A COMPARATIVE STUDY OF DIFFERENT LOAD BALANCING ALGORITHMS IN CLOUD COMPUTING

Parminder Kaur
Lecturer, Department of Computer Science Engineering, SBSSTC, Ferozepur
(E-mail: parminder92sandhu@gmail.com)

***Abstract*—** Cloud computing is Internet based development and use of computer technology. It is a style of computing in which dynamically scalable and often virtualized resources are provided as a service over the Internet. Users need not have knowledge of, expertise in, or control over the technology infrastructure "in the cloud" that supports them. Scheduling is one of the core steps to efficiently exploit the capabilities of heterogeneous computing systems. On cloud computing platform, load balancing of the entire system can be dynamically handled by using virtualization technology through which it becomes possible to remap virtual machine and physical resources according to the change in load. However, in order to improve performance, the virtual machines have to fully utilize its resources and services by adapting to computing environment dynamically. The load balancing with proper allocation of resources must be guaranteed in order to improve resource utility.

Keywords— Cloud Computing, Load Balancing, Virtual Machine, Packages.

I. INTRODUCTION

Cloud Computing (CC) [1] is an emerging technology that has abstruse connection to Grid Computing (GC) paradigm and other relevant technologies such as utility computing, distributed computing and cluster computing. The aim of both GC and CC is to achieve resource virtualization. In spite of the aim being similar, GC and CC have significant differences. The main emphasis of GC is to achieve maximum computing, while that of CC is to optimize the overall computing capacity. CC also provides a way to handle wide range of organizational needs by providing dynamically scalable servers and application to work with. Leading CC service providers such as Amazon, IBM, `Dropbox', Apple's `iCloud', Google's applications, Microsoft's `Azure', etc., are able to attract normal users through out the world. CC have introduced a new paradigm, which helps its users to store or develop applications dynamically and access them from anywhere and anytime just by connecting to an application using Internet. Depending on customer's requirement CC provides easy and customizable services to access or work with cloud applications. Based on the user requirement CC can be used to provide platform for designing applications, infrastructure to store and work on company's data and also provide applications to do user's routine tasks. When a customer chooses to use cloud services, data stored in the local repositories will be sent to a remote data center. This data in remote locations can be accessed or managed with the help of services provided by cloud service providers. This makes clear that for a user to store or process a piece of data in cloud, he/she needs to transmit the data to a remote server over a channel (internet). This data processing and storage needs to be done with utmost care to avoid data breaches.

It is the model for convenient on-demand network access, with minimum management efforts for easy and fast network access to resources that are ready to use. It is an upcoming paradigm that offers tremendous advantages in economic aspects, such as reduced time to market, flexible computing capabilities, and limitless computing power. Popularity of cloud computing is increasing day by day in distributed computing environment. There is a growing trend of using cloud environments for storage and data processing needs. To use the full potential of cloud computing, data is transferred, processed, retrieved and stored by external cloud providers. However, data owners are very skeptical to place their data outside their own control sphere.

II. LOAD BALANCING

One of the foremost usually used applications of load balancing is to produce quality of service from multiple servers, typically called a server data center. Usually load-balanced systems are properly working inside popular internet sites, big chat networks, high-bandwidth file transfer protocol sites, and domain name System (DNS) servers. It additionally prevents the clients from contacting back-end servers directly, which can have security advantages by hiding the structure of the inner network. Some load balancers give a mechanism for improving the one parameter specially within back end server Load balancing offers the IT team an opportunity to attain a considerably higher fault tolerance. It will mechanically give the capability required to handle any increase or decrease of application traffic.

It is additionally necessary that the load balancer itself doesn't become the cause of failure. Sometimes load balancers enforced in high-availability servers can additionally replicate the user's session needed by the application. Load balancing is dividing work load between a set of computers in order to receive the good response time and all the nodes are equally loaded and, in general, all users get served quicker. Load balancing may be enforced with hardware, software, or a mix of each. Typically, load balancing is that the main reason for server's unbalanced response time. Load balancing plans to

optimize the usage of resources, maximize overall success ratio, minimize waiting time interval, and evade overloading of the resources. By the utilization of multiple algorithms and mechanisms with load balancing rather than one algorithm might increase reliability and efficiency. Load balancing within the cloud differs from classical thinking on load balancing design and implementation by misusage of data center servers to perform the requests on the basis of first come first serve basis. The older load balancing algorithm allocates the requests according to the incoming requests of the client.

III. RELATED WORK

Nguyen Khac Chien et al. (2016) has proposed a load balancing algorithm which is used to enhance the performance of the cloud environment based on the method of estimating the end of service time. They have succeeded in enhancing the service time and response time of the user.

