A RBRM Approach for Virtual Desktop Cloud Computing

Rasmi.K
Department of Computer Science and
Engineering, Karunya University,
Coimbatore,
Tamilnadu -641 114, India

Vivek.V
Department of Computer Science and
Engineering, Karunya University,
Coimbatore,
Tamilnadu -641 114, India

ABSTRACT

The speciality of virtual desktop cloud computing is that user applications are executed in virtual desktops on remote servers. Resource management and resource utilization are very much significant in the area of virtual desktop cloud computing. Handling a large amount of clients in most efficient manner is the main challenge in this field. This is because we have to ensure maximum resource utilization along with user data confidentiality, customer satisfaction, scalability, minimum SLA violation etc. Assigning too many users to one server may cause overloaded condition and which will lead to customer dissatisfaction. Assigning too little load will result in high investment cost. So we have taken in to consideration these two situations also. Here the proposed Rule Based Resource Management (RBRM) scheme assures the above mentioned parameters like minimum SLA violation. The concept of virtual desktop cloud computing is extended to a hybrid cloud environment. This is because to make the private cloud scalable. And priorities are assigned to user requests in order to maintain their confidentiality. The results of the paper indicate that by applying this RBRM scheme to the already existing overbooking mechanism will improve the performance of the system with significant reduction in SLA violation.

KEYWORDS

Virtual desktop cloud computing, Resource management, Resource Overbooking, Rule Based Resource Management, SLA violation.

1. INTRODUCTION

The major part of computation and storage components in virtual desktop cloud computing are shifted from the client device to the network. The user applications are executed in a virtual desktop (i.e. VD) of remote server. So the client device deals with user interaction.

There are some highly beneficial features regarding the virtual desktop cloud computing like lower client hardware requirements, the end user has no tension regarding the difficult installation and configuration of softwares. The central management system for virtual desktops results in lower IT management costs.

In figure 1 [1], the system architecture for virtual desktop cloud computing in order to support remote desktops as a cloud service is shown. When a user connects to the service, the service manager handles the authorization, authentication and accounting information. We have to assume that there is some profiling database information for each of the user which is maintained by the service manager.

Here we are assuming that the profiling information is already given. After all the procedures explained above a thin client protocol session will be established between the user and the virtual desktop.

It is very much important to provide crisp interactivity for the above mentioned service. It should be even for every client related to that service. The client's virtual desktop's need enough resources available immediately to ensure quick interactivity. For that a good resource management system should be implemented. The resource management system should be one with minimum SLA violation [2].

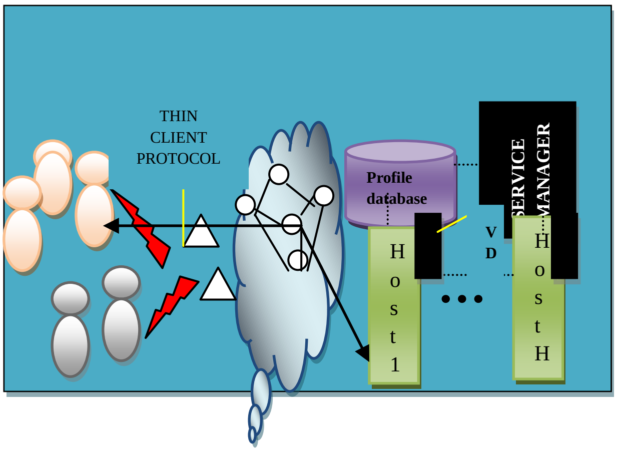

Figure1. Virtual desktop cloud computing as a cloud service.

Resource provisioning is important issue in cloud computing and in the environment of heterogeneous clouds [3]. The private cloud with confidentiality data has to configure accordingly with users need. But the real fact is that scalability of the private cloud limited. If the resources of private clouds are busy with fulfilling other requests then new requests are not able to be fulfilled. As a result, new requests are kept in waiting queue to process later. It creates lots of delay in order to fulfil these requests and is costly. In this paper we propose a Rule Based Resource Manager for the Hybrid cloud environment, which increase the scalability of private cloud on-demand.

The contributions of this paper are (i) resource overbooking technique (ii) resource allocation mechanism (iii) resource reallocation mechanism and (iv) rule based resource manager in order to minimize the SLA violation in the already explained system [1]. The remainder of this paper is organised as follows. In section 2 some already existing systems in relation to resource management strategies are discussed. Section 3 discusses about the overall system model and in section 4 it explains the simulation environment in which the system performance is evaluated. Sections 5, 6 and 7 are related to resource overbooking, resource allocation and resource reallocation respectively. In section 8 the idea of Rule-Based Resource Manager is expanded. The experimental outcomes and observations are given in section 9 and finally conclusions are drawn in section 10.

