Self - Adaptive Load Balancing Using Live Migration of Virtual Machines in Cloud Environment

V. Lavanya
Information Technology, SRMIST, Kattankulathur, Chennai. E-mail: lavanyav@srmist.edu.in

M. Saravanan
Information Technology, SRMIST, Kattankulathur, Chennai.
E-mail: saravanm7@srmist.edu.in

E.P. Sudhakar
Computer Science, SRMIST, Kattankulathur, Chennai. E-mail: sudhakarep@gmail.com

Received July 26, 2020; Accepted September 30, 2020
ISSN: 1735-188X
DOI: 10.14704/WEB/V17I2/WEB17064

Abstract

In this paper, a self-adaptive load balancing technique is proposed using live migration of heterogeneous virtual machines (VM) in a Hyper-V based cloud environment. A cloud supported plugin as a management activity within the infrastructure as a service strategy. It is proposed to assist the load balancing process in such a way so that all hypervisors are almost equally loaded once the overload status gets triggered. In the cloud computing environment, load balancing plays a major role if the large number of events triggered has a high impact on the performance of the system. The efficiency of cloud computing is based on the efficient load balancing having a self-adjustable technique using live migration of VMs across clusters of nodes. The proposed load balancing model is efficient in performance improvement by efficient resource utilization and also it helps to avoid the situation occurrence of server hanging by the cause of server overload within the infrastructure of multiple Microsoft Hyper-V hypervisors environment.

Keywords

Hyper-v, Virtual Machine, Cloud Environment, Load Balancing, VM Migration.

Introduction

[1]Cloud computing plays a predominant role in the computing environment provides various resources as a service. The resources provided by the cloud are hardware, memory, processor, network, and storage and these are provided as infrastructure as a service, platform as a service, and software as a service by on-demand pay per use model.

[2]According to the National Institute of Standards and Technology (NIST), the important characteristics of cloud computing are elasticity, scalability, on-demand self-service, broad network access, Resource Pooling, and Measured Service. [3]The cloud is deployment based on the cloud users' preferences like public, private, hybrid, and community cloud.

[4]Easticity and scalability are the important characteristics of cloud computing. It deals with adapting the nature of workload changes by providing and releasing the resources based on the assigned resource with the current requirement in the system automatically. So, it is based on the need, the provisioning or de-provisioning of resources will be provided by the application with elasticity.

Based on the current resource demand on the cloud, the service provider needs to allocate the required quantity of resources to the cloud users. [5]Under-provisioning of the resource requested by the users causes violation in the Service Level Agreements, leads to dropping in Quality of Service(QoS), and finally the customer dissatisfaction. This may lead to the loss of customers and a decrease in revenue.

Then again, Over-provisioning of resources prompts vitality and assets squander and it even expands systems administration, upkeep, and cooling costs. Due to this, the management of the resources available in the cloud becomes a confusing procedure and a productive Load adjusting method is required[6,7].

[8]Effective balancing of the workload and proper scheduling and allocation of resources are major aspects of efficient resource utilization in the cloud environment. There is a need for efficient load balancing for efficient and optimized use of resources available in cloud (processors, memory, disks) and to exacts high performance in the machines. The performance of the system lies on the load balancing.

[9]The clustered virtualized environment consists of many Hyper-v Servers which are clustered by hyper-v failover cluster service to ensure the high availability of the server and also to ensure the avoidance of single point of failure (SPOF).

In this research paper, the proposed methodology is based on the self - adaptive load balancing achieved by migrating the virtual machines live in the cloud environment to achieve its efficient load balancing of the virtual machine using live migration in the cloud environment.

Related Work

In this section, the various researches work on load balancing is discussed. Gaochao Xu[10] proposed the conceptual framework which divides the public cloud into many cloud partitions and load is balanced among the cloud partitions using a round-robin algorithm. Sandeep Bhargava[11] proposed a methodology for redistributing the load among various servers in a distributed environment.

