Financial Option Market Model for Federated Cloud Environments

Adel Nadjaran Toosi*, Ruppa K. Thulasiram† and Rajkumar Buyya*
*****_**Clou**d Computing and **D**istributed **S**ystems (CLOUDS) Laboratory_
Department of Computing and Information Systems
The University of Melbourne, Australia
Email: adeln@student.unimelb.edu.au, rbuyya@unimelb.edu.au
†Department of Computer Science
The University of Manitoba, Canada
Email: tulsi@cs.umanitoba.ca

***Abstract*—Pay-per-use service by Cloud service providers has attracted customers in the recent past and is still evolving. Since the resources being dealt within Clouds are non-storable and the physical resources need to be replaced very often, pricing the service in a way that would return profit on the initial capital investments to the service providers has been a major issue. Moreover, to maintain Quality of Service (QoS) to customers who reserve the resources in advance and may or may not be using the resources at a future date makes the resources wasted, if not allocated to other on-demand users. Therefore, a need for a mechanism to guarantee the resources to reserved users whenever they need them, while keeping the resources busy all the time is in very high demand. The concept of federation of Cloud service providers has been proposed in the past wherein resources are traded between the providers whenever need arises. We propose a financial option based Cloud resources pricing model to address the above situation. This model allows a provider to hedge the critical and risky situation of reserved users requesting the resources while all the resources have been allocated to other users, by trading (buying or outsourcing) resources from other service providers in the Cloud federation. We show that using financial option based contracts between Cloud providers in a Cloud federation, providers are able to enhance profit and acquire the needed resources at any given time. It would also help creating a trust and goodwill from the clients on the Cloud service providers by less number of QoS violation.**

***Keywords*-Financial Option; Market Model; Reservation; Reserved Instances; Cloud Federation;**

I. INTRODUCTION

The term Cloud computing has become widely known in both industry and academia [1], [2]. It focuses on offering application, platforms, or IT infrastructures (e.g. processing power, storage and networking). Among the different methods to deliver Cloud services, Infrastructure as a Service (IaaS) allows Cloud providers to offer storage and computational service in the form of Virtual Machine (VM) instances. IaaS Cloud providers, called Cloud provider or provider from now onward, usually offer customers two well-known payment plans: *reservation* and *on-demand*. Amazon EC2[1] and GoGrid[2], for example, provide reservation and on-demand plans of the infrastructure services.

Customers pay in advance to reserve the instances for the possible future usage and in exchange receive a significant discount on the charge for running VMs in the reserved capacity. Moreover, customers receive higher availability of the service for reserved than on-demand instances. On the one hand, reservation plans allow the customers to acquire resources in cheaper price and higher availability than that of on-demand plans. On the other hand, it helps providers to attain more efficient resource management and procurement. In addition reservation can guarantee cash flow even if the reserved resources are not fully utilized by the customers.

Since Cloud applications such as web applications experience huge and unpredictable variation in the load over time, defining the required amount of instances to cope with the load experienced in a given moment is a challenging task. If the load was known beforehand, users could reserve the required amount of instances, which is cheaper than acquiring on-demand instances. However, as loads are unpredictable and variable, users have to combine reserved instances with on-demand instances for the situations in which the former is not enough [3]. This provides a balance between cost and utilization of the resources.

The pattern of utilization at user side causes reserved instances not to be deployed at all times. This offers providers the opportunity to explore this non-storable underutilized capacity for additional cash flow by releasing them to the on-demand requests. It is important for IaaS providers to maximize their profit. Therefore, if the unreserved part of the data center experiences high utilization, providers are able to accommodate on-demand requests on the underutilized reserved capacity of the data center. However, providers are liable to provide guaranteed availability for the reserved requests according to the service level agreement (SLA). Consequently, providers face a risk of SLA violation by using the reserved capacity for accommodating on-demand requests.

Cloud cooperation is a possible solution in order to hedge against the mentioned risks by letting providers increase their resources dynamically [4], [5]. Recent works demon-

---

[1] http://www.aws.amazon.con/ec2/

[2] http://www.gogrid.com/

strate that federation of providers and interoperability between Clouds to trade resources in a market helps providers to enhance their profit, resource utilization, and QoS [6], [7]. The use of shared pool of physical nodes for on-demand and reserved instances along with outsourcing requests mitigate the risk of violation of the QoS mentioned in the SLA for reserved instances substantially. Moreover, it helps providers to increase their profit. But, the provider faces a risk of being unable to acquire required resources in the market. Essentially, they may end up short selling resources without having a good knowledge of usage loads and hence violating the QoS. Furthermore, according to the efficient market hypothesis in economic markets, providers can not precisely predict the future price variations in the federation market using past price history [8], [9].

In this paper, a financial option-based market model is introduced for a federation of Cloud providers, which helps providers increase their profit and mitigate the risks (risks of violating QoS or paying extra money). A financial option [10] is a contract for a future transaction between two parties: *holder* and *seller* of the contract. A financial option gives the holder the right, but not the obligation, to buy (or to sell) an underlying asset at a certain price, called the *strike price* (exercise price), within a certain period of time, called *maturity date* (expiration date). The seller is obligated to fulfill the transaction. As a compensation the seller collects an upfront payment at the beginning of the contract, called *premium*. In our proposed framework model, a provider buys option contracts as a backup capacity for the reserved resources used by on-demand instances, to gain the right to acquire resource from the seller provider, as need arises. Since the seller is obligated to fulfill the request, risk of not acquiring resources is removed. Moreover, buying option contract protects provider against high variation of the market price. In summary, this paper has the following main contributions:

1) A financial option-based market model in the presence of the Cloud federation to help providers to manage their reserved capacity and achieve higher QoS guarantee.
2) Evaluation of the proposed model to show its effectiveness in increasing provider's profit without imposing any SLA violation.

