Improved Hybrid Algorithm Approach based Load Balancing Technique in Cloud Computing

By Srinivasa Rao Gundu & T. Anuradha
Dravidian University Kuppam

Abstract- The routine life of modern citizens is completely dominated by the computer aided services. The computer aided services depends on information and communication technologies. The success behind this cloud computing are data centers with virtualization technology equipped with fastest internet and the wide acceptance of the users due to its affordable price to the common people. Effective services can be provided to the end user only when proper scheduling of tasks are done in peak hours when heterogeneous collection of requests are coming to the data center. The load is balanced in such situations, to attain equilibrium position in load. It is a difficult task to balance the load. Load can be balanced using either by software or hardware.

Keywords: cloud computing, cpu, latency, performance, software, networking, load, load balancing.

GJCST-B Classification: C.2.4
Improved Hybrid Algorithm Approach based Load Balancing Technique in Cloud Computing

Srinivasa Rao Gundu & T. Anuradha

Abstract: The routine life of modern citizens is completely dominated by the computer aided services. The computer aided services depends on information and communication technologies. The success behind this cloud computing are data centers with virtualization technology equipped with fastest internet and the wide acceptance of the users due to its affordable price to the common people. Effective services can be provided to the end user only when proper scheduling of tasks are done in peak hours when heterogeneous collection of requests are coming to the data center. The load is balanced in such situations, to attain equilibrium position in load. It is a difficult task to balance the load. Load can be balanced using either by software or hardware. In software approach algorithm based load balancing makes us to achieve the better results. Hybrid approach in algorithm is a combination of more than one algorithm. RTEA Hybrid algorithm has shown better results in the considered parameters of makespan, waiting time, and burst time.

Keywords: cloud computing, cpu, latency, performance, software, networking, load, load balancing.

I. Introduction

The cloud computing is a specialized environment with a huge collection of computer systems which are connected to internet with either public or private networks. This cloud computing environment provides facility to the end user on demand. This technology has reduced the price to acquire the different softwares, operating systems, and tools which were only available to the larger companies upon huge investments. These services are available on demand [1] and the services are metered and the user needs to pay only for the services are used. It is one or other way to customized usage of information technologies. Cloud computing is a collection of heterogeneous services under a single umbrella of cloud service provider. Therefore in many cases we do not need to install or acquire new devices to have some specialized services by the cloud computing service providers such as, Google provides its several type of services to the end users such as Gmail, Google docs, Google maps etc. Microsoft Company is also providing the services such as Microsoft Office 365 and Microsoft azure to the end users. Sales force, Amazon is also providing the user needed services.

The cloud computing services are provided on demand and as per the user requirements. This process does not need to be done by somebody manually perhaps automatically which is called as software automation process. This type of dynamic provisioning provides not only increased service capacity but also provides reliability, security and privacy. The cloud services can be had using any one of the devices using such as personal computers, mobiles etc using any type of network access.

II. Computing at Data Centre Level

The data centre and cloud architectures [4] are keeping on advancing, to address the requirements of expansive scale multi server farms in clouds. These requirements are revolved around seven measurements called (i) scalability, (ii) storage, (iv) bandwidth, (v) speed in network services, (vi) efficiency in memory utilization, (vii) agility in service creation, (viii) cost productivity.

III. Review of Related Literature

1. T. Kokila vani has proposed Load Balanced Min Min scheduling set of rules [5] which produces higher effects than minmin scheduling set of rules. It reduced the make span and attained improvement.
2. Jens Buysse et al. described a new method to minimize [6] the power intake. This routing and scheduling algorithm selects unused nodes and turns off those nodes. In step one, it chooses the special nodes and which are marked as the destination nodes. In the second step, it decides the unicast route to a particular destination to supply. The results illustrated the power consumption reducing by 50% lesser in comparison with other standard scheduling and routing algorithms.
3. Wanneng Shu et al. Proposed a new type of power effective algorithm in which the resources [7] are allocated in a cloud computing environment. The CloudSim device kit is used to study the performance of the set of rules. Time, cost and energy intake are used as the parameters in this set of rules. The response time and make span has drastically reduced.
4. M. Hemamalini, M. V. Srinath, et al carried out their study on a heuristic set of rules for data placement. In this study, the heuristic method focuses [8] on
the node scheduling which is used to reduce the energy consumption in the cloud. This study has accommodated the maximum data on the minimum quantity of nodes. It has used the greedy approach for this process. The algorithm is implemented with the use of Cloud Sim.

