



Regione Toscana



---

# Documento con modello di funzionamento del processo di brokeraggio, implementazione dell'algoritmo ID 3.26.1

---

Versione 1.4  
Data: 31/12/2013

---



**Progetto iCaro**

La piattaforma cloud per l'accelerazione  
del business delle PMI toscane  
[CUP 6408.30122011.026000074]

**COMPUTER  
GROSS**

  
liberologico.com



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE  
**DINFO**  
DIPARTIMENTO DI  
INGEGNERIA  
DELL'INFORMAZIONE



 **altro  
lavoro**  
Agenzia per il lavoro

---



## Informazioni sul documento

ID Deliverable	3 26.1
Titolo Deliverable	Documento con modello di funzionamento del processo di brokeraggio, implementazione dell’algoritmo.
ID Attività	3.5
N. Versione / Revisione	1.4
Natura: Bozza / Definitivo	Definitivo
Partner responsabile	UNIFI
Distribuzione: Riservato / Pubblico	Pubblico
Riferimenti Autore	Brunelli
Data redazione	23/12/2012
Riferimenti revisore	Paolo Nesi
Data revisione	31/12/2012
Riferimenti soggetto che approva	Paolo Nesi
Data approvazione e consegna	31/12/2012

## Controllo delle revisioni

Oggetto	Numero	Data
Prima stesura	1.0	03/12/2013
Seconda stesura	1.1	06/12/2013
Terza stesura	1.2	12/12/2013
Quarta stesura	1.3	20/12/2013
Chiusura	1.4	31/12/2013



## Nota di riservatezza

Il presente documento sarà utilizzato esclusivamente ai fini del progetto ICARO, ha carattere riservato e non potrà quindi essere divulgato se non in seguito ad esplicita autorizzazione scritta da parte dell'ATS, salvo il caso in cui di richieste di ottemperare ad obblighi di legge o a richieste di pubbliche autorità.



## Indice

1. Introduzione .....	6
2. Metodologia .....	6
3. Stato dell’arte .....	9
3.1 Schema di classificazione .....	9
3.1.1 Tipologie di implementazione .....	9
3.1.2 Metodi di analisi e tecniche risolutive.....	10
3.1.3 Funzioni obiettivo e risorse considerate .....	11
3.1.4 Modalità della validazione .....	12
3.1.5 Cloud commerciali.....	13
3.1.6 Software per la simulazione cloud .....	14
3.1.7 Processo di brokeraggio .....	15
4. I sistemi di brokeraggio .....	16
4.1 Le funzionalità di predizione nel broker.....	18
Bibliografia.....	23

## Indice figure

Figura 1 – Algoritmo di analisi .....	8
Figura 2 – Trend di pubblicazione dei paper in ambito CC.....	8
Figura 3 – Principali Journal di riferimento .....	8
Figura 4 – Architettura del Cloud broker (fonte: Lucas-Simarro et al. 2012 [83]) .....	18
Figura 5 – L’architettura del modello (fonte [151]).....	19
Figura 6 – L’architettura di cloud scaling (fonte [141]) .....	19
Figura 7 – Carico di lavoro attuale e predetto (fonte [141]) .....	20
Figura 8 – Previsione del carico di lavoro di una CPU (fonte [150]).....	20
Figura 9 – L’architettura del modello predittivo (fonte [148]).....	22

## Indice tabelle

Tabella 1 – Tipologie di implementazione.....	10
Tabella 2 – Tipologia di analisi.....	10
Tabella 3 – Tecniche di risoluzione.....	11
Tabella 4 – Obiettivo dell’ottimizzazione matematica.....	11
Tabella 5 – Risorse oggetto di ottimizzazione .....	12
Tabella 6 – Modalità di validazione .....	12
Tabella 7 – Caratteristiche principali prodotti commerciali adattato da Zang 2010 [146] .....	14
Tabella 8 – Principali prodotti commerciali.....	14
Tabella 9 – Software di simulazione per il cloud (fonte: Núñez 2012 [98]) .....	15
Tabella 10 – Software di simulazione per il cloud .....	15
Tabella 11 – Servizi cloud broker.....	16



## Legenda Acronimi e sigle

Acronimo / Sigla	Dettaglio
CC	Cloud Computing
VM	Virtual Machine
SLA	Service Level Agreement
QoS	Quality of Service
CP	Cloud Provider



## 1. Introduzione

Questo documento riporta i risultati di un’analisi strutturata della letteratura scientifica relativa al cloud computing. In particolare nel prossimo capitolo verrà illustrata la metodologia utilizzata per condurre l’analisi bibliografica. Il capitolo 3 presenterà i risultati di suddetta analisi, mentre nel capitolo 4 verranno presentati in maniera più approfondita i sistemi di brokeraggio.

## 2. Metodologia

L’analisi della letteratura ha seguito un processo strutturato che è iniziato con l’interrogazione dei principali motori di ricerca, tra i quali:

- Ebsco
- Emerald
- Ingenta
- Sage
- ScienceDirent
- Springerlink
- Taylor & Francis

In ciascun motore di ricerca è stata svolta la “ricerca avanzata”, inserendo alcune parole chiave: “cloud”, “computing”, “simulation”, “optimization”, “elastic”, “infrastructure”.

Nella ricerca avanzata è stato possibile anche indicare i campi specifici nei quali cercare le key words. La prassi è stata di limitarsi al così detto “All except full text” che sintetizza in generale la ricerca nei campi del titolo, dell’abstract e delle keywords, escludendo il testo, campo troppo vasto per ottenere utili risultati.

Per la ricerca sono state svolte nello specifico 5 diverse query, combinazione delle keywords prescelte. Le query in questione sono:

- cloud computing simulation
- cloud computing optimization
- cloud elastic simulation
- cloud infrastructure optimization
- cloud infrastructure simulation.

Per ciascun motore di ricerca sono stati estratti tutti i dettagli (titolo, autori, journal, anno pubblicazione, abstract, etc...), realizzando un unico documento. Sono stati ottenuti 1141 risultati, al netto dei duplicati (299).

A seguito di una prima lettura del titolo e in alcuni casi dell’abstract, si è proseguito ad un’analisi più approfondita dei risultati, per selezionare le pubblicazioni di nostro interesse.

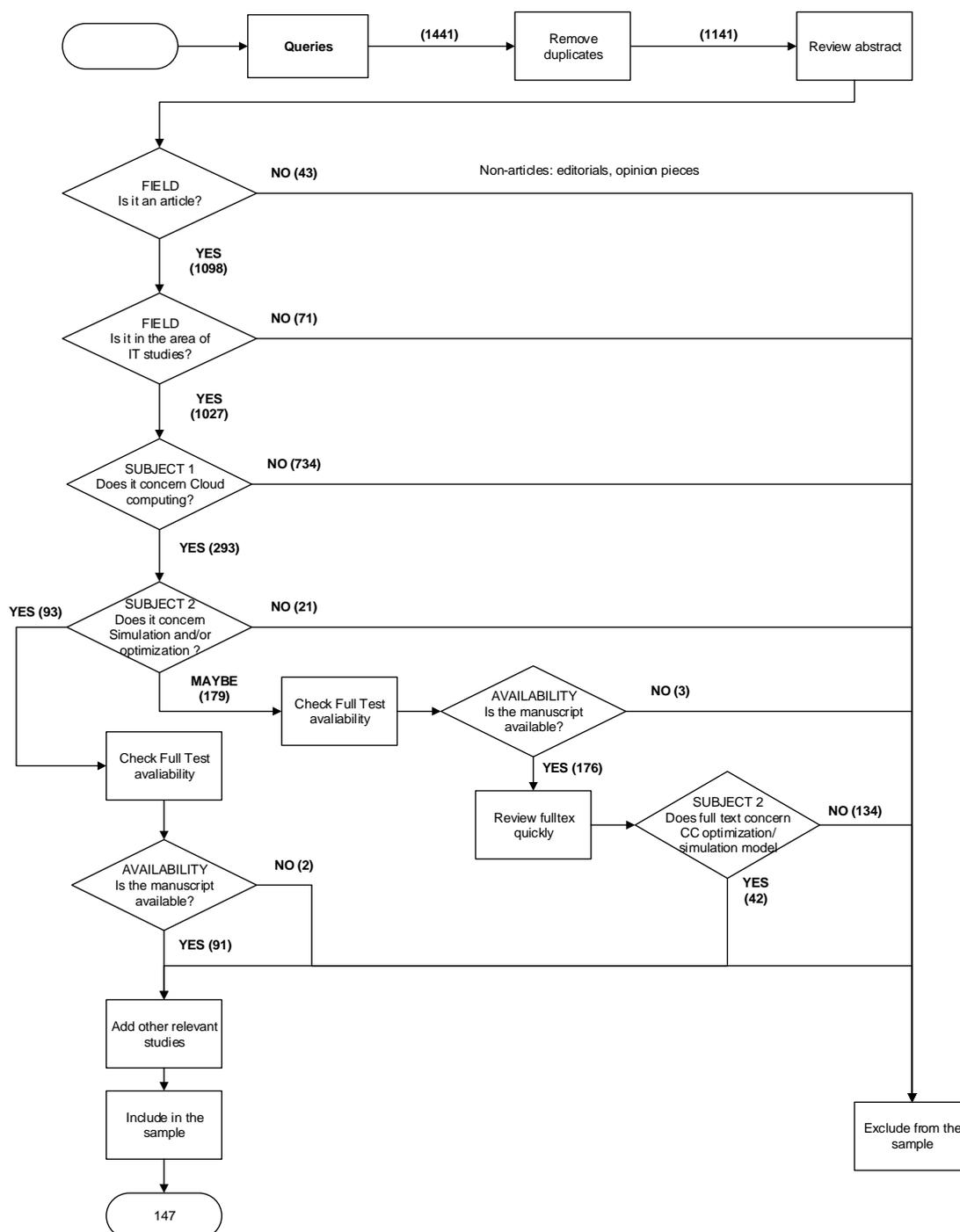
La nostra ricerca si è concentrare sui soli paper, scartando i reference work, books, editorial e altri tipi di pubblicazioni. Ai fini di questa scelta sono state filtrate a monte le pubblicazioni di nostro interesse, inserendo nella query di ricerca il filtro “only paper”.



A questo punto la ricerca si è concentrata sui soli articoli del settore d’interesse, ad esempio escludendo le pubblicazioni in campo fisico e metereologico, che a causa del termine “cloud” risultavano in numero consistente.

Infine sono stati selezionati i soli articoli che riguardassero l’argomento approfondito, studio e sviluppo di algoritmi e modelli di ottimizzazione nella gestione del Cloud Computing. Inoltre, nel caso in cui il titolo e l’abstract non consentisse di selezionare il paper, sono state condotte delle analisi nello specifico, attraverso la lettura dell’intera pubblicazione, ottenendo così infine 133 articoli.

