RESEARCHERID
THOMSON REUTERS

[Singh * *et al.*, 7(7): July, 2018]
IC™ Value: 3.00

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

OPTIMIZED ENVIRONMENT ALLOCATION FOR STATIC AND DYNAMIC TASKS BASED ON THROTTLE ALGORITHM IN CLOUD

Harjit Singh*[1] & Harpreet Kaur[2]
*[1]Student, Department of Computer Science & Engineering
[2]Assistant Professor, Lala Lajpat Rai Institute of Engineering & Technology, Moga

DOI: 10.5281/zenodo.1305803

ABSTRACT

Cloud computing is a developing computing technology that has tend to every entity in the digital organization, it can be personal or government sector. Taking into account the significance of cloud computing, finding new ideas in advancements of cloud computing services is an area of research field. With the initiation of the Cloud, deployment and hosting became easier and cheaper with the use of pay-per-use model offered by Cloud providers. Usually clouds have powerful data centers and data controller to handle large number of users. Cloud is a platform providing dynamic pool of resources and virtualization of services. To properly manage the resources of the service contributor, load balancing is required for the jobs that are submitted to the data center controller. Load balancing is a technique to cut up workload across many virtual processing unit in a server over the network to achieve least data processing time, optimal resource utilization and least average response time. There are various load balancing algorithms that are round robin, connection least, active monitoring, equally spread current execution and throttle.

In the existing work, the throttle load balancing approach distributes the incoming jobs uniformly among virtual machines based on its states whether it is busy or available. In this, all the virtual machines have same configuration in term of processing of task. There is no difference whether the task is static or dynamic. To overcome these, a new work is proposed which is based on appropriate allocation of virtual machines in terms of static and dynamic tasks and to find out the optimum cost for user as well as service provider. The proposed model is implemented and tested on simulation toolkit (CloudAnalyst). Results validate the correctness of the framework and show a significant improvement over existing work.

Keywords: Cloud computing, Load balancing algorithms, Cloud Analyst, Static and dynamic cloudlets, Virtual machine allocation, and optimized cost.

I. INTRODUCTION

Cloud computing is a developing computing technology that has tend to every entity in the digital organization, it can be personal or government sector. Cloud computing is a terminology used to elaborate a variety of computing concepts in which varied number of computers interconnected through a real time transmission medium like internet. Cloud computing is a model that used for on demand network access to a shared pool of configurable resources (e.g. servers storage, Networks, services and applications) that can be rapidly provisioned nominal management efforts or service provider interaction[1].

Figure 1: Architecture of Cloud Computing [1].

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

The popularity of term can be attributed to its use in marketing to sell hosted service in the sense of application service provisioning that run client server software on a remote location. As shown in Figure 1 Cloud computing relies on sharing of resources to achieve coherence and economies of scale. Cloud computing technique is fascinating to business holders that eliminate requirements for end users to plan beforehand and also allow industries to start from basic and increase resource only with the rise in service demand [1].

This prototype is generally referred as "pay per use model" that service provider provide virtual machines on rent as per requirement of the user. There are several challenges in Cloud Computing that need to be resolved before exploiting the features this technology. Some challenges include security issues, legal and compliant issues, load balancing, reliability, owner ship, performance and QOS, interoperability issues, data management issues, multi-platform support. Load balancing is an approach to divide work load among multiple computers and other resources over the network to achieve maximum output, optimal resource consumption, minimum response time, and bypass overload [2].

1.1Characteristics of Cloud Computing

Following are the main characteristics of the cloud computing:

- Reduced Cost
- Increased Storage
- Flexibility
- Reliability
- Location independence

1.2Cloud Computing Services

Cloud provides different types of services to number of users depending upon their requirements.

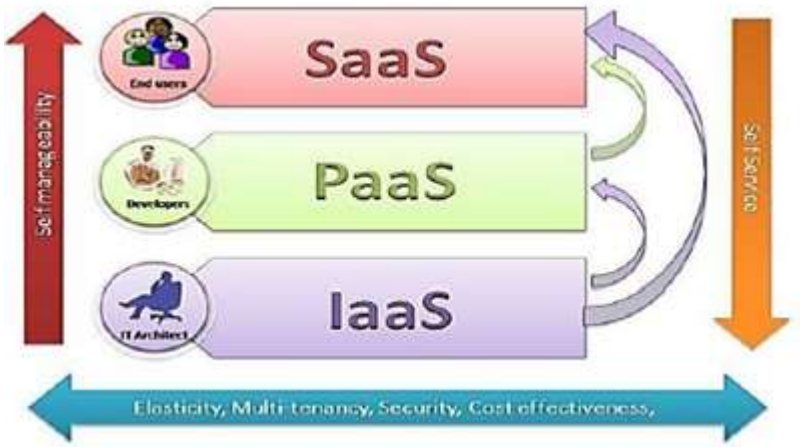

Figure 2: Service model of Cloud computing[2]

i. **Software as a Service (SaaS):** A complete application is offered to the customer, as a service on when he/she demanded[2].
ii. **Platform as a Service (PaaS):** PaaS is an application development and deployment platform delivered as a service to developers over the Web as they demanded[2].
iii. **Infrastructure as a service (IaaS):** Infrastructure as a Service is the delivery of hardware (server, storage and network), and associated for the original users[2].

