ISSN 2277-3061
Volume 15 Number 9
International Journal of Computer and Technology

AN IMPLEMENTATION OF LOAD BALANCING ALGORITHM IN CLOUD ENVIRONMENT

Sheenam Kamboj [(1)], Mr. Navtej Singh Ghumman [(2)]
[(1)] Research Scholar, Department of Computer Science & Engineering, SBSSTC, Ferozepur, Punjab.
sheenam31.sk@gmail.com
[(2)] Assistant Professor, Department of Computer Science & Engineering, SBSSTC, Ferozepur, Punjab.
navtejghumman@yahoo.com

ABSTRACT

Cloud Computing is an emerging computing paradigm. It aims to share data, calculations, and service transparently over ascalable network of nodes. Since Cloud computing stores the data and disseminated resources in the open environment.So, the amount of data storage increases quickly. As we know that a cloud is the collection of many nodes, which can support various types of application that is used by the clients on a basis of pay per use. Therefore, the system, which is incurring a cost for the user should function smoothly and should have algorithms that can continue the proper system functioning even at peak usage. In this paper, a load balancing algorithm has been discussed and implemented in CloudSim environment. Multiple number of experiments have been conducted to analyze the results.

Keywords

Cloud Computing, Load Balacing, Virtual Machine, Data Center, Data Center Broker, Cluster.

INTRODUCTION

Cloud computing promises to increase the velocity with which applications are deployed, enhance modernization, and lower expenses, all at the same time increasing business agility. Cloud Computing is a concept that has many computers interconnected through a real time network like internet. Cloud computing mainly refers to distributed computing. Cloud computing enables well-situated, on-demand, dynamic and reliable utilization of distributed computing assets. The cloud is altering our life by providing users with new kinds of services. Users acquire service from a cloud without paying attention to the details. Cloud computing is a on demand service in which shared resources work together to perform a task to get the results in minimum possible time by distribution of any dataset among all the connected processing units. Cloud computing is also referred to refer the network based services which give an illusion of providing a real server hardware but in real it is simulated by the software's running on one or more real machines. Such virtual servers do not exist physically so they can be scaled up and down at any point of time [1]. Cloud computing is high utility software having the ability to change the IT software industry and making the software even more attractive [2]. Hence, It helps to accommodate changes in demand and helps any organization in avoiding the capital costs of software and hardware [3] [4].

There are many problems prevalent in cloud computing [6],[7]. Such as:

- ✓ Ensuring appropriate access control (authentication, authorization, as well as auditing)
- ✓ Network level migration, so that it requires least cost and time to shift a job
- ✓ To offer correct security to the data in transit and to the data at rest.
- ✓ Data availability issues in cloud
- ✓ Official quagmire and transitive trust issues
- ✓ Data lineage, data origin and inadvertent leak of sensitive information is possible. And the most prevalent problem in Cloud computing is the problem of Load Balancing.

Necessity of Load Balancing

Load balancing is a computer network method for distributing workloads across multiple computing resources, for example computers, a computer cluster, network links, central processing units or disk drives. Load balancing plans to optimize resource use, maximize throughput, minimize response time, and evade overload of any one of the resources. By the use of multiple components with load balancing instead of a single component may increase reliability through redundancy.

Load balancing in the cloud differs from classical thinking on load-balancing architecture and implementation by using commodity servers to perform the load balancing because it's difficult to predict the number of requests that will be issued to a server. This provides for new opportunities and economies-of-scale, also presenting its own unique set of challenges. Load balancing is one of the central issues in cloud computing [8]. It is a mechanism that distributes the dynamic local workload evenly across all the nodes in the whole cloud to avoid a situation where some nodes are heavily loaded while others are idle or doing little work. It helps to attain a high customer satisfaction and resource utilization ratio, consequently improving the overall performance and resource utility of the system. It also makes sure that every computing resource is distributed efficiently and fairly [9]. It further prevents bottlenecks of the system which may occur due to load imbalance. When one or more components of any service stop working, load balancing facilitates in continuation of the service by implementing fair-over, i.e. in provisioning and de-provisioning of instances of applications without fail. Fig 1 depicts the Load Balancing necessity in cloud when there are requests from multiple clients. The existing load balancing techniques in clouds, consider various parameters such as performance, response time, scalability,

throughput, resource utilization, fault tolerance, migration time and associated overhead. The emerging cloud computing model attempts to address the explosive growth of web-connected devices, and handle massive amounts of data [10] and client demands. Thereby, giving rise to the question whether our cloud model is able to balance the ever-increasing load in an effective way or not.