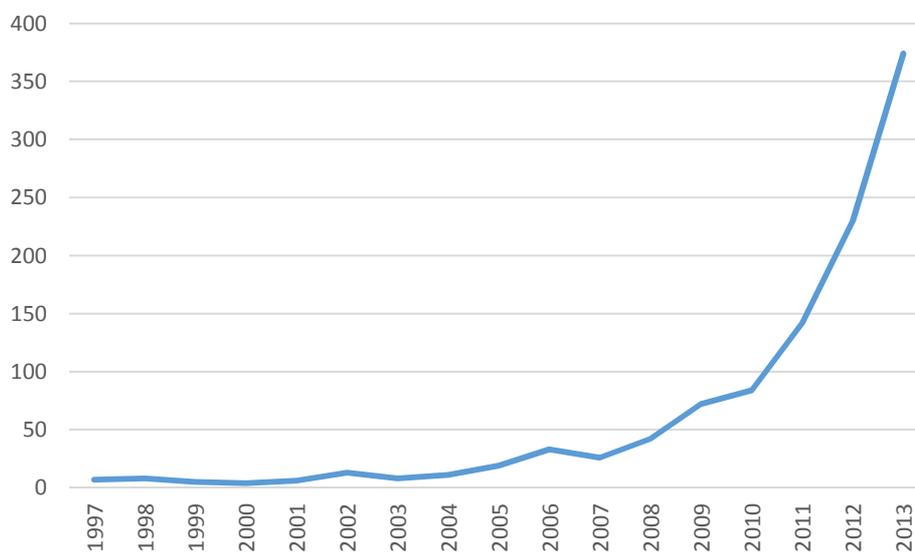
Nella figura che segue è stato riportato il flow chart di tutte le fasi di ricerca per la classificazione degli articoli scientifici descritti in questo capitolo.





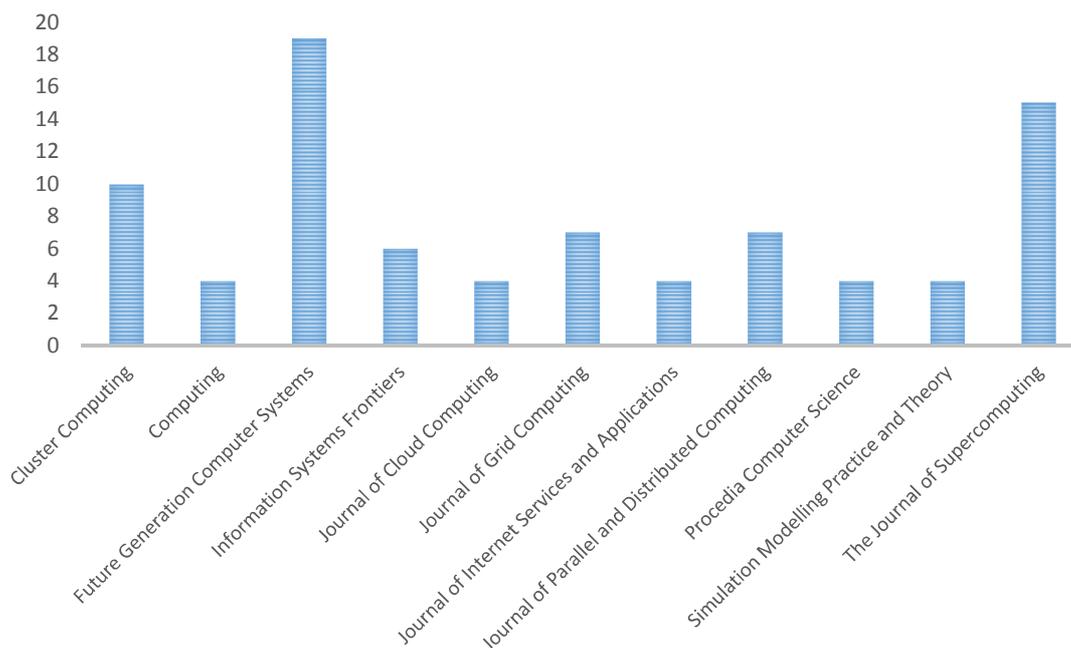
**Figura 1 – Algoritmo di analisi**

Prima di presentare lo schema di classificazione sono state riportate delle statistiche sui risultati ottenuti. Ad esempio la Figura 2 mostra un grafico (riferito ai paper selezionati) con il trend di pubblicazione che conferma quanto ipotizzato in ambito di studio di fattibilità del progetto in merito al crescente interesse della comunità scientifica per l’argomento della ricerca.



**Figura 2 – Trend di pubblicazione dei paper in ambito CC**

Infine sono stati riportati i principali Journal di riferimento (Figura 3). Il criterio di selezione si basa sul numero minimo, quattro paper, afferenti al journal selezionato, che riguardano lo studio e sviluppo di algoritmi e modelli di ottimizzazione nella gestione del Cloud Computing.



**Figura 3 – Principali Journal di riferimento**



## 3. Stato dell'arte

### 3.1 Schema di classificazione

Seguendo il processo descritto nel precedente capitolo, i paper sono stati analizzati usando sette dimensioni di classificazione mutuata, in parte, dalla letteratura. A ciascuna dimensione è stata dedicata uno specifico paragrafo.

- 1) *Tipologie di implementazione*, si distinguono le tre principali tipologie di implementazione del cloud computing: cloud privati, cloud pubblici e cloud ibridi.
- 2) *Metodi di analisi e tecniche risolutive*. La prima parte tratterà dei metodi di analisi utilizzati nei contributi, come problemi di ottimizzazione combinatoria, euristiche, analisi di benchmark e analisi di scenario. La seconda parte approfondisce la natura delle tecniche risolutive adottate, facendo distinzione tra programmazione matematica, simulazione e algoritmi euristici.
- 3) *Funzione obiettivo e risorse considerate*. In questa sezione i lavori che presentano modelli di ottimizzazione del cloud sono classificati in relazione alla funzione obiettivo ed al tipo di risorse considerate.
- 4) *Modalità della validazione*. In questa sezione verrà illustrata la modalità attraverso la quale i modelli presentati sono stati validati.
- 5) *Cloud commerciali*. Questa sezione presenta e descrive i principali cloud commerciali presenti sul mercato.
- 6) *Software per la simulazione*. In questo paragrafo sono raggruppati i principali software dedicati alla simulazione in ambito cloud computing.
- 7) *Cloud broker*. Infine l'ultimo criterio riguarda i sistemi di brokering adottati.

I criteri 1, 3, 5, 6 sono mutuati da Sakellari et al. (2013) [107], mentre 2, 4 da Cardoen et al. (2010). Nei prossimi paragrafi, per posizionare ciascun contributo all'interno delle matrici di classificazione proposte, a ciascun articolo è stato assegnato un numero identificativo (vedi Bibliografia).

#### 3.1.1 Tipologie di implementazione

Come sottolineato da Qi Zhang et al. (2010) [146] e ripreso nelle review presentate in [8, 15, 35, 37, 51, 55, 65, 127, 134, 144, 146] la letteratura distingue tre tipologie fondamentali di cloud: privati, pubblici e ibridi.

Nel caso di cloud privati, tutti i servizi sono implementati in un data centre privato e utilizzati esclusivamente da una singola organizzazione. In questi cloud privati possono essere implementate tecnologie proprietarie, potendo scegliere il grado di accessibilità ad utenti interni o esterni al cloud stesso. Infatti un cloud privato offre il più alto grado di controllo sulle prestazioni, affidabilità e sicurezza rispetto a gli altri modelli presenti sul mercato.

I cloud pubblici sono forniti da terze parti. La differenza fondamentale tra privato e pubblico è che i cloud pubblici sono più competitivi, offrendo in genere una gamma più ampia di servizi, con prezzi più bassi. Inoltre i cloud pubblici forniscono diversi vantaggi chiave, tra cui nessun investimento iniziale nelle infrastrutture. Tuttavia, i cloud pubblici mancano di controllo granulare sui dati in rete e riguardo le impostazioni di sicurezza. Questi fattori ostacolano la loro efficacia in molti scenari di business.

Molte aziende, piuttosto che investire su un modello rispetto che un altro, utilizzano una terza strategia di implementazione nota come modello ibrido. Questo modello nasce dall'esigenza di superare i limiti di entrambe le configurazioni private e pubbliche. Cloud ibridi si concentrano principalmente su data center di proprietà, ma si appoggiano su risorse cloud pubbliche, per fornire



supporto alla potenza di calcolo e lo storage, elementi necessari per la protezione da picchi imprevisti di domanda. D’altra parte aspetti negativi si riscontrano in merito alla progettazione della struttura. Sarà necessario determinare con cura la migliore suddivisione tra i componenti che afferiscono alla componente pubblica e a quella privata.

Nella Tabella 1 riportiamo come i contributi sono classificati in base al diverso modello di implementazione del cloud.

TIPOLOGIE DI IMPLEMENTAZIONE	
Privato	[29, 31, 40, 46, 75, 83, 89, 116]
Pubblico	[4, 6, 7, 17, 30, 38, 45, 49, 79, 89, 100, 102, 121, 122, 126]
Ibrido	[1, 5, 9, 11, 13, 19, 32, 34, 58, 82, 84, 85, 88, 89, 92, 95, 98, 120, 128, 136]

Tabella 1 – Tipologie di implementazione

### 3.1.2 Metodi di analisi e tecniche risolutive

Nella Tabella 2 sono analizzati i metodi di analisi in base a:

- 1) problemi di ottimizzazione combinatoria tramite metodologie esatte o euristiche;
- 2) analisi di benchmark;
- 3) analisi di scenario;
- 4) analisi di complessità.

Per la risoluzione dei problemi (vedi Tabella 3) sono state adottate tecniche come:

- 1) programmazione matematica;
- 2) simulazione (a eventi discreti o Monte Carlo);
- 3) algoritmi euristici.

METODI DI ANALISI	
Ottimizzazione	
Esatta	[1, 6, 7, 9, 13, 18, 20, 21, 23, 24, 26, 27, 30, 32, 33, 39, 40, 43, 48, 50, 52, 58, 60-63, 66, 68, 69, 72-74, 76, 79, 82, 83, 86-88, 90-93, 95, 99-102, 105, 110, 111, 114, 117, 118, 120, 126, 128-130, 138, 139, 141, 145, 147]
Euristica	[2, 3, 42, 53, 56, 70, 75, 84, 85, 109, 112, 113, 116, 136]
Benchmark	[8, 11, 12, 15, 25, 28, 35, 37, 41, 47, 49, 51, 54, 55, 59, 65, 71, 78, 80, 89, 94, 96, 103, 107, 108, 122-125, 132-134, 142-144, 146]
Analisi di scenario	[1-3, 5-7, 9, 13, 16, 17, 19, 23, 24, 26, 31-34, 39, 48, 50, 52, 53, 56, 58, 62, 63, 67, 70, 79, 81, 92, 93, 95, 99, 101, 102, 110, 112, 114, 116, 117, 130, 136, 138, 139, 145, 147]

Tabella 2 – Tipologia di analisi



TECNICHE RISOLUTIVE	
Programmazione matematica	[1, 7, 18, 20, 21, 30, 43, 48, 60, 61, 66, 72-74, 81, 87, 91-93, 95, 99, 101, 102, 110, 114, 117, 118, 120, 130, 147]
Simulazione	
A eventi discreti	[5, 9, 16, 27, 29, 33, 40, 44, 50, 57, 58, 62, 63, 67, 69, 77, 82, 86, 98, 104, 121, 128, 129, 135, 138, 139, 141]
Monte Carlo	[45, 100]
Euristiche	[2, 3, 42, 53, 56, 70, 75, 84, 85, 87, 88, 112, 113, 116, 119, 136]

Tabella 3 – Tecniche di risoluzione

### 3.1.3 Funzioni obiettivo e risorse considerate

In questa sezione i lavori che presentano modelli di ottimizzazione del cloud, verranno riclassificati in relazione alla funzione obiettivo ed al tipo di risorse considerate. Per quanto riguarda le funzioni obiettivo, in prima istanza è possibile ravvisare come queste considerino in maniera preminente due tipologie di prestazioni: efficienza energetica e QoS.

