IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

CLOUD COMPUTING RESOURCES PROVISIONING TO MAXIMIZE QUALITY OF SERVICE

[1]Author: Imtiyaz Khan, research scholar department of Computer science & Engineering at Sri Satya Sai University of Technology & Medical Sciences, Sehore-MP.

[2]Author: Dr. Pankaj Kawadkar, Professor at department of Computer science & Engineering at Sri Satya Sai University of Technology & Medical Sciences, Sehore-MP.

Abstract: Enterprise apps have migrated to the cloud in large numbers in recent years. One of the challenges of cloud applications is the administration of Quality-of-Service (QoS), which is the issue of allocating resources to an application in order to deliver a service level in terms of performance, availability, and reliability. It is the goal of this paper to give scholars with an overview of current state-of-the-art QoS modelling techniques. Cloud systems are suited. We also examine and characterise their early use to a variety of decision-making issues. emerging in the administration of cloud quality of service. "Resource provisioning" is the term used to describe the selection, deployment, and management of software and hardware resources (e.g., CPUs, storage and networks) to provide guaranteed application performance. In large-scale distributed systems, such as Cloud computing, resource supply is a major and complex problem. Static and dynamic resource provisioning systems both have their own advantages and disadvantages. Quality of Service (QoS) parameters including availability, throughput, response time, security and dependability must be satisfied by these resource provisioning systems in order to prevent SLA breaches.

Keywords: Cloud Computing, Resource Provisioning, Static, Dynamic, Quality of service management, Modelling.

1 Introduction

Information services in the Internet era are best represented by cloud computing, but achieving this goal also raises new challenges for information technology. Cloud computing's on-demand service, scalability, and massive data storage are all being implemented in the data centre as an important application research project. In data centres, virtualization technology is utilised to provide a decoupled paradigm between physical resources and applications.. An application may share physical resources with other programmes using a virtual machine (VM). As a result, fine-grain virtual machines (VMs) are used to represent resource scheduling entities rather than coarse-grain service machines. The data centre benefits from virtualization technology, but there are certain drawbacks.

Asset provisioning is a key and testing issue in enormous scope circulated frameworks, for example, Cloud figuring conditions. As a result, by logically adopting resource management methods and procedures, it may ensure efficient resource provision and utilisation. As a result, achieving successful cloud computing resource management faces a variety of new obstacles, which are mostly shown in three forms of imbalance.

By using virtualization technology to aggregate widely dispersed resources into one massive pool, resource management, one of the most critical parts of cloud computing, attempts to safeguard the variety and complexity of the underlying resources. As a result, by logically adopting resource management methods and procedures, it may ensure efficient resource provision and utilisation. As a result, achieving successful cloud computing resource management faces a variety of new obstacles, which are mostly shown in three forms of imbalance.

Control-concentrated programs, for instance, require more prominent CPU power for branch forecast, while information serious projects require more memory to diminish the quantity of perused and compose tasks. Because of cloud computing's multitenant architecture that enables diverse applications to share a pool of data centre resources, server-loading efficiency may be difficult to quantify. An absence of equilibrium in the utilization of assets on a similar server could prompt a lessening in the general productivity of the framework. While basic interest guaging and sane buying can't meet this trouble in the distributed computing situation, a reasonable asset provisioning method should be proposed to additionally conquer the new struggle between differed applications and the brought together assets sharing pool.

Second, there is a divergence in how much time it takes to present an application. Server use in server farms goes from 5% to 20%, yet top jobs for some, administrations are 2-10 times bigger than the normal. Despite the fact that assistance loads vary the entire day, they may likewise be impacted via occasional or other intermittent varieties (e.g., an occasional high in December before Christmas shopping, and a post-occasion spike in picture handling locales' heap needs), as well as unforeseen occasions (like letting the cat out of the bag). Non-peak hours squander resources since few users deploy resources in excess of their peak demands. Consequently, the more volatile the load, the more resources are wasted by consumers. Static configuration mode cannot be used to fix this issue in a cloud computing environment. However, even if the cloud VM excels at isolation, the real-world interference caused by VM resource competition can't be avoided, lowering the overall performance of the cloud computing system.

