A Load Balancing Analysis of Cloud Base Application with different Service Broker Policies

Pradeep Singh Rawat
Computer Science & Engg. Department
DIT University
Dehradun, India

Priti Dimri
Department of Computer Science and Applications
G.B. Pant Engineering College, Ghurdauri, India

G.P. Saroha
University Computer Centre
M.D. University Rohtak, India

Varun Barthwal
Information Technology Department H N B
Garhwal University
Srinagar, India

ABSTRACT
Resource provisioning and resource optimization are the key issues in cloud computing. To balance the load in across virtual machine load balancing algorithms are classified into two categories i.e. static, dynamic. For homogeneous and stable environment we prefer static load balancing algorithms. For heterogeneous, dynamic environment we prefer dynamic load balancing algorithms. Load balancing may take place in the public, private or hybrid cloud. In this paper, we focus on a load balancing policy i.e. Closest data Center with different no of virtual machines. The evaluation metrics is the response time and data center processing time. Cloud Environment is simulated for the scenario of "Internet banking" of an international bank in simulation toolkit CloudAnalyst. Using these two evaluation metrics we identify that for real deployment of such customers application what should be a threshold value of key parameters which are supported by the Cluster of users across the Globe.

Keywords
MIPS, Cloudlet, Clouds, DVFS, VM, CPU;

1. INTRODUCTION
In order to tolerate the high level of delay in distributed environment, we consider effective load balancing mechanism among the nodes. In some cases it is necessary to replicate the data at the different node. In the case of full replication space complexity increases, while partial replication load balancing complexity increases in a distributed environment. Other parameters for public cloud are taken into consideration. In the case of Public, hybrid cloud accessibility model there is a distributed environment so the speed of network link among the node is also taken into consideration. So my goal is to develop the simplest load balancing algorithm to improve the performance in distributed environment. In the case of static load balancing, we have prior knowledge about the capability of the node which includes storage, computing and network resources. Static algorithms are not adaptive in nature. Round Robin algorithm is useful for scheduling cloud tasks, resources. It is further improved in terms of CLBDM (central load balancing decision model). In simple round robin, algorithm request is sent to the a node having least number of connections. While in the case of central load balancing decision model suggested by **"*Rado device"*** i.e. a threshold value is set for connection time between client and cloud node. If connection time is threshold value then the connection is terminated. **"*Junjie"*** proposed a load balancing algorithm for private cloud having limited accessibility. In this load balancing algorithm, there is a mapping between a virtual and physical machine with the central controller and central resource monitor. Only two attributes are taken into consideration i.e. node capability and network bandwidth. Hence, it is clear that we have to develop the load balancing algorithm which will support the distributed environment[1].

Cloud computing is the provider of dynamic services using virtualized resources over the complex network of node. Today, researchers attempt to build scheduling polices which are applicable in cloud base application. Job scheduling is a most important task in cloud computing environment because the user has to pay for resources used based upon time. Efficient utilization of resources must be important for cloud users. Scheduling plays a vital role to get maximum benefit from the cloud resources. Job scheduling is one of the major activities performed in all the computing environments. A key goal of cloud computing is resource utilization, availability, reliability, resource optimization [2][3]. There are two main categories of the scheduling algorithm, Static scheduling algorithm and Dynamic scheduling algorithm. Both have their own advantage and limitation. The dynamic scheduling algorithm has higher performance than static algorithm but has a lot of overhead compared to it. The main advantage of job scheduling algorithm is to achieve a high-performance computing and the best throughput i.e maximum number of tasks is executed in per unit time. Traditional job scheduling algorithms are not able to provide scheduling in the cloud environments. According to a simple categories, job scheduling algorithms in cloud computing can be categorized into two main class i.e. Heuristic scheduling algorithms (BMHA) which supports the batch mode processing of requests, online mode heuristic algorithms. In BMHA, Jobs are queued and collected into a set when they arrive in the system. The scheduling algorithm will start to schedule the tasks on cloud environment using fixed time interval.