1.3 Deployment Models of Cloud Computing

Various Deployment model types of cloud are available. Which are described below:

1.3.1 Public Cloud

Public Cloud is the most common deployment model where services are available to anyone on web in a pay-as-you-go manner[3].

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

1.3.2 Private Cloud

Private Clouds are deployed within the premise of an organization to provide IT or any services to its grouped users [3].

1.3.3 Community Cloud

The cloud infrastructure is shared among
several organizations that have common requirements or concerns and operated specifically for a targeted group. Their ultimate goal is to work together to achieve their business objectives [3].

1.3.4 Hybrid Cloud

Hybrid Clouds is the deployment which emerged due to diffusion of both public, private and community clouds [3].

This paper presents an approach with improved Throttled load balancing algorithm based on static and dynamic tasks and assigned all incoming jobs uniformly among the available Virtual Machines in an efficient way. The rest of this paper organized as follows:- Types of load balancing algorithms, VM selection of different load balancing algorithms, Limitations of existing methodologies. At last we proposed a model is to achieve better results for load balancing in cloud.

II. RELATED WORK

Nayak and Patel (2015) Investigated the various existing algorithms to resolve the point of load balancing in Cloud Computing. Discussed and compared those algorithms and presented a model based on improved throttled load balancing[2].

Kaur and Luthra (2014) Specified load balancing was a main challenge in cloud environment. Load balancing was approached to distribute the workload across multiple machines to ensure that no single machine was overloaded. It provides the proper consumption of resources. It also enhance the performance of the system. Many current algorithms equip better resource usability and load balancing. There were various types of loads were possible in cloud computing like memory, CPU and network load. Load balancing was the tactic of search out overloaded nodes and then move the extra load to other nodes[4].

Berwal and Kant (2015) Viewed the aim of cloud computing is to provide services to the users on demand. The point of revealing the availability of virtual machines to the client will enhance the performance level of the cloud. For allocation of efficient virtual machines on demand all have to decide efficient load balancing algorithm. Whenever the user came to know about any free available virtual machines they can choose whether they want to use service from that cloud or not. Thus while caring out these issue it can able to have a better service from the cloud computing[5].

Singh, Sharma and Kumar (2016) Discussed the Cloudanalyst tool. Cloudanalyst tool was used to determine the better load balancing algorithm from various scheduling and load balancing techniques e.g. round robin algorithm. This learning will help valued understanding to design infrastructure services of the Cloud. Interaction between one data center with the other, algorithms of load balancing and other value added services also present, like service broker policies, which synchronize efficiently among data centers and enhance cost and throughput which was given to the owners[6].

Wickremasinghe, Calheiros and Buyya (2009) Viewed that with advances in Cloud computing opens up many new possibilities for Internet applications developers. Lastly, a main cause of web application developers was deployment and hosting of applications, because it required adoption of a server with a fixed limit to handle the expected application high demand and the installation and support of the whole software platform supporting the application. Moreover, server was underutilized because peak traffic only at particular times. With the apearance of Cloud, deployment and hosting became easier and cheaper with help of pay per use model offered by Cloud providers. Several Cloud providers were available, everyone propose dissimilar pricing and located in different regions of the world, a new problem of engineers was to select providers and data center locations for applications. There was a shortage of tools that facilitate developers to assess requirements of large Cloud applications in terms of geographic distribution of servers and user's workloads. To solve this, modeling and evaluation of Cloud environments and applications, proposed CloudAnalyst. To simulate large Cloud applications with the intention of examining the behavior of such applications under various deployment structure. CloudAnalyst provide helps to developers, that how to divide apps between infrastructure of cloud and value added services like optimization of performance of application[7].

Patel and Jha (2016) Proposed new throttle load balancing algorithm. To properly manage the resources of the service contributor, load balancing was required for the jobs that were submitted to the service contributor. Load balancing is a technique to share load among virtual machines in a server computer over the network to achieve least data processing time, optimal resource utilization and least average response time. At present, a existing throttle load balancing approach was

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

presented for dividing of incoming jobs uniformly among the servers or virtual machines. Further, the performance was analyzed using CloudAnalyst simulator and compared with existing Throttled algorithms[8].

