VOL. 10, NO. 8, MAY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

BROKER BASED QoS CENTRIC RESOURCE PROVISIONING FRAMEWORK WITH FINANCIAL OPTIONS

Sahar Arshad[1], Saeed Ullah[2], Shoab Ahmed Khan[3], M. Daud Awan[4] and M. Sikandar Hayat Khayal[5]
[1,3]Department of Computer Engineering, College of EME, National University of Sciences & Technology (NUST), Islamabad, Pakistan
[2, 4, 5] Faculty of Computer Science, Preston University, Islamabad, Pakistan
E-mail: sahar.arshad87@gmail.com

ABSTRACT

Cloud computing has grasped the attention of scientific community and business industry towards the provisioning of computing resources as a utility and software as a service over a network. Access to software or hardware resources are seamless to cloud consumers in a way that provider of the service could be anywhere in the world by providing services in nebulous cloud. Most of the research studies have focused on the benefits associated with single cloud provider by provisioning and managing resources at provider's end. This has led to the problems associated with pricing and quality of the services that cloud provider guarantees in SLA to their consumers. To tackle this issue, financial options theory in cloud computing economics has been introduced in the recent past where options can be bought in advance so that these may be utilized when required. However, QoS factors, that are vital for a cloud provider in order to accommodate cloud consumer trust, are not yet been addressed. We propose a federated cloud resource provisioning framework based on financial options theory to address the issue of job denial during peaks by leasing spare capacity from other cloud providers as well as satisfying cost and QoS requirements. We assess cloud providers based on the services they provide and best-fit selection is made while purchasing options to meet service level objectives. Also the issue of VM placement across multiple clouds is addressed to provide a resource sharing platform where the objectives of avoiding resource rejection and efficient service level agreement management are incorporated in the proposed framework. Using CloudSim simulator, various workloads were used to test our proposed framework. Results indicate that our broker based cloud federation framework provides a win-win scenario for both cloud providers and consumers by avoiding job denial and achieving better SLA management and resource utilization.

Keywords: cloud federation, resource provisioning, quality of service, financial options, advanced resource reservation, service level agreement, broker.

1. INTRODUCTION

In recent years, cloud computing has grasped the attention of scientific community and business industry towards the provisioning of computing resources as a utility and software as a service over a network [1]. Huge investments in infrastructure, maintenance and up-gradation is required for a business that needs to maintain records of inventory, payroll, sales etc. Cloud computing benefits in a way that it cuts down the capital cost required to purchase and install computing devices and provision the capacities in form of services as and when required in just pay as you go fashion. Business owners can simply acquire or rent-out the hardware, software or any other service according to business needs and can just pay to the cloud provider without being worry for its maintenance and other associated issues [2].

In cloud market, cloud providers provision computing resources to their clients by providing interface for their pool of resources. The resources can be provisioned on demand, by reserving the resources in advance or via spot instances from the pool. Reserved instances are mostly cheaper, almost half of the price and ensures higher availability than on-demand instances, and thus these instances are more demanding as compared to on-demand instances. Moreover, cloud provider benefits the guaranteed cash flow from these instances even if resources are not fully utilized by the consumer.

Advanced reservation is not a new concept in cloud computing. Many cloud middle-wares support advanced reservation of VM instances within a tight time interval [3]. However, this may not be a feasible solution for scientific and high performance applications where execution intervals are not fixed in advance and job placement can even be delayed for weeks due to technical reasons. This is also not a reasonable solution for cloud providers where load spikes are not known in advance by the consumer and a cloud vendor cannot hedge against unavailability of resources by utilizing advanced reservations not in use by the consumers at a certain period of time. During such valleys, this leads to a probable situation of partial or less utilization of reserved instances. Furthermore, a single cloud provider can't hold the perception of unlimited availability of their resources to the consumer who simply rely their business by renting resources. For instance, Amazon EC2 consumer has a limit of 20 reserved instances per availability zone. Thus, cloud providers (CPs) need to adopt an approach of allocating reserved instances that are currently not being used by owner to the new incoming on-demand requests without violating their Service Level Agreement (SLA) with the consumers of reserved instances who are guaranteed with availability of resources. However, in order to make sure the availability of reserved instances, spare resources may be required to reserve by CP from other CPs in a Federated Cloud Environment [4]. Such cloud cooperation provides a way to maximum resource utilization as well as the revenue for provider to compete in cloud market.

By definition [5]:"*Cloud federation comprises services from different providers aggregated in a single pool supporting three basic interoperability features - resource migration, resource redundancy and combination of complementary resources resp. services.*"

With Cloud Federation, cloud provider makes sure of accommodating the agreed SLA terms maintaining trust and reliability with provided services. In such an environment, cloud providers are capable of managing on-demand and reserved instances along with insourcing/ outsourcing the resources through federation as and when required to keep maintaining the perception of unlimited availability of the resources in data center. However, since CPs are independent of each other in federation, there is no legal binding to offer resources when required. This may result with issues of:

- Resource shortage during peaks as cloud resource utilization of different vendors in federation may reach at maximum level.
- Varying future price for the resources by the providers in federation for their spare resources. These prices cannot be predicted by using historic prices simply.

The above the stated issues can be addressed by using the concept of advanced reservation of resources through financial options theory in finance [6]. In cloud computing, financial option is applied to resources that are considered as underlying assets to which a premium price has to be paid by the consumer to the provider as a security of the asset.

By definition [7]: "*Option is a contract which gives the buyer the right, but not obligation, to buy or sell an underlying asset at a specified strike price on or before a specific date*". Option is then exercised on request by the consumer to the provider with a strike price (price paid to exercise an option in future date). Option can either be call or put; in case of call option; an underlying asset (computing resource) is purchased by the consumer. However, in case of put option that particular asset is sold by the provider at a strike price. Option can be either American or European; in case of American option it may be exercised at any time within the time period of a contract. On contrary, European option can only be exercised after the maturity of a contract [7].

In terms of federated option contracts, a consumer can benefit by buying and exercising the options from the federated clouds in order to satisfy its customers as well as keeping the reputation up. And by selling its own option to the concerned parties would be a profitable business activity for the consumer. Even if the option is not exercised, still the worth associated in purchasing it from the provider would be effecting in mitigating the risks of untrustworthiness and unmet requirements of the clients.