Finally, the distribution of applications is skewed. This mandates that the load balancer be able to dynamically modify the server cluster to the current user access in order to minimise resource waste and the circumstance where present resources cannot fulfil the user's demands [1]. Some limitations remain in spite of the many scholarly investigations on load trends and elastic resource assignment that have been conducted. We discovered the following issues in this field's research after doing a detailed investigation of the present state of academia and industry: It is, first and foremost, inflexible. Additionally, other characteristics of cloud computing, such as its flexibility and the capacity to adapt resources to the needs of the user, aren't fully depicted in this paper. This goes against the pattern expectation and asset portions appear to be behind, influencing the client's insight and on occasion making it challenging to address sure of the client's issues.

Server farms should likewise manage issues including asset multiplexing, relationship and dynamic administration. Virtualized resources must be managed in a dynamic, high-efficiency manner by the optimal resource scheduler if different resource service systems are to provide consumers with useful and satisfying resources.

SLA and UUTC are introduced in this paper, alongside a streamlining asset provisioning plan in light of reinforcement learning (RL) and queuing theory (QT), which is aimed at accurately scaling cloud computing environments and efficiently allocating resources under service level agreement (SLA) and user cost constraints in the context of cloud computing.

(a) Based on QT and RL, a unique dynamic resource provisioning strategy is suggested.

(b) To evaluate the performance of various resource providing schemes, two ideas are introduced.

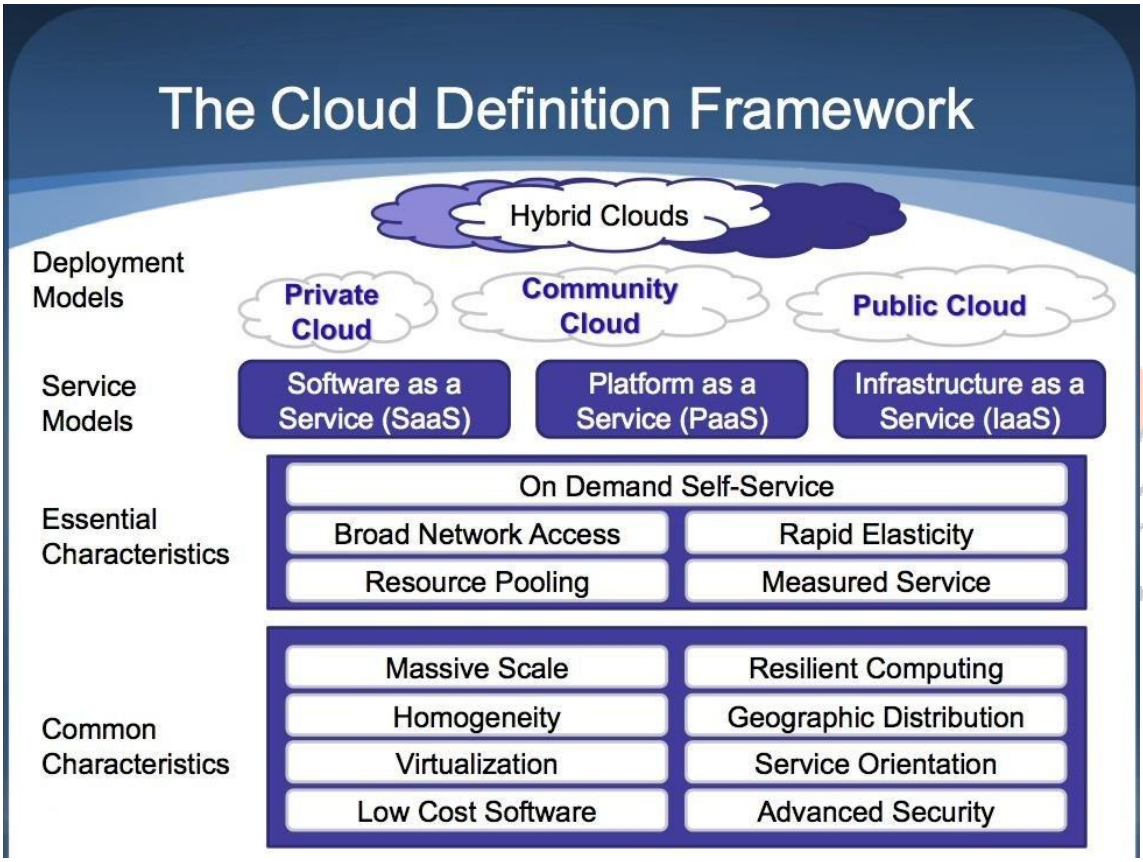
(c) A cloud computing platform is an excellent example of how we employ our method.