Mesbahi, Hashemi and Rahmani (2016) Presented an analytical comparison for the combinations of VM load balancing algorithms and different broker policies. Evaluate these approaches by simulating on CloudAnalyst simulator and the final results were presented based on different parameters. The results of this research specify the best possible combinations[9].

Ahmed and Singh (2012) Presented a comparison of various policies utilized for load balancing using a tool called cloud analyst. Cloud computing, as of today has boomed the use of internet services and has enhanced the use of network services where the capability of one node can be utilized by other nodes. Central to the architecture is SoA and IS which raises a big question on performance and complexity. To properly manage the resources of the service provider we require balancing the load of the jobs that are submitted to the service provider. Load balancing is required as we don't wish centralized server's performance to be debased. Various types of algorithms have been presented to do this type of task[10].

Buyya et. al.(2009) Presented some cloud platforms, that are developed in organizations, with the existing work towards realizing market oriented resource allocation of Clouds. They also describe the difference between High Performance Computing (HPC) workload and Internet-based services workload[11].

Chankar, Mishra and Sharma (2016) Proposed a VM scheduling technique that provide the priority for each VM based on their cost and then give the VM 1st that has the highest priority. To make the cloud system, the improved algorithm was using simulator and analyze with the current VM scheduling technique[12].

III. PROPOSED WORK

In this work, throttle algorithm has been implemented for providing optimized environment for static and dynamic tasks.

3.1 Experimental Tool

The Cloud Analyst is GUI based simulation tool built on top of CloudSim tool kit, by extending CloudSim functionality with the introduction of concepts that model Internet and Internet Application behaviors.

Figure 3: Shows the user interface of CloudAnalyst simulator.

In the proposed work, try to display the results by using graphical charts. After pressing the run simulation button, the cloudlets are provided to particular virtual machine

Development of large-scale applications in the cloud using the CloudAnalyst simulator is quite cost-effective and easy. CloudAnalyst is basically made in Java for a appraise performance and cost of geographically distributed cloud system that have vast user workload based on different parameters.

3.2 Existing Load Balancing Algorithms

A. Round Robin

It is the random selection of the Virtual Machine. The datacenter controller refer the inputs to a list of Virtual Machines on the basis of rotation. The first input is allocated to a Virtual Computer picked randomly from the list and then the Data Center controller refers the inputs in a circular order[2].

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

B. Equally Spread Current Execution Load

Load balancer maintains an index table of Virtual Machines as well as number of inputs currently given to the Virtual machine. If the request comes from the data centre to allocate the new Virtual resource, it scans the index table for least loaded[2].

C. Throttled Load Balancing Algorithm

Throttled load balancing algorithm is relied on status of Virtual Machine. Here status is referred as allocation of Virtual Machine, allocated or not. This information is stored in an index table at load balancer. This index table containing two parameters, one is to identify an individual Virtual Machine, so it's ID and status of that Virtual Machine in the form of (Available or Busy).

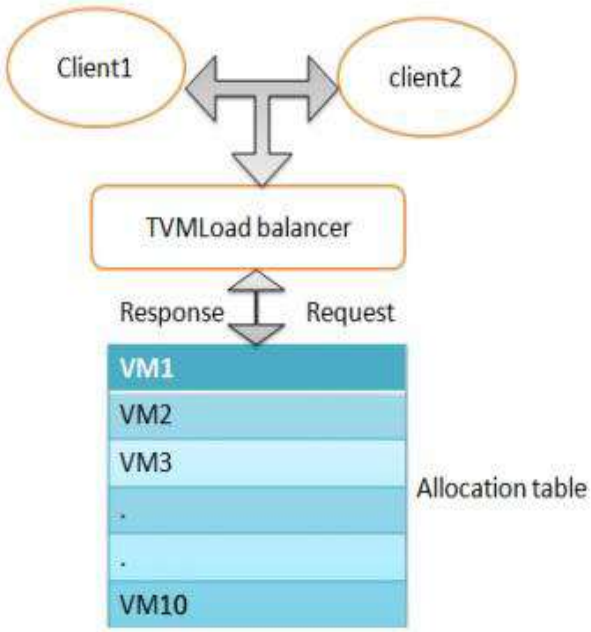

Figure 4: VM Placement in Throttle algorithm[2].