Thus, the motivation of this research work is to propose a federated cloud resource provisioning framework based on financial options to address the issue of job denial during peaks by leasing the spare capacity from other cloud providers as well as to satisfy the cost and QoS requirements being singed in SLA document.

Problem statement

Cloud federation provides a win-win situation for both cloud providers and consumers. As, Cloud providers can get a premium price against an option being sold and this non-refundable money can be utilized in business demands. On the other hand, cloud consumers can buy financial options for advanced reservations and may use these options during load spikes when existing resources of consumers are not sufficient to meet the demand.

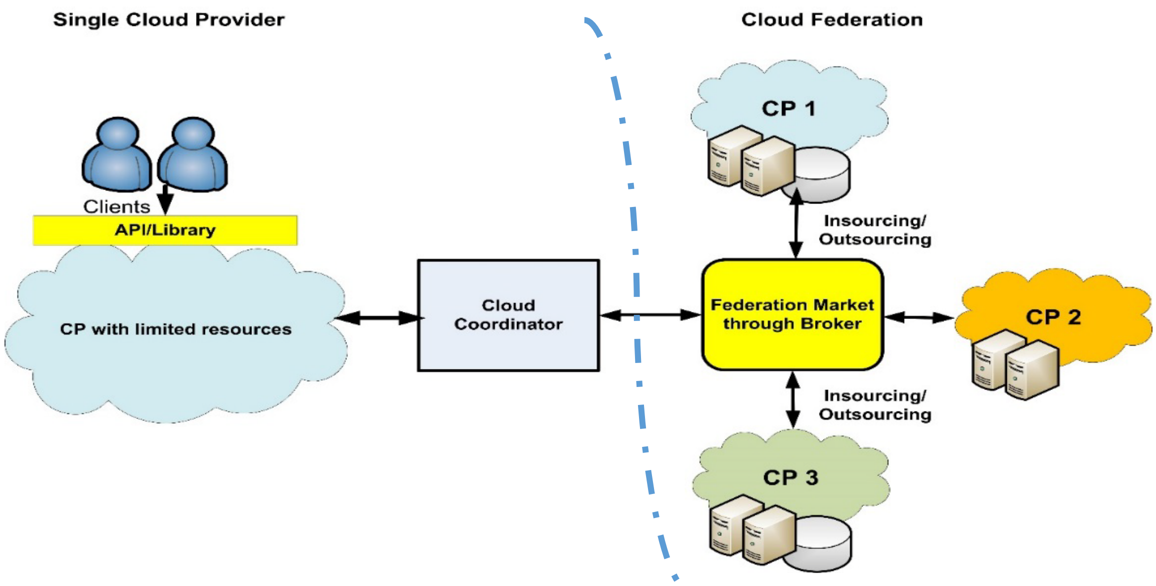


Figure-1. Single cloud vs. Federated cloud environment, adapted from [8].

Different research efforts have been made to use the concept of options in grid, cloud and federation of cloud but none of these have addressed the issue of Quality of Service while purchasing options [4],[8]. Different cloud vendors offer different SLAs (Service Level Agreements), focusing on different a variety of Quality of Service (QoS) attributes i.e., performance, network bandwidth, storage capacity. Since job quality requirements may vary considering the nature of job, incoming job request should be matched with exact QoS while purchasing an option while is a missing area in contemporary research. The stuy is aimed to address the following research question:

- How can a financial option based resource acquisition framework be designed in Cloud Federation while meeting QoS requirements?

The main contributions of the research are as under:

- A novel financial options based resource provision framework is proposed that addresses the issue of job satisfaction by leasing computational resources that satisfies user QoS requirements. This issue is not yet addressed in the literature.
- We assess different cloud providers based on the services they provide and best-fit selection is made while purchasing options to meet service level objectives (SLOs).

The remainder of this paper is organized as follows: The next section discusses related work followed by our proposed framework and methodology. Section 4 descries workload and simulation setup along with experimental results. The final section includes conclusion and possible future directions.

2. RELATED WORK

Cloud computing is an evolving research area in the field of distributed computing. Most of the research is focused on cloud providers' profit maximization by considering different pricing strategies. However, not much focus is given to the consumer of cloud services, whose primary concern is to attain high Quality of Service (QoS) along-with affordable cost. Our study is focused on a new pricing strategy, option based pricing model, for provisioning of computing resources based on QoS requirements provided in SLA.

We performed a systematic literature review to identify and retrieve related studies in cloud computing considering the guidelines proposed by [9].We derived the following PICOC criteria to structure our research question (as mentioned in problem definition) in order to identify the gap in current research of cloud computing.

- **Population:** Cloud Computing Models including IaaS, SaaS, PaaS etc.
- **Intervention:** Pricing Models based on Quality of Service
- **Outcome:** Pricing strategies/frameworks/models

The issue of pricing has always been a topic of consideration in research with the advancement of cloud computing services. As, the dynamic nature of the commodities for varying job utilization needs the varying pricing factors other than the static pricing models, thus, considering this scenario, Wubin Li et al. [10] evaluated the varying dynamic pricing schemes for service placement in cloud. Various algorithms from heuristics to combinatorial optimization solutions were deployed at providers end to get the optimum results. Greedy algorithmic approach was considered as the best optimization solution for service placement under dynamic pricing scheme.

Another challenge that cloud providers face in resource provisioning is the underutilization or over-provisioning of the resources in a cloud environment. To overcome underutilization, provider has to invest more in the capital to serve the rejected requests due to over utilization /over provisioning of the resources. On other hand, underutilization of resources from the customer due to high cost or off peak times, maintenance charges of spare capacities are costing more to the provider. Even in cloud federation, this problem cannot be sort out as cloud providers in a federation are not liable to any contract between them. Thus authors in [4] proposed a financial option (future contract) based solution for cloud federation by proposing Cloud Asset Pricing Tree (CAPT) model to get optimal premium price for options purchased.

