Agent-Based Pricing Determination for Cloud Services in Multi-Tenant Environment

Masnida Hussin, Azizol Abdullah, and Rohaya Latip

***Abstract*—Cloud computing acts as a resource sharing pool that provides services to multiple customers, which are called tenants through the Internet. One of the big challenges in cloud is providing a price for leasing the services while adapting with budget limit of the tenants. In order to meet the rapidly growing and dynamic demands of tenants, this paper proposes a pricing determination scheme for cloud services using mathematical analysis. It aims to balance satisfaction between tenants and service provider in terms of budget and profit. Specifically, our pricing determination procedure aggregated the budget constraint of tenants and service cost to calculate the potential price of service. Service level agreement (SLA) is handled by an agent for determining minimum and maximum prices that represent in a range. Hence, the service cost that charged by the provider is identified within the price range in order to meet tenants' requests. The results from our simulation demonstrate that the proposed pricing determination scheme provides better tenant satisfaction while sustaining provider profitability.**

***Index Terms*—Cloud computing, multitenant, pricing determination scheme, service level agreement.**

I. Introduction

Cloud computing is formally defined as an IT resource supply model that provides users with configurable resources over network. It is becoming a trend that the resources (e.g. servers, storage, and bandwidth) are available for large numbers of existing business applications from companies and institutes. The cloud services that offered to end users through network access are charged using a 'pay-per-use' model [1]-[3]. The payment model needs to determine when, how many and for how long such resources are required by the users. A traditional application service provider typically manages one dedicated application instance per user. In contrast, Cloud providers typically adopt a multi-tenant architecture [1]. It means that a shared middleware platform is used to host multiple users/tenants on top of a shared operating system, which may be either placed on a physical or virtualized hardware.

Specifically, the cloud service provider's interest is to improve the system throughout while satisfying as many tenant requests as possible. Satisfaction on Clouds in regards to their services is an important indicator that reflect quality of IT resource management. Therefore, more and more Cloud services hosted by Cloud service providers are available and deployed on virtual machine (VM). At the same time, rental cost is another issue that concerned by tenants. There is still a limited solution for pricing assignment towards sharing and managing Cloud services especially in terms of suitable cost model for multi-tenant environments. It is hard to accurately determine general rental cost for service due to diversity in service demands from tenants. Meanwhile, each service provider aims at optimizing its objectives (e.g., profit) rather than the performance of system as a whole. In addition, the process of 'buying and selling' of the available services over network, normally is controlled by particular service provider rather than by considering other providers as well.

This paper is motivated by the pricing assignment issue in Clouds. We present a service costing framework by representing relationships of system environments in Cloud as expressions, equations and inequalities. We effectively identify entity in the system environment (i.e., service provider, tenant, applications) and their features to design a cost model. While aiming to satisfy both service provider and tenants relationship interests, system's agent handled the Service Level Agreement (SLA) between tenants and provider for determining minimum and maximum service prices that represent in a range. With the price range, tenants easy to adjust their service request with the budget limit. Meanwhile, the service provider able to satisfy its service cost.

II. Related Work

In Clouds with heterogeneous services, economic-based resource management approaches are often adopted for various reasons including effective cost. Fair price for Cloud services is a good way of motivating entities in such systems to interact and utilize available services. The concept of service cost in Clouds is popular in resource management system [4]-[7] for dealing with variability and instability of resources. In particular, the service cost is influenced by various factors, such as prior performance, network capacity and resource availability. Their resource management approaches that based on computing cost have demonstrated the effectiveness in improving resource utilization; however, the efficacy of the approaches in dealing with fair Cloud service pricing for multi-tenant is limited to a certain level.

Inspired by the on-demand services from consumers, the pay-as-you-go model has been used to be pricing model in Clouds [1]. This type of pricing model is appropriate for situations where reliable and capable services are continuously available for the tenant to rent them. Basically, the price is charged based on number of CPU, availability of live storage, software license fee, backup and maintenance [8].