At start, all Virtual Machine set to available mode. First of all client request received by data centre, then it will forward them to load balancer to find an appropriate Virtual Machine for user request. Now which Virtual Machine can take this load of new job will be decided by load balancer. Load balancer will start scanning the index table From the top, and search for an available Virtual Machine. If any of the available Virtual Machine is found from the index table, than id of that Virtual Machine will be notified to the datacenter controller for request assignment. If datacenter successfully allocate that Virtual Machine id to specified request than it will notify the success of its operation to load balancer and accordingly, load balancer updates its index table. If datacenter controller found any trouble during the allocation it will give negative feedback notification within specific time duration, which leads to no updates in index table. If a situation occurs when all Virtual Machine having status busy than datacenter controller receives -1 from the load balancer[2].

3.3 Limitations Of Existing Methodology

Throttle load balancing algorithm maintains a hash table for current state of Virtual Machine whether it is Busy or Available, that assist much in placement of Virtual resources. Among all described algorithm above throttle load balancing algorithm have best approach for load balancing. But it works properly only when the hardware configuration of all Virtual Machines of data center had similar hardware Configuration. Throttle algorithm can be improved to take software environment also into consideration while selecting the Virtual Machine for the task. If the cost per request is kept according to the environment need, we can get much optimized figure which will be beneficial both for the cloud customer and the service provider.

3.4 Proposed Model

Proposed throttle algorithm is taking decision of VM selection with hash table with more parameters. The throttle algorithm will be improved by adding parameters for type of task and for the software environment needed if the task is a dynamic task when the VM is selected. It will help in better allocation of VM because not all the VMs need to have all the software environments needed depending upon the underlying host environment. It helps in much better provisioning of resources than traditional throttle algorithm. The cost will be based on the type of task and the software environment present at the VM to generate an optimized cost solution as per the task to be executed in that environment.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

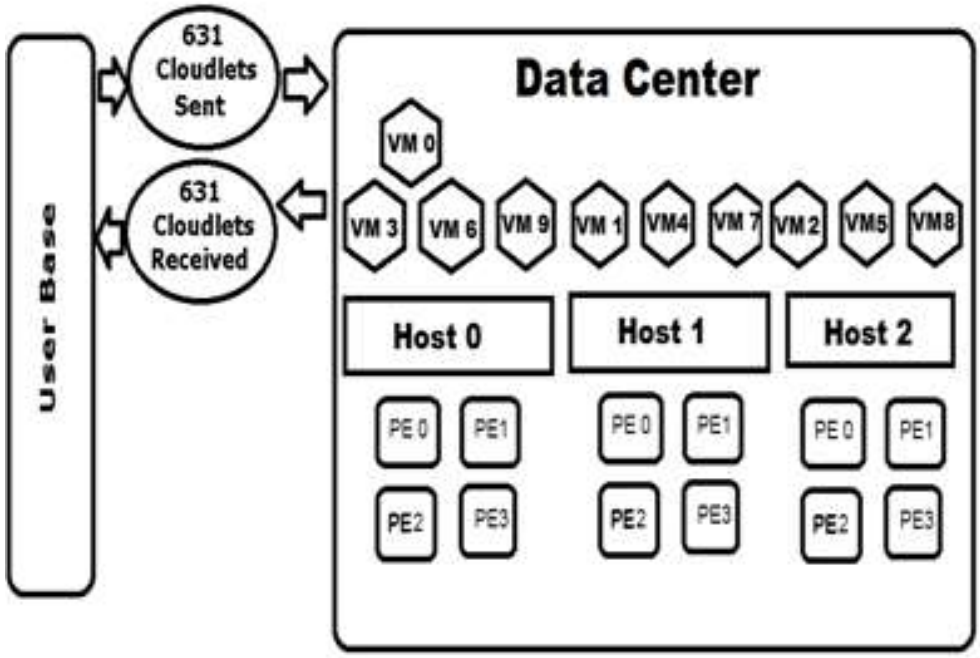

Figure 5: Proposed Model Architecture

In figure 5, shows the hardware configuration of the cloud, In which there are three hosts (machines) having 10 virtual machines and each host have four processing elements. Each virtual machine have its own functionality. Virtual Machine configuration are as follows:

VM 0 and VM 1 are only for the static tasks and rest are for dynamic tasks.
VM 3, VM 6 and VM 9 are used for PHP environment tasks.
VM 2, VM 5 and VM 8 are used for JAVA environment tasks.
VM 4 and VM 7 are used for DotNet environment tasks.

3.4.1 Metrics used for evaluation of proposed work
Following are the metrics used for evaluation of proposed work

- **Appropriate allocation of virtual machine**

Each virtual machine has its own functionality in terms of static and dynamic tasks. In the architecture, it is clearly shown that which virtual machine is used for a particular task type. Static tasks should go on VM with static configuration and dynamic tasks go on for processing on particular vm with special software configurations.

