International Journal of Grid Distribution Computing
Vol.8, No.2 (2015), pp.9-14
http://dx.doi.org/10.14257/ijgdc.2015.8.2.02

Efficient and Enhanced Load Balancing Algorithms in Cloud Computing

Prabhjot Kaur and Dr. Pankaj Deep Kaur

M. Tech, CSE P.H.D
prabhjotbhullar22@gmail.com, pankajdeepkaur@gmail.com

Abstract

A cloud computing is a new paradigm that enables users to access applications, infrastructures and service resources by using thin clients at anytime and anywhere. Through this paradigm users can set up the required resources and they have to pay only for the required resources. Due to rapid growth of the demand for computational power by web, scientific and business applications which led to the creation of large-scale data centers that consuming enormous amounts of electrical power. Thus, in the future providing a mechanism for efficient resource management will be an important objective of Cloud computing. In this we propose a method that allocate resources based on the load of Virtual Machines (VMs). To solving the problem of VM resource scheduling in a cloud computing environment, a load predictor is used to predict the load on each node, according to the this, it allocate the resources.

Keywords: *CloudSim, Load balancer, Virtual Machine, Skewness*

1. Introduction

Cloud computing is emerging as a new technology of large scale distributed computing. The main categories of cloud computing services are software-as-a-service, Infrastructure-as-a-service and platform-as-a-service. Commercial cloud computing providers, such as Google, Amazon, Yahoo and Microsoft deliver cloud computing service to customers all over the world. Cloud computing is platform independent as there is no need to install software's in PC's it provide online applications to users on-demand. Sharing resources, information, software via internet are the main functions of cloud computing with an objective to reduced operational cost and capital, better performance in terms of data processing time and response time, maintain the system stability for future modification in the system. In cloud environment, resources are shared but if they are not properly managed and distributed then it will result into resource wastage. Today the need of resources are increasing drastically day by day. Therefore, it is essential to allocate the resources properly so we moving on to the dynamic resource allocation. For this we using the virtualization technology that migrates virtual machines to physical machines effectively. Virtualization is a technique in which a complete installation of one machine is run on another.

There are various technical challenges that needs to be addressed in cloud computing like Virtual machine migration, fault tolerance, server consolidation, scalability and high availability but central issue is the energy consumption. Green cloud computing is envisioned to achieve the efficient processing and utilization of a computing infrastructure, and also to minimize energy consumption. To the best and the most commonly used techniques for server's level power management mainly include DVS (dynamic voltage scaling) or we called as on/off power. This technique applied to minimize power consumption is concentrating the workload on the minimum of physical nodes and switching idle nodes off. Another major technique we used is called as load balancing, it is the mechanism of distributing the load among various nodes to improve

ISSN: 2005-4262 IJGDC

both resource utilization and job response time while also avoiding a situation where some of the nodes are idle or doing little work while other are heavily loaded. It also ensures that all the processor in the system does approximately the equal amount of work at any instant of time.

Load balancing can dividing the traffic between servers that facilitates networks and resources by providing a maximum throughput without any delay. Load Balancing is done with the help of Data Center Controllers which uses a VmLoadBalancer to determine which VM should be assigned the next request for processing. Based on predetermined parameters, such as current load and availability, the load balancer uses various scheduling algorithms to determine which server should forwards and handle and forwards the request on to the selected server. Load balancing must take into account two major tasks, one is the resource allocation or resource provisioning and other is task scheduling in distributed environment. Resources are allocated in such a manner that all the available resources in cloud do not undergo any kind of wastage and no node in the cloud is overloaded. Task scheduling is done after the allocation of resources to all cloud entities. Task scheduling defines the manner of allocated resources which are available to the end user (*i.e.,* whether the resource is available on sharing basis or fully available). Task scheduling also provides "Multiprogramming Capabilities" to the cloud environment.

CloudSim is most efficient tool for measuring the efficiency and effectiveness of Load Balancing algorithms simulation. CloudSim allows Virtual Machines to be managed by hosts which in turn are managed by datacenters. CloudSim provides architecture with four basic entities these are host, datacenters, virtual machine and application which allows user to set-up a basic cloud computing environment to measuring the effectiveness of Load Balancing algorithms. Datacenters are responsible for providing Infrastructure level Services to the Cloud Users. Hosts are responsible for providing Software level service to the Cloud Users. Hosts have their own memory and storage. Processing capabilities of hosts is measured in MIPS (million instructions per second). Virtual Machine allows deployment as well as development of custom application service models. System software and Application are executed on Virtual Machine on-demand.