2. RELATED WORK

Cloud computing that is based on resources acquired on demand is generating a great deal of interest among service providers and consumers. Here in this section we are going to analyse the resource allocation and reallocation (load balancing) methods that are already present in the cloud environment and their basic principle. In this section we are going to analyse different resource management strategies and their positive as well as negative aspects.

In [4] it proposes a new autonomic workload provisioning that addresses the challenges of enterprise grids and clouds. The main aim of this mechanism is that to improve the resource utilization and which is achieved with the help of reducing the overprovisioning. It can be reached through two levels. In order to reduce the overprovisioning caused by the difference between the virtual resources allocated to VM instances and those requested by the individual jobs a new mechanism is introduced. And this technique is based on decentralized online clustering, and it helps to characterize the resource requirement classes and it is used for proactive VM provisioning. This paper also introduced another way of resolving the overprovisioning problem which may be happened due to inaccuracies in client resource requests. This paper also explored the use of workload modelling techniques and their application. The mechanism for dynamic and decentralized VM provisioning monitors the flow of arriving jobs from different queues in a decentralized manner during ongoing analysis windows of duration in the order of the startup time of new VMs.

The resource allocation is taken into consideration generally the parameters like CPU utilization, memory utilization and throughput etc. The cloud environment has to take into consideration all these things for each of its clients and could provide maximum service to all of its clients. In [5] it suggests that when we are taking the scheduling of resources and tasks separately it imposes large waiting time and response time. In order to overcome this drawback a new approach namely Linear Scheduling for Tasks and Resources (LSTR) is introduced.

Here scheduling algorithms mainly focus on the distribution of the resources among the requestors which will maximize the selected QoS parameters. The QoS parameter selected in this approach is the cost function. The scheduling algorithm is designed based on the tasks and the available virtual machines together and named LSTR scheduling strategy. This is designed in order to maximize the resource utilization.

In [6], it talks about the live migration of the virtual machines. In this paper they suggest that migrating the operating system instances across distinct physical hosts is a useful tool for the administrator of data centers and clusters. It also provides a separation between hardware and software and provides fault management, low level system maintenance and load balancing. Here an approach namely “pre-copy approach” is introduced. In this approach pages of memory are iteratively copied from the source machine to the destination host and in addition there is a fact that all these things are done without ever stopping the execution of the system. Pagelevel protection hardware is used to make sure that a consistent snapshot is transferred. For controlling the traffic of other running services a rate-adaptive algorithm is used. And during the final phase it pauses the virtual machine and copies any remaining pages to the destination and after that resumes the execution there. The factors affecting the total migration are link bandwidth, migration overhead and page dirtied rate [7]

Roy et al. [8] describes about the cost based workload provisioning and “just- in- time resource allocation”. Workload Prediction is the prediction of the workload on the application and estimation of the system behavior over the prediction horizon is using a performance model. Here optimization of the system behavior is carried out by taking into consideration the minimization of the cost incurred to the application. This cost can be a combination of various factors such as cost of SLA violations, leasing cost of resources and a cost associated with the changes to the configuration. The advantage of such types of methods is that it can be applied over various performance management problems from systems with simple linear dynamics to systems with complex dynamics. The performance model can also be varied and affected with system dynamics as conditions in the environments like workload variation or errors in the system change.

MiyakoDori [9], is a memory reusing mechanism to reduce the amount of transferred data in a live migrating system. When we are considering the case of dynamic VM consolidation, virtual machines may migrate back to the host where it was once executed and so the memory image in that host can be reused, thus contributing to shorter migration time and greater optimizations by VM placement algorithms. In [10] it shows that this technique enables to reduce the total migration time. In this technique dirty pages alone need to be transferred to the former host.

The aim of load balancing in the cloud computing environment is to provide on demand resources with high availability. But often load balancing approaches suffer from various overheads. And they also fail to avoid deadlocks when there more requests competing for the same resource at the same time when the available resources are insufficient to service the arrived requests. The ELBA approach [11] using the efficient cloud management system helps to overcome the aforementioned limitations. This approach yields less response time compared to the existing approach. Less response time reduces job rejections and accelerates the business performance.

In our work we propose a Rule-Based Resource management system which is able to provide increased resource utilization, increased security consideration, scalability and minimized SLA violation.