Manuscript received April 9, 2014; revised June 25, 2014.

Masnida Hussin, Azizol Abdullah, and Rohaya Latip are with the Department of Communication Technology and Networks, Faculty of Computer Science and Information Technology, University of Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia (e-mail: masnida@upm.edu.my, azizol@upm.edu.my, rohayalt@upm.edu.my).

DOI: 10.7763/IJCCE.2014.V3.368

There are also additional charges that recommend by Cloud providers in offering the services. Amazon web services [9] charged the tenants not only for their main services but also for the upfront infrastructure and global reach in quick access. Some Cloud providers used package of service for fix price e.g., [10]-[12]. There are also where the Cloud provider offers contract basis pricing method [13] to charge the service that demanded by the tenants. Specifically, the pricing strategy is different from one Cloud provider to another. However, their cost service models have similar scheme in order to charge the tenants.

They have taken into account the two major factors of requests' descriptions: (i) how many and (ii) how long. For example, most of the pricing method considered for number of CPU that utilized by tenants and size of storage used. The time or rental duration is also significant factor to charge the tenant. The pricing methods to charge over Cloud services are still an open question due to they are all controlled by the providers but disappear to tenants. While most pricing approaches in Cloud having their own pricing strategy, our work designs pricing determination scheme for Cloud services that represent the price in a range (with minimum and maximum prices). We also take into account the Cloud entities (i.e., service provider, tenant) and their goals (i.e., profit, satisfaction) in designing the price range.

III. Models

In this section, we describe the Cloud service infrastructure and cost model used in our study. The pricing determination process scheme is induced and presented based on the service infrastructure in Fig. 1.

A. Service Infrastructure

The target Cloud service infrastructure (Fig. 1) used in this work consists of a set T of *n* tenants that are loosely connected by a communication network, and each has a separate application. Each tenant requested for Cloud services either for storage, CPU or bandwidth where each is associated with a set of parameters, given as {rental duration, weight of workload, budget limit}. It is assumed that the services that requested by the tenant are always available. Hence, there is no issue on service provisioning and allocation.

The middleware platform uses agent-based agreement strategy to host multiple tenants. It is where the Service Level Agreement (SLA) for identifying the price of requested service occurred. The agent can be local or distributed entity in the infrastructure. The agent attached to tenants through communication network. The inter communication cost between them is insignificant. Note that, there is only one service provider in the Cloud infrastructure of our system model. Therefore, the agent merely focuses on pricing determination rather than negotiation process.

The Cloud service provider aims to lease its services i.e., storage, bandwidth and CPU to tenants. Each service has its own profile (e.g., type of service, parameters and history performance). It is assumed that the agent has complete knowledge of services that offer by the provider. In this work, we only consider the interaction between tenants and provider through the agent for developing a price range for Cloud services.

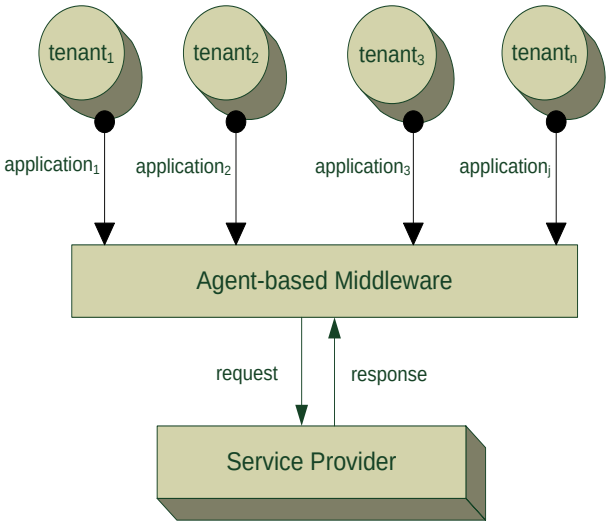

Fig. 1. Service infrastructure.