Pricing the computing commodities with options in grid leads the way to propose similar pricing scheme for the cloud as well. Thus, B. Sharma et al. [6] proposed a novel option based financial economic model, capable to consider the QoS factors for the consumers while provisioning resources in cloud. Cloud providers were facilitated by the lower bound price to charge the consumer using options, while the upper bound was set using the compounded Moore's law. It has been suggested by the authors to charge the commodity within the range of two boundaries (upper, lower) for effective usage of resource(s). However, mapping of cloud parameters to Black and Scholes. Model (BSM) are inconsistent with respect to the quality parameter vs. interest rate which is used to calculate compounded Moore's law.

In [8], author proposed financial options in federated clouds to overcome the underutilized issues as well as availability of the resources when required to the users with reserved instances, trading of resources by buying and selling options are suggested. However, selling strategies are not covered in the experiments.

Since our study is aimed at proposing a framework based on advanced reservations using financial options as well as considering the QoS requirements as required at user end, we can infer that the above studies [6], [8], [12], [13], and [14] have not yet addressed this core issue. Although different techniques and models have been proposed by using BSM, Trinomial lattice and so on, but such schemes do not cover the QoS requirements that need to be satisfied in order to achieve service level objectives, defined in SLA.

ARPN Journal of Engineering and Applied Sciences

Table-1. Comparison of varying pricing models and QoS attributes from the selective studies.

Study #	Objective	Methodology	Platform/tool	Novelty	QoS coverage	Limitations	Future Work
[10]	To investigate the providers' cost optimization problem and to propose the optimal service placement algorithm under dynamic pricing scheme	Simulation based on Experiments	Optimus Cloud Toolkit	----	No	----	To design scheduling mechanism for service placement in dynamic pricing scenario
[15]	To propose a hybrid optimal model to maximize the revenue for the provider and with minimum cost charged to the clients	Numerical Modeling	Mathematical formulations	Using the economic concept of movement along the demand curve and tiered pricing policy in order to increase revenue with more clients	No	• No application has been provided to the proposed model • Job arrival and exit are assumed uniform	Job utility can be added to evaluate the proposed model
[11]	To investigate the VM procurement cost with the achieved QoE of end user	Numerical Modeling	Theoretical Model	Addressed the joint problem of resource procurement and provisioning under mixed pricing strategies	Not explicitly mentioned	Bidding strategies for spot VM are not addressed	Simulation to evaluate the proposed model and to cover the QoS availability factor
[4]	To propose a model in order to calculate the optimal premium price for cloud federation options	Simulation	Trinomial Tree	Decision making for the cloud provider regarding the option purchase and exercising	No	-----	To accommodate QoS factors in the proposed model
[16]	To provide a tool that helps in reducing the cost of running the HPC applications within deadline by providing the appropriate cluster size and cloud instance type	Experimentation by implementing CAP^3 on top of Amazon EC2	Star Cluster to manage VMs Python with NumPy Lib to program CAP^3 Eight MPI C/Fortran	Framework capable to automatically profile the HPC application and predict its performance by providing the proper cluster size to finish job within its	Not mentioned	Present schedule module of CAP^3 only supports one task per customer	▪ To support multiple tasks from multiple customers in schedule module ▪ To integrate sophisticated bidding strategies and recovery mechanism during preemption

			benchmarks from NAS Parallel Benchmarks (NPB)	deadline while minimizing the cost			
[12]	To propose a financial option based model that maintains an equilibrium between the user QoS expectation and providers' profitability in grid computing	Experimentation	Option price calculation using Trinomial lattice	Determination of best time to exercise the option	availability	----	To create an efficient resource provisioning model across heterogeneous grid environment
[13]	Integration of pricing models with GridSim tool to simulate compute resource price in grid	Simulation	GridSim toolkit	----	Not mentioned specifically	-----	Establishment of relation between cost of the compute resource and to exercise an option and relation between the average cost of the computing instance to the users' budget
[6]	To calculate the realistic value of cloud commodities by employing financial option theory and set resources as underlying cloud asset	Mapping Algorithm from cloud parameters to the BSM model	BSM and compounded Moores' Law	A novel financial economic model handling the QoS issues in cloud computing	Not mentioned specifically	Maintenance cost of computing resources is not considered in pricing the commodities Initial cost for provider and consumer is set to be same that is not possible	More realistic cloud pricing solutions
[8]	To overcome the underutilization issue of the resources by hedging against the reserved instance consumers	Experimentation	Binomial tree	---	availability	No strategy for providers willing to sell their unused option is given	Selling option strategies

3. PROPOSED FRAMEWORK

This section presents our proposed framework and the relevant algorithms in order to achieve the research objectives.

Process for resource acquisition

To represent resource acquisition process in cloud federation, pseudo-code for process flow followed by the Broker/facilitator and cloud consumer in cloud federation is presented in figure 2 and 3 respectively.

Broker process flow

In this process, a facilitator is periodically acquiring SLA details from the cloud providers in federation dealing with options (i.e. trading of resources in cloud). With reception of incoming job from the consumer of the cloud, the facilitator filters job requirements according to the SLA details of the providers. Providers' SLA not fulfilling the QoS details of the requester are discarded from the list and a sorted list based on QoS of prioritized cloud providers is returned to the consumer/requestor of the cloud.

```
While now < schedule_time
    Request SLA details from each option provider
    While (true) do
        Receive incoming job j
        for each job j in J
            Filter job requirements based on cloud providers' SLA
            Remove cloud providers with insufficient QoS
            Prioritize each candidate cloud provider based on QoS
            Return sorted list of best-matched Cloud providers
        end for
    end while
schedule_time = schedule_time + t
end while
```

Figure-2. Algorithm for Broker/Facilitator in Cloud.

Pseudo code for cloud consumer

In the figure below, pseudo-code for a consumer, adapted from [26], is presented, in which incoming on-demand job requests are accommodated in the VMs pool of on-demand and reserved instances. Whenever a request arrives, a decision is made based on the availability of the on-demand instances. If for instance, no on-demand instance is available within the allowed waiting time of a request, the available reserved instances from the VM pool is granted to complete the job request. However, before assigning the job to the reserved instance, the coordinator of CP communicates with the broker in order to calculate the premium price of the options available with various cloud providers in the federation and compare whether there is a profit with the purchase and exercise of option or not. In case of no profit, the provider with job request may simply reject it and would not take risk of buying and exercising options with no utility. However, options available with profit is considered by the provider and purchased by the provider and the requested job is allocated to the spare reserved instance.