5. Ms. Nitika, Ms. Shaveta, Mr. Gaurav Raj carried their research on Equally Spread Current Execution Algorithm. In this study, the set of rules are used to handle the system with priorities and it has distributed the burden randomly [9]. This distribution is done with the help of a parameter scale. This process transfers the load to the lightly loaded digital gadget or handles that venture and takes much less time. It is observed by the researchers that it provides the maximize throughput. This set of rules makes use of the spread spectrum method to balance the weight of the task considering into more than one virtual machines.

6. Bhuvnesh Pathania et al. proved the performance of his proposed approach which has reduced the electricity consumption. In this approach virtual machines are loaded with variable load [10]. This load is balanced in the grid clusters which are enough to manipulate the required Quality grade.

7. Jianfeng Zhao and Hongze Qiu et al. provided their research related information with the help of a replication approach to achieve many unique possibilities, such as lowering the system running time and power consumption. The algorithm has [11] employed genetic protocol and used ant colony optimization technique.

8. Javid Taheri et al. has worked on match making scheduling segment and offered algorithms to lessen the make span for executing all jobs and their transfer time [12]. It used two distinct set of rules for scheduling the activity to mirror the information fields to the linked nodes.

9. Fabrizio Ferrandi et al. has offered a new method on ant colony optimization technique. This technique efficiently solves the problems which encounter in multiple scheduling [13]. It reduces the overall execution time. It gives mapping of undertaking and gives a message.

10. A. P. Nirmala, Dr. S. Veni, et al. Proposed a new hybrid scheduling mechanism and a step rescheduling approach. In this mechanism, suitable resources are decided based on the availability [14]. All scheduling activities are done with the help of dynamically allocated resources.

11. Poonam Singh, Maitreyee Dutta, and Naveen Aggarwal et al. has made a proposal of new algorithm for optimizing the make span and the reliability of workflow scheduling in cloud systems. This technique uses the nearby search or shortest path for designing [15] a new type of and quicker heuristic method. This method tried to solve the hassle of mapping independent scheduling to a virtual machine.

IV. Scheduling

Generally, scheduling [16] is a term used for a set of rules that will govern the order by the execution of a particular process. Here the process is a word used for an executable programming part. A process execution will undergo in the computer system such as a CPU (central processing unit) burst followed by an IO (input output) burst again a CPU (central processing unit) burst followed by an IO (input output) burst and so on. ‘Burst’ is an action with as fast as possible highest speed and continued action. Scheduling is an important property of an operating system. Scheduling process can be looked as ‘Service request scheduling' and ‘Resource response scheduling’. Generally, service request scheduling occurs due to (i) When the user submits his or her request to the service provider (ii) Service provider executes the request (iii) Processing the request in the service request architecture (iv) Dynamic virtual machine generation and dispatch at the provider site. A computer system needs scheduling before use; CPU (central processing unit) is one of the most critical parts in the computer. Multiprogramming is one of the basic and important scheduling techniques. Generally CPU (central processing unit) scheduling is done in such a way as to keep it busy as much as possible. The are two issues (i). Jobs must be smaller so that they could receive large fraction of number of processors they requested. (ii). Execution efficiency of smaller jobs may be low.

V. Virtualization

In cloud computing, virtualization [17] is a key idea. Virtualization improves the effectiveness of computing resources utilization and additionally the dynamic resource provisioning capacities in a cloud. The virtual machine, whose associated necessities cannot be domestically fulfilled, are selected for migration. This set of rules searches the maximum loaded virtual machines and allocate the load in an effective manner. Virtualization provides the applications to migrate from one server to another. This migration is possible dynamically, which means there is no need to make the server down, and the workload can be easily managed.

