ABSTRACT: Many task scheduling algorithms have been designed by various researchers which address the problem of task scheduling effectively and efficiently. With the passage of time and continuous research, almost every factor which increases the performance of the cloud, has been considered minutely, but cost was the only factor which has not been taken seriously. Cost plays an important role for the organization because at the end which matters the most to set up any business, is money. To measure the cost, there is need to work upon some parameters like the cost of resources, CPU time, turnaround time etc. In this paper, an algorithm has been proposed which takes care of the cost of these primary factors and the overall cost of the activity. ABC has been considered to be the best technique to schedule the task with the consideration of cost but there is always hope of more improvement. The implementation of the algorithm of ABC has been compared with the new algorithm, which has been proposed in this paper.

Keywords: Activity Based Costing, Cloud Computing, Cost Factor, Execution Time, Task Scheduling.

1. Introduction

Cloud Computing is a way to access, store and manipulate data on internet environment without wasting lots of computer memory of individual systems. This technique decreases the processing burden at the user end. Traditionally, users spend lots of money on hard-disks for a large amount of storage, on processors with high processing speed, on coolants to maintain the temperature etc. But with the help cloud computing, this problem has been solved to a great extent. The organization team of the companies is giving attention to improve the performance of the services they provide instead of spending time on money issues [1]. Due to so many advantages offered by Cloud Computing, it became very popular and developed rapidly in few years. Cloud Computing is based on three fundamentals [2]:

- Distributed Computing: Cloud is not just a single server, it is a hub of servers where millions of servers interact to provide efficient services to the end users. When a task is requested by the user, that task is splitted into modules and these modules are distributed to different servers according to their processing speed and available memory.
- Dynamic Extendibility: Extendibility is a basic principle which comes into existence with a hope of future growth. In the cloud, users are allowed to add more servers dynamically according to their requirements [3]. In case, any present server fails or crashes in between then its tasks are dynamically shifted to the new server without creating any interruption in work of end user.
- Virtualization: It is a way of creating a virtual vision that is a virtual memory, virtual resources etc. Virtualization and cloud computing work together to provide best and un-interruptible services to the end users.

Cloud Computing provides services to the users broadly in three forms- Infrastructure as a Service (IaaS), Software as a Service (SaaS) and Platform as a Service (PaaS). Users can use these services according to their requirements. All the services are provided to the user and it works on the concept of pay per use[4].
1.1 Cost factor in Cloud Computing

As cloud computing technique frees the user from the overhead cost of hardware, but still some cost factors are always involved and these cost factors are comparatively very low as they are charged according to the services requested by the end user. For example, if a user requests for any task, then the cost is charged according to the resource required for accomplish on of the task, time of acquisition, turnaround time, I/O cost, the cost of resources etc. [5]. As each task is totally different from the other task so it is required to compute the cost of every individual task uniquely when it is requested. Different task results in the different cost factor.

2. Proposed Work

Many researchers have worked in the field of cloud computing to improve the cost factor but still, it has never been the main objective of the research. There are many parameters to be worked upon in order to minimize the cost and maximize the performance.

2.1 Activity Based Costing

Activity-based costing (ABC) is a way which measures the cost of the respective resources as well as the performance of the activities [6]. In cloud computing, each application which was requested by the user will be split into a number of tasks and each task is uniquely executed on the virtual machine, where each task gets its related resources virtually. These resources are distributed among various tasks which run on various VM’s virtually. Every task is independent of the working of another task, as every task requires different memory, different C.P.U time and might run on different VM. In order to measure the cost of the whole application, it is required to individually compute the cost of each task and then merge them [7]. The main objective of ABC is to manage the activities so that less time and fewer resources will be used. It also focuses on reducing the need for costly hardware.

2.2 Improved Work

In order to formulate the problem, two array lists need to be defined one for the independent tasks and other for computing resources.

- Task : Ti = {1,2,3,……..n}
- Resources : Rj = {1,2,3,……..n}

The main objective here is to minimize the cost and maximize the performance. The processing speed of resources is measured in million instruction per second (MIPS) and the size of the task is measured in term of million instruction (MI) [4]. As the resource, time of processing and memory needed by each task is different so as their priorities that’s why the tasks are needed to be divided into different VM’s list i.e.
high, medium and low according to the priority of the task.

$L_k$: Priority level of the $k$th task.

$R_{i,k}$: The $i$th individual use of the resource by the $k$th task.

$C_{i,k}$: The Cost of $i$th individual use of resources by the $k$th task.

$T_k$: Size of $k$th task.

$P_k$: Profit gained from task $k$th.

The priority of each task has been calculated by the given equation (1) and the number of resources used is from 1 to $n$ so that the priority level of any task can be calculated.

$$L_k = \sum_{i=0}^{n} \left( R_{i,k} \cdot C_{i,k} \cdot T_k \right) / P_k$$  \hspace{1cm} \text{(1)}$$

The number of the virtual machine are arranged in ascending order on the basis of equation (2) in three lists i.e. High, Medium and Low so that the tasks with the highest priority move to the VM list ‘High’.

$$VM = \frac{\text{Resource cost}}{\text{Resource processing power}}$$  \hspace{1cm} \text{(2)}$$

To calculate the turnaround time for each resource, the waiting time and execution time of each task has been analyzed.

$$\text{Turnaround time of each task} = \text{Waiting time} + \frac{\text{task length}}{\text{Resource Processing Power}}$$  \hspace{1cm} \text{(3)}$$

2.3 Improved ABC Algorithm

This Algorithm is about arranging the tasks with respect to their priority.

i. Each task is separately addressed by scheduler from all the available tasks.

ii. The priority level ($L_a$) of the tasks is calculated on the basis of equation (1).

iii. Sort the tasks according to their Priority level.

iv. The sorted tasks are stored on VM lists High, Medium and Low on the basis of equation (2).

v. On the arrival of new task again the priority is calculated and then it is put into the appropriate list.