(d) The results of the experiment show that our created technique can make correct provisions at a variety of work arrival rates while avoiding SLA conflicts.

2 System Model

2.1 Cloud Computing Platform Framework

The distributed computing stage utilized in this examination will be presented in this part, as found in Figure 1. Framework construction and capacities are illustrated underneath, remembering subtleties for each part's job.



Job Queue for Users (UJQ). Two user-job queues are included in the cloud computing platform. The jobs enter the cloud computing platform after being submitted via the user interface.In order, queue 1 at a time and wait to be scheduled. When they're readyAfter they've been executed, they'll go into user queue 2 and wait to be called.The transmitter sends a message back to the end user.

Job Scheduling for Users (UJS). The task scheduling strategy assigns the requests in queue 1 to the appropriate VM clusters.

Cluster of Virtual Machines (VMC). A VMC is made up of many VMs that all conduct the same sort of task. Each VMC VM is purpose-built for a certain sort of work in order to maximise operational efficiency. Create, administer, and terminate virtual machine instances are all handled by a VMCA (Virtual Machine Cluster Agent). Due to the increasing amount of user workloads coming in, VMCA must add more VMs to the VMC so that throughput may be increased; otherwise, some of the VMC's virtual machines (VMs) might fail.

VM is in charge of the execution of each particular task. Execution results are sent to a platform interface and results are extracted from the queue. request from the next user A performance monitor is installed on each virtual machine. A Resource Management Agent (RMA) and a Monitor Agent (PMA) are two distinct classes of agent in the game. (RMA). The PMA is in charge of the performance indicators. Keep an eye on the VM's response time during the process. and the allocation of resources The virtual machine (VM) resource is under the control of RMA. Dynamic scheduling of CPU, memory, bandwidth, and data centre capacity are all critical components of management.

User-Sent Job Submissions (UJT). Queue 2's execution results are forwarded to the linked user in line with the transmission strategy.

Response Time to a Job (JRT). A cloud computing environment's Job Response Time (JRT) is based on the job queueing time (JQT), job execution time (JET), and job transfer time (JTT), as shown in Figure 1. (JTT). In other words, JRT is made up of JQT, JET, and JTT.

Standard queueing theory says that the jth VM in the cluster i's arrival rate is ij and its service rate is ij, given the platform's task arrival rate. This results in a [13] average queueing time for JQT.

$$\mathrm{JQT} = \frac{\rho_{ij}}{\mu_{ij}\left(1-\rho_{ij}\right)}, \qquad (1)$$