- **Optimized cost**

As described in the simulator, cost will be generated as according to the task type. Cost should be optimum, which clearly show that there is no loss for user as well as service provider.

3.4.2 Proposed Algorithm
Follow the flow chart in figure 6.

Input
Datacenter requests r1,r2, ….. rn
Available VMs vm0, vm1,….. vmn
Output
Datacenter requests r1,r2,….. rn are allocated available VMs vm0,vm1,……. Vmn

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

Steps:

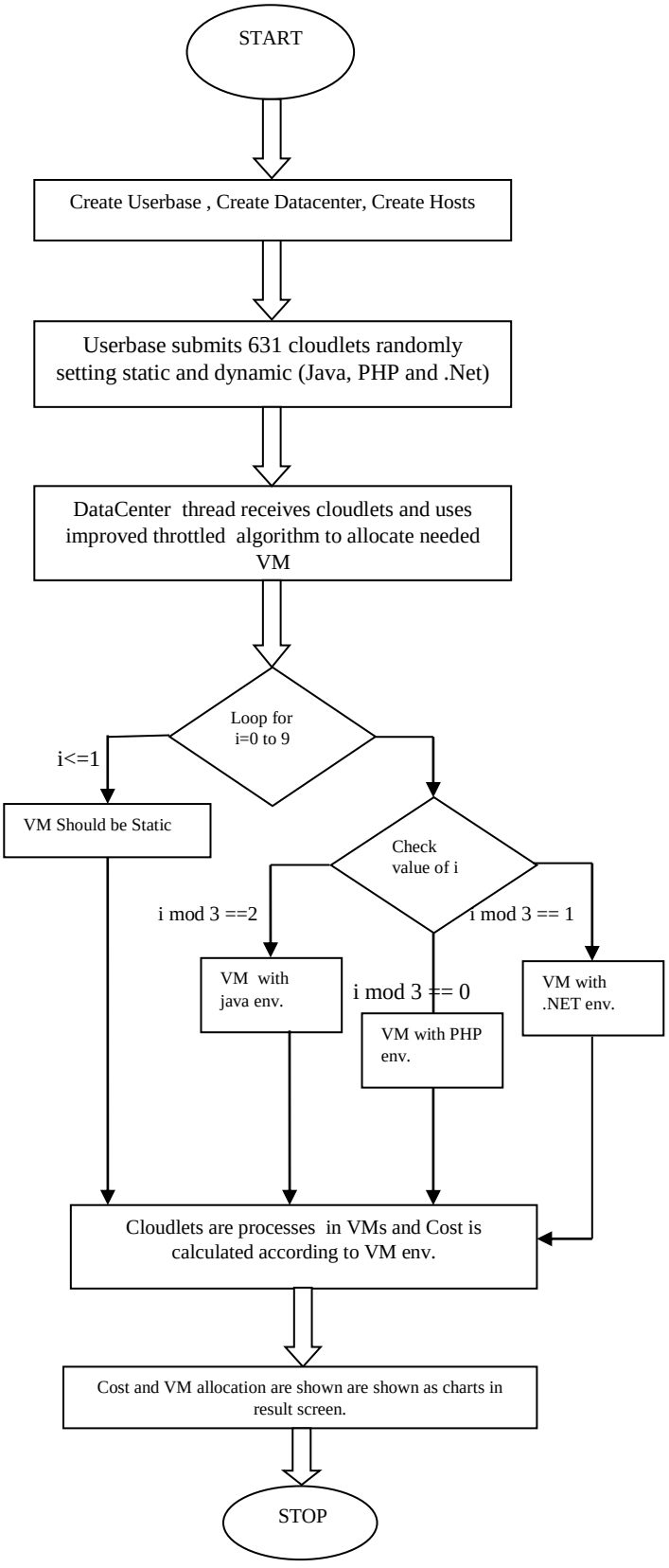

Figure 6: Work Flow of Proposed Work

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

IV. RESULTS

4.1 Simulation Setup

Eclipse is a software platform that has been made for building web and application development tool. The environment does not hand over a great deal of user purpose by itself. The value of the platform is what it boost: rapid development of integrated features based on a plug-in model. Eclipse gives a common user interface model for functioning with tools.

