC2, respectively, where the sound effect card of PC1 sets its sampling frequency 44KHz and data resolution 16 bits, 2-channel, another sound effect card of PC2 sets its sampling frequency 11KHz and data resolution 16 bits, 2-channel. When the MIDI instrument plays the music, the vocal concurrently inputs into two sound effect cards. On PC1, the song wave file including vocal and music will be produced, at mean while, another wave file only including vocal is also produced on PC2. After recording, we combine the vocal wave file from PC2 with the original MIDI file as a new file called "MIDI-wave" file. In playing process, we play the song wave file and the "MIDI-wave" file on PC, respectively. By using multimedia application software, "Spectrum Analyzer Pro Live", we compare the spectrum from the output of the proposed audio system with the spectrum of the original sound played by the wave data.

Fig. 6. Recording process of emulation

A. Data Size Comparison

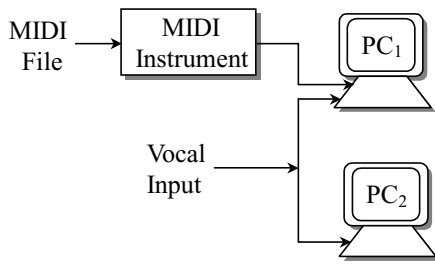Due to only storing vocal wave data sampled by 11 KHz, consequently, the vocal wave data of our proposed audio system can reduce about 3/4 data size compared with the wave data sampled by 44 KHz. Besides, the MIDI data representing the original music also can effectively reduce a large number of data size. Therefore, the proposed MIDI-based audio system has the main benefit to significantly reduce data size. In our experiment, we record 4 songs with different types such as soft or rock rhythms. Table II shows the total data size comparison for the MIDI-wave file of our proposed audio system and the wave data file of the conventional music CD.

Table I
Data size comparison for two data files

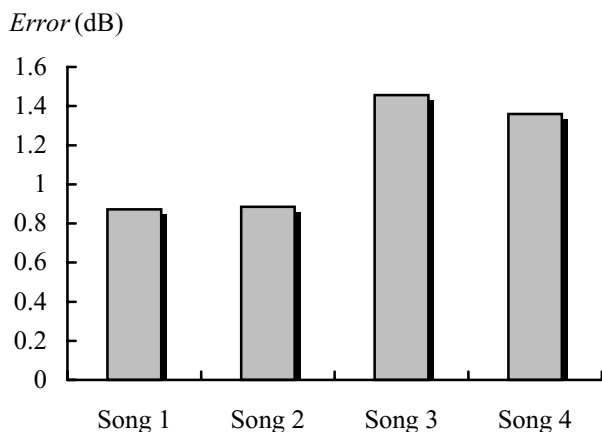| Songs	Wave Data File	MIDI-wave File	Data Saving Rate
Song1 (Soft rhyme)	36.6 MB	10.1 MB	72.2%
Song 2 (Soft rhyme)	23.6 MB	9.91 MB	58%
Song 3 (Soft rock rhyme)	21.1 MB	9.89 MB	53.1%
Song 4 (Rock rhyme)	23.7 MB	5.79 MB	75.6%

Fig. 8. Spectrum error for different songs

B. Spectrum Error Comparison

In our simulation, we observe two spectrums using wave data and using MIDI-wave data for different songs, respectively. We find these two spectrums are very similar for the same song. Moreover, we also make a spectrum error comparison related to the original sound wave for the output waves of different songs using MIDI-wave files. Evaluating the RMS (Root Mean Square) error of all energy at different frequency, we find that the largest spectrum error is less than 1.5 dB from Fig. 8.

V. CONCLUSIONS

By using the MIDI file to replace the music wave data file, and only recording the vocal wave data using 11 KHz sampling, a new audio system is proposed to reduce the data size of songs in this paper. In our experiment, we indeed find the data saving of the proposed MIDI-based audio system is generally up over 50%, even up to75%. To further reduce the data size required, the well-known MP3 compression/decompression technique also can be applied in the vocal wave data of this MIDI-based audio system. As demonstrated in this paper, the output spectrum of the player in the proposed audio system suitably approximates that of the original sound wave, where RMS error is less than 1.5 dB. In the future, the proposed MIDI-based audio system can be applied to multimedia data transmission on the Internet such as on-line KTV to achieve higher sound quality and better real-time effect.

REFERENCES

[1] T. T. Doan, "Understanding MIDI. "The key to creating and conducting the music to your own MTV video," IEEE Potentials, vol. 13, pp. 10-11, Feb. 1994.
[2] G. Hansper, "An introduction to MIDI," July 1998. http://crystal.apana.org.au/ ghansper/midi_introduction/contents.html
[3] J. Heckroth, "Tutorial on MIDI and music synthesis," The MIDI Manufacturers Association, 1995. http://www.harmony-central.com/MIDI/Doc/tutorial.html
[4] MMA and JMSC, "MIDI 1.0 detailed specification," version 4.11, IMA, Woodland Hills, CA, Feb. 1990.
[5] K. N. Kim, U. P. Chong, and J. H. Choi, "Conversion from CD-DA format to MIDI format maintaining a sound quality," Proc. 3rd Russian-Korean International Symposium on Science and Technology, vol. 1, pp. 300-303, June 1999.
[6] S. Yim, Y. Ding, and E. B. George, "A real-time MIDI music synthesis system using sinusoidal modeling on a TI TMS320C32 digital signal processor," IEEE 1st Workshop on Multimedia Signal Processing, pp. 469 – 474, June 1997.
[7] N. J. Sieger and A. H. Tewfik, "Audio coding for conversion to MIDI," IEEE 1st Workshop on Multimedia Signal Processing, pp. 101 – 106, June 1997.
[8] A. S. Moussa and T. P. Baker, "A dynamic microtunable MIDI system: problems and solutions," Proc. 5th Biannual World Automation Congress, pp. 339-344, June 2002.

Proceedings

The 2004 International Conference on Chinese Language Computing (ICCLC'2004)

Co-Editors

J. H. Lee, Pohang University of Science & Technology, China
T. S. Yao, North East University, China
K. J. Chen, Academic Sinica, Taiwan
A. C. Liu, Feng Chia University, Taiwan

An Intelligent System for Learning and Indexing Chinese Characters

Eileen V. Moy, Kai Chu, Cindy Katri and Lee Henderson
Institute of Emerging Technologies and Learning, Dept. of Computing Sciences
Humboldt State University, Arcata, CA 95521
Contact: Dr. Eileen V. Moy   Email: evm7001@humboldt.edu

Abstract

Given the increasing demand for Chinese language proficiency and the advancement in educational technology, it has become apparent that the traditional way of teaching and learning Chinese orthography is facing a serious challenge by the multimedia mode of delivery. In our newly designed software, the Chinese Character Learning System v 2.0, students learn Chinese orthography while exploring the intuitive and student-centered interfaces that ultimately tell the story of the Chinese language, art, history, and culture. The Chinese Character Learning System v. 2.0 integrates vector-based animation, digital video, interactive games, database management systems, geographic information systems, and cartography with artificial intelligence and cognitive psychology. The companion web site houses a relational database that facilitates interactive learning and collaboration. The result is an effective and flexible learning system that challenges traditional systems, and is especially useful for teaching Chinese as a Second Language.

1. Introduction

The Chinese Ministry of Education reported in February 2003 that 30 million people in 85 countries are learning Chinese [1]. In China, a document-intensive society where writing has surpassed oral communication in importance for thousands of years, non-Chinese must read and write Chinese in order to be truly effective. For non-Chinese, however, the usual barrier to mastering the Chinese language is Chinese orthography, the traditional system of character writing. Chinese orthography is typically learned through rote memorization, and to date, very few non-Chinese and overseas Chinese have been able to master the system. Upon discovering that they must learn six or more "styles" of a given character, as "ancient" forms co-exist with "modern traditional" and "modern simplified" characters in contemporary usage, students are further dismayed. Clearly, there is a need for improved materials in the teaching of Chinese orthography.

2. Theoretical Basis of the Project

Of the current Chinese character indexing systems in usage, both Kangxi's 214 radicals and the contemporary 227 modified radicals are based on character strokes and character parts that encompass some suggestions of patterns that may be useful for learning. By tradition, however, Chinese teachers instruct students to learn radicals and characters through rote memorization. In a departure from tradition, our new approach is to start with meaningful elements that have form, sound, and meaning and can be learned through logic and reasoning. Our system has advantages for the learner who has little or no preparation in the Chinese language, and whose primary language is English.

In our systems approach to Chinese orthography, first elucidated by Professor Kai Chu [2], we have defined three types of formal elements in the Chinese writing system, *primary significs, phonetics, and phonograms:*

- *Primary significs* refer to the most fundamental characters in the Chinese language, those that cannot undergo further decomposition into simpler meaningful units.

  Example: 一 "yi 1"
  one

- *Phonetics* refer to characters that represent the primary sound aspect of a character. They can be characters of any level in the hierarchical system.

  Example: 霍 "huo 4"
  sudden (adj.)

- *Phonograms* refer to characters that are composed of a signific and a phonetic. 90% of Chinese characters fall into this category.

Examples:

Phonogram　　*Phonetic*　　*Signific*
藿 "huo 4" ＝ 霍 "huo 4" ＋ 艹 "cao 3"
　　pea (n.)

Phonogram　　*Phonetic*　　*Signific*
攉 "huo 3" ＝ 霍 "huo 4" ＋ 扌 "shou 3"
　to knead (v.)
　to mix up (v.)

The 108 root characters are a set of reusable components, in some ways analogous to an "alphabet" for building Chinese characters. We consider these root characters to be first level (L1) characters. In order to form second level characters (L2), we use a set–theoretic model to combine two root characters. In a similar manner, we combine a root character (L1) with a second level character (L2) to obtain a third level character (L3). We repeat this process to obtain higher level characters, up to as many as seven levels. Using this system, we have systematically constructed a hierarchical tree model that can accommodate all Chinese characters.

In addition, we apply the mathematical concept of "series" to reveal the structure of different patterns of Chinese character relationships in two ways:

- *Direct Descendants*--a "vertical series" or direct descendants of derived characters.
- *Sibling*s--a "horizontal series" of siblings of derived characters.

For example, the series 一 *one*, 二 *two* , 三 *three*, 王 *king*, and 主 *prince* is a *Direct Descendant* group, while 二 *two* and 正 *correct* belong to a *Sibling* group.

3. A Chinese Character Indexing System

We strongly believe that effective language learning cannot be conducted in a vacuum. Therefore, we created a cultural hierarchy to help the learner, which was first developed in a copyrighted pilot version [3] of the *Chinese Character Learning System.* Our learning environment allows the learner to be steeped in a cultural setting during their study. In the *Chinese Character Learning System v. 2.0*, we categorize Chinese characters into eight distinct cultural groups: *Universe, Reverence, Five Elements, Humanity, Physical Body, Four Necessities, Animals,* and *Plants.*

We divide the groups into 25 subgroups, which are further divided into 108 roots. Through this cultural hierarchy, the student also learns to appreciate the Chinese cultural mind set. For example, if a learner wishes to search the cultural hierarchy for the root character *Horse,* they will find *Horse* indexed under the major group *Animals* and the subgroup *Land.*

4. The Software Interface

The *Chinese Character Learning System v 2.0* is a media-rich, student-centered environment for non-Chinese to learn and explore Chinese characters. Our group created vector-based animations in FLASH MX 2004 Professional and Illustrator CS. We produced original digital video footage, edited them in Adobe Premiere Pro and Final Cut Pro, and embedded them directly into our FLASH animations. We recorded sound files as narratives, background music, and special sound effects, and edited them in the Cubase sound editing program. Integrating these multimedia components in a shell based on the object-oriented FLASH scripting language, Actionscript 2.0, we developed learning software compatible for both the Windows and MacOS environments.

Applying the 108 root characters recursively for learning a new character, the *Chinese Character Learning System v. 2.0* promotes internalization of knowledge by the student. Conjoined with a multimedia mode of delivery, the student finds learning Chinese characters to be a pleasurable, intuitive and intellectually challenging experience.

Our hierarchical design guides students through a progression of learning activities. Short learning modules surround each character with a cultural and historical context as well as describe the character's historical evolution. Navigational buttons encourage students to explore different modes of learning.

As an example, consider a lesson module for the root character 馬/马 *Horse.* Students are able to choose from several options: They may read or listen to a story about the ancient horse in China. They may watch an animation showing the gradual transformation and abstraction of the pictogram into its modern forms as in Figure 2. They may learn about the invention of the stirrup and the horse-drawn carriage. They may practice their pronunciation of the Chinese word for *Horse* ("mǎ"). As students traverse deeper into the lesson, they may entertain themselves with 3-D games. They may create their own interactive maps and historical timelines to further their understanding of the rich background underlyingChinese orthography.

On the other hand, students may choose to explore the rich web-based relationship database and discover the hierarchical tree that displays the Direct Descendants and Siblings of the 馬 /马 *Horse* character. For some sample screen shots of the software interfaces, please refer to Figures 3 and 4.

5. Future Work

In order to obtain the feedback necessary for refinement of our work, Beta testing of the *Chinese Character Learning System v 2.0* will occur at various institutions of higher learning in both the United States and China, including Humboldt State University. In addition to the English edition of our software, we will be releasing a Spanish language version.

Citations:

[1] Source: Global Times News: http://www.chineseon.net (June 2004)
[2] Chu, Kai, Doctoral Thesis: A Hierarchical System for Chinese Orthography with Applications to Education and Computers; University of California, Berkeley, California, 1983.
[3] Chinese Character Learning System: ISBN 0-9625491-0-X (8 Volume Set): Learning Systems International, 1990. Copyright Documentation: "A Hierarchical System for Chinese Orthography," U.S. Copyright Office Registration #TX-1-863-987. Effective Date: 6/11/86.
[4] Macromedia Flash ActionScript and Object Oriented Programming. Documentation: http://www.macromedia.com/devnet/mx/flash/actionscript.html (June 2004)

Figure 1: Horse

| Picture | Small Seal Character (c. 200 B.C. E.) | Official Hand (c.200 A.D.) | Classical Script (c.1000 A.D.) | Simplified Character (c.1950 A.D.) |

Figure 2: CCLS v 2.0 Main Menu

Figure 3: CCLS v 2.0 Game: The Lexicon

A Chinese Character Database and Query System

Kai Chu, Eileen V. Moy, Cindy Katri, Harold Ricker and Lee Henderson
Humboldt State University, California, United States of America
Contact: Dr. Kai Chu   kc1@humboldt.edu

Abstract

We present a Chinese Character Database and Query System for searching and indexing Chinese characters as an adjunct to the Chinese Character Learning System. The relational database system consists of 17 tables that contain describable attributes of Chinese characters and the relationships among them. This database is hosted on a Sun Sparc Server that is running Apache and MySQL, and is accessible on the Internet through the Humboldt State University network. The Web location is http://cnrs-sun1.humboldt.edu.

1. Introduction

Our group has designed a relational database for searching and indexing Chinese characters, scripted in the REBOL language [1].]. In addition, created on-line user views and query interfaces in the English language, using PHP [2], a scripting language for web development, and the MySQL query language [3]. The database is a companion to the *Chinese Character Learning System,* a stand alone system described in a separate publication in the current issue of this journal, entitled *An Intelligent System For Learning and Indexing Chinese Characters* [4].

2. System Analysis

In this section, we describe users' query needs and the resulting organization of information in a database of Chinese characters, designed for students of Chinese orthography whose primary language is English.

2.1 User Needs

English language users of the *Chinese Character Learning System* typically perform an initial search for a Chinese character in one or a combination of several ways: by its English name or translation, by the number of strokes contained in a character, or by its Pinyin pronunciation. To this end, we have provided a Search Menu that includes all of these basic search features

Several related matches may be found. Once the desired character is selected from the available choices, the user may wish to learn more about the many attributes of the character. More advanced attributes include its form in Chinese Classical Script, the original meaning, extended meaning[s], its Unicode identifier and even its frequency of Internet usage by Google searches. Complex attributes include syntactic categories and semantic information about the character. See Figure 1 for an example of a search result.