VI. Load Balancing

The main objective of load balancing [18] in a computing environment is to guarantee that no single system is overloaded with tasks, while the other physical node is left idle. The key criteria for a good load balancer are to maintain a balanced state of workload among the actively participating computational nodes.
Apart from the central task of load balancing, scheduled jobs, it also focuses on maximizing the throughput, minimizing the response time and better resource utilization. A cloud load balancer should have the characteristics in such a way that the cloud service providers will not be overloaded with the set of requests.

VII. Internal Working Mechanism of Hybrid Algorithms

The algorithm is a step by step method with a set of rules that can provide solutions within a finite number of steps. A hybrid algorithm [19] is a combination of two or more algorithms. Before a user sends a ‘request’ to the load balancer, it resolves the load balancer’s domain name using a domain name server (DNS). Domain name server returns one or more internet protocol addresses (IP address) to the client machine. With network load balancers, load balancing creates a network interface for each available virtual machines list which is either busy or idle. Each load balancer node in this virtual machine list uses this network interface to get a static internet protocol address (static IP Address). It can be optionally associated one internet protocol address with each network interface when it creates the load balancer. As traffic to the application changes over time, load balancing scales the load balancer and updates the DNS entry. Here one important aspect is that domain name server entry specifies the TTL (Time to live) as sixty seconds. TTL makes the internet protocol address can be re mapped quickly in response to change the traffic. The client machine determines which internet protocol address need to use for sending the requests to the load balancer. Load balancer node requests the server. Load balancer node selects a virtual machine which can handle the request. Now the load balancer node sends the request to the target using its private internet protocol address (Private IP Address). When a load balancer accepts incoming traffic from the client and routes the requests to its virtual machine pool. The load balancer also checks whether the virtual machine is either idle or busy and ensures that it routes the traffic only to the virtual machines which are capable to handle. If it is being used equally spread current execution algorithm, for example, it is cross zone load balancing. In this load balancing, the nodes of the load balancer would distribute the requests regardless of the availability of virtual machines and distributes the traffic evenly across all the virtual machines in the pool of virtual machines. With the application of load balancer, the load balancer node receives the request and evaluates the priority order to determine which rule to apply and then selects a virtual machine from the pool of virtual machines for the rule action using the routing algorithm. Routing is performed independently for each one of the virtual machines in the virtual machine pool, even though when the virtual machine is assigned to the pool of virtual machines. With network load balancers the load balancer node receives the connection to select the virtual machine from the virtual machine pool for the default rule using a flow hash algorithm. Based on the protocol, source internet protocol address (Source IP Address), source port, destination internet protocol address (Destination IP Address), destination port and transmission control protocol sequence number, select the virtual machine. The transmission control protocol (TCP) connections from a client have different source ports and sequence numbers and can be routed to a different virtual machine. Each individual transmission control protocol connection is routed to a single virtual machine.

VIII. Load Balancing Parameters Focused

1. Make Span: is also called as job completion time. It should be less than response time. Job completion time = Response time - Waiting time which are measured in nano seconds
2. Waiting time: is the time difference between turnaround time and burst time. Waiting time = Turnaround time - Burst time which are measured in nano seconds
3. Burst time: is the Time required by a process for CPU execution Burst time = Turn Around Time - Waiting time which are measured in nano seconds

IX. Implementational Details

Technologies used in this tool are (i). Java programming language, (ii) Java Swing, (iii) Cloud sim (iv) Simjava, (v) Operating system are Windows Xp to Windows7, (vi) IDE: Eclipse, (vii) JDK 1.8 and above. It is aimed to investigate and simulate the large scale internet applications in the cloud environment. Most of the internet applications depend upon many parameters and most of the time the values for those parameters need to be assumed. Therefore it is important to change those parameters and repeat the simulations.

Round Robin Algorithm: Round robin is the easiest algorithm used for the logic division of time and nodes. In this algorithm, the overall time is partitioned into the number of segments, and each node in the system is allocated a with a particular time segment or time period.

