ISSN:2320-0790

An International Journal of Advanced Computer Technology

Efficient Approach for Load Balancing in Virtual Cloud Computing Environment

[1]Harvinder singh, [2]Rakesh Chandra Gangwar

Dept. of CSE, Associate Professor, Dept. of CSE
BCET
Gurdaspur, India

Abstract: Cloud computing technology is changing the focus of IT world and it is becoming famous because of its great characteristics. Load balancing is one of the main challenges in cloud computing for distributing workloads across multiple computers or a computer cluster, network links, central processing units, disk drives, or other resources. Successful load balancing optimizes resource use, maximizes throughput, minimizes response time, and avoids overload. The objective of this paper to propose an approach for scheduling algorithms that can maintain the load balancing and provides better improved strategies through efficient job scheduling and modified resource allocation techniques. The results discussed in this paper, based on existing round robin, least connection, throttled load balance, fastest response time and a new proposed algorithm fastest with least connection scheduling algorithms. This new algorithm identifies the overall response time and data centre processing time is improved as well as cost is reduced in comparison to the existing scheduling parameters.

Keywords: Cloud Computing, load balancing, simulation, CloudSim.

I. INTRODUCTION

Over the past few years, Cloud computing technology drawn the attention of IT world and is the changing the focus of enterprises. Cloud computing can be defined as a style of computing where IT-related capabilities are provided to consumer as "service" rather than a product using the internet. Cloud stands as a metaphor for internet. Cloud computing gained attention due to the growth of internet technologies, reduced costs in storage, growth technologies of visualization and advancement in internet security. Cloud computing is, at its core, about delivering applications or services in an on-demand environment. Cloud computing providers will need to support hundreds of thousands of users and applications/services and ensure that they are fast, secure, and available. Along with visualization, infrastructure like load balancer, which does load balancing are key component to a successful cloud-based implementation. The following figure.1 shows the load balancing in cloud computing environment.

Load balancing is a computer networking method for distributing workloads across multiple computers or a computer cluster, network links, central processing units, disk drives, or other resources. Successful load balancing optimizes resource use, maximizes throughput, minimizes response time, and avoids overload. A variety of scheduling algorithms are used by load balancer to determine which back-end server to send a request to data center. Choosing the right load balancing algorithm is imperative to the success of cloud computing. The right load balancing will be able to provide the basics required to lay the foundation for more advanced cloud computing architectures. The following figure.2 shows the diagrammatical representation of the algorithm used for load balancing in cloud computing environment.

The remainder of this paper is organized as follows: A brief review of Cloud Computing is given in section II. Section III describes existing load balancing algorithms. Section IV proposed the research work. Section V describes the research setup and analysis. Section VI gives the research result description. Paper is concluded in section VII.

II. CLOUD COMPUTING

I. Brief Literature Survey

Cloud Computing, a forefront research channel in computer science, has the potential to change the face of the IT industry. There has been a significant amount of disagreement in how cloud computing is defined. Buyya et.al [1] have defined it as follows: Cloud is a parallel and distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically

provisioned and presented as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service provider and consumers. Due to the recent emergence of cloud computing research in load balancing this is in the preliminary stage. N J Kansal Jiyan [2] has proposed a service models are provided by the cloud. Rimal B.P. et. al [3] discussed the existing issues like Load Balancing, Virtual Machine Migration, Server Consolidation, Energy Management etc. Bhathiya. et. al [4] present execution environment considers Datacenter, Virtual Machine (VM), host and Cloudlet components from CloudSim for execution analysis of algorithms.

Zenon Chaczko. et. al [5] gives an idea about the basic concepts of Cloud Computing and Load balancing availability and load balancing in cloud Computing. R P Padhy et. al [6] studied about some existing load balancing algorithms, which can be applied to clouds. In addition to that, the closed-form solutions for minimum measurement and reporting time for single level tree networks with different load balancing strategies were also studied. The user or researcher can actually analyse the proposed design or existing algorithms through simulation. They can check the efficiency and merit of the design before the actual system is constructed.

Rajkumar Buyya et. al [7],in this paper he has studied the features of a CloudSim simulator to compare the performance of three dynamic load balancing algorithms. Bhathiya. et. al [4], have illustrated CloudSim architecture. W.Bhathiya et. al,"Cloud Analyst: A cloud sim-based visual modeller for analysing cloud computing environments and applications" [8], which present how cloud analyst can be used to model and evaluate a real world problem through a case study of a social networking application deployed on the cloud. The cloud analyst is a GUI based tool which is developed on CloudSim architecture. How the simulator can be used to effectively identify overall usage patterns and how such usage patterns affect data centres hosting the application. M.Sharma et. al [14], have discussed performance evaluation of adaptive virtual machine load balancing algorithms for cloud computing.

