Comparison of P_BRR Scheduling Algorithm with Round Robin and Heuristic Priority Scheduling Algorithm in Virtual Cloud Environment

Er. Navreet Singh[1]

[1]Asst. Professor, Computer Science Department
ASSM College, Mukandpur
Punjab, India

Abstract

While considering the virtual cloud environment and various terms associated with it resource provisioning and load balancing are the main areas where research work has been going on at vast pace. Here we will discuss the various things related to the cloud environment. The main objective of our work is to compare the load balancing and resource provisioning algorithm (i.e. Batch Mode Heuristic Priority with Round Robin Scheduling) with other existing algorithms so that we can check and conclude that which algorithm is more efficient in virtual cloud environment. We will evaluate the performance of various resource provisioning algorithms one by one based on the certain parameters and after that compare their performance along with our algorithm.

Keywords: *Cloud computing,resource provisioning, load balancing, scheduling, virtualization.*

1. Introduction

1.1 Cloud Computing: What it is – and what it isn't

First of all let us discuss about traditional desktop computing that was a clear scenario of PC-centric i.e. in which you run copies of software programs on every computer that you do have. The files and documents that you create, edit are stored on the computer on which they are developed.
But in case of cloud computing the softwares and applications we use are not run particularly from your PC but instead of that they are stored on various servers accessed via the interconnecting network. Same is the scenario for the creating and editing of the documents. Cloud computing is a style of computing, in which dynamically scalable (and mostly virtualized) resources are provided as a service over the Internet.

1.2 What Cloud Computing Isn't:

Cloud Computing is not related to the inter-network computing i.e. where the applications/files are hosted on a single organization's server and then can be accessed over the company's network[12]. Cloud Computing is expanded more than that. It exhibits multiple companies, their servers and their multiple networks.

The actual Cloud Computing definition is the cloud itself. Before considering what actually cloud computing is, we should be clear about the definition of Cloud [12]. Cloud is a large group of inter-connected computers and they can be either public or private. For example, Google has its own clouds for public and private purposes. The private cloud is that which is owned by Google itself and the public cloud is overall publicly accessible by its users.

1.3 Infrastructure as a Service (IaaS)

Cloud computing is the result of various research areas such as grid computing and virtualization technologies and service-oriented architecture. Cloud computing offers a major service that is IaaS which is based on the two models i.e. on-demand computing model and pay-as-you-use model which are provided to the end users for their convenience. But there is a need a better provisioning of the cloud infrastructure in data centers to avail this service for the end users. To get the most out of the IaaS model we have to concentrate on two major services. These are resource (VM) provisioning and load balancing & scheduling methods.

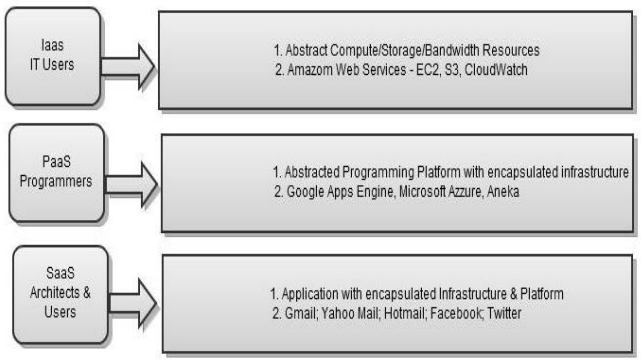

Fig. 1 Difference between various service classes.

To optimize the performance of the cloud architecture and infrastructure various resource provisioning and load balancing mechanisms should be followed in a well manner. Therefore, in cloud computing load balancing is mainly required to distribute the dynamic load evenly and equally

across all the different nodes. Our main issue is to look for the proper load balancing aids that will be required in minimizing the resource consumption, starvation & over resource provisioning etc. The random arrival of the load in such a cloud computing environment can cause some nodes to be heavily loaded. We have to maintain the load across the various nodes in such a manner so that the overall performance of the servers and the network will be up to the mark and maximum at the peak level [1], [3]. Equally load distributing improves the performance of the overall network by transferring the load from a heavily loaded server.

2. Overview: Load Balancing, Scheduling And Resource Provisioning

2.1 Load Balancing And Scheduling

Load Balancing basically ensures that all the processors in the system or every node in the network does approximately the equal amount of work at any instant of time. The load can be CPU load, memory capacity, delay or network load. Hence Load Balancing is the procedure 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 [3]. Now these challenges are also faced in the storage cloud side where in we have to balance the load across the various nodes or at the server sides.