The remainder part of the paper is organized as follows: In the next section we give an overview of the related work. The system model and problem definition are identified in Section III. Our proposed option model including the parameters setting and pricing mechanism is explored in Section IV. In Section V, the base-line policies and the proposed policy based on our model is introduced. A detailed discussion on workload and simulation setup, performance metrics, and experimental results is given in Section VI. Finally, Section VII presents the conclusions and the future work.

II. RELATED WORK

Resource provisioning for IaaS Cloud providers is a challenging issue because of the high variability in load over the time. Providers must be able to dynamically increase the available resources to serve requests [6]. In order to enable such scenario, coordination between providers has to be achieved, possibly through the establishment of a Cloud federation. In recent years, different platforms for Cloud federation have been proposed in literatures [4], [5], [11]. Economic aspects of the Cloud federation including motivations and incentives for parties joining the federation have been investigated by several works [6], [7].

Works related to systems for market-making such as works by Song et al. [12], Mihailescu and Teo [13], Gomes et al. [14], and Vanmechelen et al. [15] concern about mechanisms for creating markets and trading resources. In the current work, a financial option-based market model has been introduced for a federation of Cloud providers. An introduction to the foundations and basics of financial option theory can be found in [10].

The authors in [16] propose a model based on financial option theory to price Grid resources. They use option for pricing Grid resources in order to maintain equilibrium between service satisfaction of Grid users and profitability of the service providers. They do not propose a model to sell and buy options as we do and their model proposed for a single resource provider environment. In our study, we consider multiple providers in a Cloud where option contracts are traded between service providers. Moreover, the risk factors that they are concerned with are different from the risks investigated in this paper.

Another work devoted to option theory in resource allocation for Clouds, proposes an approach based on the option theory to minimize cost and mitigate the risk for Cloud users [8]. They introduce a novel pricing scheme based on the option that Cloud providers should provide for their own customers. Using option plan, customers can reduce the cost of using IaaS Cloud provider resources. Our work, on the other hand, mainly aims to increase profit and mitigate risks for providers, which leads to better QoS for the customers.

Meinl and Neumann [17] analyze the use of real options in a contract market, to economically manage resource reservation in distributed IT environments. In fact, they use option as a contract to perform reservation for time and budget sensitive customers. Grid consumers want to minimize expenses, whereas Grid providers want to maximize their return on investment. Our work is similar as we also focus on reservation, but we use option as a hedging mechanism for the reserved capacity to enhance provider profit. Besides, our reservation scheme is also different. They consider reservation for Grid jobs with deadline and budget while our reservation scheme is like what IaaS Cloud providers offer.

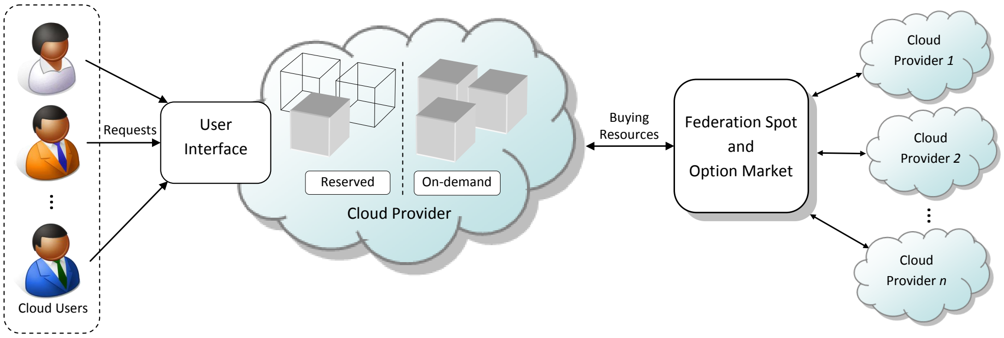

Figure 1. Model elements and architecture

In [18], Bossenbroek et. al investigate the application of option contracts in the context of a market for Grid resources, in order to deal with price volatility. They assess the performance of three classic hedging strategies for buying options in this regard. In order to define the underlying asset, the concept of *leases* is introduced. These correspond to a right to use a Grid resource for a fixed time period (e.g. one hour). The size of a task is expressed in a multiple of such leases, and it is assumed that the task's load is entirely divisible over such leases. The model is therefore not directly applicable to the cloud computing domain considered in this work.

Cloud providers offer customers reservation (e.g. prepaid) and on-demand plan (e.g. pay per use). There are always incentives for both Cloud providers and customers to use reservation. For example, reservation may result in better capacity planning, guaranteed cash flow for providers, and availability of resources and discount on the usage of such resources for customers. However, it is important for the customers to optimize the amount of resources that they reserve to reduce the cost. Reservation is a challenging issue, since it should be done in advance when there is an uncertainty about the actual future demand. Under-provisioning (Reserving less) and over-provisioning (Reserving more) of reserved instances may result in extra costs for the customers. Authors in [3] propose an algorithm to minimize total cost of resource provisioning and avoid over-provisioning and under-provisioning using reserved instances.

On the other side, Cloud providers supporting reservation should answer the question of how to allocate resources between reserved and other types of requests to maximize their revenue. A sample solution for this can be found in [19].

III. THE SYSTEM MODEL

In this section, we outline the system model including the markets, market participants, their motivations, and all the corresponding parameters of the model (Figure 1).

Cloud users (IaaS Cloud customers) submit their IT infrastructure requirements in the form of VM requests to the *Cloud provider* (Figure 1). Such a request submission may lead to acquisition of VM instances for a certain amount of time by the customer. Since customers decide when to terminate the instances, the Cloud provider does not have a priori knowledge of the holding time of the instances. Requests can be submitted either for *reserved* or *on-demand* service and charges will be applied accordingly.

