International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307, Volume-3, Issue-1, March 2013

Efficient and Enhanced Algorithm in Cloud Computing

Tejinder Sharma, Vijay Kumar Banga

Abstract*— — A class of systems and applications that procure distributed resources to execute the function in the decentralized manner is referred as cloud computing. It enables a wide range of users to access scalable, virtualized hardware, distributed and/or software infrastructure over the Internet. One of the challenging scheduling problems in Cloud datacenters is to take the allocation and migration of reconfigurable virtual machines into consideration as well as the integrated features of hosting physical machines. In order to select the virtual nodes for executing the task, Load balancing is a methodology to distribute workload across multiple computers, or other resources over the network links to achieve optimal resource utilization, minimum data processing time, minimum average response time, and avoid overload. The objective of this paper to propose efficient and enhanced scheduling algorithm that can maintain the load balancing and provides better improved strategies through efficient job scheduling and modified resource allocation techniques. Load balancing ensures that all the processors in the system as well as in the network does approximately the equal amount of work at any instant of time. The results discussed in this paper, based on existing Equally Spread Current Execution, Round Robin, Throttled and a new proposed enhanced and efficient scheduling algorithms.

***Index Terms*—- Cloud Computing , Cloud Analyst, Equal Spread Current Execution, Round Robin , Throttled, VM.**

I. INTRODUCTION

In 1969, Leonard Kleinrock , one of the chief scientists of the original Advanced Research Projects Agency Network (ARPANET) which seeded the Internet, said: "As of now, computer networks are still in their infancy, but as they grow up and become sophisticated, I will probably see the spread of "computer utilities" which, like present electric and telephone utilities, will service individual homes and offices across the country[14][15]." Cloud computing is emerging as a new paradigm of large scale distributed computing. It has moved computing and data away from desktop and portable PCs, into large data centers[11]. It has the capability to harness the power of Internet and wide area network (WAN) to use resources that are available remotely, thereby providing cost effective solution to most of the real life requirements[3][10].

Today, there are more than a hundred million computing devices connected to the Internet and many of them are using cloud computing services daily. According to the IDC's anticipation , the SaaS (Software As A Service) market reached $13.1 billion in revenue at 2009 will grow to $40.5 billion by 2014 at a compound annual growth rate(CAGR) of 25.3%[4][9][17]. These networked devices submit their requests to a service provider and receive the results back in a timely manner without the involvement of the service complexity related to information storage and process, interoperating protocols, service composition, communications and distributed computation, which are all relied on the network and the backend servers to offer desirable performance. Thus, applications associated with network integration have gradually attracted considerable attention. In a cloud computing environment, users can access the operational capability faster with internet application [13], and the computer systems have the high stability to handle the service requests from many users in the environment. However, the internet infrastructure is continuous grow that many application services can be provided in the Internet.

Cloud Computing thus involving distributed technologies to satisfy a variety of applications and user needs. Sharing resources, software, information via internet are the main functions of cloud computing with an objective to reduced capital and operational cost, better performance in terms of response time and data processing time, maintain the system stability and to accommodate future modification in the system .So there are various technical challenges that needs to be addressed like Virtual machine migration, server consolidation, fault tolerance, high availability and scalability but central issue is the load balancing , it is the 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[1][2].

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. Based on predetermined parameters, such as availability or current load, the load balancer uses various scheduling algorithm to determine which server should handle and forwards the request on to the selected server. To make the final determination, the load balancer retrieves information about the candidate server's health and current workload in order to verify its ability to respond to that request. Load balancing solutions can be divided into software-based load balancers and hardware-based load balancers. Hardware-based load balancers are specialized boxes that include Application Specific Integrated Circuits (ASICs) customized for a specific use. They have the ability to handle the high speed network traffic where as Software-based load balancers run on standard operating systems and standard hardware components.

Manuscript received , on March, 2013.

