An Optimized Algorithm for Task Scheduling based on Activity based Costing in Cloud Computing

Ashutosh Ingole
Dept.of Computer Engg.
SIT,Lonavala
Pune University,
Lonavala 410401,India

Sumit Chavan
Dept.of Computer Engg.
SIT,Lonavala
Pune University,
Lonavala 410401,India

Utkarsh Pawde
Dept.of Computer Engg.
SIT,Lonavala
Pune University,
Lonavala 410401,India

ABSTRACT

Large scale data processing is increasingly common in Cloud Computing systems. In these systems, files are split into many small blocks and all blocks are replicated over several servers. To process files efficiently, each job is divided into many tasks and each task is allocated to a server to deal with a file block because network bandwidth is a scarce resource. In Cloud Computing, traditional way for task scheduling cannot measure the cost of cloud resources accurately by reason that each of the tasks on cloud systems is totally different between each other. There may be no relationship between the overhead application base and the way that different tasks cause overhead costs of resources in cloud systems. The traditional way for task scheduling cannot meet the cloud market well enough. An optimized algorithm for task scheduling based on ABC (Activity Based Costing) in Cloud Computing reduces the total time required to schedule the task.

General Terms

Task scheduling, optimized algorithm, activity based costing, cloud computing.

Keywords

Cloud computing; task scheduling; activity based costing; optimized algorithm.

1. INTRODUCTION

Popularity and usage of internet is expanding like a forest fire nowadays. We use internet daily for Different purposes but never think about the vast number of services provided by different service providers. How could they manage the Juggernaut of data? Where do they store and how they control? The answer is Cloud Computing.

Cloud Computing is the technology specially builds to manage such kind of data and provide it whenever needed. Most of the data storages are servers in themselves. They provide services, manipulate data, and share with the other servers. Now if we find a perfect way to do all these activities among the servers so that it will take less time, resources, then it will be far beneficial for any private or public organization. It will save time as well as reduce the need of costly hardware. With more cloud platforms and more Saas (Software as a service) applications which run entirely in them, how to benefit more from them with existent systems is considered to be an important question.

According to the current statistical data, the average service rate of resource in some systems is just 30%, while some other companies or organizations in the opposite side which need to buy a lot of expensive hardware for the ever growing computing tasks. Hence it is necessary to ensure that the resources are used in the most beneficial manner.

Traditionally, task scheduling in cloud computing totally depends on user’s requirements. The ABC (Activity Based Costing) actually works on an optimized way of resource allocation in cloud computing. This paper will focus on an optimized activity based costing algorithm in order to get an optimized solution for every single user requirement in order to profit more than those in traditional ones.

First we will see in short how the traditional system works and where it is lagging behind. Then we will see how we can overcome these deficiencies.

2. TASK SCHEDULING IN TRADITIONAL SYSTEM

Before understanding how task scheduling is done in traditional system, first we need to understand the architecture of cloud computing. Then we will introduce the traditional way of task scheduling.

2.1 Architecture of Cloud Computing

Cloud computing is nothing but the way in which an infrastructure deploys and delivers applications. Cloud computing basically includes data centers and application delivery networks. A request from client reaches to the infrastructure and through application delivery network it is send to the cloud data center locations.

Now at this point there are hundreds and thousands of client requests are approaching. To tackle with such situation these data centers are always interconnected and uses the technique called virtualization. They create the number of replicas of original data in order to provide the service to maximum number of clients which will improve the overall efficiency of system. They make user to work on these virtual platforms. Because the resources are limited and the need is growing day by day.

Now one thing should be noted that the infrastructure and architecture of cloud computing system, changes according to the purpose for what it is established. But the basic idea behind every system remains same which is shown in fig.1 below.

2.2 Disadvantage of task scheduling in traditional Cloud Computing

As we have seen earlier that the task scheduling in traditional system is based on the task. It does not have any direct relationship with resources it will need. This may lead to over-costing and over-priced for some simple task.

To compete in the marketplace, some companies have had to cut the prices of high volume popular items. But they have been able to achieve high mark-ups on low volume specialty tasks. Price cuts produce increased sales in units but decreased total revenue and high mark-ups on specialty tasks didn’t offset the fall in profit margins of high volume ones [4].
Here are some causes of the problems: increasing demand of resources, cost variations, increasing complexity of infrastructure, variations in product demands and along with all this increasing maintenance.

3. RELATED WORK

Now the focus of this paper is to develop an optimized algorithm which will help to optimize all the performance parameters of the system. When we will consider all the parameters and try to optimize their performance for the system and the user, ultimately the performance of the system will improve.

4. ACTIVITY BASED COSTING AS A SOLUTION

Activity based costing is evaluated separately for every task. It is decided on the basis of resources, space and time taken by every activity of every task. There are other performance parameters also which are to be considered, we will see them in next subtitle. Now there will be some cost drivers which will decide the average cost of activities and hence tasks, which will lead us to form a basic tree structure. And this tree structure will guide us about allocation of space, time and resources to activities.