VOL. 10, NO. 8, MAY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

```
While (true) do

while now < schedue_time          do
        queue incoming job j
         vms  ← all current on-demand and reserved VMs in pool

        foreach   job  j in J
                    decision ← FindFreeOnDemand(j.jobspecification, vms)
                    if   (decision.allocated=true)   then
                         AllocateJobToOnDemandVM(j, decision.VM)

                     else if   (CheckReservedSpace(j. jobspecification, vms)= true)  then
                      price← Check option and exercise price having reserved j VM specifications
                      if (price < j.price + α)
                            Purchase option for reserved instance
                            AllocateJobToReservedVM( j, decision.VM)
                    end if

schedule_time = schedule_time + t

end while
end while
```

Figure-3. Algorithm for Cloud Consumer in Cloud Federation, adapted from [26].

Where t is a small timestamp used for scheduling of incoming jobs and α is a minor profit adjustment factor to exercise an option.

Employed financial model using BSM model with options

In our work, BSM (Black-Scholes-Merton) model is considered for employing the financial model that could map its attributes to the cloud parameters. BSM model was proposed in 1973 by Fisher Black, Myron Scholes and Robert Merton. With BSM model, calculating the options in large amount is possible in a short interval of time considering the requirement of cloud federation where call and put an option is very frequent.

BSM model is comprised of capital asset pricing model where a relationship is set between the required return from the market on the option to the required return on the underlying asset. This relationship thus exists between both the asset price and time. The model incorporates the important volatility factor that can be calculate either from the historical data or can be implied from option price using model [7].

Thus to put and call an option price following equations are used:

$$Call = S_0 N(d_1) - Ke^{-rT}\, N(d_2)$$

$$Put = Ke^{-rT}\, N(-d_2) - S_0 N(-d_1)$$

$$d_1 = \frac{\ln(S_0/K) + (r + {\sigma^2}/{2})T}{\sigma\sqrt{T}}$$

$$d_2 = \frac{\ln(S_0/K) + (r - {\sigma^2}/{2})T}{\sigma\sqrt{T}}$$

where,
S_0 is the resource price underlying
K is the strike price for contract
r is the interest rate
σ is volatility
T is the maturity date of the option contract
N(d) is the probability that the option will be exercised in a risk-neutral environment
σ can be calculated based on the historical data

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} u_i^2 - \frac{1}{n(n-1)}\left(\sum_{i=1}^{n} u_i\right)^2}$$

$$\sigma = \frac{s}{\sqrt{\tau}}$$

We propose an algorithm based on the employed BSM model to map the BSM parameters to the cloud resource in provisioning with quality metrics (through SLA) to estimate option price of VM for consumers of cloud federation. American Options are used in our work as it suits best to the dynamic nature of cloud federation.

VOL. 10, NO. 8, MAY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

1. Input Cloud Parameters
   Commodity Investment → Calculated through statistical analysis of particular resource
   Option Contract Time
   Resource Strike Price → based on resource deprecation rate (mostly assumed as 4 years [17])
   Rate of Interest (For Amazon, 1.5% monthly or 19.5% yearly interest rate [18])
   Volatility estimate from historical data (Amazon: 15%)
2. Map cloud parameters to BSM model to evaluate option price
   S_0 ← Commodity investment in cloud
   K ← Strike price of resource
   r ← Rate of interest
   T ← Option contract time
   σ ← Volatility estimate
3. Using BSM, calculate call and put in pricing cloud resources

Figure-4. Proposed Algorithm in order to map cloud parameter to BSM model for calculating Option price.

Proposed framework

Figure 5 depicts a federated cloud scenario of different cloud providers with following assumptions:

- Each Cloud Provider is an autonomous cloud provider having its own clients in datacenter and is also associated with a facilitator/broker in the federation.
- Each provider is supporting on-demand and reserved instance pricing model based on Amazon pricing model to its incoming client requests [19].
- Each cloud provider ensures its quality metrics by providing the details in the agreement, i.e. SLA.

In our work, we consider the quality parameters specified in SLAs such as availability, bandwidth, and storage capacity along with the cost of the resources (VMs) providing these quality metrics while operating. The values of these metrics update periodically depending upon the utilization of the resources in their datacenters as well as the type of resources under use [20].

The broker periodically updates its record and consumer (CP) may check its SLA against the one in the provided list of SLA by the broker in order to match or relate requirements. Later, the cloud provider selects the most suitable cloud provider and makes contract, in order to trade the resources at time when demand spike.

The core modules of the broker in a federated cloud environment are:

Job management module

This module is responsible for managing the requests from the consumer of cloud federation along with their SLAs and negotiation contracts to interact with the match maker module in order to invoke resource provisioning mechanism between the consumer and provider which suits best to the requirements specified in form of SLA.

Three further subcomponents in this module are:

i. Negotiation engine

With the negotiation terms, we refer the strategies that need to be adopted in form of penalties in case of SLA violations form the cloud provider. Moreover, this component is solely negotiating the QoS parameters (availability, bandwidth, storage as in our case) that are to be required by the consumer form the provider after the one time negotiation settlement with the provider along with their priorities. This also covers the cost factors for the resources that will be utilized once the settlement has been made. All these contents of the negotiation terms are mentioned in XML schema of SLA (section 3.4). The negotiating contents, as presented in figure 3.10 of SLA specified by the consumer, are availability: 99.9%, communication bandwidth: high, while storage I/O and cost: medium [21].

ii. SLA manager

This component manages, monitors and validates the agreed upon terms between provider and consumer after negotiation phase. The SLAs (XML schema) of both the parties are managed here. Any SLA violation is monitored and the penalty is executed by the manager to the consumer [22].

iii. Resource provisioning

This module ensures the provisioning of resources by the provider to its consumer once the request is initiated by the Job management module considering the negotiation terms and settlements into account. The degree of provisioning SLA assurance however is evaluated by the SLA Manager.

