

Effective Load Balancing Mechanism for Migration of Virtual Machine in Dynamic Cloud Environment

Dr. Girish Kumar D¹,
Associate Professor,
Department of AIML,
BITM, Ballari,
Affiliated to VTU, Belagavi,
Karnataka, India

Mr. Shiva Kumar K S²
Assistant Professor,
Department of ECE,
BITM, Ballari,
Affiliated to VTU, Belagavi,
Karnataka, India

Mr. Srisailanath³
Assistant Professor,
Department of CSE
BITM, Ballari
Affiliated to VTU, Belagavi,
Karnataka, India

Abstract

Cloud Computing is a technology which is widely been used by most of the organizations to carry out their business and day by day many users are using cloud to perform their day to day activities, this increases the load on the virtual machines in the data centers to effectively perform load balancing, such that resources are effectively utilized. In this paper we proposed an algorithm for dynamically handling the load balancing in the virtual machines migration at the data centers such that the processing time, response time, data transfer cost is being minimized. The results after simulation clearly shows that the proposed system has better load balancing of resources in dynamic cloud environment.

Keywords: Cloud Computing, Virtual Machines, Load Balancing, Data Center

1. Introduction

Cloud computing is a technique for providing dynamic on demand services to all the users of the cloud through internet and is been made available subscription basis. It provides various services such as platform as a Service(PaaS), Infrastructure as a Service(IaaS) and Software as a Service (SaaS). Cloud ensures the users with reliable services with less cost and this led to the deployment of cloud by many business enterprises, specially the startup business with less capital [1]. So the cloud users need not go for deployment of physical infrastructure for data storage, networking etc.

Through virtualization, cloud provides an abstract view of the physical resources where the users access it based upon the need and flexibility. The main aim of cloud computing is to provide maximum services with less cost that too without any delay in the services time to time.

In cloud simultaneously the request is being processed made by all the users without any delay. Here load balancing plays a very important role in real time applications. Virtual machines are being used in order to solve these kind of issues but still load balancing is one of the issues which has to be addressed in order to achieve scalability and along with that still there are security, availability, integrity [2] and many more. Load balancing is the process of distributing the load across various virtual machines and concurrent execution of all the task has to be carried out without any delay and should get accurate results.

Static allocation of resources in the cloud has led to many issues [3] for instance a web based platform used by static hosting leads to unreliable service, where a single outage may make the entire platform unusable and at the same time it is very expensive in terms of costs due to static allocation of resources by the cloud, that is why almost all business organization are shifting to dynamic cloud based services.

Dynamic allocation of the resources is the essential need of any business organization and cloud ensures limitless storage and computation to the users based upon the demand. An efficient resource allocation mechanism is needed in order to balance the load of the virtual machines by providing services to the users at different levels and a strong mechanism is required to utilize the resource effectively by ensuring proper load balancing.

In this paper, an efficient dynamic load balancing algorithm is being proposed which ensures effective utilization of the resources in dynamic cloud environment.

2. Related Study

Various techniques and mechanism are being implemented for effective utilization of the resources in cloud. [4] defined cloud as a fast evolving technology in the present internet world. The increasing growth of cloud users has led to the increase in computational costs, administrative costs and consumption of energy. An algorithm named krill herd is being implemented in order to optimize consumption of energy and to minimize the violations of SLA occurred in the data centers. [5] proposed an algorithm for effective utilization of energy in order to minimize the number of Physical Systems(PS) and by using the concept of trigger time the utilization of energy has been reduced by setting a double threshold value and the outcome of this technique, minimizes the consumption of energy by 7% and violations of SLA by 13%.

[6] introduced an active and Virtual Machine (VM) load balancing method, been utilized in order to achieve load balancing in VM by distributing the load equally among all the VM's, such that the computational time and response time of a task is been reduced. [7] described strategies for minimizing the consumption of energy along with utilization of resources. Errors occurred in scheduling techniques are due to peak loads, which in turn demises the energy efficiency. The task of scheduling will be more complex at peak time, due to non-availability of technique that can predict the utilization of resource in future. This algorithm aims to provide better quality of service, effective resource utilization along with minimal consumption of energy.