II. Cloud Computing

Buyya have defined cloud computing as follows: "Cloud is a parallel and distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service provider and consumers"[1].

Any cloud computing system consists of three major components such as clients, data center and distributed servers [3].

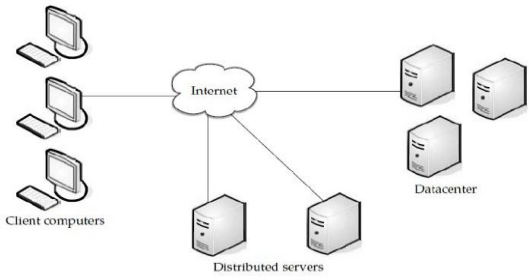

Fig. 1. Components of Cloud [9]

Client: End users interact with the clouds to manage information related to the cloud. Clients generally fall into three categories-Mobile: windows mobile smart phone like a blackberry or an I Phone. Thin: They don't do any computation work. They only display the information. Servers do all the work for them. The clients don't have any internal memory. Thick: These use different browsers like internet explorer or Mozilla fire fox or Google chrome to connect to the different cloud environment.

Datacenter: Datacenter is nothing but collection of servers hosting different applications. An end user connects to the datacenter to subscribe different applications. A datacenter may exist at a large distance from the clients.

Distributed Servers: A server, which actively checks the services of their hosts, known as Distributed server. It is the part of a cloud which is available throughout the internet hosting different applications. But while using the application from the cloud, the user would feel that they are using this application from its own machine [9].

III. LOAD BALANCING

Load balancing is a methodology to distribute workload across multiple computers, or other resources over the network links [10].Load balancing achieve optimal resource utilization, maximize throughput, minimum response time, and avoid overload. Cloud vendors are based on automatic load balancing services, which allow clients to increase the number of CPUs or memories for their resources to scale with increased demands. Load balancing serves two important needs, first to promote availability of Cloud resources, second to promote performance [14].

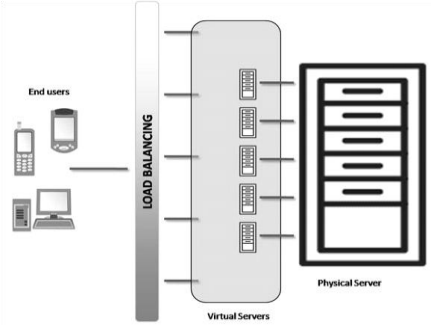

Fig. 2. Load balancing in cloud computing

1. Existing Load balancing algorithm

We use four existing algorithms to distribute the load.

A. Round Robin Algorithm (RR)[11]:

Round Robin algorithm distributes jobs evenly to all slave processors. Each process is assigned to the processor in a round robin order. The process allocation order is maintained locally independent of the allocations from remote processors. The work load distributions between processors are equal but the job processing time for different processes is not same. So at any point of time some nodes may be heavily loaded and others remain idle. This algorithm is mostly used in web servers where Http requests are of similar nature and distributed equally. This algorithm simply allots the job in round robin fashion which doesn't consider the load on different machines.

B. Least Connection Algorithm(LC)[8]:

This algorithm just keeps track of the number of connections attached to each server, and selects the one with the smallest number to receive the connection. This load balancing policy attempts to maintain equal work loads on all the available VMs. This load balancing policy attempts to maintain equal work loads on all the available VMs. Least Connection Load Balancer (LCLB) maintains an index table of VMs and the number of requests currently allocated to the VM. At the start all VM's have 0 allocations. When a request to allocate a new VM from the data center arrives, it parses the table and identifies the least loaded VM. If there are more than one, the first identified is selected. LCLB returns the VM id to the data center. The data center sends the request to the VM identified by that id.