Inputs: Virtual machine image size 10000 (Image size is operating system, application, data to be installed on multiple virtual machines- Usually it is tested for security, reliability and has the best tested configuration), Virtual machine memory - 1024 Mb, Virtual machine bandwidth -1000, Virtual machine operating system- Linux, Virtual machine manager - Xen, RAM – 204800 Mb, Storage...
space – 100TB, Bandwidth - 10,0000 ,VM scheduling
policy - Time shared policy 4 1000 MIPS., Data Centre
Configuration ,Data Centre Architecture – X86, Data
Centre Operating System – Linux, Data Centre virtual
machine manager – Xen, Number of machines in the
data centre – 20, Memory per machine in data centre –
2048 Mb, Storage per machine in data centre–100000
Mb, Available band width per machine in data centre –
10000, Number of processors per machine in data
centre – 4, Processor speed in data centre –100 MIPS,
Virtual machine policy in data centre – Time shared
policy, User grouping factor in data centre –1000,
Request grouping factor in data centre–100, Executable
instruction length in data centre– 250.

Results obtained on round robin algorithm: The results
obtained in round robin algorithm as given below in
Table 1. Here the Average response time in nano
seconds of time.

Table 1: Result obtained on round robin algorithm

<table>
<thead>
<tr>
<th></th>
<th>RR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.37</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.04</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Demerits of round robin algorithm: (i) Clients have to wait
in the waiting queue until and unless the suitable
virtual machine is available (ii) The additional load on the
scheduler to decide the size of quantum.

a) Throttled algorithm and its features
   (i). It is a static algorithm or static scheduling
   algorithm, (ii). It is a virtual machine based algorithm,
   (iii).Throttled virtual machine load balancer keeps a list
   of virtual machines and their status whether they are
   busy or idling when the user cloudlet request comes
   assigns to a proper virtual machine and lets the work to
   be done.

Inputs: In case of inputs, same virtual machine
configuration and data centre configurations need to
give in order to get proper evaluation of throttled
algorithm.

b) Results obtained on Throttled algorithm
   Throttled algorithm results are as given below in
   Table 2. The values stated here are taken on average and
   measured in nano seconds.

Table 2: Result obtained on Throttled Algorithm

<table>
<thead>
<tr>
<th></th>
<th>Throttled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.38</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.03</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Demerits of throttled algorithm: It works properly only if
all virtual machines in data centre have the same
hardware configuration. Therefore there is a need to
improvement. Hybrid approach can be used to improve
this algorithm. Modifications in the algorithm would bring
flexibility.

Combination of round robin and throttled algorithm: As
since cloud computing needs heterogeneous nature of
virtual machines, throttled algorithm does not support.
Round robin algorithm can be combined with throttled
algorithm in a hybrid approach.

Inputs: In case of inputs, same virtual machine
configuration and data centre configurations need to
give in order to get proper evaluation of RTH algorithm.

Results obtained from the execution of RTH algorithm:
RTH algorithm results are as given below in Table 3. The
values stated here are taken on average and measured
in nano seconds.

Table 3: Result obtained from the execution of RTH
algorithm

<table>
<thead>
<tr>
<th></th>
<th>RTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.34</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.02</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.24</td>
</tr>
</tbody>
</table>

RTH algorithm comparison with round robin and throttled
algorithm: RTH algorithm comparison with round robin
and Throttled algorithms results are as given below in
Table 4. The values stated here are taken on average and
measured in nano seconds.

Table 4: RTH algorithm in comparison with Round Robin
and Throttled Algorithms

<table>
<thead>
<tr>
<th></th>
<th>RR</th>
<th>Throttled</th>
<th>RTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.37</td>
<td>300.38</td>
<td>300.34</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.04</td>
<td>0.03</td>
<td>0.02</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.25</td>
<td>0.25</td>
<td>0.24</td>
</tr>
</tbody>
</table>

Advantages of RTH Algorithm: A slight improvement is
observed in makespan waiting time and burst time.

Problem statement: It is observed that it requires all
virtual machines configuration to be the same then only
it is showing better performance. In other words,
updating the index table is getting delayed in providing
the response to the arrived requests.