1) *On-demand plan*: On-demand plan allows customers to pay for compute capacity by its usage without long-term commitment. Price is calculated at a fixed rate per usage time, e.g. hourly, from the time an instance is launched until it is terminated. If the provider possesses enough resources, resource provisioning for VM requests is done, otherwise the request is rejected by the provider. After instantiation of VMs, customers can retain machines as long as they require them.
2) *Reserved plan*: In this plan, customers pay an upfront fee, called reservation fee, and in return receive a discount on the usage for the VMs. Reserved plan also assures that the reserved capacity is always available when it is required.

We believe that customers do not always fully utilize the reserved capacity in the reservation lifetime. Partial utilization of the reserved capacity still has benefit for customers. For example Amazon EC2 users gain economic advantage of using *Reserved* Instances in comparison with *On-Demand* Instances, even if they can utilize only slightly more than 8% of the reserved capacity in a 3-year contract[3].

The *Cloud provider* offers its resources for each plan based on the fixed price (Figure 1). The Cloud provider offers best-effort and high availability of the service for on-demand and reserved instances respectively. Allocating data center capacity (physical nodes) to reserved and on-demand instances in order to meet the QoS of each plan is performed by the provider. Suppose that the provider has a data center with defined capacity and it is able to accommodate maximum n VM instances of similar types simultaneously. Two different strategies can be assumed to structure such a

---

[3] http://aws.amazon.com/ec2/reserved-instances/

system to support aforementioned plans [19]:

1) *Isolated pools*: In this strategy, two different pools of servers (nodes in data center) for instances of each plan is considered in isolation of each other. The number of nodes for reserved instances, in this case, is defined according to the total number of reserved VMs by customers. If the entire reservation size is r instances, the on-demand pool is capable of accommodating $n - r$ instances at most.
2) *Shared pool*: In the shared pool strategy, on-demand VMs are offered to use physical nodes of the reserved capacity if on-demand capacity is fully utilized by on-demand requests. The opposite scenario, using on-demand capacity by reserved requests is not conceivable, since the reserved capacity is always as large as the required capacity for the reserved VM requests. In the shared pool strategy, If the data center maximum capacity in unit of VMs of the similar type is n, total reservation size is $r(r \leq n)$, and m reserved VMs are running $(m \leq r)$ then accommodating $n-m$ on-demand requests is possible while m remains unchanged.

Given that reserved instances and on-demand instances are just different in pricing and they function identically during execution, there is no technical barrier to set up the shared pool strategy. However, shared pool strategy suffers from the risk of violating availability of the reserved instances.

A *Cloud federation* allows providers to trade their resources through Federation Level Agreements (FLA) [7]. In this paradigm, providers aim to overcome the resource limitation by buying resources from the market. Underutilized providers sell their resources in this market usually at cheaper prices compared to what they would charge their own customer, in order to avoid wasting their non-storable compute resources (e.g. *SpotCloud*[4]).

It is worth mentioning that our model can be applied to a hybrid Cloud scenario in which a private Cloud provider buys option contracts from a large public Cloud provider.

The main element in our model is the *federation spot market* in which a group of federated Clouds trade their resources with each other (Figure 1). Different types of underlying market mechanism can be considered for the spot market, such as combinatorial double auctions, commodity exchanges, reverse Dutch auctions, and etc. The main focus of the current work is to build an *option market* on top of the spot market. Due to general nature of our proposed model, the option market is modeled independently from the underlying spot market mechanism. The only outcome of the spot market which is required by the model is the spot price at which resources are offered. Therefore, in our setting we do not specify a particular spot market mechanism, and the scheme does not directly influence the model. Readers are referred to our previous work for an example of a federation spot market mechanism [7] or a similar work [6]. In the current work, we assume that the IaaS Cloud provider is able to buy resources from the spot market at the current spot price. Strategies regarding selling resources to the market will be explored as an extension of this work.

The Cloud provider in our model attempts to increase its revenue by using a shared pool strategy. This avoids wasting the underutilized reserved capacity by serving excess on-demand requests on that capacity. However, the IaaS Cloud provider faces the risk of violating availability for the reserved requests. Thus joining the federation spot market could mitigate the risk by allowing the provider to outsource reserved and on-demand requests. However, a Cloud provider participating in the federation spot market could still bear two risks, namely

1) the risk of price fluctuation in the market and high cost of outsourcing, and
2) the risk of not being able to acquire resources, which leads to rejection of the reserved requests.

In order to hedge against above risks, we propose a market model based on the financial option on top of the spot market for a federation of Cloud providers. Using our model, the provider is able to enhance its profit by deploying a shared pool of physical nodes for on-demand and reserved requests. Moreover, the provider ensures the availability of the reserved instances and avoids buying resources at a price that is higher than the one charged to its own customers.

In the current model, each time the provider accommodates an on-demand request in the reserved capacity due to lack of space in the on-demand pool, it buys an option to hedge the situation of running short of resources. The option is exercised (i.e. the requests are outsourced to other Cloud providers in the federation) if the reserved request arrives and the provider does not have enough resources to serve it locally. The important advantage of buying options in comparison to other future agreements is that it gives the provider the right to buy resources (outsource requests) in the future and that the provider is not obligated to do so. Therefore, if the Cloud client does not request the reserved instances, the provider will simply let the contract expire without responsibility to buy unnecessary resources. The only cost for providers in such an arrangement is the premium paid at the beginning of the contract. This cost, however, can translate into trust and goodwill by the clients on the provider.

In our model, providers transfer the risk of violating SLAs to other providers by buying option contracts and paying option premium. Therefore, Sellers of the option contracts must consider the trade-off between the risk and expected profit. However, the scope of the current work is limited to buyer's strategies for purchasing required options. Strategies regarding selling options requires further attention. Interested readers are referred to the work of Markowitz et al.

