SLA-Aware Dynamic Resource Provisioning for Profit Maximization in Shared Cloud Data Centers

Jing Bi[1], Zhiliang Zhu[1,2], and Haitao Yuan[2]

[1] School of Information Science and Engineering
[2] College of Software
Northeastern University, 110004 Shenyang, P.R. China
neubijing@gmail.com, zzl@mail.neu.edu.cn, neuhaitao@gmail.com

Abstract. Dynamic resources provisioning is necessary for the multi-tier different virtualized application services in shared cloud data centers to meet different service quality targets. For an appropriate provisioning mechanism, we proposed a novel dynamic provisioning technique and employ a flexible hybrid queueing model to determine the virtualized resources to provision to each tier of the virtualized application services. We further developed meta-heuristic solutions, which is according to different performance requirements of clients from different levels. Simulation experiment results show that these proposed approaches can provide appropriate way to judiciously provision cloud data center resources, especially for improving the overall performance while effectively reducing the resource usage extra cost. So, it verifies the benefit of our methodology.

Keywords: Infrastructure as a service (IaaS), Service level agreement (SLA), Dynamic provisioning, Performance.

1 Introduction

The main purpose of management of cloud data center is to ensure the quality and cost-effectiveness of cloud computing services so as to achieve much economic profits. Large-scale and diverse cloud computing services are running in cloud data center. The online clients can get services from cloud data center by sending their requests with corresponding parameters and invoking automatic execution of flows in cloud servers. The services should provide effective Service Level Agreements (SLAs) to ensure and differentiate services quality. Therefore, cloud service providers wish to ensure a concrete SLA level for every cloud computing service. And consumers also agree to pay for cloud service providers according to specified SLA levels. The degree of satisfaction according to performance consumers experience is linked directly to the profit of cloud service providers. However, one of the main problems ensured by SLA is that in actual cloud computing environment, due to dynamic variations of workload, it's difficult to estimate the requirement of services resources in advance, and obviously, it's infeasible and inefficient to prepare for the worst cases. To meet the constraint of SLA and allocate existing services resources

Y. Wu (Ed.): ICHCC 2011, CCIS 163, pp. 366–372, 2011.

optimally, the dynamic provisioning technology is adopted in cloud data center [1], which can be adjusted to allocate resources among different workloads.

In recent years, some researches have focused on the problem of resource management in data center. Some research works [2, 3, 4] proposed autonomic approach to replace traditional behavior strategy and target strategy. However, nowadays, most of those methods can not sufficiently adapt to complex cloud computing environment. These researches usually assume the system as the equilibrium state, and employ the method of average value analysis which is not sufficiently precise. We focuses on the problem of virtualized resources provisioning for existing cloud data center, which is to satisfy the requirement of clients' business [5], and maximize the overall profit of IaaS providers when SLA guarantees are satisfied or violated.

In this paper, firstly, with a constrained non-linear optimization technique, we can dynamically provision the virtualized resources and establish performance optimization model for cloud environment in multi-tier virtualized application services. We further develop meta-heuristic solutions based on the mixed tabu-search optimization algorithm to solve the optimization problem, which is according to different performance requirements of clients from different levels. With experiment results, the benefit of our methodology is verified.

2 The System Optimization Model

The constrained non-linear optimization problem is defined for dynamic virtualized resources optimization. Assume that *N M*-tier virtualized application services environments (VASEs) run in cloud IaaS, which include multiple different user class *K*. The capacities of physical servers from each tier are shared by virtual machines (VMs) serving different virtualized applications. A VASE may include multiple VMs that are distributed on physical servers from several tiers. Assume the number of clients class in VASE *i* is K_i, and there are $n_{i,j}$ VMs in the *j*th tier. So, the crucial variable of the problem is defined as a $N\times(M+1)$ matrix, *ConfigMAT*, which refers to the provisioning plan of VMs on physical servers in each tier, formally:

$$ConfigMAT = \begin{bmatrix} c_{1,0} & c_{1,1} & \cdots & c_{1,M} \\ c_{2,0} & \ddots & c_{i,j} & \vdots \\ \vdots & & \ddots & \\ c_{N,0} & \cdots & & c_{N,M} \end{bmatrix}$$