In particolare con efficienza energetica si indica la capacità di sfruttare l'energia per soddisfarne un determinato fabbisogno. Minori sono i consumi relativi al soddisfacimento di suddetto fabbisogno, migliore sarà l'efficienza energetica del sistema.

Con QoS si individua in generale l'ottimizzazione delle performance del sistema, esprimibili in termini di qualità del servizio fornito all'utente. Quando è previsto un contratto di servizio con relativi parametri di QoS (con relative penali nel caso questi parametri non vengano rispettati) si parla anche di Service Level Agreement (SLA).

Nella Tabella 4 si riporta come i diversi articoli sono classificati in base al diverso obiettivo dell'ottimizzazione condotta.

OBIETTIVO DELL'OTTIMIZZAZIONE	
Efficienza energetica	[1-3, 17, 24, 25, 60, 61, 63, 64, 70, 96, 105, 118, 130, 137]
Performance/QoS	[4-16, 18-23, 26-59, 62, 65-68, 70-95, 97-104, 106-117, 119-129, 131-136, 138-147]

Tabella 4 – Obiettivo dell'ottimizzazione matematica

I modelli presenti in letteratura considerano un insieme limitato delle risorse che caratterizzano un sistema cloud. Tra queste le risorse maggiormente considerate sono:

- CPU (CORE, computing power);
- Storage;
- RAM (memory);



- Rete (IP, bandwidth);

Nella tabella che segue (Tabella 5) riportiamo i contributi distinti per risorsa.

RISORSE	EFFICIENZA ENERGETICA	PERFORMANCE/QoS
CPU	[24, 25, 63, 69, 126]	[1, 6, 9, 10, 16, 23, 34, 50, 52, 62, 66, 73, 74, 98, 104, 110, 111, 117, 136]
Storage	[64]	[16, 49, 80, 125, 126, 136, 140]
RAM	[17, 60, 61, 76, 96, 105, 118, 130, 137]	[5, 7, 11, 13, 18-21, 26, 29-32, 40, 42, 45, 46, 48, 53, 56, 58, 75, 85, 88, 94, 95, 99, 101, 102, 104, 106, 112, 114, 116, 126, 131, 139, 141, 145, 147]
Server		[93]

Tabella 5 – Risorse oggetto di ottimizzazione

Il documento *ICARO\_Deliverable3\_27.1\_Funzione multi obiettivo* esamina questa sezione, analizzando con un grado di dettaglio maggiore le funzioni obiettivo trattate e le risorse critiche che caratterizzano i sistemi cloud oggetto maggiormente considerate.

### 3.1.4 Modalità della validazione

In Tabella 6 i contributi sono suddivisi a seconda delle modalità di validazione dei modelli in essi presentati. In particolare si distingue tra modelli:

1. Non validati
2. Validati su dati realistici, ma non reali;
3. Validati su dati reali, provenienti in genere dai sistemi informativi aziendali.

MODALITA' DI VALIDAZIONE	
Nessun test	[8, 12, 15, 22, 25, 35, 37, 44, 49, 51, 55, 59, 64, 65, 71, 72, 78, 89, 93, 94, 104, 107, 108, 119, 122-124, 127, 132-134, 142-144, 146]
Dati per i test	
Sulla base di dati reali	[2-7, 9-11, 16, 19, 20, 28, 32, 36, 41, 45, 47, 54, 73, 80, 92, 116, 136]
Dati teorici	[1, 13, 14, 17, 18, 21, 23, 24, 26, 27, 29-31, 34, 38-40, 42, 43, 46, 48, 50, 52, 53, 56-58, 60-63, 66-70, 74-79, 82-85, 87, 88, 90, 91, 95, 96, 98-103, 105, 106, 109-114, 117, 118, 120, 121, 125, 126, 128-131, 135, 137-141, 145, 147]

Tabella 6 – Modalità di validazione



### 3.1.5 Cloud commerciali

Per sviluppare un ambiente cloud per scopi sperimentali è necessario l'accesso ad una infrastruttura hardware e ad un software framework per gestirlo. La letteratura riporta molte soluzioni che utilizzano servizi cloud commerciali.

Il servizio cloud commerciale utilizzato maggiormente in ambito di ricerca è Amazon Elastic compute Cloud (EC2), che fornisce servizi IaaS (Infrastructure as a Service) su larga scala. Consente agli utenti di aumentare o ridurre il numero di VM necessarie e di modulare il carico in rapporto al numero delle istanze e/o la capacità richiesta. Infine essendo storicamente uno dei servizi cloud maggiormente significativi, una serie di software frameworks garantiscono la compatibilità con EC2 per lo sviluppo di cloud privati.

La piattaforma Microsoft Windows Azure è composta da tre componenti e ognuno di loro fornisce all'utente finale un insieme specifico di servizi:

1. Windows Azure fornisce un ambiente basato su Windows per l'esecuzione di applicazioni e la memorizzazione dei dati su server in data center.
2. SQL Azure fornisce servizi di dati nel cloud sulla base di SQL Server.
3. .NET Services offrono servizi di infrastruttura distribuiti per il cloud based e applicazioni locali.

La piattaforma Windows Azure può essere utilizzato sia da applicazioni in esecuzione nel cloud, sia da applicazioni in esecuzione sui sistemi locali.

Infine Google App Engine è una piattaforma per applicazioni web tradizionali afferenti a data center proprietari di Google.

La Tabella 7 riporta le principali caratteristiche dei prodotti commerciali Zang et al. (2010) [146], mentre la Tabella 8, riporta in quali articoli tali prodotti siano stati l'oggetto di studio e ricerca.

CARATTERISTICHE PRINCIPALI PRODOTTI COMMERCIALI			
Cloud Provider	Amazon EC2	Windows Azure	Google App Engine
Classes of Utility Computing	Infrastructure service	Platform service	Platform service
Target Applications	General-purpose applications	General-purpose Windows applications	Traditional web applications with supported framework
Computation	OS Level on a Xen Virtual Machine	Microsoft Common Language Runtime (CLR) VM; Predefined roles of app. instances	Predefined web application frameworks
Storage	Elastic Block Store; Amazon Simple Storage Service (S3); Amazon SimpleDB	Azure storage service and SQL Data Services	BigTable and MegaStore
Auto Scaling	Automatically changing the number of instances based on parameters that users specify	Automatic scaling based on application roles and a configuration file specified by users	Automatic Scaling which is transparent to users



Tabella 7 – Caratteristiche principali prodotti commerciali adattato da Zang 2010 [146]

PRINCIPALI PRODOTTI COMMERCIALI	
Amazon EC2	[2-7, 9, 11, 16-20, 28, 30, 32, 34, 38, 45, 48, 50, 57, 58, 84, 85, 92, 95, 98, 99, 106, 121, 122, 128, 136]
Windows Azure	[49, 57, 100]
Google App Engine	[28, 29, 57, 128]

Tabella 8 – Principali prodotti commerciali

### 3.1.6 Software per la simulazione cloud

Al momento esistono ancora poche soluzioni per la simulazione di architetture cloud, forse perché la virtualizzazione ha permesso l'implementazione in virtual private cloud su banche prova di piccola scala. Tuttavia, ci sono state alcune proposte importanti in questo ambito.

Ad esempio, CloudSim simulation framework ha dimostrato di essere in grado di istanziare 100.000 macchine in meno di 5 minuti, richiedendo solo 75 MB di RAM. CloudSim si basa su SimJava, simulatore ad eventi discreti, che è collocato ad un livello più basso, mentre gli strati superiori implementano il toolkit GridSim al fine della modellazione del cluster, comprese anche le reti, i profili di traffico e le risorse.

CloudSim estende efficacemente le funzionalità principali del GridSim, dalla modellazione di storage, application services, resource provisioning tra VM e data center di intermediazione. CloudSim è stato modificato e ampliato da diversi gruppi di ricerca, come dimostrano le molteplici pubblicazioni presenti in letteratura.

Un'altra soluzione, che si concentra in particolare sulla misurazione del consumo energetico è GreenCloud. I modelli di potenza utilizzati per stimare il consumo energetico, assumono la proporzionalità della potenza assorbita dal carico della CPU nel server e dimostrano come il consumo di energia degli switch è quasi costante e proporzionale alla velocità di trasmissione solo in un rapporto molto piccolo.

Un altro simulatore è MDCSim, che è dedicato in particolare ai data center multi-livello. Con questo strumento si può analizzare un data center cluster-based con l'implementazione dettagliata di ogni singolo livello. E' stato configurato in tre strati, tra cui un livello di comunicazione, un livello kernel e un strato a livello utente, per modellare i diversi aspetti di un cloud. Inoltre è in grado di stimare il throughput, i tempi di risposta e il consumo energetico.

Infine, il più recente, è il simulatore iCanCloud. Si basa su SIMCAN, che è un framework software di simulazione, dedicato a reti per l'archiviazione con grandi dimensioni. iCanCloud può prevedere il trade-off tra costi e prestazioni di una particolare applicazione con un hardware specifico, al fine di mostrare agli utenti i costi. Il team di sviluppo, in particolare, si è concentrato su politiche simili a quelle di Amazon del tipo pay-as-you-go.

iCanCloud ha un'interfaccia grafica completa per mezzo della quale gli esperimenti possono essere progettati ed eseguiti, anche se gli attuali sistemi software possono essere modellati solo manualmente. In fine è permessa l'esecuzione in parallelo di esperimenti su più macchine virtuali.



La Tabella 9, proposta da [98], riporta una panoramica sulle funzionalità dei suddetti software, mostrando le caratteristiche principali e i punti di forza/debolezza di ciascun applicativo.

PARAMETER	CloudSim	MDCSim	GreenCloud	iCanCloud
<i>Platform</i>	-	CSIM	NS2	OMNET, MPI
<i>Language</i>	Java	C++/Java	C++/OTcl	C++
<i>Availability</i>	Open source	Commercial	Open source	Open source
<i>Graphical support</i>	Limited (though CloudAnalyst)	None	Limited (though Nam)	Full
<i>Communicatio model</i>	Limited	Limited	Full	Full
<i>Models for public cloud providers</i>	None	None	None	Amazon
<i>Support for parallel experiments</i>	No	No	No	Wip
<i>Support for power consumption modeling</i>	Limited	Yes	Yes	Wip

Tabella 9 – Software di simulazione per il cloud (fonte: Núñez 2012 [98])

La Tabella 10 riporta come i diversi articoli sono classificati in base al diverso software di simulazione.