Tejinder Sharma,Amritsar College Of Engineering And technology Amritsar,India.

Dr. Vijay Kumar Banga Amritsar College Of Engineering And technology, Amritsar,India.

II. PROBLEM FORMULATION

The primary purpose of the cloud system is that its client can utilize the resources to have economic benefits. A resource allocation management process is required to avoid underutilization or overutilization of the resources which may affect the services of the cloud. Some of the jobs may be rejected due to the overcrowding for the virtual machines by the current jobs in the cloud system. Resource allocation and Efficient scheduling is a precarious characteristic of cloud computing based on which the performance of the system is evaluated. The examined characteristics have an impact on cost optimization, which can be obtained by improved overall response time and data processing time with the help of enhanced and efficient algorithm.

Hence we have proposed an algorithm in which live migration of load is done in virtual machine to avoid the underutilization and hence improving the and data transfer cost

III. EXISTING LOAD BALANCING ALGORITHMS FOR CLOUD COMPUTING

Distribute workload of multiple network links to achieve maximum throughput, minimize response time and to avoid overloading. We use three algorithms to distribute the load. And check the performance time and cost

A. Round Robin Algorithm (RR):

It is the simplest algorithm that uses the concept of time quantum or slices Here the time is divided into multiple slices and each node is given a particular time quantum or time interval and in this quantum the node will perform its operations. The resources of the service provider are provided to the client on the basis of this time quantum. In Round Robin Scheduling the time quantum play a very important role for scheduling, because if time quantum is very large then Round Robin Scheduling Algorithm is same as the FCFS Scheduling. If the time quantum is extremely too small then Round Robin Scheduling is called as Processor

Sharing Algorithm and number of context switches is very high. It selects the load on random basis and leads to the situation where some nodes are heavily loaded and some are lightly loaded. Though the algorithm is very simple but there is an additional load on the scheduler to decide the size of quantum[5] and it has longer average waiting time, higher context switches, higher turnaround time and low throughput.

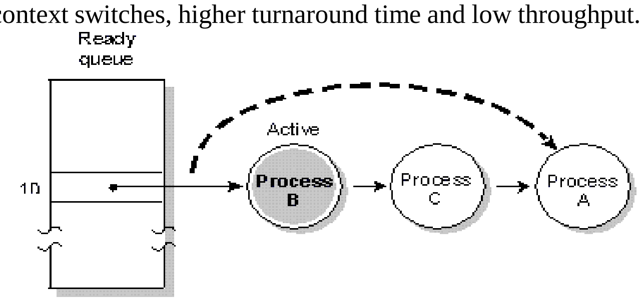


Figure 1. Round Robin Algorithm

B. Equally Spread Current Execution Algorithm (ESCE):

In spread spectrum technique load balancer makes effort to preserve equal load to all the virtual machines connected with the data centre. Load balancer maintains an index table of Virtual machines as well as number of requests currently assigned to the Virtual Machine (VM). If the request comes from the data centre to allocate the new VM, it scans the index table for least loaded VM. In case there are more than one VM is found than first identified VM is selected for handling the request of the client/node, the load balancer also returns the VM id to the data centre controller. The data centre communicates the request to the VM identified by that id. The data centre revises the index table by increasing the allocation count of identified VM. When VM completes the assigned task, a request is communicated to data centre which is further notified by the load balancer. The load balancer again revises the index table by decreasing the allocation count for identified VM by one but there is an additional computation overhead to scan the queue again and again

C. Throttled Load Balancing Algorithm (TLB):

In this algorithm the load balancer maintains an index table of virtual machines as well as their states (Available or Busy). The client/server first makes a request to data centre to find a suitable virtual machine (VM) to perform the recommended job. The data centre queries the load balancer for allocation of the VM. The load balancer scans the index table from top until the first available VM is found or the index table is scanned fully. If the VM is found, the load data centre. The data centre communicates the request to the VM identified by the id. Further, the data centre acknowledges the load balancer of the new allocation and the data centre revises the index table accordingly. While processing the request of client, if appropriate VM is not found, the load balancer returns -1 to the data centre. The data centre queues the request with it. When the VM completes the allocated task, a request is acknowledged to data centre, which is further apprised to load balancer to de- allocate the same VM whose id is already communicated.