iv. Match maker module

This module is responsible to provide the best match list to the consumer by matching the SLA requirements of the consumer request with the one periodically received and updated from the providers in the federation. The two main components of this module are:. 1) QoS Manager deals with the promised service quality level specified by the provider in terms of quality parameters. 2) The QoS ontology is a set of quality terms specified to capture the VM Performance, Network, Storage and Memory metrics in order to measure the level of quality provided to the consumer.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

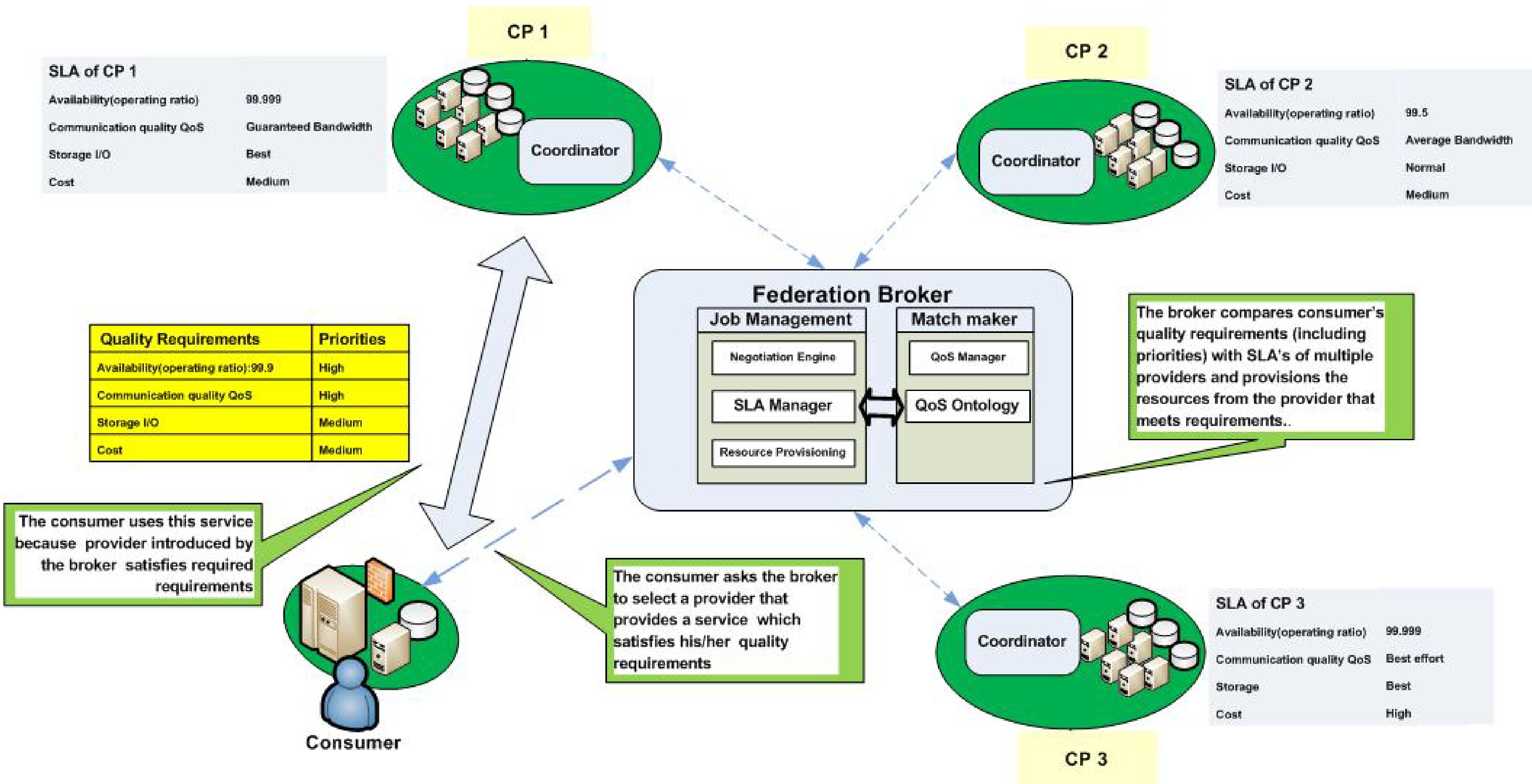


Figure-5. Architecture of QoS centric resource provision framework in cloud federation.

4. SIMULATION RESULTS

In this section, we present the implementation and validation details of our framework. We used CloudSim simulator for evaluation of our proposed policies which is an IaaS resource provisioning, resource pricing, scheduling and cloud policy management library developed in Java. It simulates cloud datacenters, hosts, VMs and job management of isolated as well as federated cloud environment [23]. We extended broker and VM resource provisioning policies of the library to evaluate our proposed framework for cloud federation. We developed and used the following two policies in order to evaluate our proposed model:

Local resource allocation

Local resource allocation policy is used by autonomous cloud providers in first place for provisioning of on-demand and reserved VM instances. The policy is sub-divided in following two policies:

- Local Resource Allocation of isolated VM pools (LRAI)
- Local Resource Allocation of shared VM pools (LRAS)

LRAI ensures the high availability of VM instances and thus maintains the reputation of the cloud provider while keeping the perception of unlimited availability of resources in the pools as long as the on-demand and reserved pools are not fully occupied. However, further on-demand and reserved requests crossing the limited availability of the resources in the isolated pools are not be entertained under the local resource allocation of isolated VM pools [8],[17]. However, in case of LRAS, On-demand requests are entertained even if on-demand VM pools are fully utilized. This could be possible by using the spare capacity of reserved pool, if possible.

Outsourcing Oriented Resource Allocation (OORA)

This policy accommodate the ‘options’ concept proposed in our framework for cloud federation. In order to maintain a better reputation and to accommodate excessive on-demand requests from the clients, this policy can used by the provider. Hence, this policy solves the issue of denial of service in earlier policies. The idea behind is to accommodate on-demand requests using unutilized reserved space available in the pool of resources. However, before allocating the new on-demand request, a provider must buy an option for spare resource by paying the premium price to the option provider in the federation [8], [17]. This is to hedge against the risk of unavailability for the unutilized reserved capacity that is expected to be requested by the client meanwhile the on-demand request is entertained in its place. However, the cost charged by the provider of the spare resource upon exercising the option (strike price) should be less than the one charged by the provider to its client for revenue perspective. Also the quality metrics required by the cloud provider must meet accordingly with spare resource. The revenue in this case, can be calculated from the no. of outsourced VM by the provider.