3. SYSTEM MODEL

We assume that there are M hosts in the datacenter and N users are subscribed to the service in the hybrid environment. All hosts are having the limited processing power, which is modelled as FLOPS (Floating Point Operations per Second). Based on the FLOPS requirements the resources are allotted which will be less than their worst case requirement. This reservation of resources in advance can be named as Resource overbooking which will be discussed in the following section. The overall system model is given in figure 2.

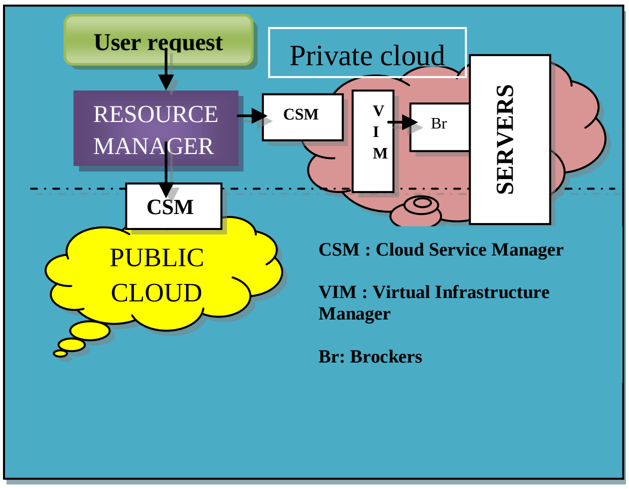

Figure2. Rule-Based Resource Management system

Here a Rule-Based resource manager is implemented which will monitor the priority of the incoming requests. If the priority is high it will certainly be allotted to the private cloud. If the priority is low then it will be forwarded to the public cloud. The resource allocation and reallocation based on the “3-Rules” are discussed in the following sections. And here one more level of SLA optimization is introduced in addition to the one proposed in [1].

4. SIMULATION ENVIRONMENT

Our simulation environment is an extension of CloudSim 3.0 toolkit [12]. There are some unique features for CloudSim, they are (i) availability of virtualization engine, which aids in creation and management of multiple, co-hosted and independent virtualized services on a data center node (ii) greater flexibility to switch between time-shared and space-shared allocation of processing cores to virtualized services. These popular features of CloudSim would speed up the development of new resource allocation policies and scheduling algorithms for Cloud computing [13].

CloudSim possesses a realistic model to migrate virtual desktops from one host to the other. And the period of duration depends on the time taken for migrating the assigned memory to the virtual desktop. Here a Rule-Based Resource Management system is implemented which will perform the VM scheduling according to the priority of the incoming requests.

5. RESOURCE OVERBOOKING

Resource overbooking is a technique that can achieve an increase of the average utilization of hosts in a data center by reserving fewer resources than needed in worst case; at the cost of a larger probability that a virtual desktop does not receive all requested resources. Since a lot of virtual desktops can be allocated to a host, the cost for the service provider associated with investment in hardware equipment, server maintenance cost and energy cost can be reduced.

The overbooking degree can be defined as the percentage of resources that are not reserved for a virtual desktop. For instance, if the 90th percentile of the resource usage distribution from the virtual desktop profile is reserved, then the overbooking degree is 10%. The overbooking factor should provide balance between economic considerations and performance objectives such as delay and packet loss [14].

The aim of the resource overbooking technique discussed in this section is that to increase the system utilization with little impact on the user experience. This is achieved by reserving some amount of resources for the virtual desktop. And this reservation is done based on the profile information of a virtual desktop. Here we are assuming that the profile information is already available for us. The relevance of resource overbooking on the user experience can be represented based on the number of SLA violations experienced by the user’s virtual desktop.

6. RESOURCE ALLOCATION

When there are N hosts present in the system, a decision has to be made that on which host the virtual desktop of arriving application requests should be allocated.

6.1 Allocation algorithm

The allocation algorithm is considered as a classical bin-packing strategy [15]. The allocation algorithm (i) calculates for all the hosts i the corresponding cost C_i. (ii) after that it selects the host with the best cost.

The cost calculating function can be represented as

$$C_{i\,=}\ \alpha \times Prob[No.of\ SLA\ violations\ on\ host\ i \geq 1 \mid VDj \in Hi\,]$$

$$+\beta \times Prob\ [next\ user\ rejected\ by\ host\ i \mid VDj \in Hi]$$

The first term indicates the probability that the sum of the resource requirements of virtual desktops executed on the host is greater than the total amount of resources available on that host. The second term indicates that there are no enough resources available for the next user request in the host in which it gets allocated also.

