INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

A NOVEL RESOURCE PROVISIONING APPROACH FOR VIRTUALIZED ENVIRONMENT

Abinaya R[1]
PG Scholar, Computer Science and Engineering,
[1]abiarun.cse@gmail.com
Preethi Harris[2]
Associate Professor, Department of IT,
[2]preehars@gmail.com Sri Ramakrishna Engineering College, Coimbatore,
Tamilnadu, India.

Abstract: - The cloud computing is an emerging technology that relies on underlying resource provisioning, which is achieved through virtualization. Virtualization is commonly implemented with Hypervisor to handle Virtual Machine (VM). In this paper, the decision engine is used to provision the resource, which continuously learns take action to change the amount of provisioned resources according to the current performance status of the system. The Q-learning algorithm mainly focuses on provisioning the resource elastically to the customer based upon the workload and best fit is used to allocate the resources. This concept has been used to increase performance and reduces response time.

Keywords: Cloud Computing, Resource Provisioning, QoS, Q-Learning, best-fit.

I. INTRODUCTION

Cloud computing is a most promising emerging technology in the modern world having a broad array of web-based services and its main aim to allow users to obtain a wide range of functional capabilities, it offers pay per use based services such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) through different cloud providers [1]. Cloud provider provides the computing resources dynamically to the Cloud users based on their requirements over the internet. The dynamic resource provisioning enables service providers to manage and provisioning of their resources for each individual user where resources are normally provided to serve requests in the form of VM.

Virtualization is the key feature of cloud computing. It is a technique which allows multiple Operating System (OS) to run simultaneously on a single Physical Machine (PM). This is implemented through the hypervisor. A hypervisor is computer software that creates and run virtual machine

The problem of VM provisioning in clouds has been investigated from different points of view and many research studies have been done on resource provisioning mechanisms for cloud systems. This is a challenging task to maintain the required Quality of Service (QoS) level of service to fulfil the expectations of cloud consumers [2].

The resource management in cloud which observed that there are very few Cloud Service Providers (CSPs) can only provision few services. A dynamic VM scheduling technique is proposed, which minimizes the resource provisioning time of user request, provide efficient utilization of resources, increases throughput. It is very important to schedule workloads and it directly affects the entire resource utilization. The goal of workload scheduling is to optimize time and cost and improve resource utilization by organizing and optimizing the scheduling process.

In this paper, the proposed work is about resource provisioning technique based on best fit resource allocation to lowers the service cost and reduces the execution time [3]. To take care of users needs which employs agents in each VM to monitor the resource encompasses collecting information about resource and using this information to make decisions related to other components in the cloud environment. The agents collect the information about each VM from time to time and finally send the information to the scheduler for workload characterization [4]. Scheduler uses the best fit allocation strategy for allocating the resource to each VM. It collects all workload details about each VM to optimize resource provisioning [4]. Then it sends the workload details to decision engine to make a decision for provisioning the resource to the needed user.

II. LITERATURE SURVEY

Resource provisioning techniques in cloud computing environment is suggested by Bhavani and Guruprasad [5]. The resource provisioning techniques is suited for web applications to scale up or down where the response time is minimized and it is considered as one of the important factors. For web applications guarantying average response time is difficult for cloud provider because traffic patterns are highly dynamic and difficult to predict accurately and also because of the complex nature of the multi-tier web applications it is difficult to identify bottlenecks and resolving them automatically. This resource provisioning technique proposes a working prototype system for automatically detects and resolution of bottlenecks in a multi-tier cloud hosted web applications. This improves response time and also identifies over provisioned resources.

Model based self-adaptive resource allocation in virtualized environments is suggested by Huber et al [6]. A novel approach to self-adaptive resource allocation in virtualization environment. It is used to estimate the effects of changes in user workloads as well as predict the effect of respective reconfiguration actions, which is taken to manage performance goals of different workloads or inefficient resource usage. This approach is applied to react on changes during runtime to find efficient resource allocation while satisfying high performance.