The total execution time is estimated in three phases. In the first phase the formation of the virtual machines and they will be idle waiting for the scheduler to schedule the jobs in the queue, once jobs are allocated, the virtual machines in the cloud will start processing, which is the second phase, and finally in the third phase the cleanup or the destruction of the virtual machines. The throughput of the computing model can be estimated as the total number of jobs executed within a time span without considering the virtual machine formation time and destruction time The proposed algorithm will improve the performance by providing the resources on demand, resulting in increased number of job executions and thus reducing the rejection in the number of jobs submitted.

IV. PROPOSED WORK

A. Efficient and Enhanced Algorithm (EEA):

Algorithm:

Step1. Initially VM index table will be 0 as all the VMs are in available state.

Step2. DataCenterController receives a new request.

Step3. DataCenterController queries new LoadBalancer for next allocation

Step4. DataCenterController parses the VM list to get next available VM:

If found:

LoadBalancer returns the VM id to DataCenterController

Step2 continues

If not found:

Using round robin fashion VM index is reinitialised to 0 and in increment manner VMs are checked to find VM in available state

Step5. When the VM finishes the processing the request, and the DataCenterCOntroller recieves the cloulet response, it notices the load balancer of the VM de-allocation

Step6. The Load Balancer updates the status of VM in allocation table to available.

Step7. Continue from Stp2

The purpose of the algorithm is to find the expected Response Time of each Virtual Machine , which is calculated as:

Response Time = Fint - Arrt + TDelay (1)

Where,Arrt is the arrival time of user request and Fint is the finish time of user request and the transmission delay can be determined using the following formulas

TDelay = Tlatency + Ttransfer….(2)

Where, TDelay is the transmission delay T latency is the networklatency and T transfer is the time taken to transfer the size of data of a single request (D) from source location to destination.

Ttransfer = D / Bwperuser (3)

Bwperuser = Bwtotal / Nr (4)

Where, Bwtotal is the total available bandwidth and Nr is the number of user requests currently in transmission. The Internet Characteristics also keeps track of the number of user requests in between two regions for the value of Nr.

V. SIMULATION AND RESULT ANALYSIS

The Simulation and Result Analysis will be done by using the cloud analyst tool

A. Cloud Analyst [6][7]:

The Cloud Analyst is a GUI based tool which is developed on CloudSim architecture. CloudSim[11][16] is a toolkit used for model ling, experimentation and simulation. The deployment of large scale applications is quite economical and easy by using clouds. The cloud also generates the new issues for developers. The various users access the internet applications around the world, and because popularity of applications may vary along the world, so experience in the use of application can also vary.

In order to analyze various load balancing policies configuration of the various components of the cloud analyst tool need to be set. We have set the parameters for the user base configuration, application deployment configuration, and data center configuration as shown in figure 6, figure 7 and figure 8 respectively. As shown in figure the location of user bases has been defined in six different regions of the world. We have taken three data centers to handle the request of these users. One data center is located in is located in region 0 , second in region1and third in region 2. On DC1, DC2 and DC3 number of VM 2 are 50. In order to analyze various load balancing policies configuration of the various component of the cloud analyst tool need to be done. We have set the parameters for the user base configuration, application deployment configuration, and data center configuration as shown in figure 3, figure 4 and figure 5 respectively. As shown in figure the location of user bases has been defined in six different regions of the world. We have taken three data center to handle the request of these users. One data center is located in is located in region 0, second in 1, third in 2. On DC1, DC2 and DC3 number of VM allocated are 50. The duration of simulation is 60hrs.