The proposed algorithm will work for every single task which it received from the user.

Arrays has been used in between so that intermediate data can be stored and used efficiently. Here two arrays are used:

SET-PRI – This array is used to store the tasks according to their priority and is initialized with 0.

Task-Turn - This array is used to store the tasks in ascending order according to their turnaround time.

a) Input the task from the client

b) For available input task

Calculate the priority level

$$L_k = \sum_{i=0}^{n} \left( R_{i,k} \cdot C_{i,k} \cdot T_k \right) / P_k$$

c) Insert the task into an array SET-PRI according to its priority.

d) For every task in SET-PRI do

Sort them and put them in appropriate VM list in ascending order as

$$VM = \frac{\text{Resource cost}}{\text{Resource processing power}}$$

To calculate the turnaround time for each resource, the waiting time and execution time of each task has been analyzed.

$$\text{Turnaround time of each task} = \text{Waiting time} + \frac{\text{task length}}{\text{Resource Processing Power}}$$

Sort the task in ascending order on the basis of turnaround time.

Store them in array Task-Turn.

h) Until (Task-Turn|| $R_j$) == EMPTY)

Do for each task in Task-Turn

i. Process the task

ii. Select resource from resource list $R_j$

iii. Schedule the task with resource

iv. Update the task and resource status.

Done

e) End
3. Experimental Results

CloudSim simulator has been used for simulation. By using java language the complete coding has been implemented within the environment of cloudsim.

3.1 Experimental Parameters:

I. Configuration of Host:

Table 1: Configuration of Host

<table>
<thead>
<tr>
<th>RAM (MB)</th>
<th>1024</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processing Power (MIPS)</td>
<td>110000</td>
</tr>
<tr>
<td>VM Scheduling</td>
<td>Time Shared</td>
</tr>
</tbody>
</table>

II. Configuration of VMs:

Table 2: Configuration of VMs

<table>
<thead>
<tr>
<th>Virtual Machines</th>
<th>VM 1</th>
<th>VM 2</th>
<th>VM 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ram(MB)</td>
<td>5024</td>
<td>5024</td>
<td>5024</td>
</tr>
<tr>
<td>Processing Power(MIPS)</td>
<td>22000</td>
<td>11000</td>
<td>11000</td>
</tr>
<tr>
<td>Processing Element (CPU)</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

3.2 Comparative analysis

A comparative analysis has been done between the proposed algorithm and the ABC algorithm.

A. Processing Time / Execution Time:

Execution time in the simulation environment has been calculated and compared with ABC algorithm.

<table>
<thead>
<tr>
<th>No. of tasks</th>
<th>Execution time of Proposed Algorithm (mips)</th>
<th>Execution time of ABC algorithm (mips)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>2.3</td>
<td>2.6</td>
</tr>
<tr>
<td>50</td>
<td>55.45</td>
<td>60.76</td>
</tr>
<tr>
<td>100</td>
<td>94.76</td>
<td>110.32</td>
</tr>
<tr>
<td>200</td>
<td>174.31</td>
<td>197.54</td>
</tr>
<tr>
<td>500</td>
<td>431.57</td>
<td>476.89</td>
</tr>
</tbody>
</table>

Graphical Representation

Fig 2: Graphical representation of execution time between proposed algorithm and ABC algorithm

By doing the comparative analysis between ABC algorithm and proposed algorithm it has been observed that the execution time taken by the tasks in proposed algorithm is less than the time taken in ABC algorithm.

B. Cost Factor

Processing Cost for simulation environment in cloud computing is calculated and compared with ABC algorithm.
Table 4: Comparison table of Cost factor between ABC algorithm and proposed algorithm

<table>
<thead>
<tr>
<th>No. of cloudlets</th>
<th>Cost factor of Proposed algorithm (Rs)</th>
<th>Cost factor of ABC algorithm (Rs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>234.21</td>
<td>358.13</td>
</tr>
<tr>
<td>50</td>
<td>556.02</td>
<td>675.78</td>
</tr>
<tr>
<td>75</td>
<td>751.45</td>
<td>878.52</td>
</tr>
<tr>
<td>100</td>
<td>948.21</td>
<td>1053.41</td>
</tr>
<tr>
<td>200</td>
<td>1556.42</td>
<td>1806.82</td>
</tr>
</tbody>
</table>

Graphical Representation

4. Conclusion

In this paper, various parameters have been discussed which affect the performance in cloud computing. The main objective here was to maximize the performance and minimize the cost of the activity. Proposed algorithm gives more efficient performance with less cost factor as compared to the ABC algorithm. This work can be extended further by investigating about some other minute parameters in some other framework like Matlab, Simgrid etc.

References