---

[4] http://www.spotcloud.com/

on decisions under uncertainty in financial markets [20], and the work of Michalk et al. on the translation of the model for Cloud Service providers [21].

A detailed discussion of the option market including pricing, buying mechanism and exercising options is provided in the next section.

IV. THE OPTION MARKET

The main contribution of this work is to propose a financial option-based market mechanism for a Cloud federation. The main element in such a market is the *option*. There are two types of option: *call* and *put*. A call option gives its holder the right to buy the underlying asset at a specific price (strike price) by a certain time (expiration date) [10]. A put option gives the holder the right to sell the asset at a specific price over a given period of time. Providers with large amounts of physical resources may buy put options that will give them the right to sell resources at their will. However, we do not consider the use of put option in this work.

Suppose that a market participant purchases a call option for \$2 with strike price of \$28 for expiration date in two months. Within two months, the spot price goes to \$35, in this case, he/she exercises the option and gains the advantage of $(35 - 28) \ - \ 2 = \$5$. If the spot price stays below the strike price, the option holder might buy at the spot price and allow the option to be expired. In this case, the \$2 premium paid at the beginning of the option contract is lost.

An option contract is defined by a tuple (P, K, T), where P is the price of buying the option, K is the strike price, and T is the expiration date. In our model, the provider buys an option, each time it accommodates an on-demand request in the reserved capacity. Terms of the option contract (P, K and T) need to be determined at the time the contract is signed. In the following paragraphs, the way we set the terms of the contract is explained.

Given that the price per time unit for the reserved instances is R, the provider buys an option with a strike price lower than R to secure its future profitability. It means that the provider assures that the price it pays to outsource a reserved request is always lower or equal to the price it charges its own customers, i.e. R. As long as the spot market price, S, is lower than R, the provider submits a request for buying an option with strike price $K = S$, otherwise $K = R$.

The provider is oblivious to the duration of the VM and its future load, so we consider T as a fixed value, e.g. one month. We investigate the impact of the time to maturity of the options on the model in our study. Optimization strategies regarding buying option with the best expiration date requires load prediction strategies. It can be considered as an extension of this work.

The option can either be exercised at expiration date (*European option*), or any time during its life (*American option*). Since the provider buys an option to hedge the risks for reserved requests and it needs to exercise the option any time in the future according to the load and upcoming reserved requests, the American call option is the most appropriate for our work.

When the provider desires to purchase an option, it needs to pay the option price or the premium to the seller. The value of an option (option price or premium) can be estimated using a variety of quantitative techniques based on the concept of risk neutral pricing [22], [23].

A useful and popular technique for pricing an option involves constructing the price movement in a structured manner known as binomial lattice or tree [24]. The binomial tree represents different possible paths that might be followed by the underlying asset price over the life of the option. In our model, the underlying asset is the available resource in the federation market.

Consider the current spot market price is S_0. S_0 goes to S_0u with probability of p and to S_0d with probability $1 - p$ at each time step ΔT. Let $T = n.\Delta T$, where T is the option expiration date, then a lattice of spot price movement for $n = 3$ is presented in Figure 2. The value of the option can be evaluated for each point at the leaf nodes of tree (time T). The value of the option at starting node can be calculated through a procedure known as backward induction. A call option is worth $max(S_T - K, 0)$, where S_T is the spot market price for underlying asset at time T.

Assuming risk neutral world, the value at each node at time $T - \Delta T$ is computed according to the expected payoff value at time T and the risk-free interest rate, r, for the time period ΔT. In this study we assumed $r = 0$. Going backward using the above procedure, the option value, P, can be obtained at time zero. American call option that pays no dividend is never exercised early. Consequently, the following procedure is valid for both American and European call options [10].

Factors u, d and p play crucial role in option pricing. All of the aforementioned parameters can be calculated according to a parameter called *volatility*. The volatility, σ, in stock market is a measure of uncertainty about the returns provided by the stock and future stock price. There is a wealth

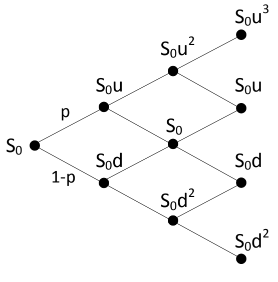

Figure 2. Binomial tree for option pricing

of literature on calculating σ in finance community [25]. One common method to obtain the volatility is by using the history of the stock price movements. We used the method provided in [10] to estimate volatility according to the historical data. Choosing a proper size for the time frame of the historical data to calculate σ is non-trivial. In this paper, we set this value equal to the maturity time for the option contract. u, d, and p can be calculated according to σ as:

$$p = \frac{1-d}{u-d}, \;\; u = e^{\sigma\sqrt{\Delta T}}, \;\; d = e^{-\sigma\sqrt{\Delta T}}$$

V. POLICIES

Three different policies to evaluate our model including two base-line policies and a policy using our option market model are described in this section. We first present base-line polices based on an isolated pool of servers with and without federation respectively, and then the federation option-market enabled policy using a shared pool of physical nodes is proposed.

A. Baseline In-house Isolated Pool Policy (IIP)

The first baseline policy is the simplest policy in which the provider works independently, without participating in the federation. Moreover, the provider in this policy uses isolated pools of physical nodes for on-demand and reserved instances in order to guarantee high availability of reserved requests. As a result this policy rejects extra on-demand requests even when the reserved capacity of the data center is not fully utilized.

B. Baseline Federated Isolated Pool Policy (FIP)

In this policy, we assume that the provider is able to access the federation spot market. Resources are bought at the spot price from the Cloud federation market to outsource on-demand requests if the provider is not able to serve them locally. To be always cost-effective, if the spot price in the federation market is higher than the local on-demand price then the provider rejects the on-demand request. In this policy, the pool of physical nodes for reserved and on-demand instances are isolated to prevent rejection of reserved requests.