C. Throttled Load Balancer(TLB)*[15]:*

In this algorithm the throttled load balancer (TVLB) maintains an index table of VMs as well as their state of the VM (Busy/Available). At the start all VM's are available. The data center controller (DCC) receives a new request from client/server to find a suitable virtual machine (VM) to perform the recommended job. The data centre queries the load balancer for the next allocation of VM. The load balancer parses the allocation table from top until the first available VM is found or the table is parsed completely. If the VM is found returns the VM id to the DCC. Further, the data centre acknowledges the load balancer of the new allocation and the data centre updates the allocation table accordingly. While processing the request of client, if appropriate VM is not found, the load balancer returns -1 to the data centre. The DCC queues the request with it. When the VM finishes processing the request, and the DCC receives the response, it notifies the load balancer a request is acknowledged to data centre to de- allocate the same VM whose id is already communicated. The DCC checks if there are any waiting requests in the queue. If there are, it continues.

D. Fastest Response Time(FR)[16]:

The Fastest method passes a new connection based on the fastest response time of all servers. The load balancer looks at the response time of each attached server and chooses the one with the best response time. Fastest VM Load Balancer (FLB) maintains a table which contains VMs and the response time of the VM. At the start all VM's are available. Datacenter receives a new request and queries the FLB for the next allocation.FLB scans the table from top until the first available the fast available VM is found. If the VM is found the data centre communicates the request to the VM and returns the VM id to the datacenter. 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.

IV. RESEARCH WORK

A. Fastest With Least Connection(FLC):

This algorithm is a combination of the logic used in the Least Connections and Fastest algorithms. With this method, servers are ranked based on a combination of the number of current connections and the response time. Servers that have a better balance of fewest connections and fastest response time receive a greater proportion of the connections

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

Response Time = Fint - Arrt + TDelay …… (1)

Where, Arrt- Arrival time of user request.
Fint- user request finish time.
TDelay - transmission delay.

TDelay = Tlatency + Ttransfer …… (2)

Where, T latency –network latency

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 - total available bandwidth

Nr - number of user requests currently in transmission.

The RANK of VM based on the response time and no of active connections of each Virtual Machine, which is calculated as:

(Rank) n= (Vmmax Ac-(VmAc) n) + (Vmmax Rt-(Vm Rt) n)

Vmmax Ac -VM maximum Active Connection value

(VmAc) n -Active Connection value of nth VM

Vmmax Rt -VM maximum Response Time value.

(VmRt) n -Response Time value of nth VM.

If one or more VM gets the highest rank, we will randomly choose a VM out of that with least connections or lowest response time.

Cost = totalTime * costPerVmHour.
total Time=toalTime + (end - start).
CostPerVmHour-1 hour cost per VM
start – start vmAllocationTime
end – end vmAllocationTime

V. RESEARCH SETUP & ANALYSIS

1. *Simulation*

Simulation is a technique where a program models the behavior of the system (CPU, network etc.) by calculating the interaction between its different entities using mathematical formulas', or actually capturing and playing back observations from a production system. The available Simulation tools in Cloud Computing today are: simjava, gridsim and CloudSim.

CloudSim is a framework developed by the GRIDS laboratory of University of Melbourne which enables seamless modeling, simulation and experimenting on designing Cloud computing infrastructures. CloudSim is a self-contained platform which can be used to model data centers, host, service brokers, scheduling and allocation policies of a large scaled Cloud platform. This CloudSim framework is built on top of GridSim framework which is also developed by the GRIDS laboratory. Hence, the researcher has used CloudSim to model datacenters, hosts, VMs for experimenting in simulated cloud environment [7].

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

2. *Cloud Analyst*[8][13] *:*

The Cloud Analyst is a GUI based tool which is developed on CloudSim architecture. The location of user bases has been defined in six different regions of the world as shown in figure4.

Two data centers used to handle the request of these users. One data center (DC) is located in is located in region 0, second in region1.On DC1 and DC2 number of VMs are 50. The duration of simulation is 60 min. In order to analyse various load balancing policies set the parameters for the user base configuration, application deployment configuration, and data center configuration as shown in figure 3, Table 1, Table 2 and Table 3 respectively.

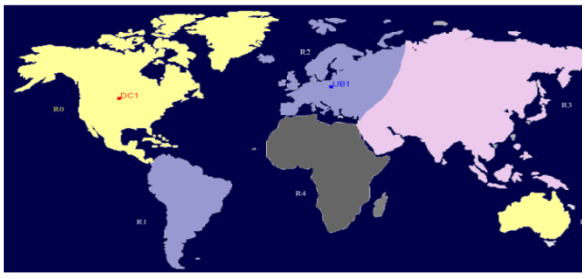

Fig.3. Cloud Analyst Region Screen.