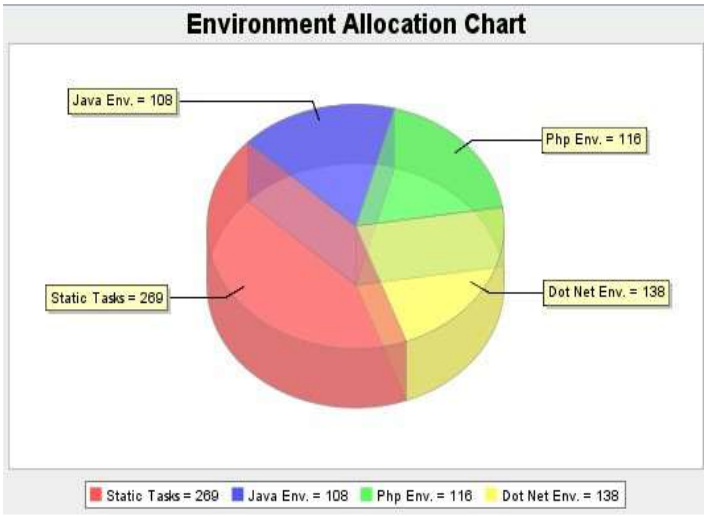

Figure 7: Environment Allocation Chart

From the total cloudlets, how many cloudlets are static and dynamic? And dynamic further described how many of them need java environment, php environment and dotnet environment.
In figure 7, Pie Chart displays the number of static cloudlets are 269 and Dynamic are 362 in which the software environment need Java for 108, Php for 116 and Dotnet for 138.

4.2 Appropriate VM Allocation

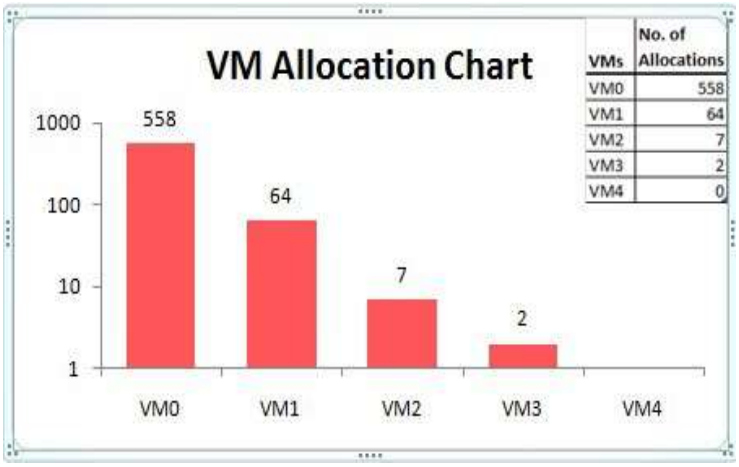

Figure 8: VM Allocation Chart by using Existing Throttle algorithm

Figure 8 shows that, there are 5 virtual machines in the existing throttle algorithm, in which 558 cloudlets sent to VM0, 64 to VM1, 7 to VM2, 2 to VM3 and zero to VM4.

The throttle algorithm works with the state like whether the virtual machine is busy or available. When VM0 is busy then the task will be send to next virtual machine that is VM1 and similarly for VM2, VM3 and VM4.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

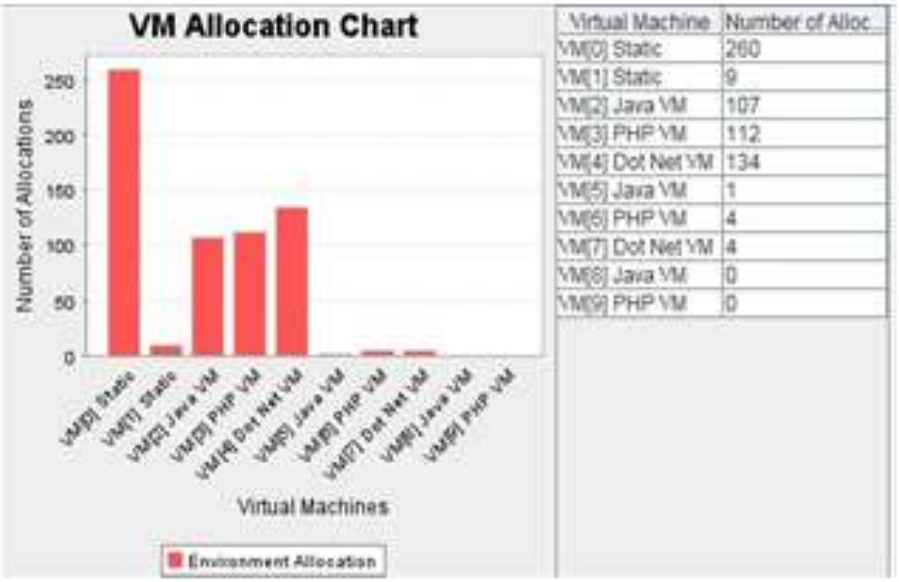

Figure 9: VM Allocation chart by using proposed throttle algorithm.