2.2 Information Organization

To support user needs, we organized the information on Chinese characters in our database as a hierarchical structure, analogous to a genealogy tree. Based on Professor Kai Chu's hierarchical system for Chinese orthography as displayed in his copyrighted pilot software [5], we classified Chinese characters into different levels. We started with 108 root characters in the first level (L1). Each root character is a parent node, and gives rise to an entire series of child nodes, or descendants. Some characters may extend to a depth of up to seven levels from the root character.

The result is a family tree of related characters, and a system of parent-child and sibling-sibling relationships. Because of its intuitive structure, this model offers a practical system for easy retrieval of information pertaining to a given character.

3. System Design

In this section, we address two design aspects: user views and database design. The database design is further divided into conceptual, logical and physical database design.

3.1 Design of User Views

We have designed several user views, each corresponding to a distinct search method. We have collected and will continue to collect and analyze different users' applications. Our goal is to apply a "centralized approach" that will combine all user views and culminate in one integrated database management system.

3.2 Phases of the Database Design

Our design process follows the three well-documented phases of design: the conceptual, logical, and physical phases.

3.3 Conceptual database design

We used a set-theoretic model for organizing all Chinese characters. In this model, we first combine two root characters (L1) to obtain a more complex character at the second level (L2). Subsequently, we obtain a third level character (L3) by combining a root character with a character from the second level (L2), and so on to get other higher level characters. Using this method, we constructed a deep hierarchical tree for all Chinese characters. At this point our data model is entirely independent of any implementation details.

3.4 Logical database design

We first organized the essential attributes of Chinese characters into 17 tables, or relations. Then we identified the relationships among tables, depending on whether the relation is 1:1, 1:many, or many:many. We also established the primary keys, foreign keys, alternate keys, and the proper normalization processes to avoid database anomalies.

3.5 Physical database design

In our Chinese Character Learning System students learn Chinese orthography while exploring the intuitive and student-centered interfaces that ultimately tell the story of Chinese language, art, history, and culture. To support this we integrate digital video, interactive games, geographic information systems, and cartography into the system. This means organizing the database on the secondary storage to include many types of files including the physical locations of the image, video, audio, pictogram, character, and pronunciation files that are part of the multimedia learning modules in the *Chinese Character Learning System*.

The table organization is supported by several indices to achieve efficient access to the data. Database integrity and security are also built into the relational model.

4. System Implementation

We chose the REBOL language for implementing our database design because it is a powerful language that was designed for distributed computing ideal for the next generation of Internet applications. We selected MySQL because of the advantages of open source software, and because we needed a fully relational tool for querying our database. We chose PHP for interfacing between our database and MySQL. The system resides on a Sun Sparc server that is hosting an Apache web server at Humboldt State University, a campus of the California State University system. Users may access the database and query system through the following web address:

http://cnrs-sun1.humboldt.edu

5. Future Work

We are currently in the process of identifying and optimizing subsets of the database and query system that would address specific user needs. One of the targeted subsets is the 500 most frequently used characters in the People's Rebublic of China standardized examination of proficiency in the Chinese Language, the *Hanyu Shuiping Kaoshi* [6].
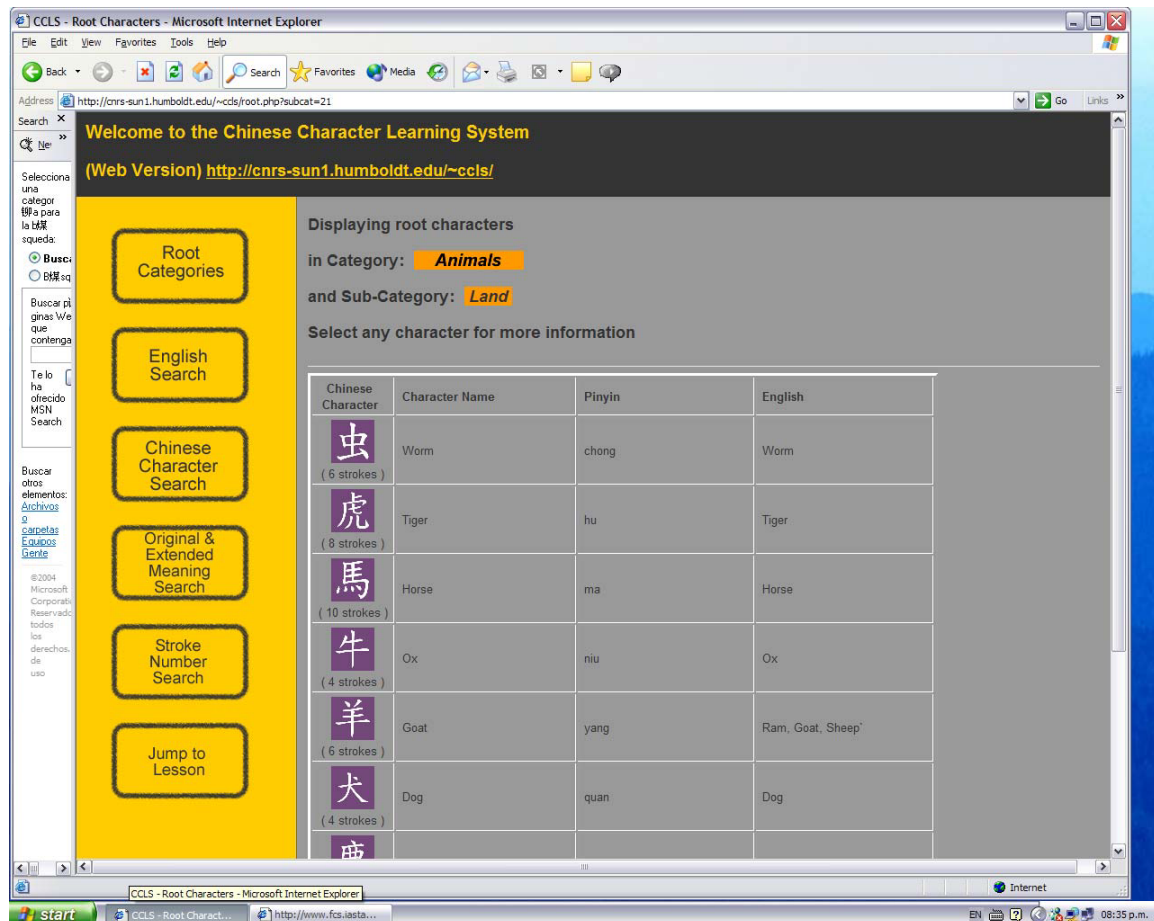
We anticipate the expansion of work in this area to include more advanced queries that would include logical operators and artificial

intelligence (AI) rules to facilitate scholarly
research, such as a system for investigating
Chinese history and geography.

Citations:

[1]REBOL Language Documentation: http://www.rebol.com/docs.html (June 2004)

[2] PHP Manual: http://www.php.net/manual/en/ (June 2004)

[3] MySQL Reference Manual: http://dev.mysql.com/doc/mysql/en/index.html (june 2004)

[4] Moy, E.V., Chu, K, Katri, C., Ricker, H. and Henderson, L.   An Intelligent System For Learning and Indexing Chinese Characters. *International Journal of Computer Processing of Oriental Languages*. (September 2004)

[5] Chinese Character Learning System: ISBN 0-9625491-0-X (8 Volume Set): Learning Systems International, 1990. Copyright Documentation: "A Hierarchical System for Chinese Orthography," U.S. Copyright Office Registration #TX-1-863-987. Effective Date: June 11,1986.

[6] Zhang Jun Guo, ed. Hanyu Shuiping Da Gang (汉语水平考试大纲). Beijing: Xiandai Chuban She. ISBN: 7-80028-279-1/G.074. (1995)

Figure 1: Query Result for a Chinese Character

A Systemic View on Mandarin-based Computer-Assisted Language Teaching and Learning

Yoon Fah Chuah
Mara University of Technology
Malaysia
cyf@tm.net.my

Chee Weng Khong
Multimedia University
Malaysia
cwkhong@mmu.edu.my

Abstract:
With the tremendous advancement achieved in the arena of Mandarin-based computational technology, performing Mandarin-based word processing and even transmitting Chinese characters over the Internet is now no longer a daunting task. This paper provides a systemic viewpoint on the advances made in present day Mandarin-based computational technology. In addition, it highlights the pedagogical relevance of Computer-assisted language learning (CALL), as well as multimedia-based CALL and Web-based Instruction (WBI) in the field of Mandarin teaching and learning. This paper was inspired by need for logistics, infrastructure and pedagogical scaffoldings for WBI in the field of Mandarin teaching and learning at Mara University of Technology, Malaysia.

Keywords: Mandarin education, Computer-assisted language learning, Multimedia, Web-based Instruction

Introduction

While using computer technology to perform word processing in alphabetic languages like English is relatively easy, doing the same with pictographic languages such as Chinese poses a unique frustration to professionals who work with the Mandarin. The exclusion of the Mandarin in the original design of computer technology was perceived to be the major holdup to computerised processing of the language. Similarly, owing to the intrinsic technical difficulties of Mandarin-based computational technology and Mandarin teachers' lack of relevant "know-how" in computer technology, the realm of Mandarin teaching and learning has always ended up resorting to more traditional, less technology-oriented methods of dissemination (Yeo, 1997). However, with the tremendous advancement achieved in this realm performing Chinese word processing and even transmitting Chinese characters over the Internet is no longer a difficult task (James, 1996).

The Keying In of Chinese Characters

In recent years, input of Romanised Mandarin characters carrying tone marks has captured attention from some major developers of Chinese software. The latest versions of NJStar and ChineseStar 2.97, amongst others, both come with this feature (L. Zh. Zhang, personal communication, Jan 24, 1999). For a more detailed description on keyboard input method, please refer to *Learn and Utilise Chinese Star 2.97 Step-by-Step* by Yong (1998). According to Zhang (personal communication, Jan 24, 1999), the final version of Windows Internet Explorer (IE) version 5.0 released by Microsoft in March 1999 comes with a very distinguished feature with enhanced Chinese support. After the installation of two Chinese texts (BIG 5 & GB) from the support system, IE5.0 will display BIG 5, GB and HZ codes quickly without the help of other Chinese systems. These Chinese add-ons embedded with IE5.0 have greatly facilitated online Chinese applications within Windows environment. James (1996) sums up the advancement achieved in the realm of Mandarin-based computational technology well. He asserts:

"Now, after a short span of a dozen years, it has become clear that rather than the characters creating a serious impediment to computerisation, the computer makes any difficulties of the Chinese writing system recede into the background, and anything that we can do with western languages – from desktop publishing to database manipulations and on-line communications – we can also do, quite easily, with Chinese (p.32)."

As revealed by Ferreira-Cabrera and Atkinson-Abutridy (1998), in the last 15 years, members of the foreign language teaching profession have begun to examine seriously notional and functional communicative method as a more effective approach to train language learners develop functional language ability. In the field of Teaching Chinese as a foreign language, Chu (1996) believes that " …to teach Chinese is to help students acquire useful knowledge and apply this knowledge to communicate in a linguistically accurate and socially and culturally appropriate manner. In other words, to teach Chinese is to help students learn to use Chinese as a functional language (p.139)." Founded on this belief, Chu further states that "…to teach Chinese is thus to make a commitment to assist students in acquiring communicative abilities (p. 154)."

Communicative CALL

The advent of new media and its new uses resulted in the movement towards adoption of many alternative approaches to learning and in a similar logic, the rapid growth of computer technology has propelled the field of foreign language instruction to seek for a more innovative instructional approach and it subsequently gives rise to CALL (Zhang, 1998). According to Ferreira-Cabrera and Atkinson-Abutridy (1998), more recently, CALL has been regarded as an endeavour that "covers the fundamentals of and concrete propositions for integrating computers into a methodology for the communicative teaching of languages (p.75)." Warchauer and Healey (1998) reveal that the effectiveness of CALL has been well documented and it is learned that the foremost advantage of it is that it facilitates the process of developing learners' language skills such as listening, speaking, reading and writing. According to Zhang (1998, p.55), the pedagogical values offered by CALL are:

- Random and rapid access allows the instant retrieval of vocabulary and grammar explanations. It also contributes to easy learner control and recycling of material.

- CALL's ability to store and manipulate data also makes it possible to keep scores, log errors and track learner performance.
- The consistency and patience of CALL is crucial for learning by association and repeated exposure. Paradoxically, without the possible ill effects of an over-bearing human teacher, the patient computer can provide a very user-friendly and learner-centred learning environment.

Yao (1996) reports that the earliest attempts at seriously using computer technology to teach Chinese characters took place in the 1970s. He further informs that the Chinese component of the Plato System was developed at the University of Illinois at Urbana – Champaign. The project was considered a success as the supervisor of the project managed to lay a theoretical foundation for the teaching of Chinese characters and to establish some practical parameters for designing computer software for the teaching of Chinese characters. The success of the project has helped demystify the assumption postulating that, as the Mandarin is non-alphabet and non-syllable, it did not lend itself to computerisation. He further reveals that this success and another software programme, developed by Smith at Brigham Young University in 1981 for teaching Chinese characters, proved that in actual fact computer technology can accommodate the delivery of Mandarin education and therefore CALL for the Mandarin learners started emerging.

The necessary linguistic components that should be mastered by novice learners of Mandarin to develop basic communicative (both verbal and written) competence in the language outlined by the U.S.-based National Foreign Language Centre (NFLC)'s *Guide for Basic Mandarin Programmes* (1997) are the pronunciation of the four tones, grammar and sentence structure, the writing of the transcription systems (Romanised Chinese phonetic systems), the writing system of Chinese characters, and the cultural aspects of the language. Zhang (1998) reveals that, in recent years, with the escalating development in Mandarin-based computational technology, there are hundreds of commercially available multimedia-based CALL courseware programmes tailored to the teaching and learning of virtually all aspects of the above-mentioned linguistic skills.

The subsequent section illustrates the degree to which multimedia-based CALL has benefited Mandarin learners. A multimedia-based CALL package developed

by Wang (1997) for teaching Chinese characters has recorded positive outcome. The students were put on the trial for eight weeks and a survey gauging their perceptions towards the multimedia mode of learning was administered in the last week of the trial. Data in the survey showed an overwhelmingly positive response of the multimedia mode of delivery where learning Chinese characters was concerned. What follows are some of the comments from the respondents on the strengths of the multimedia-based CALL package:

- "It is great to watch the stroke order [through the video capture] and listen to pronunciation at the same time"
- "The video recordings of the stroke order and the tracing function are very useful"
- "The character history, the bit on radicals, audio and video, everything really except writing with a mouse is a lot different to writing with a pen"
- "Easy to use, stimulating and interesting"
- [Characters are] "learnt at own pace and given in context"
- [For] "the clarification it offers and accessibility as well"

To sum up, the students welcomed the multimedia-based teaching and learning of Chinese characters. Wang (1997) concludes that through multimedia, students appeared to have the option for more individualised interaction with the materials than a teacher could possibly offer in a traditional classroom.

Web-based Mandarin Instruction

According to Godwin-Jones (1998), the availability of web-based interactive web tools such as dynamic HTML, Java Applets, CGI, Plug-ins and JavaScript has transformed the WWW from a sole repository of information into an exciting medium for realising an authentic interactive learning environment. In the area of web-based Mandarin instruction, Zhang (1999) reveal that there are basically two types of online delivery programmes. The programmes are either those with structured materials and are offered totally online as a distance-learning package, or those that serve as a supplement to existing lessons conducted in the traditional classroom setting. The International Conference on Internet Chinese Education (ICICE) held in Taiwan in 2001 has showcased a host of research projects related to WBI in the Mandarin classrooms. The projects showcased in the conference provide samples of good practice and suggest that the use of WBI, if carefully employed,

would lead to improved performance and motivation on the part of the Mandarin learners (ICICE, 2001).

