Proceedings of
National Conference on Innovative Trends in Computer Science Engineering (ITCSE-2015)
held at BRCMCET , Bahal on 4th April 2015

An Algorithm for Task Scheduling in Cloud Computing Environment

[1]Sukhdev Singh, [2]Pooja Anand, [3]Gurpreet Kaur
[1,2,3]D.A.V. College(Lahore) , Ambala City
Email: [1]sukhdev_kuk@rediffmail.com

Abstract: Cloud computing is based on the concepts of distributed computing, grid computing, utility computing and virtualization. It is a virtual pool of resources which are provided to users via Internet. It gives users virtually unlimited pay-per-use computing resources without the burden of managing the underlying infrastructure. Cloud computing service providers' one of the goals is to use the resources efficiently and gain maximum profit. This leads to task scheduling as a core and challenging issue in cloud computing. In this paper we introduce an optimized algorithm which addresses the major difficulties of task scheduling in cloud. Incoming tasks are grouped on the basis of their priority and the requested resources and the resource selection is done on its turnaround time and cost. A priority based formula is used for task selection. By doing this we can get the better results over the traditional scheduling.
Keywords: Cloud computing, Scheduling, Traditional Scheduling, Proposed Algorithm

1. INTRODUCTION

Cloud computing is the new concept in information technology. Basically it is the enhancement of already existing technologies like grid computing, parallel computing and the distributed computing. It is the use of computing resources like hardware and the software that are delivered as a service over the network. A way to provide the applications of computer to the users without the need to purchase, installs on their pc. It deals with three types of services: Infrastructure as a Service(IaaS),Platform as a Sevice(Paas),Software as a Service(Saas)[1].All these services are provided to the users in a pay-per-use fee, users access the shared resources, data storage and so on through the internet. All this is done virtually. [2].

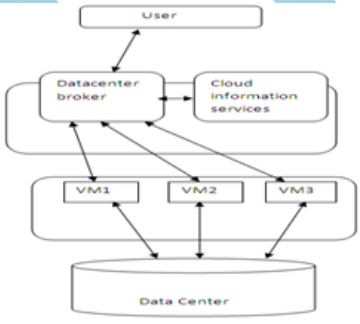

Figure 1: Virtualization feature of cloud computing.

Virtualization is the main feature of cloud computing. Scheduling is the process of mapping tasks to the available resources on the basis of task requirements. there are number of parameters that need to be considered for scheduling. Scheduling process can be classified in three stages:
1 Validation and filtering-To discover the resource that presents in the network and filter the resource by collecting the status information.
2 Selection- Resource selections is done based on the parameters of task and resource.
3 Task submission-Tasks is submitted to the required resource

II. TASK SCHEDULING IN CLOUD COMPUTING

It uses the direct tasks of users as the overhead application base. The problem is that there may be no relationship between the overhead application base and the way that different tasks cause overhead costs of resources in cloud systems[3,4]. For large number of simple tasks this increases the cost and the cost is decreased if we have small number of complex tasks.

III. PROPOSED ALGORITHM

We proposed more flexible algorithm for scheduling in order to profit more as compared with traditional ones. We try to achieve better balancing among all the nodes in the cloud environment by calculating the task time given to certain processor and make a better decision over the entire group of tasks. Cloud computing systems fundamentally provide access to large amounts of data and computational resources that can be acquired and released on demand.

A) Formulation of problem

To formulate the problem for n independent tasks and m computing resources with the objective of minimization of cost and minimization of completion of the time. Let there be three lists of tasks with high, medium and low priorities. For the computation of tasks the system can take from high priority list first, then medium and then low priority list. The priority level of each task can be calculate as in formula (1), the total individual resources use is supposed to be n, so the priority level of the k_{th} task is

$$L_k = \sum R_{i,k} \times C_{i,k} / P_k \quad \text{.....(1)[5]}$$

Parameters are defined as followed:

- $R_{i,k}$: The i^{th} individual use of resources by the k^{th} task.
- $C_{i,k}$: The cost of the i^{th} individual use of resources by the k^{th} task.
- P_k: The profit earned from the k^{th} task.
- L_k: The priority level of the k^{th} task.

In cloud computing, each application of users will run on a virtual operation system, the cloud systems distributed resources among these virtual operation systems. Every application is completely different and is independent and has no link between each other whatsoever, for example, some require more CPU time to compute complex task, and some others may need more memory to store data, etc. Resources are sacrificed on activities performed on each individual unit of service. In order to measure direct costs of applications, every individual use of resources (like CPU cost, memory cost, I/O cost, etc.) must be measured. When

the direct data of each individual resources cost has been measured, more accurate cost and profit analysis based on it than those of the traditional way can be got. Cost of every individual resources use is different. The priority level can be sorted by the ratio of task's cost to its profit. For easy management, three lists can be built for the sorted task; each list has a label of priority level such as high, medium and low. Cloud systems can take someone out from the highest priority list to compute. Maps should be scanned every turn to modify the priority level of each task. The scheduler accept the number of tasks, average MI of tasks, resources are selected. The priority level of each task is calculated using equation (1).