Let $c_{i,j}$ represent the number of active VMs allocated in the *j*th tier of VASE *i*. If $c_{i,j}$ is 0, it means that no active VMs exist in the *j*th tier of VASE *i*. In order to control the granularity of VMs provisioning, the upper limit of $c_{i,j}$ is set as C_i, which refers to the maximal number of VMs of VASE *i*, in all cloud IaaS. $n_{i,j}$ refers to maximal number of VMs occupied in the *j*th tier of VASE *i*. The provisioning matrix *ConfigMAT* is considered as valid if following constraint is met.

$$\begin{cases} 0<\sum_{j=0}^{M} c_{i,j} \le C_i, & \forall i \in [1,N] \\ 0 \le c_{i,j} \le n_{i,j}, & \forall i \in [1,N], \forall j \in [0,M] \end{cases} \tag{1}$$

The constraint (1) restricts that the total number of VMs occupied in all IaaS and the number of VMs occupied in the same tier can not exceed the total number of available virtualized resources. i.e., the total number of VMs in IaaS and the number of VMs of the same tier are both restricted by corresponding total physical resources in the cloud IaaS. The global profit value P_g is function of every local profit value P_i of VASE, so the whole optimization problem can be formulized as following problem (P_1):

$$\max\left\{P_g = g\left(P_1, P_2, ..., P_N\right)\right\} \tag{2}$$

In order to maximize the profit of cloud IaaS providers based on SLA, on the condition that equation (1) is met, the global profit value in equation (2) is optimized. Furthermore, virtualized resources of cloud data center can be used effectively. The concrete form of problem (P_1) will be presented in latter part of this section.

The profit function is described as follows. Here, the analysis is focus on multi-tier VASE, in which include multiple classes of online businesses. The arrival rate of the request class k in the jth tier of VASE i is represented as $\lambda_{i,k,j}$, and response time $R_{i,k}$ is considered as a performance metric. Assume the SLA agreement has been signed between cloud IaaS and clients before the system runs, where the specific performance requirements and charging model are defined as follows:

- $\overline{R}_{i,k}$ - the expected SLA target response time of request class k in VASE i. If a request is served in target response time, the positive revenue is contributed for cloud IaaS providers, i.e., if $R_{i,k} \le \overline{R}_{i,k}$, SLA_i is the revenue type. Otherwise, the case that a request is served beyond target response time will bring cloud IaaS providers penalty, i.e., if $R_{i,k} > \overline{R}_{i,k}$, SLA_i is the penalty type.
- C_i - maximal VMs number of VASE i in all cloud IaaS. If $\sum_{j=0}^{M_{i,k}} c_{i,j} \le C_i$, the refusal of clients' requests will lead to the penalty of $d_{i,k}$. i.e., when actual the number of VMs exceeds the concerted upper limit value, the refused clients' requests will not be counted into penalty. This makes clients must estimate actual requirements of applications services carefully and make an appropriate plan of expense before deployment of applications services.
- $c_{i,k,j,w}^{active}$ - average price of active VM w in the jth tier of request class k in VASE i.
- $c_{i,k,j,w}^{spare}$ - average price of dormant VM w in the jth tier of request class k in VASE i.

Our goal is to maximize profit value of cloud IaaS providers. Furthermore, the difference between revenue, and penalty, loss and cost of VMs from SLA can be maximized. The profit function can be formulated as follows:

$$Profit(E)=\sum_{i=1}^{N}\sum_{k=1}^{K_i}\left\{\Lambda_{i,k}\cdot\left(\left(-m_{i,k}\right)\cdot R_{i,k}+u_{i,k}\right)-\left(d_{i,k}\cdot x_{i,k}\right)-\left(LV_{i,k}\cdot\left(1-A_{i,k}\right)\right)\right\} \\ -\sum_{i=1}^{N}\sum_{k=1}^{K_i}\sum_{j=0}^{M}\left(\sum_{w=1}^{c_{i,j}}c_{i,k,j,w}^{active}+\sum_{w=1}^{n_{i,j}-c_{i,j}}c_{i,k,j,w}^{spare}\right) \tag{3}$$