Dynamic scaling of web applications in a virtualized cloud computing environment is suggested by Chieu et al [7]. It is based on thresholds approach in a virtualized cloud computing environment. In this work, the front-end load balancer is used in scaling approach to scale up or down for balancing user request to web applications. They were introduced a scaling algorithm it is based on threshold number of active sessions for dynamic resource provisioning of virtual machine elastically. Dynamically allocate and aggressively provision of resource to users is to be discussed based on-demand capability of the cloud environment. The existing works have been shows that maximum work has to be done in analysis of resource provisioning from the cloud service provider perspective.

An Metrics based workload analysis technique for IaaS is developed for cloud computing and suggested by Sukhpal Singh, Inderveer Chana [8]. It addressed problem faced for building an effective external controller is used for automated self-adaptive scaling of application deployed in the cloud environment. Different workload have been identified and categorized along with their characteristics and constraints. They recommended the proportional threshold based on virtualized resource provisioning approach that adjusting the resource dynamically up to a target range that is high and low threshold based the virtual machine instances are adjusted. Thus the relative effect of allocating resources becomes better as the number of accrued resources increases suddenly and eventually the results shows that being adaptive and more resource are efficient.

Elastic VM for cloud resource provisioning optimization is suggested by Herbst et al [9]. The rapid growth of E-Business and frequently changes in web-sites contents as well as customers' interest make it is difficult to estimate workload surge. To maintain a good and efficient QoS, system administrators must provision enough resources to support with workload fluctuations considering that resources over provisioning and under provisioning reduces business profits and degrades performance. In the elastic system architecture is created for dynamic resources management and applications optimization in virtualized environment. In this architecture, they implemented three controllers for CPU, Memory, and Application. These controllers run in parallel to guarantee client resources allocation and optimize application performance on co-hosted VMs dynamically. The controller aggressively allocates more memory when the previously allocated memory is close to the saturation then slowly decrease memory allocation if it is under loading.

A Reinforcement Learning Approach to Virtual Machines auto-configuration is suggested by Jia Rao et al [10]. The Reinforcement learning (RL) approach namely Virtual Configuration (VCONF) to automate the virtual machine configuration process by continuously learning the Resource performance relationship from the applied configurations and feedbacks. VCONF is reinforcement learning based automatic resource configuration system used to increase or decrease the CPU and memory capacity automatically. The central design of VCONF is the use of model based RL algorithms they define the reward signal based on summarized performance of each VM. In the controlled environment, our approach was able to find the best optimal configuration for single and multiple VMs running homogeneous workloads. The goal is to optimize the overall VM(s) performance VCONF manages the VM configuration by monitoring feedback from each VM.

III. PROPOSED SYSTEM

The users application are submitted to each VM and it requires dynamic provision of resource to virtual machine are provided to fulfil all types of requirements for user, which are configured with required virtual CPU and virtual memory.

In this paper, the proposed work is about resource provisioning technique in cloud to support the elasticity of providing the resource to the user have proven very effective [3,4]. To take care of users need to employs agents in each virtual machine to monitor the resource encompasses collecting information about resource and using this information to make decisions related to other components in the cloud environment. The agents collect the information about each VM from time to time and finally send the information to the scheduler for workload characterization [11]. Scheduler uses the best fit allocation strategy to allocate the resource to each VM. It collects all load details about each VM to optimize resource provisioning. Then it sends the workload details to decision engine to make a decision to provisioning the resource to the needed user.