SOFTWARE DI SIMULAZIONE PER IL CLOUD	
CloudSim	[9, 16, 27, 29, 33, 40, 44, 57, 58, 69, 86, 104, 128, 129, 135, 138, 139, 141]
GreenCloud	[62, 63]
MDCSim	[77]
iCanCloud	[98]

Tabella 10 – Software di simulazione per il cloud

### 3.1.7 Processo di brokeraggio

Questa sezione presenta i contributi in base alla working definition del NIST (National Institute of Standards and Technology), che inquadra tre servizi principali svolti da un cloud broker e sono: servizio di intermediazione, di aggregazione e di arbitraggio.

Un approfondimento su tale definizione segue nel Capitolo 4.



SERVIZI CLOUD BROKER	
Intermediazione	[32,47]
Aggregazione	[31, 83]
Arbitraggio	[8, 15, 32, 47, 51, 83, 106]

Tabella 11 – Servizi cloud broker

## 4. I sistemi di brokeraggio

Questo capitolo riporta un approfondimento dei risultati relativi ai sistemi di brokeraggio.

In economia e nel diritto il broker è un soggetto intermediario. Può essere una persona (fisica o giuridica) o un gruppo di persone (agenzia) che organizza delle transazioni tra un buyer (acquirente) e un seller (venditore). Il suo guadagno risiede in una commissione sulla transazione.

La definizione di Rogers et al. 2012 [106], si avvicina molto a quella classica, dove il broker corrisponde ad un vero e proprio intermediario, una terza parte “a third-party intermediary...the Coordinator”. Inoltre riportiamo come si svolge il lavoro: “A broker makes a profit by matching buyer’s demands with seller’s supplies: the broker uses a variety of methods to achieve a best price between these parties, and typically makes a profit either by taking a commission fee from any completed deal, or by varying the broker’s *spread*, or some combination of fees and spread”.

Una definizione che si accosta a quella classica viene riportata da Höfer et al. 2011 [51]. Essi descrivono come sia attivo nel network dei cloud un sistema di brokering che si occupa di uno “scambio di cloud”. Su ciascuno di essi è presente un “Cloud Coordinator which publishes offers based on the service the cloud offers. The Cloud Brokers requests the required capacity and negotiates or bids on the offers.”

Cuomo et al. 2012 [31] presentano uno studio per la progettazione di un cloud provider, Cloud@Home abbreviato C@H) con servizi IaaS che utilizzano strumenti di computing, storage e rilevatori di risorse, anche acquisiti da collaboratori volontari. L’assunzione di base su cui C@H si basa è che le risorse offerte, su base volontaria, non sono affidabili e non possono fornire livelli di QoS paragonabili a quelli offerti dai cloud commerciali.

Anche in questo contesto il concetto di broker è di notevole rilevanza e viene descritto come un intermediario, ma con un’eccezione più orientata all’acquisizione di risorse: “*from a high level point of view, a C@H provider is an intermediary (or broker) for the acquisition of resources from different Resource Owners. In this way, it delegates the low-level management of the infrastructure to interface with such Resource Owners. Two important tasks of a C@H provider are therefore the search for Resource Owners (providers) and the acquisition of resources that eventually will be delivered to the final C@H Users.*”

Simile definizione viene fornita da D’Agostino 2012 [32], che pone maggiore attenzione al rispetto degli SLA, ad una gestione consapevole delle risorse e al QoS stabilito, “*at the core of the developed hybrid Cloud stays a QoS-aware broker able to dynamically changing the infrastructural platform, thus to fulfill the actual requirements according to the agreed SLAs.*” A tal fine il broker “*will transparently manage the allocation of the requested service to the public or to the private Cloud infrastructures, depending on the QoS expectations and the workload of the private Cloud resources, in order to try satisfying the highest number of user requests and maximizing the profit of the private provider.*”



Broberg 2007 [15] e Arutyunov 2012 [8] descrivono come il broker siano uno o più soggetti che operano in un *“Grid systems, acting as middlemen between end-users and service providers.”* Questi broker possono eseguire una serie di importanti funzioni come *“aggregating resources from many providers, negotiating and enforcing quality of service targets, and reducing the complexity of access for end-users.”*

Riportiamo anche la definizione di Habib et al. 2012 [47] i quali riportando come il broker abbia due ruoli distinti, infatti *“Firstly, there are brokers that concentrate on negotiating relationships between consumers and providers without owning or managing the Cloud infrastructure. They provide, for example, consultancy services to the potential Cloud Consumers for moving their IT resources into a suitable cloud. Secondly, there are brokers that add extra services on top of a Cloud Providers’ infrastructure / platform/ software to enhance and secure the Cloud environment for the consumers. For example, a broker might offer identity and access management service on top of Cloud Provider’s basic service offerings to consumers”.*

In entrambi i casi il broker agisce come un *Cloud Provider* che fornisce ai consumatori valore aggiunto o bundle di servizi.

Riportiamo in Figura 4 anche la panoramica dell’architettura del cloud broker proposta da Lucas-Simarro et al. 2012 [83]. Al centro della rappresentazione troviamo il DataBase, composto da tre principali componenti *“the Cloud manager, which collects information from cloud providers; the Scheduler, which reads the user description file, invokes the selected scheduling strategy, and makes the placement decision; and the VM manager, which performs the deployment action”.*

Il quadro che emerge dalle definizioni riportate fino ad ora, riflette un interessante aspetto sull’importate ruolo che i cloud brokers svolgono. Infine riportiamo per completezza la definizione del NIST (National Institute of Standards and Technology), che riassume tutte le precedenti, un cloud broker è:

*“An entity that manages the use, performance, and delivery of cloud services, and negotiates relationships between Cloud Providers and Cloud Consumers.”*

Continuando con la definizione del NIST, un cloud broker può fornire tre diverse categorie di servizi, che possono essere erogati da un distinto componente oppure da un insieme di entità presenti del cloud environment:

- i. *“Service intermediation”*: estende un servizio cloud fornendo servizi a valore aggiunto ai cloud consumer, per esempio la gestione dell’accesso, dell’identità o della sicurezza.
- ii. *“Service aggregation”*: combina e integra servizi diversi in un servizio nuovo, assicurando l’integrazione e la sicurezza dei dati trasferiti tra il cloud consumer e i differenti cloud provider.
- iii. *“Service arbitrage”*: sceglie i servizi cloud da fornitori diversi in modo flessibile e dinamico facendo ricorso a criteri di economicità o di disponibilità.

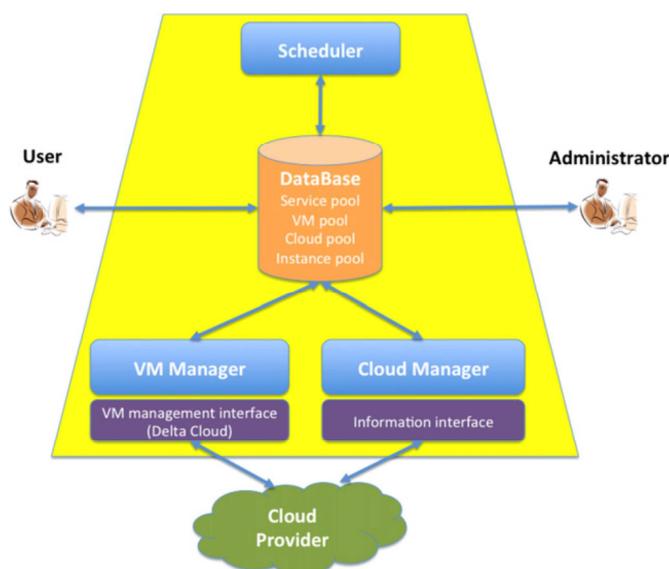


Figura 4 – Architettura del Cloud broker (fonte: Lucas-Simarro et al. 2012 [83])

#### 4.1 Le funzionalità di predizione nel broker

In generale gli strumenti di predizione come algoritmi e/o funzioni matematiche, svolgono un ruolo molto importante nel migliorare la stabilità e le prestazioni delle decisioni di allocazione delle risorse. In questo contesto lo specifico compito di predizione è stato assegnato al sistema di brokeraggio, che per svolgere le funzionalità di intermediazione, necessita anche di predire il carico delle risorse (es. CPU, storage, etc.), dell'host e dei workflow durante le loro attività. Sono riportati di seguito i modelli e gli strumenti utilizzati nei contributi risultato della ricerca.

Resta difficile indicare, al momento, quale sia il miglior modo per integrare le funzionalità di predizione nel processo di brokeraggio. Ciò detto è stato possibile prendere spunto da alcuni paper che affrontano l'argomento della predizione in condizioni critiche.

Nel contributo [151] il carico è stato previsto tramite la combinazione di due algoritmi, Phase Space Reconstruction (PSR) e Group Method of Data Handling (EA-GMDH). Il PSR è la teoria di base per l'analisi dei sistemi dinamici caotici, elaborato da F. Takens. Il EA-GMDH, evoluzione del GMDH consiste nella possibilità di trovare interrelazioni automaticamente nei dati, per selezionare una struttura ottimale di reti neurali, aumentandone la precisione. In particolare l'evoluzione ha previsto un maggior numero di variabili di input e un diverso ordine di polinomiale.

Come riportato in Figura 5, il metodo proposto contiene due fasi principali. In particolare nella prima il metodo PSR è stato utilizzato per ricostruire la traccia di carico da una serie temporale unidimensionale. Nella seconda la ricostruzione della serie temporale è utilizzata come input al EA-GMDH. Le performance del metodo proposto sono state valutate utilizzando due valori di carico derivanti da dati reali. I primi sono dati di carico in un sistema tradizionale, mentre i secondi derivano da un data center di Google. I risultati mostrano che il metodo proposto ottiene prestazioni al di sopra di metodi classici. In Figura 5 riportiamo l'architettura del modello proposto.

