ijecs open access www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 8 August, 2014 Page No. 7926-7931

Extended Round Robin Load Balancing in Cloud Computing

Priyanka Gautam[1], Rajiv Bansal[2]

[1]CSE Department. JMIT, Radaur
Priyankagautam111@jmit.ac.in

[2] CSE Department. JMIT, Radaur
rajivbansal@jmit.ac.in

Abstract: Cloud computing is a new platform for the developers where they can store and use their contents but as this is a new area and it is expected that sooner it will become very popular among all developers and its users. So, to avoid clusters and waiting in queues a load balancing is required. Most of the researchers are already working on it and we have also tried to work on same area with the use of round robin scheduling technique for load balancing.

Keywords: Cloud Computing, Load Balancing, Extended Round Robin Scheduling, Clouds.

1. Introduction

Cloud Computing is a very new area for researchers and developers for utilization of resources even from the remote locations with high speeds. So, cloud computing refers to accessing shared resources, software and other resources [1]. There is no standard definition for cloud computing. But different researchers have given different definitions like: According to National Institute of Standards and Technology (NIST), It is a computing environment which consists of a bunch of distributed servers known as masters, providing demanded services and resources to different clients known as clients in a network with scalability and reliability of datacentre [1].

Alternatively, cloud computing is a model that enables on demand access to a shared pool of configurable computing resources [7]. Cloud computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services [3]. Cloud computing can also be defined as a style of computing where software applications are provided to consumer as “service” rather than a product using the internet [2]. Cloud Computing can be considered as a model for Developing, maintaining and accessing applications by paying for the resources which are only used for certain time [8].

2. What We Need It?

Cloud computing is currently used many commodity nodes that can cooperate to perform a specific service together. In addition, the internet applications are continuously enhanced with multimedia, and vigorous development of the device quickly occurs in the network system [58]. The new concept of cloud computing allows for more applications for internet users. In a cloud computing environment, users can access the operational capability faster with internet application [58], and the computer systems have the high stability to handle the service requests from many users in the environment. As the cloud is made up of data centers which are very much powerful to handle large numbers of users applications ranging from those that run for a few seconds to those that run for longer periods of time on shared hardware platforms. Hence the resource allocation policies and scheduling algorithms in cloud environments for various applications and service models is essential. However a technique has to be designed for distributing the user application workload among various data centre so as to minimize the response time, minimize the cost, minimize the resource utilization, and minimize the overhead. The data centers are targeted according to the availability and best response time with respect to location of data centre in the same region or different region [8].

3. Advantages of Cloud Computing

- It is a model that enables on demand access to a shared pool of configurable computing resources.
- It delivers infrastructure, platform, and software that are made available as subscription-based services in a pay-as-you-go model to the consumers. The services provided bu cloud computing is in the form of Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).
- To serve huge number of requests from different users located at different parts of the globe on a pay-per-usage model the process of virtualization has been followed in cloud environment.
- With the advent of the Cloud, deployment and hosting became cheaper and easier with the use of pay-per- use flexible elastic infrastructure services offered by Cloud providers.

4. Challenges

Although there are many advantages for cloud computing but there are some challenges associated with cloud computing. Some of the challenges are like Virtual machine migration, server consolidation, fault tolerance, high availability and scalability. One of the major challenges for the cloud computing environment is to adjust/balance the load on the cloud computing environment. **Load balancing** is a 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 [9]. Load Balancing is done with the help of load balancers where each incoming request is redirected and is transparent to client who makes the request. The main issue associated with load balancing is because of traffic is increasing due to highly demand of services, so incoming network traffic is distributed on network level by using network load balancing algorithms (like: random allocation, round-robin allocation, etc.). These algorithms use network based parameters of incoming traffic to decide where to forward traffic, without any information from other components of computer system, like current load of application or database servers.