[8] came with a solution to the problem occurred due to migration of VMs, with an intend to minimize the computational costs occurred due to migration. There are two algorithms used to achieve this goal, a greedy approach is being used to select the specific VM. This can be achieved by mapping among the underutilized host and VMs, the another one is by applying double auction mechanism for migration of VMs.

In order to get better performance and achieve good QoS [9] used SLA based strategy for VM selection, which makes use of a method for detection of VM which are underload using single and double pass optimization function. [10] made use of directory agents, which minimizes migration time of VMs. In this approach a mobile agent will maintain the record of each and every occurrence of all VMs present in the cloud. Whenever a request is received from the mobile user, then the migration agent will initiate the process for migration and in order to reduce the consumption of energy [11] used a technique, in which the VMs are selected from the overloaded host and this in turn helps to calculate the time required for migration. Here the operations such as allocation, reallocation and deallocation are being carried out dynamically by the physical server. [12] came with an idea of obtaining optimal performance by adopting hybrid approach for migration, which includes the process such as push and pull. When the migration process is initiated, under loaded and overloaded servers helps in pull and move of the VM. [13] During migration, the overloaded server will move the VM to the servers which are less loaded in order to achieve better performance.

3. Proposed System

The proposed techniques is being adopted with an objective of minimizing the energy consumed in data centers during VM migration and efficiently achieve better load balancing in dynamic environment in the cloud.

3.1 Linear Regression(LR)

This depicts the relationship among variables

$$U_p = \acute{\alpha}_v + E_v$$

Where,

$\acute{\alpha}_v \rightarrow$ Vector capacity of PS(*Physical System*)

$E_v \rightarrow$ Expected vector

The coefficients of regression are,

$$E = \sum_{i=1}^n \frac{(xi - mean(x))(yi - mean(y))}{(xi - mean(x))^2}$$

$$\acute{\alpha}_v = mean(y) - (E * mean(x))$$

The prediction of effective utilization of PU is dependent upon the linear function that portrays the relationship among the Consumed Vector Capacity(UV_c) and predicted vector capacity(PV_c) for each VM is:

$$PV_c = \acute{\alpha}_v + (E_v * UV_c)$$

3.2 Calculating the load

One of the main task in cloud is computation of the load and the load will always depend upon the utilization of CPU, memory and bandwidth. In this work the load is been calculated based upon the effective utilization of CPU and is given by,

$$\text{Load(PS)} = \sum_{x \in \{1 \dots 1 \times 1\}} Ru$$

Here,

$$Ru = \frac{\text{Consumed Capacity}}{\text{Total Consumption}}$$

3.3 Calculating the Threshold

Threshold values are being utilized in order to compute the underload and overload situations of PS. Here if the load over the CPU of any machine exceeds high threshold value, then that machine is known as overloaded and if the threshold value of PS is less, then it is considered as under loaded.

The $Max_{threshold}$ value is the average load of all PS present in the data center and $Min_{threshold}$ value is kept constant (0.1) [P1-24].

$$PS_{max} = \frac{\sum_{j=1}^n PS_{load}}{S}$$

$$PS_{min} = 0.1$$

3.4 VM Migration Algorithm

1. Move all the VMs to Ready state
2. Accept user request for VM migration
3. For each node, obtain max(CPU)
 - Move VM between PS to PS
 - If PU of PS < $Min_{threshold}$
 - emigrate VM
 - Else if PU of PS > $Max_{threshold}$
 - emigrate VM
 - Else not within the range
4. If usage of Node(N) < $Min_{threshold}$
 - emigrate VM and let node be in IDLE state
- Else if Optimal Target
 - Send information to LR

Else repeat step 3

5. PU of each host is been obtained by LR (using equation 3.1)
6. Allocate the task to PU and then Stop
7. Repeat Step 2
8. Stop

4. Experiment Results

The experiment is carried out by using a simulator tool named Cloud-Sim, which is one of the tool used for cloud based experiments. The main objective of this tool is to provide a generalized framework for simulating, modelling and experimenting for cloud based applications related to IaaS and SaaS by allowing the users with an abstract view of all the features available without worrying about the underlying implementation details. There are some set of parameters required to be set in order to implement the load balancing mechanism, such as configuration of the application, data centers and users.