B. Cost Model

The cost model is applied in multi-tenant environment where the service requests are varies and unpredictable. The pricing determination process between provider and multi-tenant in this work aims to satisfy their objectives in terms of cost for leasing the services. The agent in the Cloud infrastructure plays an important role to set the range of price without discriminate any entities (provider and tenant).

Specifically, the agent evaluates the service request by the tenant to match with the service supply from the provider. The tenants submitted the service request and waited for SLA procedure before agreed to rent the service. The Cloud services in this work are charged by the provider based on a service value. The service value is not necessary in dollar ($) where it can represents in variable instances price e.g., time, volume and reward. For each tenants, the service request is given by,

$$T\_req = (dem, b\_limit) \quad (1)$$

where *dem* refers to demand or service descriptions including rental duration and weight of workload, and *b_limit* is budget limit that sets by tenant in order to pay for service, respectively. More specifically, tenants aims to reach the service price within their budget. Meanwhile, the service provider has its operational goal, given as

$$P\_goal = (c, pf) \quad (2)$$

where *c* is a service cost, *pf* refers to profit or maximum margin between service cost and rental costs, respectively. The provider intentions to maximize the profit while supplying the services to tenants. It is assumed that only the agent has knowledge about the value of service request and operational goal.

IV. Service Demand and Supply Formation

The efficacy of service sharing activity in distributed systems is greatly influenced by the capability of the services (i.e., storage, bandwidth and CPU). For this work we consider

the Cloud service refers to any of the services that can satisfy the tenants.

Due to there are diversity of service requirements from the tenants, we invent a grouping strategy for classifying the tenants' request into appropriate group. The service request is grouped according to tenant budget limit and can be classified into *low-budget* and *high-budget.* The tenant assigned in the *low-budget* group if a percentage different of its request and budget is more 50% than average budget limit in the group. Otherwise, the tenant is considered as a member in *high-budget* group. For these service request groups, the classification is primarily determined based on two different characteristics of the request; (i) weight of workload and (ii) duration of tenancy, as shown in Table I.

TABLE I: SERVICE FEATURES IN EACH SERVICE GROUP

Group Name	Weight of workload, *w*.	Duration of tenancy, *d*.
low-budget or *high-budget*	Small, Medium or Large	Long, Moderate or Short

Note that each of the request group, i.e., *low-budget* and *high-budget* is associated with a weight of workload *w* and expected duration for renting the service *d*. Both parameters are compared by their level of significant with 20% and 70% measurement. These percentages of measurement satisfy with a number of tenants that used in this work. The weight of workload is associated with a threshold (high, average or small) to verify its processing weight in an attempt to balance between provider's revenue and tenant's expense. Hence, the average weight of workload in the service request group is calculated (i.e., $ave_w = \sum w$/number of applications). For the weight of workload *w*, the threshold given as,

$$Small_w = \{when\ w \leq 20\%\ ave_w\}$$
$$Medium_w = \{when\ 20\%\ ave_w < w \leq 70\%\ ave_w\}$$
$$Large_w = \{when\ w > 70\%\ ave_w\}$$

Hence, the workload threshold is written as:

$$S_{dem}w = \{ Small_w, Medium_w, Large_w\} \quad (3)$$

The rental duration is associated with three different lengths; long, moderate and short. These lengths are used to ensure the duration of processing time is always within the tenants' expected expenses. The group's rental time emphases based on average rental duration in the group, ave_t. The weight for the rental duration *d* given as follows:

$$Short_d = \{when\ d \leq 20\%\ ave_d\}$$
$$Moderate_d = \{when\ 20\%\ ave_d < d \leq 70\%\ ave_d\}$$
$$Long_d = \{when\ d > 70\%\ ave_d\}$$

