ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

Profit Maximization for Cloud Brokers in Cloud Computing

Srikanth S.P[1], Manzoor Ahmed Khan[2], Manmeet Singh Sahni[3], Rabeea Omar O[4], Lakshmi Prasad M[5]

Assistant Professor, Department of Computer Science and Engineering, Sambhram Institute of Technology, Bengaluru, India[1]

Student, Department of Computer Science and Engineering, Sambhram Institute of Technology, Bengaluru, India[2]

Student, Department of Computer Science and Engineering, Sambhram Institute of Technology, Bengaluru, India[3]

Student, Department of Computer Science and Engineering, Sambhram Institute of Technology, Bengaluru, India[4]

Student, Department of Computer Science and Engineering, Sambhram Institute of Technology, Bengaluru, India[5]

ABSTRACT: Cloud computing is becoming more and more popular and has received considerable attention recently. As a new kind of Information Technology (IT) commercial model, understanding the economics of cloud computing becomes critically important. From the cloud service providers' perspective, profit maximization is the top issue for them. Because a multi-server system is devoted to serving one type of service requests and application, service providers should build multiple multi-server systems to satisfy the market requirements of different application domains. Because available funding for a service provider is generally limited, it cannot afford to invest in all application domains. Hence, how to select appropriate application domains for investment and allocate funding such that the total profit is maximized are important issues for service providers. To address this problem, a fund-constrained profit maximization model is proposed. However, the exact solution of this optimization model is very difficult to formulate due to its complexity. Hence, this paper presents a heuristic strategy to search for a high quality solution. In our strategy, the optimization problem is solved in four stages, and the solution is optimized gradually. Through the proposed heuristic investment strategy, an appropriate investment scheme can be developed that synthesizes the market requirement, the fund constraint, the service level agreement, and so forth. A series of numerical calculations is executed to assess the performance of the proposed strategy. Then, six other investment strategies are compared to our strategy. Our results show that the investment scheme designed using our strategy can produce much more profit than these six other strategies.

KEYWORDS: Cloud computing, guaranteed service quality, multi-server system, profit maximization, queuing model, service-level agreement, waiting time.

I. INTRODUCTION

As an effective and efficient way to consolidate computing resources and computing services, clouding computing has become more and more popular [1]. Cloud computing or if any centralizes management of resources and services, and delivers hosted services over the Internet. The hard- ware, software, databases, information, and all resources are concentrated and provided to consumers on-demand [2]. Cloud computing turns information technology into ordinary commodities and utilities by the the pay-per-use pricing model [3, 4, 5]. In a cloud computing environment, thereare always three tiers, i.e., infrastructure providers, services providers, and customers (see Fig. 1 and its elaboration in Section 3.1). An infrastructure provider maintains the basic hardware and software facilities. A service provider rents resources from the infrastructure providers and provides services to customers. A customer submits its request to a service provider and pays for it based on the amount and the quality of the provided service [6].

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

In this paper, we aim at researching the multi-server configuration of a service provider such that its profit is maximized.

Like all business, the profit of a service provider in cloud computing is related to two parts, which are the cost and there venue. For a service provider, the cost is therenting cost paid to the infrastructure providers plus the electricity cost caused by energy consumption, and the revenue is the service charge to customers. In general, a service provider rents a certain number of servers from the infrastructure providers and builds different multi-server systems for different application domains. Each multi-server system is to execute a special type of service requests and applications. Hence, the renting cost is proportional to the number of servers in a multi-server system [2].

The power consumption of a multi-server system is linearly proportional to the number of servers and the server utilization, and to the square of execution speed [7, 8]. The revenue of a service provider is related to the amount of service and the quality of service. To summarize, the profit of a service provider is mainly determined by the configuration of its serviceplatform.

To configure a cloud service platform, a service provider usually adopts a single renting scheme. That's to say, the servers in the service system are all long-term rented. Be- cause of the limited number of servers, some of the incoming service requests cannot be processed immediately. So they are first inserted into a queue until they can handled by any available server. However, the waiting time of the service requests cannot be too long. In order to satisfy quality-of-service requirements, the waiting time of each incoming service request should be limited within a certain range, which is determined by a service-level agreement (SLA). If the quality of service is guaranteed, the service is fully charged, otherwise, the service provider serves the request for free as a penalty of low quality. To obtain higher revenue, a service provider should rent more servers from the infrastructure providers or scale up the server execution speed to ensure that more service requests are processed with highservice quality. However, doing this would lead to sharp increase of the renting cost or the electricity cost. Such increased cost may counterweight the gain from penalty reduction. In conclusion, the single renting scheme is not a good scheme for service providers. In this paper, we propose a novel renting scheme for service providers, which not only can satisfy quality-of-service requirements, but alsocan obtain more profit. Our contributions in this paper can be summarized as follows:

- A novel double renting scheme is proposed for service providers. It combines long-term renting with short-term renting, which can not only satisfy quality-of-service requirements under the varying system workload, but also reduce the resource waste greatly.
- A multi-server system adopted in our paper is modeled as an *M/M/m+D* queuing model and the performance indicators are analyzed such as the average service charge, the ratio of requests that need short- term servers, and soforth.
- The optimal configuration problem of service providers for profit maximization is formulated and two kinds of optimal solutions, i.e., the ideal solutions and the actual solutions, are obtained respectively.
- A series of comparisons are given to verify the performance of our scheme. The results show that the proposed Double-Quality-Guaranteed (DQG) renting scheme can achieve more profit than the com- pared Single-Quality-Unguaranteed (SQU) renting scheme in the premise of guaranteeing the service qualitycompletely.

The rest of the paper is organized as follows. Section 2 reviews the related work on profit aware problem in cloud computing. Section 3 presents the used models, including the three-tier cloud computing model, the multi-server system model, the revenue and cost models. Section 4 pro- poses our DQG renting scheme and formulates the profit optimization problem. Section 5 introduces the methods of finding the optimal solutions for the profit optimization problemintwoscenarios.Section6demonstratesthepresentationoftheproposedschemethroughcomparisonwiththe traditional SQU renting scheme. Finally, Section 7concludes thework.

II. RELATED WORK

In this section, we review recent works relevant to the profit of cloud service providers. Profit of service providers is related with many factors such as the price, the market demand, the system configuration, the customer satisfaction and so forth. Service providers naturally wish to set ahigher price to get a higher profit margin; but doing so would decrease

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

the customer satisfaction, which leads to a risk of disheartening demand in the future. Hence, selecting a reasonable pricing strategy is important for service providers.

The pricing strategies are divided into two categories, i.e., static pricing and dynamic pricing. Static pricing means that the price of a service request is fixed and known in advance, and it does not change with theconditions. With dynamic pricing a service provider delays thepricing decision until after the customer demand is revealed, sothat the service provider can adjust prices accordingly [9]. Static pricing is the dominant strategy which is widely used in real world and in research [2, 10, 11]. Ghamkhari *et al.* [11] adopted a flat-rate pricing strategy and set a fixed price for all requests, but Odlyzko in [12] argued that the predominant flat-rate pricing encourages waste and is incompatible with service variation. Another kind of static pricing strategies are usage-based pricing. For example, the price of a service request is proportional to the service time and task execution requirement (measured by the number of instructions to be executed) in [10] and [2], respectively. Usage-basedpricingrevealsthatonecanuseresourcesmore efficiently [13,14].

Since profit is an important concern to cloud service providers, many works have been done on how to boost their profit. A large body of works have recently focused on reducing the energy cost to increase profit of service providers [22, 23, 24, 25], and the idle server turning off strategy and *dynamic CPU clock frequency scaling* are adopted to reduce energy cost. However, only reducing energy cost cannot obtain profit maximization. Many researchers investigated the trade-off between minimizing cost and maximizing revenue to optimize profit. Both [11] and [26] adjusted the number of switched on servers periodically using different strategies and different profit maximization models were built to get the number of switched on servers. However, these works did not consider the cost of resource configuration. Chiang and Ouyang [27] considered a cloud server system as an *M/M/R/K* queuing system where all service requests that exceed its maximum capacity are rejected. Aprofit maximization function is defined to find an optimal combination of the server size *R* and the queue capacity *K* such that the profit is maximized. However, this strategy has further implications other than just losing the revenue from some services, because it also implies loss of reputation and therefore loss of future customers [3]. In [2], Cao *et al.* treated a cloud service platform as an *M/M/m model*, and the problem of optimal multi-server configuration for profit maximization was formulated and solved. This work is the most relevant work to ours, but it adopts a single renting scheme to configure a multi-server system, which cannot adapt to the varying market demand and leads to low service quality and great resource waste. To overcome this weakness, another resource organization strategy is used in [28, 29, 30, 31], which is cloud federation. Using federation, different providers running services that have complementary resource requirements over time can mutually collaborate to share their respective resources in order to fulfill each one's demand [30]. However, providers should make an intelligent decision about utilization of the federation (either as a contributor or as a consumer of resources) depending on different conditions that they might face, which is a complicatedproblem. In this paper, to overcome the shortcomings mentioned above, a double renting scheme is designed to configure a cloud service platform, which can guarantee the service quality of all requests and reduce the resource wastegreatly. Moreover, a profit maximization problem is formulated and solved to get the optimal multi-server configuration which can product more profit than the optimal configuration in[2].