Figure 9 shows that, there are 10 virtual machines in the proposed model to deal with specific cloudlets, in which VM0 and VM1 for the static tasks and processes 260 and 9 tasks respectively, VM2, VM5 and VM8 for java tasksand processes 107, 1 and 0 tasks respectively, VM3, VM6 and VM9 for PHP tasks and processes 112, 4 and 0 tasks respectively and similarly VM4 and VM7 for .NET tasks and processes 134 and 4 tasks respectively. In proposed work there is clearly shown that static and dynamic cloudlets deal with specified virtual machines. The VM allocation policy of proposed work is better than existing work.

4.3 Optimized Cost Evaluation

The simulation experiments were performed to find optimized cost of the cloud. A particular cloudlet is allocated to an appropriate virtual machine because the incoming job could be either static or dynamic. As cloud computing is the pay-per-use model, So by doing this, user should pay for what they use. If the user needs to perform their static task, then user should need to pay (Rs. 30) for static virtual machine which is cheaper than dynamic virtual machine. Similarly if user want to perform their java task and user need a virtual machine of java platform then the service provider provide the java vm and charge accordingly Rs. 60 for Java, Rs. 55 for dotnet and Rs. 45 for PHP which is set in the proposed work.

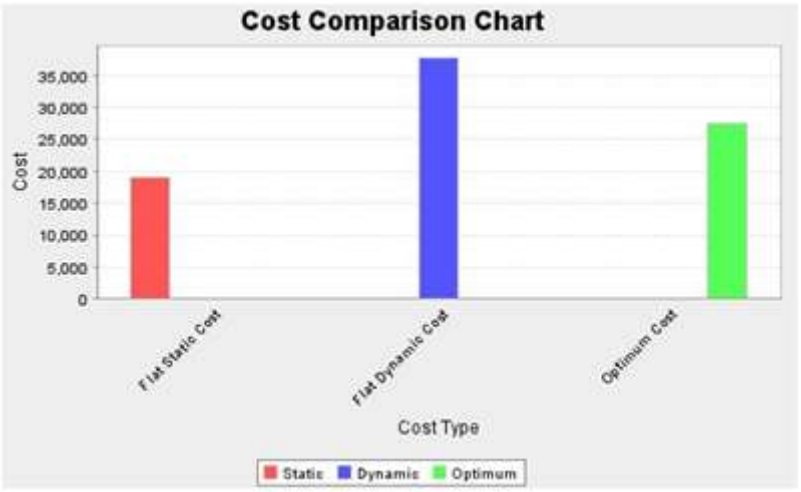

Figure 10: Cost comparison between existing and proposed throttle algorithm.

In figure 10, There is a comparison of costs and displays the optimized figure of cost of total of 631 cloudlets. If all the cloudlets sent from userbase to datacenter considers static then the cost is around 18930 and service provider could be in the loss, if they are considered dynamic then the cost is around 37860 and user could be in the loss and the optimized figure is 27360 and neither user nor service provider in the loss.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

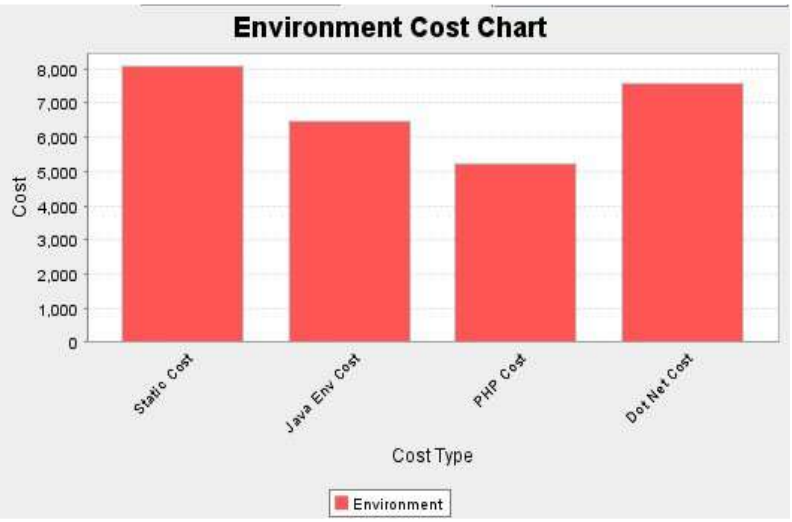

Figure 11: Environment Cost Chart

Figure 11, displays that, the cost of 631 cloudlets, in which 269 cloudlets were static having cost of 8070, 108 cloudlets of java environment having cost of 6480, 116 cloudlets of php environment having cost of 5220 and 138 cloudlets of dotnet env. having cost of 7590.