The duration threshold is below,

$$S_{dem}d = \{ Short_d, Moderate_d, Long_d\} \quad (4)$$

Since the demand from the tenants is grouped according to different level of thresholds, there is product space of threshold as follow:

$$dem_{space} = \{< Small_w, Long_d >, < Small_w, Moderate_d >,$$
$$< Small_w, Short_d >, < Medium_w, Long_d >,$$
$$< Medium_w\ Moderate_d >, < Medium_w, Short_d >,$$
$$<Large_w, Long_d >, < Large_w, Moderate_d >,$$
$$< Large_w, Short_d >\}$$

It also can be written as:

$$dem_{space} = S_{dem}w \times S_{dem}d \quad (5)$$

In response to number of tenants in the service provider infrastructure, the probability of demand event is the sum of the probabilities of number of demand send by tenants:

$$P(\{dem_{space}\}) = P(dem_{space\ 1}) + P(dem_{space\ 2}) + \ldots + P(dem_{space\ n}) \quad (6)$$

where *n* is number of tenant in the event. The service request that extended from Eq. (1) is then calculated as:

$$T\_req = (dem_{space(n)}, b\_limit) \quad (7)$$

As mentioned before, if the percentage difference; i.e., $(\sum (dem_{space(n)} + b\_limit) / average(\sum (dem_{space(n)} + b\_limit)))$ is more 50% than average *b_limit* of the tenants, it is categorized as *low-budget* group. Otherwise, it is considered as *high-budget* group.

The provider charged the service according its operational goal. Basically, it depends on service cost and profit; given in Eq. (2). The service cost is defined as *c* = f (*service*) * rental cost; where f (*service*), is function of operating cost including execution cost, maintenance cost and administration cost. We assume that the cost's profile is available and can be provided by the provider using service profiling. Hence, from the Eq. (1), the operational goal extended as follows:

$$P\_goal = ((f\ (service) * rental\ cost), pf) \quad (8)$$

The information related to service request and operational goal is then sent to the agent for pricing determination process.

V. PRICE RANGING FORMULATION

The agent-based pricing determination scheme essentially aims to identify the best range of price for Cloud services that satisfies tenant's expenses and provider's revenue. The price range aims to deal with variability in the service request from tenants. The agent considered service request and operational cost in order to capture elements that significantly affect service price.

In this work, the agent plays an important role to identify the minimum and maximum prices for the service. The agent received information regarding service request from tenants and operational cost from the provider. The agent then compares the value of service request with the operational cost; given in Eq. (7) and Eq. (8), respectively. Specifically,

the price determination process occurred for two different interactions. First, the interaction happen between the agent and tenant, and second interaction is between agents and provider. In the first interaction, the agent needs to ensure the tenants' requests must not exceed tenant's budget. Meanwhile, the agent needs to guarantee that service charged to tenant must not explicitly affect the provider's profit. In other words, the price of service that charged by provider to tenants must be able establishing cost requirements for the Cloud operation.

It is important to note that, the profit is related to service and rental costs in Cloud infrastructure and generated by the provider prior to pricing determination process. Specifically, when the agent identified the value of service request and operational goal, it generated interaction channel with tenants and provider. As mentioned earlier, there are two type of interaction (e.g., agent ↔ tenants and agent ↔ provider). So, the agent compared the function of operational goal *P_goal* and function of service request *T_req* for constructing the price range. The function of operational goal f(*P_goal*) is assumed to be higher than the function of service request, f(*T_req*). It is normal exercise in the real economy where supply must be more than demand.

From the provider perspective, it provides the service to tenant as long as the agreed price does not falling behind its service cost and profit. It is assumed that the provider meets its goal (maximize profit) with probability *PG* while considering the tenants' budget with probability (1 - *PG*). It means that the price of service must be determined within the range of *PG* and (1 – *PG*). Due to the operational goal determined to be higher than the agreed price, the agent is then performed the price range by regulating the probability of *PG* with function of operational goal f(*P_goal*) and function of service request, f(*T_req*) . Specifically, if it is determined that the value of f(*P_goal*) is more 50% than value of f(*T_req)* then the agent performed the price range according to Rule 1. It aims to satisfy tenant's budget limit.