Alam[12] presented a searching technique which groups the tasks based on the priority of the task. Alharbi[13] proposed an algorithm that maximizes the utilization of the resource by reducing the makespan. Toosi and Buyya[14] proposed a fuzzy logic-based technique for balancing the load. Hadji and Zeghlache[15] presented a model using a modified Bin-Packing technique in the cloud for optimal placement of resources as a maximum flow problem to serve the increased requirements of the user. Panwar and Mallick[16] proposed an algorithm that performs dynamic load balancing based on the present status all servers considered and figures the response time of their calculation. Ranjan Kumar and G Sahoo[17] proposed the Ant colony optimization technique for effective load balancing improves the overall performances of the system. Ratan Mishra and Anant Jaiswal[18] provides a solution to the effective load balancing problem using Ant colony optimization technique by decreasing the makespan of the system. Shagufta Khan and Niresh Sharma[19] combined the optimization technique along with the scheduling algorithm for effective load balancing.

Present Work

The improvement in stability and performance of the system is based on the proper control of workload. It uses the available resources within the cloud environment efficiently such that all the resources are used to achieve better performance.

The centralized cloud manager can be the solution for better load balancing across the cloud and it can be deployed outside of the cloud with in the same network domain so that it can run independently with the cloud.

The cloud manageable plugin or standalone application can be installed in any of the client machines in the network which is capable of fetching the system information by remote procedure calls and continuously monitor the processor load of every hosting server. It saves the processor load history to keep track of the server’s load and decides to migrate the virtual machines to balance the load between the servers by making the optimal decision with the optimal decision logic.

In the overall architecture, the computational nodes and the centralized monitoring client machines are connected through the network such that the client manger can get the information of the computational nodes through the remote procedure calls.

Self Adjustable Load Balancer

In the overall architecture, the computational nodes and the centralized monitoring client machines are connected through the network such that the client manger can get the information of the computational nodes through the remote procedure calls.

Models Need for Load Balancing

The system which needs load balancing consists of a Clustered virtualized environment (Using Hyper-v) and the centralized load balancer which has been installed as a plugin.