2. RELATED WORK
In distributed environment we need to develop effective load balancing mechanism among the nodes for resource optimization. Some time it is necessary to replicate the data at the different node. In the case of full replication space complexity increases, while partial replication load balancing complexity increases in a distributed environment. So my goal is to develop the simplest load balancing algorithm to improve the performance in distributed environment. In the case of static load balancing, we have prior knowledge about the capability of the node which includes storage, computing and network resources. static algorithms are not adaptive in nature. Round Robin algorithm is useful for scheduling cloud tasks, resources. It is further improved in terms of CLBDM (central load balancing decision model). In simple round robin, algorithm request is sent to the node which is under loaded. While in the case of centralized load balancing load balancing decision model suggested by **"*Rado device"*** i.e. a threshold value is set for connection time between client and server associated with data center. If connection time is

threshold value then the connection is terminated. ***"Junjie"*** proposed a load balancing algorithm for private cloud having limited accessibility. In this load balancing algorithm, there is a mapping between a virtual and physical machine with the central controller and central resource monitor. Only two attributes are taken into consideration i.e. node capability and network bandwidth. Hence, it is clear that we have to develop the load balancing algorithm which will support the distributed environment.

Cloud computing provides dynamic services using shared pool of virtualized resources across the globe with basic Internet facility. Today, researchers attempt to build job scheduling algorithms which are applicable and applicable in Cloud Computing environment Job scheduling is a most important Task in cloud computing environment because user has to pay for resources used based upon time. Efficient utilization of resources must be important and for that scheduling plays an important role in resource optimization. Job scheduling take palce in all computing environments but here we focus on cloud environment. A key goal of cloud computing is resource utilization, availability, reliability, resource optimization.

There are two main categories of the scheduling algorithm. Static, Dynamic scheduling algorithm. Both types of scheduling depend on current state of the system. The dynamic scheduling algorithm provides the better performance than static algorithm. The main advantage of job scheduling algorithm is to achieve a high-performance computing and the best system throughput with optimal resource utilization. Traditional job scheduling algorithms are not able to provide scheduling in the cloud computing environments. In simple way job scheduling algorithms used in cloud computing can be categorized into two main groups, Batch mode (BM) and online mode heuristic algorithms. In batch mode, Jobs are queued and collected into a set after arriving in a system. Examples of BMHA based algorithms are; First Come First Served scheduling algorithm (FCFS), Round Robin scheduling algorithm (RR), ACO(Ant colony optimization)[7][8]. First Come First Serve algorithm is simple and fast. In the round robin scheduling, processes are dispatched in a FIFO manner but are given a time-slice or a quantum. If a process does not complete before its time quantum then it is preempted and processor is allocated to the next process waiting in a queue. The preempted process is then placed at the back of the ready list [11][12]. Load balancing is a methodology to distribute workload across multiple computers, or other resources over the network links [10]. Load balancing achieves optimal resource utilization, maximize throughput. Automatic load balancing services allow clients to scale up and scale down the resources on demands. Load balancing serves two important needs, first to promote the availability of Cloud resources, second to promote performance [14]. Min-Min algorithm: This algorithm chooses small tasks to be executed firstly, which in turn large task delays for a long time. Max – Min algorithm: This algorithm chooses large tasks to be executed firstly, which in turn small task delays for a long time. In most fit task scheduling algorithm task which is best fit in the queue are executed first. This algorithm has high failure ratio i.e. less preferred. In priority scheduling algorithm each process is assigned a priority, and priority is allowed to run. Equal-Priority processes are scheduled in FCFS order. The shortest-Job-First (SJF) algorithm is a special case of the general priority scheduling algorithm. An SJF algorithm is simply a priority algorithm in which longer the CPU burst, the lower the priority and vice versa. Priority can be defined either internally or externally. Internally defined priorities use some measurable quantities or qualities to compute priority of a process.