Rule 1:

$$[(1\text{-PG}) * f(P\_goal)] \leq P_{agreed} \leq [PG * f(T\_req)]$$

Otherwise, the agent followed the Rule 2 to construct the price range, given below:

Rule 2:

$$[PG * f(P\_goal)] \leq P_{agreed} \leq [(1 - PG) * f(T\_req)]$$

In the Rule 2, the agent controlled the price in order to keep provider's profit within the right charged.

The provider then will be charged the tenant for the requested service according to the agreed price P_{agreed}. It means that the actual price for the service request must within $P_{agreed.}$ In this work, the agent is limited to develop the price range for the service. The actual price that charged by the provider to the tenant for the service usage can be determined using different Service Level Agreement (SLA) procedure (e.g., Game Theory, Bargaining Theory etc.). Our pricing determination decisions for Cloud services are considered to be effective in that provider meet its operational goal while satisfying tenants' budget limits.

VI. Results and Discussions

A. Evaluation Methodology

We have evaluated our agent-based pricing determination scheme via simulations with number of tenants ranging from 6 to 20. Each tenant dynamically submits the requests in Poisson distribution. The set of service request of tenant is randomly assigned with a diverse set of application. The rental duration selecting randomly from the following set: {2.5, 5.8, 10.5, 15.8, 25.5, 60.8, and 90.5}.

The relative weight of workload is selected within the range of 1 and 7.5. The rental duration and weight satisfy with a single provider used in this work. The budget limit and service cost are generated based on the total execution time exet. Here we have $b\_limit = \alpha * exet$, $c = \beta * exet$ and $\alpha < \beta$.

B. Performance Metrics

- Tenant agreement satisfaction rate

It is defined as the ratio of service charged and budget limit. The service charged refers to different between maximum and minimum prices.

- Provider profit

We define profit of service provider as average profit divided by utilization rate *RU* (i.e., *RU* = *busy* / (*busy* + *idle*) where *busy* is the total time of service usage and *idle* is total idle time of service, respectively.)

C. Results

In this experiment, we investigate on how price negotiations between tenant and provider are influenced by the agent capability. In response to that, we compare our pricing strategy (group-based agreement or *GBA*) with another pricing scheme named single-based agreement (*SBA*). The formulations of tenant request and provider goal in *SBA* are same as *GBA* except that it does not support the service grouping component.

Results in Fig. 2 clearly show that *GBA* and *SBA* have contradictory performance with difference in plotting pattern. *GBA* illustrates that satisfactory rate increases linearly while *SBA* is shown linear reduction curve. It is demonstrated that group-based pricing determination process is able to work in variability on the service requests. Hence, there is a tendency of tenants' satisfactoriness growth towards extended period of service time. The single-based pricing determination process, however, decreases satisfactory rate due to the great variance between budget limit and agreed price over time.

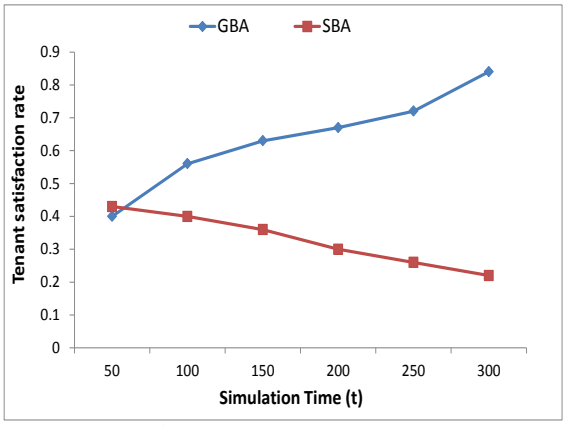

Fig. 2. Tenant satisfactory rate between GBA and SBA approaches.