2. Existing Load Balancing Algorithms

Load balancing solutions can be divided into hardware-based load balancers and software-based load balancers. Hardware-based load balancers have the ability to handle the high speed network traffic. They are specialized boxes that include Application Specific Integrated Circuits (ASICs) customized for a specific use whereas Software-based load balancers run on standard hardware components and standard operating systems. Distribute workload of multiple network links to achieve minimum response time and maximum throughput and to avoid overloading. We uses different algorithms to distribute the load.

2.1 Round Robin Algorithm (RR): In this datacenter controller uses the concept of time quantum or slices it assigns the requests to a list of VMs on a rotating basis. The resources of the service provider are provided on the basis of the time quantum. The first request is allocated to a VM randomly from the group and then the datacenter controller assigns the subsequent requests in a circular order. Once the VM is assigned the request, the VM is moved in circular motion to the end of the list. In Round Robin Scheduling if time quantum is very large then Round Robin Scheduling Algorithm is same as the FCFS Scheduling and if the time quantum is extremely too small then Round Robin Scheduling is called as Processor so time quantum play an important role in RR algorithm. In this RR algorithm; there is a better allocation concept known as **Weighted Round Robin Allocation** in which we can assign a weight to each of VM so that if one VM is capable of handling twice as much load, then powerful server gets a weight of 2. In this cases, the datacenter controller will assign two requests to the powerful VM for each request assigned to a weaker one.

2.2 Throttled Load Balancer (TLB): In this algorithm the load balancer maintains the record of each state (busy or ideal) in an index table of virtual machines. First the client or server makes a request to data center to find a suitable virtual machine to perform the recommended job. The data center queries the load balancer for allocation of the VM. The load balancer check the index table from top until the first available VM is found, if the VM is found, the load data center communicates the request to the VM identified by the id. Further, if appropriate VM is not found, the load balancer returns -1 to the data center. When the VM completed the allocated task, a request is acknowledged to data center, which is apprised to load balancer to de- allocate the same VM whose id is already communicated. The total execution time is estimated in three phases. In the first phase the formation of the virtual machines and scheduler will be idle waiting to schedule the jobs in the queue ,in second phase once jobs are allocated, the virtual machines in the cloud will start processing, and finally in the third phase the destruction of the virtual machines. The throughput of the model can be estimated as the total number of jobs executed within a required time span without considering the any destruction time. This algorithm will improve the performance by providing the resources on-demand, by reducing the rejection in the number of jobs submitted and resulting in increased number of job executions.

2.3 Active Monitoring Load Balancer (AMLB): The AMLB maintains information about each VMs and also the number of requests currently allocated to which VM. When a request is to allocate a new VM arrives, first it identifies the least loaded VM. If there are more than one loaded VM, then the first identified is selected. Active VmLoadBalancer returns VM id to datacenter controller. The datacenter controller sends the request to the VM identified by that id. This algorithm is quite similar with Weighted Round Robin Algorithm of cloud computing in order to achieve better response time and processing time.

2.4 Adaptive Resource Allocation (ARA): We propose a new load balancing algorithm, called as ARA, adaptive resource allocation in cloud systems, which improve the overall system performance, availability and counteract the effect of burstiness. The main contributions of this algorithm are (1) to present an on/off prediction approach which accurately forecasts changes in user demands by the knowledge of burstiness in workloads; and (2) to develop a smart load balancer, “random” (i.e., randomly select one among all sites) and which on-the-fly shifts between the schemes that are “greedy” (i.e., always select the best site) and based on the predicted information. ARA reduces the response times by optimizing the dispatch of loads across computing sites and adapts a smart site selection for cloud users under both bursty and non-bursty workloads. Amazons EC2 reveal the effectiveness of ARA in a real cloud environment. Under non-bursty conditions the “greedy” methods will always select the best site, obtain better performance than the “random” ones. But under bursty conditions we distributing jobs randomly among all computing sites. ARA algorithm tunes the load balancer by adjusting the trade-off between greediness and randomness in the selection of sites.