Here the distribution of resources taking place based on the best-fit strategy. That is here selects the host for which the amount of available resources is closest to the requested amount of resources. As a result virtual desktops are gathered as much possible on a single host as possible and it leaves the other hosts idle. This strategy is very much efficient compared to the random allocation algorithm [16].

The cost-based allocation algorithm has to calculate the cost C_i for every host *i* and after that it has to select the host for which C_i is minimal. The complexity of the cost based allocation algorithm is determined as $O\,[\sum_{i=0}^{N} \#Hi\,]$ with #Hi the cardinality of the set Hi or in other words the number of VDs allocated to host i.

7. RESOURCE REALLOCATION

The reallocation action is meant for the optimization of SLA violations that are going to take place during the system interaction with the user. We have to give maximum customer satisfaction by reducing the SLA violations caused by the system. In order to implement this optimization we have to take into consideration two problems associated with that. First we have to take into consideration the cost [18] of reallocating a virtual desktop. Then the reallocation algorithm has to decide which virtual desktop need to be reallocated. The allocation algorithm discussed in section 6.1 can be used for placing the virtual desktops in the target hosts. The second point we have to determine is that when the reallocation algorithm should be activated.

7.1 Reallocation algorithm

By rebalancing virtual desktops among the available hosts, the probability on SLA violations can be reduced. In this algorithm, it does not adopt the number of FLOPS reserved for the virtual desktops, it only migrates them to hosts which has more free resources than the current host. This means that the target hosts have more free resources that can be shared among virtual desktops requesting more resources than reserved. This can result in even faster responses of the applications executed in the virtual desktop and hence a better user experience.

The target of the reallocation algorithm is to achieve a smaller probability on SLA violations in the data centre with as less reallocations as possible. Therefore, the algorithm starts with sorting the hosts by descending probability of SLA violations on the host. The first host in the list, H, is the host with the highest probability on SLA violations. To reduce the load on this host, VDs should be moved to other hosts with as few reallocations as possible. Therefore, for every virtual desktop on host H, the probability on SLA violations without that virtual desktop is calculated. The virtual desktop that can reduce the load on the host the most when it would be moved to another host, is selected for reallocation. Next, the cost-based allocation algorithm introduced in the previous section is used to select a new host for this virtual desktop. The six major steps for the reallocation procedure can be explained as follows.

- Starts with sorting the hosts by descending probability of SLA violations on the host.
- The first host in the list, H, is the host with the highest probability on SLA violations.
- To reduce the load on this host, VDs should be moved to other hosts with as few reallocations as possible.
- For every virtual desktop on host H, the probability on SLA violations without that virtual desktop is calculated.
- The virtual desktop that can reduce the load on the host the most when it would be moved to another host, is selected for reallocation.
- The cost-based allocation algorithm (introduced in the resource allocation section) is used to select a new host for this virtual desktop.

Since we have to reduce the probability on SLA violations, a high ratio for α/ β is preferred.

8. RESOURCE MANAGER

Resource management is very important issue in cloud computing and within the surroundings of heterogeneous clouds. And the private cloud with confidentiality data configure related to users need. But the resource availability of the private cloud restricted. If the resources private clouds are busy in fulfilling some other requests then the arriving new request cannot be accomplished [17]. As a result new requests are kept in waiting queue to process in a later period. It will take lot of time to manage these requests and this method is costly. Rule Based Resource Manager proposed by Rajkamal et.al, 2012, for the Hybrid environment, which increase the scalability of private cloud on-demand and minimizes the cost.

8.1 Rule-Based Resource Manager

“Rule-Based Resource Manager” is introduced in the hybrid cloud environment. The “Rule-Based Resource Manager” successfully utilizes the private cloud resources and considering the security requirements of applications and data. With this resource manager a private cloud can be scaled up-to allocate resources on demand even if private cloud overloaded. And the scalability beyond the capacity of private cloud is achieved by using public cloud resources. And these decisions are made according to some set of “rules”. The overall system architecture for “Rule-Based Resource Manager” is shown in figure 2.

Here some priority values are taken into consideration. It is based on categorizing the user’s request into two types based on the resource requirements that is critical data processing and data security. Two types of priorities are assigned ie. high priority and low priority. If the users need to perform critical data processing and the security demand is high then the request is classified as high priority. The high priority request always access resources from the private cloud itself because it has confidential (secure) information. But the low priority cloud can be executed from either private or public clouds.