Let's consider WBI targeted at novice learners of Mandarin, so as to help portray that WBI is a viable instructional tool. The discussion starts with a project titled *Internet-based Chinese Teaching and Learning* (http://chinese.bendigo.latrobe.edu.au, 1998) developed by an Australian Mandarin instructor. This fee-paying, web-based project is specifically launched to provide non-native speakers, who are geographically and temporally dispersed in various localities, with fundamental skills in listening, speaking, reading and writing in Mandarin. The programme provider, Zhang (1998) has tested and evaluated some of the methods and techniques made available with interactive web-based technologies for the course he developed. The methods and techniques examined include: Java-based animation, CGI-enabled multiple choice, streaming audio and video, Internet telephone, and online discussion forum. The online Chinese course has been running on an experimental basis and the positive responses given by the participants so far have been overwhelming. Zhang thus concludes, "…the Internet has added a new dimension to CALL. From a pedagogical point of view, Internet technologies will enable students to increase their learning motivation through non-linear interactivity. Certainly Chinese Studies can make use of this advantage if teaching and learning strategies are properly adopted (p. 109)." Chang and Low (1999), two American Mandarin instructors have developed a similar web site, where their site provides self-accessed interactive exercises to facilitate self-directed learning for the learners. The web-based lessons make use of text, images, audio, video and animation to present material and to provide immediate feedback to the learners. According to them, the lessons would help learners develop effective reading strategies and listening skills. To this end, in relation to the basic assumptions of what first time learners of Mandarin need to learn as articulated by NFLC in the previous discussion, Liu (2001) has seemingly summed up precisely how WBI could accommodate the delivery of various language skills deemed necessary to be acquired by the Mandarin learners.

The above discussion has demonstrated that the interactive, collaborative and communicative features of WBI enable Mandarin learners attend web-based classes in a pedagogically relevant manner and enjoy the functionality of the language classroom.

Summary

Once relegated to "novelty" status, Mandarin-based CALL has seemingly achieved the recognition it deserves owing largely to the tremendous advancement in educational computing. Though further empirical evidences are needed to validate the legitimacy and effectiveness of Mandarin-based CALL and WBI particularly, the language instructors should take the plunge and approach the innovation as a teaching and learning instrument. Albeit the innovation can not supplant the language classroom and the face-to-face interaction between instructor and learner, it should serve as an efficient and powerful dissemination aid. This review was inspired by the imminent establishment of an intranet-based (local area network) and web-based, Chinese language facility at the Mara University of Technology, Malaysia. The need for logistics, infrastructure and pedagogical scaffoldings resulted in the research and publications of various literatures in the area of Chinese language teaching and learning.

*Part(s) of this paper is extracted from the main author's postgraduate thesis titled "Teaching and Learning Chinese Language through Web-based Instruction at a Malaysian University – An Exploratory Study".

References

Chu, M. (1996). Class plan for teaching Chinese as a functional language. In S. McGinnis (Ed.), Chinese pedagogy: An emerging field (pp. 135-158).Ohio: The Ohio State University Foreign Language Publication.

Chang, K.Y., & Low, M. (1999). Self-directed interactive Chinese web exercises. Proceedings of the First International Conference on Internet Chinese Education (pp. 102). Taiwan: OCAC.

Ferreira-Cabrera, A.A., & Atkinson-Abutridy, J.A. (1998). Venus: Enhancing motivation to learn Spanish as a foreign language through multimedia. In J.Gassins, M. Smith., & D. Cunningham (Eds.) Proceedings of the 1998 worldCALL conference[PDF format] (pp.75-77). Melbourne: The Horwood Language Centre, The University of Melbourne.

Godwin-Jones,B. (1998).Emerging technologies: Dynamic web page creation. Language Learning & Technology, 1(2), 7-13.

James E.D. (1996). Advances in computerisation of Chinese. JCLTA, 31(3),15-32.

Kubler, C.C. et al. (Eds.) (1997). NFLC guide for basic Chinese language programs - Pathways to advanced skills, Vol.III. Ohio: The Ohio State University National Foreign Language Resource Centre & the National Foreign Language Center at the Johns Hopkins University.

Liu, J. (2001). Distributed Chinese language learning: A model for foreign language learning [Online]. Available: http://amfellow.iu.edu/LiuProposal.pdf [2003, July 21].

The Second International Conference on Internet Chinese Education (ICICE 2001) Home Page (2001). [Online]. Available: http://edu.ocac.gov.tw/netedu02/default.htm [2001, May 13].

Wang, Y.P. (1997). Learning Chinese characters through multimedia [Online]. Available: http://www.lerc.ritsumei.ac.jp/callej/4-1/wang2.html [1999, Sept 9].

Warchauer, M., & Healey, D. (1998). Computers and language learning: An overview. Language Teaching, 31, 57- 71.

Yao, T.C. (1996). A review of some computer-assisted language learning (CALL) software for Chinese. In S. McGinnis (Ed.), Chinese pedagogy: An emerging field (pp. 255-284).Ohio: The Ohio State University Foreign Language Publication.

Yeo, G.K. (1997). CALL in Chinese [Online]. Available:http://www.iscs.nus.sg/~yeogk/doc/call.html [1997, Dec 15].

Yong, S.H. (1998). Learn and Utilise Chinese Star 2.97 Step by Step. Hope Computer Systems Sdn. Bhd. : Subang Jaya.

Zhang, L.Zh. (1998). Interactive web design for Chinese language acquisition: Methods and techniques. In J.Gassins, M. Smith.,& D. Cunningham(Eds.), Proceedings of the 1998 worldCALL conference[PDF format] (pp.108-109).Melbourne: The Horwood Language Centre, The University of Melbourne.

Zhang, Zh.Sh. (1998). CALL for Chinese: Issues and practice. JCLTA, 33(1), 51-82.

Zhang, L.Zh.. (1999). Creativity and Online Chinese Education. Proceedings of the First International Conference on Internet Chinese Education (pp.134 -139). Taiwan: OCAC.

Contrastive Analysis and Feature Selection
for Korean Predicate Generation
in Chinese-Korean Machine Translation System

Jin-Ji Li, Dong-Il Kim*, Jong-Hyeok Lee
Department Computer Science and Engineering,
Electrical and Computer Engineering Division and
Advanced Information Technology Research Center (AITrc)
Pohang University of Science and Technology (POSTECH)
San 31 Hyoja Dong, Pohang, 790-784, R. of Korea
E-mail: {ljj, jhlee@postech.ac.kr}

*Language Engineering Institute,
Division of Computer, Electron and Telecommunication Engineering,
Yanbian University of Science and Techonology (YUST)
Yanji, Jilin, 133-000, P.R. of China
E-mail: {dongil@ybust.edu.cn}

Abstract

To generate a proper Korean predicate, a natural expression of modality is the most important factor on machine translation (MT) system. Tense, aspect, mood, negation and voice, which relate to modality information, should be provided correctly by Chinese analysis in Chinese-Korean MT system. In this paper, a new applicable categorization of Korean modality system will be proposed through contrastive analysis of Chinese and Korean and from the viewpoint of a practical MT system. And according to the new categorization of modality system, we will describe the feature selection in Chinese. A variety of machine learning methods are adopted to show the effectiveness of re-categorization of the modality system in Korean and feature selection in Chinese.

Keywords: Modality Information; Chinese-Korean MT System; Tense; Aspect; Mood; Negation; Voice;

1 Introduction

Tense, aspect, mood together with negation and voice are hard to be translated naturally in Chinese-Korean MT system. Especially, Chinese and Korean belong to the totally different language family in terms of linguistic typology and genealogy, which can cause differences of modality system between the two languages.

In most of the previous papers, the taxonomy of the modality system is discussed respectively in each language from the linguistic viewpoint [3, 10 and so on]. But the modality system is so different from language to language, which will cause problems such that some concepts cannot be found in the other language. In this paper, we will discuss the modality system through a contrastive analysis of Chinese and Korean and from the viewpoint of a practical Chinese-Korean MT system. We also consider the distribution of modality information through a real corpus-based analysis.

As an isolating language, Chinese does indeed lack functional markers in a sentence. The modality information in Chinese is provided by a variety of combination of related morphemes, and these morphemes always scatter in the whole sentences. So, feature selection in Chinese, which decide modality system, is also an important task. And by revealing the correlations between modality information in Chinese, we will present the reasons why tense, aspect, mood, negation and voice should be considered synthetically.

Usually, the modality information is processed by rule-based system. But, making the heuristic rules includes a labor-intensive process and a rule-based system has a chronic problem of low coverage. To resolve all these problems, in this paper we proposed a machine learning (ML)-based method to process modality information effectively.

The rest of the paper is organized as follows. Chapter 2 will briefly introduce some issues of modality information in Chinese and Korean. And chapter 3 will describe a new categorization of modality system in Korean, which is proper to Korean generation in Chinese-Korean MT system. According to the new catego-

rization of modality system, feature selection in Chinese will be given in chapter 4. In chapter 5, a variety of machine learning methods are adopted to show the effectiveness of re-categorization of the modality system in Korean and feature selection in Chinese. Finally, conclusions and future work will be given in chapter 6.

2 Some Issues of Modality Information in Chinese and Korean

2.1 General Comparison between Chinese and Korean

The systems of modality information are substantially different between Chinese and Korean in terms of typology and genealogy. We will show the differences by comparison of two languages according to several linguistic views.

Table 1 shows the difference of modal expression in terms of language typology. Generally a Korean predicate is composed of a main predicate and an auxiliary predicate. Main predicate describes the core content of predicate, and auxiliary predicate refers to modality information. As an agglutinative language, Korean modality information is intensively expressed by the auxiliary predicate at the end of the sentences. But as an isolating language, in Chinese the modality information is scattered in all sentences, which will definitely increase the processing complexity to detect correct modality information.

Table 1. Difference of modal expression

Language	Typology	Modal Expression
Chinese	Isolate	By combination of related morphemes
Korean	Agglutinative	Mainly by auxiliary predicate

As shown in Table 2, Chinese does not have a fixed grammatical category in tense and voice. Besides this, each system of modality information is so different between two languages, which means some concepts probably cannot be found in the other language. We will explain this phenomenon in detail in chapter 3 through a contrastive analysis.

Table 2. Differences of modality information from the viewpoint of grammatical category

Language	Tense	Aspect	Mood	Negation	Voice
Chinese	Not Exist	Exist	Exist	Exist	Not Exist
Korean	Exist	Exist	Exist	Exist	Exist

2.2 Correlations of Modality Information in Chinese

There exist some correlations among modality information. Thus, in a Chinese-Korean MT system we should consider synthetically the modality information: tense, aspect, mood, negation and voice. And basically, tense-aspect-mood relate to time category, so correlations between them are a natural result.

1. *Tense and aspect*
    The event of perfective aspect mostly occurs in past time. Tense information can be inferred from aspect markers. Especially in Chinese, it is a salient characteristic, because there is no grammatical form of tense markers.
2. *Tense and mood*
    Future tense is closely tied with the mood of presumption and volition.
3. *Negation and aspect*
    There exist affirmative-negative correspondences holding for sentences containing aspect markers [3]. In other words, some negative particles can imply aspect information.
4. *Negation and mood*
    There are special negative imperative particles like '*bie*', '*buyao*' and so on. And "verb + '*bu*' + RVC" structure is a negative potential form.

3 Re-categorization of Modality System in Korean

The taxonomy of modality system is so different between Chinese and Korean. And the majority of them are discussed from linguistic perspectives. It is not reasonable to directly apply the result to MT system, because most of the discussions did not consider the MT environment and also did not consider the differences of the two languages.

We will target a practical Chinese-Korean MT system to re-categorize the modality system of Korean. And will give the contrastive analysis between Chinese and Korean.

3.1 Tense

In Chinese and Korean, it is generally agreed that there are three tense forms: past, present and future.

But Chinese does not have the grammatical category of tense, because the concept denoted by tense is indicated by content words like temporal adverbs. Definitely, it will increase the difficulty to provide correct tense information. And the future tense in Korean, in comparison with past and present tense does not have a formally fixed form. And interestingly, both in Chinese and Korean, there exists a correlation between tense and

mood, which is usually based on the future tense and the mood of presumptions and volitions. For example, the future tense suffix '-*kess*' can be analyzed not only as a temporal marker, but also as a modal marker.

According to the facts presented above, it is not essential to have a future tense system in Korean, especially in Chinese-Korean MT system. In this paper, we just define two tense forms: *past and non-past*.

3.2 Aspect

Chinese grammatically marks aspect but not grammatically mark tense. As such, Chinese is exclusively an aspect language. While Chinese is recognized as an aspect language, there is no generally unified description of the aspect system of this language, since different researches define aspect in their own ways. In this paper, we will adopt the aspect system classified by [8], which is the most acceptable classification in Chinese. Table 3 shows the aspect system in Chinese.

Table 3. Aspect system in Chinese

Category	Subcategory/Aspect Marker
Perfective	Actual / '*le*' Experiential / '*guo*' Delimitative / reduplicant Completive / RVC[1]
Imperfective	Durative / '*zhe*' Progressive / '*zai*' Inceptive / '*qilai*' Successive / '*xiaqu*'

Chinese is an aspect language with a complete set of markers to express aspectual distinctions. Here, some aspect markers like '*le*', '*guo*', '*zhe*' and '*zai*' are functional words, which have a well-established system to express aspectual meaning. But other ways of conveying aspectual values, they retain their original lexical meaning. It will increase the complexity to detect whether the aspect marker just works as aspect marker or not in a MT system. And also, it is not enough to generate a correct Korean predicate just give the aspect information like completive. In this paper, we will suppose that aspect markers such as reduplicant, RVC, '*qila*i' and '*xiaqu*' just provide pure aspect information.

The aspect system of Korean is relatively simple than the Chinese one. Generally, Korean aspect is divided into two categories, such as perfective and imperfective. And the imperfective category is sub-classified such as progressive, iterative, habitual and stative duration. But except for progressive, the other three subcategories

don't have specially fixed grammatical forms, which are semantically encoded to convey aspectual meaning. In other words, these three subcategories are classified semantically rather than syntactically. An unmarked form can be used in these three sub-categories.

There is a correlation between past tense and perfective aspect in Korean, because both of them use the same suffix '-*eoss*' to represent the related information. In other words, from the generation perspective, it is enough to use just one of the information.

To take all these facts into consideration, we will define the aspect system in Korean like this: *perfective (experiential) and imperfective (progressive)*.

3.3 Mood

Mood is a formally grammaticalized category of the verb with modal function. We will consider *mood,* a grammatical category, is expressed by verbal inflection, and *modality* is the semantic concepts of moods that can be expressed by some specific lexicon. And the honorific in Korean, its relevance for the analysis of mood lies in the fact that they are marked with a set of differing verbal suffixes, thus creating a paradigmatic set of forms within each and every mood [1]. But in Chinese, there is not such phenomenon, so it is hard to provide the honorific information in Chinese analysis to generate the proper speech style of Korean. Based on this difference, we will neglect the honorific in Korean generation.

As a well-known isolating language, Chinese does not have a verbal inflectional form, so it lacks a system of marking sentences for mood. Actually, the mood system of Chinese is mostly indicated by modal auxiliaries and particles. Here we will regard the system, which expressed by modal auxiliary and particles as *modality* in Chinese. And in Chinese, almost all of the modal auxiliaries and particles can decide the category of *modality* independently. For this reason, we will just consider *mood* in this paper.

The mood system in Korean is more elaborate than the one in Chinese. But considering that the genre of the corpus is news of People's Daily in China, we will set these four categories such as: *simple declarative, imperative, interrogative and exclamatory declarative.*

3.4 Negation

The negation system in Chinese and Korean are very similar. Commonly, there are standard negation, double negation and imperative/propositive negation that are necessary in Chinese-Korean MT system. But the mood information of imperative/propositive negation will be given in mood part, so we will just define two catego-

---

[1] RVC: Resultative Verb Construction

ries in Korean negation system, such as: *standard negation and double negation.*

3.5 Voice

As mentioned before, Chinese voice belongs to lexical category because of one of the characteristics of Chinese, but not so in Korean.

In Chinese, it has often noted that the message carried by passive sentences with verbs of perception or cognition is unfortunate or pejorative, whereas the meanings of their verbs are neutral. Also, topic-comment construction in Chinese also can indicate passive message, which does not use any passive marker. Although the non-adversity usage of passive sentences increase in modern Chinese due to the influence of Indo-European languages, the topic prominence of Chinese together with the restriction of the '*bei*' construction to adverse message combine to reduce the usage of the passive in the language [3].

And there is no clear explanation about causative voice. Causative sentences are mostly discussed as a category of serial verb construction. Usually a variety of notional causative forms are adopted in Chinese to express causative voice.

Generally, there are more productive sentences without any passive/causative markers than with markers in Chinese, which can make the detection more difficult.