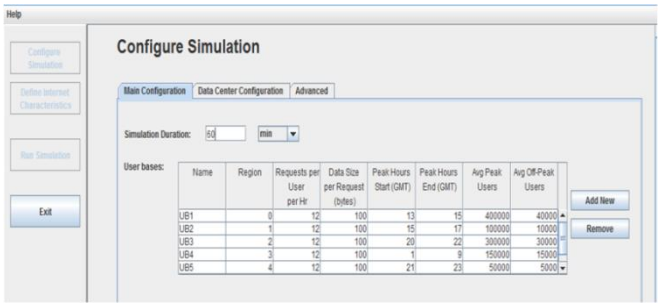

Table 1. Configure Screen in Simulator.

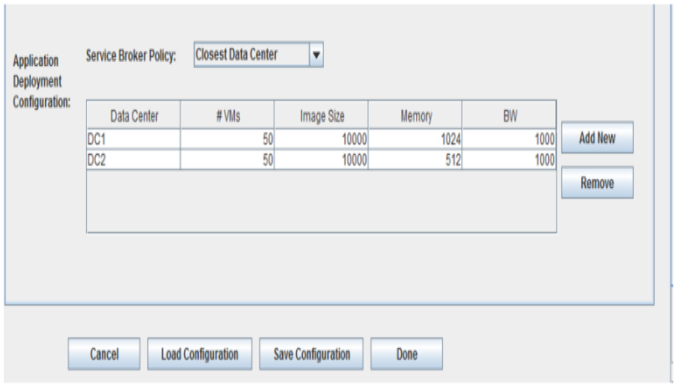

Table 2. Broker Policy Configuration.

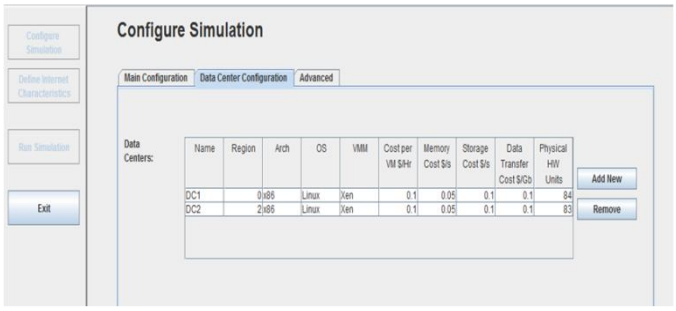

Table 3. Data centre Configuration

The simulation result computed by cloud analyst is as shown in the following figures. Output Screen of Cloud Analyst is as shown in the figure 7.Using 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.

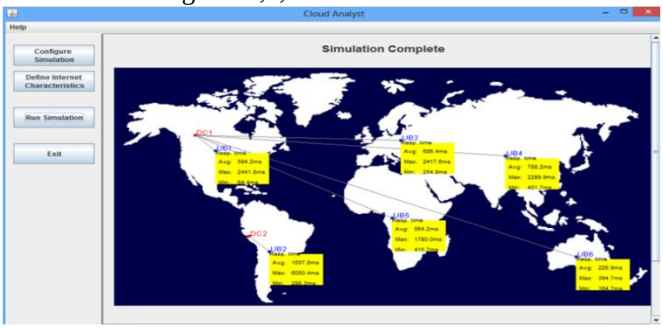

Fig. 4 Output Screen of Cloud Analyst.

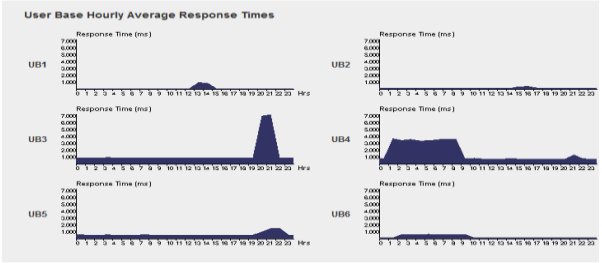

Fig. 5. UserHourlyResponseTime

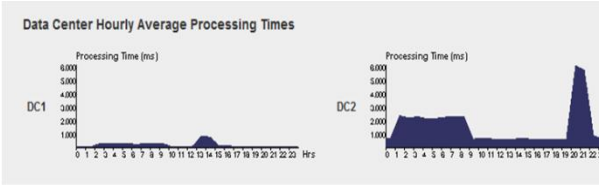

Fig. 6. DataProcessingTime

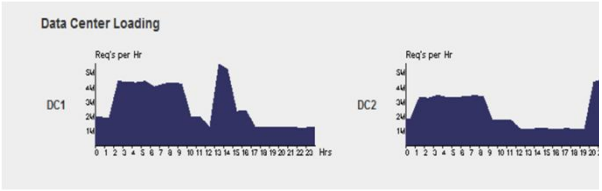

Fig. 7. Data Centre Loading.