In our experiment we are using Eclipse, an IDE used by most of the object oriented programming languages and we have implemented our work in JAVA because cloud sim is also written in JAVA, which makes us easy to overcome compatibility issues and also it is an open source utility which means that we need not purchase this utility and also install this, we can directly use it by un packing the library. Once you unpack this package we need to set the java class path and once it is done then this library is ready for use.

4.1 Parameters Used

In this experiment we are considered 5 Datacenters with 100-500 tasks in the simulating platform, 20 VMs, and task length varies from 5000 to 100000 million instructions, File size is up to 500KB with unlimited storage.

The proposed system is being implemented and the results are compared with LBCML (Load Balancing in Cloud Computing Utilizing Machine Learning Techniques) and Throttled algorithms.

No. of Tasks	Throttled	LBCML	ELBVM
--------------	-----------	-------	-------

50	120	100	80
100	230	200	160
150	400	380	300
200	780	700	600
250	1100	1000	900

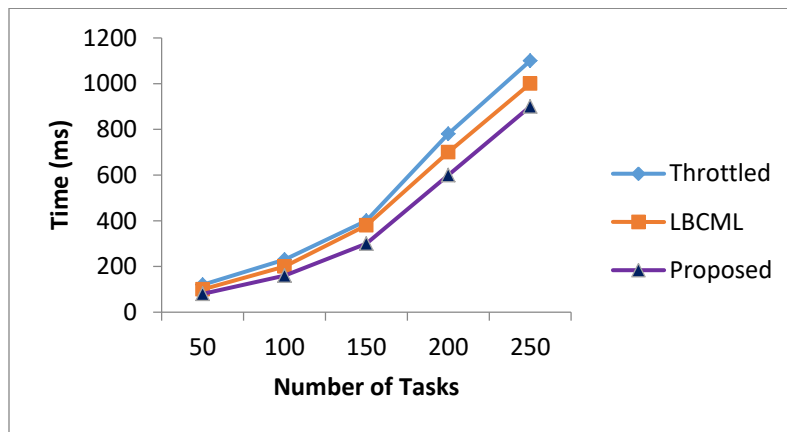


Figure 4.1: Processing Time

Figure 4.1 depicts the processing time consumed to perform the tasks. The results show that the throttled processing time varies from 120ms to 1100ms and LBCML processing time varies from 100ms to 1000ms, whereas the processing time of ELBVMM varies from 80ms to 900 which means that the processing time of ELBVMM is 10% less than the existing algorithms.

No. of Tasks	Throttled	LBCML	ELBVMM
50	0.6	0.59	0.55
100	0.65	0.62	0.57
150	0.69	0.65	0.59
200	0.72	0.7	0.65
250	0.75	0.72	0.68