In Korean, voice is a well-defined grammatical category, and we will set the voice system in Korean as: *passive and causative.*

3.6 Taxonomy of Modality System in Korean

Finally, we defined the taxonomy of modality system in Korean as follows, which is proper to a practical Chinese-Korean MT system.

1. Tense: {*past, non-past*}
2. Aspect: {*perfective (experiential), imperfective (progressive)*}
3. Mood: {*simple declarative, exclamatory declarative, imperative, interrogative*}
4. Negation: {*standard negation, double negation*}
5. Voice: {*passive, causative*}

4 Feature Selection in Chinese

According to the modality system, which is defined in section 3.6, we will select features that can represent the modality information saliently in Chinese.

[6] used 1-gram to 10-gram strings at the ends of the input Japanese sentences and all of the morphemes from

each of the sentences to process modality information in Japanese-English translation. But as mentioned before, Chinese is an isolating and not a verb-final language, so n-gram strings at the ends of the input Chinese sentence and all the morphemes may not have great influence in Chinese. We will show the experiment result to support this perspective. Through this comparative experiment, we derived the conclusion that the feature selection should sufficiently consider the characteristics of the source language.

In this paper, we will select the features according to linguistic knowledge, which closely relate to the modality information in Chinese. The overview of features is as follows.

Table 4. Overview of features[2]

Feature	Description
1.Time_t	Temporal marker, POS[3] is *t*
2.Time_d	Temporal marker, POS is *d*
3.Time_auxv	Temporal marker, POS is *auxv*
4.MP[4]_motion	MP is a motion verb of non-duration
5.MP_pos	POS information of MP
6.Aspect_u	Aspect marker, POS is *u*
7.Aspect_d	Aspect marker, POS is *d*
8.RVC_a	MP has a RVC, POS is *a*
9.RVC_vd	MP has a RVC, POS is *vd*
10.RVC_v	MP has a RVC, POS is *v*
11.MP_redup	MP has a reduplicant construction
12.Mood_y	Sentence final particle, POS is *y*
13.Mood_w	Sentence final punctuation mark, POS is *w*
14.Negation	Negative particle
15.RVC_bu	'*bu'* before RVC
16.MP_buliao	'*buliao'*after MP
17.Voice	Words function as passive/causative markers
18.MP_voice	Property of MP, which can be used in a sentence without passive/causative markers

As previously noted, Chinese is an aspect language and have a complete set of aspect marker. But in contrast, Chinese has no markers of tense. The language does not use verb affixes to signal the relation between the time of the occurrence of the situation and the time that situation is brought up in speech [3]. Tense information is provided by temporal words like adverb and

---

[2] *t*: time; *d*: adverb; *u*: particle; *auxv*: auxiliary verb; *y*: mood particle; *w*: sentence final punctuation mark; *a*: adjective; *vd*: directional verb; *v*: verb;
[3] POS: Part Of Speech
[4] MP: Main Predicate

time or can be inferred from aspect markers. Some auxiliary verbs and adverbs in Chinese, which are used to show presumptive/volitional *modality*, can refer future tense according to the correlation between tense and mood. Feature 1, 2 and 3 are selected for these reasons.

Through corpus analysis, we have found an interesting result that most of Chinese sentences are organized in unmarked form, even without any temporal or aspect markers. These unmarked sentences are hard to provide correct tense information when they are translated into Korean. About 59.7% ((2663+4570)/(5165+6949)) of sentences are in unmarked form. The distribution of tense is as follows.

Table 5. Distribution of tense with/without markers

Category	Frequency	With / Without Marker
Past	5165	2502 / **2663**
Non-Past	6949	2379 / **4570**

Especially, as an isolating language, we can say that the unmarked form in Chinese is one of the salient characteristics to make sentences very simple. But even though there are no temporal or aspect markers and no context information, people can recognize that the event is occurred in past time or non-past time by linguistic competence. Through corpus analysis, we have found that the property of main predicate can be a useful clue to identify tense. Actually, the motion verbs of non-duration can only represent past or future tense in a moment. And motion verbs of duration, stative verbs and adjectives are usually translated into present tense in Chinese-Korean MT. For this reason, we will also consider feature 4 and 5 as temporal markers.

The features from 6 to 11 have been described in section 3.2 already which represent the aspect information well in Chinese.

Traditional Chinese grammar refers to the sentence-final particles as *yuqici* (mood word). The sentence final particles and punctuation marks such as question mark and exclamatory mark can be good features to show mood information.

Negation can be expressed by general negative particles, so feature 14 is selected for this reason. Feature 15 and 16 reflects some special cases in Chinese to express negation. One is a negative word '*bu*' before RVC and the other is '*buliao*' after main predicate to show negative view.

The last two features are used to classify voice information in Chinese. As mentioned before, voice system in Chinese can be expressed by two ways. One is with passive/causative markers and the other one is without these markers. Feature 17 represents the passive/causative markers. And we already knew that most

productive sentences do not have the markers, which can show voice information. The last feature will be a good feature that can provide the information whether the main predicate can be used in a sentence without any passive/causative markers.

5 Experiment

In this paper, in order to make a modality detection model, we randomly extracted sentences from the PFR corpus[5] and segmented them into simple sentences to make the training corpus in Chinese. In our Chinese-Korean MT system, first, it will segment long sentences into several simple sentences to reduce the parsing complexity [7]. Finally, we collected 12,114 simple sentences and annotated the main predicates with the tense-aspect-mood-negation-voice information by three bilinguists, considering the modality information when they are translated into Korean. 46 categories have been found in the corpus and the distribution of the major categories is as follows.

Table 6. Distribution of major categories

Category	Frequency	Rate (%)
Non-past	5731	47.30
Past	4592	37.90
Non-past, standard negation	432	3.57
Non-past, imperfective	249	2.06
Non-past, causative	238	1.96
Past, causative	230	1.90
Past, passive	133	1.10
Past, standard negation	117	0.97
Non-past, interrogative	117	0.97

In this paper, we adopted 9 classifiers and their variants provided by WEKA 3.4[6]. The classifiers are as follows.

1. Naïve-Bayes classifier (NB)
2. Decision Tree using C4.5
3. Instance-based learning: K-NN, K*
4. Decision Table (DT)
5. Multi class classifier using SVM and AltDT
6. Boosting: AdaBoost (AB), LogitBoost (LB)

Concerning multi class classifiers, we used SVM of polynominal kernal function and AltDT as base learners. AltDT is a learning algorithm of alternating decision tree. It is a 2-class classifier like SVM. And two boost-

---

[5] PFR corpus: POS tagged corpus of People's Daily (1998) made by Peking University of China.
[6] WEKA is a ML s/w in Java, which is developed by the University of Waikato.
(http://www.cs.waikato.ac.nz/~ml/weka/index.html)

ing method were adopted with the base classifiers such as, decision stump (DS) and decision tree (C4.5). Decision stump is a binary 1-level decision tree. It is usually used in conjunction with a boosting algorithm.

The accuracy of 9 classifiers is displayed in table 7 as follows.

Table 7. Accuracy of various ML methods

Method	Accuracy (%)
NB	75.17
C4.5	71.90
K-NN	82.33
K*	77.94
DT	**82.71**
Multi+SVM	75.97
Multi+AltDT	79.49
LB+DS	81.85
AB+C4.5	71.95

For a more accurate evaluation, 10-fold cross-validation was adopted in all experiments and it was repeated ten times and the results were averaged. We can see that K-NN, DT and LB+DS have the best performance. But, through the two-sided *t*-test, we can say that the differences between these three performances are not 'significantly different'. But, we will select decision table for our classifier because for future classification, the resulting decision table provides a constant classification time on the average and is therefore well suited for applications in real-time environments [2]. Decision table was adopted to perform the following experiments.

In order to identify the best features and the optimal feature set, we also performed two more experiments. First is to handle experiment with one of the features omitted each time. We displayed the five central features in table 8. The results show us that the MP_motion feature is the most important factor. This is because about 59.7% of Chinese sentences in the corpus are in unmarked form as we mentioned before. In other words, the MP_motion can function as a unique clue in the sentence to provide modality information, especially correct tense information. The distribution of the major categories can really support this judgment in table 6.

Table 8. List of most central features

Feature	Reduction of accuracy (%)
MP_motion	11.72
Aspect_u	4.69
Time_auxv	3.23
Negation	3.09
Voice	2.67

Secondly, we adopted scheme-specific feature selection of Wrapper[7] approach [9], which is essential for learning decision tables to find the optimal feature set. Here, the optimal features may not include all relevant features. Bestfirst, forward and genetic search were performed. The optimal feature sets of each search method are given in table 9. Again, we applied the optimal feature sets respectively to evaluate and achieved the accuracy such as, 83.10%, 83.10% and 83.05%. There is a little improvement of accuracy, but we can still expect an accuracy improvement through optimal feature set selection.

Table 9. Optimal feature set of each search method

Search Method	Optimal Feature Set
Bestfirst	3,4,6,7,9,13,14,15,16
Forward	3,4,6,7,9,13,14,15,16
Genetic	1,2,3,4,6,7,8,9,12,13,14,15,16

As mentioned before, we also performed a comparative experiment with the feature set of [6]. The result is shown in the table 10. Both of the experiments are performed by SVM method in the same parameter environment.

Table 10. Comparative experiment result with Murata's method

Feature Set	Accuracy (%)
Murata's feature set	69.53
Our feature set	**75.97**

The result with our feature set improved the performance by 6%. In summery, we can come to the conclusion that feature selection is a very important factor and it should take the linguistic characteristics into consideration.

6 Conclusions and Future work

To generate a proper Korean predicate, the natural modal expression is the most important factor on MT system. Usually, the modality information is processed by a rule-based system. But, to make the heuristic rules is a labor-intensive process and the rule-based system has a chronic problem of low coverage. To resolve all these problems, in this paper we proposed a ML-based method to process modality information effectively.

First, we redefined a new applicable categorization of the modality system in Korean through contrastive analysis of Chinese and Korean and from the viewpoint of a practical MT system. And according to the new categorization of modality system and linguistic knowl-

---

[7] Attribute selection is implemented as wrapper around learning scheme.

edge, we described the feature selection in Chinese. The comparative experiment to [6] has shown that feature selection should sufficiently reflect the linguistic characteristics.

Through applying several ML techniques, in our experiment environment, we have found that the decision table showed the best result considering the computation time together with the performance. And we also found the property of main predicate whether it is a motion verb of non-duration is the most important clue. This result is supported by the distribution of the unmarked sentences and the distribution of the major categories in the corpus.

The corpus we used in this paper is not a bilingual corpus and the modality information is annotated by bilinguists. But to evaluate the experiment objectively, it is better to use a bilingual corpus and automatically extract the modality information through corpus analysis. And to satisfy a practical MT system, we will explore the effect of various domains in future. There also remained the translation problem of RVCs, which can work as an aspect marker, and also retain the original lexical meaning. It is essential to process the RVC problem because the RVCs occur frequently in Chinese sentences. Further research should be done with these problems.

Acknowledgements

References

[1] Adrian Thomas Wymann and Sumiswald BE, *The Expression of Modality in Korean*, PhD thesis, University of Bern, 1996.

[2] Kohavi, R., *The Power of Decision Tables*, in Proceedings of the European Conference on Machine Learning (ECML), 1995.

[3] Li, Charles N. and Sandra A. Thompson, *Mandarin Chinese: A functional reference grammar*, University of California Press, USA, 1996.

[4] Liu Y.H., Pan W.Y. and Gu H., *Shiyong Xiandai Hanyu Yufa (The Practical Grammar of Modern Chinese)*, Shangwu Press, 2002.

[5] Lü S.X., *Yufa Xuexi (The Study of Chinese Grammar),* Zhungguo Qingnian Press, 1953.

[6] Masaki Murata, Kiyotaka Uchimoto, Qing Ma and Hitoshi Isahara, *Using a Support-Vector Machine for Japanese-to-English Translation of Tense, Aspect and Modality*, the Data-Driven Machine Translation of ACL workshop, 2001.

[7] Meixun Jin, Mi-young Kim, Dongil Kim and Jong-Hyeok Lee, *Segmentation of Chinese Long Sentences Using Commas*, ACL 2004 Workshop: 3rd SIGHAN Workshop on Chinese Language Processing, 2004. (Accepted)

[8] Richard Zhonghua Xiao, *A Corpus-Based Study of Aspect in Mandarin Chinese*, PhD thesis, University of Lancaster, 2002.

[9] Ron Kohavi and George H. John, *Wrappers for Feature Subset Selection*, Artificial Intelligence, special issue on relevance, 1996.

[10] Suh Cheongsoo, *Korean Grammar*, The Deep-Rooted Tree Publishing House, 1994.

[11] Zhang, Z. G. *Xiandai Hanyu (Modern Chinese),* Renmin Jiaoyu Press, 1985.

Story Representation based on Term Distribution on Timelines for Chinese News Event Link Detection*

Hainan Jin*, Dongun An**
Dept. of Computer Engineering, Chonbuk National University, Korean
* hnkim@duan.chonbuk.ac.kr
** duan@chonbuk.ac.kr

Abstract

We investigate a system for Chinese news event detection that automatically discovers relevant among the news events and reports event topic to users in human-readable form. The TDT link detection aims to determine whether two stories discuss the same topic. This task much depends on similarity calculation that more depends on story representation. However, the study of feature word extraction strategy impacts on Chinese event link detection is still not deep enough especially. This paper provides some new issues of feature term extraction on timeline for Chinese event link detection. Feature selection is commonly known as dimensionality reduction approaches. It attempts to remove non-informative words from documents in order to improve categorization effectiveness and reduce computational complexity. We propose a simple statistical model of feature occurrence over time, and extract significance of terms appearance on an individual date. The groupings of terms display the major events and topics covered by the corpus on a given date. We use TDT2 Mandarin data as training data, and TDT3 Mandarin data as evaluation data. The experiment results show our feature term extraction strategy is useful to represent news stories; an empirical Chinese news event description improve the efficiency of event detection; improved story expansion is helpful.

1. Introduction

Traditional methods of information organization and exploration classify information in a corpus or sub-corpus according to the human psychology referring to the training documents. This traditional technique is useful for content-focused queries, but deficient for generic queries such as "What happened?" or "What's new?"

We are interested in using time as a categorization benchmark [1, 2]. This development not only satisfies the

necessary of retrieval based on immediate-content-focused queries for obtaining a variety of relevant stories, but also reduces complexity because on-line news stories have obvious time mark and doesn't need training data, so construct them automatically is very simple.

This paper discusses the influence of story representation based on timelines on Chinese event link detection. In this paper, we provide an empirical news event description based on conventional feature term extraction, improve existing story expansion strategy and construct story weight vector.

2. Corpora and Evaluation

The corpora used in this paper are the TDT2 Mandarin corpus. It spans from January 1,1998 to June 30,1998. There are 20 topics for Mandarin, and 18,712 story pairs for link detection tasks. We evaluated the performance with the augmented version of TDT3 corpus spanning October 1,1998 to December 31,1998. These corpora are provided by Voice of America, Xinhua and Zaobao News Agencies. No translation is required in this paper.

Because cost function's dynamic range makes it difficult to interpret, in this paper we use a Normalized Detection Cost $(C_{Det})_{Norm}$ that its lower value corresponds the better link detection [3].

$$(C_{Det})_{Norm} = C_{Det} / \min(C_{Miss} \times P_{t \arg et}, C_{FA} \times P_{non-t \arg et}) \qquad (1)$$

3. Link Detection System

This paper's primary aim is automatically selecting features that reflect and summarize individual event from a corpus for display. Classify the story with the same time tag, extract feature terms, explicitly rank how likely these features are to be high content relation and group these feature terms. The groups of feature terms signify individual event topic successfully.

3.1. Feature Term Extraction

Automatic feature selection methods include the removal of non-informative terms according to corpus statistics, and the construction of new features which combine lower level features into higher level orthogonal dimensions.

3.1.1. General Feature Term Selection In this paper, Chinese lexical analysis using Hierarchical Hidden Markov Model (HHMM) provided by free software ICTCLAS [4]. After processing, obtain segmented feature words with corresponding 39 lexical Part-Of-Speech tags. Remove non-informative words from the inverted word list, such as preposition, auxiliary word, pronoun and exclamation etc.; preliminarily choose noun, verb, adjective, time and number, etc. At the same time, we take some simple rules to overcome the trouble that brought by too careful segmentation. For example, mark people name "周恩来/nr" by combining the surname "周/nr" and the first name "恩来/nr".