The pattern of provider profit rate in Fig. 3 does not significantly differ as observed to be about 15%. It is due the fact that both strategies (i.e., *GBA* and *SBA*) rely on service demand and supply during their price adjustment process. However, *GBA* still outperforms *SBA*. Involvement of group of service request in the pricing determination process provides better information discovery. Hence, the provider is able to sustain better resource utilisation that improves provider's profit.

We enhance the analysis of *GBA* by comparing the balance between tenant satisfactory rate and provider profit rate. In this experiment, a goal completion rate is introduced to measure the percentage of tenant and provider satisfaction (e.g., demand, profit) in order to achieve balance in the service charges. As shown in Fig. 4, in the early simulation time when there is the range of price started to construct, tenants demonstrate higher completion rate compared to provider. This is can be explained by many tenants satisfied with the service charges. Meanwhile, the provider continuously shows improvement in its completion rate. This is particularly true in a real market where the provider tries to achieve balance between demand and supply while considering for equilibrium price [14].

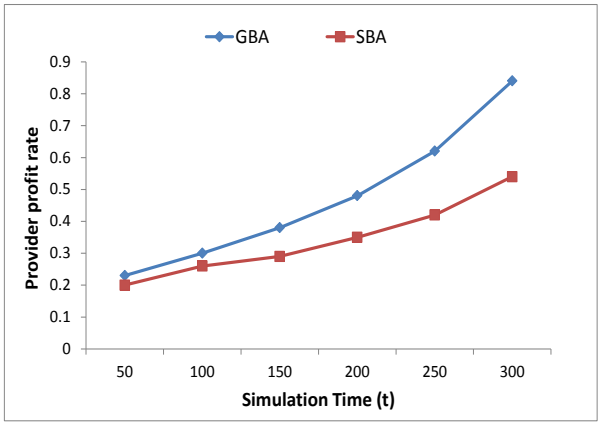

Fig. 3. Provider profit rate between GBA and SBA approaches.

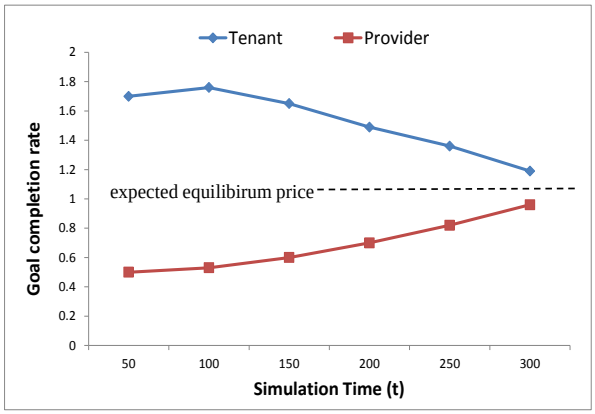

Fig. 4. Goal-completion rate for tenant satisfaction and provider profit.

VII. Conclusion

In this paper, we addressed the pricing determination process through agent-based middleware for Cloud service infrastructure. Our pricing scheme groups the service request from tenant to pave the way in making fair price determination. While considering the provider's profit, the agent constructs the range of service price that within tenants' budgets. The service price in this work is represented in minimum and maximum prices where the actual price is charged within the range. This group of service request takes into consideration demand details (i.e., budget limit, usage duration and weight of workload). The incorporation of tenant-provider relationships by using mathematical expressions, equations and inequalities into agent-based middleware highlights better information discovery in the service infrastructure. Our service price determination scheme gives appealing performance results in terms of satisfaction in both tenants and provider.

Acknowledgment

This work is supported by a Department of Higher Education Malaysia Grant 08-02-13-1363FR.

References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the clouds: a berkeley view of cloud computing," EECS Department, University of California, Berkeley, 2009.

[2] A. Beloglazov, R. Buyya, Y. C. Lee, and A. Y. Zomaya, "A taxonomy and survey of energy-efficient data centers and cloud computing systems," *Advances In Computers*, vol. 82, pp. 47-111, 2011.