Our simulation setup is based on QoS reference values obtained after extensive benchmarking of three cloud providers: Amazon, Rackspace and Google [24]. All of the above cloud providers charge users based on hourly rate; for instance Amazon charge cloud user $0.28 per hour for Linux based m3.xlarge on-demand instance while the same instance can be purchased for $0.1716 per hour in case of reserved instance for one year all up-front agreement. For simulation purpose, three data centers were assigned to every cloud providers with a capacity of 30 hosts each.

Table-2. Details of VM for Cloud Provider 1.

CP1			
VM Type	Configuration	QoS Characteristics	Cost/hr (USD)
Small	1 vCPU, 1.75GB RAM	Bandwidth < 120 Mbps, Disk processing and latency: very high with below average processing speed	0.04
Standard2	2vCPU, 7.5 GB RAM	Bandwidth < 120 Mbps, Disk processing and latency: very high with average processing speed	0.15
HighMem	2vCPU, 13GB RAM	Bandwidth < 250 Mbps, Disk processing and latency: very high with above average processing speed	0.2

Similar details were added for other data centers in the federation. Upon the job submission, user is requested to set job preferences to classify incoming tasks as CPU/ bandwidth/ storage/ memory intensive. If local cloud resources are insufficient to meet incoming job demands, cloud federation resources can be utilized based on user preferences.

VOL. 10, NO. 8, MAY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

For Discrete/ Continues Distribution workload pattern, which is assumed as usual workload pattern, we used Poisson distribution. For workload spikes, we adapted the model proposed by [25]. The model involves four time values t_0, t_1, t_2 and t_3. Starting from a normal traffic pattern, a flash pattern is generated at time t_0. During the first phase, known as Ramp-up, more and more resource requests are received which reaches to a maximum level at t_1. The process could be presented as:

$$\text{Ramp-up} = t_1 - t_0$$

At this stage, the resource requests remain steady for a time interval till time t_2. This phase is known as steady phase:

$$\text{Steady-phase} = t_2 - t_1$$

In this phase, job requests being completed are proportionally equal to new incoming job requests. Finally, resource surge gradually decreases and returns to normal rate at t_3. This phase is known as Ramp-down.

$$\text{Ramp-down} = t_3 - t_2$$

Based on the workload pattern studies for cloud computing [20], higher workload pattern was generated on weekdays as compared to lower number of requests during weekends. For Poisson distribution, Lambda (λ) was set to 10 for most of the times and some short random spikes were added to test the workload distribution of the system. However, number of incoming job requests during weekends were almost half of the normal load during the weekdays. Our final workload model is presented in the following figure:

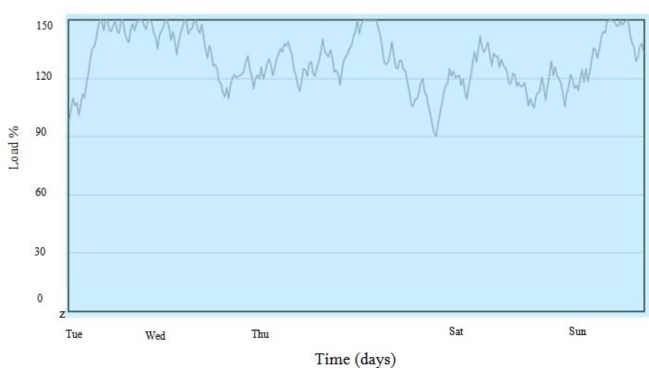


Figure-6. Proposed workload model for simulation.

We used Standard2 VM instance as a reference, charged at US$ 0.18 per hour with an additional cost of US$ 0.001 per GB storage and network operations. Reserved instances were set at US$ 0.13 per hour. In our simulation setup, the data center offers a capacity of 150 on-demand and 80 reserved VMs. Simulation data was collected for a period of one month. Average utilization of reserved instances was set at 55%. However, the load of on-demand instances varied over time; during weekends. Workload pattern was set around 1700 VM requests per day during weekdays while the same was reduced to 1150 during weekends. The configuration parameters for calculation of options are given as under:

- Contract time: 15 days
- Volatility: 20%
- Depreciation period: 4 years
- QoS: As presented in table 2
- Interest rate: 10%

In our first experiment, we used a load pattern with varying demands for one hour. The simulation was started with an on-demand VM utilization of 40% with a trend of gradually increasing VM demands. Load request was reached to 100% in 28 minutes. At this phase, OORA and LRAS started utilizing spare capacity from reserved VMs while on-demands request above 100% was rejected by LRAI. The load pattern is given as under:

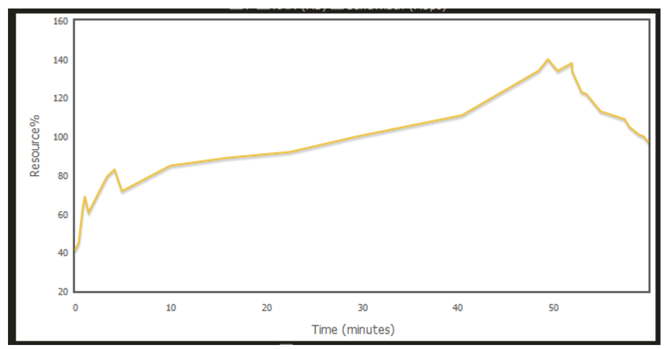


Figure-7. Resource demand with sharp spikes.