B) Algorithm to arrange the tasks according to priority:

Step 1: Scheduler receives the task.
Step 2: for all the tasks that are available
Calculate the priority levels L_k of tasks
End for
Step 3: Repeat for every L_k do
Sort the tasks and then put into an appropriate list according to their priority level.
End for
Step 4: Repeat while the system is running do
If there is new task do
Calculate priority and put it into an appropriate list.
End if
End while

C)Algorithm for task grouping and scheduling

Step 1: Number of tasks n to be scheduled and the number of resources m are received by the scheduler.It receives the resource list.
Step 2: All the available tasks are submitted to the scheduler.
Step 3: Set tot_len=0, Set resource_id j=1 and index i=1
Step 4: Get the MIPS of resource j
Step 5:Multiply MIPS of the j^{th} resource with the granularity size (given by the user)
Step 6: Get the length (MI) of the task from the list
Step 7: If the resource MIPS is less than task length then
7.1 The task can-not allocated to the resource
7.2 get the MIPS of the next resource
7.3 go to step 5
Step 8: If the resource MIPS is greater than task length then Repeat while (tot_len <=resource MIPS) and there exist ungrouped tasks in the list
8.1 tot_clen=previous total length + current task length
8.2 get the length of next task
If the total length is greater then resource MIPS then
Subtract length of the last task from the tot_clen
End while
Step 9: If tot_clen is not zero repeat 9.1 to 9.4 then
9.1 create a new task group of the length equal to the tot_clen
9.2 assign a unique id to newly created task group
9.3 insert the task group in to a new task group list GJ_k
9.4 insert the allocated resource_id into the target resource list targetR_k
Step 10: set tot_len=0
Step 11: get the MIPS of next resource
Step12: multiply the MIPS with the granularity size specified by the user
Step13: get the length (MI) of the task
Step 14: go to step 7
Step 15: repeat above steps until the tasks in the list are grouped into the task group
Step 16: when all the tasks are grouped,assigned to a resource, send all the task group to their GJ_k.
Step 17: After the execution of the tasks groups by the assigned resources send them back to the target resource list target R_k

D)Profit based scheduling algorithm

Step 1: Run the algorithm to arranging task to their priority levels (3.1.1)
Step 2: all the lists are processed
Step 3: Run job grouping algorithm to schedule the tasks (3.1.2)

IV. SIMULATION OF RESULTS

To simulate the algorithm of task scheduling described previously.CloudSim3.0.3 provides a number of function for the starting and simulation of heterogeneous distributed computing environment, particularly suitable for simulation and research of task scheduling on cloud.

Resource	MIPS
R1	120
R2	131
R3	153
R4	296
R5	126
R6	210

Table 1: Mips of cloud resources

We simulated the algorithm with

- ✓ six nodes
- ✓ five seconds of granularity time
- ✓ Average MI of tasks 10.

Tabulation of results in Table 1 and Table 2 below.

Process Time in Seconds

No of Cloudlets	Proposed Profit Based Algorithm	Sequential Algorithm
25	152.23	159.1
50	272.34	294.01
75	422.4	460.03
100	532.12	657.5

TABLE 2 Simulation of processing time profit based Task Scheduling cloud computing

Process Cost in Rs.

No of Cloudlets	Proposed Profit Based Algorithm	Sequential Algorithm
25	324.21	453.31
50	745.02	875.61
75	881.45	978.61
100	1034.41	1178.31

Table 3 Simulation of processing cost for profit based task scheduling in cloud computing

V.CONCLUSION

In this paper we discusses about Task scheduling in cloud computing environment. By doing research and analysis of problem, that aims at task scheduling with minimum total tasks completion time and minimum cost. It explores methods of scheduling done in cloud computing.CloudSim3.0.3 is employed to carry out and simulate the tasks assignment algorithm, and distributed task scheduling. In future, the more accurate comparative data will be measured based on Sim Grid, and more concrete algorithm will be investigated. Profit based task scheduling algorithm only takes the initial research on task scheduling in Cloud platform. However many issues remain open. Further improvement should be done to handle more complicated scenario involving dynamic factors like as dynamically changing cloud environment and other QoS attributes. The improvement of this algorithm should concentrate on discussing simultaneous instead of independent task scheduling in Cloud environment.

REFERENCES

[1] K. Keahey and T. Freeman, "Science Clouds: Early Experiences in Cloud Computing for Scientific Applications," in proceedings of Cloud Computing and Its Applications 2008, Chicago, IL. 2008.

[2] S. Singh and K. Kant, "Greedy grid scheduling algorithm in dynamic job submission environment," in International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT), 2011, pp. 933-936.

[3] YM Babad "Cost driver optimization in Activity based costing",review 1993.

[4] JA Brimson,Activity Accounting: An Activity-based Costing Apporoach,john Wiley and Sons,1991 5. Monika Choudhary,