C. Federated Shared Pool Option-Enabled Policy (FSPO)

The third policy exploit the potential of our proposed option market model to hedge against risk of using the shared pool strategy. In this policy, the provider accommodates excess on-demand requests in the underutilized reserved capacity. To facilitate this, the provider buys an option whenever he accommodates on-demand requests in the reserved capacity. Consequently, if a reserved request comes in and the provider is not able to serve it locally, the option is exercised and the reserved request is outsourced at the strike price of the option.

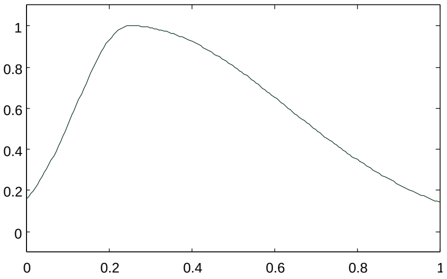

Figure 3. Combination of two Gaussian Functions

It is necessary to mention that policies with the shared pool strategy without considering our proposed option model (e.g. In-house Shared Pool or Federated Shared Pool) are not taken into account here. Those policies might lead to rejection of reserved requests and QoS violation. However, the number of rejections occurring for the reserved requests for those policies are reported in our experimental evaluation.

VI. PERFORMANCE EVALUATION

A. Experimental Setup

1) Workload setup: Due to the lack of publicly available traces of real-world IaaS Cloud requests, we created a synthetic workload model to generate VM requests for an IaaS Cloud.

Our workload model complies with previously reported workloads for IaaS providers in the literature [6], [26]. Generating a VM request requires a pair (S, D), where S is the arrival time of the request and D is the holding time of the instance by the customer.

In order to model the holding time of the instance by users, D is taken to be a Pareto distributed random variable, with shape parameter $\alpha = 1.1$ and location parameter $\beta = 1$ [26]. The Probability Density Function (PDF) of the Pareto distribution is

$$f(x) = \begin{cases} \alpha\beta^{\alpha}/x^{\alpha+1} & \text{for } x \geq \beta \\ 0 & \text{for } x < \beta \end{cases}. \tag{1}$$

The value of the random variable D represents the holding time of the VM by the user in the scale of hours.

Arrival times of the requests are generated as follows. Given that our workload follows daily pattern, the combination of two Gaussian functions in a range of $[0, 1]$ with the given equation has been chosen. A Gaussian function is defined by Equation 2.

$$f(x) = e^{-\frac{(x-a)^2}{2b^2}} \tag{2}$$

The function is shown in Figure 3. As it can be seen in Figure 3, the function can be fit into two-parts for two sets of values of a and b. With $a = 0.13$ and $b = 0.25$ the first part of the curve can be fit; and with $a = 0.38$ and $b = 0.13$, the second part of the curve is fit. The resulting shape in Figure 3

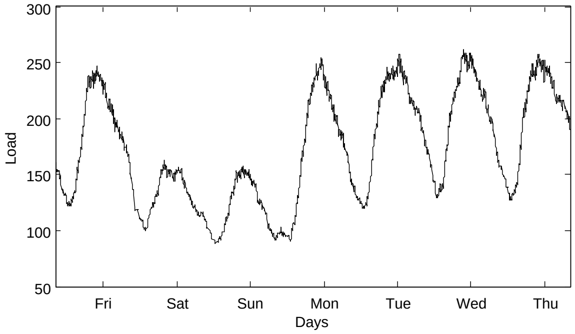

Figure 4. Generated workload during a week

has been divided in 24 buckets of equal width, where buckets are related to a specific hour in a day and they starts at 4:00 AM. Then, the proportion of the number of requests arriving every hour of the day is calculated according to the ratio of the area in a bucket to the total area under the curve. Afterwards, we generate a uniformly distributed arrival time for requests in every hour. In order to create the weekly patterns for the workload, we divided days per week in two parts: weekdays and weekend (Saturday and Sunday) with a lower number of requests for the weekend (reduction of 50%). In our simulation the number of requests per day in weekdays and weekends vary to increase or decrease the data center load. An example of load generated via our workload model for one week is depicted in Figure 4.

We intend to study the behavior of the financial option-based market model in different situations of the load. For this purpose, our workload model is applied to generate reserved and on-demand requests separately with different values for weekdays and weekends.

2) Simulation Setup: The experiments presented in this work were developed using the CloudSim [27], a discrete-event Cloud simulator. The simulated scenario is created according to the model in Section III.

Considering the components in Figure 1, the Cloud provider in our simulation receives the VM requests generated by the described workload model in the previous subsection. Requests are either on-demand or reserved. The VM configuration is inspired by Amazon Elastic Compute Cloud (Amazon EC2) small instances[5]. The pricing is also adopted from the Amazon EC2 price of small instances in the US east region for on-demand and medium utilization reserved instances[6]. The provider charges their customers based on hourly usage, at the cost of \$0.085 and \$0.030 per hour for on-demand and reserved VMs, respectively. We do not take into account the premium fee for the reservation as it only adds a constant value to the provider revenue.

The provider capacity is set equal to the maximum number of simultaneously runnable VMs. In our simulation, the provider capacity is set to 200 VMs. For the sake of better analysis, the reserved capacity of the data center is considered as steady constant value for the whole simulation. The reserved capacity is set to 100 VMs in all the experiments.

Federation spot market prices are generated according to the statistical model presented in [28]. The model fits well with the price fluctuation of the spot instances in the Amazon EC2 spot market. The model is applied in our work because of two main reasons. First, correlation between generated spot prices and the price of on-demand and reserved instances makes the evaluation of our model more significant. Second, the statistical model provides more flexibility to investigate our proposed market mechanism in comparison to the spot price traces of Amazon as we can modify the parameters.