The scheduler use the Dynamic Bin packing (DBP) technique that is best fit approach to allocate the resource to each virtual machine. The DBP is nothing but a variety of classical bin packing, which assumes that items may arrive and depart at correct times. This generally aims to minimize the maximum no of bins in cloud environment. In DBP the best fit approach is used to allocate the resource based upon the number of free Central Processing Unit (CPU) and amount of free memory. It results in a better scheduling of jobs by maintaining unbalanced CPU or memory usage across the machines with high resource provisioning [3]. By using the concept of resource provisioning it is an efficient way of managing and continuously provisioning the resource to the each virtual machine**.**

By using proposed algorithm, a novel resource management framework to ensure highest QoS and lowest request lost rate in the cloud computing system [12]. It utilize an aggressive strategy to encourage Q-learning to make bold attempts of provisioning more resource to user when facing rapidly increasing workload therefore it will increase the performance at high level in each time. The proposed algorithms have no additional overhead compared with existing efforts and only results in an acceptable waste of resources.

The proposed framework represent in figure 1 that learns and applies optimized resource provisioning technique in virtualized environments to continuously deals with the ever changing workloads [13]. By exploiting RL-based learning mechanism, can continuously learn the relationship between resource allocations and performance, and through long time running, the efficient resource allocations determined by Q-learning will improve in accuracy.

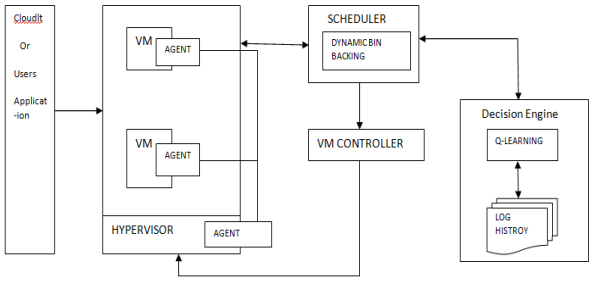

Fig.1 Proposed Framework

The best fit approach and Q-learning algorithm is used to allocate the resource to the customer at correct time to improve performance at high level. It provisions the resource accurately according to the VM specifications and avoids the waste of resource. By using this approach the cloud provider could minimize the response time, cost and high performance.

1. Agent

An agent that runs either in the VM that hosts the services or the PM that hosts the VMs to periodically collect the status and profile the workload in log history. The agent collects the status of the VM from time to time and sends it to the scheduler for workload characterization. The status is represented by performance metrics which are chosen as the workload signature to ensure QoS. It collects the performance metrics of the each VM at the system level regardless of the actual application it works with [3,4,13]. Hence, the agent is a general agent that can easily and quickly cooperate with any kinds of applications and it avoids intrusion to the hosted applications while monitoring it performance and thereby reducing overheads.

2. Scheduler

The scheduler use best fit to allocate the resource to each VM to utilize the resource efficiently. By using best fit approach is used allocate the resource based upon the user requirements to each virtual machine capacity[4,8]. It results in a better scheduling of user requests by managing unbalanced CPU or memory usage across machines with high resource requirements. If the workload increases suddenly then the scheduler will send the workload details to the decision engine to provision the resource continuously.

The scheduler calculates the Current Service Load Status (CSLS), which indicates the performance of a service during the previous time period, and then sends it to the decision engine to make a schedule decision [7]. If there is more than one service hosted in the cloud, CSLS is calculated separately for each service. Then consider the safe status as the CSLS to be maintained within a reasonable range. To be fair, we set the expected CSLS to 60 percent with a tolerance of 10 percent for all approaches in this experiment [6].

The scheduler must be able to allocate resources accurately and efficiently so that the performance can be maintained. To implement the action with a large range of alternatives to make the learning engine capable of adjusting the resource allocation efficiently enough to match the changing workload by using best fit approach. The actual action, which directly denotes how many VM instances have to be added or removed based upon the workload [10]. The user request are mentioned as input parameters as given in table 1. The input parameters are number of VMs, CPU, bandwidth and storage capacity. The bandwidth is used for resource transmission from the provider to the consumer.

