A New Task Scheduling Approach to Reduce The Execution Time Of The Task

Bhavika Mulia¹, Rikin Thakkar², Bhargav Modi³
¹Department of Computer Engineering, Silver Oak College of Engineering and Technology, Ahmadabad, India
²Asst. Prof. Computer Engineering, Silver Oak College of Engineering and Technology, Ahmadabad, India

Abstract—Cloud computing allows user to use different resources as a service over the internet on the bases of pay per use. Task scheduling and load balancing is the most important part in a cloud computing environment. To utilize all resources efficiently and reduce the task completion time, the task scheduling is used. Main use of task scheduling is to reduce the task execution time and many different algorithms of task scheduling have been proposed to reduce the execution time of task. Here we have proposed a new approach which will reduce the execution time of task. Our proposed task-scheduling algorithm uses a nonlinear programming model for divisible task scheduling, which assigns the correct number of tasks to each virtual machine. Based on the allocation, we design an algorithm for divisible load scheduling by considering the highest network bandwidth.

Index Terms—cloud computing, task scheduling task execution time, highest bandwidth

I. INTRODUCTION

A cloud computing is a type of Internet-based computing that provides shared computer processing resources and data to computers and other devices on demand. Cloud computing can be simply defined as computing services delivered to the user over the internet. Cloud computing is nothing but the accessing the pooled resources required for computing through your browser’s window [1]. The definition by the National Institute of Standards and Technology (NIST) “Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction”.

Scheduling can be described as the set of policies to control the order of work to be performed by a computer system. The main advantage of scheduling algorithm is to achieve a high performance computing and high system throughput. Scheduling manages availability of CPU memory and gives maximum utilization of resource. Scheduling of task is based on various parameters. A good scheduling algorithm should lead to better resource utilization and better system throughput. The main objective of the scheduling algorithms in cloud environment is to utilize the resources properly while managing the load between the resources so that to get the minimum execution time.

Task scheduling is the process of allocating the resources to the particular job in specific time. The main objective of scheduling is to maximize the resource utilization. Minimizing the waiting time is the goal of scheduling [1]. The objective of the scheduling algorithms in cloud computing environment is to utilize the resources properly while managing the load between the resources so that to get the minimum execution time.

Here system resources like CPU, memory and bandwidth are used by many users, so it is little difficult to construct an efficient task scheduling algorithm. The efficiency of the algorithm is effected by many things like the processor power, speed, space and memory [3].

Task scheduling is a fundamental issue in achieving high efficiency in cloud computing. However, it is a big challenge for efficient scheduling algorithm design and implementation (as general scheduling problem is NP-complete). Most existing task-scheduling methods of cloud computing only consider task resource requirements for CPU and
memory, without considering bandwidth requirements.

In order to obtain better performance, in this paper, we propose a algorithm for divisible task scheduling in cloud computing environments. A nonlinear programming model for the divisible task-scheduling problem under the bounded multi-port model is presented. The performance of algorithm is evaluated using CloudSim toolkit.

II. RELATED WORK

Nonetheless, these task-scheduling algorithms did not consider the impact of network bandwidth. However, tasks scheduled based only on the availability of CPU and memory resources may be delayed because of insufficient network bandwidth, resulting in waste of CPU and memory resources. With these task-scheduling algorithms, service providers of the cloud computing often have to upgrade hardware to meet the demand of network bandwidth. Li-der Chou [9] to improve energy efficiency, the dynamic power saving resource allocation (DPRA) mechanism based on a particle swarm optimization algorithm is proposed. The DPRA mechanism not only considers the energy consumption of physical machine (PM) and virtual machine (VM) but also newly tackles the energy efficiency ratio of air conditioner. Kaijun Ren [6] the pre-determined scheduling scheme can not perform as expected, causing that scientific workflows can not be completed within deadline or the total cost is far beyond users’ budget. To address this problem, we proposed a new dynamic resource allocation and task scheduling (DRATS) strategy. At build-time, we use Path cut(PC) algorithm to generate a static task-Virtual Machine(VM) mapping scheme, and use task duplication(TD) algorithm to reduce the affect caused by their potential uncertain task runtime. DRATS strategy can decrease the impact brought by uncertain task runtime, improve the probability of completing scientific workflows on time, and reduce the execution cost of scientific workflows effectively while satisfying the deadline constraint.