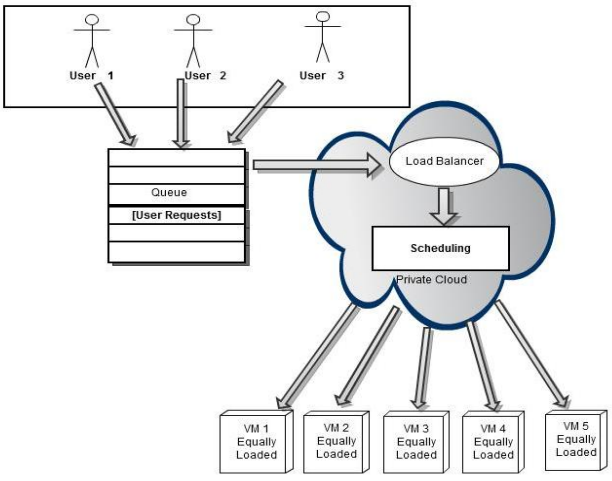

Fig. 2 Load Balancing and Scheduling in Cloud Computing Environment

Equally load distributing improves the performance of the overall network by transferring the load from a heavily loaded server. Efficient scheduling and resource allocation is a critical characteristic of the cloud computing based on which the performance of the system is estimated. Load Balancing is one of the main prerequisites to utilize full resources of parallel and distributed systems.

2.2 Load Balancing Parameters in Clouds

The factors that always be considered in various load balancing techniques in cloud computing are as follows [3]. Detailed description of the load balancing factor is as follows:

- *Response Time*- It is the amount of time taken to provide the response by some load balancing algorithm in a distributed environment. This parameter should be minimized. It is represented as R (t).
  Formula to calculate the Response Time is:
  $$R(t) = Finish\ Time - Start\ Time.$$
  $$= T(f) - T(s) \quad (1)$$
  Where T(f) is finish time and T(s) is start time.
- *Communication Time*- It is defined as time taken by number of hops to travel in the communication channel. It is represented by C (t). Formula to calculate the Communication Time is:
  $$C(t) = 2(Number\ of\ hops * Time\ to\ traverse\ between\ hops) \quad (2)$$
- *Processing Time*- It is defined as the difference between Communication Time and Response Time. It is represented by P (t). Formula to calculate the Processing Time is:
  $$P(t) = Response\ Time - Communication\ Time$$
  $$= R(t) - C(t) \quad (3)$$
- *Throughput*- is used to calculate the number of tasks whose execution has been completed. It should be high to improve the reliability and the performance of the system. It is represented as Th (V_i).
  $$Th(V_i) = (Cloudlet\ length * Number\ of\ cloudlets) / Response\ Time$$
  $$= [Length(Ci) - N_i] / R(t) \quad (4)$$
  Where Length (Ci) is cloudlet length and N_i is number of cloudlets for specific virtual machine.
- *Network Delay*---Delay in sending request and receiving response. It is the time taken by the network to send the number of cloudlets to particular VM and time taken by the VM to receive the cloudlets.
  $$D(t) = No.\ of\ cloudlets / Rate\ of\ transmission$$
  $$= N/r \quad (5)$$
  Where "r" is the rate of transmission.

2.3 VM Provisioning

Resource provisioning mechanism is able to converge to an optimal or near-optimal CPU or VM allocation within a reasonable amount of time [17]. Secondly, it is capable of adapting even more extreme cases of resource over or under utilization. Finally, the algorithm is able to adapt to dynamically varying data rates and converges at new resource allocations in a short frame [1],[21]. Virtual machine allocation should be adjusted at runtime based on the variation in workloads to achieve the goal of application **QoS** metrics. The work model of Cloud computing environment is shown below in the diagram:

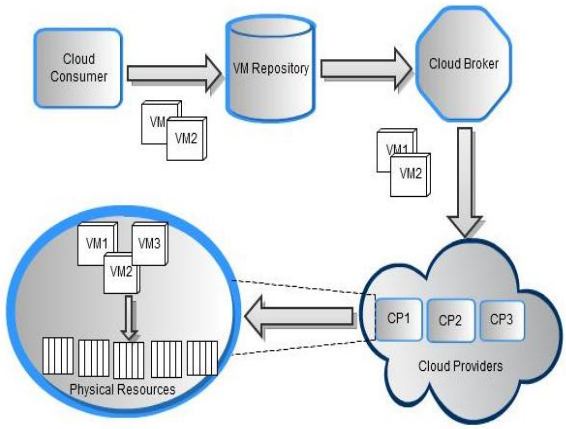

Fig. 3 Work Model of Cloud Computing Environment

Resource Provisioning in the storage clouds often requires an estimate of the capacity needs of the Virtual Machines (VM's) [17]. The estimated VM size is the basis for allocating resources according to the required demands. With proper resource provisioning unused resources of a low utilized VM can be borrowed by the other co-located VM's with high utilization [4]. Hence in cloud computing, a resource provisioning mechanism is required to supply cloud consumers a set of computing resources, for processing the jobs and storing the data. Here we will compare the Round Robin and Priority Scheduling algorithms with our algorithm on the basis of various factors.

2.4 Task Formulation

To optimize and measure the performance of the cloud architecture various load balancing and scheduling mechanisms have been developed till now. Overloaded nodes across the server and storage side often lead to performance degradation and are more vulnerable to various failures. To remove this limitation the load must be migrated from the overloaded resource to an underutilized one without causing harm and disruption to the application workload so that the load across the various nodes can be equally divided and properly managed across them.

In order to optimize the performance the problem is how to efficiently perform the load balancing mechanisms on the storage cloud so that the resources are better provisioned according to the user's requests. We have to use some sort of scheduling algorithm which in turn handle or divide the workload between the various nodes so that no machine will be overloaded and all the other nodes or machines are not assigned any task and are totally free. Now the problem is to identify which scheduling methodology is best suitable for the load balancing and resource provisioning purposes. From the list of various load balancing and scheduling algorithms we have to choose the best and relevant one which can be adjusted properly inside the virtual cloud environment. In this paper we will compare some of the recent developed scheduling algorithms with our algorithm i.e. Batch Mode Heuristic Priority Round Robin (P_BRR) Scheduling Algorithm.

3. P_BRR Algorithm.

3.1 Algorithm

Suppose that A= {J1, J2…, Jm} is a set of jobs that request resources in a cloud environment. Also let us assume that B= {R1, R2…, Rn} is a set of resources available in cloud environment. Each job requests a resource for its processing in a distributed environment. The jobs are allocated to the available resources in a round-robin fashion and also based on load balancing factor.

- Batch-Mode Heuristic Priority scheduling algorithm schedule the jobs statically among the various servers according to calculated priority factors such as no. of jobs, execution time to execute all running jobs.
- In our algorithm we have taken load balancing factor as our priority factor. We have done calculation for load balancing factor of each server and corresponding to that load factor our jobs will be executed among the various servers and resources will be allocated accordingly [3]. Formula to calculate the Load Balancing Factor:

$$L.B = [Throughput * Bandwidth] / [Response\ Time * Network\ Delay]$$
$$= [T(h) * B(w)] / [R(t) * N(d)] \quad (6)$$

- Load Balancing factor for each of the server updates after executing each task by corresponding server which is holding the priority for executing respective task for a single instant of time.
- We can check the status of each of the server and can change the status of the particular server from on to off if it is not ready to execute the jobs or if it is not willing to participate in the load balancing and resource provisioning.
- Four parameters are maintained in a list by each of existing virtual machines or servers. Request Count and Active State of each of the server is maintained along with other information.
- The Request Count is incremented one by one whenever a single job is executed by that particular server.
- The active state of the server can be either true or false depending upon the state of the participation of the corresponding server i.e. whether it is willing to participate or not.
- If it is not willing to participate the active state of the machine will be false and vice-versa. Jobs will be allocated to these servers according to round robin basis.

The main goal of our algorithm is to allot the resources (VM's) to the set of jobs according to some priority and in a round robin fashion such that workload across the storage cloud should be properly managed.

4. Comparison of PBRR Scheduling Algorithm with other Scheduling Algorithms on the basis of various Load Balancing Parameters.

To solve the problem of under-utilization of resources and to balance the load across the various virtual machines and for better resource provisioning we have implemented a load balancing and scheduling algorithm which is a combination of two algorithms. These are:

- Round Robin Scheduling Algorithm.
- Batch Mode Heuristic Priority Algorithm.