Option pricing is done by the previously described method in Section IV. The depth of the binomial tree to calculate the option value is 30. Volatility, as we mentioned earlier, is calculated according to the examination of the past spot prices leading to the contract time with the same length of option maturity time.

In our experiments, the length of a simulation period is 6 months. Each experiment is carried out 30 times and the mean value of the results are reported.

B. Performance Metrics

Two different performance metrics for evaluation purposes have been used in our experiments. Our model has been developed to increase the provider's profit while availability of the reserved instance remains high. Therefore, profit has been selected as a first measure to evaluate the model. Provider's profit calculated according to the following equation:

$$P = R_O + R_R - C_{out,O} - C_{out,R} - C_{option} - C_p \ , \quad (3)$$

where R_O and R_R are the revenue of the on-demand and reserved instances. $C_{out,O}$ and $C_{out,R}$ are cost of buying resources from another provider to outsource on-demand and reserved instances, respectively. All R_O, R_R, $C_{out,O}$ and $C_{out,R}$ are computed by the summation of the running expenses for the served VMs of each type during the experiment's period. The running expense of a VM is calculated according to the price per hour multiplied by the instance-hours consumed for each VM, from the time an instance is started until it is terminated. Each partial instance-hour is considered as a full hour. The reservation fee for the reserved instances is not considered here as it imposes a same fixed value in calculation of the R_R for all the policies. C_{option} is the total cost for buying options in the federation (premium costs) incurred by the provider. Finally, C_P is the provider's cost, which includes the operational cost of the data center

---

[5] Small instance: 1 CPU core, 1.7 GB RAM, 1 EC2 Compute Unit, and 160 GB of local storage

[6] http://aws.amazon.com/ec2/pricing/

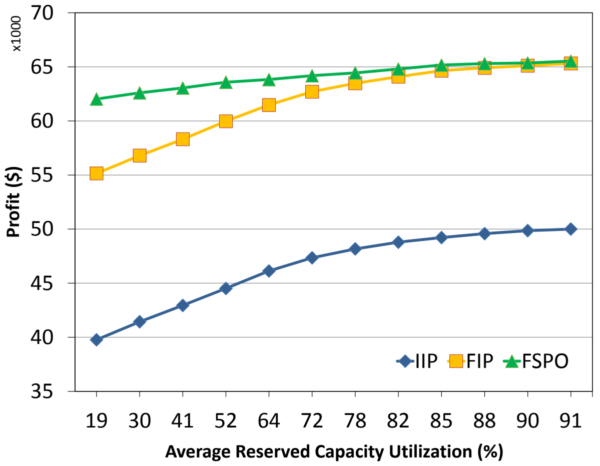

Figure 5. Impact of the reserved capacity utilization on provider's profit with different policies.

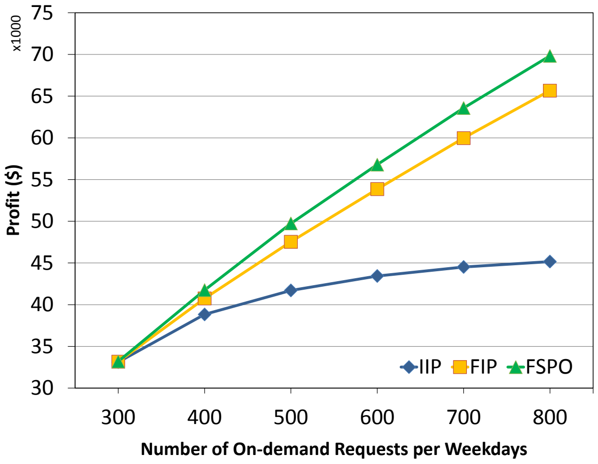

Figure 6. Impact of the number of on-demand requests on provider's profit with different policies.

(i.e., hardware and software acquisition, staff salary, power consumption, cooling costs, physical space, amortization of facilities, etc.). We ignore C_p in our experiments as it imposes a constant value to all policies.

The second metric is the number of rejected reserved requests, which shows the number QoS violations for the reserved instances.

C. Experimental Results

In the first set of experiments we evaluate the profitability of the model by changing the loads of the on-demand and reserved requests. First, all parameters of the system were fixed and the load of the reserved instances was increased (Figure 5). The capacity of the data center is 200 VMs, the reserved capacity is 100 VMs, the maturity time of the options is 30 days, and the number of on-demand requests per weekdays and weekends is 700 and 350, respectively.

Average utilization of the reserved capacity in the data center was changed by increasing the number of reserved requests. As shown in Figure 5, the generated profit for the IIP and FIP, which do not use the underutilized capacity of the reservation for accommodating on-demand VMs, rises smoothly with the increment of the reserved capacity utilization. The difference between IIP and FIP is caused by the outsourcing of the excess on-demand requests in the FIP. On the other hand, as the reserved capacity utilization is increased, FSPO experiences less growth in profit in comparison to the other policies as the underutilized capacity for in-house accommodation of the on-demand requests decreases. Eventually, FSPO generates the same profit as the FIP policy.

We use the same configuration of the previous experiment while utilization of the reserved capacity is fixed at 52% and the number of on-demand requests is varied from 300 to 800 per day for weekdays and half of that for weekends. All policies generate the same profit at very low on-demand requests because outsourcing on-demand requests or accommodating them in the reserved capacity is not required. The IIP policy that does not benefit from the potential of outsourcing, generates less profit in comparison to the other policies. The FSPO is the most dominant policy as it utilizes both federation and the underutilized reserved capacity.