Cloud analyst enables the modeler to execute the simulation repeatedly with the modifications to the parameters quickly and easily. The graphical output of the simulation results can be analyzed more easily and efficiently.

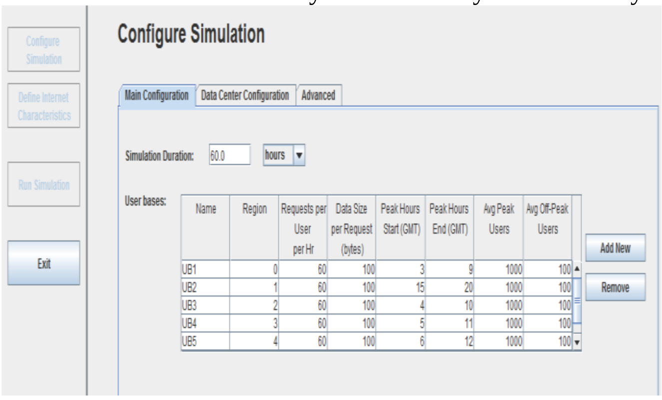


Figure 2. Configure Screen in Simulator

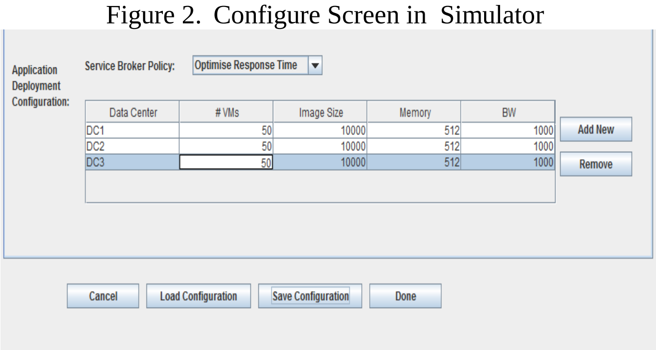


Figure 3. BrokerPolicy Configuration

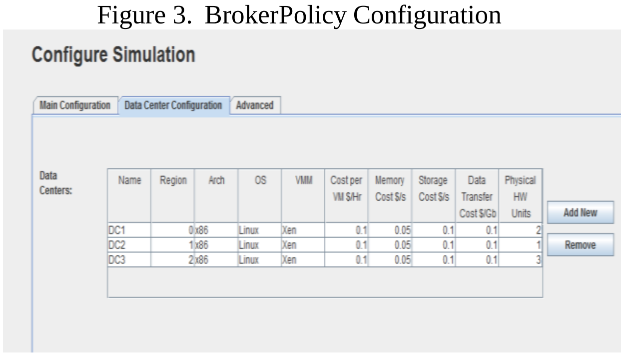


Figure 4. Datacentre Configuration

After performing the simulation the result computed by cloud analyst is as shown in the following figures. We have used the above defined configuration for each load balancing policy one by one and depending on that the result calculated for the metrics like response time, request processing time and cost in fulfilling the request has been shown in Figures 5,6,7,8.