In general, nouns and verbs are important terms to identify the topic that the story discusses. Especially, some special nouns such as people names, location names and organization names, etc. that denote interesting entities and some important verbs denote the specific events. But a lot of researches indicate that compared with other morphological feature terms the verbs and adjectives bring more information [5]. We found that there are some important adjectives provide significant information. Some important nouns are mis-tagged as adjectives. Some unimportant verbs which do not have the actual meaning take much noise. All these lead this result.

In order to signify the news event better, we added compound nouns as the feature term based on the above tactics. Table 1 shows all performance improved when using compound nouns. The best strategy is to choose nouns, verbs, adjectives and compound nouns as the feature term to represent stories. In the experiments, we have discovered the increment of similarity threshold that corresponding best strategy. It indicates matching compound nouns in two different news stories is more difficult than matching single terms.

	Similarity Threshold				
	0.1	0.15	0.2	0.25	0.3
N&CNs	0.4621	0.4179	0.4301	0.4423	0.4664
N&V&CNs	0.4349	0.4093	0.4103	0.4207	0.4432
N&J&CNs	0.3910	0.3879	0.3964	0.4279	0.4097
N&V&J&CNs	0.4192	0.3940	0.3873	0.4005	0.4380

N:nouns V:verbs J:adjectives CNs:compound nouns
Table 1: Performance of Link Detection under Different Feature Selection strategies

3.1.2. Feature Word Distribution on Timelines We provide a model of feature term distribution on timelines usage is appropriate for automatically selecting and grouping. Group the stories with same time mark and choose x^2 statistic method that provided by the traditional text category [6, 7] to calculate the weight of an individual feature term f_0 on a certain date t_0 relatives to feature term distribution during a period of time Δt.

Table 2 is a 2×2 contingency table, t_0 is the certain day that we want to get intraday news events from different resources and different reporter's reports. Accordingly, the collection of all news stories during a period of time Δt is the all news events reported up to today.

	The stories that include the feature term f_0	The stories that not include the feature term f_0
The stories that are reported on t_0	a	b
The stories that are not reported on t_0	c	d

Table 2: Individual Feature Term Distribution based on Special Data

We use this distribution table to calculate the weight of individual feature term.

$$x^2 = \frac{(a+b+c+d)\cdot(ad-bc)^2}{(a+b)(a+c)(b+c)(b+d)} \qquad (2)$$

From a series of inverse lists, knowing a indicates the number of stories on that day t_0 contains f_0, a and b total amount is the number of stories from a given day t_0, a and c total amount is the document frequency of the feature term (f_0) in the whole stories collection, a、b、c and d total amount indicates the total number of news stories reported during a period of time Δt. Formula 3 express the relation between feature word and time figuratively.

$$w\text{Time}(t_0, f_0) = x^2(t_0, f_0) \qquad (3)$$

We select the term that its x^2 value is above our threshold. The groups of these terms can indicate different event. Through feature word distribution situation, we can clearly see the degree that event attract attention on its spanned dates.

3.1.3. Empirical News Event Description We found some general laws that dependence on coexistence of some simple terms can signify a kind of relevant event. For example, if readers see that there are such things of "U.S.A. and Iraq" in news title, will think of the war between two countries naturally, and some relevant news stories. Another example, one group about U.S.A. Colombia space shuttle crash events, the news title is various for emphatic content of the news story. Some

titles such as "The Chinese people express the compliments to the astronaut died in an accident of Colombia." "U.S.A. looks for the fall of Colombia space shuttle promptly." "Verify that seven astronauts die in this accident" etc., all these correspond to the same event and can be denoted with some significant words such as "American plane crashed".

Our research increase rule-based empirical news event description for story representation. According to artificial statistics rules, assign higher weight value to signify individual event. The experiment results are shown in Table 3.

	Similarity Threshold				
	0.1	0.15	0.2	0.25	0.3
Empirical news description	0.4401	0.4149	0.3670	0.3769	0.3898

Table 3: Performance of Chinese Link Detection with Empirical News Event Description

Experiment results demonstrate that empirical news event description is useful for event link detection.

3.2. Stories Similarity Measure

Each story is represented as a vector with the weights that are calculated with Formula 4.

$$wgt_{f_0} = tf_{f_0} \bullet wNE_{f_0} \bullet wTime\,(t_0, f_0) \qquad (4)$$

For general feature terms, we set wNE is equal to 1. For Named Entities such as <people's names> <place names> <organization names> <time> and <number>, etc., we set it a higher value. The cosine function is used to measure the similarity of two stories. Finally, a predefined threshold $TH_{decision}$ is employed to decide whether two stories are on the same topic or not.

3.3. Story Expansion

The length of stories may be diverse. There may be very few features remaining for short stories. Already existing technology is not reliable for low-frequency terms. The similarity of two stories may be too small to tell if they belong to the same event. To solve this problem, a lot of researches introduced story expansion technique [5].

The method introduced in paper [5], when the similarity of two stories is higher than a predefined threshold $TH_{expansion}$, which is always larger than or equal to $TH_{decision}$, the two stories are related some topic in more confidence. Thus their relationship will be used for story expansion later. For example, if the similarity of a story pair (A, B) is very high, we will expand the vector A with B when a new pair (A, C) is considered. If story is

represented as a vector with $tf * idf$ weights with formula 5, performance increases after story expansion.

$$w(t_i \in \{A, B\}, d_j) = tf_{i(A+B)} * idf_{i(all)} \qquad (5)$$

Let $idf_{i(all)}$ be the inverted document frequency of t_i for all documents, don't care whether the similarity of two stories is higher than TH_e. But story expansion with the non-relevant terms would reduce the performance of a link detection system. It may introduce some noise into the story and make the detection more difficult.

Our method improved this technique, only expanded some terms that both occur in the expansion story pair and counted idf relatives to the stories that the similarity of two related stories is higher than $TH_{expansion}$

$$w(t_i \in \{A\}, d_j) = tf_{i(A+B)} * idf_{i(S(\text{related part}) > TH_e)} \qquad (6)$$

We took the best strategy to represent story. The results are shown in Table 4 shows that improved story expansion outperforms the basic method. The total miss rate was decreased to nearly 80 percent of original amount. So story expansion is a good strategy to improve the performance of link detection task.

$TH_{decision}$	0.2				
$TH_{expansion}$	0.2	0.25	0.3	0.35	0.4
N&J&CNs	0.4210	0.4174	0.3767	0.4073	0.3783
N&V&J&CNs	0.4406	0.4237	0.3673	0.3910	0.4079

Table 4: Performance of Chinese Link Detection with Best Strategy

4. Experiment and Evaluation
4.1 Goals

We place test kernel on investigating the influence of feature term distribution on timelines and particular weight expression on Chinese news event link detection.

4.2 Evaluation Criterion

Heavy workload, our experiment only uses thirteen training events to test the performance of Chinese link detection task. These events are a portion of TDT2 event collection. Table 5 lists the news events with simple topic title.

Upcoming Philippine Elections
1998 Winter Olympics
Current Conflict with Iraq
China Airlines Crash
Tornado in Florida
Asteroid Coming
Viagra Approval
India, A Nuclear Power
Israeli-Palestinian Talks (London)
Anti-Suharto Violence
Anti-Chinese Violence in Indonesia
Afghan Earthquake
Clinton-Jiang Debate

Table 5: Detection Events

The standard evaluation is precision, recall, miss alarm, false alarm and micro-averaged F_1.

$$F_1 = 2 * precision * recall / (precision + recall) \quad (7)$$

4.3 Experiment Result

The performance of event link detection depends on feature term distribution, and term distribution more depends on weight calculation of the feature term. We use traditional $tf * idf$ to compare with our proposed weight calculation that provided with Formula 4. Figure 1 shows comparative distribution of story vector with different strategy.

<div align="center">(a)   (b)</div>

<div align="center">Figure 1: Story Weight Vector</div>

(a) $tf \cdot idf$ (b) bases on feature term distribution on timelines

The results of event link detection are shown in Table 6. Better performance is obtained corresponding similarity threshold from 0.1 to 0.35. Compare with traditional weight calculation, micro-averaged F_1 value has increased 13.6%.

	$tf \cdot idf$	$proposed$
precision	0.3313	0.3774
recall	0.9131	0.8506
miss alarm	0.0869	0.1494
false alarm	0.5765	0.2947
micro-averaged F_1	0.4862	0.5228

<div align="center">Table 6: Event Link Detection Result</div>

5. Conclusions and Future Work

We present a view for generating clusters of feature term and labeling the cluster automatically that capture the information corresponding to major news event. The research of story representation based on term distribution on timelines for Chinese news event link detection considers some issues in this paper. Compound noun is help for story representation. The improvement story expansion and empirical event description are useful.

Because the complexity of Chinese language, it's difficult to obtain correct segmentation and POS-tagging, the difficulties of feature term extraction influence system performance directly. Example, how to identify the people names and organization names may be spelled or be abbreviated in different way in different news agencies? All problems ask us to find better feature term extraction

strategy and optimal similarity threshold to satisfy the requirement of news event link detection task.

We applied the best strategy for TDT3 corpora in above experiments. The result of our method showed the total miss rate was decreased. But its value has not still reduced to a satisfying degree. This is another challenge issue to overcome.

For simplicity we have restricted ourselves to a retrospective task here, have not yet tested in an on-line setting.

Whether some already succeed techniques such as topic segmentation and thesaurus can improve the performance, need our experiment authentication in the future.

Future work we can apply these techniques in multilingual link detection task. For same event, different national, various sources and different news reporter, views are different. Consulting the content of these reports synthetically will help readers to understand the event more clearly. It will promote culture exchanges across different national boundaries.

6. References

[1] Russell Swan, James Allan. *Automatic Generation of Overview Timelines*. SIGIR 2000, pp.49-56

[2] Kyung-Soon Lee. *Multilingual Event Link Detection based on Term Distribution on Timelines and Multilingual Spaces*. Communications of the Korea information science society, pp.28-34

[3] Fiscus J.G., Doddington G.R.(2002) *Topic Detection and Tracking Evaluation Overview. In "Topic Detection and Tracking: Event-based information Organization"*, Kluwer Academic Publishers, pp.17-32

[4] ICTCLAS: "*Inst. of Computing Tech., Chinese Lexical Analysis System*". The Chinese Academy of Science, Beijing. www.nlp.org.cn

[5] Yin-JU Chen, Hsin-His Chen. *NLP and IR Approaches to Monolingual and Multilingual Link Detection*. COLING 2002: The 17th International Conference on Computational Linguistics, C02-1006.

[6] Yiming Yang and Jan O.Pederaen. *A Comparative Study on Feature Selection in Text Categorization*. ICML 1997:412-420

[7] Kjersti Aas and Line Eikvil. *Text Categorisation: A Survey*. Norwegian Computing Center, Oslo, June 1999.

Handwritten Chinese Character Font Generation Using Stroke Correspondence

Jungpil Shin and Kazunori Suzuki

Graduate School of Computer Science and Engineering, University of Aizu

Aizu-Wakamatsu City, Fukushima, 965-8580, Japan

`jpshin@u-aizu.ac.jp`

Abstract

This paper presents a novel system for font generation based on an individual's handwriting. The concept is to compress the reference character database by using vector quantization, with one input character corresponding to many reference characters for the purposes of font generation. We call that correspondence stroke correspondence. A compression rate of 1.4% is achieved by the vector quantization. The system is also applicable to characters that include smoothly connected strokes, by using a Fourier descriptor. In addition, the cost of the font generation is low for users and the font is generated on demand using several parameters. A strongly individual font, a standard font, and a font that includes many connected strokes are generated on the user's command. Two evaluation experiments involving 25 subjects show that generated fonts reflect the user's individual handwriting, using both subjective and objective criteria.

1 Introduction

Handwriting is a skill that is personal to individuals. Handwriting was developed as a means to expand human memory and to facilitate communication. The printing press and type writer opened up the world to fomatted documents, increasing the number of readers that, in turn, learned to write and to communicate [7]. Nowdays, typical fonts such as *Times Roman* and *Times New Roman* used widely in newspapers, articles and on the Internet. While these fonts are easy to read, they lack individual characters and appear overly mechanical. Most Japanese people may use an Oriental brushed-penned fonr for a New Year's card, for example, yet wish to express their feelings using their own individual characters. Receiving a letter or email written in a personal font makes communication more pleasant and expressive than if written in a typical font.

Our goal is to generate a handwritten-style font that reflects user's indibiduality. Our target for font generation is Chinese characters. There are over 3000 Chinese characters used in Japan, and each Chinese char-

acter has up to 25 strokes, giving more than 32,000 strokes in Chinese characters, and making it very difficult to enter all strokes for reflecting user's indivisuality. An efficient automatic method for character pattern generation is therefore required.

Many methods have been developed for the generation of handwritten-style fonts. For example, there are methods used in the pre-processing stage of handwriting character recognition [4][3], the main purpose of which is to normalize handwriting to ensure highly accurate handwriting character recognition. There are also methods for generating handwritten-style fonts that geometrically modify an existing font [5], however these use parameters that are very complex, and the resulting fonts do not reflect individual style. Methods also exist to extract parameters from a user's on-line handwritten characters, and modify a reference pattern using parameters based on them [6]. Obtaining an optimized reference pattern is difficult, as generated fonts depend greatly on a reference pattern, and applying this to a character containing connected strokes is considered difficult for the same reason. Furthermore, methods exist that extract geometry information from a user's off-line handwriting characters using a scanner, and use this to modify an existing font [8][9]. Only abstract aspects of a user's handwriting are reflected by these methods, and they ignore the prospect of reflecting a user's individuality.

The purposes of the method proposed in this paper are to:

1. generate a handwritten-style font that reflects the user's individuality,

2. reduce the cost of generating this font to a minimum,

3. include characters in the generated font having connected strokes, and

4. arbitrarily adjust to reflect a user's individuality by generating connected strokes.

2 Stroke Correspondence

Chinese characters are composed of a combination of a radicals. A radical may be separated accurately from text using the generation rules of Chinese characters, and most Chinese characters are assembled using two or more common radicals. A radical may be separated into several strokes, with each stroke corresponding to two or more common strokes. By paying attention to the characteristic of Chinese characters and strokes, we consider each stroke as a vector and compress the stroke pattern using vector quantization.

2.1 Target Characters

This font generation system aims to implement the first level of the Japanese Industry Standard (JIS) Chinese character code set. The code set includes 2,965 Chinese characters. A Chinese character has up to 25 strokes, and there are 32,398 strokes in total for all characters. The character database generated by [10] is used as the reference character database in this system. The database was compiled by 90 subjects, and the data were obtained using a stylus pen on an input character area of 256x256 pixels. Subjects were directed to write input characters cursively in a normal manner similar to handwriting in one's own notebook. Input characters are then normalized in size, and averaged. An averaged character is used as the reference character data.

2.2 Vector Quantization

The starting point of each stroke of the reference character database is moved to the origin, and the data are used as a specimen to divide the character data into several strokes, where $P_i(i = 1, ..., n)$ is the ith point of a stroke. The coordinates of the stroke are expressed as (x_i, y_i).

Despite a total of 32,398 strokes across all Chinese characters, most strokes are common since Chinese characters are assembled from a combination of radicals. Each stroke is therefore considered a vector, and many strokes may be compressed using vector quantization. Vector quantization is the method of classifying input vectors into several clusters. In this system, a simple clustering algorithm and the K-means clustering algorithm are used for the vector quantization. The simple clustering algorithm determines the number of compressed clusters and the initial values used by the K-means clustering algorithm. The K-means clustering algorithm classifies N vectors into an optional K categories [2], where K is the number of categories obtained from the simple clustering algorithm.

The simple clustering algorithm calculates the distance between one stroke and others, and classifies strokes based on this distance. If the distance is less than a threshold, these strokes are classified as within the same cluster. A Euclidean distance measure is used

Table 1: Total number of strokes using simple clustering.