Ankit Kumar et al (2016) focuses on the load balancing algorithm which distributes the incoming jobs among VMs optimally in cloud data centers. The proposed algorithm in this research work has been implemented using Cloud Analyst simulator and the performance of the proposed algorithm is compared with the three algorithms which are preexists on the basis of response time. In the cloud computing milieu, the cloud data centers and the users of the cloud-computing are globally situated, therefore it is a big challenge for cloud data centers to efficiently handle the requests which are coming from millions of users and service them in an efficient manner.

S.Yakhchi et al. (2015) discusses that the energy consumption has become a major challenge in cloud computing infrastructures. They proposed a novel power aware load balancing method, named ICAMMT to manage power consumption in cloud computing data centers. We have exploited the Imperialism Competitive Algorithm (ICA) for detecting over utilized hosts and then we migrate one or several virtual machines of these hosts to the other hosts to decrease their utilization. Finally, we consider other hosts as underutilized host and if it is possible, migrate all of their VMs to the other hosts and switch them to the sleep mode.

Surbhi Kapoor et al. (2015) aims at achieving high user satisfaction by minimizing response time of the tasks and improving resource utilization through even and fair allocation of cloud resources. The traditional Throttled load balancing algorithm is a good approach for load balancing in cloud computing as it distributes the incoming jobs evenly among the VMs. But the major drawback is that this algorithm works well for environments with homogeneous VMS, does not considers the resource specific demands of the tasks and has additional overhead of scanning the entire list of VMs every time a task comes. The issues have been addressed by proposing an algorithm Cluster based load balancing which works well in heterogeneous nodes environment, considers resource specific demands of the tasks and reduces scanning overhead by dividing the machines into clusters.

Shikha Garg et al. (2015) aims to distribute workload among multiple cloud systems or nodes to get better resource utilization. It is the prominent means to achieve efficient resource sharing and utilization. Load balancing has become a challenge issue now in cloud computing systems. To meets the user's huge number of demands, there is a need of distributed solution because practically it is not always possible or cost efficient to handle one or more idle services. Servers cannot be assigned to particular clients individually. Cloud Computing comprises of a large network and components that are present throughout a wide area. Hence, there is a need of load balancing on its different servers or virtual machines. They have proposed an algorithm that focuses on load balancing to reduce the situation of overload or under load on virtual machines that leads to improve the performance of cloud substantially.

Reena Panwar et al. (2015) describes that the cloud computing has become essential buzzword in the Information Technology and is a next stage the evolution of Internet, The Load balancing problem of cloud computing is an important problem and critical component adequate operations in cloud computing system and it can also prevent the rapid development of cloud computing. Many clients from all around the world are demanding the various services rapid rate in the recent time. Although various load balancing algorithms have been designed that are efficient in request allocation by the selection of correct virtual machines. A dynamic load management algorithm has been proposed for distribution of the entire incoming request among the virtual machines effectively.

Mohamed Belkhouraf et al. (2015) aims to deliver different services for users, such as infrastructure, platform or software with a reasonable and more and more decreasing cost for the clients. To achieve those goals, some matters have to be addressed, mainly using the available resources in an effective way in order to improve the overall performance, while taking into consideration the security and the availability sides of the cloud. Hence, one of the most studied aspects by researchers is load balancing in cloud computing especially for the big distributed cloud systems that deal with many clients and big amounts of data and requests. The proposed approach mainly ensures a better overall performance with efficient load balancing, the continuous availability and a security aspect.

Lu Kang et al. (2015) improves the weighted least connections scheduling algorithm, and designs the Adaptive Scheduling Algorithm Based on Minimum Traffic (ASAMT). ASAMT conducts the real-time minimum load scheduling to the node service requests and configures the available idle resources in advance to ensure the service QoS requirements. Being adopted for simulation of the traffic scheduling algorithm, OPNET is applied to the cloud computing architecture.

Hiren H. Bhatt et al. (2015) presents a Flexible load sharing algorithm (FLS) which introduce the third function. The third function makes partition the system in to domain. This function is helpful for the selection of other nodes which are present in the same domain. By applying the flexible load sharing to the particular domains in to the distribute system, the performance can be improved when any node is in overloaded situation.

IV. LOAD BALANCING ALGORITHMS

A set of heuristic algorithms has been designed to schedule meta-tasks to heterogeneous computing systems. It is assumed that the heuristic derive mapping statically for the collection of independent meta-task. The scheduling problem is computationally hard even though there are no data dependencies among the jobs.