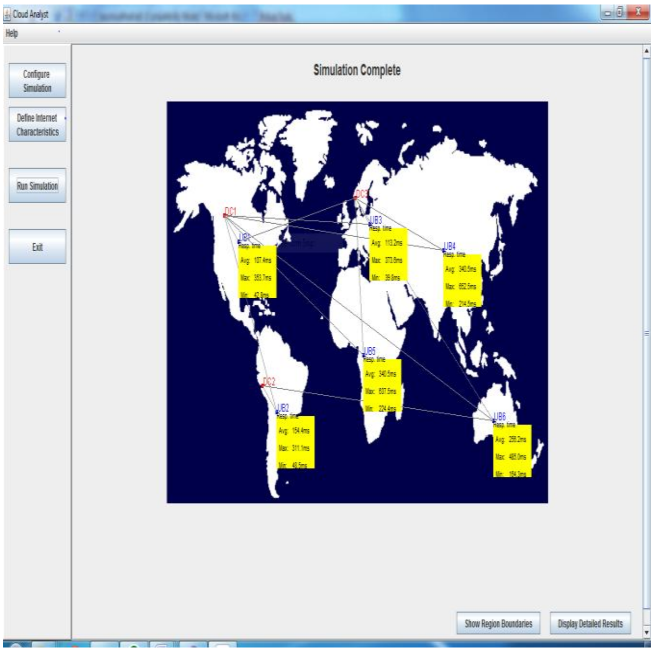


Figure 5. Output Screen of Cloud Analyst.

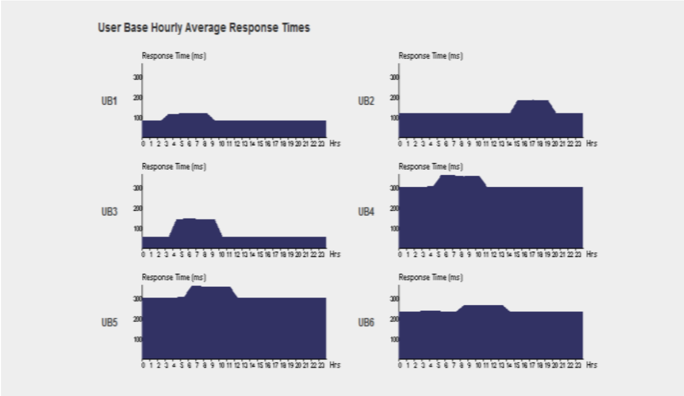


Figure 6. UserHourlyResponseTimes

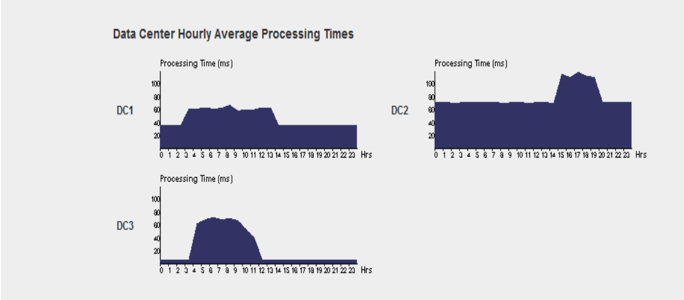


Figure 7. DataProcessingTimes

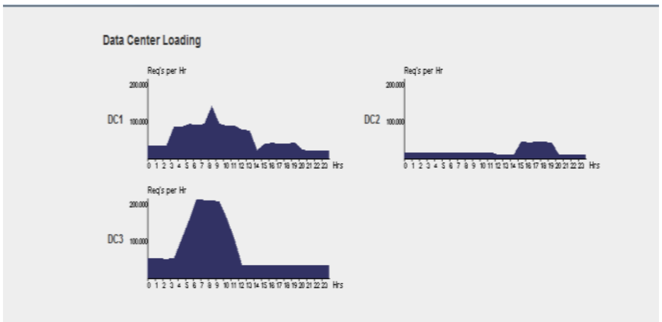
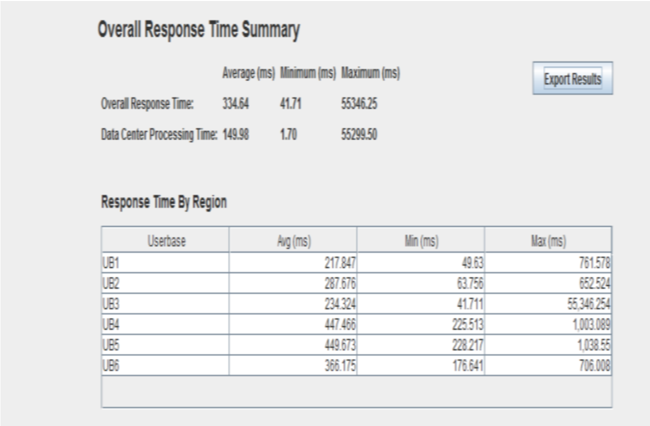


Figure 8. DataCentreLoading Times.