where

- $\Lambda_{i,k}$ is total arrival rate of request class k in VASE i.
- $R_{i,k}$ is end-to-end response time of request class k in VASE i, formulized as:

$$R_{i,k}=\frac{1}{\Lambda_{i,k}}\left(\lambda_{i,k,0}\cdot R_{i,k,0}+\sum_{j=1}^{M}\sum_{w=1}^{c_{i,j}}\lambda_{i,k,j,w}\cdot R_{i,k,j,w}\right)$$

- $\lambda_{i,k,j,w}$ is arrival rate of VM w in requests class k in the jth tier in VASE i.
- $m_{i,k}=u_{i,k}\big/R_{i,k}>0$, $-m_{i,k}$ refers to slope of utility function $u_{i,k}$.
- $u_{i,k}(x)=\dfrac{bVal-x}{bVal-wVal}\in[0...1]$, here, x equals to $R_{i,k}$, $bVal$ is 0, $wVal$ is $\overline{R}_{i,k}$.
- $A_{i,k}$ is the availability of VMs for request class k in VASE i, formulized as:

$$A_{i,k}=\prod_{j=0}^{M}\left(1-FV_{i,k,j}\right)=\prod_{j=0}^{M}A_{i,k,j}$$

- $LV_{i,k}$ is the loss value of failure for request class k in VASE i.
- To request class k, $x_{i,k}$ is the number of refused requests which can lead to penalty. $d_{i,k}$ is each unit penalty of requests class k in VASE i.

3 The System Performance Model

The section mainly aims on online VASEs, so response time is viewed as main performance metric to measure quality of services in VASE. To make cloud IaaS resources can be allocated in a dynamic way according to requirements of clients, we propose VMs dynamic provisioning model, as showed in Fig. 1.

In cloud computing environment, a large amount of clients request resources in cloud IaaS. The hybrid queueing network is adopted to establish performance resolu- tion model for our system. In the manner of request class k, clients' requests arrive in cloud data center and visit services in VASE i, and the requests rate is $\lambda_{i,k}$. The locus analysis of actual network business website [6] has shown that network workload conforms to Poisson distribution. So it is assumed that requests arrival stream are Poisson distribution, and the interval of arrival time conforms to exponential distribution.

Let $\Lambda_{i,k}=\lambda_{i,k,0}$, where, $p_{i,k,j}$ refers to probability of request class k which finishes serving requests of the jth tier and return to initial state to reserve requests. $p_{i,k,q}^{(un)}$ represents probability of request class k which finishes serving requests of the jth tier and arrive in the j+1 tier in VASE i, meanwhile, the probability of $1-p_{i,k,q}^{(un)}$ of

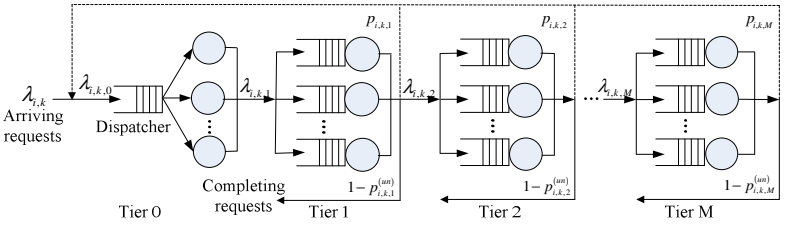

Fig. 1. Network queueing model

clients in the *j*th tier in VASE *i* finish the process of request class *k* and return. $\lambda_{i,k}$ refers to the arriving requests rate of request class *k* in VASE *i*. As showed in Fig. 1,

$$\lambda_{i,k,0} = \lambda_{i,k} + \lambda_{i,k,1} p_{i,k,1} + \lambda_{i,k,2} p_{i,k,2} + \cdots + \lambda_{i,k,M} p_{i,k,M} \tag{4}$$