The LRAI policy was designed to fully utilize on-demand VMs from the pool. However, if the demand is reached above the available capacity, all incoming VM requests are not entertained. The resource usage pattern for this policy is given in the following figure:

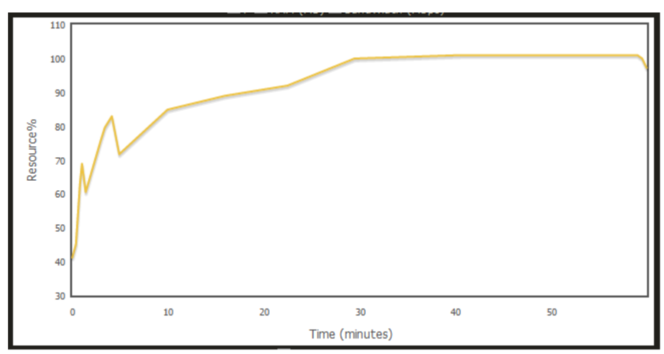


Figure-8. Overall resource utilization of LRAI policy.

For LRAS, the policy was to fully utilize on-demand VM pool for incoming on-demand VM requests.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

However, under peaks, if the capacity is not sufficient to meet incoming jobs, available capacity from reserved pool was also used to gain maximum economic benefits from fully resource utilization. During the simulation, reserved VM utilization was set at 62% on average and the spare capacity was utilized during peaks. However, since reserved VMs were fully utilized during a short spike, incoming reserved VMs were rejected and as a result, SLA penalty was paid to reserved VM users. The resource usage pattern is given as under:

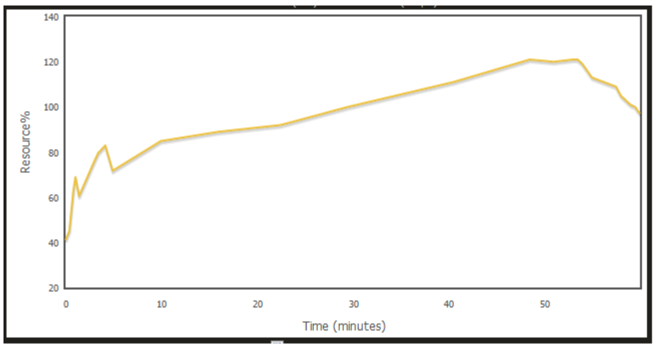


Figure-9. Resource utilization of LRAS policy.

Finally, OORA was aimed at utilizing both on-demand and reserved pools of VM instances for incoming on-demand VM requests while avoiding SLA violations. Before the simulation, cloud consumer was requested to provide feedback about QoS requirements, critical to job execution. These requirements were then matched with SLA requirements of the cloud provider. While outsourcing VMs, only those providers were considering who meet SLA requirements of the cloud consumer. During the peaks, when the reserved queue was under full utilization, incoming reserved VM requests were outsourced to the selected cloud provider(s) to avoid SLA violations as well as take economic efficiency of scalability. The resource usage pattern of OORA is given as under:

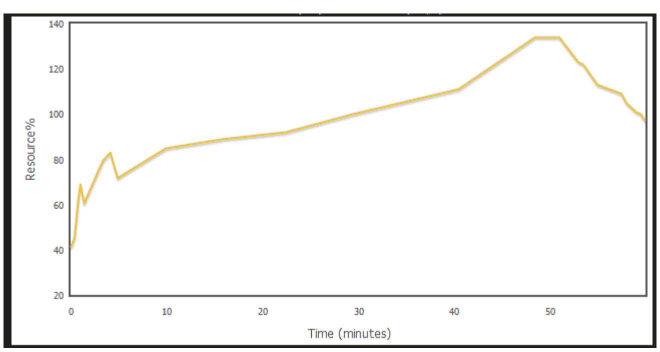
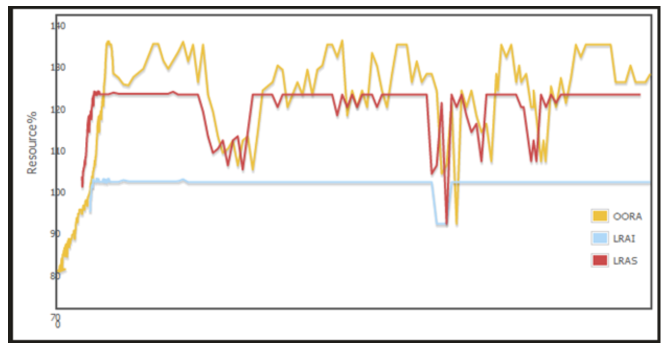


Figure-10. Resource utilization of OORA.

We used the workload pattern presented in figure 6 as our baseline experiment. Since it was not possible to simulate and record 30 days simulation data in one go, these workload patters were divided in 24 equal slots for individual days and statistics from these simulations were logged. At the end, the average data per week was calculated for all rounds of simulation. The overall load pattern of the simulation is presented in the following figure:



Initially the load patterns produced same results for all three policies. However, OORA showed tendencies to accept more workload during the peaks and gain better economic efficiency. Furthermore, it was also able to gain users' trust and avoid SLA violations.

As the statistics suggest, LRAI generated a net profit of US$ 9050, profit per node was US$ 301 but with higher rejection rate. Such a policy may discourage cloud users as job rejection rate (36%) is much higher and consumers may look forward to other cloud providers during peaks, resulting in loss of users' trust. LRAS performed better in terms of economic efficiency and resource utilization but at the rate of job rejection for reserved VMs. The policy resulted in a profit of US$ 9306 with an average profit of US$ 310 per node. A total of 5989 reserved VM job requests (19%) were rejected due to high workload and as a result, not only SLA penalty was paid but such policy may discourage regular cloud consumers to reserve virtual machines in the future.

Table-3. Simulation results of 30 days experiments.

Total Workload (30 days) => 46310 VM Requests

Cost of CPU (30 nodes) =>2580

Reserved Resource Utilization =>55%

Policy	OD utilized	OD rejected	Options bought	Options exercised	Options cost ($)	OD profit ($)	SLA penalty Cost ($)	Res. income ($)	Profit ($)
OORA	35997	4292	9178	6021	664	8999	-	4118	11254
LRAI	36399	9911	-	-	-	10002	360	3144	9306
LRAS	29638	16671	-	-	-	7512	-	4118	9050

OORA showed better statistics in all aspects; on the one hand, job rejection rate was marginal (9%) but also no SLA violation was encountered during the course of simulation. The scheme generated a profit of US$ 11254 with a profit of US$ 374 per node.