Thre-shold	Total stroke number	Compre-ssion rate	Maximum strokes per character
0	32398	-	-
5	1156	3.57%	2721
6	817	2.52%	3303
7	601	1.85%	3824
8	455	1.40%	4308
9	364	1.12%	4825
10	294	0.91%	5344

for the distance calculation. In the following equation, S is the reference stroke, S' is the input stroke, and $D(S, S')$ is the distance between S and S'.

$$D(S, S') = \frac{1}{n} \sum_{j=1}^{n} \left(\sqrt{(x_{ij} - x'_{ij})^2 + (y_{ij} - y'_{ij})^2} \right).$$

(1)

Table 1 shows the result of classification of the reference character database using simple clustering, where, at maximum, 2,721 strokes may be compressed into one single stroke. Moving the starting point to the origin is considered the reason for the high compression. The number of clusters that are obtained by this algorithm, and the average value of a cluster, are used by the K-means clustering algorithm. In the K-means algorithm, K is the number of strokes in Table 1 and the first centroids are average values of clusters.

2.3 Stroke Correspondence Database

The main purpose of the stroke correspondence database is to analyze a correspondence stroke, which is one between a compressed stroke and an uncompressed stroke. The database is used to assemble a generated character from user's input. The stroke correspondence database is generated from the relationships between the strokes of the uncompressed reference characters and the compressed reference characters. Since a compressed stroke corresponds to many uncompressed strokes (for example, in the case of $\theta = 8$, one stroke corresponds to 4,308 strokes), fonts are generated effectively.

3 The Handwritten-Style Font Generation System

The method generates a handwritten-style font by applying the following procedure. First, the character's nonlinear parameters are extracted as global features from a user's on-line handwriting characters. Secondly, user's input strokes are corresponded to compress a

reference stroke pattern and assemble a Chinese character from these strokes. Thirdly, an abstract of the character shape is extracted using a Fourier descriptor, and connected strokes are generated. Furthermore, the generated character is modified by nonlinear parameters and a handwritten-style font is generated.

3.1 Global Features Parameters

The nonlinear transformation proposed by [6] is used as the global features parameters in this system. The nonlinear transformation is defined as follows:

$$
\begin{aligned}
x' = &((r_w - x)(r_h - y)a + (r_w - x)yc \\
&+xye + x(r_h - y)g)/(r_w r_h) \\
\\
y' = &((r_w - x)(r_h - y)b + (r_w - x)yd \\
&+xyf + x(r_h - y)h)/(r_w r_h)
\end{aligned}
\tag{2}
$$

where a, b, c, d, e, f, g, h are the nonlinear parameters and (x, y) is a coordinate of a character stroke and (x', y') is a coordinate after transformation, r_w and r_h are the character's width-height. These eight parameters are decided automatically by the least squares method for fitting the reference character and the input character.

3.2 Character Assembly

Figure 1 shows examples of correspondence strokes for the 6th stroke of the Chinese character "."." A character is assembled to correspond one input stroke with the strokes of many characters. Since a character is assembled stroke-by-stroke, there is an incomplete character in this case so parameter α is used to indicate whether to complete a character if it is assembled by using a certain number of strokes. The range of α is $0 < \alpha < 1$, and N_{MAX} and N_{inp} are the number of all strokes and the number of input strokes for a character, respectively. If α is $N_{MAX} < N_{inp}$, the character appears complete and the next process is conducted. A stroke in an insufficient section compensates for the same stroke in the compressed reference character database. A larger α indicates that fewer generated characters are used but the user's input is reflected more strongly. Parameter β is used to control the blend between a reference stroke and an input stroke, where β is $0 < \beta < 1$. A reference stroke and an input stroke are set to S_{ref} and S_{inp}, and S_{gen}, which is a generated stroke between S_{ref} and S_{inp}, is determined by $S_{gen} = \beta S_{inp} + (1 - \beta)S_{ref}$. The generated character remains identical to the user input stroke in the case of $\beta = 0$. In the case of $\beta = 1$, the generated character becomes is identical to the reference character.

Figure 1: An example of a corresponding stroke.

3.3 Connected Stroke Analysis Database

The connected stroke analysis database contains the criteria to connect the last point of a stroke with the first point of its following stroke. Not all generated characters necessarily include connected strokes. Connected strokes are analyzed using the input data from on-line handwritten characters that have been input by 90 people. The analysis determines a stroke that is appropriate for connecting to an another stroke, based on the connected stroke analysis database. The parameter C_θ is a threshold used to connect several strokes. If the number of writers who used connected strokes is greater than C_θ, the strokes are then connected to another stroke. Strokes $s1$ and $s2$ are candidate strokes of the connected strokes. The ending point of $s1$ connects with the starting point of $s2$, and the connected stroke $s1'$ is obtained and normalized so that the distance between each point is equal.

3.4 Connected Stroke Generation

This process explores a combination of strokes that can tend to connected strokes from the connected stroke analysis database. A parameter C_θ is calculated indicating the tendency for strokes to connect. If C_θ is exceeded, those strokes will be connected and they will combined as one stroke. After this, smoothly connected strokes are generated to describe the frequency domain using a Fourier descriptor and to obtain the low-pass characteristic.

The Fourier descriptor is used to describe a curve on a plane using a frequency domain representation. Z-type [12], G-type [1], and P-type [11] descriptors are possible. A Z-type descriptor carries out a Fourier transformation on a linear function of the accumulation of angle change and length from one point of a closed curve. A Z-type descriptor carries out a Fourier transformation to a complex function of the length, from one point of a closed curve. A P-type descriptor carries out a Fourier transformation on the P-expression using a direction vector between one point and the others. Since a Z-type Fourier descriptor and a G-type Fourier descriptor are difficult to apply to open curves, a P-type Fourier descriptor, which may

be applied to open curves, is used in this process.

Information about an abstract shape of an original curve resides in the low-pass components of the power spectrum, and information about its details resides in high-pass components. The original curve C and the N dimensional reproduction curve C_N approach the original curve C as N becomes large. Detail features are lost and C_N approaches an abstract shape of C as N decreases. The abstract shape of C appears to omit features, such as when a user writes connected strokes., as shown as Figure 2. The generated character has an abstract shape of several combined characters. The connected strokes, which are a natural handwriting, can be obtained using the connected stroke analysis database.

Figure 2: Sample characters of each parameter N of a P-type Fourier descriptor.

3.5 Character Modification

The values of $a, b, c, d, e, f, g,$ and h obtained in the global features extraction are used for the parameters of a nonlinear transformation. A nonlinear transformation is a conversion process for geometric modification. Since the global features of the character are reflected by a nonlinear transfomation, a handwritten-style font reflects the user's individuality, such as their character size and distortion.

4 Font Generation Experiment and Evaluation

The generation results vary according to the differences between three parameters α, β, and C_θ. This chapter describes the font generation, subjective evaluation and the objective evaluation experiments using 25 subjects. The input characters are selected from a set of character having many correspondence strokes, as shown in Figure 3.

4.1 Generation Experiment

Table 2 shows the number of characters that are generated in this system by eight input characters. A large number of characters are generated, even with only eight input characters. There is little cost for users to input characters. Figure 4 shows examples of the generated characters. (a) of the figure shows characters that are using a nonlinear transformation only without

Figure 3: Input characters for the generation experiment.

stroke correspondence. (b) and (c) of the figure show characters that are using stroke correspondence and a nonlinear transformation. (d) of the figure shows the connected strokes by setting $C_\theta = 1.0$. The left part of Figure 4 shows the characters that are identical to the input characters, and the right part shows the new characters that different to the input. It can be seen that individuality is reflected by comparing these with the input characters for every subject.

Table 2: The number of generated characters.

α	0.5	0.6	0.7	0.8	0.9	1.0
Number	2243	1375	723	302	60	22

4.2 Evaluation Experiment

The individuality reflected in the generation results was evaluated using both the degree ranking of subjectively similarity by subjects, and objectively using the degree ranking of the distance [6].

The degree ranking of subjective similarity is performed through evaluation of subjective test results. The conditions of the test are as follows.

1. There are 25 subjects, who are identical to those who supplied input characters.

2. The subjects are shown Figure 4, which is generated from the input characters from the same subjects in this system.

3. The following directions are provided to subjects, and they rank the generation results by their subjective similarity.
   "The following characters were assembled based on your handwriting. Please choose the five sets of characters that you judge to be closest to your handwriting, in order."

The results of the degree ranking of the subjective similarity are shown in Figure 5. "SC" means stroke

No. 1 仁 礼 考 忠 信 悌 智 義 注 恋
No. 2 仁 礼 考 忠 信 悌 智 義 注 恋
No. 3 仁 礼 考 忠 信 悌 智 義 注 恋
No. 4 仁 礼 考 忠 信 悌 智 義 注 恋
No. 5 仁 礼 考 忠 信 悌 智 義 注 恋

(a)

No. 1 仁 礼 考 忠 信 悌 智 義 注 恋
No. 2 仁 礼 考 忠 信 悌 智 義 注 恋
No. 3 仁 礼 考 忠 信 悌 智 義 注 恋
No. 4 仁 礼 考 忠 信 悌 智 義 注 恋
No. 5 仁 礼 考 忠 信 悌 智 義 注 恋

(b)

No. 1 仁 礼 考 忠 信 悌 智 義 注 恋
No. 2 仁 礼 考 忠 信 悌 智 義 注 恋
No. 3 仁 礼 考 忠 信 悌 智 義 注 恋
No. 4 仁 礼 考 忠 信 悌 智 義 注 恋
No. 5 仁 礼 考 忠 信 悌 智 義 注 恋

(c)

No. 1 仁 礼 考 忠 係 悌 智 義 注 恋
No. 2 仁 礼 考 忠 係 悌 智 義 注 恋
No. 3 仁 礼 考 忠 係 悌 智 義 注 恋
No. 4 仁 礼 考 忠 係 悌 智 義 注 恋
No. 5 仁 礼 考 忠 係 悌 智 義 注 恋

(d)

Figure 4: Examples of generated characters. (a) A nonlinear transformation only without stroke correspondence, (b) stroke correspondence and a nonlinear transformation ($\beta = 0$), (c) stroke correspondence and a nonlinear transformation ($\beta = 0.5$), and (d) stroke correspoindence and anonlinear transformation and connected strokes using a P-type Fourier descriptor.

correspondence. Stroke correspondence was not used for "Nonlinear transformation only" curve in the figure to compare an effectiveness of using stroke correspondence.

The ranking consists only of the ten highest, because it is difficult to attach rankings to all characters. The proportion of those characters generated by using stroke correspondence and nonlinear transformation within the tenth place ranking is 80%. It was determined that individuality was subjectively distinguishable, as reflected in the generation results.

The objective degree ranking is based on the distance between characters. The distance is determined from 1 input character and 25 generated characters. The objective degree ranking is determined using the following procedure.

1. Let $C_{t_n}^{(i)}$ be an input character of a category i from subject t_n, and $C_{t_n}^{'(i)}$ be the generated result using $C_{t_n}^{(i)}$ as input.

2. Let $D = (C_{t_n}^{(i)}, C_{tm}^{'(i)})$ denote the distance between $C_{t_n}^{(i)}$ and $C_{tm}^{'(i)}$.

3. Find $D_{n,m}^{(i)}$, with i, n fixed.

4. Find $D_{n,m}^{(i)}$ for all n, m, i.

5. The ranking is determined to be in $D_{n,m}^{(i)}$ order.

Figure 6 shows the ratio of the objective degree ranking in kth order corresponding to $C_{t_n}^{(i)}$, which is the user's own handwriting character of the generation result $C_{t_n}^{'(i)}$. The proportion of results generated by using stroke correspondence and a nonlinear transformation within fifth place is 92%. The objective evaluation was omitted when connected strokes are generated because the distance calculation is too complex. As mentioned above, it was confirmed that individuality was reflected in the generation result.

The ratio of the objective degree ranking is such a high percentage because of the global features extraction and the modification by using nonlinear transformation of generated characters and the local features extracted by using stroke correspondence in this system. The size of input characters and the distortion are extracted as parameters and a nonlinear transformation is conducted to generated characters using the parameters. The global and local features of generated characters resemble the input characters, due to the appearance of global and local features of the user's input characters from the subjective evaluation experiment.

5 Conclusion

This paper describes a method for generating a handwritten-style font that reflects user's individual

of connected strokes for softer and more natural curve.
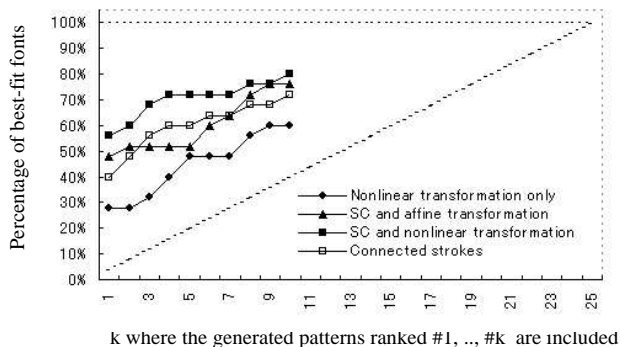

Figure 5: The ratio of subjectively ranked fonts to best-fit fonts.
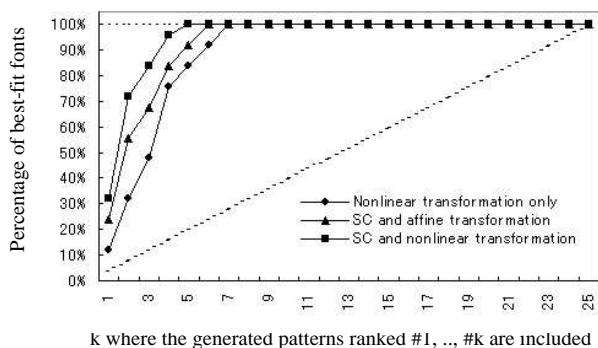

Figure 6: The ratio of objectively ranked fonts to best-fit fonts.

handwriting. The reference stroke database was compressed using vector quantization and an uncompressed stroke was found to correspond to a large number of compressed strokes. Since the user's input strokes correspond to a large number of reference strokes, a handwritten-style font was assembled from the correspondence strokes. In addition, this system is applicable to connected strokes using an optional parameter. Since this system needs few input characters to create a large number of generated characters, it requires little cost for users. It is applicable to creating a font including connected strokes, by using a Fourier descriptor and the connected stroke database. It was also shown that a strongly individual font, a standard font and a font including many connected strokes, correspond to users' selections using the 3 parameters, α, β and C_θ. The subjective and objective evaluation of the font generation experiment by 25 subjects was performed, and confirmed that the user's individuality was reflected by the font.

There are several subjects for future work, including: the improvement of the vector quantization to obtain a better stroke correspondence database; and the analysis of input characters for more effective generation

References

[1] G. Granlund. Fourier preprocessing for hand print character recognition. *IEEE Trans. Computers*, C-21:195–201, 1972.

[2] H. H.Kasuga and M.Okamoto. Color quantization using fast k-means algorithm. *IEICE Trans.*, J82-D-II(7):1120–1128, 1999.

[3] H.Yamada. Continuous nonlinearity in character recognition. *IEICE Trans.*, E79-D(5):423–428, 1996.

[4] K. M.Hamanaka and J.Tsukumo. On-line japanese character recognition based on flexible pattern matching method using normalization-cooperative feature extraction. *IEICE Trans.*, E77-D(7):825–831, 1994.

[5] M.Shiono. A method for generation of handwritten-style character patterns using non-linear geometrical distortion. *IEICE Trans.*, J74-D-II(2):209–219, 1991.

[6] T. M.Yasumoto and H.Horii. A method of generation handwriting style fonts based on global and local features in individuals' handwriting. *IEICE Trans.*, J80-D-II(11):2930–2939, 1997.

[7] R. Plamondon and S. N. srihari. On-line and off-line handwriting recognition: A comprehensive survey. *IEEE Trans. PAMI*, 22(1):63–84, 2000.

[8] J. S.Ando and M.Nakajima. Automatic handwritten-like font creation that reflects individual writing habit. *The seventh Symposium on Sensing via Image Information Trans.*, SSII:177–180, 2001.

[9] T. S.Ando and M.Nakajima. Development of writing habit including character font generator. *The eighth Symposium on Sensing via Image Information Trans.*, SSII:191–196, 2002.