Let $M_{i,k} = M$ and $j = 0$, then $\lambda_{i,k,1} = p_{i,k,0}^{(un)} \lambda_{i,k,0}$, $\lambda_{i,k,2} = \left(p_{i,k,1}^{(un)} - p_{i,k,1}\right) \cdot \lambda_{i,k,1}$, $\lambda_{i,k,3} = \left(p_{i,k,2}^{(un)} - p_{i,k,2}\right) \cdot \lambda_{i,k,2}$, ... , $\lambda_{i,k,M} = \left(p_{i,k,M-1}^{(un)} - p_{i,k,M-1}\right) \cdot \lambda_{i,k,M-1}$, i.e., $\lambda_{i,k,j} = \left(p_{i,k,j-1}^{(un)} - p_{i,k,j-1}\right) \cdot \lambda_{i,k,j-1}$, and $p_{i,k,0}^{(un)} = 1$, $0 \le p_{i,k,j-1}^{(un)} \le 1$, $p_{i,k,M} = p_{i,k,M}^{(un)}$, $(\forall j \in [1, M])$.

Then

$$\lambda_{i,k,0} = \lambda_{i,k} \Bigg/ \left(1 - p_{i,k,1} - \sum_{j=2}^{M_{i,k}} \left(p_{i,k,j} \cdot \prod_{q=1}^{j-1} \left(p_{i,k,q}^{(un)} - p_{i,k,q}\right)\right)\right) \tag{5}$$

Here, on-demand dispatcher (ODD) ($j = 0$) is modeled as an *M/M/c* system model, in which, there are *c* schedulers for VMs all together. The effective utilization rate of ODD is ensured as 60%~80%. According to Little's law [7], we can compute the average end to end response time of ODD in VASE *i*, namely $R_{i,k,0}$.

Then establish multiple *M/G/*1 performance resolution models for other tier in multi-tier VASE *i*. The common distribution requirement is solved by the approach of embedding Markov chain [8]. It is assumed that clients' requests are scheduled arrive in VM *w* at the rate of $\lambda_{i,k,j,w}$, $1 \le w \le c_{i,j}$. We can compute the value of average response time of every tier in VASE *i*, $1 \le j \le M$, namely $R_{i,k,j,w}$. What's more, $\rho_{i,k,j} = \lambda_{i,k,j} \Big/ \sum_{w=1}^{c_{i,j}} \mu_{i,k,j,w} < 1$ is the utilization rate of resource (e.g., CPU) allocated to VMs in every tier of VASE *i*. We further developed meta-heuristic solutions based on the hybrid stochastic optimization algorithm. Here, we would not discuss the details and results of the optimization algorithm.

4 Performance Evaluation Result and Analysis

The performance of virtualized resources optimization method based on SLA in cloud data center is evaluated by concrete experiments in this section. The total profit of

cloud IaaS providers can be maximized. What's more, the virtualization technologies are employed to isolate different applications and add availability rate, and the performance of cloud IaaS can be further improved. We adopt another two resource provisioning methods (DPM-RA and Stat-RA) for comparing with our method (DVM-Pro), which verity the effectiveness of DVM-Pro.

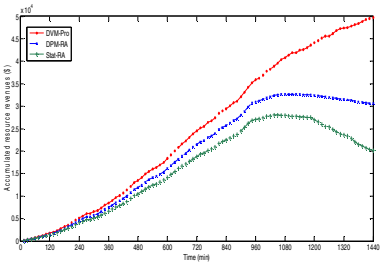

Fig. 2. Accumulated resource revenues

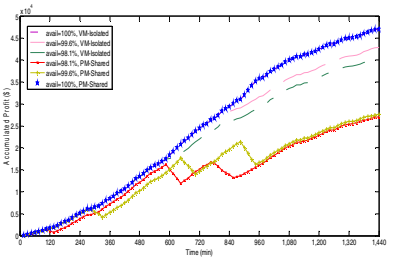

Fig. 3. Accumulated profits

Fig.2 shows the variation of accumulative net earnings of resources with time. It can be clearly found that when DVM-Pro method is used, though requests workload present obvious concussion, the profit keeps on increasing in a stable way all the time. When DPM-RA method is used, the rising trend is not effectively capture in the time of 960min~1440min, which causes a lot of punishment, so net earnings dropped significantly. Similarly, in accordance with the expected, the result of Stat-RA method is the worst. The reason is that its provisioning amount of resources never variations, so the changing trend of requests workload can not be captured completely. So the profit value presents the obvious tend of declining in this time of 960min~ 1440min while the method of DVM-Pro still keeps higher profit value, i.e., the adoption of DVM-Pro ensures to maximize profit of cloud IaaS providers.