Table 1 Input Parameters

Number of VMs	10
MIPS of CPU	500,1000,1500,2000,2500
Bandwidth	250,500,1000,1500
Storage capacity	2.5GB

3 Decision Engine

The decision engine is implemented as a learning based agent, which continuously learns take action to change the amount of provisioned resources according to the current performance status of the system. Then decision engine is used to make a decision to provision the resource when workload suddenly increases [4]. The decision engine contains log history to profile the workload metrics continuously. The engine then learns from the effect of applying the said actions. Typically, they choose the Q-learning algorithm as the implementation of RL.

At the core of the decision engine is an RL-based learner called RLearner. It first generates a decision and action to be taken to reconfigure the resource allocations based on these metrics and existing knowledge, and then updates the knowledge with a reward that indicates the effectiveness this decision.

3.1 Q-Learning

The Q-learning algorithm is proposed to accelerate the resource provisioning when the workload increases suddenly by continuously monitoring the each virtual machine. In order to allocate the resource to each VM by using best fit approach. This Q-learning is typically called as Model-free reinforcement learning technique [4]. By exploiting RL-based learning mechanism, can continuously learn the relationship between resource provisioning and performance, and through long time running, the efficient resource provisioning determined by Q-learning which

improves in accuracy. When such an action-value function is learned, the optimal policy can be constructed by simply selecting the action with the highest value in each state.

The Q-learning will encourage the decision engine to make bold attempts of provisioning more resource to user when facing rapidly increasing workload therefore it will increase the performance at high level in each time. It is proposed by defining Q function. Q(s,a) is a value function that denotes the maximum discounted cumulative reward when by performing an action a€A, the agent can move from state to state. Then each time the agent selects an action, and observes a reward and a new state that may depend on both the previous state and the selection action, Q is updated. The goal of agent is to maximize its total reward. It does this by learning which action is optimal for each state.

$$Q(s,a)=r(s,a)+\gamma V^{*}(T(s,a)) \qquad (1)$$

From equation (1), where V*(S) denotes the V(s) under optimal policy Q(s,a) is the value function that the agent has to learn through continuous trial. The r(s,a) is represents immediate reward(r) [4]. The Q function can be updated each time an action is applied to the system, using the immediate reward with the following method

$$Q(s_t,a_t) \leftarrow r(s,a)+\gamma \max Q(s_t+1,a_t+1) \qquad (2)$$

From equation (2), an action represents the resource management decision that enlarges or decreases the amount of allocated resources [4]. When selecting actions, the agent has a probability to randomly select one rather than choose the best one on the basis of current knowledge. The Q-value function of taking action a in state s can be defined as:

NOTATION	DESCRIPTION
s	State
r	Reward
a	Action
s_t	Current state
a_t	action state
γ	Discount factor
T	Decision time
α	Learning parameter

Q-Learning Algorithm

Initialize Q-table

Initialize all the resource states‘s’

For each state‘s’ do

a_t=get_action(s_t)

For life or until learning is stopped

Take action a_t observe r and S_{t+1}

Compute

Q(state, action) = R(state, action) + γ * Max[Q(next state, all actions)]

Update Log History

end for

End For

The Q(s,a) represents the state and action which gives immediate reward(r) for making an action and best utility(Q) for the resulting state.

The Q-learning can be used to find an optimal action selection policy for any given finite process. It works by learning an action value function that ultimately gives the expected utility of taking a given action in a given state and following the optimal policy thereafter one of the strengths of Q-learning is that it is able to compare the expected utility of the environment. Q-learning eventually finds an optimal policy. In each time the agent selects an action, and observes a reward and a new state that may depend on both the previous state and the selected action, Q is updated.

3.2 Log History

The aforementioned knowledge, which is actually the mapping between workload metrics and actions, is stored in a table called log history table. It is a memory-based table which can be updated every time an action is chosen and applied to the system.

The log history table is updated by an aggressive reward strategy to make the decision engine capable of adjusting the resource provisioning efficiently. Then log history has automated backups enabled. You can enable/disable automated backups at any time for an instance. For example, during a long-running task such as loading data, you might want to temporarily disable automated backups. The actual content of log history is stored in a memory table, which is implemented as a hash table for better space utilization.