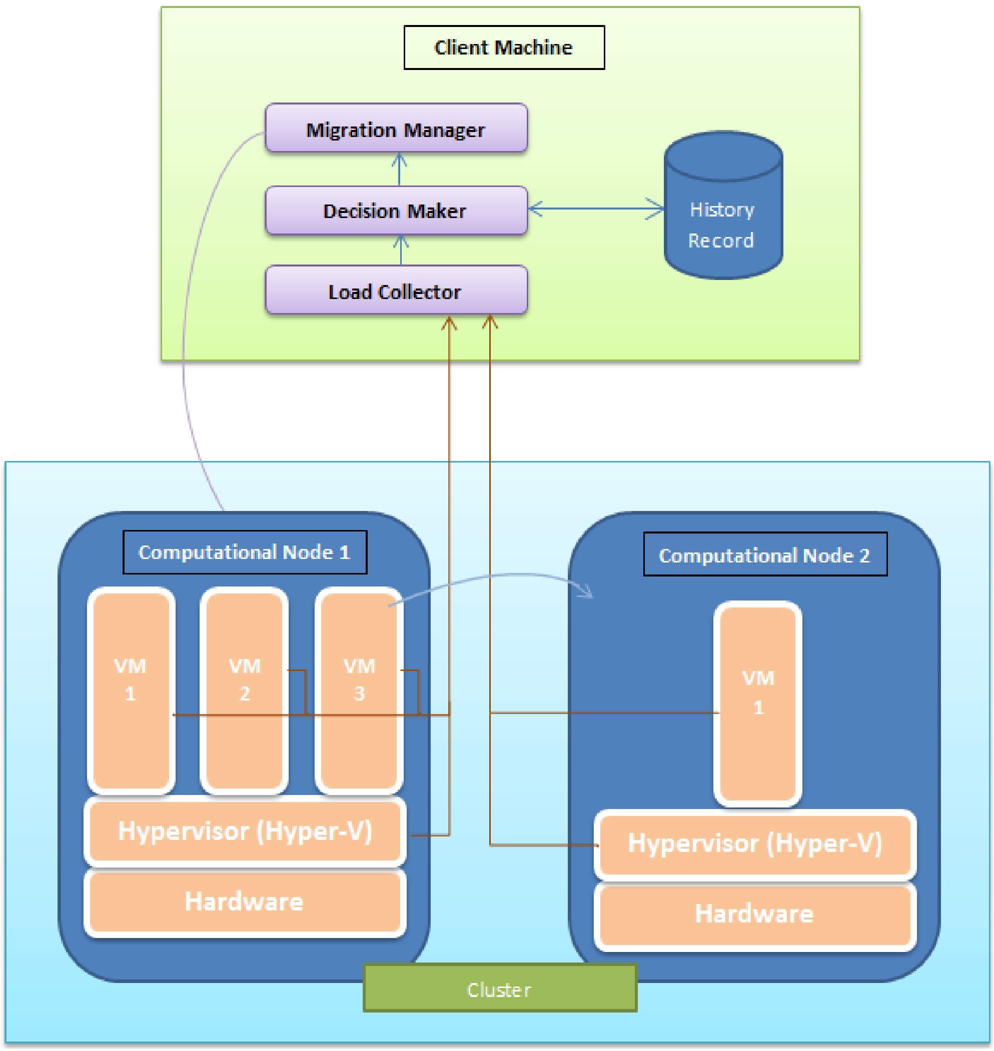

Figure 1 System Architecture

a) Clustered Virtualized Environment

The clustered virtualized environment consists of many Hyper-v Servers which are used to create the virtualized environment and which are also clustered by hyper-v failover cluster service to ensure the high availability of the server and also to ensure the avoidance of single point of failure (SPOF).

The clustered Hyper-v servers are deployed with several running virtual machines which are running with heterogeneous operating systems and clustered through hyper-v cluster service as below figure:

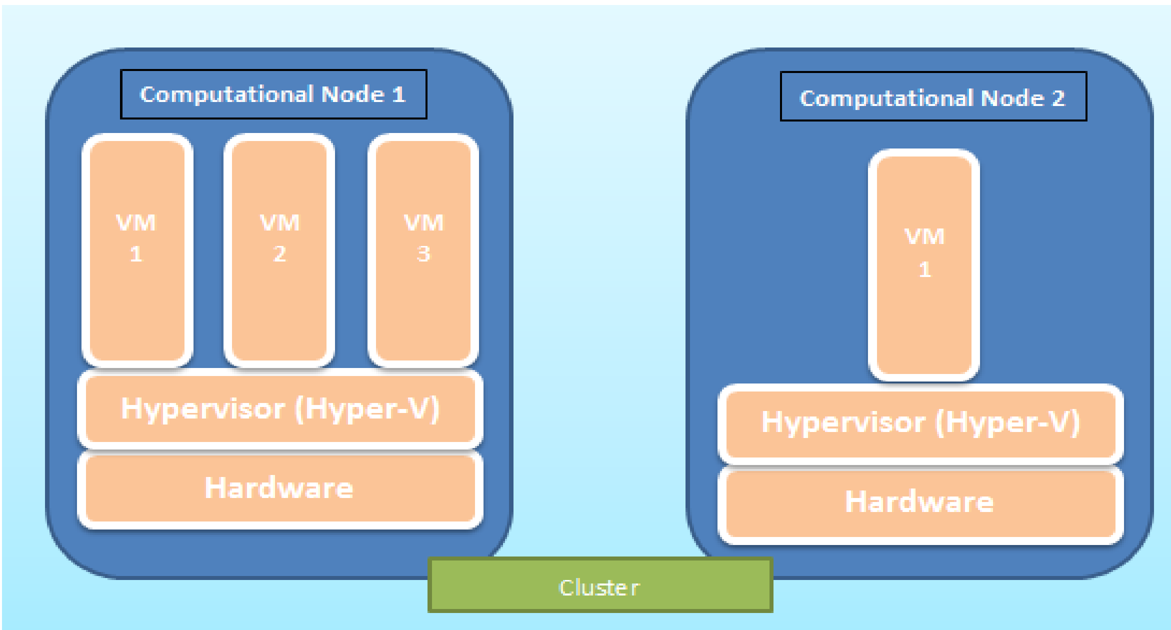

Figure 2 Clustered Hyper-V environment

b) Centralized Load Balancer

The cloud manageable plugin has been installed in one client machine in the network which is capable of fetching the system information by remote procedure calls and continuously monitor the processor resource load of every host servers and migrate the virtual machines to make an optimal solution by performing load balancing between the servers in the cloud.

The cloud load balancer can be divided into three functional modules as Load Collector, Decision Maker, Migration Manager.