4.1 Activity based Costing in Cloud Computing

Activity based costing is the way to measure both, cost of the object and its performance. It will help to solve the problem like poor cost control, distorted product costs and also the starvation.

Activities will be performed on virtual operating systems and the resources are provided over these virtual systems by the original system. There might be tasks which are totally independent or dependent on the other task. There might be some tasks whose all required resources are not available on any single data center. So after considering all possibilities, we divide the task into different groups. These groups are, a) Available b) Partially available.

- Available: Dependent and Independent.
- Partially Available: cat1, cat2, cat3…….catN.

This grouping is done on the basis of resources. Available is the group of tasks which can be complete performed on a single data center. And partially available is the group of tasks which will require resources from other data centers. Again independent is the group of tasks which are independent on the other task’s result. And hence dependent require some results from previous.

Now the partially available is the group of tasks which needed data from different data centers. Hence further we have sorted them in different categories, cat1, cat2, cat3… and so on till N number of categories. These categories are done on the basis of data need. Cat1 tasks will need the data from same data centers. Similarly cat2 tasks will need the data from same data centers and so on.

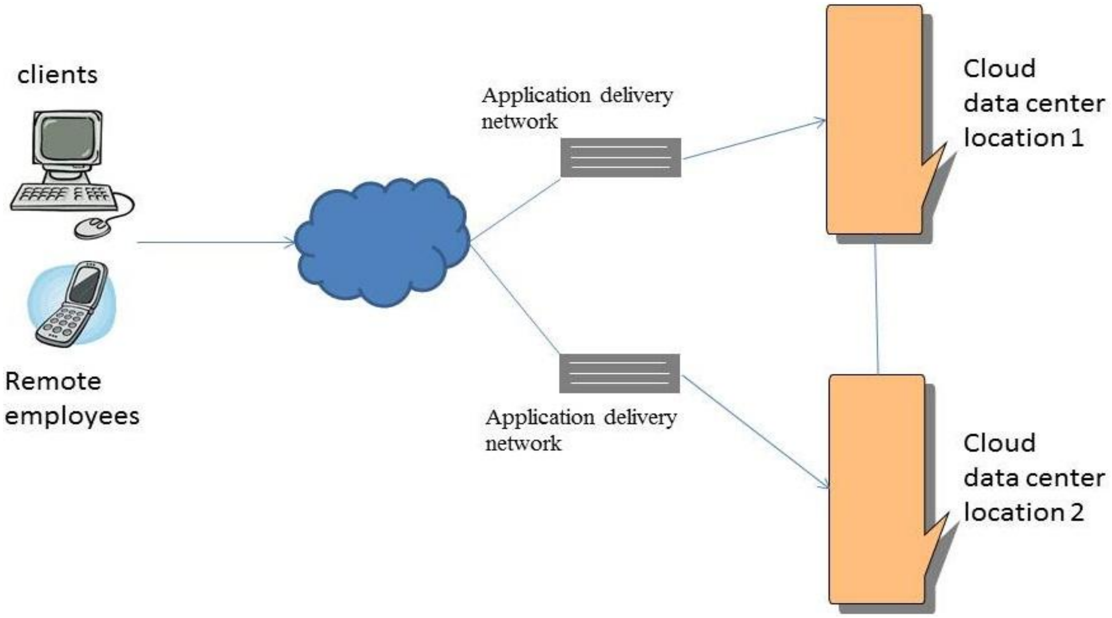

Fig.1 Architecture of Cloud Computing

4.2 Implementing Activity based Costing in Cloud Computing

Implementation of activity based costing algorithm can be understood by expressing it in a tree structure. The tree structure is shown in fig.2.

There will be one parent queue in which we will store the tasks according to their arrival time, i.e. first in first out approach. Then we will check for data and requested resources by the task and sort them into two different queues, available and partially available. Again in available queue we will check if the task is dependent or independent and according to that we will store them in their respective queues. Now in partially available queue we are having tasks which will need data resources from other data centers. Then we sort them in different queues named cat1, cat2… and so on. This will be done on the basis of resources they need. For example if task1 need resources from data center location1 and data center location3 and similar for task2 and task4. Then tasks 1, 2, 4 will be in one category say cat1. Similarly for other tasks we will make cat2, cat3 and so on till we finish the tasks in the partially available queue.

Now we have major queues as, independent, dependent, and cat1, cat2…..catN. For every major queue we will again make three different queues based on priority, High, Mid, Low. Now the question is how to decide the priority?