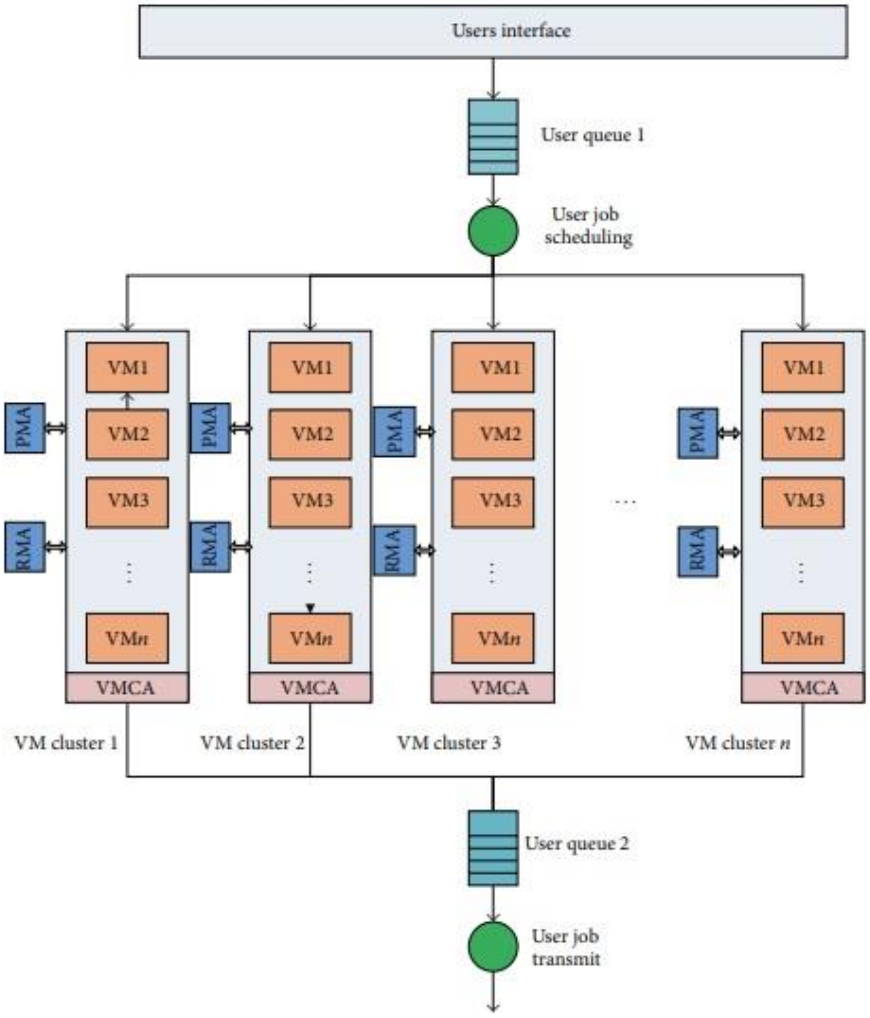


Fig 1: Architecture of Cloud Computing Frame work

where $\rho_{ij} = \lambda_{ij}/\mu_{ij}$ and the Probability Density Function (PDF) of JQT is

$$f_{\mathrm{JQT}}(t) = (1-\rho_{ij})\,\delta(t) + \mu\rho_{ij}(1-\rho_{ij})\,e^{-\mu_{ij}(1-\rho_{ij})t} \quad (t \geq 0). \tag{2}$$

Similarly, the response time of JET and JTT [14, 15] is

$$\mathrm{JET} = \frac{1}{\mu_{ij}}, \qquad \mathrm{JTT} = \frac{D_{ij}/B_{ij}}{1-\lambda_{ij}D_{ij}/B_{ij}}, \tag{3}$$

whereDij is the user job's result size and Bij is the user job's provided bandwidth resources. JET and JTT PDFs are also available.

$$f_{\mathrm{JET}}(t) = \mu_{ij}e^{-\mu_{ij}t} \quad (t \geq 0), \tag{4}$$

$$f_{\mathrm{JTT}}(t) = \frac{B_{ij}}{D_{ij}}e^{-(B_{ij}/D_{ij})t} \quad (t \geq 0), \tag{5}$$

respectively.

Thus the total response time in the VMC can be given as

$$T_{\mathrm{tot}} = \mathrm{JQT} + \mathrm{JET} + \mathrm{JTT} = \frac{\rho_{ij}}{\mu_{ij}(1-\rho_{ij})} + \frac{1}{\mu_{ij}} + \frac{D_{ij}/B_{ij}}{1-\lambda_{ij}D_{ij}/B_{ij}}. \tag{6}$$

SLA segmentation . The cloud computing platform's response time, which is a performance parameter, is constrained by the QoS or SLA that it provides. This study's SLA is broken down into the many stages at which users are expected to accomplish their task in order to properly offer cloud computing resources. environment. The resources might be used when the project is completed.resource provided according to the various phasesprovisioning method, which allows each step to be restricted by SLA, as shown in the diagram below:

$$\mathrm{JQT} \leq \mathrm{SLA}_{\mathrm{JQT}}, \quad \mathrm{JET} \leq \mathrm{SLA}_{\mathrm{JET}}, \quad \mathrm{JTT} \leq \mathrm{SLA}_{\mathrm{JTT}}. \tag{7}$$

As long as the task fulfils SLA restrictions at each stage of execution, the overall response time can meet the global SLA constraints. Furthermore, introducing segment SLAs can significantly enhance the cloud computing platform's QoS. A higher need is given to the gig execution in ensuing stages when a JQT

neglects to meet its SLA in light of a lack of assets or an I/O stop or clashes, for instance. This guarantees that work assets are accessible and diminishes time for JET and JTT, separately. to guarantee the general SLA of the client activity and meet QoS limitations.