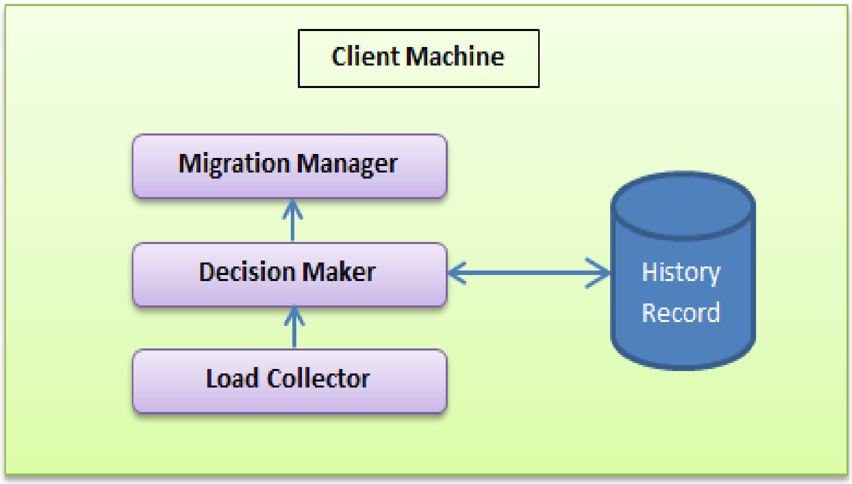

Figure 3 Load balancer

c) Load Collector

This module is to collect and update the CPU load continuously from the host hypervisor servers and its hosted VMs. The load collector module continuously collects and monitors the hyper-v servers' processor load through fetching the instances of the "MSVM_Processor" class.

This WMI class gives the instances and their values for each running virtual processor.

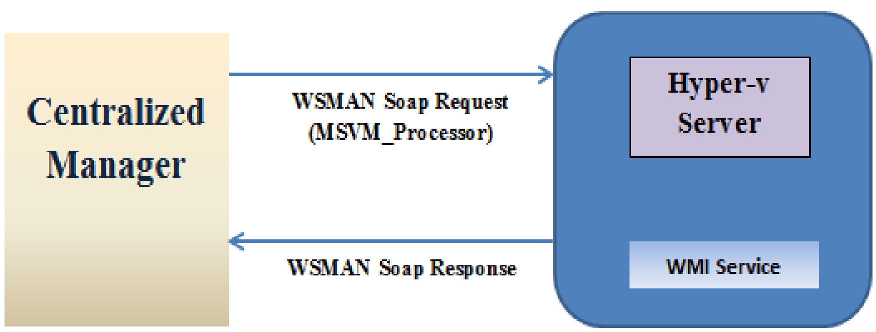

Figure 4 Load Collector

d) Decision Maker

This module works with the load collector. As the load collector continuously monitor the current load of all the servers, the decision-maker module gets the current processor load of each server and decide with an efficient computational logic.

This module works with load collected, this module makes the decisions whether to migrate or continue running the VMs in the same hosted server by checking the current threshold against the calculated average and maximum thresholds. If any of the servers are overloaded then this module selects the VM which needs to be migrated and the destination server to which the migration needs to be done, such that the VM can be run in the probable destination server without shortage for the resource for longer. This module ensures the optimal decision by selecting the proper VM and destination server.

e) Migration Manager

This module works with the decision-making module. The decision-making module decides to migrate the VM/VMs by selecting the VMs which are suitable for migration and also the destination server to achieve the server load balancing.

The migration manager initiates the migration based on the entry from the decision-maker module and continuously monitors the migration status after the migration has been initiated. This module also adds the database entry for the initiated migrations to maintain the history data to trace and for making better decisions based on the migration history and its status.