The objective of the third experiment is to examine the effects that the volatility of market prices has on the profit making opportunity from the proposed model. Since the real data set from Amazon EC2 has low price volatility, we increased the standard deviation of the proposed distributions for spot price modeling by Javadi et al. [28] to generate a highly volatile prices in the spot market, and prices below \$0.025 were ignored. Simulation parameters for this experiment were set as follows. The capacity of the data center is 200 VMs, the reserved capacity is 100 VMs, the maturity time of the options is 30 days, and the number of on-demand requests per weekdays and weekends is 700 and 350, respectively, and the utilization of the reserved capacity is 64%. As shown in Figure 7, FSPO is more resilient to the higher degree of volatility in comparison to FIP, as it secures the outsourcing cost at prices below the reserved instances price. However, when the spot price fluctuates more, the provider has to buy options in a higher price to hedge against price variation.

We added a shared pool strategy to the baseline policies. The number of rejected reserved instances reported in Table I. It demonstrates how the option model helps providers to hedge against the risk of QoS violations for the reserved instances. We considered that resources are not available in the spot market of the federation, if the spot price is higher that \$0.085. Higher risk of unavailability in the market will cause more QoS violations for the *Federated Shared Pool* policy. The Federated Shared Pool policy shows a small

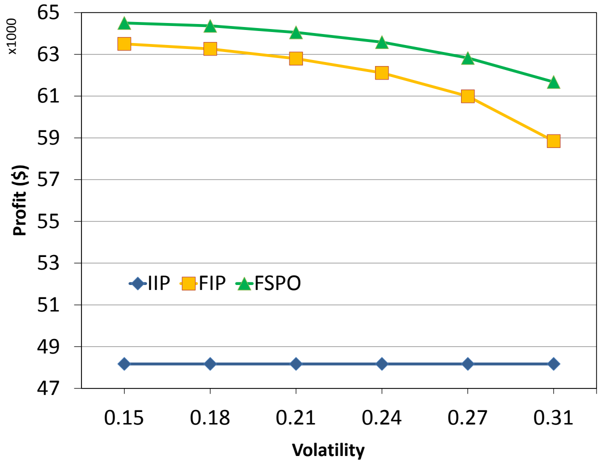

Figure 7. Impact of price volatility on behaviors of policies.

Table I
NUMBER OF ON-DEMAND (O), RESERVED (R), REJECTED ON-DEMAND (RO), REJECTED RESERVED (RR), OUTSOURCED ON-DEMAND (OO), AND OUTSOURCED RESERVED(OR) REQUESTS FOR THE PROVIDER WITH POLICIES.

Policy	O	R	RO	RR	OO	OR
IIP	106340	36590	42450	**0**	0	0
In-house+ Shared Pool	106340	36590	19188	**6636**	0	0
FIP	106340	36590	219	**0**	42230	0
Federated+ Shared Pool	106340	36590	111	**38**	19078	6598
FSPO	106340	36590	111	**0**	19078	6636

number of rejections in our experiments, as the model we used here to generate spot prices according to the Amazon EC2 spot market rarely generates prices higher than the on-demand price.

We also investigate the impact of the maturity time of the option in profitability of the FSPO policy. The maturity time for the option contract was varied from 7 to 90 days and the generated profit was reported in Table II. The configuration of this experiment is the same as the first experiment when the utilization of the reserved capacity is 52%. The results show that the maturity time of the option contracts does not have a significant impact on the gained profit because when the maturity time rises, the number of bought option contracts falls.

VII. CONCLUSIONS

Table II
IMPACT OF OPTION MATURITY TIME ON PROVIDER'S PROFIT USING FSPO.

Maturity time (Day)	Bought Option	Profit ($)
7	3086	63588
10	2612	63588
30	1913	63583
60	1745	63578
90	1701	63574

Cloud providers usually offer on-demand and reserved plans. Since the resources traded in clouds are non-storable and the physical resources need to be replaced very often, exploiting part of the reserved capacity that is not used by the reserved users has a great benefit for the providers. Nevertheless, Quality of Service (QoS) to customers who reserves the requests in advance should be satisfied. Therefore, a need for a mechanism to guarantee resources to reserved users whenever they need them, while keeping resources loaded all the time is of value.

We proposed a financial option model for a federation of Cloud providers to address the above situation. This model allows a provider to hedge the critical and risky situation of reserved users requesting the resources while all the resources have been allocated to other users, by trading (buy or outsource) resources from other service providers in the Cloud federation. Experimental results showed that financial option based contracts between Cloud providers in a Cloud federation, would help them to exploit the underutilized reserved capacity without any concern to acquire the needed resources at any given time. The provider's profit will be increased by using our model, while the number of rejections of the reserved requests is negligible. The model therefore contributes to obtaining a trust and goodwill from the provider's client base.

In our model, we did not consider strategies regarding selling options. As a future work, we plan to study and evaluate issues regarding selling call options. Moreover, we will explore the situations that providers with a large amount of physical resources want to buy put options that will give them the right to sell resources at their will.

ACKNOWLEDGMENT

The authors would like to thank Rodrigo N. Calheiros, Kurt Vanmechelen, William Voorsluys, and Amir Vahid Dastjerdi for their helpful discussions and suggestions. We also thank our CLOUDS lab. colleagues, in particular Mohsen Amini Salehi and Deepak Poola.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of Cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, Apr. 2010.

[2] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, Jun. 2009.

[3] S. Chaisiri, B.-S. Lee, and D. Niyato, "Optimal virtual machine placement across multiple cloud providers," in *Proceedings of IEEE Asia-Pacific Services Computing Conference (APSCC'09)*. Jeju Island: IEEE, Dec. 2009, pp. 103–110.

[4] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, I. M. Llorente, R. Montero, Y. Wolfsthal, E. Elmroth, J. Caceres, M. Ben-Yehuda, W. Emmerich, and F. Galan, "The Reservoir model and architecture for open federated Cloud computing," *IBM Journal of Research and Development*, vol. 53, no. 4, pp. 1–11, Jul. 2009.