To decide the priority we will consider four major factors viz. time, space, resources and profit. So we have derived the following formula which will decide the priority of the task.

$$K_i = \sum_{j=0}^{n}(T_{i,j} + S_{i,j} + C_{i,j}) \ /P_i$$

The above notations are explained below:

K_i : priority of the ith task
$T_{i,j}$: time required to complete jth activity of ith task
$S_{i,j}$: space needed to operate jth activity of ith task
$C_{i,j}$: cost of jth activity in terms of resources of ith task
P_i : profit from complete ith task
n : it is the total number of activities of any ith task

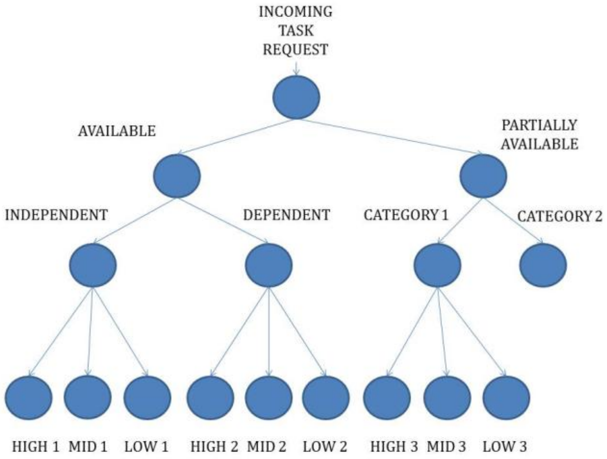

Fig.2 Task Scheduling Tree Structure

While calculating priorities of the task, we will need to check newly calculated priority with the previous ones. Then according to that we will place them into priority queues.

When we will finish placing tasks in their queue then we can select task from High priority queue and when this task will be finished, the first task of the Mid priority queue will be shifted to High priority queue. And in this way tasks from all the queues will start executing.

One question still remain there that what will be the sequence of task selection, as there will be more than one High priority queues. Now we will use a simple logic, as below:

- Compare priority of all tasks which are on number one of every High priority queue.
- Then select the highest priority and assign resources and space.
- Then check if resources are remaining then again choose the next task with the same strategy and allocate the resources and space.
- Repeat the above procedure till the end of all resources.
- When all resources get allocated then waits for any task to finish and as soon as any task finishes then again choose next one and allocate the resources.

When all resources get allocated then waits for any task to finish and as soon as any task finishes then again choose next one and allocate the resources.

4.3 Algorithm for Activity Based Costing in Cloud Computing

```
for all tasks do
        store in a queue parent
end for
for every ith task do
        check if all the resources available or not
        if yes
        move to queue available
        else
        move to queue p_available
end for
for all tasks at queue available do
        check if dependent
        if yes
        move to queue dependent
        else move to queue independent
end for
for all tasks at queue p_available do
        sort tasks into queues cat1, cat2, cat3……catN
end for
for all tasks at all queues do
        calculate priority $K_i$
end for
for every $K_i$ do
        put tasks in appropriate queues of priority, High,
        Mid, Low
end for
compare High1j,High2j,High3j,……High(2+N)j
        select task with highest priority for execution
while system is running task do
```

```
        check if new task is available
        if yes
        calculate priority and place at appropriate queue
        else continue
        end if

        scan queues to modify priority
        if queues are not empty
        select new task of highest priority
        else
        wait for new task to arrive
end while
```

5. ACKNOWLEDGEMENT

We sincerely acknowledge with due courtesy the help and inspiration given to us by our internal guide **Prof. R.B. Singh.** We are highly obliged and resound to express our admiration to the best conscious to convey respect toward **Prof. A. Lamgunde, H.O.D.** (Computer Engineering, SIT).

We also thank to Prof. V. Dhawas mam for her kind and rightful guidance regarding the subject. And we are grateful toward all the Authors for their papers which we have studied to complete this paper.

6. REFERENCES

[1] Sandeep Tayal, University School of Information Technology, Guru Gobind Singh, Indraprastha University, Delhi- 10006,India "Tasks Scheduling optimization for the Cloud Computing System2011, IJAEST,2011

[2] Archana Ganapathi, Yanpei Chen, Armando Fox, Randy Katz, David Patterson Computer Science Division, University of California at Berkeley," Statistics-Driven Workload Modeling for the Cloud", IEEE 2011

[3] QI CAO, ZHI-BO WEI, WEN-MAO GONG International School of Software Wuhan University Wuhan,China,"An Optimized Algorithm for Task Scheduling Based On Activity Based Costing in Cloud Computing",IEEE 2009

[4] Jinhua Hu, Jianhua Gu, Guofei Sun, Tianhai Zhao, NPU HPC Center Xi'an, China "A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment", IEEE 2010

[5] Jiahui Jin, Junzhou Luo, Aibo Song, Fang Dong and Runqun Xiong School of Computer Science and Engineering, Southeast University Nanjing, P.R. China "BAR: An Efficient Data Locality Driven Task Scheduling Algorithm for Cloud Computing" ,IEEE 2011

[6] GAN Guo-ning, HUANG Ting-Iei, GAO Shuai School of Computer science and engineering Guilin University of Electronic Technology Guilin, China "Genetic Simulated Annealing Algorithm for Task Scheduling based on Cloud Computing Environment",IEEE 2010

[7] Suraj Pandey, Department of Computer Science and Software Engineering, the University of Melbourne, Australia, "Scheduling and Management of Data Intensive Application Workflows in Grid and Cloud Computing Environments", Dec 2010