5. Need for Load Balancing

Load balancing in cloud computing provides an efficient solution to various issues residing in cloud computing environment set-up and usage. Load balancing must take into account two major tasks, one is the resource provisioning or resource allocation and other is task scheduling in distributed environment [11]. Load balancing is one of the critical components for efficient operations in the cloud computing environment. In recent years many clients from all over the world are demanding the various services at rapid rate. Many algorithms have been designed to carry out the client's request towards the cloud nodes. Accordingly the cloud computing platform will dynamically configure its servers and these servers may be present physically or virtually in the computing environment. Hence, selecting the virtual machines or servers has to be scheduled properly by using an appropriate load balancing approach. In the present work, a local optimized load balancing approach is proposed for distributing of incoming jobs uniformly among the servers or virtual machines. Further, the performance is analyzed using Cloud Analyst simulator and compared with existing Round Robin and Throttled algorithms. Simulation results have demonstrated that the proposed algorithm has distributed the load uniformly among virtual machines [10].

6. Existing Load Balancing Algorithms

Although Cloud Computing is a new field but a lot of work has already been done by many researchers in the field of cloud computing. They have their own advantages and disadvantages as well. The algorithms are based on round robin, min-min, max-max, randomized algorithms and many more. Some of them are also listed below.

Token Routing: The main objective of the algorithm [12] is to minimize the system cost by moving the tokens around the system. But in a scalable cloud system agents cannot have the enough information of distributing the work load due to communication bottleneck. So the workload distribution among the agents is not fixed. The drawback of the token routing algorithm can be removed with the help of heuristic approach of token based load balancing. This algorithm provides the fast and efficient routing decision. In this algorithm agent does not need to have an idea of the complete knowledge of their global state and neighbor's working load. To make their decision where to pass the token they actually build their own knowledge base. This knowledge base is actually derived from the previously received tokens. So in this approach no communication overhead is generated.

Equally Spread Current Execution Load: This algorithm requires a load balancer which monitors the jobs which are asked for execution. The task of load balancer is to queue up the jobs and hand over them to different virtual machines [10]. The balancer looks over the queue frequently for new jobs and then allots them to the list of free virtual server. The balance also maintains the list of task allotted to virtual servers, which helps them to identify that which virtual machines are free and need to be allotted with new jobs. The experimental work for this algorithm is performed using the cloud analyst simulation. The name suggests about this algorithm that it work on equally spreading the execution load on different virtual machine.

Randomized: Randomized algorithm is of type static in nature. In this algorithm [12] a process can be handled by a particular node n with a probability p. The process allocation order is maintained for each processor independent of allocation from remote processor. This algorithm works well in case of processes are of equal loaded. However, problem arises when loads are of different computational complexities. Randomized algorithm does not maintain deterministic approach. It works well when Round Robin algorithm generates overhead for process queue.

It is concluded from the above study that cloud computing is very emerging concept in the IT world and many researches have been proposed by various authors for the betterment of this area. There are different scheduling policies which can be further extended and can be taken to the next level of advancement like round robin scheduling method can be further extended to accept different input values as compared to the fixed size input values which is the main objective of this implementation also.

7. Proposed Work

The problem of resource selection in distributed environment has received lots of attention in last few years. In many previous works, resource selection refers to the selection of computational resource in grid environment. In [4], the authors presented a resource selection model using decision theory for selecting the best machine to run a task. In [5], they proposed an algorithm for resource selection problem of computational grids, based on the resource availability prediction using frequent workload patterns. Recently, with the rapid development of data intensive computing, many researchers turned their attention to resource selection of data-intensive environment, such as data grid [6].

The main objective behind this exercise is to balance the load on different data centers according to the tasks/cloudlets received and to allocate the appropriate data center or virtual machine to handle new cloudlets. This will result in better resource allocation and minimize overall propagation time of input and output data. Previously most of work has been done in this area. There are a lot of scheduling algorithms available for load balancing like Round Robin scheduling algorithm, Min-Min scheduling, Max-Max scheduling, FCFS Scheduling, Randomized scheduling and many more. All of them have some of their advantages and disadvantages as well.