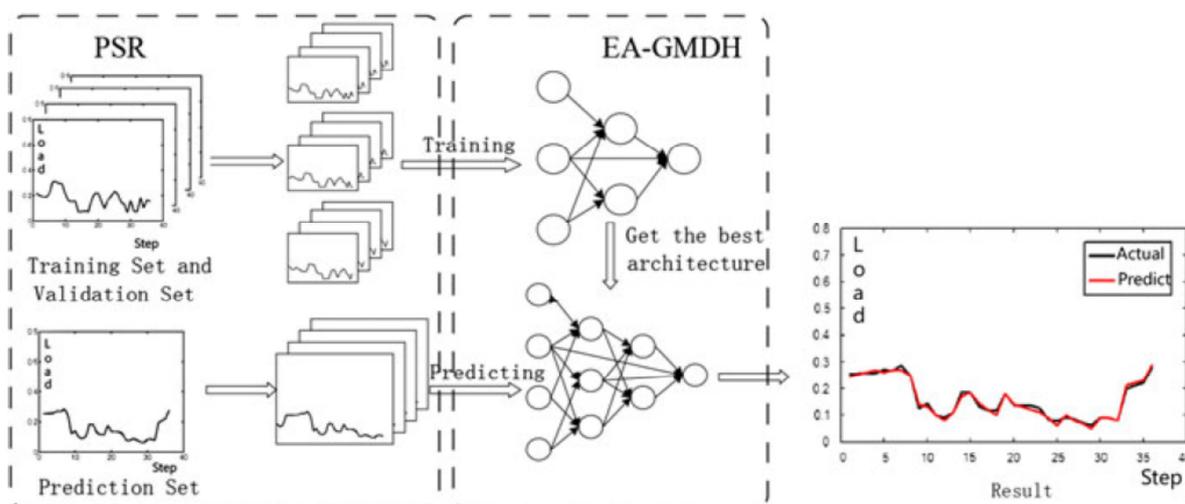


Figura 5 – L’architettura del modello (fonte [151])

Nel secondo contributo [141] è stato utilizzato un modello più semplice. Poiché i carichi di lavoro di un sistema di servizi cloud sono irregolari, è necessario un metodo in grado di regolare rapidamente il modello a seconda della variazione della loro tendenza. Inoltre si osserva che il carico di lavoro è lineare ma in un periodo relativamente breve di tempo. I risultati dell’analisi hanno mostrato che per risolvere questo tipo di problema sia efficace il modello di regressione lineare (Linear Regression Model).

Il modello ha la finalità di scalare il costo delle risorse virtuali, come ad esempio la CPU, la memoria e le altre risorse utilizzate. Nello specifico, sulla base di questo carico di lavoro (previsto), è stato proposto un meccanismo di autoscaling delle risorse virtuali. Tale meccanismo di autoscaling combina il ridimensionamento in tempo reale (real-time scaling) e il ridimensionamento in pre-scaling.

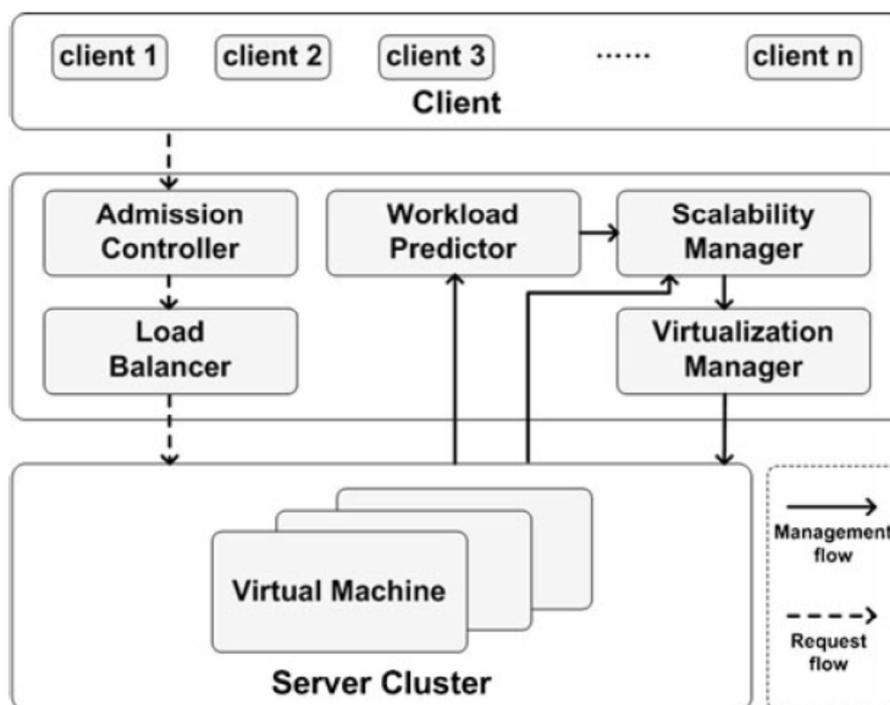


Figura 6 – L’architettura di cloud scaling (fonte [141])



In particolare il pre-scaling inizia con la raccolta di informazioni dal predittore e propone una strategia di pre-scaling. Successivamente il real-time scaling agisce in quanto la previsione del carico di lavoro, a causa di variazioni (anormali) risulta essere inferiore a quella effettiva e le risorse stanziare risultano non essere sufficienti per gestire le richieste. Se tale problema non viene gestito su tutta la durata del processo, la prestazione nel tempo risulterà influenzata in negativo.

Infine, sono stati condotti dei test per dimostrare che l'approccio presentato può soddisfare i Service Level Agreement (SLA) stabiliti con gli utenti, mantenendo bassi i costi.

Come mostra la Figura 7, nella maggior parte delle situazioni, la bontà del modello di regressione lineare supera gli altri metodi, inoltre il LRM ha una bassa deviazione. Tale metodo può regolare la propria funzione in base all'andamento del carico di lavoro, in modo che possa produrre valori di carico puntuali e precisi. Viceversa, la previsione ARMA (Autoregressive Moving Average Method) è sempre in ritardo, la previsione media è meno sensibile delle variazioni del carico di lavoro e la previsione max è maggiore del valore effettivo per la maggior parte del tempo. Segue in Figura 7, un grafico relativo al carico di lavoro attuale e predetto.

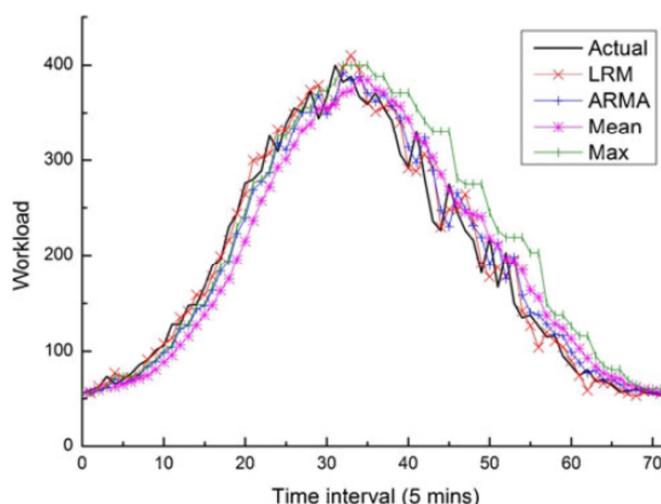


Figura 7 – Carico di lavoro attuale e predetto (fonte [141])

Sempre in linea con questo in [150] sono stati utilizzati degli strumenti di predizione e forecast come la EWMA (Exponentially Weighted Moving Average) e il FUSD (Fast Up and Slow Down).

Nella figura che segue è stata riportata la previsione del carico di lavoro di una CPU.  $W$  è la finestra di misurazione.

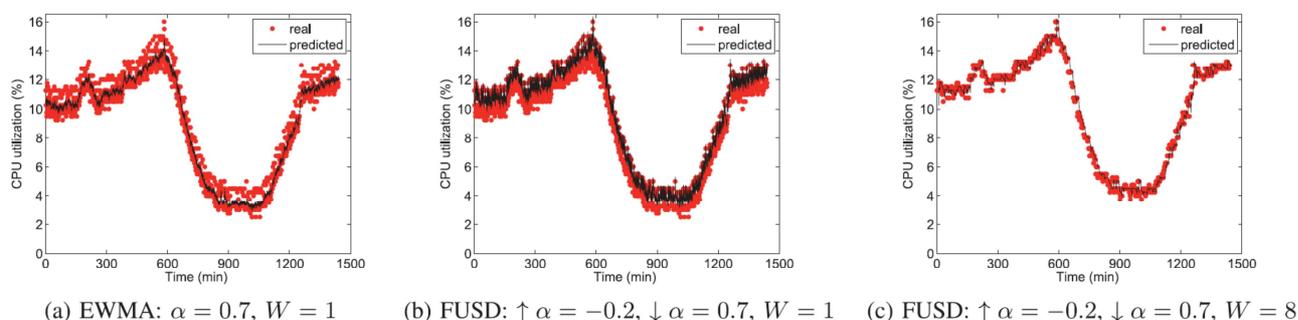


Figura 8 – Previsione del carico di lavoro di una CPU (fonte [150])



La misurazione è stata effettuata misurando il carico della CPU ogni minuto per prevederlo nel minuto successivo. Nel primo grafico (a) notiamo come l'utilizzo della EWMA graficamente attraverso la curva a metà dei punti. Questo indica una previsione abbastanza precisa.

Nei grafici successivi (b) e (c) cambiando strumento di predizione otteniamo risultati superiori a quelli osservati nella maggior parte del tempo. Nel grafico (b) ricaviamo valori del 77%. Mentre il grafico (c) mostra un risultato di previsione migliore, andando a variare la finestra di misurazione ( $W=8$ ) e prendendo il 90% di picco della domanda.

Sul piano teorico riteniamo che entrambi i metodi presentati in [141, 150] siano in grado di soddisfare le nostre esigenze di analisi delle serie storiche del carico delle risorse.

Un altro aspetto di possibile approfondimento riguarda la dinamica con la quale un utente crea un'istanza di una nuova VM. Di solito, i prezzi per un utilizzo a breve termine dei diversi tipi di istanze sono fissati dai cloud provider, oppure, se richiede l'utilizzo di risorse nel medio periodo, l'utente può stipulare un contratto nel lungo termine ottenendo un prezzo inferiore. Nel prima modalità, il meccanismo di assegnamento dei prezzi si compone di tre fasi:

1. Il meccanismo raccoglie le richieste da parte degli utenti.
2. Il meccanismo calcola l'assegnazione e il prezzo in base alle richieste raccolte e la disponibilità di risorse.
3. Gli utenti che ottengono l'assegnazione pagano il cloud provider per ottenere accesso alle risorse richieste.

I meccanismi proposti in Zaman et al. (2012) [145] trovano una possibile soluzione a questo tipo di problema. Nel contributo propongono delle estensioni di due meccanismi ad asta combinatori, il Linear Programming-Based Mechanism (CA-LP) e il Greedy-Based Mechanism (CA-GREEDY). Inoltre sono stati svolti degli esperimenti simulativi per confrontare i due meccanismi combinatori con un meccanismo a prezzo fisso (Fixed-Price Mechanism). Senza entrare nel merito degli algoritmi il problema generale dell'asta combinatoria può essere informalmente descritto come l'allocazione e l'assegnazione del prezzo di bundle di elementi, tali che la somma delle valutazioni dell'utente sia massimizzata. Mentre meccanismi a prezzo fisso assegnano bundle di elementi con una logica First Come First Served e con prezzi fissi predeterminati. Gli esperimenti rivelano che i meccanismi CA-LP e CA-GREEDY possono migliorare significativamente l'efficienza allocativa, generando maggiori ricavi per i fornitori di cloud.