PROS OF RUNNING SIMULATION

Use of cloud computing is increasing at a very fast pace everywhere because it turns the capital expenditure cost into operational cost. In addition to that, use of simulation tools is considered a better option in spite of being on the real cloud as performing experiments in a controlled and dependent environment is difficult and costly to handle [2].Moreover, effective resource utilization is not possible in case of Cloud. So, we just shift towards cloud simulation tools. Following are the advantages of running simulation tools in cloud:

a. No capital cost involved: As we discussed earlier, that cloud computing makes a shift from capital expenditure cost to operational cost. Having a cloud simulation tool also involves having no installation cost or maintenance cost as well.

b. Leads to better results: Using such tools helps to change inputs and other parameters as well very easily which results in better and efficient output

c. Evaluation of risks at an early stage: Because simulation tools involve no cost while running as is in case of being on cloud, so user can identify and solve any risk that is associated with the design or with any parameter.

d. Easy to learn: While working with such simulation tools, user need to have only programming abilities and rest all depend on that. If the user is well versed with the language, then simulation tools offer no problem [3].

CLOUD SIMULATION TOOLS

There are various simulation tools for cloud ,some of which are as follows:

Cloud Sim: Analysing the performance, policies in real cloud is difficult to achieve because of its altering nature, so in such a situation, we can opt for CloudSim. CloudSim is a famous tool that is actually a toolkit for simulation of cloud scenarios [4].CloudSim has been developed as a CloudBus project in Australia [4]. CloudSim actually enables the users to have a proper insight into cloud scenarios without worrying about the low level implementation details [5]. CloudSim is invented as CloudBus Project at the University of Melbourne, Australia and supports system and behavior modeling of cloud system components such as data centers, virtual machines (VMs) and resource provisioning policies. It implements generic application provisioning techniques that can be extended with ease and limited efforts. CloudSim helps the researchers to focus on specific system design issues without getting concerned about the low level details related to cloud-based infrastructures and services [7]. CloudSim is an open source web application that launches preconfigured machines designed to run common open source robotic tools, robotics simulator Gazebo. SimJava is a toolkit for building working models of complex systems. It is based around a discrete event simulation kernel at the lowest level of CloudSim. It includes facilities for representing simulation objects as animated icons on screen [7,8]. B. CDOSim

CDOSim is a cloud deployment option (CDO) Simulator which can simulate the response times, SLA violations and costs of a CDO. A CDO is a decisions concerning simulator which takes decision about the selection of a cloud provider, specific runtime adaptation strategies, components deployment of virtual machine and its instances configuration. Component deployment to virtual machine instances includes the possibility of forming new components of already existing components. Virtual machine instance's configuration, refer to the instance type of virtual machine instances. CDOSim can simulate cloud deployments of software systems that were reverse engineered to KDM models. CDOSim has ability to represent the user's rather than the provider's perspective. CDOSim is a simulator that allows the integration of fine-grained models. CDOSim is best example for comparing runtime reconfiguration plans or for determining the trade-off between costs and performance [16]. CDOSim is designed to address the major shortcomings of other existing cloud simulators such as

1. Consequently oriented towards the cloud user perspective instead of exposing fine-grained internals of a cloud platform.

2. Mitigates the cloud user's lack of knowledge and control concerning a cloud platform structure.

3. Simulation is independent of concrete programming languages in the case appropriate KDM extractors exist for a particular language. 4. Workload profiles from production monitoring data can be used to replay actual user behavior for simulating CDOs.