2.5 Skewness Algorithm: Skewness is the measure of unevenness resource utilization of a server. By minimizing the *skewness*, we can combine different types of workloads nicely and improve the overall utilization of server resources. We define the resource skewness of a server *p* as

$$skewness(p) = \sqrt{\sum_{i=1}^{n}(\frac{r_i}{\overline{r}} - 1)^2}$$

where *r* is the average utilization of all resources for server *p*. Skewness algorithm consists of three parts:

2.5.1 Hotspot: It is an area in which there is relatively higher temperature than surrounding that means if the utilization of its resources is above a hot threshold. This indicates the overload server. The goal of the proposed algorithm is to eliminate all hot spots if possible or we can keep their temperature as low as possible. We define the temperature of a hot spot p as the square sum of its resource utilization beyond the hot threshold:

$$temperature(p) = \sum_{r \in R}(r - r_t)^2$$

Where, R is the set of overloaded resources in server p and rt is the hot threshold for resource r

2.5.2 Coldspot: It is area in which there is decrease in temperature that means if the utilization of its resources are below threshold it indicates that most of the servers are idle and can turn off to save energy. We can sort the list of cold spots on the ascending order of their memory size. Since the proposed system needs to migrate away all its VMs before we can shut down an under-utilized server.

2.5.3 Green computing: Green cloud computing not only provides a solution to save energy for environment but also reduces operational cost. The challenge is to reduce the number of servers at low load without sacrificing performance. Load skewness algorithm is used when utilization of servers are below green computing threshold.

3. Conclusion

Load Balancing is an essential task in Cloud environment to achieve maximum utilization of resources. In this paper a VM load balancing algorithm was proposed that uses virtualization technology to allocate data center resources dynamically based on application demands and also it supports green computing by optimizing the number of servers in use. We also introduce new adaptive load balancing algorithms for clouds under bursty workloads. We describe the concept of skewness to measure the unevenness in the multi-dimensional resource utilization of a server. By minimizing the skewness, we can combine different types of workloads to improve the overall utilization of server resources. The future work includes to overcome the problem of server overflow and deadlocks. The main goal of proposed system is energy efficiency and overload avoidance.

References

[1] R. Lee and B. Jeng "Load Balancing Tactics In Cloud" International Conference On Cyber Enabled Distributed Computing And Knowledge Discovery, **(2011)**.

[2] N. J. Kansal, "Cloud Load Balancing Techniques: A Step Towards Green Computing", IJCSI International Journal Of Computer Science Issues, ISSN (Online): 1694-0814, vol. 9, Issue 1, no. 1, **(2012)** January, pp. 238-246.

[3] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt and A. Warfield, "Xen and the art of virtualization", in: Proceedings of the 19th ACM Symposium on Operating Systems Principles, SOSP 2003, Bolton Landing, NY, USA, **(2003)**, pp. 177.

[4] E. Caron and L. Rodero-Merino "Auto-Scaling, Load Balancing And Monitoring In Commercial And Open-Source Clouds", Research Report, **(2012)** January.

[5] M. Sharma, P. Sharma and S. Sharma, "Efficient Load Balancing Algorithm in VMCloudEnvironment", IJCST, vol. 3, Issue 1, **(2012)** January-March.

[6] Q. Zhang, L. Cheng and R. Boutaba, "Cloud computing: sate-of-art- and research challenges", Published online, Copyright: The Brazillian Computer Society, **(2011)** April 20.

[7] B. Wickremasinghe, R. N. Calheiros and R. Buyya, "CloudAnalyst: A CloudSim-based Visual Modeller for Analyzing Cloud Computing Environments and Applications", **(2010)** April 20-23, pp. 446-452.

[8] I. Psoroulas, I. Anagnostopoulos, V. Loumos and E. Kayafas, "A Study of the Parameters Concerning Load Balancing Algorithms", IJCSNS International Journal of Computer Science and Network Security, vol. 7, no. 4, **(2007)**, pp. 202-214 .

[9] R. Buyya, R. Ranjan and R. N. Calheiros, "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities", Proceedings of the 7th High Performance Computing and Simulation Conference (HPCS 2009, ISBN: 978-1-4244-4907-1, IEEE Press, New York, USA), Leipzig, Germany, **(2009)** June 21-24.

[10] B. Radojevic and M. Zagar, "Analysis of issues with load balancing algorithm in hosted (cloud) environments", In proceedings of 34th International Convention on MIPRO, IEEE, **(2013)**.

[11] G. Chen, W. He, J. Liu, S. Nath, L. Rigas, L. Xiao and F. Zhao "Energy-Aware Server Provisioning and Load Dispatching for Connection-Intensive Internet Services", Dept. of Computer Science, University of Illinois, Urbana-Champaign, IL 61801.

Author

Prabhjot Kaur, M.Tech