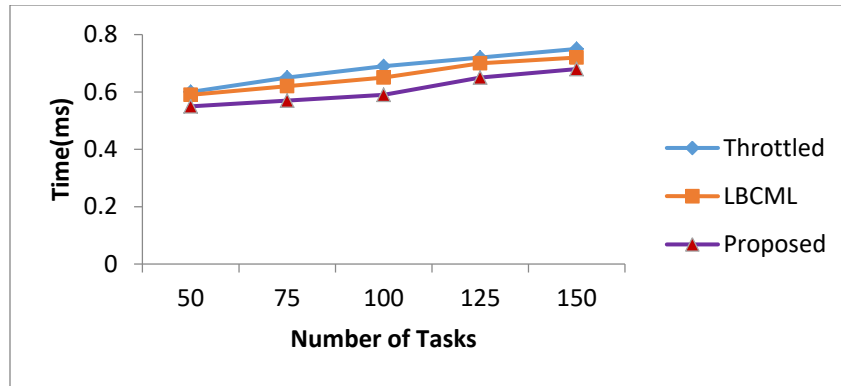


Figure 4.2: Data transfer cost

Figure 4.2 depicts the data transfer cost consumed to perform the tasks. The results show that the throttled data transfer cost varies from 0.6ms to 0.75ms and LBCML data transfer cost varies from 0.59ms to 0.72ms, whereas the data transfer cost of ELBVMM varies from 0.55ms to 0.68 which means that the data transfer cost of ELBVMM is 14% less than the existing algorithms.

No. of Tasks	Throttled	LBCML	ELBVMM
50	0.61	0.59	0.55
100	0.65	0.62	0.57
150	0.69	0.65	0.59
200	0.72	0.71	0.65
250	0.75	0.72	0.68

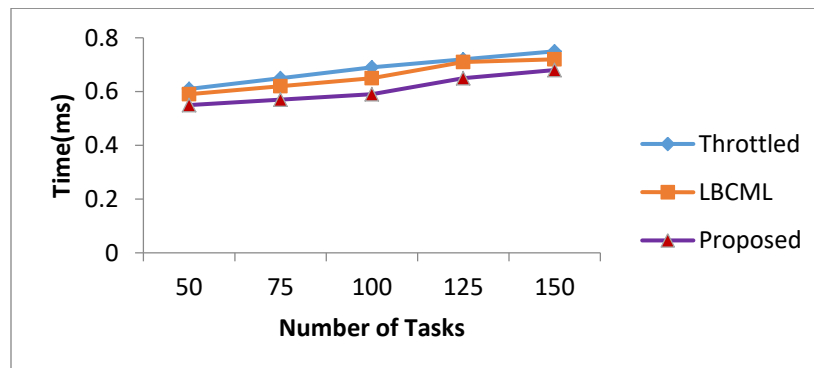


Figure 4.3: Data center processing time

Figure 4.3 depicts the processing time consumed to perform the tasks. The results show that the throttled processing time varies from 0.61ms to 0.75ms and LBCML processing time

varies from 0.59ms to 0.72ms, whereas the processing time of ELBVMM varies from 0.55ms to 0.68 which means that the processing time of ELBVMM is 15% less than the existing algorithms.

No. of Tasks	Throttled	LBCML	ELBVMM
50	0.61	0.59	0.55
100	0.65	0.62	0.57
150	0.69	0.65	0.59
200	0.72	0.71	0.65
250	0.75	0.72	0.68

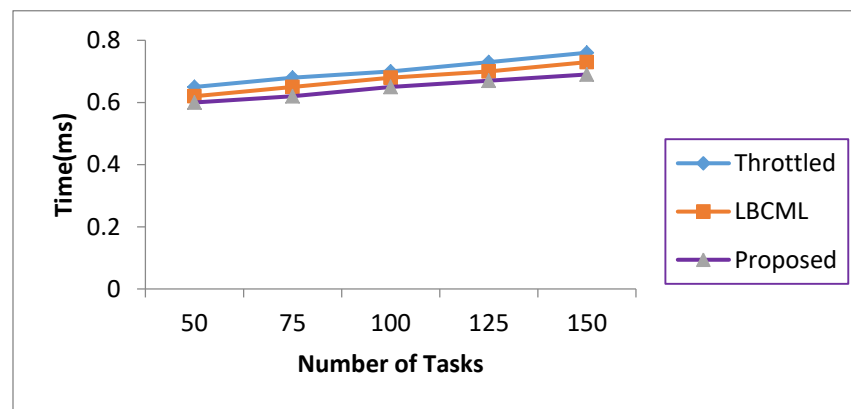


Figure 4.4: Response time

Figure 4.4 depicts the response time consumed to perform the tasks. The results show that the throttled response time varies from 0.61ms to 0.75ms and LBCML response time varies from 0.59ms to 0.72ms, whereas the response time of ELBVMM varies from 0.55ms to 0.68 which means that the response time of ELBVMM is 18% less than the existing algorithms.