My proposed work is basically an extension of round robin scheduling and randomized scheduling algorithm. With round robin scheduling algorithm there is a limitation of same mips or mb for each and every cloudlet and it takes each task in round robin order. The whole simulation is done on cloudsim simulator. This concept is further extended to support the cloudlets with different mips and mbs with the added functionality of random cloudlet/task selection. But our cloud environment can be considered as a set of Kcenters D = {d1, d2... dk}, which are located in different place and connected by links of different bandwidths. For an application composed of a set of N independent jobs J = {j1, j2… jN} (N >>K), each job j is subset of J, requires a set of K datasets, denoted by Fj, which are accessed on a subset of D. Consider a task j has been submitted to a VM, which is created 03 data center d, for execution. For each dataset, the time needed to transfer it from d_f to d is denoted by $T_{t(}f,d_f,d)$.The estimated data transfer time for the V_m, $T_t(j)$, is the maximum value of all the times for transferring all the datasets required but the $V_{m.}$

Work flow

Service Proximity Policy- The broker selects the data center according to round robin scheduling and allocates the cloudlets with different mips and mbs to the selected data center. Allocation of cloudlets required following steps.

Begin Main

1. Start the CloudSim Simulator.
2. Initialize data centers with their configurations i.e. their processing speeds, RAM, bandwidth etc.
3. Register each data center with CIS and in turn that will report to the main broker responsible for handling cloudlets.
4. Then broker will receive the cloudlet requests.
5. Based on the datacenters chosen in round robin order broker allocated the received cloudlets to different data centers.
6. Repeat step 4 nd 5 until there are no more cloudlets.

End Main

8. Implementation

All the implementation related work is done with the help of cloudsim simulator and code is implemented in NetBeans using advanced concepts of JAVA. A more viable alternative is the use of simulation tools. These tools open up the possibility of evaluating the hypothesis in a controlled environment where one can easily reproduce results. Simulation-based approaches offer significant benefits to IT companies. And for modeling Cloud Computing environment the simulator that is used is CloudSim: a new, generalized, and extensible simulation framework that allows seamless modeling, simulation, and experimentation of emerging Cloud computing infrastructures and application services. By using CloudSim, researchers and industry-based developers can test the performance of a newly developed application service in a controlled and easy to set-up environment. Based on evaluation results reported by CloudSim, they can further fine tune the service performance.CloudSim offers the following novel features: (i) support for modeling and simulation of large scale Cloud computing environments, including data centers, on a single physical computing node; (ii) a self-contained platform for modeling Clouds, service brokers, provisioning, and allocations policies; (iii) support for simulation of network connections among the simulated system elements; and (iv) facility for simulation of federated Cloud environment that inter-networks resources from both private and public domains.

9. Results and Discussions

The test method in this evolution contains two data centers with five virtual machines each and any number of tasks/cloudlets(we have considered 10/15) and brokers. Data center contain number of hosts with their configuration (Processing Speed, RAM, Bandwidth etc.) initialized with CIS. In our experiments, we randomly generate the bandwidth and processing speeds of data centers. Then submit the jobs to the data centers. Let us have some random cloudlets with their required parameters as instruction volume in MI (Millions of instructions) and Data Volume in MB.