In fine è stato presentato un modello predittivo, che riassume i compiti principali del broker descritti nel documento, iniziando dalle fasi di monitoraggio, forecast e predizione, si chiude il processo con la fase di calcolo delle funzioni di utilità del sistema, che sono state presentate nel documento "Modello di valutazione e ottimizzazione del cloud, implementazione dell'algoritmo". In figura segue il modello concettuale proposto [148].

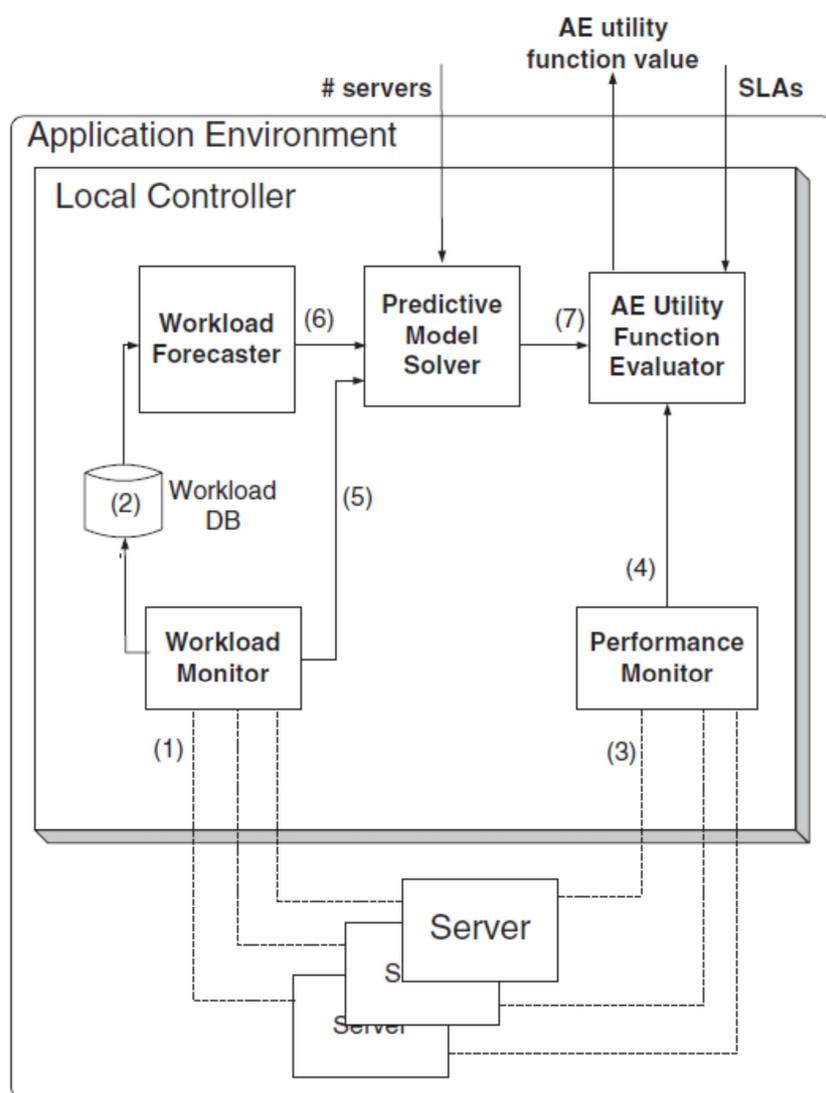


Figura 9 – L’architettura del modello predittivo (fonte [148])

Nella Figura 9 il Workload Monitor raccoglie i dati sui livelli di intensità del carico di lavoro (1) (ad esempio sul tasso di arrivo delle transazioni) e li memorizza in un Workload DB (2). Il componente Workload Forecast utilizza i dati di questo database per fare previsioni sui futuri livelli di intensità del carico di lavoro (6). Per questo strumento è ammessa qualsiasi tecnica di previsione.

Il modello predittivo-risolutivo usa sia misure di carico di lavoro attualmente osservate/misurate (5) oppure carichi di lavoro predetti (6). Queste previsioni, insieme a gli SLA per tutte le classi degli AE (Application Environment) sono utilizzati per calcolare la Utility Function. Se la funzione di utilità deve essere calcolata per la configurazione corrente, si ottengono delle misure di performance (tempo di risposta, throughput, etc.) ottenute dal Performance Monitor (3) al posto delle previsioni.

I contributi approfondiscono nel dettaglio il concetto di predizione, presentando per esteso lo sviluppo di algoritmi, modelli ed i risultati di numerose simulazioni. Si ritiene pertanto che tali modelli e loro eventuali evoluzioni possano costituire una base di partenza per le prossime fasi di test.



## Bibliografia

- [1] Abdullah, M. and M. Othman, *Cost-based Multi-QoS Job Scheduling Using Divisible Load Theory in Cloud Computing*. Procedia Computer Science, 2013. **18**: p. 928-935.
- [2] Abrishami, S. and M. Naghibzadeh, *Deadline-constrained workflow scheduling in software as a service Cloud*. Scientia Iranica, 2012. **19**(3): p. 680-689.
- [3] Abrishami, S., M. Naghibzadeh, and D.H.J. Epema, *Deadline-constrained workflow scheduling algorithms for Infrastructure as a Service Clouds*. Future Generation Computer Systems, 2013. **29**(1): p. 158-169.
- [4] Afgan, E., et al., *Galaxy CloudMan: delivering cloud compute clusters*. BMC Bioinformatics, 2010. **11 Suppl 12**: p. S4.
- [5] Amoretti, M., F. Zanichelli, and G. Conte, *Efficient autonomic cloud computing using online discrete event simulation*. Journal of Parallel and Distributed Computing, 2013. **73**(6): p. 767-776.
- [6] Angeli, D. and E. Masala, *A cost-effective cloud computing framework for accelerating multimedia communication simulations*. Journal of Parallel and Distributed Computing, 2012. **72**(10): p. 1373-1385.
- [7] Ardagna, D., et al., *Dual time-scale distributed capacity allocation and load redirect algorithms for cloud systems*. Journal of Parallel and Distributed Computing, 2012. **72**(6): p. 796-808.
- [8] Arutyunov, V.V., *Cloud computing: Its history of development, modern state, and future considerations*. Scientific and Technical Information Processing, 2012. **39**(3): p. 173-178.
- [9] Assunção, M.D., A. Costanzo, and R. Buyya, *A cost-benefit analysis of using cloud computing to extend the capacity of clusters*. Cluster Computing, 2010. **13**(3): p. 335-347.
- [10] Atif, M. and P. Strazdins, *Adaptive parallel application resource remapping through the live migration of virtual machines*. Future Generation Computer Systems, 2013.
- [11] Begnum, K., *Simplified cloud-oriented virtual machine management with MLN*. The Journal of Supercomputing, 2010. **61**(2): p. 251-266.
- [12] Benedict, S., *Performance issues and performance analysis tools for HPC cloud applications: a survey*. Computing, 2012. **95**(2): p. 89-108.
- [13] Bittencourt, L.F. and E.R.M. Madeira, *HCOC: a cost optimization algorithm for workflow scheduling in hybrid clouds*. Journal of Internet Services and Applications, 2011. **2**(3): p. 207-227.
- [14] Brandic, I., et al., *VieSLAF Framework: Enabling Adaptive and Versatile SLA-Management*. Grid Economics and Business Models, 2009. **5745**: p. 60-73.
- [15] Broberg, J., S. Venugopal, and R. Buyya, *Market-oriented Grids and Utility Computing: The State-of-the-art and Future Directions*. Journal of Grid Computing, 2007. **6**(3): p. 255-276.
- [16] Buyya, R., *CloudAnalyst A CloudSim-based Visual Modeller for Analysing Cloud Computing Environments and Applications*. AINA, 2010.
- [17] Caballer, M., et al., *EC3: Elastic Cloud Computing Cluster*. Journal of Computer and System Sciences, 2013. **79**(8): p. 1341-1351.
- [18] Calcavecchia, N.M., et al., *DEPAS: a decentralized probabilistic algorithm for auto-scaling*. Computing, 2012. **94**(8-10): p. 701-730.
- [19] Calheiros, R.N., et al., *A coordinator for scaling elastic applications across multiple clouds*. Future Generation Computer Systems, 2012. **28**(8): p. 1350-1362.
- [20] Calheiros, R.N., et al., *The Aneka platform and QoS-driven resource provisioning for elastic applications on hybrid Clouds*. Future Generation Computer Systems, 2012. **28**(6): p. 861-870.



- [21] Canali, C. and R. Lancellotti, *Exploiting ensemble techniques for automatic virtual machine clustering in cloud systems*. Automated Software Engineering, 2013: p. 1-26.
- [22] Chang, Y.-S., et al., *Scaling database performance on GPUs*. Information Systems Frontiers, 2011. **14**(4): p. 909-924.
- [23] Chapman, C., et al., *Software architecture definition for on-demand cloud provisioning*. Cluster Computing, 2011. **15**(2): p. 79-100.
- [24] Chen, H., et al., *Energy-aware application performance management in virtualized data centers*. Frontiers of Computer Science, 2012. **6**(4): p. 373-387.
- [25] Chen, M., et al., *Enabling Technologies for Future Data Center Networking: A Primer*. Ieee Network, 2013. **27**(4): p. 8-15.
- [26] Chen, T., R. Bahsoon, and G. Theodoropoulos, *Dynamic QoS Optimization Architecture for Cloud-based DDDAS*. Procedia Computer Science, 2013. **18**: p. 1881-1890.
- [27] Choi, S., K. Chung, and H. Yu, *Fault tolerance and QoS scheduling using CAN in mobile social cloud computing*. Cluster Computing, 2013.
- [28] Chun, S.-H. and B.-S. Choi, *Service models and pricing schemes for cloud computing*. Cluster Computing, 2013.
- [29] Chung, L., et al., *A goal-oriented simulation approach for obtaining good private cloud-based system architectures*. Journal of Systems and Software, 2013. **86**(9): p. 2242-2262.
- [30] Coelho da Silva, T.L., et al., *Non-Intrusive Elastic Query Processing in the Cloud*. Journal of Computer Science and Technology, 2013. **28**(6): p. 932-947.
- [31] Cuomo, A., et al., *An SLA-based Broker for Cloud Infrastructures*. Journal of Grid Computing, 2012. **11**(1): p. 1-25.
- [32] D'Agostino, D., et al., *A QoS-aware broker for hybrid clouds*. Computing, 2012. **95**(S1): p. 89-109.
- [33] Deboosere, L., et al., *Efficient resource management for virtual desktop cloud computing*. The Journal of Supercomputing, 2012. **62**(2): p. 741-767.
- [34] Delamare, S., et al., *SpeQuloS: a QoS service for hybrid and elastic computing infrastructures*. Cluster Computing, 2013.
- [35] Dhar, S., *From outsourcing to Cloud computing: evolution of IT services*. Management Research Review, 2012. **35**(8): p. 664-675.
- [36] Dou, W., et al., *An evaluation method of outsourcing services for developing an elastic cloud platform*. The Journal of Supercomputing, 2010. **63**(1): p. 1-23.
- [37] Durkee, D., *Why cloud computing will never be free*. Communications of the ACM, 2010. **53**(5): p. 62.
- [38] Faniyi, F., R. Bahsoon, and G. Theodoropoulos, *A Dynamic Data-Driven Simulation Approach for Preventing Service Level Agreement Violations in Cloud Federation*. Procedia Computer Science, 2012. **9**: p. 1167-1176.
- [39] García García, A., I. Blanquer Espert, and V. Hernández García, *SLA-driven dynamic cloud resource management*. Future Generation Computer Systems, 2014. **31**: p. 1-11.
- [40] Garg, S.K. and R. Buyya. *NetworkCloudSim: modelling parallel applications in cloud simulations*. in *Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on*. 2011. IEEE.
- [41] Garrison, G., S. Kim, and R.L. Wakefield, *Success factors for deploying cloud computing*. Communications of the ACM, 2012. **55**(9): p. 62.
- [42] Ghorbannia Delavar, A. and Y. Aryan, *HSGA: a hybrid heuristic algorithm for workflow scheduling in cloud systems*. Cluster Computing, 2013.
- [43] Gomez-Migueluez, I., V. Marojevic, and A. Gelonch, *Deployment and management of SDR cloud computing resources: problem definition and fundamental limits*. EURASIP Journal on Wireless Communications and Networking, 2013. **2013**(1): p. 1-11.
- [44] Goyal, T., A. Singh, and A. Agrawal, *Cloudsim: simulator for cloud computing infrastructure and modeling*. Procedia Engineering, 2012. **38**: p. 3566-3572.