As a result, commercial cloud computing platforms often charge hourly fees for their services. Table 1 displays the pricing of a standard on-demand instance on the Amazon EC2 cloud computing platform. The Utility Unit Time Cost (UUTC) may be defined as follows for each user's job:

$$\text{UUTC} = \frac{\text{Total cost}}{T_{\text{tot}}}. \tag{8}$$

$$\begin{aligned} &\underset{\{\forall \text{user job}\}}{\text{Maximum}} && \text{UUTC} \\ &\text{subject to} && \text{JQT} \le \text{SLA}_{\text{JQT}} \\ & && \text{JET} \le \text{SLA}_{\text{JET}} \\ & && \text{JTT} \le \text{SLA}_{\text{JTT}}. \end{aligned} \tag{9}$$

For now, we'll stick to the following definitions of key terminology in the different RL-based cloud computing resource provisioning schemes: state space, action set, and reward function.

one physical machine.

Within the same physical computer, the virtual ones are conceptually separate, but they compete for resource provisioning. The state space for each VM is stated as a vector consisting of VCPU, RAM, and bandwidth since these resources are regarded in this research to be part of the state space. The machine's upper limit is not exceeded by any element's value. An example of state space in a virtual machine is (1), (2), which implies that the virtual machine has a single virtual CPU, 2 M bandwidth, and 2 G of memory.

Actionable Space. The ith VM resource's action space includes increasing, maintaining, or decreasing resources, which may be indicated by 1, 0, and 1. Specify the appropriate action. In the meanwhile, the increasingor a reduction in VCPU, memory, and bandwidth resource each with a VCPU, 512 M RAM, and 0.5 M bandwidthIt's time to make a decision. After that, for the ith VM, assumethat its state space is (1, 2, 2), and that its decision action isThe instant can be written as (0, 1, 1), indicating that theThe number of VCPUs stays the same, however memory has increased toThe bandwidth is reduced to 0.5 M at 512 M. Following the action,The state space of a virtual machine is represented as a vector when it is implemented.

Immediate Reward: is a term that is used to describe a reward that is The instant reward is utilised to represent the proper running condition and task scheduling efficiency. The three scenarios are taken into account while creating a reward function. For example, if the current user's task has a higher than average UUTC that meets SAL or QoS requirements, the reward is 1. If it does not meet SLA or QoS, it receives a 0 reward.

Reinforcement Learning Algorithms: A Basic Allocation Scheme. Q learning, a well known support learning calculation, is utilized to resolve the successive decision issue in the MDP (9). Calculation 1 shows the pseudocode for the major Q esteem learning calculation. We compare the suggested resource provisioning strategy's performance to that of the Amazon cloud computing platform's utilisation ratio

provisioning technique [16]. Figure 3 illustrates the limited number of results that are relevant to the allocation of VCPU resources because of the huge number of experiments. To give our simulations greater credibility, we base them on real-world data, such as the features and pricing of Windows Azure.

(1) **Initialize** Q value table
(2) **Initialize** state s_t
(3) error = 0
(4) **repeat**
(5) **for** each state s **do**
(6) a_t = get_action(s_t) using ε-greedy policy
(7) **for** (step = 1; step < **LIMIT**; step++) **do**
(8) **take action** a_t observe **r** and S_{t+1}
(9) $Q_t = Q_t + \alpha * (r + \gamma * Q_{t+1} - Q_t)$
(10) error = MAX(error | $Q_t - Q_{\text{previous-}t}$)
(11) $s_t = s_{t+1}$, a_{t+1} = get_action(s_t), $a_t = a_{t+1}$
(12) **end for**
(13) **end for**
(14) **Until** error < θ

Q value learning algorithm.

It is possible to do real-time provisioning utilising virtual machine resources in virtual machines effectively while avoiding SLA conflicts, as shown in Figure 3 as the arrival rate of work rises. (b). Essential Q advancing regularly changes VCPU asset provisioning because of horrible showing (e.g., when the time has come to build the VCPU assets, they are not expanded yet decreased) because of the investigation double-dealing instrument, which in the end prompts continuous asset provisioning and SLA clashes because of successive asset provisioning.

The similar discoveries in Figure 3 permit us to reach a few inferences concerning the utilization of essential support learning in distributed computing asset provisioning.

(i) Convergence is slow.