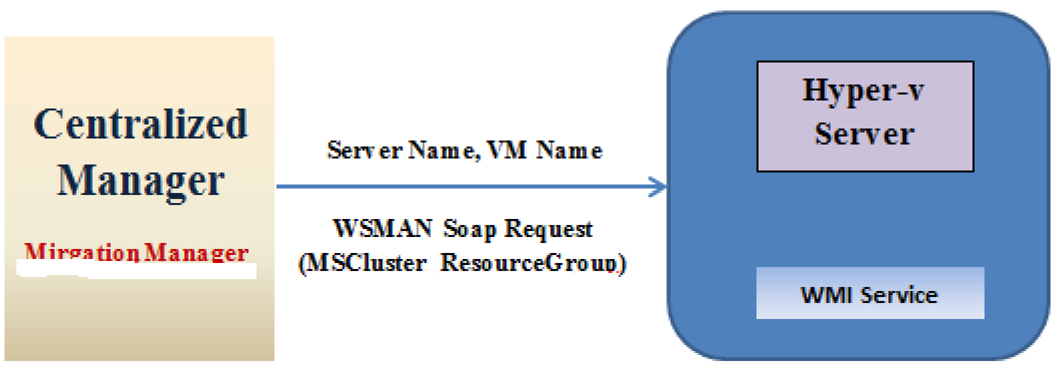

Figure 5 Migration Manager

Decision-Making Logic

The live migration of the virtual machines will be triggered if and only if C_m, the average value of the sum of the maximum and minimum utilization of the processor load percentage values (respectively $C_{diffmax}$ and $C_{diffmin}$) from the average utilization (C_{avg}), should be lesser than the threshold T (The VMs processor utilization).

C_i is the total processor utilization of the i^{th} computational node, where i belongs to 1,2,3,…,n;

Where n is the currently available number of nodes in the Clustered environment.

$$C_{avg} = 1/n \sum^{n}_{i=1} C_i$$
$$C_{diff\ i} = C_i - C_{avg}$$
$$C_{diffmax} = max_{i\in\{1,..,n\}}\ C_{diff\ i}$$
$$C_{diffmin} = min_{i\in\{1,..,n\}}\ C_{diff\ i}$$
$$C_m = (C_{diffmax} + C_{diffmin})/2$$

From these calculations, the node which gives $C_{diffmax}$ CPU usage percent load is selected as the source node and the node with the $C_{diffmin}$ selected as the destination node.

If the VM's current load is lesser than the calculated Cm then the VMs are inversely sorted according to the current processor load and initiates the migration.

Overall Process Flow

The overall process flow includes collecting the workload from the server in day to day basis. From the data collected calculate the average value of the total sum of the maximum and minimum load percentage of the processor. If the sum of the maximum and minimum processor load percentage values from the average utilization is lesser than a threshold T (The utilization of the VMs processor), the system has to continue to collect the values. If it exceeds the threshold, the system has to decide to initiate the migration.

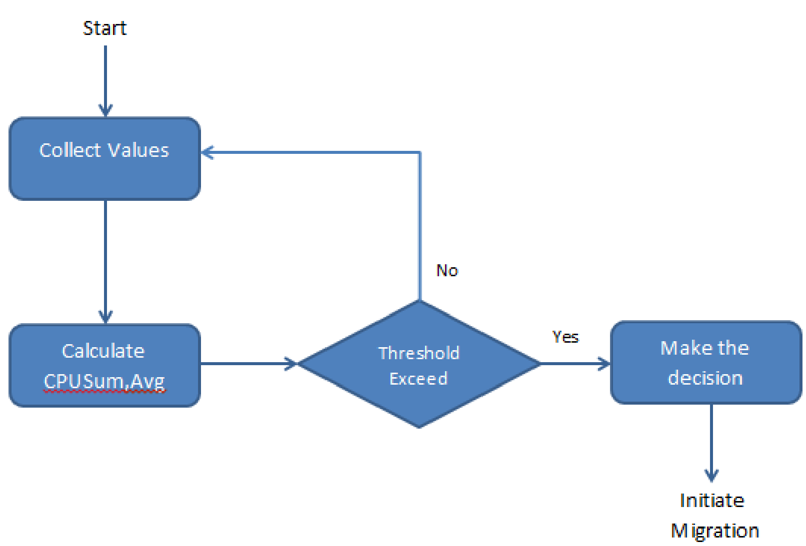

Figure 6 Overall Process flow

Server Load Analysis

The server workload is analyzed throughout the day before and after implementing the centralized load balancer. The results are figured using the chart. Chart 1 shows the results of the server workload are analyzed throughout the day before implementing the centralized load balancer and the results are figured in the below chart,