B. Response Time:

Response time for each user base and overall response time calculated by the cloud analyst for each loading policy has been shown in the figure 9, 10 ,11 and 12 respectively.



Overall Response Time Summary

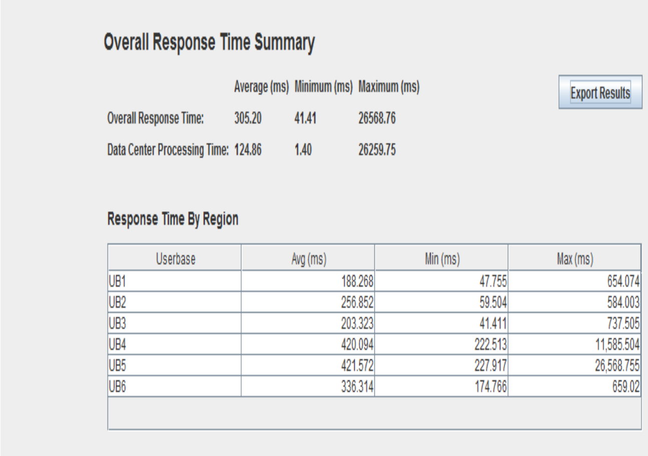
	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	334.64	41.71	55346.25
Data Center Processing Time:	149.98	1.70	55299.50

Export Results

Response Time By Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	217.847	49.63	761.578
UB2	287.676	63.756	652.524
UB3	234.324	41.711	55,346.254
UB4	447.466	225.513	1,003.089
UB5	449.673	228.217	1,038.55
UB6	366.175	176.641	706.008

Figure 9. ResponseTime For RR with 6 Userbases



Overall Response Time Summary

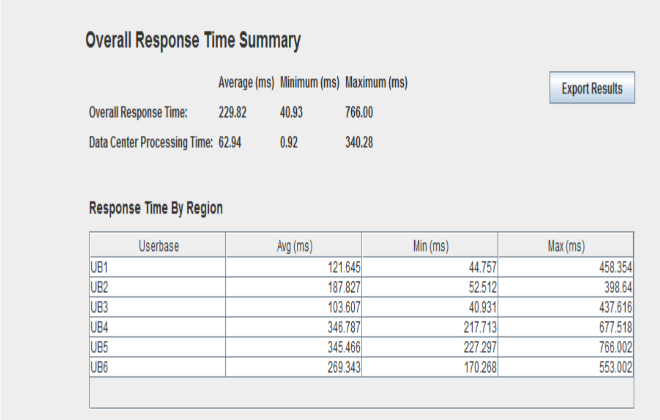
	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	305.20	41.41	26568.76
Data Center Processing Time:	124.86	1.40	26259.75

Export Results

Response Time By Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	188.268	47.755	654.074
UB2	256.852	59.504	584.003
UB3	203.323	41.411	737.505
UB4	420.094	222.513	11,585.504
UB5	421.572	227.917	26,568.755
UB6	336.314	174.766	659.02

Figure 10. ResponseTime For ESCE with 6 Userbases



Overall Response Time Summary

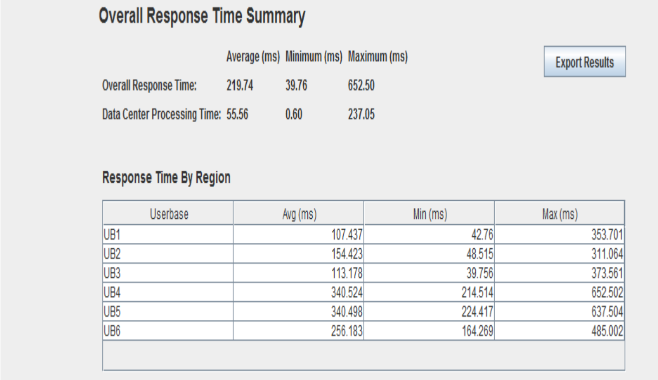
	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	229.82	40.93	766.00
Data Center Processing Time:	62.94	0.92	340.28