These above 2 algorithms will help us to transfer the load from one sever to another in round-robin fashion and then will schedule the jobs between them so that all servers would participate in order to process the jobs and provide the better resources to execute those tasks. Here we will compare our scheduling algorithm with the Round Robin and Batch Mode Heuristic Priority Scheduling Algorithm on the basis of various load balancing parameters one by one:

4.1 Comparison on the basis of Response Time:

Our P_BRR algorithm provides minimum response time than the other two algorithms because in our algorithm the priority is decided purely on the basis of load balancing factor for each VM and we do not turn off the nodes very frequently. Response Time is calculated as:

$$R(t) = Finish\ Time - Start\ Time = T(f) - T(s) \quad (7)$$

Table 1 Comparison of Response Time

Virtual Machines	Response Time(ms)		
	P_BRR	RR	BMHP
V1	1.24	2.47	3.91
V2	1.63	3.18	3.09
V3	2.09	3.84	4.13
V4	3.12	4.90	5.61
V5	1.93	3.74	4.54
V6	4.23	5.81	6.31

As shown in fig. 5 the response time of our scheduling algorithm is relatively lesser than that of the Round Robin scheduling algorithm and Batch Mode Heuristic Priority Scheduling Algorithm as the number of virtual machines increases.

4.2 Comparison on the basis of Communication Time:

It is defined as time taken by number of hops to travel in the communication channel. It is represented by C (t). Formula to calculate the Communication Time is:

C (t) =2(Number of hops*Time to traverse between hops) (8)

P_BRR Scheduling Algorithm takes less communication time between the two communicative nodes as compared to the Round Robin and BMHP scheduling algorithm. Communication time is calculated with the help of two factors i.e. time required to traverse between the two hops and number of hops that are participating in the process. Hence is more efficient and suitable than the Round Robin and BMHP scheduling algorithms.

Table 2 Comparison of Communication Time

Virtual Machines	Communication Time(ms)		
	P_BRR	RR	BMHP
V1	0.67	0.99	1.41
V2	0.98	1.42	1.89
V3	1.16	2.04	2.53
V4	1.53	2.37	3.01
V5	0.82	1.98	2.34
V6	2.24	3.21	3.58

4.3 Comparison on the basis of Processing Time:

It is defined as the difference between Communication Time and Response Time. It is represented by P (t) and Response time and communication time are represented by R(t) and C(t) respectively. Formula to calculate the Processing Time is:

$$P(t) = Response\ Time - Communication\ Time = R(t) - C(t) \quad (9)$$

Table 3 Comparison of Processing Time

Virtual Machines	Processing Time(ms)		
	P_BRR	RR	BMHP
V1	0.57	1.48	2.50
V2	0.65	1.76	1.20
V3	0.93	1.80	1.60
V4	1.59	2.53	2.60
V5	1.11	1.76	2.20
V6	1.99	2.60	2.73

P_BRR Scheduling Algorithm takes less processing time as compared to the Round Robin (RR) and BMHP scheduling algorithm. So from the above table it is clear that the P_BRR Scheduling Algorithm is most relevant than the Round Robin and BMHP scheduling algorithms.

4.4 Comparison on the basis of Throughput:

It is used to calculate the number of tasks whose execution has been completed. It should be high to improve the reliability and the performance of the system. It is represented as Th (V_i).

Th (V_i) = (Cloudlet length*Number of cloudlets)/ Response Time

$$= [Length(Ci) - N_i] / R(t) \quad (10)$$

Where Length C_i is cloudlet length and N_i is number of cloudlets for specific virtual machine.

Let us consider the following simulation result of which we have to calculate the throughput of all the 3 algorithms and have to compare them. After comparing the results of these two algorithms we have to conclude that which is more efficient out of these three scheduling algorithms. The process of simulation is done within Cloud environment i.e. implemented with the help of CloudSim toolkit. So, the result

of the experimentation and simulation is shown below in the table:

Table 4 Show the result of our simulation on output screen

Simulation Result

Cloudlet ID	Status	Data center ID	VM ID	Cloudlet Length	Start Time	Finish Time
1	success	2	2	9457	0	56

Here Cloudlet length=9457, No. of cloudlets=1, Starting time=0 and Finish Time= 56. Now make a table corresponding to these values and the value of response time for each VM and for each algorithm.