ESCE algorithm: It is based on the spread spectrum
method. In this method, the load balancer monitors the
scheduled jobs. Equally Spread Current Execution load
balancer puts all the tasks in the job pool and assigns
them to the virtual machine. As the load balancer
monitors the Scheduled Jobs, balancer keeps track of
job’s which are in the queue frequently.

Inputs: In case of inputs, same virtual machine
configuration and data centre configurations need to
give in order to get proper evaluation of equally spread
current execution algorithm.
ESCE algorithm simulation result: Equally spread current execution algorithm simulation result table would look like as given below in Table 5.

**Table 5: Equally Spread Current Execution (ESCE) Algorithm Simulation Result**

<table>
<thead>
<tr>
<th></th>
<th>ESCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.38</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.03</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.26</td>
</tr>
</tbody>
</table>

Problem with ESCE algorithm: There is a computational overhead to scan the Queue again and again RTH, ESCE algorithm comparison table is given below in Table 6. The values stated here are taken on average and measured in nano seconds.

**Table 6: RTH Algorithm and Equally Spread Current Execution (ESCE) Algorithm Comparison**

<table>
<thead>
<tr>
<th></th>
<th>RTH</th>
<th>ESCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.37</td>
<td>300.38</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.04</td>
<td>0.03</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.25</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Combination of RTH and ESCE algorithms: To improve RTH algorithm, Equally spread current execution algorithm can be combined with hybridization concept.

Results obtained from the execution of RTH Algorithm: RTEH algorithm results are as given below in Table 7. The values stated here are taken on average and measured in nano seconds.

**Table 7: Result obtained from the execution of RTEH algorithm**

<table>
<thead>
<tr>
<th></th>
<th>RTEH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.34</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.02</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.24</td>
</tr>
</tbody>
</table>

RTEH Algorithm in comparison with earlier Algorithms: RTEH Algorithm in comparison with earlier algorithm table are look like as given below in Table 7. The values stated here are taken on average and measured in nano seconds.

**Table 7: RTEH Algorithm in comparison with Earlier Algorithms**

<table>
<thead>
<tr>
<th>RR</th>
<th>Makespan</th>
<th>Throttled</th>
<th>RTH</th>
<th>ESCE</th>
<th>RTEH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>300.37</td>
<td>300.38</td>
<td>300.34</td>
<td>300.38</td>
<td>295.35</td>
</tr>
<tr>
<td></td>
<td>0.04</td>
<td>0.03</td>
<td>0.02</td>
<td>0.03</td>
<td>0.02</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td>0.25</td>
<td>0.24</td>
<td>0.24</td>
<td>0.24</td>
</tr>
</tbody>
</table>

Advantages of new RTEH Algorithm: A slight improvement is observed in makespan waiting time and burst time.

Disadvantages of new RTEH Algorithm: It is observed that updating the index table is getting overhead.

Search for a new algorithm for the betterment in RTEH Algorithm: The main drawback of RTEH Algorithm is updating in the table is getting overhead. There is a need to improve this algorithm.

Artificial bee colony optimization Algorithm: Karaboga, proposed a foraging behaviour of honey bee swarm in 2005. Artificial Bee Colony Optimization technique was inspired by the above mentioned behaviour of the bee swarm. Artificial Bee Colony Optimization Algorithm is a novel metaheuristic approach which was developed in 2005. It is a dynamic algorithm. Best Suited for cloud Environment. Best suited for Parallel Processing. There will be three types of Bees (a) Scout Bees: Arbitrarily searches for Honey (b) Onlooker Bees: Determines the food source by calculating the fitness (c) Employed Bees: Gather the information about the food resource and exchange the information gathered with onlooker bees.

Artificial Bee Colony Optimization Algorithm Simulation Result: ABCO Algorithm simulation results is given below in Table 8.

**Table 8: Load balancing in Artificial Bee Colony Optimization (ABCO) Algorithm**

<table>
<thead>
<tr>
<th></th>
<th>ABCO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.34</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.02</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.24</td>
</tr>
</tbody>
</table>

Problem statement: It is observed that this algorithm is slow to obtain an accurate solution and it requires new fitness tasks on the new algorithm parameters to improve performance. Therefore there is a need for improvement.