Export Results

Response Time By Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	121.645	44.757	458.354
UB2	187.827	52.512	398.64
UB3	103.607	40.931	437.616
UB4	346.787	217.713	677.518
UB5	345.466	227.297	766.002
UB6	269.343	170.268	553.002

Figure 11. ResponseTime For TLB with 6 Userbases



Overall Response Time Summary

	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	219.74	39.76	652.50
Data Center Processing Time:	55.56	0.60	237.05

Export Results

Response Time By Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	107.437	42.76	353.701
UB2	154.423	48.515	311.064
UB3	113.178	39.756	373.561
UB4	340.524	214.514	652.502
UB5	340.498	224.417	637.504
UB6	256.183	164.269	485.002

Figure 12. ResponseTime For EEA with 6 Userbases

C. Data Center Request Servicing Time:

Data Center Request Servicing Time for each data center calculated by the cloud analyst for each loading policy has been shown in the figure 13, 14 and 15 respectively.

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	146.316	10.626	428.752
DC2	218.021	21.253	455.034
DC3	141.005	1.7	55,299.504

Figure 13. Data Center Request Servicing Time For RR

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	121.655	8.75	345.013
DC2	182.407	17.501	359.275
DC3	116.897	1.4	26,259.754

Figure 14. Data Center Request Servicing Time For ESCE

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	68.436	5.75	168.303
DC2	124.64	11.508	340.281
DC3	42.564	0.92	302.357

Figure 15. Data Center Request Servicing Time For TLB

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	55.335	3.75	133.531
DC2	90.149	7.501	237.052
DC3	46.672	0.603	213.519

Figure 16. Data Center Request Servicing Time For EEA

D. Processing Cost:

The processing cost for each load balancing policy computed by the cloud analyst as can be seen from the figures 17,18,19 and 20

Cost

Total Virtual Machine Cost : $630.02
Total Data Transfer Cost : $72.18

Grand Total : $702.20

Figure 17. Processing Cost of RR

Cost

Total Virtual Machine Cost : $630.02
Total Data Transfer Cost : $59.44

Grand Total : $689.46

Figure 18. Processing Cost of ESCE

Cost

Total Virtual Machine Cost : $630.02
Total Data Transfer Cost : $42.46

Grand Total : $672.48

Figure 19. Processing Cost of TLB

Cost

Total Virtual Machine Cost : $630.02
Total Data Transfer Cost : $9.34

Grand Total : $639.36

Figure 20. Processing Cost of EEA

The above shown figures and graphs clearly indicates that the parameters : response time, data processing time and processing cost is almost similar in Round Robin and Equal Spread Current Execution scheduling algorithms whereas these parameters are bit improved in Throttled algorithm, and as per the EEA is concerned these are much improved. Therefore, we can easily identify that EEA is best among all, as live migration of load is done in virtual machine to avoid the underutilization and hence improving the and data transfer cost.