V. CONCLUSUIONS AND FUTURE SCOPE

5.1 Conclusions

As seen in the normal cloud architecture is studied with all the virtual machines having same configuration. 631 cloudlets were sent from the Userbase to the datacenter and cost and time graphs were plotted. In the present study the cloud architecture is modified so that we can create Virtual Machines with different environments. The cloudlet classes are also modified so that they are divided into different categories according to the environment they need to be executed. Throttle algorithm which was used to allocate Virtual Machines to cloudlets has also been modified so that the type of environment needed is also taken into account. Now when 631 cloudlets are submitted each one is allocated the Virtual Machine that has the appropriate execution environment.

We see that the cost can be highly optimized now because the software needed for a different environment may vary in price and the cloud customer can be charged according to the environment (software), the cloudlets of customer userbase need.

5.2 Future Work

Cloud Architecture can be modeled in a variety of ways. We have discussed one of the ways of improving optimizing the cost. The approach uses modified Throttle algorithm. The algorithm can be further improved or some other load balancing algorithms can be modified for this architecture scenario, so that cloudlets are allocated pertinent execution environments and the customer charged accordingly and the cloud provider can manage resources optimally based on the environment software cost incurred.

REFERENCES

[1] Kaur, J., & Kinger, S. (2012). A survey on load balancing techniques in cloud computing. International Journal of Science and Research (IJSR), 2662-2665.
[2] Nayak, S., & Patel, M. P. A Survey on Load Balancing Algorithms in Cloud Computing and Proposed a model with Improved Throttled Algorithm.
[3] Gulati, A., & Chopra, R. K. (2013). Dynamic round robin for load balancing in a cloud computing. IJCSMC, 2(6), 274-278.
[4] Kaur, R., & Luthra, P. (2012,December). Load balancing in cloud computing. In Proceedingsof International Conference on Recent Trends in Information, Telecommunication and Computing, ITC.
[5] Berwal, M., & Kant, C. (2015) Load Balancing in cloud computing. International Journal of Computer Science and communication vol, 6, 52-58.
[6] Singh, S. P., Sharma, A., & Kumar, R. (2016). Analysis of Load Balancing Algorithms using Cloud Analyst. International Journal of Grid and Distributed Computing, 9(9), 11-24.
[7] Wickremasinghe, B., Calheiros, R. N., & Buyya, R. (2010, April). Cloudanalyst: A cloudsim-based

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

visual modeller for analysing cloud computing environments and applications. In Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on (pp. 446-452). IEEE.

[8] Patel N.H. and Jha J.(2016) Improved Throttling Load Balancing Algorithm With Respect To Computing Cost and Throughput For Cloud Based Requests IJARIIE (VOl-2 Isseue 3)

[9] Mesbahi, M. R., Hashemi, M., & Rahmani, A. M. (2016, April). Performance evaluation and analysis of load balancing algorithms in cloud computing environments. In Web Research (ICWR), 2016 Second International Conference on (pp. 145-151). IEEE.

[10] Ahmed, T., & Singh, Y. (2012). Analytic study of load balancing techniques using tool cloud analyst. International Journal of Engineering Research and Applications, 2(2), 1027-1030.

[11] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation computer systems, 25(6), 599-616.

[12] Chakankar, P., Mishra, N., & Sharma, S. (2016). A Survey on Different VM Scheduling Approach in Cloud Computing Environment. International Journal of Computer Applications, 136(9), 1-3.

[13] Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A., & Buyya, R. (2011). CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Software: Practice and experience, 41(1), 23-50.

[14] Jadhav, S. P., & Deshpande, P. R.(2014).Load Balancing in Cloud Computing. International Journal of Science and Research (IJSR), 3(6), 2282-2285.

[15] Makkar, G., & Kaur, P. D. (2015). A Review of Load Balancing in Cloud Computing. International Journal of Advanced Research in Computer Science and Software Engineering, 5(4).

[16] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. National Institute of Standards and Technology Special Publication 800-145.

[17] Ray, S., & De Sarkar, A. (2012). Execution analysis of load balancing algorithms in cloud computing environment. International Journal on Cloud Computing: Services and Architecture (IJCCSA), 2(5), 1-13.

[18] Sonawane, S., Arnikar, P., Fale, A., Aghav, S., & Pachouly, S. (2014). Load balancing in cloud computing. International Journal, 4(2)..

CITE AN ARTICLE

Singh, H., & Kaur, H. (2018). OPTIMIZED ENVIRONMENT ALLOCATION FOR STATIC AND DYNAMIC TASKS BASED ON THROTTLE ALGORITHM IN CLOUD. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, 7*(7), 1-11.