Table 5 Comparison of Throughput

Virtual Machines	Throughput(n tasks executed)		
	P_BRR	RR	BMHP
V1	7626	3828	2418
V2	5801	2973	3060
V3	4525	2463	2290
V4	3031	1930	1686
V5	4900	2529	2083
V6	2236	1628	1499

5. Conclusion And Future Scope

In this paper, we have compared the results of the Round Robin Scheduling Algorithm and Batch mode heuristic priority scheduling with our scheduling algorithm i.e. P_BRR load balancing and scheduling algorithm on the basis of various load balancing parameters. These parameters are response time, communication time, processing time, throughput and network delay. In the previous paper we have discussed about the load balancing and scheduling mechanism in detail and also designed an algorithm i.e. P_BRR which has efficiently overcome the various load balancing problems like under-utilization and over-utilization of resources and then compared it with two priority scheduling algorithms. In this realistic study we have worked on the future work of the previous study that is we have compared the P_BRR scheduling mechanism with more two scheduling algorithms. In our future implementation, we will use this P_BRR algorithm by considering *load balancing factor* as a priority factor, the response time and throughput increases and power consumption varies based on CPU utilization. Load balancing factor is a parameter which shows that how many request a single VM can handle.

References

[1] Navreet Singh, "Resource Provisioning using Batch Mode Heuristic Priority with Round Robin Scheduling [Final Implementation]," IASTER, 2014, 25-34.

[2] Borja Sotomayor, R. S, "An Open Source solution for Virtual Infrastructure Management in Private and Public Clouds," IEEE, 2009, 1-11.

[3] Navreet Singh, Gaurav Raj, "Resource Provisioning using Batch Mode Heuristic Priority with Round Robin Scheduling," IJET, 2013, 2959-2966.

[4] Jeanna Matthews, T. G, "Virtual Machine Contracts for Datacenter and Cloud Computing Environments," ACM, 2009, 25-30.

[5] Jennings., R, Cloud Computing with the Windows Azure Platform, Indianapolis: WILEY (2009).

[6] John W. Rittinghouse, J. F, Cloud Computing: Implementation, Management and Security, London, New York: CRC Press (2010).

[7] Marcos Dias de Assuncao, A. d, "Evaluating the cost benefit of using cloud computing to extend the capacity of clusters," ACM, 2009, 141-150.

[8] Meenakshi Sharma, P. S, "Efficient Load Balancing Algorithm in VM Cloud Environment," IJCSE, 2012, ISSN: 439-441.

[9] Miller., M, Cloud Computing: Web-Based Applications That Change The Way You Work and Collaborate Online, Indianapolis: QUe (2009).

[10] Mohammed Alhamad, T. D, "Response Time for Cloud Computing Providers," ACM, 2010, 603-606.

[11] Nidhi Jain Kansal, I. C, "Existing Load Balancing Techniques in Cloud Computing: A Systematic Review," ISSN, 87-91 (2012).

[12] Rajkumar Buyya, J. B, Cloud Computing: Principles and Paradigm, New Jersey: WILEY (2011).

[13] Rodrigo N. Calheiros, R. R, "CloudSim: A toolkit for modeling and simulation of cloud computing environment and evaluation of resource provisioning algorithms," *D0I*, 2010, 24-50.

[14] Shamsollah Ghanbari, M. O, "A Priority based Job Scheduling Algorithm in Cloud Computing," Proceedia Engineering, 2012, 778-785.

[15] Smita Vijayakumar, Q. Z, "Dynamic Resource Provisioning for Data Streaming Applications in a Cloud Environment," IEEE, 2010, 441-448.

[16] Soumya Ray, A. D, "Execution Analysis of Load Balancing Algorithms in Cloud Computing Environment," IJCCSA, 2012, 1-13.

[17] Subramanian S, N. K, "An Adaptive Algorithm For Dynamic Priority Based Virtual Machine Scheduling In Cloud," IJCSI, 2012, 397-401.

[18] Yue Tan, Y. L, "Provisioning for Large Scale Cloud Computing Services," ACM, 2012, 407-408.

[19] Jasmin James, D. B, "Efficient VM Load Balancing Algorithm for Cloud Computing Environment," IJCSE, 2012, 1658-1663.

AUTHOR'S PROFILE

Er. Navreet Singh

Completed Bachelor of Technology in Computer Science and Engineering from LIT, Phagwara in the year 2011. Completed Masters of Technology in Computer Science and Engineering from LPU, Phagwara in the year 2013. Currently posted as an Assistant Professor in ASSM College, Mukandpur, one of the well known regional campus of GNDU, Amritsar . Area of the research work is Cloud Computing. Published 2 international journals before. First was published in the IJET in year 2013 and second journal was published in the IASTER in year 2014.