III. THE MODELS

In this section, we first describe the three-tier cloud computing structure. Then, we introduce the related models used in this paper, including a multi-server system model, a revenue model, and a cost model.

1) A Cloud SystemModel

The cloud structure (see Fig. 1) consists of three typical parties, i.e., infrastructure providers, service providers and customers. This three-tier structure is used commonly in existing literatures [2, 6,10].

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

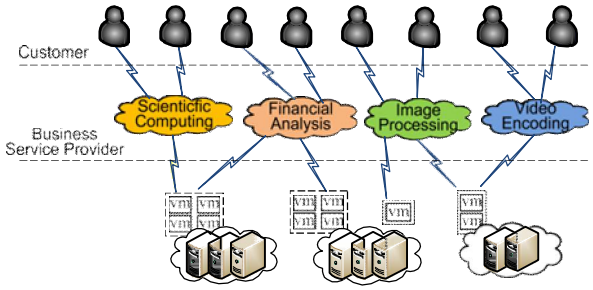

Fig. 1: The three-tier cloud structure.

In the three-tier structure, an infrastructure provider the basic hardware and software facilities. A service provider rents resources from infrastructure providers and prepares a set of services in the form of virtual machine (VM). Structure providers provide two kinds of resource renting schemes, e.g., long-term renting and short-term renting. In general, the rental price of long-term renting is much cheaper than that of short-term renting. A customer submits a service request to a service provider which delivers services on demand. The customer receives the desired result from the service provider with certain service-level agreement, and pays for the service based on the amount of the service and the service quality. Service providers pay infrastructure providers for renting their physical resources, and charge customers for processing their service requests, which generates cost and revenue, respectively. The profit is generated from the gap between the revenue and the cost.

2) A Multi-serverModel

In this paper, we consider the cloud service platform as a multi-server system with a service request queue.Fig.2gives the schematic diagram of cloud computing[32].

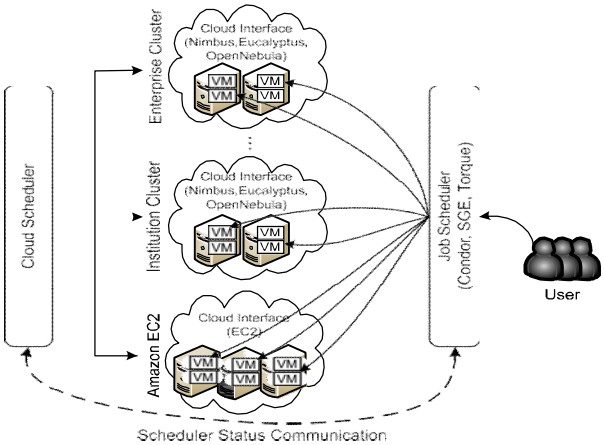

Fig. 2: The schematic diagram of cloud computing.

In an actual cloud computing platform such as Amazon EC2, IBM blue cloud, and private clouds, there are many work nodes managed by the cloud managers such as Eucalyptus, OpenNebula, and Nimbus. The clouds provide resources for jobs in the form of virtual machine (VM). In addition, the users submit their jobs to the cloud in which a job queuing system such as SGE, PBS, or Condor is used. All jobs are scheduled by the job scheduler and assigned to different VMs in a centralized way. Hence, we can con- sider it as a service request queue. For example, Condor is a specialized workload management system for compute- intensive jobs and it provides a job queuing mechanism,

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

scheduling policy, priority scheme, resource monitoring, and resource management. Users submit their jobs to Condor, and Condor places them into a queue, chooses when and where to run them based upon a policy [33, 34]. Hence, it is reasonable to abstract a cloud service platform as amulti-server model with a service request queue, and the model is widely adopted in existing literature [2, 11, 35, 36,37]. In the three-tier structure, a cloud service provider serves customers' service requests by using a multi-server system

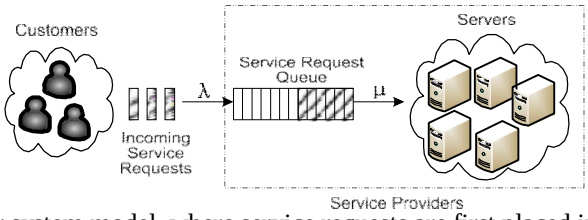

Fig. 3: The multi-server system model, where service requests are first placed in a queue before they are processed by any servers.