Cloudlets	Instruction Volume(MI)	Data Volume (MB)
Task 1	116236	330
Task 2	71351	364
Task 3	45396	311
Task 4	88141	386
Task 5	133876	312
Task 6	91386	384
Task 7	83779	338
Task 8	43121	371
Task 9	105290	396
Task 10	88998	318
Task 11	111167	382
Task 12	65731	301
Task 13	65731	303
Task 14	96856	305
Task 15	117649	345

Table 8.1: Cloudlet Parameter Details

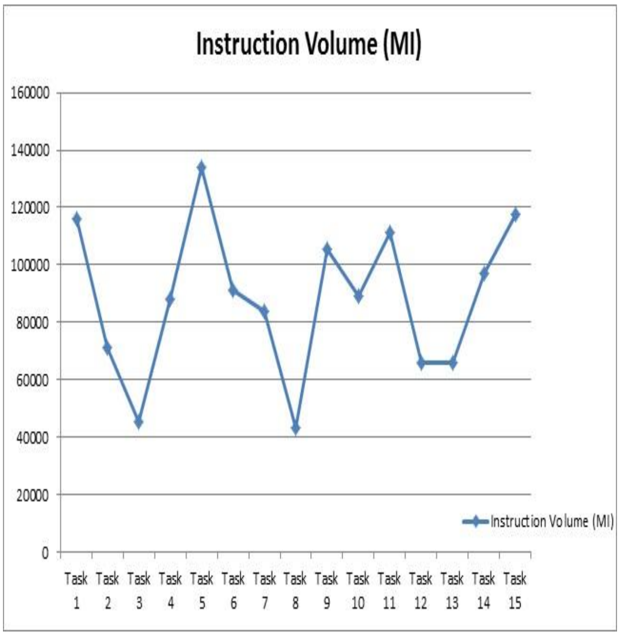

Fig. 8.1 Graphical View of cloudlet parameter (Instruction Volume)

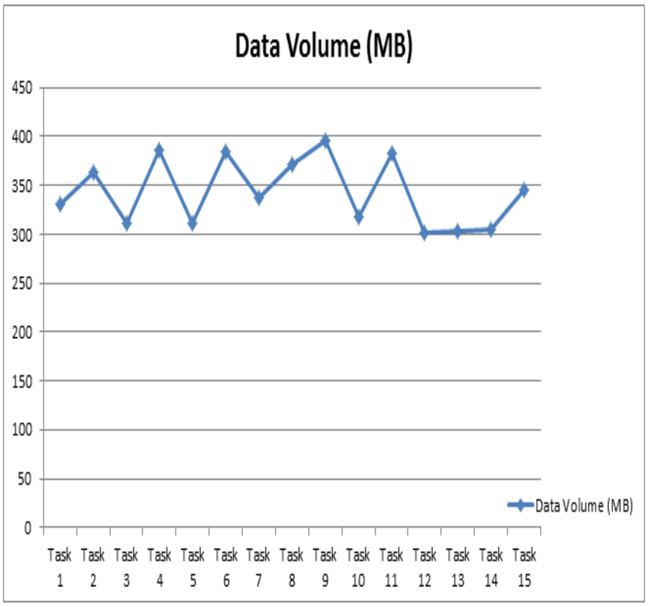

Fig. 8.2 Graphical View of cloudlet Parameter (Data Volume)

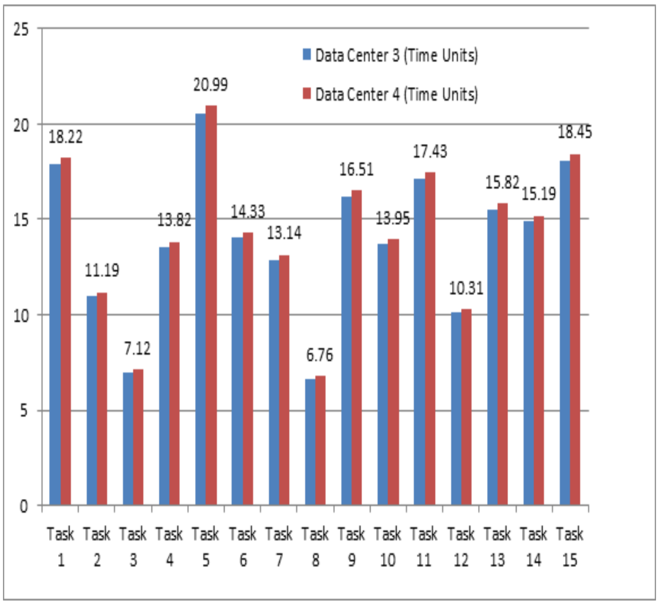

Fig 8.3 Graphical View of time required by different data centers

Let us have two data centers with their resource id 3 & 4 and along with the following parameters.