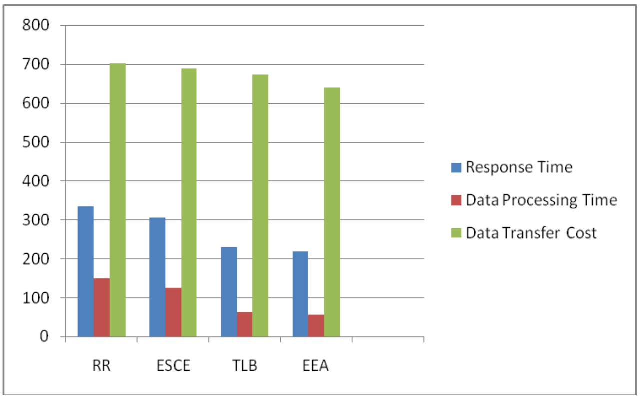


Figure 21. Analytical Comparison Of Various Algorithm In Cloud Computing

VI. CONCLUSION AND FUTURE SCOPE

In this paper, a new enhanced and efficient scheduling algorithm is proposed and then implemented in cloud computing environment using CloudSim toolkit, in java language. By visualizing the cited parameters in graphs and tables we can easily identify that the overall response time and data centre processing time is improved as well as cost is reduced in comparison to the existing scheduling parameters. The future work includes to overcome the problem of deadlocks and server overflow. We can also implement a new service broken policy in the simulator.

VII. REFRENCES

[1] Mishra , Ratan , Jaiswal, Anant,P"Ant Colony Optimiza tion: A Solution Of Load Balancing In Cloud",April 2012, International Journal Of Web & Semantic Technology;Apr2012, Vol. 3 Issue 2, P33

[2] Eddy Caron , Luis Rodero-Merino "Auto-Scaling , Load Balancing And Monitoring In Commercial And Open-Source Clouds " Research Report ,January2012

[3] Nidhi Jain Kansal, "Cloud Load Balancing Techniques : A Step Towards Green Computing", IJCSI International Journal Of Computer Science Issues, January 2012, Vol. 9, Issue 1, No 1, , Pg No.:238-246, ISSN (Online): 1694-0814

[4] Rich Lee, Bingchiang Jeng "Load Balancing Tactics In Cloud" International Conference On Cyber Enabled Distributed Computing And Knowledge Discovery, 2011

[5] Saroj Hiranwal , 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

[6] Bhathiya Wickremasinghe " Cloud Analyst: A Cloud-Sim-Based Tool For Modeling And Analysis Of Large Scale Cloud Computing Environments. MEDC Project" ,Report 2010 .

[7] Bhathiya Wickremasinghe ,Roderigo N. Calherios "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.

[8] Zhang, L. Cheng, And R. Boutaba, "Cloud Computing: State-Of-The-Art And Research Challenges", Journal Of Internet Services And Applications, , April 2010.

[9] A. Bhadani , And S. Chaudhary , "Performance Evaluation Of Web Servers Using Central Load Balancing Policy Over Virtual Machines On Cloud", Proceedings Of The Third Annual ACM Bangalore Conference (COMPUTE), January 2010.

[10] R. P. Mahowald, Worldwide Software As A Service 2010–2014 Forecast: Software Will Never Be Same ,In, IDC, 2010

[11] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling And Simulation Of Scalable Cloud Computing Environments And The Cloudsim Toolkit: Challenges And Opportunities," Proc. Of The 7th High Performance Computing And Simulation Conference (HPCS 09), IEEE Computer Society, June 2009.

[12] M. Armbrust , A. Fox, R. Griffith, A. D. Joseph, R. Katz, A.Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, And M. Zaharia, "Above The Clouds: A Berkeley View Of Cloud Computing", Eecs Department, University Of California , Berkeley,, February 2009,Technical Report No., Ucb/Eecs-2009-28, Pages 1-23.

[13] K.D. Devine, E.G. Boman , R.T. Hepahy, B.A.Hendrickson, J.D. Teresco, J. Faik,J.E. Flaherty,L.G. Gervasio," New Challenges In Dynamic Load Balancing,Applied Numerical Mathematics,52(2005)133-152.

[14] L. Kleinrock, A Vision For The Internet St Journal Of Research, Nov, 2005,(1),Pg4-5.

[15] http://www.livinginternet.com/w/wi_online.htm.

[16] www.cloudbus.org/cloudsim.

[17] http://www.ca.com/~/media/Files/whitepapers/turnkey_clouds_turnkey_profits.pdf.