2. *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 Table 4,5,6,7 and 8 respectively.

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	1761.53	42.11	5082.71
Data Center processing time:	1592.38	2.50	4629.56

Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	2241.36	1533.44	4504.30
UB2	54.03	42.11	69.44
UB3	2035.44	1447.95	4951.16
UB4	1305.06	522.63	5037.57
UB5	1241.31	643.03	5082.71
UB6	947.63	258.21	4634.34

Table 4. Response Time for RR with 6 User bases

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	1759.45	41.52	5067.66
Data Center processing time:	1593.40	2.50	4631.61

Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	2230.94	1528.53	4501.39
UB2	54.30	41.52	67.35
UB3	2031.20	1435.70	4958.62
UB4	1014.64	452.52	4710.31
UB5	1229.70	603.02	5067.66
UB6	948.53	305.69	4634.87

Table 5. Response Time for LC with 6 User bases

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	1057.88	41.52	4769.71
Data Center processing time:	893.26	2.50	4234.70

Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	1176.18	288.62	2549.38
UB2	53.93	41.52	66.86
UB3	1330.04	478.69	4212.86
UB4	1292.57	452.52	4467.55
UB5	1197.14	603.02	4769.71
UB6	709.95	287.76	2896.89

Table 6. Response Time for TLB with 6 User bases

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	1012.40	41.52	4770.29
Data Center processing time:	827.14	2.50	4185.28

Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	1173.71	288.62	2549.38
UB2	53.93	41.52	68.18
UB3	1247.05	478.69	4212.85
UB4	992.82	452.52	4502.79
UB5	1002.90	603.02	4770.29
UB6	628.67	253.04	2896.67

Table 7. Response Time for FRT with 6 User bases.

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	1001.76	41.52	4770.46
Data Center processing time:	811.71	2.49	4208.45

Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	1190.27	285.98	2800.95
UB2	53.97	41.52	68.18
UB3	1219.24	478.69	4213.05
UB4	925.61	452.52	4671.69
UB5	918.54	592.85	4770.46
UB6	598.83	253.04	3152.30

Table 8. Response Time for FLC with 6 User bases.

3. 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 Table 9,10,11,12 and 13 respectively.

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	1794.56	7.51	4631.61
DC2	3.32	2.50	5.69

Table 9. Data Center Request Servicing Time for RR

Least connection

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	1791.13	75.10	4629.56
DC2	3.02	2.50	4.78

Table 10. Data Center Request Servicing Time for LC.

Throttled Load Balancer

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	1005.89	7.51	4234.70
DC2	2.98	2.50	3.50

Table 11. Data Center Request Servicing Time for TLB.

Fastest Response Time

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	927.24	7.51	4185.28
DC2	3.00	2.50	3.73

Table 12. Data Center Request Servicing Time for FRT.

Fastest with Least Connection

Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	905.57	7.51	4208.45
DC2	3.00	2.49	3.73

Table 13. Data Center Request Servicing Time for FLC.

4. Processing Cost

The processing cost for each load balancing policy computed by the cloud analyst as can be seen from the figures 8,9,10,11 and 12.

RRB

Cost

Total Virtual Machine Cost ($):	10.04
Total Data Transfer Cost ($):	11.27
Grand Total: ($)	21.31

Fig. 8. Processing Cost for RRB.

Least Connection

Cost

Total Virtual Machine Cost ($):	10.04
Total Data Transfer Cost ($):	11.16
Grand Total: ($)	21.20

Fig. 9. Processing cost For LC.

Throttled Load Balancer

Cost

Total Virtual Machine Cost ($):	10.04
Total Data Transfer Cost ($):	11.09
Grand Total: ($)	21.13

Fig. 10. Processing cost For TLB.