[5] R. Buyya, R. Ranjan, and R. N. Calheiros, "InterCloud: Utility-oriented federation of Cloud computing environments for scaling of application services," in *Proceedings of the 10th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP'10)*, ser. Lecture Notes in Computer Science, vol. 6081. Busan: Springer, May 2010, pp. 13–31.

[6] Í. Goiri, J. Guitart, and J. Torres, "Economic model of a Cloud provider operating in a federated Cloud," *Information Systems Frontiers*, pp. 1–17, 2011, 10.1007/s10796-011-9325-x.

[7] A. N. Toosi, R. N. Calheiros, R. K. Thulasiram, and R. Buyya, "Resource provisioning policies to increase IaaS provider's profit in a federated Cloud environment," in *Proceedings of 13th IEEE International Conference on High Performance Computing and Communications (HPCC'11).* Banff: IEEE, sept. 2011, pp. 279 –287.

[8] M. R. Rahman, Y. Lu, and I. Gupta, "Risk aware resource allocation for Clouds," Department of Computer Science, University of Illinois at Urbana-Champaign, Tech. Rep., 2011.

[9] W. F. Sharpe, "Capital asset prices: A theory of market equilibrium under conditions of risk," *Finance*, vol. XIX, no. 3, pp. 425–442, Sep. 1964.

[10] J. Hull, *Options, futures and other derivatives*. Pearson Prentice Hall, 2009.

[11] L. Rodero-Merino, L. M. Vaquero, V. Gil, F. Galán, J. Fontán, R. S. Montero, and I. M. Llorente, "From infrastructure delivery to service management in Clouds," *Future Generation Computer Systems*, vol. 26, no. 8, pp. 1226–1240, Oct. 2010.

[12] B. Song, M. M. Hassan, and E.-N. Huh, "A novel Cloud market infrastructure for trading service," in *Proceedings of the 9th International Conference on Computational Science and Its Applications (ICCSA'09).* Suwon: IEEE Computer Society, Jun. 2009, pp. 44–50.

[13] M. Mihailescu and Y. M. Teo, "Dynamic resource pricing on federated Clouds," in *Proceedings of IEEE International Symposium on Cluster Computing and the Grid.* Los Alamitos: IEEE Computer Society, May 2010, pp. 513–517.

[14] E. R. Gomes, Q. B. Vo, and R. Kowalczyk, "Pure exchange markets for resource sharing in federated Clouds," *Concurrency and Computation: Practice and Experience*, vol. 23, 2011.

[15] K. Vanmechelen, W. Depoorter, and J. Broeckhove, "Combining futures and spot markets: A hybrid market approach to economic Grid resource management," *Journal of Grid Computing*, vol. 9, no. 1, pp. 81–94, Mar. 2011.

[16] D. Allenotor and R. K. Thulasiram, "Grid resources pricing: A novel financial option based quality of service-profit quasi-static equilibrium model," in *Proceedings of the 9th IEEE/ACM International Conference on Grid Computing*. Tsukuba: IEEE, Oct. 2008, pp. 75–84.

[17] T. Meinl and D. Neumann, "A real options model for risk hedging in Grid computing scenarios," in *Proceedings of the 42nd Hawaii International Conference on System Sciences, 2009 (HICSS'09).* Hawaii: IEEE, Jan. 2009, pp. 1–10.

[18] A. Bossenbroek, A. Tirado-Ramos, and P. M. A. Sloot, "Grid resource allocation by means of option contracts," *Systems Journal, IEEE*, vol. 3, no. 1, pp. 49–64, March 2009.

[19] M. Mazzucco and M. Dumas, "Reserved or On-demand instances? a revenue maximization model for Cloud providers," in *Proceedings of 4th IEEE International Conference on Cloud Computing (CLOUD'11).* Washington: IEEE, Jul. 2011, pp. 428–435.

[20] H. Markowitz, "Portfolio selection," *The journal of finance*, vol. 7, no. 1, pp. 77–91, 1952.

[21] W. Michalk, L. Lilia Filipova-Neumann, B. Blau, and C. Weinhardt, "Reducing Risk or Increasing Profit? Provider Decisions in Agreement Networks," *Service Science*, vol. 3, no. 3, pp. 206–222, 2011.

[22] F. Black and M. Scholes, "The pricing of options and corporate liabilities," *Journal of Political Economy*, vol. 81, pp. 637–654, Jan 1973.

[23] R. C. Merton, "Theory of rational option pricing," *Bell Journal of Economics*, vol. 4, pp. 141–183, 1973.

[24] J. C. Cox, S. A. Ross, and M. Rubinstein, "Options pricing: a simplified approach," *Journal of Financial Economics*, vol. 7, pp. 229–263, 1979.

[25] S. Rahmail, I. Shiller, and R. K. Thulasiram, "Different estimators of the underlying asset's volatility and option pricing errors: parallel Monte Carlo simulation," in *Proceedings of International Conference on Computational Finance and its Applications*, Bologna, April 2004, pp. 121–131.

[26] K. Mills, J. Filliben, and C. Dabrowski, "Comparing vm-placement algorithms for on-demand Clouds," in *Proceedings of IEEE International Conference onCloud Computing Technology and Science*. Los Alamitos: IEEE Computer Society, Dec. 2011, pp. 91–98.

[27] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities," in *Proceedings of the International Conference on High Performance Computing & Simulation (HPC&S'09).* Leipzig: IEEE Computer Society, Jun. 2009, pp. 1–11.

[28] B. Javadi, R. K. Thulasiram, and R. Buyya, "Statistical modeling of spot instance prices in public Cloud environments," in *proceedings of the Fourth IEEE International Conference on Utility and Cloud Computing (UCC'11).* Melbourne: IEEE, Dec. 2011, pp. 219–228.