Resource Id	Processing speed(MIPS)	Bandwidth(MBPS)
3	6504	19357
4	6378	19632

Table 8.2: Data Center Details

Estimated time required by each data center according to mips and mbs they can handle. The following table shows the values which are maximum out of mips and mbs because the overall time required depends upon maximum of mips and mbs.

Task	Data Center 3 (Time Units)	Data Center 4 (Time Units)
Task 0	17.87	18.22
Task 1	10.97	11.19
Task 2	6.98	7.12
Task 3	13.55	13.82
Task 4	20.58	20.99
Task 5	14.05	14.33
Task 6	12.88	13.14
Task 7	6.63	6.76
Task 8	16.19	16.51
Task 9	13.68	13.95
Task 10	17.09	17.43
Task 11	10.11	10.31
Task 12	15.51	15.82
Task 13	14.89	15.19
Task 14	18.09	18.45

Table 8.3: Maximum Time Required By Data Center 3 & 4

Task Id	Start Time	Time	Finish Time
7	517.44	0.1	517.54
2	535.64	0.1	535.74
12	757.6	0.1	757.7
11	787.77	0.1	787.87
1	833.73	0.1	833.83
6	883.44	0.1	883.54
3	1057.69	0.1	1057.79
9	1067.97	0.1	1068.07
5	1096.63	0.1	1096.73
13	1127.4	0.1	1127.5
8	1161.14	0.1	1161.24
10	1254.87	0.1	1254.97
0	1275.15	0.1	1275.25
14	1297.18	0.1	1297.28
4	1362.09	0.1	1362.19

Table 8.4: Starting and Finishing Times of Cloudlets

Random distribution of cloudlets at data center 3 and its virtual machines with their starting time and we have assumed that all the cloudlets are required with 0.1 units of time. And in the meanwhile datacenter is available and so only data center is occupied but if data center3 is busy then data center 4 will be chosen according to round robin technique.

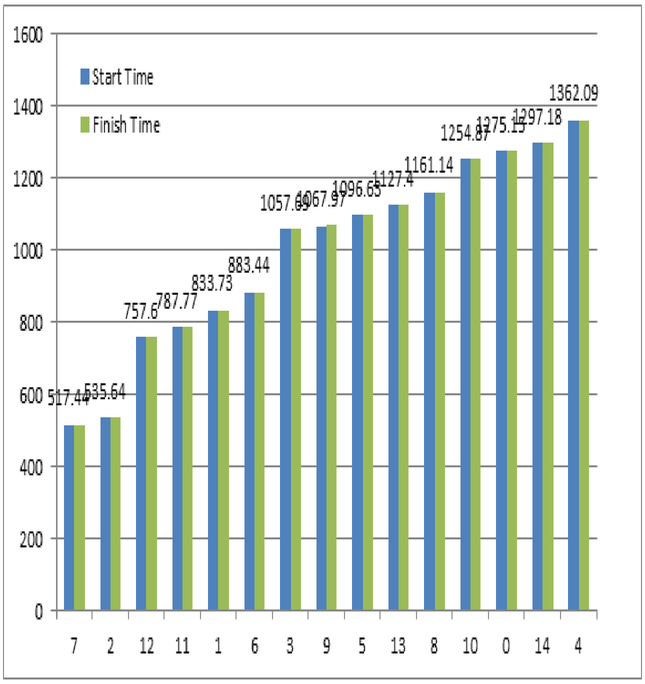

Fig 8.4 Graphical View of Starting & Finishing Times

Similar process can be repeated for any number of cloudlets and any number of datacenters. So basically it is an extension of round robin method of load balancing in cloud computing which is further mapped with randomized allocation of cloudlets. And it gives more efficient results.