3. CLOUDANALYST

Cloud computing supports the features of utility base computing. It follows pay as you go model for service level agreement so before deployment of applications on the real cloud we need to test it on a simulation test bed. For modeling and simulation of cloud computing environment, we need a simulation tool. Here we use CloudAnalyst toolkit for modeling simulation of scalable cloud computing environment. To support the infrastructure and application-level requirements on-demand virtualization enabled resources, simulators are required. Few simulators like CloudSim[10] and CloudAnalyst[4][9] are available. CloudAnalyst has been used in this paper as a simulation tool. CloudAnalyst developed on CloudSim is a GUI-based simulation tool. CloudSim facilitates modeling, simulation and other experimentation at user codel level. CloudAnalyst uses the functionalities of CloudSim and it provides the Graphical user interface based simulation. It allows us to setup the parameters for different cloud configuration to study any research problem of the cloud. Based on the parameters the tool compute, the simulation result also shows them in graphical form[9].

4. CASE STUDY

4.1 Simulating The Scenario Of "Internet Banking" Of An International Bank

For experimentation different scenarios are considered with the single data centre. All the requests generated from different userbase in single Data centre located in Region 0-N. America. 25, 50, 75 virtual machines are used for application with different service broker policy and same load balancing policy for application deployment. Single Data centre DC1 uses one physical machine having no of processor=4 with time-shared policy for virtual machine allocation. Main configuration parameters are shown in table 1 in section v [15]. In simulation setup for application advanced Configuration parameters includes user grouping factor= 1000, request grouping factor=100, executable instruction length per request=100 Byte and to balance the load across the virtual machine in a single data centre round robin policy is used.

5. INPUT PARAMETERS FOR SIMULATION

Table 1. Main Configuration

S. No.	User Base	Region	Online user during peak hrs	Online users during off-peak hrs
1	UB1	0- N. America	4,70,000	80,000
2	UB2	1-S. America	6,00000	1,10,000
3	UB3	2-Europe	3,50000	65,000
4	UB4	3-Asia	8,00000	1,25000
5	UB5	4-Africa	1,25000	12,000
6	UB6	5-Oceania	1,50000	30,500

Above table 1 describes the main configuration for cloud configuration. The simulation setup has main configuration with Cluster level details in the perticular Geographic Region. A cluster of a node in the geogrphic region is known as the

UserBase. The parameters used in the main configuration are taken into consideration as a data set for Internet banking application of international bank [15]. Main configuration parameters cover the node associated along with geogrphic area of the entire world. The entire world is divided into 6 continents. Basic features of Cloudsim based cloud simulator i.e. CloudAnalyst is used as a test bed [9] [10].

Table 2. Data Centre Configuration

Name	Region	Arch.	OS	VMM
DC1	0- N. America	X86	Linux	Xen

Above Table 2 shows the data center configuration which is located at Region-0 N. America. Cloud main resource has following characteristics i.e. architecture, operating system used for guest machine and host machine inside server farm of associated data center. To monitor the virtual machine the Xen virtual machine monitor is used for virtualization control.

Table 3. Physical Hardware details at data center DC1

Id	Memory (MB)	Stor-age (MB)	Available Band width	No Of Processor	Processor speed	VM Policy
0	2048	1024000	1000000	4	10000	Time shared

Above table 3 shows the details of each virtual machine at data center. In our research work, we use one physical machine with at data center DC1. Physical machine has 4 processing element with time shared policy to allocate the virtual machine to hosts for task scheduling. Physical hardware is identified by unique id having specific storage, computing and network resources as shown in Table 3.

Table 4. Application deployment configuration

Data centre	No of VM	Image size	Memory	B.W.
DC1	25,50,75	10000	512	1000

Above Table 4 shows the configuration for deployment of the application. It includes virtual machine properties, i.e. Number of virtual machine associated data center etc.

Table 5. Advanced Configuration Parameters

User grouping factor in userbases	Request grouping factor in data center	Executable instruction length per request (Byte)	Load balancing policy across VM in a single data center
1000	100	100	Round Robin

Above Table 5 shows the advance parameters associated with Datacenter, user base and instruction size. It includes the number of simultaneous requests generated from a cluster of a node in different user base. These requests are processed by the data center located in Region-0. VM load balancer plays an important role to schedule the tasks using load balancing policy. A key objective of load balancing is resource optimization and removes the situation of overutilization and underutilization condition.