In order to evaluate the performance isolation effect and efficiency when VMs are used as servers, a series of experiments are conducted to evaluate the impact that availability of system makes on performance when sharing and isolating services environment. In the experiments, assume availabilities of an application instance in every tier are equal. The length of time is set to 15 minutes. To compare variations of profit, the experiment for comparison is conducted between isolation and share of environments. The result is illustrated in Fig.3. It can be found that the accumulative global profit of shared environment is equal to that of isolated environment when failures do not happen. However, with the decrease of availability, the performance of shared environment declines more quickly than that of isolated environment. It can also be found that with the increase of availability, isolated environment can serve more requests than shared environment, so superiority of isolated environment is more obvious. The results show that virtualized infrastructure resources can provide the ability of performance isolation. Furthermore, the availability and performance of the whole system can be improved.

5 Conclusions

In this paper, the IaaS is divided flexibly in dynamic and variable cloud computing environment, and the system resources are optimized by employing VMs appropriately, thus the various SLA requirements of clients can be served more effectively. Firstly, a multi-tier architecture oriented virtualized application performance resolving model is developed according to the dynamic characteristics of VMs and SLA restriction of clients. According to this model, virtualized resources can be allocated intelligently for various requests from clients of different levels. Therefore, the resource profit of cloud IaaS providers can be maximized together with expense of operation and maintenance reduced. Simulation experiments show that the proposed method can provide appropriate services for clients of different levels. The proposed fine-grained and dynamic provisioning of resources based on VMs can ensure high profit and the SLA requirement of clients in every application even though workload varied consequently. At the same time, the evaluating result of availability shows that the method based on virtualized resources provisioning can not only support isolated application services, but also reduce the occurrence of failure, then the availability and performance of all cloud IaaS can be improved correspondingly.

Acknowledgments. This work was supported in part by the IBM Ph.D. Fellowship, and the National Natural Science Foundation of China under Grant 60872040.

References

1. Wang, L.Z., Laszewski, G.V., Younge, A., et al.: Cloud Computing: a Perspective Study. Journal of New Generation Computing 28, 137–146 (2010)
2. White, S.R., Hanson, J.E., Whalley, I., et al.: An architectural approach to autonomic computing. In: Pro. of the International Conference on Autonomic Computing, New York, USA (2004)
3. Kalyvianaki, E., Charalambous, T., Hand, S.: Self-Adaptive and Self-Configured CPU Resource Provisioning for Virtualized Servers Using Kalman Filters. In: Pro. of the 6th International Conference on Autonomic Computing, Barcelona, Spain (2009)
4. Tesauro, G., Kephart, J.O.: Utility functions in autonomic systems. In: Pro. of the First IEEE International Conference on Autonomic Computing, New York, USA (2004)
5. Barham, P., Dragovic, B., Fraser, K., et al.: Xen and the Art of Virtualization. In: Pro. of the Nineteenth ACM Symposium on Operating Systems Principles, New York, USA (2003)
6. Menascé, D.A., Bennani, M.N.: Autonomic virtualized environments. In: Pro. of IEEE International Conference on Autonomic and Autonomous Systems, Silicon Valley, California, USA (2006)
7. McKenna, J.: A Generalization of Littles Law to moments of queue lengths and waiting times in closed, product form queueing networks. Journal of Analysis and Optimization of Systems 111, 1000–1011 (1988)
8. Meyer, P.A., Smythe, R.T., Walsh, J.B.: Birth and death of Markov Processes. Probability theory (1972)