But if the private cloud resources are available it must be used first. Sometimes high priority requests which have to be fulfilled by private cloud but its resources are already assigned to fulfil previous requests of low and high priority. In this section we have to find the already allocated low priority requests and reallocate those low priority requests for which the remaining cost is minimal on public cloud. And if all the allocated requests in the private cloud are having high priority then sort them and reallocate the request with minimum SLA violation to the public cloud.

The three important rules regarding the “Rule-Based Resource Manager” are given below.

```
RESOURCE MANAGER (NEW-REQUEST)
  {
RULE 1:
If(NEW-REQUEST Requirement <= Available)
{
Then :
ALLOCATE NEW-REQUEST on Private cloud
Response.Redirect (Private cloud);
}
RULE 2:
If(NEW-REQUEST Requirement>Available && NEW-REQUEST ==HIGH-PRIORITY )
{
Check LOW-PRIORITY request on private server
If (LOW-PRIORITY.REQUEST.count<0)
{Sort the HIGH-PRIORITY requests on private cloud and REALLOCATE the request with less SLA violation to public cloud}
Else { REALLOCATE existing LOW-PRIORITY-REQUEST to public server}

Then:

Handle NEW-REQUEST to private server

}
```

RULE 3:

```
If(NEW-REQUEST Requirement>AVAILABLE && NEW-REQUEST==LOW-PRIORITY)

{ Redirect request to public cloud }

}
```

9. EXPERIMENTAL RESULTS

The main functions that are implemented under the rule based resource management scheme are as follows.

- We have to create two types of clouds i.e. private cloud and public cloud.
- Each new request has to be assigned with some set of priority i.e. high priority and low priority according to their nature. That the requests which are critical and confidential can be considered as having high priority. And the remaining requests are having low priority.
- Perform resource overbooking by using the same set of rules performed for the previously implemented system [1].
- Then the resource manager has to analyze the incoming requests for their priority and which mode of operation they prefer.
- Perform the steps that are discussed in section 8.1.
- Note that the allocation and reallocation that are described under "Rule-Based Resource Manager" are same as that shown in sections 6.1 and 7.1.
- Two levels of SLA optimization is carried out. That is during the allocation and the reallocation phase.

Finally a system which is scheduled based on the priority of the requests is created and we have to prove that the average SLA violation imposed by the Rule-Based resource management scheme is very less as compared to the previously implemented system [1].

We can also find that the resource utilization is high and the system can be extended for security concerns because confidential information are having high priority and they will redirect to the private cloud in most of the normal situations. The high priority requests are only replaced when there are no allotted low priority requests in the private cloud. But there is one added advantage that it will only replace the requests which will cause less SLA violation.

Here a comparison is made between the system which is already implemented in [1] and the proposed RBRM approach. A comparison is made between the SLA values after reallocation is performed. The results are shown in the graphs in figure 3.

10. CONCLUSION

The concept of virtual desktop cloud computing, i.e. executing applications in virtual desktops on remote servers, is very interesting because it enables access to any kind of application from any device. Here the optimization of the quality experienced by the customers by optimizing the distribution of the customers among the available hosts is carried out. In order to improve the quality experienced by the users the number of SLA violations experienced is reduced. And also implemented the minimum migration policy for each of the virtual desktops.

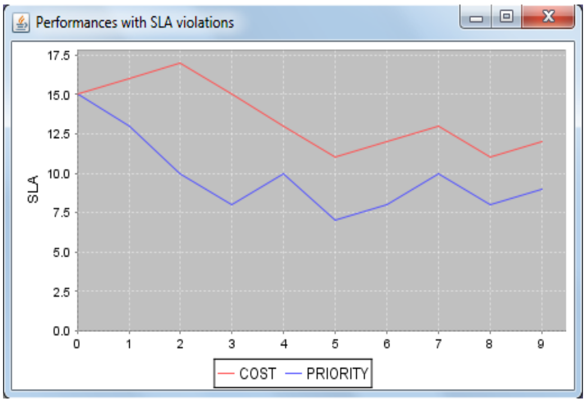

Figure 3. Performance comparison of RBRM (priority-based) approach with the cost-based strategy.

First, an optimization has been introduced to increase the average utilization on a single host. It was shown that the proposed overbooking approach, together with an advanced scheduler.