6. SIMULATION RESULTS

Table 6. Simulation Results for three Cloud Configurations

Cloud configuration	Load Balancing Policy Across VM's in a Single Data Centre	Service Broker policy	Over all response time: Avg. (ms)	Data Center processing Avg. (ms)	Number of VM
1	Round Robin	Closest Data Center	484.25	193.05	25
2		Optimized Response Time	484.43	193.14	50
3		Reconfigure Dynamically	484.28	193.09	75

From Table 1-5 we simply prepare for simulation by setting the parameters at infrastructure, platform and user level. Using prepared simulation setup we got the simulation results shown in Table 6 which shows the best policy used by the service broker along with load balancing scenarios using 4 processing element with MIPS rating 10000. The processing power of physical hardware is allocated to different no of vm i.e. 25, 50, 75 in three scenarios. Simulation results provide the best combination service broker and load balancing policy i.e. closest data center service broker policy with round robin load balancing policy. This scenario provides the minimum average response time and minimum average data center processing time.

7. COMPARISION OF SIMULATION RESULTS IN CLOUD CONFIGURATION

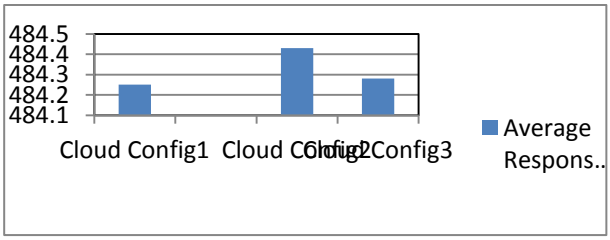

Fig 1: Comparison between Average Response Time and Cloud configuration with (Same load balancing policy round robin with three different service broker policies)

As shown in the above figure 1 it shows the best combination of service broker and load balancing policy with variable number of virtual machine. We can say that for real deployment of such an application these polices pair can provide the optimal solution. Evaluation metrics in figure 1 is the average response time. Cloud configuration 1 provides the best result i.e. closest data center service broker policy is the first choice with same load balancing policies in three scenarios. The second choice will be cloud configuration 3 which is corresponding to the reconfiguration dynamically. We know that broker acts as a mediator between CIS and cloud user so broker policy plays an important role along with load balancing policy at advance configuration level.

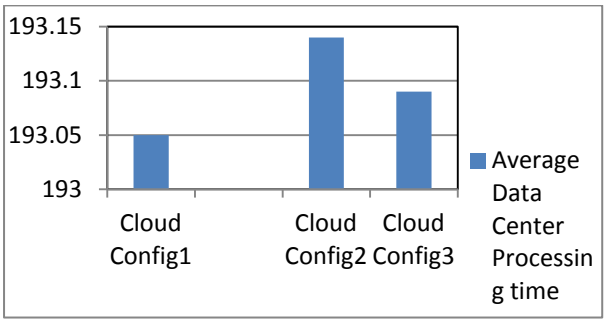

Fig 2: Comparison between Average Data center processing time Time and Cloud configurations with (Same load balancing policy round robin with three different service broker policies)

As shown in the above figure 1 it shows the comparison between average data center processing time to process the request coming from different userbase node in different continent. Cloud configuration1 provides the consistent results for real cloud deployment. Cloud configuration 1 will be our first choice of optimal solution in real deployment. Cloud configuration 1, 2, 3 follows the same load balancing policy with different broker policies. Using simulation results we can assure the good quality of service for service level satisfaction of cloud user and resource optimization with minimizing cost paid for cloud resources.

8. CONCLUSION

In this paper performance of three existing service broker policies is studied with same load balancing policy for three cloud configurations. The results evaluate are based on existing round robin load balancing algorithm, experimental result conclude that if we increase the number of virtual machines in a single data center we found best evaluation metrics i.e. overall average response time and average data center processing time. Service broker policy i.e. closest data center, based on the above results we also conclude that Round Robin VM load balancing algorithm is best among others in cloud configuration 1 with different service broker policy. It provides the best clue about the combination of parameters across main, data center and advanced configuration. There is some future scope of this research idea i.e. for real cloud setup we can utilize the simulation results. Broker acts as a mediator between cloud service provider and end user. Simulation results will be helpful for real deployment of application with resource optimization and cost minimization.