Need for the improvement: As since the artificial bee colony optimization algorithm requires improvements. As per the existing real time scenario there is a need for the algorithm with the following characteristics (i) The algorithm needs to work efficiently in the distributed environment (ii) The algorithm needs to be a virtual machine friendly, (iii) The algorithm should make the virtual machine to work efficiently under the heavy loads and (iv) The algorithm should assure that it is capable to provide services in the peak demand hours sufficiently.

Combination of RTEH and ABCO Algorithms: The ABCO is a novel and a heuristic algorithm. This algorithm is motivated by honey bee intelligent foraging behaviour. It is used for a searching process to solve the real time parameters optimization problems. The Disadvantages of this algorithm is, it is slow in getting the better result. Many numbers of objective function evaluations are required. There is a possibility of losing relevant information when the function is being optimized. To
improve RTEH algorithm, ABCO algorithm can be combined using the hybrid approach. This algorithm can be called as RTEAH algorithm.

**RTEAH Algorithm Simulation Result**: RTEAH Algorithm simulation results are as given below in Table 9.

**Table 9**: Load balancing in RTEAH Algorithm

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Makespan</th>
<th>Waiting time</th>
<th>Burst time</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTEAH</td>
<td>292.46</td>
<td>0.01</td>
<td>0.23</td>
</tr>
</tbody>
</table>

RTEAH Algorithm in comparison with other Algorithms

RTEAH Algorithm can be compared with other previous algorithms as given below in Table.10

**Table 10**: RTEAH Algorithm in comparison with other algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>RR</th>
<th>Throttled</th>
<th>RTH</th>
<th>ESCE</th>
<th>RTEH</th>
<th>RTEAH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Makespan</td>
<td>300.3</td>
<td>300.3</td>
<td>300.3</td>
<td>300.3</td>
<td>295.3</td>
<td>292.4</td>
</tr>
<tr>
<td>Waiting time</td>
<td>0.04</td>
<td>0.03</td>
<td>0.02</td>
<td>0.03</td>
<td>0.02</td>
<td>0.01</td>
</tr>
<tr>
<td>Burst time</td>
<td>0.25</td>
<td>0.25</td>
<td>0.24</td>
<td>0.26</td>
<td>0.24</td>
<td>0.23</td>
</tr>
</tbody>
</table>

Comparison of RR, Throttled, RTH, ESCE, RTEH and RTEAH algorithms for the makespan in the form of a graph1.

Graph 1: Comparison of different algorithms for makespan

Comparison of RR, Throttled, RTH, ESCE, RTEH and RTEAH algorithms for waiting time in the form of a graph2.

Graph 2: Comparison of different algorithms for waiting time.

Comparison of RR, Throttled, RTH, ESCE, RTEH and RTEAH algorithms for Burst time in the form of a graph3.

Graph 3: Comparison of different algorithms for Burst time.

c) **Advantages of RTEAH algorithm**

The New RTEAH Algorithm has shown betterment in case the Cloud Computing Load Balancing Parameters in terms of Makespan, Waiting time and Burst time. Therefore RTEAH is considerable.

d) **Limitations and Scope of Research**

i. The research is focused on load balancing in hybrid algorithmic approach in cloud computing environment.

ii. Efforts are done to make betterment of the performance of scheduling in the cloud environment.

iii. Research study is focused on load balancing in hybrid algorithmic approach in cloud computing environment.
Conclusion and Future work: The new RTEAH algorithm has shown betterment in case of some of the considered cloud computing load balancing parameters such as Makespan, Waiting time and Burst time, keeping in view of Data centers and virtual machines, number of requests allotted, number of requests successes, number of request failed, during the peak hours and in non peak hours as since the requests are coming in random way RTEAH algorithm has shown the considerable performance. Proposed approaches in algorithms can be used in cloud IaaS. Some parameters did not considered in the present research such as given below (i). Network Bandwidth (ii). Memory Utilization ratio (iii). CPU Utilization ratio (iv). Disk space usage (v). CPU Processing Speed (vi). Latency is the time taken to return a data packet to the sender. (vii). Fault Tolerance.

References Références Referencias