5. Conclusion

The proposed system is being implemented in Cloud Sim which is one of the tool used for cloud based experiments. In this paper we proposed an algorithm for dynamically handling the load balancing in the virtual machines migration at the data centers such that the processing time, response time, data transfer cost is being minimized. The results after simulation clearly shows that the proposed algorithm has better load balancing of resources in dynamic cloud environment.

6. References

- [1] Girish Kumar D, Dr. Rajashree V Biradar, Dr. VC Patil, "Multiple Authority Based Data Fragmentation Technique for Providing Secure Storage in Cloud Computing," *International Journal of Recent Technology and Engineering(IJRTE)*, pp. 2277-3878, 2019.
- [2] Girish Kumar D, Dr. Rajashree V Biradar, Dr. V C Patil, "A Novel Technique for Data Integrity using Block Verification in Cloud Computing," *International Journal of Computational and Theoretical Nanoscience*, vol. 15, pp. 1-3, 2018.
- [3] Q. Z. Sheng, X. Qiao, A. V. Vasilakos, C. Szabo, S. Bourne, "Web servies composition, A decade overview," *Information Sciences*, vol. 280, pp. 218-238, 2014.
- [4] Shafiei, M. Soltanshahi R. Asemi and N., "Energy-aware virtual machines allocation by krill herd algorithm in cloud data centers," *Science Direct Heliyon*, vol. 5, no. 7, pp. 1-6, 2019.
- [5] Z. Zhou, J. Yu, f. Li and F. Yang, "Virtual machine migration algorithm for energy efficiency optimization in cloud computing," *Concurrency and Computation: Practice and experience*, vol. 30, no. 24, pp. 1-10, 2018.
- [6] Shaw, S. B., "Balancing load of cloud data center using efficient task scheduling algorithm," *International Journal of Computer Applications*, vol. 159, no. 5, pp. 1-5, 2017.
- [7] G. P. Babu, A. K. Tiwari, "Energy efficient scheduling algorithm for cloud computing system based on predication model," *International Journal of Advanced Networking and Applications*, vol. 10, no. 5, pp. 4013-4018, 2019.
- [8] J. Zhu, J. Wang, Y. Zhang and Y. Jiang, "Virtual Machine migration method based on load cognition," *Soft Computing*, vol. 23, no. 19, pp. 9439-9448, 2019.
- [9] Malony, M. A. H. Moniland and A. D., "QoS-aware virtual machine consolidation in cloud data

- center," *IEEE International Conference on Cloud Engineering*, pp. 81-87, 2017.
- [10] G. Singh, M. Malhotra and A. Sharma, "An agent based virtual machine migration process for cloud environment," *4th International Conference on Internet of Things*, pp. 1-4, 2019.
- [11] N. Akhter, M. Othman and R. K. Naha, "Energy-aware virtual machine selection method for cloud data center resource allocation," *Xiv preprint arXiv:1812.08375*, pp. 1-16, 2018.
- [12] Nayak, D. Garg and P. C., "Hybrid live VM migration: An efficient live VM migration approach in cloud computing," *Springer International Conference on Advanced Informatics for Computing Research*, vol. 955, pp. 600-611, 2018.
- [13] Mr. Girish Kumar D, Dr. Rajashree V Biradar Dr. V C Patil, "A Study on various Cryptographic Techniques used in Cloud Computing," *International Journal of Advanced Research in Dynamical & Control System*, vol. 10, no. 03, 2018.