9. REFERENCES

[1] Rajkuma Buyya, James Broberg and Andrzej Goscinski CLOUD COMPUTING Principles and Paradigms, Jhon Wiley & Sons, 2011.

[2] M. D. Dikaiakos, G. Pallis, D. Katsa, P. Mehra, and A. Vakali, "Cloud Computing: Distributed Internet Computing for IT and Scientific Research", in Proc. of IEEE Journal of Internet Computing, Vol. 13, No. 5, pp. 10-13, 2009.

[3] A. Vouk, "Cloud computing- issues, research and implementations", in Proc. of Information Technology Interfaces, pp. 31-40, 2008.

[4] Roderigo N. Calherios, Bhathiya Wickremasinghe "Cloud Analyst: A Cloud-Sim-Based Visual Modeler For Analyzing Cloud Computing Environments And Applications". Proc of IEEE International Conference on Advance Information Networking and Applications, 2010.

[5] Yang Xu, Lei Wu, Liying Guo, Zheng Chen Lai Yang, Zhongzhi Shi, "An Intelligent Load Balancing Algorithm Towards Efficient Cloud Computing", in Proc. of AI for Data Center Management And Cloud Computing: Papers, from the 2011 AAAI Workshop (WS-11-08), pp. 27–32, 2008.

[6] Brototi Mondal,Kousik Dasgupta and Paramartha Dutta, "Load Balancing in Cloud Computing using Stochastic Hill Climbing-A Soft Computing Approach", in Proc. of C3IT-2012, Elsevier, Procedia Technology 4(2012), pp.783-789, 2012.

[7] Ratan Mishra and Anant Jaiswal, "Ant colony Optimization: A Solution of Load balancing in Cloud",in International Journal of Web & Semantic Technology (IJWesT), Vol.3, No.2, pp. 33–50, 2012.

[8] Li Kun, Gaochao Xu, Guangyu Zhao, Yushuang Dong, Dan Wang (2011) " Cloud Task scheduling based on Load Balancing Ant Colony Optimization " Sixth Annual ChinaGrid Conference ,2011,PP 3-9.

[9] B.Wickremasinghe,R.N.CalheirosandR.Buyya,"Coudanalyst:Acloudsimbasedvisualmodellerforanalysingcloudcomputingenvironmentsandapplications", in Proc. of Proceedings of the 24th International Conference on Advanced Information Networking and Applications (AINA 2010), Perth, Australia, pp.446-452, 2010.

[10] R.N.Calheiros, R.Ranjan, A.Beloglazov, C.Rose, R.Buyya,"Cloudsim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", in Software: Practice and Experience(SPE), Vol:41,No:1,ISSN:0038-0644,WileyPress, USA,pp:23-50,2011.

[11] Ko, Soon-Heum; Kim, Nayong; Kim, Joohyun; Thota, Abhinav; Jha, and Shantanu; (2010)"Efficient Runtime Environment for Coupled Multi-physics Simulations: Dynamic Resource Allocation and Load-Balancing" 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid), 17-20 May 2010, pp.349-358.

[12] Hiranwal Saroj, Dr. K.C. Roy, "Adaptive Round Robin Scheduling Using Shortest Burst Approach Based On Smart Time Slice" International Journal Of Computer

Science And Communication July-December 2011, Vol. 2, No. 2, Pp. 319-323.

[13] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A.Konwinski, G. Lee, D. Patterson, A.Rabkin, I. Stoica, M. Zaharia (2009). Above the Clouds: A Berkeley View of Cloud computing.TechnicalReport No. UCB/EECS-2009-28, the University of California at Berkley, USA, Feb. 10, 2009.

[14] Brain Underdahl, Margaret Lewis and Tim meeting "Cloud computing clusters for dummies" Wiley Publication (2010), [Book].

[15] Kousik Dasgupta, Brototi Mandala, ParamarthaDuttac, Jyotsna Kumar Mondal, "International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMA) 2013 A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing", Procedia Technology 10 (2013) 340 – 347.