9. Conclusion

In this paper a novel technique for job submission in Cloud environment is purposed. The purposed technique consider both cloudlets processing time and file transfer time while selecting appropriate hosts for cloudlet(job) submission on distributed resource with an objective to minimize execution time and cost.

In this paper, I have extended normal round robin scheduling method which have a primary condition for cloudlets to be of same processing time (MI) and processing size (MB) and our purposed method random submissions with respect to the make span and turnaround time of execution. The purposed technique out performs other techniques for all parameters.

10. Future Scope

Although cloud computing is a very emerging technology and many researchers are also working in this field to improve the quality parameters associated with cloud computing. Most of the work has already been done in this area and still it is under development. It is a new field to for researchers and scholars to work in this area. I have worked on round robin scheduling algorithm and with the extension over that it can also facilitate cloudlets with different processing times and different sizes. So in this area further work can be done. This algorithm can be further extended to select the host which has minimum distance. It can also check about access cost of that resources, so implementation of algorithm which has optimum access cost.

References

[1] Prathmesh Arnikar, Siddharth Sonawane, Ankita Fale, Sagar Aghav, Shikha Pachouly "Load Balancing In Cloud Computing" in Journal Of Information, Knowledge And Research In Computer Engineering.

[2] Meenakshi Sharma, Anitha Y and Pankaj Sharma "An Optimistic Approach for Load Balancing in Cloud Computing" in International Journal of Computer Science and Engineering, Volume-2, Issue-3 March 2014.

[3] Sumit Khurana and Khyati Marwah "Performance Evaluation Of Virtual Machine (Vm) Scheduling Policies In Cloud Computing (Spaceshared & Timeshared)" in IEEE July 2013.

[4] Enda Barrett, EndaHowley, Jim Duggan "A Learning Architecture for Scheduling Workflow Applications in the Cloud" 2011 Ninth IEEE European Conference on Web Services978-0-7695-4536-3/11 $26.00 © 2011 IEEE.

[5] BoonyarithSaovapakhiran, George Michailidis†, Michael Devetsikiotis "Aggregated-DAG Scheduling for Job Flow Maximization in Heterogeneous Cloud Computing" 2011 IEEE 978-1-4244-9268-8/11.

[6] Hsu Mon Kyi, Thinn Thu Naing "An efficient approach for virtual machines scheduling on a private cloud environment" 2011 IEEE 978-1-61284-159-5/11.

[7]A nist national definition of cloud computing, http://www.csre.nist.gov/groups/SNS/cloudcomputing.

[58] Shu-Ching Wang, Kuo-Qin Yan, Wen-Pin Liao and Shun-Sheng Wang "Towards a Load Balancing in a Three-level Cloud Computing Network" in IEEE 2010.

[8] Rekha P M and M Dakshayini "Service Broker Routing Polices In Cloud Environment: A Survey" in International Journal of Advances in Engineering & Technology, Jan. 2014.

[9] Tejinder Sharma, Vijay Kumar Banga "Efficient and Enhanced Algorithm in Cloud Computing" in International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-3, Issue-1, March 2013.

[10] Shridhar G.Domanal and G.Ram Mohana Reddy "Load Balancing in Cloud Computing Using Modified Throttled Algorithm" in National Institute of Technology Karnataka.

[11] Mayanka Katyal, Atul Mishra "A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment" in International Journal of Distributed and Cloud Computing Volume 1 Issue 2 December 2013.

[12] Soumya Ray, Ajanta De Sarkar "Execution Analysis Of Load Balancing Algorithms In Cloud Computing Environment" in International Journal on Cloud Computing: Services and Architecture (IJCCSA),Vol.2, No.5, October 2012.