III. NON-LINEAR PROGRAMMING MODEL

Task scheduling and maintaining of resources are the main problems in cloud computing. Mainly the task scheduling effects the high performance of the system. To reduce the wastage of resource we are considering the network bandwidth while constructing the algorithm, here we are proposing a non-linear programming model for task scheduling. We propose a nonlinear programming for divisible task scheduling, which assigns the correct number of tasks to each virtual machine. Based on the allocation to allocate the proper number of tasks to each VM on the basis of its CPU power, memory power, memory space, and network bandwidth, affects the efficiency of task scheduling. We design an algorithm for divisible load scheduling by considering the highest network bandwidth.

Algorithm 1: Execution Time Reduction
Step 1: Create list of VMs.
Step 2: Assigning memory, RAM and Bandwidth to VMs
Step 3: Create list of Cloudlets
Step 4: Performing non linear programming model
Step 5: Find VM with highest Bandwidth
Step 6: Sufficient memory and RAM available
Step 7: If yes, then assign cloudlet to that VM and go to next cloudlet
Step 8: Else no, find another VM with highest bandwidth
Step 9: Executes the tasks
IV. IMPLEMENTATION AND RESULTS

Table 1. Experimental Specification

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plateform</td>
<td>Eclipse/</td>
</tr>
<tr>
<td>Code</td>
<td>JAVA</td>
</tr>
<tr>
<td>Operating System</td>
<td>OS X El Capitan</td>
</tr>
<tr>
<td>Version</td>
<td>10.11.1</td>
</tr>
<tr>
<td>RAM</td>
<td>8 GB</td>
</tr>
<tr>
<td>Processor</td>
<td>2.7 GHz intel core5</td>
</tr>
<tr>
<td>Graphics</td>
<td>Intel ins Graphics 6100 1536 MB</td>
</tr>
</tbody>
</table>

The simulation parameters are given in Table 1 and the description of the parameters used and the platform to execute the proposed plan and the hardware requirements used. We used Java language to perform this, operating system is OS X El Capitan, and the version used is 10.11.1.

Table 2. Experimental Results

<table>
<thead>
<tr>
<th>Cloudlets</th>
<th>CPU &amp; Memory</th>
<th>CPU, Memory &amp; Bandwidth</th>
<th>CPU, Memory &amp; Highest Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>4507 ms</td>
<td>3507 ms</td>
<td>3209 ms</td>
</tr>
<tr>
<td>1000</td>
<td>9047 ms</td>
<td>7007 ms</td>
<td>6327 ms</td>
</tr>
<tr>
<td>1500</td>
<td>13457 ms</td>
<td>10507 ms</td>
<td>9381 ms</td>
</tr>
<tr>
<td>2000</td>
<td>18087 ms</td>
<td>14007 ms</td>
<td>12765 ms</td>
</tr>
<tr>
<td>2500</td>
<td>22587 ms</td>
<td>17507 ms</td>
<td>15616 ms</td>
</tr>
</tbody>
</table>

The experimental results are given in Table 2 based on CPU & Memory, CPU, Memory & Bandwidth and CPU, Memory and highest bandwidth.

Fig 2: Showing experimental results considering without bandwidth, with bandwidth and with highest bandwidth

The experimental results are given in Fig 2 we take the cloudlets on x-axis and the number of time on y-axis. While considering without bandwidth, we observed that 500 cloudlets take 4507 milliseconds to execute that is bit longer time. While considering with bandwidth, we observed that 500 cloudlets take 3507 milliseconds to execute that is bit longer time. While considering with highest bandwidth, we observed that 500 cloudlets take 3209 milliseconds to execute that is bit longer time.
Our algorithm uses the parallel executing, task is stored in the queue and automatically executed, and it does not require waiting for previous task. It is validated that execution time is reduced as compare with bandwidth.

V. CONCLUSION AND FUTURE WORK

Task scheduling is the most important in cloud computing. Almost all algorithms consider RAM and memory as a parameter of task scheduling. But they don’t consider bandwidth as parameter. We can reduced the total execution time by reducing the transition time and we can reduced the transition time by allocating the task to VM having highest bandwidth. We have selected highest bandwidth as parameter of task scheduling and got better result than basic paper. we will focus on the dependency between the task and each task require more than one resource to complete the task.

REFERENCES