- [45] Graciani Diaz, R., et al., *Belle-DIRAC Setup for Using Amazon Elastic Compute Cloud*. Journal of Grid Computing, 2011. **9**(1): p. 65-79.
- [46] Gulisano, V., et al., *StreamCloud: An Elastic and Scalable Data Streaming System*. Ieee Transactions on Parallel and Distributed Systems, 2012. **23**(12): p. 2351-2365.
- [47] Habib, S.M., et al., *Trust as a facilitator in cloud computing: a survey*. Journal of Cloud Computing, 2012. **1**(1): p. 1-18.
- [48] Han, R., et al., *Enabling cost-aware and adaptive elasticity of multi-tier cloud applications*. Future Generation Computer Systems, 2014. **32**: p. 82-98.
- [49] Han, Y., *IaaS cloud computing services for libraries: cloud storage and virtual machines*. OCLC Systems & Services, 2013. **29**(2): p. 87-100.
- [50] He, Y., et al., *A simulation cloud monitoring framework and its evaluation model*. Simulation Modelling Practice and Theory, 2013. **38**: p. 20-37.
- [51] Höfer, C.N. and G. Karagiannis, *Cloud computing services: taxonomy and comparison*. Journal of Internet Services and Applications, 2011. **2**(2): p. 81-94.
- [52] Hovestadt, M., et al., *Adaptive Online Compression in Clouds—Making Informed Decisions in Virtual Machine Environments*. Journal of Grid Computing, 2013. **11**(2): p. 167-186.
- [53] Hu, Y., et al. *Resource provisioning for cloud computing*. in *Proceedings of the 2009 Conference of the Center for Advanced Studies on Collaborative Research*. 2009. IBM Corp.
- [54] Huang, J. and D.M. Nicol, *Trust mechanisms for cloud computing*. Journal of Cloud Computing: Advances, Systems and Applications, 2013. **2**(1): p. 9.
- [55] Issarny, V., et al., *Service-oriented middleware for the Future Internet: state of the art and research directions*. Journal of Internet Services and Applications, 2011. **2**(1): p. 23-45.
- [56] Iyer, R., et al., *VM3: Measuring, modeling and managing VM shared resources*. Computer Networks, 2009. **53**(17): p. 2873-2887.
- [57] Jararweh, Y., et al. *Teachcloud: a cloud computing educational toolkit*. in *Proceedings of the 1st International IBM Cloud Academy Conference (ICA CON 2012)*, IBM, Research Triangle Park, NC, USA. 2012.
- [58] Javadi, B., J. Abawajy, and R. Buyya, *Failure-aware resource provisioning for hybrid Cloud infrastructure*. Journal of Parallel and Distributed Computing, 2012. **72**(10): p. 1318-1331.
- [59] Jing, S.-Y., et al., *State-of-the-art research study for green cloud computing*. The Journal of Supercomputing, 2011. **65**(1): p. 445-468.
- [60] Katsaros, G., et al., *A service framework for energy-aware monitoring and VM management in Clouds*. Future Generation Computer Systems, 2013. **29**(8): p. 2077-2091.
- [61] Kim, K.H., A. Beloglazov, and R. Buyya. *Power-aware provisioning of cloud resources for real-time services*. in *Proceedings of the 7th International Workshop on Middleware for Grids, Clouds and e-Science*. 2009. ACM.
- [62] Kliazovic, *A Holistic Model for Resource Representation in Virtualized Cloud Computing Data*. 2013.
- [63] Kliazovich, D., et al., *GreenCloud: A Packet-level Simulator of Energy-aware Cloud Computing Data Centers*. 2010 Ieee Global Telecommunications Conference Globecom 2010, 2010.
- [64] Kliazovich, D., P. Bouvry, and S.U. Khan, *DENS: data center energy-efficient network-aware scheduling*. Cluster Computing, 2011. **16**(1): p. 65-75.
- [65] Kossmann, D. and T. Kraska, *Data Management in the Cloud: Promises, State-of-the-art, and Open Questions*. Datenbank-Spektrum, 2010. **10**(3): p. 121-129.
- [66] Kousiouris, G., et al., *Dynamic, behavioral-based estimation of resource provisioning based on high-level application terms in Cloud platforms*. Future Generation Computer Systems, 2014. **32**: p. 27-40.
- [67] Krebs, R., C. Momm, and S. Kounev, *Metrics and techniques for quantifying performance isolation in cloud environments*. Science of Computer Programming, 2013.
- [68] Lai, K.-C. and Y.-F. Yu, *A scalable multi-attribute hybrid overlay for range queries on the cloud*. Information Systems Frontiers, 2011. **14**(4): p. 895-908.



- [69] Lee, H.M., Y.-S. Jeong, and H.J. Jang, *Performance analysis based resource allocation for green cloud computing*. The Journal of Supercomputing, 2013.
- [70] Lee, Y.C. and A.Y. Zomaya, *Energy efficient utilization of resources in cloud computing systems*. The Journal of Supercomputing, 2010. **60**(2): p. 268-280.
- [71] Li, B.H., et al., *New advances of the research on cloud simulation*, in *Advanced Methods, Techniques, and Applications in Modeling and Simulation 2012*, Springer. p. 144-163.
- [72] Li, C.-F., *Cloud Computing System Management Under Flat Rate Pricing*. Journal of Network and Systems Management, 2011. **19**(3): p. 305-318.
- [73] Li, C. and L. Li, *Efficient resource allocation for optimizing objectives of cloud users, IaaS provider and SaaS provider in cloud environment*. The Journal of Supercomputing, 2013. **65**(2): p. 866-885.
- [74] Li, C. and L.Y. Li, *Optimal resource provisioning for cloud computing environment*. The Journal of Supercomputing, 2012. **62**(2): p. 989-1022.
- [75] Li, J., et al., *Online optimization for scheduling preemptable tasks on IaaS cloud systems*. Journal of Parallel and Distributed Computing, 2012. **72**(5): p. 666-677.
- [76] Liao, X., L. Hu, and H. Jin, *Energy optimization schemes in cluster with virtual machines*. Cluster Computing, 2009. **13**(2): p. 113-126.
- [77] Lim, S.H., et al., *MDCSim: A Multi-tier Data Center Simulation Platform*. 2009 IEEE International Conference on Cluster Computing and Workshops, 2009: p. 317-325.
- [78] Lin, J., L. Zha, and Z. Xu, *Consolidated cluster systems for data centers in the cloud age: a survey and analysis*. Frontiers of Computer Science, 2012. **7**(1): p. 1-19.
- [79] Lin, K.W. and Y.-C. Lo, *Efficient algorithms for frequent pattern mining in many-task computing environments*. Knowledge-Based Systems, 2013. **49**: p. 10-21.
- [80] Liu, L., *Computing infrastructure for big data processing*. Frontiers of Computer Science, 2013. **7**(2): p. 165-170.
- [81] Liu, T., et al., *SDMS-O: A service deployment management system for optimization in clouds while guaranteeing users' QoS requirements*. Future Generation Computer Systems, 2012. **28**(7): p. 1100-1109.
- [82] Liu, X., et al., *Cloud-based computer simulation: Towards planting existing simulation software into the cloud*. Simulation Modelling Practice and Theory, 2012. **26**: p. 135-150.
- [83] Lucas-Simarro, J.L., et al., *Scheduling strategies for optimal service deployment across multiple clouds*. Future Generation Computer Systems, 2013. **29**(6): p. 1431-1441.
- [84] Maciel, P.D., et al., *Business-driven short-term management of a hybrid IT infrastructure*. Journal of Parallel and Distributed Computing, 2012. **72**(2): p. 106-119.
- [85] Malawski, M., K. Figiela, and J. Nabrzyski, *Cost minimization for computational applications on hybrid cloud infrastructures*. Future Generation Computer Systems, 2013. **29**(7): p. 1786-1794.
- [86] Manuel, P., *A trust model of cloud computing based on Quality of Service*. Annals of Operations Research, 2013.
- [87] Martens, B. and F. Teuteberg, *Decision-making in cloud computing environments: A cost and risk based approach*. Information Systems Frontiers, 2011. **14**(4): p. 871-893.
- [88] Mateescu, G., W. Gentsch, and C.J. Ribbens, *Hybrid Computing-Where HPC meets grid and Cloud Computing*. Future Generation Computer Systems-the International Journal of Grid Computing and Science, 2011. **27**(5): p. 440-453.
- [89] Mauch, V., M. Kunze, and M. Hillenbrand, *High performance cloud computing*. Future Generation Computer Systems, 2013. **29**(6): p. 1408-1416.
- [90] Maurer, M., I. Brandic, and R. Sakellariou, *Adaptive resource configuration for Cloud infrastructure management*. Future Generation Computer Systems, 2013. **29**(2): p. 472-487.
- [91] Maurer, M., et al., *Cost-benefit analysis of an SLA mapping approach for defining standardized Cloud computing goods*. Future Generation Computer Systems, 2012. **28**(1): p. 39-47.