[10] J. Shin. A study of the compression method for a reference character dictionary used for on-line character recognition. *Proceedings of the Seventh International Conference on Knowledge-Based Intelligent Information*, pages 300–309, 2003.

[11] Y.Uesaka. A new fourier descriptor applicable to open curves. *IEICE Trans.*, J67-A(3):166–173, 1984.

[12] C. Zahn and R. Roskies. Fourier descriptors for plane closed curves. *IEEE Trans. Computers*, C-21:269–281, 1972.

Chinese Unknown Word Identification Based on Local Bigram Model with Integrally Smoothing Assumption[*]

Zhuoran Wang   Ting Liu
School of Computer Science and Technology,
Box 321, Harbin Institute of Technology,
Harbin, P.R.China, 150001
{taliux, tliu}@ir.hit.edu.cn

ABSTRACT

The paper presents a Chinese unknown word identification system based on a local bigram model. Generally, our word segmentation system employs a statistical-based unigram model. But to identify those unknown words, we take advantage of their contextual information and apply a bigram model locally. To explain this local approximation, we make an "integrally smoothing assumption". As a simplification of bigram, this method is simple as well as feasible, since the complexity of its algorithm is quite low and not so many training corpora are needed. The results of our experiments show the solution is really effective.

1   INTRODUCTION

Unlike Indo-European languages, Chinese words do not have spaces to mark their boundaries. So word segmentation is the foundation of Chinese natural language processing (NLP). A system with a lexicon could perform quite well, however, the unknown words which are not registered in the lexicon, become the bottle-neck of the precision and recall. Simply enlarging vocabulary will not work, because those unknown words are so vast and various as to be collected exhaustively. Unknown word identification (UWI) therefore plays a significant role in word segmentation.

In this paper, we illustrate an approach to recognize unknown words, which is based purely on statistics and probabilities, and involves information only on word level. First, a process to specify an unknown word candidate is triggered on some certain conditions (a Chinese character that can form a surname will cause a specifying process of a candidate of person name, for example), and an initial probability of this word is calculated from its internal formation. Then the context information of it is taken into consideration, with a local bigram model applied here, whose details will be discussed later in Section 3. At last the specified candidates of unknown words, together with normal lexicon words, will go through a

searching procedure to get a solution that meets the maximum probability, or to generate N best results serving further processing [1].

Because of the limitation from our system and platform, the unknown words that we identified only include Chinese person names (PER), location names (LOC), numerals (NUM) and time expressions (TIME). The evaluation is conducted on a large set of corpus segmented and tagged manually, while the experimental results show the efficiency of our method. The precision of PER, LOC, NUM and TIME on the test set is 82.1%, 80.5%, 91.7%, 95.5%, respectively; and the recall is 87.0%, 71.3%, 90.6%, 93.7%, respectively. There are 75.1% of the results matching the standard ones perfectly, if five best results for one input are saved.

2   RELATED WORK

Recently, many techniques have been proposed for Chinese unknown words identification. Lv *et al* [2] presented a rule-based method, which used unknown words' internal structure and context relations to develop its evaluation functions, and resolved them by a dynamic programming. Tan *et al* [3] expounded a transformation-based machine learning approach. They focused on the problems of Chinese place name recognition, with a similar idea with Brill algorithm [4], and eliminated many incorrectly recognized sequences by transformation rules. And Zhang *et al* [5] revealed a universal method based on roles tagging. They tagged the roles of words' component tokens by applying a Viterbi algorithm in the fashion of a POS tagger. As a result their system could recognize several types of unknown words.

In the last few years, some class-based models were introduced. For example, Sun *et al* [6] integrated word segmentation and named entity identification into a unified framework and developed the class-based LM with their own features. Zhang *et al* [7] improved their role-based model and added a class-based segmentation. Fu and Luke [8] also combined the model based on classes with their word juncture models and word-formation patterns for UWI.

We benefit form their approaches, and explore a statistical-based model that fit our own word segmentation system and NLP platform.

1.Intersection with previous word	这里/*有*/关天培/烈士/的/遗物/。(There are relics of Martyr Guan Tianpei here.)
2.Intersection with posterior word	邓小平/*等*/领导 (Deng Xiaoping etc.)
3.Surname with a lexicon word	*王霞光* (Wang Xiaguang)
4.Being a lexicon word simultaneously	*黎明* (Li Ming, with another meaning of dawn)
5.Combining ambiguity	a) *记者/全中/报道*。（Reported by Quan Zhong.） *他/十/枪/全/中/靶心/*。(His ten shots all hit the bull's-eye.) b) *1000/余/万/元* (more than 10 million *yuans*) *村长/余万元/有/一/个/儿子/*。(Yu Wangyuan, the leader of village, has a son.)

<center>Table 1: Examples of Ambiguities Related with Person Names</center>

3 LOCAL BIGRAM MODEL FOR UWI

The noisy channel model is a classic language model for word segmentation, of which the actual application could be an n-gram model. In this section, we integrate the UWI process to a word segmentation system with such a model, and employ a local bigram method, with the assumption that the rest parts, where unigram is used, are smoothed.

3.1 The Classic Noisy Channel Model for Word Segmentation

Given a sentence that can be regarded as a sequence of Chinese characters $S=s_1s_2...s_n$, the task of word segmentation is to find a separated sequence $W=w_1w_2...w_n$, whose posterior probability:

$$P(W \mid S) = \frac{P(W)P(S \mid W)}{P(S)} \qquad (3.1.1)$$

is maximum over all possible ones. For a fixed sequence, $P(S)$ is a constant, and $P(S|W)=1$ because the sequence of Chinese characters is unique for any result word sequence, then:

$$\hat{W} = \arg\max_W P(W) \qquad (3.1.2)$$

where $\hat{W}$ is the recognized word sequence. Stochastic language models are usually used to calculate $P(W)$, with which, for a sequence of words W, $P(W)$ can be written as:

$$P(W) = \prod_{i=0}^{N} P(w_i \mid w_0...w_{i-2}w_{i-1}) \qquad (3.1.3)$$

However, in practice, it is impossible to compute the exact conditional probabilities $P(w_i|w_0...w_{i-2}w_{i-1})$ for a long sentence. So we assume that a given word's probability only depends on its *n* previous words. The equation (3.1.3) is expressed as:

$$P(W) = \prod_{i=0}^{N} P(w_i \mid w_{i-n}...w_{i-2}w_{i-1}) \qquad (3.1.4)$$

Then we call it an n-gram model. Bigram (*n=1*) and unigram (*n=0*) are both popularly used instances.

3.2 The Unigram Model's Dilemma

A research shows that ambiguities occur 1.2 times per 100 Chinese characters, while the ratio of intersection ambiguities and combining ambiguities is 12:1 [9]. So unigram could work rather well, as it solves intersection ambiguities effectively, which are major problems for word segmentation. But for UWI, it is not always competent. The formation of Chinese unknown words (especially Chinese person names) is so complex, that it can generate several types of ambiguities as Table 1 shows. It can be seen that information only from the word formation pattern itself is not adequate. Maybe, in Case 1 or Case 2, it might still work. But if the input sentence is like Case 3 or Case 4, the corresponding probabilities will probably be partial to the lexicon words. Furthermore, in Case 5, we could not even judge whether it is a person name or not. Considering that, context information is indispensable to UWI.

A rule-based weighting is efficient, but for a statistical-based model that outputs N best results for further refining, the weight added illogically may confuse the evaluation process, which obeys probability originally, to mark and order the results. Then bigram model is chosen, and indeed, it performs better. Yet, the training of the transition probabilities of every two words calls for an enormous collection of corpus, which is not handy at all time. If used as a rough segmentation, it would cost a lot of time and memory for the decoding to gain N best results. Since unigram already can do so much, the cost-efficiency of bigram is not that high. Hence we decide to find a compromise.

3.3 Local Bigram Model

Observing the $P(W)$ of bigram model, it is generally computed under the Maximum Likelihood paradigm as:

$$P(w_i \mid w_{i-1}) \approx \frac{C(w_{i-1}, w_i)}{C(w_{i-1})} \qquad (3.3.1)$$

where $C(w_{i-1}, w_i)$ is the frequency of the co-occurrences of w_i and w_{i-1}, and $C(w_{i-1})$ is the total frequency of w_{i-1} in the corpus. Here we prefer to write it as:

$$P(w_i \mid w_{i-1}) \approx \frac{P(w_{i-1}, w_i)}{P(w_{i-1})} \qquad (3.3.2)$$

by dividing the numerator and denominator with a same factor $C(Total)$, the total of the words in the

Unknown Word	Related Word Class	Expression	Example
PER	Appellation	NC	记者王南 (Reporter Wang Nan)
LOC	Preposition	P	在沈阳市 (in Shenyang)
	Orientation	F	哈尔滨北部 (northern part of Harbin)
NUM	Postfixe of numeral	PX	三千多 (more than three thousand)
	Quantifier	Q	一百张 (a hundred papers of)
TIME	None		

Table 2: The Classes Having Dependencies with Unknown Words

corpus, for the convenience of latter illations. But if $P(w_{i-1}, w_i)=0$ in the training corpus, the element whose $P(w_i/w_{i-1})$ is 0, will make it terrible on test data. Considering this, many smoothing ways have been devised, one of which is interpolation. With this method unigram and bigram are combined, and formula (3.3.2) is rewritten as:

$$P(w_i \mid w_{i-1}) \approx \lambda \frac{P(w_{i-1}, w_i)}{P(w_{i-1})} + (1-\lambda)P(w_i) \quad (3.3.3)$$

in which λ $(0 \le \lambda \le 1)$ is usually determined by optimizing on the "held-out" data.

According to this principle, we make an **"integrally smoothing assumption"** that in a sentence, probabilities of the rest parts' co-occurrences are all 0, so they are smoothed and only have factors like $(1-\lambda)P(w_i)$, except the unknown words candidates identified and some local areas associated with them whose probabilities will have dependencies with their previous words. Here, we must also make a definition to the "**local area**": We consider an unknown word together with its one previous word or one posterior word (both are the longest) as a local area. Then the bigram model is applied in this extent correspondingly. Again we divide every $P(w_i/w_{i-1})$ with a factor $(1-\lambda)$, as:

$$P*(w_i \mid w_{i-1}) = \frac{\lambda}{1-\lambda} \frac{P(w_{i-1}, w_i)}{P(w_{i-1})} + P(w_i) \quad (3.3.4)$$

And let:

$$\hat{W}* = \arg\max_W \prod_i P*(w_i \mid w_{i-1}) \quad (3.3.5)$$

which will have the same outcome with (3.1.2). After that, the general parts of the sequence inputted will have $P*(w_i/w_{i-1})=P(w_i)$ that just equal the probabilities with unigram. Whereas, take the sentence "教授孟西安" (Professor Meng Xi-An) as an example, the unknown word candidate "孟西安" (Meng Xi-An) will have such a $P*$ as:

$$P*(孟西安 \mid 教授) = \frac{\lambda}{1-\lambda} \frac{P(教授, 孟西安)}{P(教授)} + P(孟西安)$$

which is enlarged by its previous word "教授" (professor), for "教授" (professor) is a appellation word often appearing ahead a person name. In this way, the **"local bigram model"** is created.

On the second thoughts, to avoid the data sparseness that remains serious when the probabilities are computed based on exact word, we actually apply a clustering method. Besides unknown words that take

their types as their classes, we also defined several word classes that have a close relation with them. Table 2 shows the details. The other words' effects to unknown words candidates are so slight that we ignore them. For the above classes, we can believe that in the defined local area if a class C_i contains a word w_i, then w_i belongs and only belongs to C_i. So we can suppose $P(w_i) = P(w_i, C_i)$

According to that, for an unknown word candidate, formula (3.3.4) is transformed to:

$$P*(w_i \mid w_{i-1}) = \frac{\lambda}{1-\lambda} \frac{P(w_{i-1}, w_i)}{P(w_{i-1})} + P(w_i)$$

$$= \frac{\lambda}{1-\lambda} \frac{P(w_{i-1}, C_{i-1}, w_i, C_i)}{P(w_{i-1})} + P(w_i)$$

If complete probability formula is used, then:

$$P*(w_i \mid w_{i-1}) = \frac{\lambda}{1-\lambda} \frac{P(C_{i-1}, C_i)P(w_{i-1}, w_i \mid C_{i-1}, C_i)}{P(C_{i-1})P(w_{i-1} \mid C_{i-1})} + P(w_i)$$

Suppose w_{i-1} and w_i are independent under the condition C_{i-1}, C_i:

$$P*(w_i \mid w_{i-1})$$
$$= \frac{\lambda}{1-\lambda} \frac{P(C_{i-1}, C_i)P(w_{i-1} \mid C_{i-1}, C_i)P(w_i \mid C_{i-1}, C_i)}{P(C_{i-1})P(w_{i-1} \mid C_{i-1})} + P(w_i)$$

Further, suppose w_i is independent of C_j $(i \ne j)$:

$$P*(w_i \mid w_{i-1}) = \frac{\lambda}{1-\lambda} \frac{P(C_{i-1}, C_i)P(w_{i-1} \mid C_{i-1})P(w_i \mid C_i)}{P(C_{i-1})P(w_{i-1} \mid C_{i-1})} + P(w_i)$$

$$= \frac{\lambda}{1-\lambda} \frac{P(C_{i-1}, C_i)P(w_i \mid C_i)}{P(C_{i-1})} + P(w_i)$$

We can also express it as:

$$P*(w_i \mid w_{i-1}) = \left[\frac{\lambda}{1-\lambda} \frac{P(C_{i-1}, C_i)}{P(C_{i-1})P(C_i)} + 1 \right] P(w_i) \quad (3.3.6)$$

The initial probability $P(w_i)$ of the candidate is calculated as:

$$P(w_i) = P(w_i \mid C_i)P(C_i) \quad (3.3.7)$$

The conditional probability $P(w_i/C_i)$ is generated from its internal formation patterns:

$$P(w_i \mid C_i) = P(s_0^i \mid C_{Beg}^i)P(s_m^i \mid C_{End}^i)\prod_{j=1}^{m-1} P(s_j^i \mid C_{Mid}^i)$$

$$(3.3.8)$$

3.4 Decoder

The decoding algorithm is much simplified. We create a directed acyclic graph for an input sentence, of which the vertexes represent the spaces of every

two Chinese characters and the edges denote there could be words. The weight of each edge is set with $-ln[P^*(w_i|w_{i-1})]$. As it is generally a unigram model, a Dijkstra searching with some judgments is enough. It can also gain N best results conveniently just by adding a storing structure to the graph.

3.5 Sample Results and Discussion

With the above model, we solve many problems that were inextricable with unigram before. There are some examples:

1. *"朱琳夫人"* (Mrs. Zhu Lin)

$$P(朱琳_{PER})P(夫人 \mid 朱琳_{PER})$$

$$= P(朱琳_{PER})\left[\frac{\lambda}{1-\lambda}\frac{P(PER, NC)}{P(PER)P(NC)} + 1\right]P(夫人)$$

will be much greater than $P(朱琳夫_{PER})P(人)$.

2. *"他十枪全中靶心。"* (His ten shots all hit the bull's-eye.) Comparing with *"记者全中报道。"*(Reported by Quan Zhong.)

As "全" and "中" are both single-character words with high frequency in the corpus, $P(全)P(中)$ is greater than $P(全中_{PER})$. Therefore, the first sentence can be segmented correctly as: "他/十/枪/全/中/靶心/。" And because of the weighting of *"记者"* to the person name candidate *"全中"*, the second one can be segmented as *"记者/全中/报道"* correctly, as well.

3. *"1000 余万元"* (more than 1000 *yuans*)

After weighting procedure the probability of the sequence is like:

$$P(1000)P(余 \mid NUM)P(万 \mid PX)P(元 \mid NUM)$$

which prevents the result "余万元$_{PER}$".

But there remain some ambiguities that can not be solved. For example *"邓稼先生于1886 年。"* (Mr. Deng Jiaxian was born in 1886.) We can not decide whether it is *"邓稼先/生于..."* or *"邓稼/先生/于..."* only by this model. However, bigram even trigram can not solve many instances of this kind, either.