(ii) Ineffective adaptation to changing arrival rates; as a result, the policy must be updated, and the convergent solution is not always achieved.

(iii) Even with a fixed task arrival rate, the poor solution is frequently obtained instead of the ideal option.

The aforesaid drawbacks of reinforcement learning, particularly its sluggish convergence rate and inadequate adaptive ability, significantly restrict its use in cloud resource provisioning.

Allocation of Reinforcement Learning Resources has been improved. We devise a better scheme to address each of the flaws in the basic reinforcement learning scheme, with an emphasis on the following features.

.

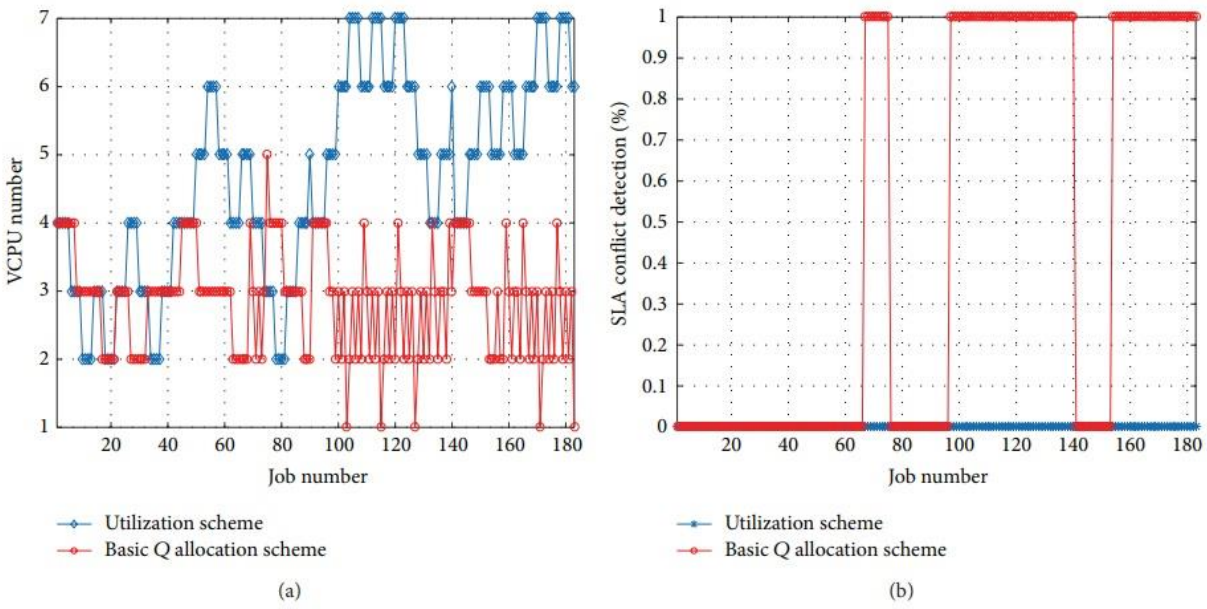


Figure 2: Comparison results of VCPU resource provisioning and SLA

Acquiring knowledge outside of a formal educational setting. It is possible to get the approximate function relationship between resource provisioning and the varied rates of job arrival using simulations based on real data sets and the Q value table, an offline training based on the core Q learning approach. Partitioning the state space during the offline learning process allows many instances to operate in parallel to learn: Figure 2 shows how this works: The basic Q learning system and the utilisation scheme were compared in terms of VCPU resource supply and the identification of SLA conflict. the fundamental Q learning system and the utilisation strategy for varied work numbers (VCPU) comparison. Identification of SLA conflicts between the basic Q learning scheme and other task number utilisation schemes

Conclusions

Using reinforcement learning and queueing theory, we provide a new approach to cloud computing platform resource provisioning optimization in this study. Utilizing the ideas of SSLA and UUTC, we transform the asset provisioning issue in distributed computing into a consecutive decision issue that can be settled by making a custom enhancement object capacity and utilizing support learning. SLA impact aversion and client costs are both improved by utilizing the proposed method, as indicated by the consequences of a progression of examinations. In addition, using the Q learning method, certain conclusions may be derived as the Q learning algorithm outperforms the competition.