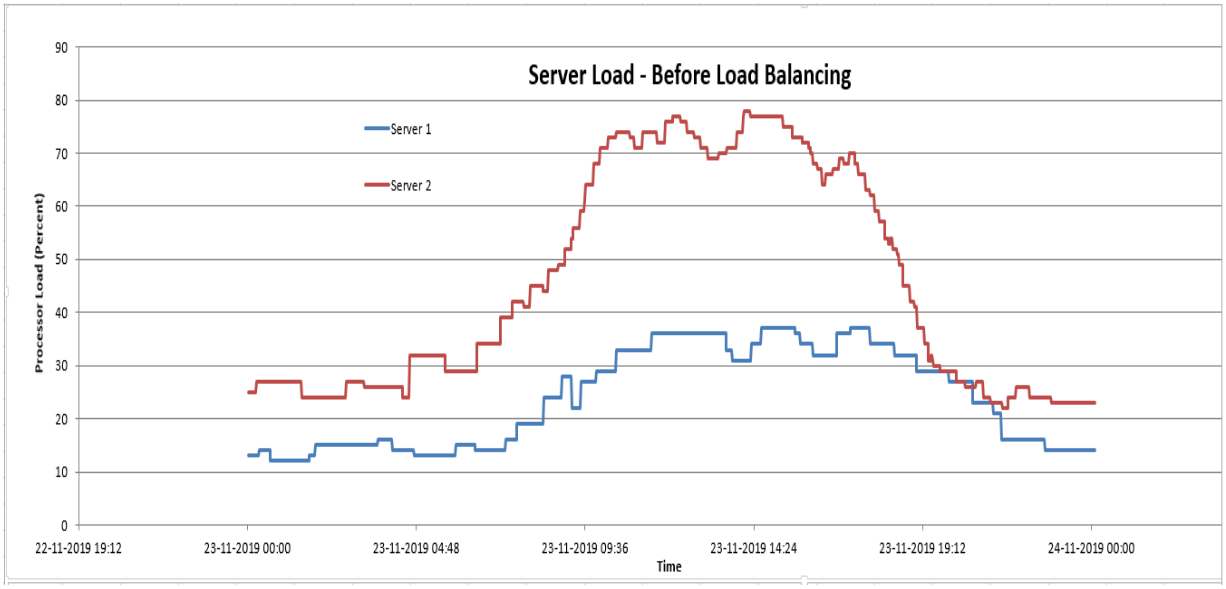

Chart 1 Before Load Balancing

In chart2, the server workload has been analyzed throughout the day after deploying the centralized load balancer.

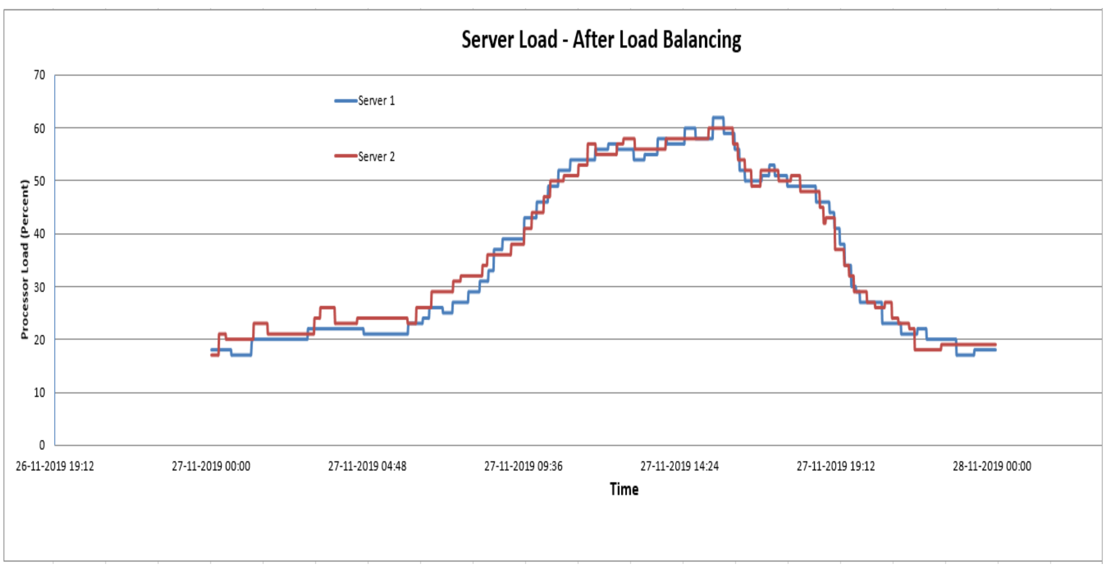

Chart 2 After Load Balancing

The above two figures clearly show the difference of the server load before and after proper usage of self – adjustable load balancer during live migration of virtual machines in the cloud environment. In chart 1, without using self - adjustable load balancer, there will be a major difference in request and response to the requirement of the resource. But in chart 2, after using the self – adjustable load balancer, the requirement of the resource is properly satisfied.