MDCSim MDCSim is a variant of CloudSim tools. It helps the user to analyze and predict the hardware related parameters of the data centers like those of servers, switches, routers etc. Also it is used predominantly because of its low overhead produced [4].

SPECI SPECI, Simulation Program for Elastic Cloud Infrastructures, is responsible for analyzing the various scalability and performance aspects of future Data centers [9].It is assumed that when data centers are made to grow big, then they do so in a non linear fashion, so there is a need to analyze the behaviour of such data centers. Here what SPECI plays a role. So, these are all about some major cloud simulation tools being used today.

Network Cloud Network Cloud is an extension of CloudSim and is capable of implementing network layer in CloudSim, reads a BRITE file and generates a topological network. Here, we have topology file which contains the number of nodes along with the various entities involved in simulation [4].In this simulation tool, each entity is to be mapped to a single BRITE node so that network CloudSim can work properly. Network CloudSim can be used to stimulate network traffic in CloudSim.

METHODOLOGY

In cloud computing, the platform, computing and software can be used as services. It is the form of utility computing, in which customer need not own the necessary infrastructure and pay for only what they use. The computing resources are delivered as virtual machines. In such a scenario, task scheduling algorithms play an important role where the aim is to schedule the tasks effectively. It helps to reduce the turnaround time and improve resource utilization. Load balancing in the cloud computing mainly impact on the performance of file system. With the load balancing technique improve the efficiency of the file system. In this thesis mainly work with the better load balance and cloud partition under the different situations. In this system to develop such a file system which can execute N number of jobs on processors which can take less time and work more. Time sharing approach helps to balance the load of number of jobs on processors and also helps to allocate that job the processor can execute according to its capacity which results in getting less weight time for the jobs. After this the time sharing technique execute jobs which are allocated jobs according to job sharing techniques and result in producing less response time then the existing file systems. The space sharing technique also allows splitting the job on different processors if one processor is not able to fulfill the requirements of the job then the job will be split on the different processors which makes job to be executed in less time.

In the work load model all tasks of jobs have equal service demand. Job cumulative service demand is dividing into maximum jobs and each job will have a demand of minimum time. This work load shows the advantage of space sharing policy. I) Job Selection: Job selection policy is used to select the jobs in the queue. The global scheduler consist the jobs in the queue. The aim of scheduling policy is to carry the job from the queue in some manner.

The algorithm adds clustering approach so as to divide VMs with similar capacities into groups. K-means clustering approach has been used to divide VMs into cluster. The load balancer will maintain a list of all the clusters with the minimum and maximum resource specific capacities of each cluster. This is range specifier list. Also the load balancer will maintain the list of VMS for each cluster. The approach is dynamic, centralized and heterogeneous in nature, considers resource specifIc demands. It reduces the overhead of scanning the entire list of VMs from the beginning.

Step1: Initialize all VMs with their specific resource types, capacities of each resource and status of VMs.

Step 2: Cluster the n VMs into k clusters using K-means clustering using the three resource types as parameters i.e. CPU processing speed, Memory and network bandwidth.

Step 3: Cloud controller receives a new request

Step 4: Cloud controller queries appropriate node controller/load balancer for next allocation.

Step 5: Load balancer scans the range specifIer list of k clusters to see that which cluster can handle the incoming request.

Step6: Load balancer will then assign the request to the appropriate VM of the chosen cluster by looking into the list of cluster members which will match the specific demands of the task and whose status is available. In case more than one VMs satisfy this, then the first one which is found will get the task.

Step7: Remaining resource quantities of that VM in the VM list of that cluster is then updated.

Step 8: Status of that VM is changed from A V AILABLE to BUSY.

Step9: When the VM finiish processing the request, the status of that VM is changed to AVAILABLE.

Step 10: The load balancer also updates the capacity of that VM in the VMs capacities list.