References

[1] H.-S. Wu, C.-J.Wang, and J.-Y. Xie, “TeraScaler ELB-an algorithm of prediction-based elastic load balancing resource management in cloud computing,” in Proceedings of the 27th International Conference on Advanced Information Networking and Applications Workshops (WAINA ’13), pp. 649–654, IEEE, Barcelona, Spain, March 2013.

[2] Y. Gao, H. Guan, Z. W. Qi, T. Song, F. Huan, and L. Liu, “Service level agreement based energy-efficient resource management in cloud data centers,” Computers and Electrical Engineering, vol. 40, pp. 1621–1633, 2013.

[3] V. Suresh, P. Ezhilchelvan, and P. Watson, “Scalable and responsive event processing in the cloud,” Philosophical Transactions of the Royal Society A, vol. 371, no. 1983, Article ID 20120095, 2013.

[4] X. Nan, Y. He, and L. Guan, "Queueing model based resource optimization for multimedia cloud," Journal of Visual Communication and Image Representation, vol. 25, no. 5, pp. 928–942, 2014.

[5] H. Nguyen, T. N. Minh, and N. Thoai, "Tool-driven strategies for resource provisioning of single-tier web applications in clouds," in Proceedings of the 5th International Conference on Ubiquitous and Future Networks (ICUFN '13), pp. 795–799, July 2013.

[6] S. Yu, Y. Tian, S. Guo, and D. O. Wu, "Can we beat DDoS attacks in clouds?" IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 9, pp. 2245–2254, 2014.

[7] K. Salah and R. Boutaba, "Estimating service response time for elastic cloud applications," in Proceedings of the 1st IEEE International Conference on Cloud Networking (Cloud Net '12), pp. 12–16, Jussieu, Paris, November 2012.

[8] H. Khazaei, J. Misic, and V. B. Misic, "Performance analysis of cloud computing centers using M/G/m/m+r queuing systems," IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 5, pp. 936–943, 2012. 12 Mathematical Problems in Engineering

[9] H. Wada, J. Suzuki, Y. Yamano, and K. Oba, "Evolutionary deployment optimization for service-oriented clouds," Software: Practice and Experience, vol. 41, no. 5, pp. 469–493, 2011.

[10] M. Bourguiba, K. Haddadou, I. E. Korbi, and G. Pujolle, "Improving network I/O virtualization for cloud computing," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 3, pp. 673–681, 2014.

[11] Z. Luo and Z. Qian, "Burstiness-aware server consolidation via queuing theory approach in a computing cloud," in Proceedings of the 27th IEEE International Symposium on Parallel & Distributed Processing (IPDPS '13), pp. 332–341, IEEE, Cambridge, Mass, USA, May 2013.

[12] F.-C. Jiang, C.-T.Yang, C.-H.Hsu, and Y.-J. Chiang, "Optimization technique on logistic economy for cloud computing using finite-source queuing systems," in Proceedings of the 4th IEEE International Conference on Cloud Computing Technology and Science (CloudCom '12), pp. 827–832, December 2012.

[13] G. Grimmett and D. Stirzaker, Probability and Random Processes, Oxford University Press, 3rd edition, 2010.

[14] Y. Wu, C. Wu, B. Li, X. Qiu, and F. C. M. Lau, "CloudMedia: when cloud on demand meets video on demand," in Proceedings of the 31st International Conference on Distributed Computing Systems (ICDCS '11), pp. 268–277, IEEE, Minneapolis, Minn, USA, July 2011.

[15] J. Zheng and E. Regentova, "Qos-based dynamic channel allocation for GSM/GPRS networks," in Network and Parallel Computing, vol. 3779 of Lecture Notes in Computer Science, pp. 285–294, Springer, Berlin, Germany, 2005.

[16] X. Bu, J. Rao, and C.-Z. Xu, "A reinforcement learning approach to online web systems auto-configuration," in Proceedings of the 29th IEEE International Conference on Distributed Computing Systems Workshops (ICDCS '09), pp. 2–11, Montreal, Canada, June 2009.

[17] Zhiping Peng, Delong Cui,JinglongZuo, and Weiwei Lin "Research on Cloud Computing Resources Provisioning Based on Reinforcement Learning" Hindawi Publishing Corporation Mathematical Problems in Engineering Volume 2015, Article ID 916418, 12 pages http://dx.doi.org/10.1155/2015/916418