4　EXPERIMENTS

4.1　Evaluation Metric

The precision, recall and F-scores of our system are conducted by following formulae:

$$P = \frac{\text{Number of Correctly Identified}}{\text{Number of Identified Unknown}}$$

$$R = \frac{\text{Number of Correctly Identified Unknown}}{\text{Total Number of Unknown}}$$

$$F = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

We also record the percentage of N-best results that entirely match the standard ones. It is defined as:

$$R_{PM} = \frac{C(PerfectMatch)}{C(Sentence)}$$

where *C(PerfectMatch)* is the number of the output sentences one of whose N results perfectly matches the standard result, and *C(Sentence)* is the total number of input sentences.

4.2　Experimental Data

The lexicon used in our system contains about 80,000 words. We also have an additional dictionary that covers the information related with (3.3.8). Several tests are taken on a corpus set containing 1,2000 sentences (expanded in Table 3) extracted from the news of *People's Daily* in the first half year of 1998 [10]. To emphasize the local bigram's effect, we compare it with the result based on simple unigram model. At last we make an open evaluation on the test data for the track of PK-open in 2003 SIGHAN Bakeoff [11] to contrast our system with the other public systems in the track.

Class	Words	Rate
Lexicon word	243997	92.1%
PER	2757	1.0%
LOC	122	0.05%
TIME	4737	1.6%
NUM	6879	2.6%

Table 3: Details of Test Corpus

4.3 Performance of UWI

Class	P	R	F
PER	82.1%	87.0%	84.5%
	84.0%	78.9%	81.4%
LOC	80.6%	71.3%	75.7%
	77.4%	67.2%	71.9%
TIME	95.5%	93.7%	94.6%
	95.5%	93.7%	94.6%
NUM	91.7%	90.6%	91.1%
	91.6%	90.4%	91.0%
All the words out of lexicon	90.3%	65.6%	76.0%
	89.7%	63.9%	74.6%
Full word segmentation system	95.5%	96.2%	95.8%
	94.0%	94.7%	94.3%

Table 4: The Results of Experiments for UWI

The table above shows the performance of our system. It displays the precision, recall and F-scores for the identification of each unknown word's type apart. The two lines in each row refer to the system with local bigram model and just with unigram model respectively.

As our system is also in charge of generating N best results of every input for further processing, we record the perfect match rate (viz. R_{PM}) in the table below while N increases from 1 to 5.

N Paths	R_{PM}
1	63.4%

2	71.4%
3	73.3%
4	74.4%
5	75.1%

Table 5: Perfectly Matching Rate

4.4 Evaluation with SIGHAN Bakeoff Data

The comparison among our system and other systems in 1st SIGHAN Chinese word segmentation bakeoff is summarized as below. We would like to give some more explanations as well.

Site	R_{OOV}	R_{iV}	R	P	F
S10	79.9%	97.5%	96.3%	95.6%	95.9%
S01	74.3%	98.0%	96.3%	94.3%	95.3%
S08	67.5%	95.9%	93.9%	93.8%	93.8%
S04	71.2%	94.9%	93.3%	94.2%	93.7%
S03	64.7%	96.2%	94.0%	91.1%	92.5%
S11	50.3%	93.4%	90.5%	86.9%	88.6%
Our System	65.0%	95.0%	93.1%	93.4%	93.2%

Table 6: The Results of SIGHAN PK-open Test

Through the results of PK-open test, we can see although there is a slight distance between our system and some excellent ones, it is not a wide gap. Except the systems that make use of the information from higher levels such as S01 [12], we prefer to compare our system with those just based on word level: S04 [13] is a good selection, which used bigram method for word segmentation and introduced a word juncture model combined with word-formation pattern to identify unknown words. We could say our system performs almost as well as theirs measured either by F-score or by P or R. So it proves local bigram model can solve the chief problems to a large extent, while the complexities are not increased.

5 CONCLUSION

We have presented a local bigram model for unknown word identification. In this model, we cluster the words that have important effect for identifying unknown words to several classes, and make use of their dependencies locally in the form of conditional probabilities, while assuming the other parts, where unigram is used, are smoothed. We take advantage of the convenience of unigram together with the dependent information of bigram. The system brings a satisfying outcome, either from its best result or from the N best results. Since its complexities of both time and space are much lower than many general algorithms, this method is quite practical when training data is not ample, or in a resource-limited environment such as PDA

REFERENCES

[1] Zhang Huaping and Liu Qun. 2002. Model of Chinese Words Rough Segmentation Based on N-Shortest-Paths Method. *Journal of Chinese Information Processing. 16, 5, 77-83.*

[2] Lv Yajuan, Li Tiejun, Yang Muyun, Yu Hao and Li Sheng. 2000. Leveled Unknown Chinese Words Resolution by Dynamic Programming. *Journal of Chinese Information Processing.15,*

[3] Tan Hongye, Zhang Jiaheng and Liu Kaiying, 2001, Research on Model of Automatic Recognition of Chinese Place Name Based on Transformation. *Journal of Software. 12, 11, 1608-1613.*

[4] Eric Brill. 1995. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging. *Computational Linguistics. 21, 4, 418-433.*

[5] Zhang Hua-Ping, Qun Liu, Hao Zhang and Xue-Qi Cheng. 2002. Automatic Recognition of Chinese Unknown Words Based on Role Tagging, *First SIGHAN affiliated with 19th COLING 2002: 71-77*

[6] Sun Jian, Jianfeng Gao, Lei Zhang, Ming Zhou and Chang-Ning Huang. 2002. Chinese named entity identification using class-based language model. *COLING 2002.*

[7] Zhang Hua-Ping, Qun Liu, Hong-Kui Yu, Xue-Qi Cheng and Shuo Bai. 2003. Chinese Named Entity Recognition Using Role Model. *Computational Linguistics and Chinese Language Processing. 8, 2, 29-60.*

[8] Guohong Fu and Kang-Kwong Luke. 2004. Chinese Unknown Word Identification using Class-based LM. *IJCNLP2004: 262-269*

[9] Maosong Sun and Benjamin K. Tsou. A Review and Evaluation on Automatic Segmentation of Chinese. *Contemporary Linguistics.200101, 22-32*

[10] Yu Shiwen, Huiming Duan, Xuefeng Zhu, Bin Swen and Baobao Chang. 2003. Specification for Corpus Processing at Peking University: Word Segmentation, POS Tagging and Phonetic Notation. *Journal of Chinese Language and Computing, 13, 2, 121-158*

[11] Richard Sproat and Thomas Emerson. 2003. The First International Chinese Word Segmentation Bakeoff. *Second SIGHAN workshop affiliated with 41th ACL; Sapporo Japan. 133-143*

[12] Zhang Hua-Ping, Qun Liu, Xue-Qi Cheng, Hao Zhang and Hong-Hui Yu. 2003. Chinese Lexical Analysis Using Hierarchical Hidden Markov Model. *Second SIGHAN workshop affiliated with 41th ACL; Sapporo Japan. 184-187*

[13] Guohong Fu and Kang-Kwong Luke. 2003. A Two-stage Statistical Word Segmentation System for Chinese. *Second SIGHAN workshop affiliated with 41th ACL; Sapporo Japan. 156-159*

Reviewers

A

Tim Arndt

B

Nadia Berthouze
Josef Bigun
Alberto Del Bimbo
Marc H. Brown

C

K. Chang
Kai H. Chang
Rwei-Ching Chang
Shi Kuo Chang
Shih-Fu Chang
Shih Fu Chang
Han-Chieh Chao
H. H. Chen
Ing-Ray Chen
K. J. Chen
K. Y. Chen
Ko-Shung Chen
Meng Chang Chen
Shu-Ching Chen
Tzung-Shi Chen
Yuh-Shyan Chen
Kent Cheung
Chuan-Feng Chiu
Yonghee Choi
Zied Choukair
C. Chu
C. C. Chu
Tat-seng Chua

D

Fadi P. Deek
Lawrence Y. Deng
Ramachandra A. Dixit
Ralph Doerner
Jing Dong
C. R. Dow

David H. C. Du
Jean-Luc Dugelay

E

Larbi Esmahi

F

David Forsyth
Borko Fuhrt
George Furnas

G

Theo Gevers
Athula Ginige
Stuart Goose
Carsten Griwodz
William Grosky
Angela Guercio
Stephen Guest

H

Masahito Hirakawa
Peter Douglas Holt
J. T. Horng
Chun-Liang Hsu
Hui-huang Hsu
I. L. Huang
Jason C. Hung
Lun-Ping Hung
W. Hwang

I

Horace Ip

J

Lin-Her Jeng
Qun Jin
Y-K J. Jong
Erland Jungert

K

Michelle T. C. Kao
Irwin King
Dieter Kranzlmller
Zenon Kulpa
T. Kunii
James Kwok
K. L. Kwok

L

Inald Lagendijk
Wei Lai
L. Lam
Robert Laurini
Dong-Liang Lee
H. J. Lee
S. W. Lee
M. K. Leong
Wee Kheng Leow
Clement Leung
Stefano Levialdi
Michael Lew
Paul Lewis
M. H. Li
Sheng-Tun Li
T. C. Li
Wen-Syan Li
Mark Liao
Fuhua Lin
Kuan-Cheng Lin
Shieh-Shing Lin
Chien-Tsai Liu
Jonathan Liu
Yanxi Liu
YingQun Liu
R. Z. Lu

M

Jinhua Ma
Jian-Hua Ma
M. Ma
Kim Marriott

Rym Mili
Nick Mirenkov

N

Marc Najork
Mario Nascimento

O

Chikara Ohta
Vincent Oria
Ming Ouhyoung

P

Jen-Kuei Peng
Fernando Pereira
Joseph J. Pfeiffer

R

Syed M. Rahman

S

Magdy Saeb
Prasan Kumar Sahoo
Masao Sakauchi
Hanan Samet
Simone Santini
Raimondo Schettini
Yoshitaka Shibata
Timothy K. Shih
Arnold Smeulders
Ronggong Song
David Stotts
C. Y. Suen
Willy Susilo

T

Makoto Takizawa
H Lilian Tang
Steven L. Tanimoto
Gustavo A. S. Torrellas

Genny Tortora

U

Kuniaki Uehara
Joseph E. Urban

V

Athena Vakali
Theodora Varvarigou
Pallapa Venkataram
Son T. Vuong

W

San-Yuan Wang
Chun-Chia Wang
Ching-Sheng Wang
H. C. Wang
J. F. Wang
S. Wang
Guido Wirtz

Ramond Wong
Marcel Worring
Jian Kang Wu

X

X. W. Xu

Y

Shi-Nine Yang
C. Yang
T. Yao
Rentaro Yoshioka
Gwo-Jong Yu
C. F. Yuan

Z

C. Zeng
Chi Zhang
Kang Zhang

Authors' Index

Call for Papers
The Eleventh International Conference on Distributed Multimedia Systems (DMS'2005)

Banff Spring Hotel, Banff, Canada
September 5 - 7, 2005

Organized by
Knowledge Systems Institute

The DMS conference is an international conference series, which covers a wide spectrum of technique discussions, demonstrations, and student program/paper contests in the fields of distributed multimedia computing. Started in 1994, the series of conference has been held at Hawaii, Hong Kong, Vancouver, Taipei, Aizu-Wakamatsu, Japan, San Francisco and Miami. This time, the conference will be held at the Sofitel Hotel, San Francisco Bay, USA with video conferencing in other universities/organizations in US, Japan and elsewhere.

DMS'2005 focuses on techniques, systems, applications, and theories in the fields of distributed multimedia computing. The conference organizers seek contributions of high quality papers, panels or tutorials, addressing various aspects of distributed multimedia systems and applications, for presentation at the conference and publication in the proceedings. Topics of interest include, but are not limited to:

- audio and video compression
- MPEG, Quicktime, Windows API standards
- image, video, audio content analysis and indexing/retrieval
- image, video and audio watermark
- 3D audio and video
- computer graphics and animation
- modeling and analysis of distributed multimedia systems
- OS support for distributed multimedia systems
- distributed multimedia databases and computing
- multi-paradigmatic information retrieval
- multimedia human-computer interaction
- multimedia communications and network architectures
- mobile networks and mobile computing
- multimedia stream synchronization and QoS control/scheduling
- multimedia software engineering
- multimedia processors and ASIC
- multimedia technologies for people with disability
- intelligent multimedia computing
- intelligent multimodal interaction
- multi agents, mobile agents and their applications
- communication encryption
- security of distributed computer systems
- Web servers and services
- XML applications
- Java, VRML and multimedia languages
- visual and multidimensional languages for multimedia applications
- multimedia digital libraries and mail systems
- multimedia applications for CSCW
- multimedia authoring tools and intelligent tutoring
- cultural heritage multimedia applications
- tele-conferencing, tele-medicine and tele-lecture
- virtual reality, distributed virtual environment, and their applications
- virtual school, virtual university, and virtual learning community
- distance learning methodology, tools and systems
- e-commerce, e-education and e-entertainment

The use of prototypes and demonstration video for presentations is encouraged.

Workshops and Special Sessions
This year, DMS'2005 will be held in conjunction with workshops, conferences and special sections. Papers submitted to workshops or special sessions are invited by the program committee chairs of the workshops/sections. This joint organization aims to collect research results from different perspectives. The following workshops and conferences are being planned:

International Workshop on Visual Languages and Computing
International Conference on Visual Information Systems
International Workshop on Mobile Systems, E-Commerce and Agent Technology
The 2005 International Workshop on Distance Education Technology DET2005

Please contact the conference program co-chairs if you are interested in organizing a workshop or a special session. A one-page proposal with the following items is required:

- *Title of the workshop/special session*
- *Name of the program committee chair(s)*
- *A list of program committee members*
- *E-mail address of the corresponding program committee chair*
- *A brief description of the theme*

Each special session should include at least 5 invited (or accepted) papers. Each workshop should include at least 2 special sessions (10 papers). Paper review process of the workshop/special session will be arranged by the individual program committee chair. It is the responsibility of the individual chairs to ensure the quality of papers in the workshop/special sessions. The schedule of workshops/special sessions will be the same as the main conference (See Important Dates below). Each individual program committee chair can set up the Web site of the workshop/special session. However papers will be printed in the same volume as the main conference.

Special issue of Journals
Best ranked papers accepted at the conference will be suggested for publication in a special issue of an international journal, subject to the review by the program committee. Papers suggested for the special issue will then be reviewed by external reviewers following the standard procedure of review stipulated by the international journal.

Conference Site (Hotel Information)
The DMS'2005 Conference will be held at the Banff Springs Hotel, one of the most famous landmark hotels in Canada. Banff Springs has graciously made available to DMS'2005 attendees a remarkable discount rate of $239 Canadian dollars for single/double (approximately $182 US dollars) per night. The regular room rate is $539.00 per night and it is worth the stay in one of the most fabulous hotels even for one night to enjoy the medieval architecture, interior decoration and delightful atmosphere in Banff Springs. If you plan to attend and would like to share with another conference attendee, please notify the DMS'2005 Conference Secretariat and provide your check-in/check-out date and your name. We will make the necessary arrangements for you; however, we can not guarantee that we will match your requirements. A limited number of rooms are available, room rate does not include tax and service fee. The hotel reservation number is 800-441-1414. For additional information, please visit the Hotel website at www.Fairmont.com/Banffsprings.

Information for Authors
Papers must be written in English. An electronic version (Postscript, PDF, or MS Word format) of the full paper should be submitted using the following URL: **http://conf.ksi.edu/dms05/submit/SubmitPaper.php**. Please use Internet Explorer as the browser. Manuscript must include a 200-word abstract and no more than 6 pages of IEEE double column text (include figures and references).

Information for Reviewers
Papers submitted to DMS'05 will be reviewed electronically. The users (webmaster, program chair, reviewers...) can login using the following URL: **http://conf.ksi.edu/dms05/review/pass.php**. If you have any questions or run into problems, please send e-mail to: **dms@ksi.edu**.

DMS'2005 Conference Secretariat
Knowledge Systems Institute
3420 Main Street, Skokie, IL 60076 USA
Tel: 847-679-3135, Fax: 847-679-3166
E-mail: dms@ksi.edu

If you cannot submit electronically, please send four copies of the complete manuscript to the above postal address.

Important Dates

March 31, 2005	*Paper submission due*
June 1, 2005	*Notification of acceptance*
July 1, 2005	*Final manuscript due*
July 1, 2005	*Early conference registration due*