To further optimize the quality of the service, a reallocation algorithm has been proposed to rebalance the virtual desktops among the available hosts after a busy period. After a busy period, some hosts could still be fully loaded while other hosts are almost not loaded and therefore, reallocating virtual desktops from fully loaded hosts to not loaded hosts can minimize the probability on SLA violations. The reallocation is done with the help of minimum migration policy for each of the virtual desktop.

The enhancement can be made to the system by extending the concepts to the hybrid cloud environment and also can introduce the concept of "Rule-Based Resource Manager". It will help to migrate the requests according to their priority and also is highly scalable and provides high resource utilization with minimum SLA violation.

11. REFERENCES

[1] Lien Deboosere , Bert Vankeirsbilck ,Pieter Simoens , Filip De Turck , Bart Dhoedt and Piet Demeester, 2012,"Efficient resource management for virtual desktop cloud computing", The Journal of Supercomputing November, Volume 62, Issue 2, pp 741-767.

[2] W. Iqbal, M. Dailey, D. Carrera, 2009, "SLA-Driven Adaptive Resource Management forWeb Applications on a Heterogeneous Compute Cloud", CloudCom 2009, LNCS 5931, pp. 243–253.

[3] Jiang Dejun, Guillaume Pierre, Chi-Hung Chi, "Resource Provisioning of Web Applications in Heterogeneous Clouds".

[4] Quiroz A, Kim H, Parashar M, Gnanasambandam N, Sharma N, 2009, "Towards workload provisioning for enterprise grids and clouds",IEEE/ACM international conference on grid computing. pp 50-57.

[5] Abirami S.P. , Shalini Ramanathan, 2012 ,"Linear Scheduling Strategy for Resource allocation in Cloud Environment",International Journal on Cloud Computing and Architecture ,vol.2, No.1, February.

[6] Christopher Clark, Keir Fraser, Steven Hand, Jacob Gorm Hanseny, Eric July, Christian Limpach, Ian Pratt, Andrew Warfield, 2005, "Live Migration of Virtual Machines", 2nd Symposium on Networked Systems Design and Implementation (NSDI) , May

[7] Rakhi k Raj and Getzi Jeba Leelipushpam. P, 2012, "Live Virtual Machine Migration Techniques – A Survey", International Journal of Engineering Research and Technology, Volume 1 Issue 7, September.

[8] Nilabja Roy, Abhishek Dubey and Aniruddha Gokhale , "Efficient Autoscaling in the Cloud using Predictive Models for Workload Forecasting".

[9] Soramichi Akiyama, Takahiro Hirofuchi, Ryousei Takano, Shinichi Honiden, 2012, "MiyakoDori: A Memory Reusing Mechanism for Dynamic VM Consolidation", Fifth International Conference on Cloud Computing, IEEE 2012.

[10] Jyothi Sekhar, Getzi Jeba, S. Durga,2012, "A Survey on Energy Efficient Server Consolidation Through VM Live Migration", International Journal of Advances in Engineering & Technology, November.

[11] Rashmi. K. S, Suma. V and Vaidehi. M, 2012, "Enhanced Load Balancing Approach to Avoid Deadlocks in Cloud", Special Issue of International Journal of Computer Applications (0975 – 8887) on Advanced Computing and Communication Technologies for HPC Applications - ACCTHPCA, June.

[12] Calheiros R N, Ranjan R, Beloglazov A, De Rose CAF, Buyya R, 2011 CloudSim: a toolkit for modelling and simulation of cloud computing environments and evaluation of resource provisioning algorithms.Softw Pract Exp 41(1):23–50.

[13] Rajkumar Buyya, Rajiv Ranjan ,Rodrigo N. Calheiros , "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities".

[14] Feng Huang, 2006, "A Selective Approach to Bandwidth Overbooking".

[15] Martello S, Toth P, 1990, Knapsack problems: algorithms and computer implementations. Wiley,New York

[16] Xing Xu, Hao Hu, Na Hu, Weiqin Ying , 2012, "Cloud Task and Virtual Machine Allocation Strategy in Cloud Computing Environment"Network Computing and Information SecurityCommunications in Computer and Information Science Volume 345, pp 113-120.

[17] Rajkamal K. Grewal , Pushpendra K. P., 2012, "A Rule-based Approach for Effective Resource Provisioning in Hybrid Cloud Environment." International Journal of Computer Science and Informatics ISSN (PRINT): Vol-1, Iss-4, 2231 –5292.

[18] Voorsluys W, Broberg J, Venugopal S, Buyya R , 2009, "Cost of virtual machine live migration in clouds: A performance evaluation". In: Proceedings of the 1st international conference on cloud computing, pp 254–265.