IV. RESULTS AND DISCUSSIONS

The results show that best fit and Q-learning can significantly accelerate the resource provisioning process when faced with drastically increasing workload, and hence can ensure a high QoS and reduce response time. Moreover, the proposed algorithm introduces no additional overhead compared with existing efforts and only results in an acceptable waste of resource.

In figure 2, the representation of performance analysis for existing and proposed work in Central Processing Unit (CPU) execution speed. It increases the performance requirement in the first place so that the QoS can be kept at high level.

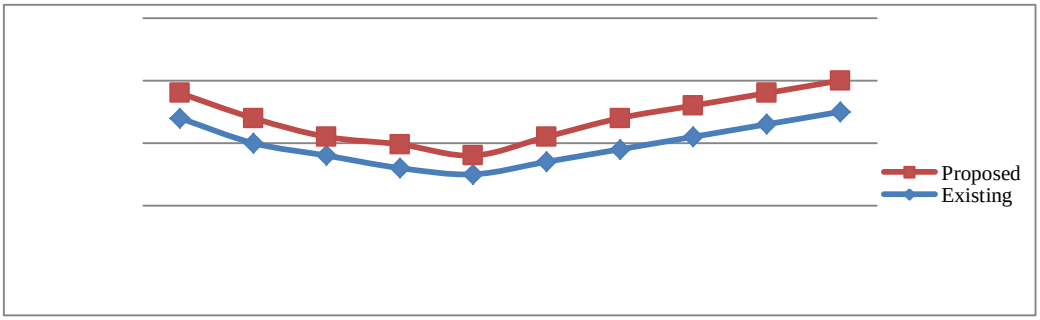

Fig.2 Performance analysis for CPU utilization

The scheduler must be able to allocate resource accurately and efficiently so that the QoS and performance can be maintained. If the resource does not provision resource in time, the performance gets degraded. As shown in figure 2 and 3, the Q-learning can deal with ever changing workload increases.

In figure 3, comparison between the existing and proposed work in memory utilizations. The result shows that, the proposed algorithm increases the memory performance requirements and ensure that high QoS.

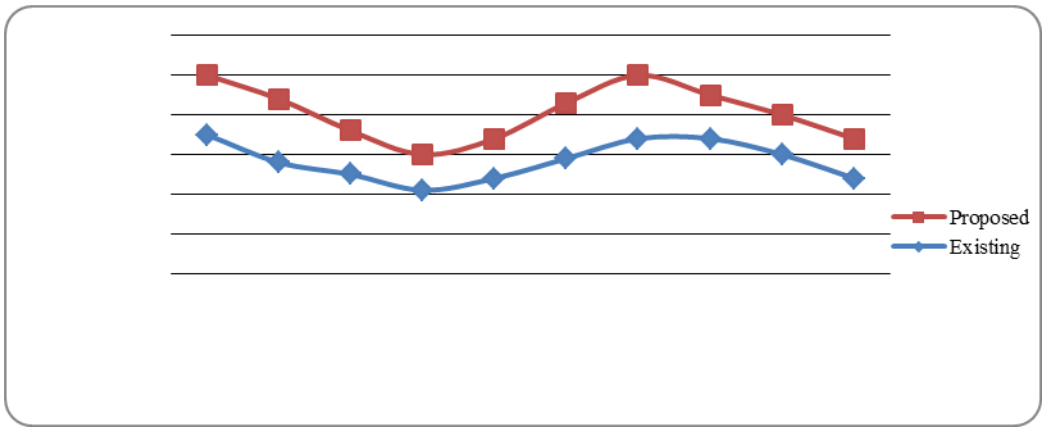

Fig.3 Performance analysis for memory utilization.