A. OLB: Opportunistic Load Balancing (OLB) assigns each task, in arbitrary order, to the next machine that is expected to be available, regardless of the task's expected execution time on that machine. The intuition behind OLB is to keep all machines as busy as possible [8, 9].

B. MET: In contrast to OLB, Minimum Execution Time (MET) assigns each task, in arbitrary order, to the machine with the best expected execution time for that task, regardless of that machine's availability. The motivation behind MET is to give each task to its best machine. This can cause a severe load imbalance across machines.

C. MCT: Minimum Completion Time (MCT) assigns each task, in arbitrary order, to the machine with the minimum expected completion time for that task. This causes some tasks to be assigned to machines that do not have the minimum execution time for them [1].

D. Min-min: Min-min heuristic uses minimum completion time (MCT) as a metric, meaning that the task which can be completed the earliest is given priority. This heuristic begins with the set U of all unmapped tasks. Then the set of minimum completion times (M), is found.

E. Max-Min: The Max-min heuristic is very similar to Min-min and its metric is MCT too. It begins with the set U of all unmapped tasks. Then, the set of minimum completion times (M) is found as mentioned in previous section. Next, the task with the overall maximum completion time from M is selected and assigned to the corresponding machine and the workload of the selected machine will be updated. And finally the newly mapped task is removed from U and the process repeats until all tasks are mapped [1, 9].

F. LJFR-SJFR: LJFR-SJFR heuristic begins with the set U of all unmapped tasks. Then the set of minimum completion times is found the same as Min-min. Next, the task with the overall minimum completion time from M is considered as the shortest job in the fastest resource (SJFR). Also the task with the overall maximum completion time from M is considered as the longest job in the fastest resource (LJFR). At the beginning, this method assigns the m longest tasks to the m available fastest resources (LJFR). Then this method assigns the shortest task to the fastest resource, and the longest task to the fastest resource alternatively [4, 11].

G. Sufferage: In this heuristic for each task, the minimum and second minimum completion time are found in the first step. The difference between these two values is defined as the sufferage value. In the second step, the task with the maximum sufferage value is assigned to the corresponding machine with minimum completion time [4, 12].

H. Work Queue: This heuristic is a straightforward and adaptive scheduling algorithm for scheduling sets of independent tasks. In this method, the heuristic selects a task randomly and assigns it to the machine as soon as it becomes available (in other word, machine with minimum workload) [4, 13].

V. RESEARCH GAP

Cloud computing thus involving distributed technologies to satisfy a variety of applications and user needs. Sharing resources, software, information via internet are the main functions of cloud computing with an objective to reduced capital and operational cost, better performance in terms of response time and data processing time, maintain the system stability and to accommodate future modification in the system .So there are various technical challenges that needs to be addressed like Virtual machine migration, server consolidation, fault tolerance, high availability and scalability but central issue is the load balancing , it is the mechanism of distributing the load among various nodes of a distributed system to improve both resource utilization and job response time while also avoiding a situation where some of the nodes are heavily loaded while other nodes are idle or doing very little work. It also ensures that all the processor in the system or every node in the network does approximately the equal amount of work at any instant of time. To make the final determination, the load balancer retrieves information about the candidate server's health and current workload in order to verify its ability to respond to that request. Load balancing solutions can be divided into software-based load balancers and hardware-based load balancers. Hardware-based load balancers are specialized boxes that include Application Specific Integrated Circuits (ASICs) customized for a specific use. They have the ability to handle the high-speed network traffic whereas Software-based load balancers run on standard operating systems and standard hardware components.

Clients request for the virtual machine and cloud broker handles the client request according to available virtual machine. If the VM is idle, then broker allocate that VM to user for the processing and if VM is not free then incoming requests of the client are send into the waiting state until the resources are free.

1. The current algorithms will work only in the homogeneous cloud system where all the resources are of same configuration.

2. Cloudlets are not assigned to the virtual machine according to their capacities. There may be a scenario where a cloudlet with highest priority is assigned to the machine with lowest capacity in the host.

3. The processing capacity (No of processors / MIPS) is not considered for assigning the VM to a job.

4. Every time before allocation, extra overhead is involved in parsing the table of virtual machines from top to bottom.

VI. CLOUD SIM

Cloud service providers charge users depending upon the space or service provided. In R&D [16], it is not always possible to have the actual cloud infrastructure for performing experiments. For any research scholar, academician or scientist, it is not feasible to hire cloud services every time and then execute their algorithms or implementations. For the

purpose of research, development and testing, open source libraries are available, which give the feel of cloud services. Nowadays, in the research market, cloud simulators are widely used by research scholars and practitioners, without the need to pay any amount to a cloud service provider.