- [92] Mazhelis, O. and P. Tyrväinen, *Economic aspects of hybrid cloud infrastructure: User organization perspective*. Information Systems Frontiers, 2011. **14**(4): p. 845-869.
- [93] Miguelez, G., *Deployment and management of SDR cloud computing resources: problem definition and fundamental limits*. JWCN, 2013.
- [94] Moltó, G., et al., *Elastic Memory Management of Virtualized Infrastructures for Applications with Dynamic Memory Requirements*. Procedia Computer Science, 2013. **18**: p. 159-168.
- [95] Moschakis, I.A. and H.D. Karatza, *Evaluation of gang scheduling performance and cost in a cloud computing system*. The Journal of Supercomputing, 2010. **59**(2): p. 975-992.
- [96] Nathuji, R., et al., *VPM tokens: virtual machine-aware power budgeting in datacenters*. Cluster Computing, 2009. **12**(2): p. 189-203.
- [97] Niezgodá, S.R., A.K. Kanjarla, and S.R. Kalidindi, *Novel microstructure quantification framework for databasing visualization and analysis of microstructure data*. Integrating Materials and Manufacturing Innovation, 2013. **2**(1): p. 1-27.
- [98] Núñez, A., et al., *iCanCloud: A Flexible and Scalable Cloud Infrastructure Simulator*. Journal of Grid Computing, 2012. **10**(1): p. 185-209.
- [99] Oliveira, D., et al., *A Provenance-based Adaptive Scheduling Heuristic for Parallel Scientific Workflows in Clouds*. Journal of Grid Computing, 2012. **10**(3): p. 521-552.
- [100] Prodan, R., M. Sperk, and S. Ostermann, *Evaluating High-Performance Computing on Google App Engine*. Ieee Software, 2012. **29**(2): p. 52-58.
- [101] Quiroz, A., et al. *Towards autonomic workload provisioning for enterprise grids and clouds*. in *Grid Computing, 2009 10th IEEE/ACM International Conference on*. 2009. IEEE.
- [102] Ranjan, R. and L. Zhao, *Peer-to-peer service provisioning in cloud computing environments*. The Journal of Supercomputing, 2011. **65**(1): p. 154-184.
- [103] Redmond, A., et al., *Exploring how information exchanges can be enhanced through Cloud BIM*. Automation in Construction, 2012. **24**: p. 175-183.
- [104] Rodero-Merino, L., et al., *Using clouds to scale grid resources: An economic model*. Future Generation Computer Systems, 2012. **28**(4): p. 633-646.
- [105] Rodero, I., et al., *Energy-Efficient Thermal-Aware Autonomic Management of Virtualized HPC Cloud Infrastructure*. Journal of Grid Computing, 2012. **10**(3): p. 447-473.
- [106] Rogers, O. and D. Cliff, *A financial brokerage model for cloud computing*. Journal of Cloud Computing, 2012. **1**(1): p. 1-12.
- [107] Sakellari, G. and G. Loukas, *A survey of mathematical models, simulation approaches and testbeds used for research in cloud computing*. Simulation Modelling Practice and Theory, 2013. **39**: p. 92-103.
- [108] Sakr, S., *Cloud-hosted databases: technologies, challenges and opportunities*. Cluster Computing, 2013.
- [109] Sethia, P. and K. Karlapalem, *A multi-agent simulation framework on small Hadoop cluster*. Engineering Applications of Artificial Intelligence, 2011. **24**(7): p. 1120-1127.
- [110] Sharma, S., et al., *Decentralized allocation of CPU computation power for web applications*. Performance Evaluation, 2010. **67**(11): p. 1187-1202.
- [111] Shiraz, M., et al., *Investigation on runtime partitioning of elastic mobile applications for mobile cloud computing*. The Journal of Supercomputing, 2013.
- [112] Son, S., G. Jung, and S.C. Jun, *An SLA-based cloud computing that facilitates resource allocation in the distributed data centers of a cloud provider*. The Journal of Supercomputing, 2013. **64**(2): p. 606-637.
- [113] Son, S. and K.M. Sim, *Adaptive and similarity-based tradeoff algorithms in a price-timeslot-QoS negotiation system to establish cloud SLAs*. Information Systems Frontiers, 2013.
- [114] Song, F., et al., *An Optimization-Based Scheme for Efficient Virtual Machine Placement*. International Journal of Parallel Programming, 2013.



- [115] Streitberger, W. and T. Eymann, *A simulation of an economic, self-organising resource allocation approach for application layer networks*. Computer Networks, 2009. **53**(10): p. 1760-1770.
- [116] Sun, G., et al., *A cost efficient framework and algorithm for embedding dynamic virtual network requests*. Future Generation Computer Systems, 2013. **29**(5): p. 1265-1277.
- [117] Taylor, R.C., *An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics*. BMC Bioinformatics, 2010. **11**(Suppl 12): p. S1.
- [118] Tian, W., Q. Xiong, and J. Cao, *An online parallel scheduling method with application to energy-efficiency in cloud computing*. The Journal of Supercomputing, 2013. **66**(3): p. 1773-1790.
- [119] Truong Huu, T., et al., *Joint Elastic Cloud and Virtual Network Framework for Application Performance-cost Optimization*. Journal of Grid Computing, 2010. **9**(1): p. 27-47.
- [120] Van den Bossche, R., K. Vanmechelen, and J. Broeckhove, *Online cost-efficient scheduling of deadline-constrained workloads on hybrid clouds*. Future Generation Computer Systems, 2013. **29**(4): p. 973-985.
- [121] Vanmechelen, K., S. De Munck, and J. Broeckhove, *Conservative distributed discrete-event simulation on the Amazon EC2 cloud: An evaluation of time synchronization protocol performance and cost efficiency*. Simulation Modelling Practice and Theory, 2013. **34**: p. 126-143.
- [122] Vázquez, C., et al., *On the use of clouds for grid resource provisioning*. Future Generation Computer Systems, 2011. **27**(5): p. 600-605.
- [123] Vouk, M.A., *Cloud computing - Issues, research and implementations*. Proceedings of the Iti 2008 30th International Conference on Information Technology Interfaces, 2008: p. 31-40.
- [124] Walraven, S., E. Truyen, and W. Joosen, *Comparing PaaS offerings in light of SaaS development*. Computing, 2013.
- [125] Wang, D., *An Efficient Cloud Storage Model for Heterogeneous Cloud Infrastructures*. Procedia Engineering, 2011. **23**: p. 510-515.
- [126] Wang, J.-z., P. Varman, and C.-s. Xie, *Optimizing storage performance in public cloud platforms*. Journal of Zhejiang University SCIENCE C, 2011. **12**(12): p. 951-964.
- [127] Wang, L.Z., et al., *Cloud Computing: a Perspective Study*. New Generation Computing, 2010. **28**(2): p. 137-146.
- [128] Wang, W.-J., et al., *Adaptive scheduling for parallel tasks with QoS satisfaction for hybrid cloud environments*. The Journal of Supercomputing, 2013. **66**(2): p. 783-811.
- [129] Wang, W., et al., *Cloud-DLS: Dynamic trusted scheduling for Cloud computing*. Expert Systems with Applications, 2012. **39**(3): p. 2321-2329.
- [130] Wang, X., Z. Du, and Y. Chen, *An adaptive model-free resource and power management approach for multi-tier cloud environments*. Journal of Systems and Software, 2012. **85**(5): p. 1135-1146.
- [131] Warneke, D. and O. Kao, *Exploiting Dynamic Resource Allocation for Efficient Parallel Data Processing in the Cloud*. Ieee Transactions on Parallel and Distributed Systems, 2011. **22**(6): p. 985-997.
- [132] Wei K. Tsai, M.I.a.J.R., *Thresholds Revisit the Strings Versus Clouds Debate for the Internet Architecture, Part II QoS, Control, Management, and TCP*. Journal of Network and Systems Management, 2002.
- [133] Wei K. Tsai, M.I.a.J.R., *Thresholds Revisit the Strings versus Clouds Debate for the Internet Architecture. Part I Control, Scalability, and QoS*. Journal of Network and Systems Management, 2002.
- [134] Weinhardt, C., et al., *Cloud Computing - A Classification, Business Models, and Research Directions*. Business & Information Systems Engineering, 2009. **1**(5): p. 391-+.
- [135] Wickremasinghe, B., R.N. Calheiros, and R. Buyya. *Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications*. in *Advanced*



- Information Networking and Applications (AINA), 2010 24th IEEE International Conference on.* 2010. IEEE.
- [136] Wright, P., et al., *A constraints-based resource discovery model for multi-provider cloud environments.* Journal of Cloud Computing, 2012. **1**(1): p. 1-14.
- [137] Wu, C.-M., R.-S. Chang, and H.-Y. Chan, *A green energy-efficient scheduling algorithm using the DVFS technique for cloud datacenters.* Future Generation Computer Systems, 2013.
- [138] Wu, L., S. Kumar Garg, and R. Buyya, *SLA-based admission control for a Software-as-a-Service provider in Cloud computing environments.* Journal of Computer and System Sciences, 2012. **78**(5): p. 1280-1299.
- [139] Xu, B., et al., *Job scheduling algorithm based on Berger model in cloud environment.* Advances in Engineering Software, 2011. **42**(7): p. 419-425.
- [140] Yan-hua, Z., F. Lei, and Y. Zhi, *Optimization of Cloud Database Route Scheduling Based on Combination of Genetic Algorithm and Ant Colony Algorithm.* Procedia Engineering, 2011. **15**: p. 3341-3345.
- [141] Yang, J., et al., *A cost-aware auto-scaling approach using the workload prediction in service clouds.* Information Systems Frontiers, 2013.
- [142] Yang, K. and X. Jia, *Data storage auditing service in cloud computing: challenges, methods and opportunities.* World Wide Web, 2011. **15**(4): p. 409-428.
- [143] Yang, S.Q., *Move into the Cloud, shall we?* Library Hi Tech News, 2012. **29**(1): p. 4-7.
- [144] Yoo, C.S., *Cloud Computing: Architectural and Policy Implications.* Review of Industrial Organization, 2011. **38**(4): p. 405-421.
- [145] Zaman, S. and D. Grosu, *Combinatorial auction-based allocation of virtual machine instances in clouds.* Journal of Parallel and Distributed Computing, 2013. **73**(4): p. 495-508.
- [146] Zhang, Q., L. Cheng, and R. Boutaba, *Cloud computing: state-of-the-art and research challenges.* Journal of Internet Services and Applications, 2010. **1**(1): p. 7-18.
- [147] Zhang, T., et al., *Typical Virtual Appliances: An optimized mechanism for virtual appliances provisioning and management.* Journal of Systems and Software, 2011. **84**(3): p. 377-387.
- [148] Bennani M.N., et al., *Resource Allocation for Autonomic Data Centers using Analytic Performance Models.* Computer Society, 2005
- [149] Son S., et al., *A Price-and-Time-Slot-Negotiation Mechanism for Cloud Service Reservations,* Systems, Man, And Cybernetics, 2012.
- [150] Xiao Z., et.al., *Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment,* Parallel And Distributed Systems, 2013
- [151] Yang, J., et al., *A new method based on PSR and EA-GMDH for host load prediction in cloud computing system.* J Supercomput, 2014.
- [152] Yeo C.S., et.al., *Autonomic Metered Pricing for a Utility Computing Service,* Future Generation Computer Systems, 2009
- [153] Zhang Q., et.al, *Dynamic Resource Allocation for Spot Markets in Clouds,* Utility and Cloud Computing, 2013