[3] E. R. Gomes, Q. B. Vo, and R. Kowalczyk, "Pure exchange markets for resource sharing in federated clouds," *Concurrency and Computation: Practice and Experience*, pp. 977-991, 2012.

[4] R. N. Calheiros and R. Buyya, "Cost-effective provisioning and scheduling of deadline-constrained applications in hybrid clouds," *Web Information Systems Engineering*, Cyprus: Springer, 2012, pp. 171-184.

[5] G. Raj and S. Setia, "Effective cost mechanism for cloudlet retransmission and prioritized vm scheduling mechanism over broker virtual machine communication framework," *International Jrnl on Cloud Computing: Services and Architecture*, vol. 2, pp. 41-50, 2012.

[6] M. M. Zhu, Q. Wu, and Y. Zhao, "A cost-effective scheduling algorithm for scientific workflows in clouds," in *Proc. IEEE 31st Int'l Performance Computing and Communications Conference*, 2012, pp. 256-265.

[7] H. Xu and B. Li, "Dynamic cloud pricing for revenue maximization," *IEEE Transaction on Cloud Computing*, vol. 1, pp. 158-165, 2013.

[8] C. Weinhardt, A. Anandasivam, B. Blau, and J. Stober, "Business models in the server world," *IT Pro*, IEEE, 2009, pp. 28-33.

[9] Amazon. what is cloud computing. [Online]. Available: http://aws.amazon.com/what-is-cloud-computing/.

[10] HP cloud pricing. [Online]. Available: http://www.hpcloud.com/pricing

[11] IBM. Licensing for IBM Smartcloud enterprise. [Online]. Available: http://www-01.ibm.com/software/lotus/passportadvantage/licensing_for_IBM_Cloud.html.

[12] Cloud. [Online]. Available: https://cloud.google.com/products/.

[13] Microsoft azure. no upon costs. Pay only for what you use. [Online]. Abailable: http://www.windowsazure.com/en-us/pricing/calculator/?scenario=virtual-machines

[14] R. Pal and P. Hui, "Economic models for Cloud service markets : Pricing and Capacity Planning," *Theoritical Computer Science*, vol. 496, pp. 113-124, 2013.

Masnida Hussin is a senior lecturer at Faculty of Computer Science and IT, University Putra Malaysia UPM. She obtained her master of science in distributed systems from the University Putra Malaysia, in 2006 and her PhD degree in engineering from University of Sydney, Australia, in 2012. She is a member of the IEEE since 2008, also has published several numbers of papers that related to parallel and distributed computing. Mainly her research interests include resource management that including task scheduling, resource

allocation and discovery for distributed systems. She also involved in green computing project.

Azizol Abdullah obtained his master of science in engineering (telematics) from the University of Sheffield, UK in 1996 and his PhD degree in parallel and distributed system from Universiti Putra Malaysia, Malaysia in 2010. Currently, he is the head of Department Communication Technology and Network, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia. He is also has been appointed as a fellow researcher for ITU-UUM Asia Pacific Centre of Excellence For Rural ICT Development (ITU-UUM). His main research areas include cloud and grid computing, network security, wireless and mobile computing and computer networks.

Rohaya Latip is a senior lecturer at Faculty of Computer Science and Information Technology, University Putra Malaysia. She holds a Ph.D degree in distributed database from University Putra Malaysia, a MSc. degree in distributed system also from University Putra Malaysia. She served as an associate professor at Najran University, Kingdom of Arab Saudi (2012-2013). She is the editor in chief of International Journal of New Computer Architectures and their Applications (IJNCAA) and a board member of editor for International Journal of Computer Networks and Communications Security (IJCNCS). Her research interests include grid computing, network management, distributed database, and cloud computing. She has published more than 35 papers in international and national Journals, proceedings and posters.