Tasks performed by cloud simulators :

The following tasks can be performed with the help of cloud simulators:

• Modelling and simulation of large scale cloud computing data centres.

• Modelling and simulation of virtualised server hosts, with customisable policies for provisioning host resources to VMs.

• Modelling and simulation of energy-aware computational resources.

• Modelling and simulation of data centre [18] network topologies and message-passing applications.

• Modelling and simulation of federated clouds.

• Dynamic insertion of simulation elements, stopping and resuming simulation.

• User-defned policies for allocation of hosts to VMs, and policies for allotting host resources to VMs.

The scope and features of cloud simulations include:

• Data centres

• Load balancing

• Creation and execution of cloudlets

• Resource provisioning

• Scheduling of tasks

• Storage and cost factors

VII. CONCLUSION

This paper is based on cloud computing technology which has a very vast potential and is still unexplored. The capabilities of cloud computing are endless. Cloud computing provides everything to the user as a service which includes platform as a service, application as a service, infrastructure as a service. One of the major issues of cloud computing is load balancing because overloading of a system may lead to poor performance which can make the technology unsuccessful. So there is always a requirement of efficient load balancing algorithm for efficient utilization of resources. Our paper focuses on the various load balancing algorithms and their applicability in cloud computing environment.

VIII. REFERENCES

[1] S. Yakhchi, S. Ghafari, M. Yakhchi, M. Fazeli and A. Patooghy, "ICA-MMT: A Load Balancing Method in Cloud Computing Environment," IEEE, 2015.

[2] S. Kapoor and D. C. Dabas, "Cluster Based Load Balancing in Cloud Computing," IEEE, 2015.

[3] S. Garg, R. Kumar and H. Chauhan, "Ef?cient Utilization of Virtual Machines in Cloud Computing using Synchronized Throttled Load Balancing," 1st International Conference on Next Generation Computing Technologies (NGCT-2015), pp. 77-80, 2015.

[4] R. Panwar and D. B. Mallick, "Load Balancing in Cloud Computing Using Dynamic Load Management Algorithm," IEEE, pp. 773-778, 2015.

[5] M. Belkhouraf, A. Kartit, H. Ouahmane, H. K. Idrissi,, Z. Kartit and M. . E. Marraki, "A secured load balancing architecture for cloud computing based on multiple clusters," IEEE, 2015.

[6] L. Kang and X. Ting, "Application of Adaptive Load Balancing Algorithm Based on Minimum Traffic in Cloud Computing Architecture," IEEE, 2015.

[7] N. K. Chien, N. H. Son and H. D. Loc, "Load Balancing Algorithm Based on Estimating Finish Time of Services in Cloud Computing," ICACT, pp. 228-233, 2016.

[8] H. H. Bhatt and H. A. Bheda, "Enhance Load Balancing using Flexible Load Sharing in Cloud Computing," IEEE, pp. 72-76, 2015.

[9] S. S. MOHARANA, R. D. RAMESH and D. POWAR, "ANALYSIS OF LOAD BALANCERS IN CLOUD COMPUTING," International Journal of Computer Sciencand Engineering (IJCSE) , pp. 102-107, 2013.

[10] M. P. V. Patel, H. D. Patel and . P. J. Patel, "A Survey On Load Balancing In Cloud Computing," International Journal of Engineering Research & Technology (IJERT), pp. 1-5, 2012.

[11] R. Kaur and P. Luthra, "LOAD BALANCING IN CLOUD COMPUTING," Int. J. of Network Security, , pp. 1-11, 2013.

[12] Kumar Nishant, , P. Sharma, V. Krishna, Nitin and R. Rastogi, "Load Balancing of Nodes in Cloud Using Ant Colony Optimization," IEEE, pp. 3-9, 2012.

[13] Y. Xu, L. Wu, L. Guo,, Z. Chen, L. Yang and Z. Shi, "An Intelligent Load Balancing Algorithm Towards Ef?cient Cloud Computing," AI for Data Center Management and Cloud Computing: Papers from the 2011 AAAI Workshop (WS-11-08), pp. 27-32, 2011.

[14] A. K. Sidhu and S. Kinger, "Analysis of Load Balancing Techniques in Cloud Computing," International Journal of Computers & Technology Volume 4 No. 2, March-April, 2013, ISSN 2277-3061 , pp. 737-741, 2013.

[15] O. M. Elzeki , M. Z. Reshad and M. A. Elsoud , "Improved Max-Min Algorithm in Cloud Computing," International Journal of Computer Applications (0975 – 8887), pp. 22-27, 2012.tion Engineering, September 2013.