CONCLUSIONS AND FUTURE WORK

The main object of this research effort was to propose a framework for isolated cloud providers (consumers) in form of federated cloud environment where a broker could mediate the matters of concerns to the consumers for resource provisioning to the clients in peak times. Our framework was aimed at trading of spare resources among the cloud providers using advanced reservation mechanisms and keeping the price and quality both in consideration. We assessed cloud providers based on the services they provide and best-fit selection was made while purchasing options to meet service level objectives. Also the issue of VM placement across multiple clouds was addressed to provide a resource sharing platform where the objectives of avoiding resource rejection and efficient service level agreement management were incorporated in the proposed framework. Results indicate that our broker based cloud federation framework provides a win-win scenario for both cloud providers and consumers by avoiding job denial and achieving better SLA management and resource utilization. However, there are still further areas to explore in the relevant studies. Real experiments could be performed on the real data acquired by the providers (Google, Amazon and RackSpace) for better evaluation of the proposed framework. Also, Future workload prediction model strategies could be designed to predict the future workload of the consumer. The current policies designed are calculating the profit instantly on incoming requests and not able to predict the future requests and the corresponding outsourcing decisions.

REFERENCES

[1].Rajkumar Buyyaa, Chee Shin Yeoa, Srikumar Venugopala, James Broberga, Ivona Brandic (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems, Vol, 25(6): 599-616.

[2]. http://en.wikipedia.org/wiki/Cloud_computing, retrieved on August 1, 2014.

[3]. http://haizea.cs.uchicago.edu/manual/node9.html, retrieved on October 20, 2014.

[4]. Qanbari, S., Li, F., Dustdar, S., & Dai, T. S. Cloud Asset Pricing Tree (CAPT): Elastic Economic Model for Cloud Service Providers, CLOSER 2014, pp. 221-229.

[5].Tobias Kurze,et al.(2011). Cloud Federation. The Second International Conference on Cloud Computing, GRIDs, and Virtualization.

[6].Sharma, Bhanu, et al. "Pricing cloud compute commodities: a novel financial economic model." Proceedings of the12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid). 2012.

[7]. Kariya, Takeaki, and Regina Y. Liu. "Options, Futures and Other Derivatives." Asset Pricing. Springer US, 2003. 9-26.

[8].Toosi, Adel Nadjaran, Ruppa K. Thulasiram, and Rajkumar Buyya. "Financial option market model for federated cloud environments." Proceedings of the 2012 IEEE/ACM Fifth International Conference on Utility and Cloud Computing. IEEE Computer Society, 2012.

[9].Kitchenham, B.A. and Charters, S. “Guidelines for performing Systematic Literature Reviews in Software Engineering Technical Report EBSE-2007-01", 2007.

[10].Li, Wubin, et al. "Cost-Optimal Cloud Service Placement under Dynamic Pricing Schemes.", In Proceedings of the 2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing, pp. 187-194.

[11].Jian He, Yonggang Wen, Jianwei Huang, Di Wu, On the Cost-QoE Trade-off for Cloud-based Video Streaming under Amazon EC2’s Pricing Models, IEEE Transactions on Circuits and Systems for Video Technology, Volume:24 , Issue: 4 , pp. 669 - 680, 2013.

[12].D.Allenotor, R.K. Thulasiram, P.Thulasiram. A Financial Option based Grid Resources Pricing Model: Towards an Equilibrium between Service Quality for User and Profitability for Service Provider. Advances in Grid and Pervasive Computing, Lecture Notes in Computer Science Volume 5529, pp. 13-24, 2009.

[13].D.Allenotor, R.K. Thulasiram. Integrating a Financial Option based Model with GridSim for Pricing Grid Resources. 10th International Grid Computing Conference IEEE, 2009.

[14].Allenotor, David, and Ruppa K. Thulasiram. "Grid resources pricing: A novel financial option based quality of service-profit quasi-static equilibrium model." Proceedings of the 2008 9th IEEE/ACM international conference on grid computing. 2008.

[15].Hao Lu, Xing Wu, Wu Zhang, Optimal Pricing of Multi-model Hybrid System for PaaS Cloud Computing, IEEE International Conference on Cloud Computing and Service Computing 2012.

[16].Huang, He, Liqiang Wang, ByungChulTak, Long Wang, and Chunqiang Tang. "CAP^3: A Cloud Auto-Provisioning Framework for Parallel Processing Using On-Demand and Spot Instances." Sixth International Conference on Cloud Computing (CLOUD), pp. 228-235. IEEE, 2013.

[17].Goiri, Íñigo, Jordi Guitart, and Jordi Torres. "Economic model of a Cloud provider operating in a federated Cloud." Information Systems Frontiers 14.4 (2012): 827-843.

[18]. Amazon Customer Agreement, http://aws.amazon.com/agreement/, retrieved on 10 January 2015.

[19].J. Varia, “Best practices in architecting cloud applications in the AWS cloud,” in Cloud Computing: Principles and Paradigms, R. Buyya, J. Broberg, and A. Goscinski, Eds. Wiley Press, 2011, ch. 18, pp. 459–490.

[20].Toosi, Adel Nadjaran, On the Economics of Infrastructure as a Service Cloud Providers: Pricing, Markets, and Profit Maximization, PhD thesis, 2014.

[21]. http://141.223.82.74/papers/APNOMS/02/sla-monitoring.pdf

[22].https://www.academia.edu/1768831/Service_Level_Service_Level_Agreements_Coordination_and_Monitoring_of_Actors_in_Smart_Grid.

[23] Cloudsim, http://www.cloudbus.org/cloudsim/

[24] S Ullah et al. (2015). Performance Benchmarking for Infrastructure-as-a-Service Cloud Providers. Manuscript submitted for publication.

[25] Bodik, Peter, et al. "Characterizing, modeling, and generating workload spikes for stateful services." Proceedings of the 1st ACM symposium on Cloud computing. ACM, 2010.

[26] Voorsluys, W., Garg, S. K., & Buyya, R. (2011). Provisioning spot market cloud resources to create cost-effective virtual clusters. Algorithms and Architectures for Parallel Processing, Lecture Notes in Computer Science Volume 7016, 2011, pp. 395-408.