server system consists of *m* long-term rented identical servers, and it can be scaled up by temporarily renting short-term servers from infrastructure providers. The servers in the system have identical execution speed *s* (Unit: billion instructions per second). In this paper, a multi-server system excluding the short-term servers is modeled as an *M/M/m* queuing system as follows (see Fig. 3). There is a Poisson stream of service requests with arrival rate λ, i.e., the inter arrival times are independent and identically distributed (i.i.d.) exponential random variables with mean $1/\lambda$. A multi-server system maintains a queue with infinite capacity. When the incoming service requests cannot beprocessed immediately after they arrive, they are firstly placed in the queue until they can be handled by any available server.Thefirst-come-first-served(FCFS)queuingdiscipline is adopted. The task execution requirements (measured by the number of instructions) are independent and identically distributed exponential random variables *r* with mean *r* (Unit: billion instructions).

IV. REVENUE MODELING

The revenue model is determined by the pricing strategy and the server-level agreement (SLA). In this paper, the usage-based pricing strategy is adopted, since cloud computing provides services to customers and charges them on demand. The SLA is a negotiation between service providers and customers on the service quality and the price. Because of the limited servers, the service requests that cannot be handled immediately after entering the system must wait in the queue until any server is available. However, to satisfy the quality-of-service requirements, the waiting time of each service request should be limited within a certain range which is determined by the SLA. The SLA is widely used by many types of businesses, and it adopts a price compensation mechanism to guarantee service quality and customer satisfaction. For example, China Post gives a service time promise for domestic express mails. It promises that if a domestic express mail does not arrive within a deadline, the mailing charge will be refunded. The SLA is also adopted by many real world cloud service providers such as Rackspace [39], Joyent [40], Microsoft Azure [41], and so on. Taking Joyent as an example, the customers order Smart Machines, Smart Appliances, and/or Virtual Machines from Joyent, and if the availability of a customer's services is less than 100%, Joyent will credit the customer 5% of the monthly fee for each 30 minutes of downtime up to 100% of the customer's monthly fee for the affected server. The only difference is that its performance metric is availability and ours is waiting time. In this paper, the service level is reflected by the waiting time of requests. Hence, we define *D* as the maximumwaiting time here that the service requests can tolerate, in other words, *D* is their deadline. The service charge of each task is related to the amount of a service and the servicelevel agreement. We define the service charge function for a service request with execution requirement *r* and waiting time *W* in Eq. (2),

$$R(r,W) = \begin{cases} ar, & 0 \le W \le D; \\ 0, & W > D. \end{cases} \quad (2)$$

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

where*a* is a constant, which indicates the price per one billion instructions (Unit: cents per one billion instructions). When a service request starts its execution before waiting a fixed time *D* (Unit: second), a service provider considers that the service request is processed with high quality-ofservice and charges a customer *ar*. If the waiting time of a service request exceeds deadline *D*, a service provider must serve it for free. Similar revenue models have been used in many existing research such as [2, 11, 42]. According to Theorem 1, it is easy to know that the probability that the waiting time of a service request exceeds its deadline *D* is

$$P(W \_ D) = 1 - FW(D) = \pi m1 / \rho e - m\mu(1-\rho)D. \quad (3)$$

V. COST MODELING

The cost of a service provider consists of two major parts, i.e., the rental cost of physical resources and the utility cost of energy feasting. Many existing research such as [11, 43, 44] only consider the power consumption cost. As a major difference between their models and ours, the resource rental cost is considered in this paper as well, since it is a major part which affects the profit of service providers. A similar cost model is adopted in [2]. The resources can be rented in two ways, long-term renting and short-term renting, and the rental price of long-term renting is much cheaper than that of short-term renting. This is reasonable and common in the real life. In this paper, we assume that the long-term rental price of one server for unit of time is β (Unit: cents per second) and the short-term rental price of one server for unit of time is γ (Unit: cents per second), where $\beta < \gamma$. The cost of energy consumption is determined by the electricity price and the amount of energy consumption. In this paper, we adopt the following dynamic power model, which is adopted in the literature such as [2, 7, 45, 46]:

$$Pd = NswCLV\,2f, \quad (4)$$

where*Nsw* is the average gate switching factor at each clock cycle, *CL* is the loading capacitance, *V* is the supply voltage, and *f* is the clock frequency [45]. In the ideal case, the relationship between the clock frequency *f* and the supply voltage *V* is $V / f\phi$for some constant $\phi > 0$ [46]. The server execution speed *s* is linearly proportional to the clock frequency *f*, namely, s / f. Hence, the power consumption is $Pd / NswCLs2\phi+1$. For ease of discussion, we assume that $Pd = bNswCLs2\phi+1 = \xi s\alpha$ where $\xi = bNswCL$ and $\alpha = 2\phi+ 1$. In this paper, we set $NswCL = 7.0$, $b = 1.3456$ and $\phi = 0.5$. Hence, $\alpha = 2.0$ and $\xi = 9.4192$. The value of power consumption calculated by $Pd = \xi s\alpha$ is close to the value of the Intel Pentium M processor [47]. It is reasonable that a server still consumes some amount of static power [8], denoted as $P\_$ (Unit:Watt), when it is idle. For a busy server, the average amount of energy consumption per unit of time is $P = \xi s\alpha + P\_$ (Unit: Watt). Assume that the price of energy is δ (Unit: cents per Watt).

VI. CONCLUSIONS

In order to guarantee the quality of service requests and maximize the profit of service providers, this paper has proposed a novel Double-Quality-Guaranteed (DQG) renting scheme for service providers. This scheme combines short-term renting with long-term renting, which can reduce the resource waste greatly and adapt to the dynamical demand of computing capacity. An *M/M/m+D* queuing model is built for our multi-server system with varying system size. And then, an optimal configuration problem of profit maximization is formulated in which many factors are taken into considerations, such as the market demand, the workload of requests, the server-level agreement, the rental cost of servers, the cost of energy consumption, and so forth. The optimal solutions are solved for two different states, which are the ideal optimal solutions and the actual optimal solutions. In addition, a series of calculations are conducted to compare the profit obtained by the DQG renting scheme with the Single-Quality-Unguaranteed (SQU) renting scheme. The results show that our scheme outperforms the SQU scheme in terms of both of service quality and profit. In this paper, we only consider the profit maximization problem in a homogeneous cloud environment, because the analysis of a heterogonous environment is much more complicated than that of a homogenous environment. However, we will extend our study to a varied environment in the future.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

REFERENCES

[1] J. Cao, K. Hwang, K. Li, and A. Y. Zomaya, "Optimal multiserver configuration for profit maximization in cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1087–1096, 2013.
[2] P. Mell and T. Grance, "The NIST definition of cloud computing," *National Institute of Standards and Technology*, vol. 53, no. 6, p. 50, 2009.
[3] 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, 2009.
[4] A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, and I. Stoica, "Above the clouds: A berkeley view of cloud computing," *Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley, Rep. UCB/EECS*, vol. 28, p. 13, 2009.
[5] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 1, pp. 50–55, 2008.
[6] J. Chen, C. Wang, B. B. Zhou, L. Sun, Y. C. Lee, and A. Y. Zomaya, "Tradeoffs between profit and customer satisfaction for service provisioning in the cloud," in *Proceedings of the 20th international symposium on High performancedistributed computing*. ACM, 2011, pp. 229–238.
[7] J. Xu, A. Lam, and V. Li, "Chemical reaction optimization for task scheduling in grid computing," *IEEE Trans. Parallel and Distributed Systems*, vol. 22, no. 10, pp. 1624–1631, Oct2011.
[8] J. Yang, H. Xu, L. Pan, P. Jia, F. Long, and M. Jie, "Task scheduling using bayesian optimization algorithm for heterogeneous computing environments," *Applied Soft Computing*, vol. 11, no. 4, pp. 3297 – 3310, 2011.
[9] K. Li, "Optimal load distribution in nondedicated heterogeneous cluster and grid computing environments," *Journal of Systems Architecture*, vol. 54, no. 1 - 2, pp. 111 – 123, 2008.
[10] Q.-M. Kang, H. He, H.-M. Song, and R. Deng, "Task allocation for maximizing reliability of distributed computing systems using honeybee mating optimization," *Journal of Systems and Software*, vol. 83, no. 11, pp. 2165 – 2174, 2010.
[11] S. Kumar, K. Dutta, and V. Mookerjee, "Maximizing business value by optimal assignment of jobs to resources in grid computing," *European Journal of Operational Research*, vol. 194, no. 3, pp. 856 – 872, 2009.
[12] 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," *Commun.ACM*, vol. 53, no. 4, pp. 50–58, Apr. 2010.