Fastest Response Time

Cost

Total Virtual Machine Cost ($):	10.04
Total Data Transfer Cost ($):	10.86
Grand Total: ($)	20.90

Fig. 11. Processing Cost of FRT

Fastest with Least Connection

Cost

Total Virtual Machine Cost ($):	10.04
Total Data Transfer Cost ($):	10.00
Grand Total: ($)	20.04

Fig. 12. Processing Cost of FLC

VI. RESULT

Analysis of the simulation we get the desire outputs for the entire five load balancing algorithms.The above shown figures and graphs clearly indicates that the parameters: response time, data processing time and processing cost is almost similar in RRB and LC scheduling algorithms whereas these parameters are bit improved in TLB and FR as per the FLC is concerned these are much improved. Therefore, we can easily identify that FLC is best among all.

VII. CONCLUSIONS

The performances of four existing algorithms are studied in the paper. The paper aims to development of enhanced strategies through improved job and load balancing resource allocation techniques. A new fastest with least connection scheduling algorithm is proposed and then implemented in cloud computing environment using CloudSim toolkit, in java language. 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. Fastest Response Time and Fastest with Least dynamically allocates the resource to the job in a queue leading reduced cost in data transfer and virtual machine formation. The simulation result shows the reduction up to 50-60%in the cost and time.

REFERENCES

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems, 25:599616, 2009.

[2] 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.

[3] Rimal B.P., Choi E. and Lumb I. (2009) 5th International Joint Conference on INC, IMS and IDC, 44-51.

[4] Bhathiya, Wickremasinghe.(2010)"Cloud Analyst: A Cloud Sim-based Visual Modeller for Analysing Cloud Computing Environments and Applications"

[5] Zenon Chaczko, Venkatesh Mahadevan, Shahrzad Aslanzadeh, Christopher Mcdermid (2011)"Availabity and Load Balancing in Cloud Computing" International Conference on Computer and Software Modeling IPCSIT vol.14 IACSIT Press, Singapore 2011.

[6] Ram Prasad Padhy (107CS046), PGoutam Prasad Rao (107CS039)."Load balancing in cloud computing system" Department of Computer Science and Engineering National Institute of Technology, Rourkela Rourkela-769 008, Orissa, India May, 2011.

[7] Calheiros Rodrigo N., Rajiv Ranjan, and César A. F. De Rose, Rajkumar Buyya (2009): CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services CoRR abs/0903.2525: (2009).

[8] Bhathiya Wickremasinghe "Cloud Analyst: A Cloud-Sim-Based Tool For Modeling And Analysis Of Large Scale Cloud Computing Environments. MEDC Project", Report 2010. Tackle your client's security issues with cloud computing in 10 steps, http://searchsecuritychannel.techtarget.com/tip/Tackle-your-clients-security-issues-withcloud-computing-in-10-steps.

[9] 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 Cloudcomputing.TechnicalReport No. UCB/EECS-2009-28, University of California at Berkley, USA, Feb. 10, 2009.

[10] Ko, Soon-Heum; Kim, Nayong; Kim, Joohyun; Thota, Abhinav; Jha, and Shantenu; (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.

[11] 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.

[12] Jinhua Hu; Jianhua Gu; Guofei Sun; Tianhai Zhao; (2010) "A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment" Third International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), 18-20 Dec. 2010, pp.89-96

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

[14] 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.

[15] Sandeep Sharma, Sarabjit Singh, and Meenakshi Sharma "Performance Analysis of Load Balancing Algorithms" World Academy of Science, Engineering and Technology 38 ,2008 page no 269- 272.

[16] Zenon Chaczko, Venkatesh Mahadevan, Shahrzad Aslanzadeh, Christopher Mcdermid (2011)"Availabity and Load Balancing in Cloud Computing" International Conference on Computer and Software Modeling IPCSIT vol.14 IACSIT Press, Singapore 2011.

[17] Cloud Security and Privacy An Enterprise Perspective on risk and compliance by Tim Mather, Subra kumaraswamy, Shaheed Latif

[18] Implementing and developing Cloud Computing Application by David E.Y Sarna .

[19] Kun Li, 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.
Huaiming Song, Yanlong Yin, Xian-He Sun, Thakur, R. and Lang, S.; (2011) "A Segment-Level Adaptive Data Layout Scheme for Improved Load Balance in Parallel File Systems" 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), 23-26 May 2011, pp.414-423.

[20] Meenakshi Sharma and Pankaj Sharma "Performance Evaluation of Adaptive Virtual Machine Load Balancing Algorithm" (IJACSA) International Journal of Advanced Computer Science and Applications,
Vol. 3, No.2, 2012

[21] Don MacVittie, Intro to Load Balancing for Developers – March 31.

[22] CLOUD COMPUTING MADE EASY by Cary Landis and Dan Blacharski,version 0.3.