The above graph here inferred to provisioning the resource (CPU and memory) based upon the workload to increase the performance. From the figure 2 and 3 shows that Q-learning algorithm can provision resource more quickly than other approaches. Therefore this approach shows that better performance when compared with existing approaches.

V. CONCLUSIONS

The resource provisioning in cloud to support the elasticity of providing the resource to the user have proven very effective. In a cloud system with multiple pools of resource (e.g. CPU and memory) where the resource is provisioned to user request in the form of VM. In this paper, the Q-learning aggressively provisioning the resource to the users to overcome the rapidly increasing workloads. The results show that the proposed scheme can significantly accelerate the high performance and minimize the time, especially for overloaded cloud system.

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz,A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Communication. ACM, Vol. 53, No. 4, pp. 50–58, 2010.

[2] B.H Bhavani and H S Guruprasad "Resource Provisioning Techniques in Cloud Computing Environment: A Survey",in International Journal of Research in Computer and Communication Technology, Vol 3, Issue 3, pp 111-124, March- 2014.

[3] Gopal Krishna Shyam, sunilkumar S.Manvi,"Resource Allocation in Cloud Computing Using Agents,"in Proceeding. IEEE International Advance Computing Conference (IACC),pp. 458-463,2015.

[4] Jinzhao Liu, Yaoxue Zhang, Yuezhi Zhou, Member,Di Zhang, and Hao Liu, A. Rabkin, I. Stoica, and M. Zaharia, "Aggressive Resource Provisioning for Ensuring QoS in Virtualized Environments," Proceeding. IEEE, Vol.3, No. 2,pp. 119–131, June. 2015.

[5] B.H Bhavani and H S Guruprasad, "Resource Provisioning Techniques in Cloud Computing Environment: A Survey", in International Journal of Research in Computer and Communication Technology, Vol 3, Issue 3, pp 111-124, March- 2014.

[6] N. Huber, F. Brosig, and S. Kounev, "Model-based self-adaptive resource allocation in virtualized environments,"inProceeding.6thInternational.Symposym Software. Engineering. Adaptive Self-Managing System., pp. 90–99, 2011.

[7] T. C. Chieu, A. Mohindra, A. A. Karve, and A. Segal, "Dynamic scaling of web applications in a virtualized cloud computing environment," in Proceeding. IEEE International. Conference. E-Business Engineering, pp. 281–286, 2009.

[8] Sukhpal Singh and Inderveer Chana," A Metrics based workload analysis technique for IaaS is developed for cloud computing, "in proceeding. International Conference on Next Generation Computing and Communication Technologies (ICNGCCT 2014), DUBAI, UAE, pp.457-464, 2014.

[9] N. R. Herbst , S. Kounev, and R. Reussner, "Elasticity in cloud computing: What it is, and what it is not," presented at the Proceeding.10th International. Conference. Autonomous. Computer. San Jose, CA, USA, 2013.

[10] J. Rao, X. Bu, C.-Z. Xu, L. Wang, and G. Yin, "VCONF: A reinforcement learning approach to virtual machines auto-configuration, "in Proceeding. 2nd International. Conference. Autonomous. Computer. pp. 137–146, 2009.

[11] F. Xu, F. Liu, H. Jin, and A. V. Vasilakos, "Managing performance overhead of virtual machines in cloud computing: A survey, state of the art, and future directions," Proceeding. IEEE, Vol. 102, No. 1, pp. 11–31, Jan. 2014.

[12] R. Thonangi, V. Thummala, and S. Babu, "Finding good configurations in high-dimensional spaces: Doing more with less," in Proceeding. IEEE International. Symposium. Model. Anal., Simulation. Computer. Telecommunication. System. , pp. 51–60, 2008.

[13] R. N. Calheiros, R. Ranjan, and R. Buyya,"Virtual machine provisioning based on analytical performance and QoS in cloud computing environments," in Proceeding. International. Conference. Parallel Process., pp. 295–304, 2011