Conclusion

Thus the proposed system continuously attempts to reduce the differences in loads (balance the load) among all the server nodes of the clustered cloud environment. By proficiently choosing and moving the best reasonable VMs from intensely loaded to daintily loaded nodes the load balancer guarantees the load adjusting among all the nodes. The load adjusting component is incorporated rather than appropriated the board, with the goal that the general framework can be controlled from a solitary point. The control node runs the cloud the executives programming that controls all the nodes and running VMs. From a framework level point, the proposed framework takes care of the irregularity issue to such an extent that a computational node is intensely loaded while others are gently loaded will take a load adjusting choice to offset the load with the best appropriate node over the suitable environment.

References

Hyper-V Planning and Deployment Guide, Microsoft Corporation 2009.

Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. *Recommendations of the National Institute of Standards and Technology*.

Savu, L. (2011). Cloud computing: Deployment models, delivery models, risks, and research challenges. *In IEEE International Conference on Computer and Management (CAMAN)*, 1-4.

Gong, C., Liu, J., Zhang, Q., Chen, H., & Gong, Z. (2010). The characteristics of cloud computing. *In IEEE 39th International Conference on Parallel Processing Workshops*, 275-279.

Patel, K.S., & Sarje, A.K. (2012). VM provisioning method to improve the profit and SLA violation of cloud service providers. *In IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, 1-5.

Mishra, N.K., & Mishra, N. (2015). Load balancing techniques: need, objectives, and major challenges in cloud computing-a systematic review. *International Journal of Computer Applications*, *131*(18), 0975-8887.

Kansal, N.J., & Chana, I. (2012). Cloud load balancing techniques: A step towards green computing. *IJCSI International Journal of Computer Science Issues*, *9*(1), 238-246.

Sidhu, A.K., & Kinger, S. (2013). Analysis of load balancing techniques in cloud computing. *International Journal of computers & technology*, *4*(2), 737-741.

http://www.crn.com/news/applications-os/199901716/10-tips-for-deploying-virtualization-technology.htm

Xu, G., Pang, J., & Fu, X. (2013). A load balancing model based on cloud partitioning for the public cloud. *Tsinghua Science and Technology*, *18*(1), 34-39.

Bhargava, S., & Goyal, S. (2013). Dynamic Load Balancing in Cloud Using Live Migration of Virtual Machine. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, *2*(8), 2472-2477.

Alam, M.I., Pandey, M., & Rautaray, S.S. (2014). A proposal of resource allocation management for cloud computing. *International Journal of Cloud Computing and Services Science*, *3*(2), 79-86.

Alharbi, F., & Rabigh, K.S.A. (2012). Simple scheduling algorithm with load balancing for grid computing. *Asian Transactions on Computers*, *2*(2), 8-15.

Toosi, A.N., & Buyya, R. (2015). A fuzzy logic-based controller for cost and energy efficient load balancing in geo-distributed data centers. *In IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC)*, 186-194.

Hadji, M., & Zeghlache, D. (2012). Minimum cost maximum flow algorithm for dynamic resource allocation in clouds. *In IEEE Fifth International Conference on Cloud Computing*, 876-882.

Panwar, R., & Mallick, B. (2015). Load balancing in cloud computing using dynamic load management algorithm. *In International Conference on Green Computing and Internet of Things (ICGCIoT)*, 773-778.

Ranjan, K., & Sahoo, G. (2013). Load Balancing Using Ant Colony In Cloud Computing. *International Journal of Information Technology & Computer Science*, 3(5).

Mishra, R., & Jaiswal, A. (2012). Ant colony optimization: A solution of load balancing in cloud. *International Journal of Web & Semantic Technology*, *3*(2), 33-50.

Khan, S., & Sharma, N. (2014). Effective scheduling algorithm for load balancing (SALB) using ant colony optimization in cloud computing. *International Journal of Advanced Research in Computer Science and Software Engineering*, *4*(2), 966-973.