Euclidean distance formula has been chosen to assign VMs to the clusters. The value of K i.e. the number of clusters has been chosen to be the highest prime factor of n where n is the number of VMs. The formulat for calculation of Eucledian distance has been mentioned over here:

$$EUD(VM)(C_j)=sqrt[\ (CPU_i- CPU_j)^2+(Mem_i- Mem_j)^2+(BW_i-BW_j)^2$$

To find new mean of each cluster when a machine gets assigned to it is mentioned below.

- $CPU_j=(CPU_i+CPU_j)/2$
- $Mem_j=(Mem_i+Mem_j)/2$
- $BW_j=(BW_i+BW_j)/2$

The Most-fit policy is used to select the cluster. The Most-fit policy is used to reduce resource fragmentation by choosing the appropriate cluster which waste less number of processor and by taking care of the other jobs in the queue. Each cluster which has enough processors for the waiting job, the file system performs a series of simulated activities, to measure how many immediate subsequent allocations can follow the allocation decision. After each cluster has been checked, the file system selects the cluster with the largest number of immediate subsequent allocations to perform current job allocation. If there is no single site having enough free processors Multi-site execution co-allocation will be used, this policy tries to run a parallel job across several sites.

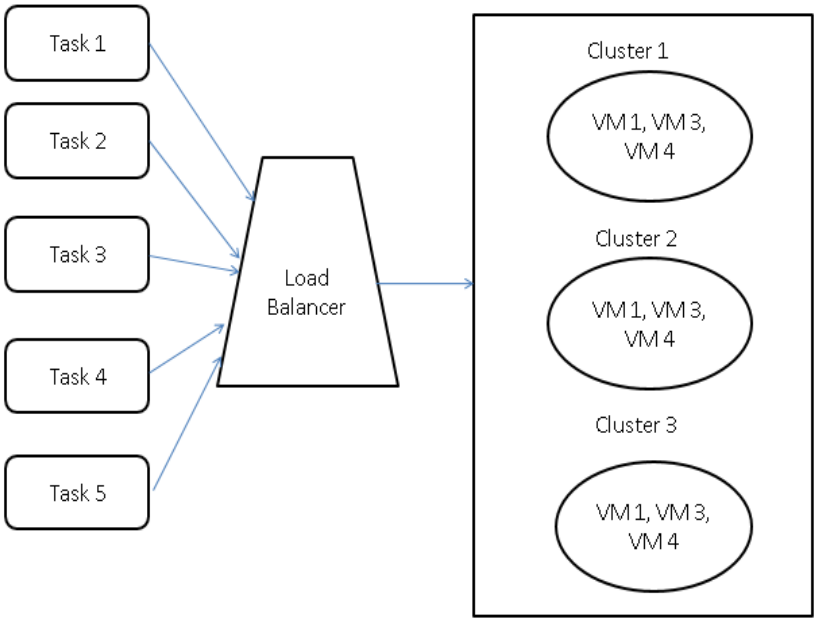

Figure 1. Flow Chart of Load Balancer

EXPERIMENTAL RESULTS

Multiple number of Virtual machines are created with different MIPS, RAM and Bandwidth. We have conducted several number of experiments to identify the number of clusters created. In the below given table 1, we have specified the ID, RAM, MIPS and Bandwidth of the 10 virtual machines inside a single host. In the first experiment, we have created 2 clusters and in the next experiment, we have created 3 clusters. The cluster number of each of the VM is specified in the table.

Table 1. Different Parameters of Virtual Machine

	VM ID	MIPS	RAM	B/W	No. of clusters= 2	No of clusters=3
					Cluster Number	
V1	0	500	512	1000	1	1
V2	1	100	450	1200	0	1
V3	2	150	700	1010	1	1
V4	3	300	1200	1700	1	1
V5	4	750	1624	1200	0	2
V6	5	700	2000	1900	0	2
V7	6	950	1800	2100	0	2
V8	7	600	2400	1800	0	2
V9	8	1200	2600	2200	1	0
V10	9	1700	3000	1600	0	2

The number of iterations for creating the clusters will increase as we increase the number of clusters. In the K-means clustering process, centroid is calculated using the mean of the bandwidth, MIPS and RAM of the virtual machines available in that cluster. The centroid will shift its position in every iteration, therefore the number of iterations will continue till the saturation is achieved. The different number of iterations for creating different number of clusters are specified in the Table 2.

Table 2. Number of Iterations for Different Number of Clusters.

No of vm's	No of cloudlets	No of Clusters	Iterations	Bandwidth Range	MIPS Range	Memory Range
10	6	2	3	1200-2200	600 - 1700	1624- 3000
				1000-1700	100 - 500	450 - 1200
10	6	3	4	1600- 1600	1700 - 1700	3000 - 3000
				1000- 1700	100- 500	450 - 1200
				1200-2200	600-1200	1624-2600
25	6	2	3	1200 - 2500	110 - 870	1000 - 2400
				760 - 1900	70- 600	400 - 1450
25	6	3	6	760 - 1600	70- 500	400- 750
				1900 - 2500	110 - 870	750 - 2200
				1100 -1800	110-660.	1090 -2400
40	6	3	11	1010 - 1120	600 - 1700	612 - 1712
				1130 - 1250	1800 - 3000	1812 - 3012
				1260-1400	3100-4500	3112-4512
40	6	5	15	1010 - 1040	600 - 900	612 - 912
				1050 - 1100	1000 - 1500	1012 - 1512
				1110-1180	1600-2300	1612-2312
				1190-1280	2400-3300	2412-3312
				1290-1400	3400-4500	3412-4512
80	6	5	23	1010 - 1120	600 - 1700	612 - 1712
				1130 - 1260	1800 - 3100	1812 - 3112
				1270-1420	3200-4700	3212-4712
				1430-1600	4800-6500	4812-6512
				1610-1800	6600-8500	6612-8512
150	6	5	32	1010 - 1230	600 - 2800	612 - 2812
				1240 - 1480	2900 - 5300	2912 - 5312
				1490-1750	5400-8000	5412-8012
				1760-2040	8100-10900	8112-10912
				2050-2340	11000-13900	11012-13912
200	6	6	40	1010 - 1180	600 - 2300	612 - 2312
				1190 - 1380	2400 - 4300	2412 - 4312
				1390-1590	4400-6400	4412-6412
				1600-1820	6500-8700	6512-8712
				1830-2070	8800-11200	8812-11212
				2080-2340	11300-13900	11312-13912

CONCLUSION

In this paper, we have implemented the cluster based load balancing technique for cloud computing. The main purpose of load balancing is to satisfy the customer requirement by distributing load dynamically among the nodes and to make maximum resource utilization by reassigning the total load to individual node. This ensures that every resource is distributed efficiently and evenly. After analyzing the results, we have reached upto the solution that the clustering can also be implemented at the client side. We can divide our tasks/cloudlets into different clusters depending upon their task length, cost and priority. It will improve the overall efficiency of the system.

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, "Efficient 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 Efficient 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.

[16] B. Kruekaew and W. Kimpan, "Virtual Machine Scheduling Management on Cloud Computing Using Artificial Bee Colony," Proceedings of the International MultiConference of Engineers and Computer Scientists 2014 Vol I,IMECS 2014, 2014.

[17] R.-S. Chang, J.-S. Chang and P.-S. Lin, "An ant algorithm for balanced job scheduling in grids," Future Generation Computer Systems 25 (2009) 20–27, pp. 21-27, 2009.

[18] Z. Chaczko, V. Mahadevan, S. Aslanzadeh and C. Mcdermid, "Availability and Load Balancing in Cloud Computing," International Conference on Computer and Software Modeling IPCSIT vol.14 (2011) © (2011) IACSIT Press, Singapore, pp. 134-140, 2011.

[19] R. K. S, S. V and V. M, "Enhanced Load Balancing Approach to Avoid Deadlocks in Cloud," Special Issue of International Journal of Computer Applications (0975 – 8887) on Advanced Computing and Communication Technologies for HPC Applications - ACCTHPCA, June 2012, pp. 31-35, 2012.

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