ISSN:0975-9646

An Efficient Fully Dynamic Algorithm to Optimize the Task Scheduling Activities in Cloud Systems

Sarwat Jahan[1], Safdar Tanweer[2], Masoorul Haque[3]

[1]*Research Scholar, M. Tech. (CS), Jamia Hamdard, New Delhi, India*
[2]*Assistant Professor, Dept. of Computer Science, Jamia Hamdard, New Delhi, India*
[3]*Principal System Analyst, National Informatics Centre, Govt. of India, New Delhi, India*

***Abstract*— The model of cloud computing has been evolved as a very popular and interesting model that concentrates on the on demand services very efficiently whenever they are needed by cloud terminals. In a quite different computing style, the delivery of massively scalable resources to the end users/clients/customers is carried out 'as a service' within the cloud systems using internet as a communication channel. In the cloud systems, the selection of the best suitable resources for the execution of arriving tasks is decided by the used scheduling strategies after considering some static and dynamic behaviour and the restrictions applied on them. From the users' point of view, some other issues like task execution cost or task completion time may become essential parameters for deciding the scheduling algorithm efficient. The service providers always try to provide the resources in optimized manner so that the utilization of resources may be optimum and the resource potential might be left minimum. This paper proposes an efficient scheduling algorithm by which these big challenges may be addressed efficiently in the cloud systems. The arriving tasks would be bound after measuring them on the basis of their requirement like minimum task execution cost or minimum task completion time and their priority levels. The selection of the resources would be made on the basis of greedy approach based task constraints. This proposed work has been implemented, tested, validated and verified on a simulator. Results Show the effectiveness and correctness of the proposed framework and come out with a significant improvement over sequential scheduling algorithms.**

***Keywords*— Cloud Computing, Task Scheduling, Optimal Scheduling, Sequential Scheduling, Dynamic task execution.**

I. INTRODUCTION

Cloud computing is one of the most current emerging computing technologies and has received a lot of interest by the researchers now a days. It can be explained as on demand pay-per-use model which provide the software, information and resources in a shared mode as per the users'/clients' requirements when the need for it [1]. It is evident that the human dependency has been focused on the cloud computing for the last decade. The most popular online gaming sites, document sharing, social networking, email hosting and business sites are moving to cloud environment from their traditional computing environment. Google, IBM, Microsoft, Yahoo, Apple and Amazon are the famous initiators in this field.

The theory of scheduling is taking a great attention of researchers with increase in its popularity. Basically the term scheduling can be defined in the context of cloud computing as 'It is the appropriate mapping process in which the arriving tasks are mapped to the available cloud resources on the basis of the requirements and characteristics associated with the tasks.' The scheduling is the very important aspect for the appropriate, efficient and effective working within the cloud systems. The various task attributes/parameters need to be considered while designing efficient and optimized scheduling. The various cloud resources must be fully optimized by used scheduling algorithm without tempering the pre-defined service guidelines/parameters of existing cloud system.

Generally, the process of scheduling is carried out within the cloud systems into three stages as shown in Figure 1:-

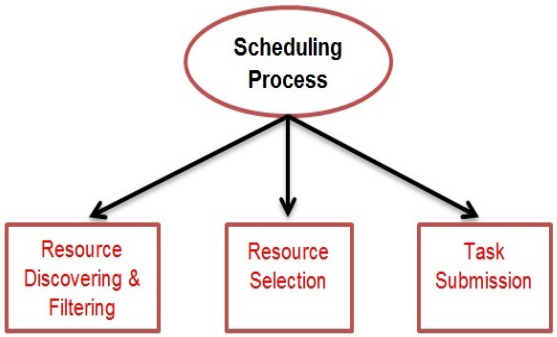

Fig 1: Scheduling Process

- Resource discovering and filtering– The discovering and filtering of the resources is the responsibility of Datacenter Broker. Datacenter Broker also collects and updates the information related to the filtering and resource discovery.
- Resource selection– The selection of the target resources is finalized on the basis of certain task parameters of the arriving tasks.
- Task submission- In this stage, the tasks are submitted to selected resources.

The above mentioned scheduling steps are further

described in Figure 2:-

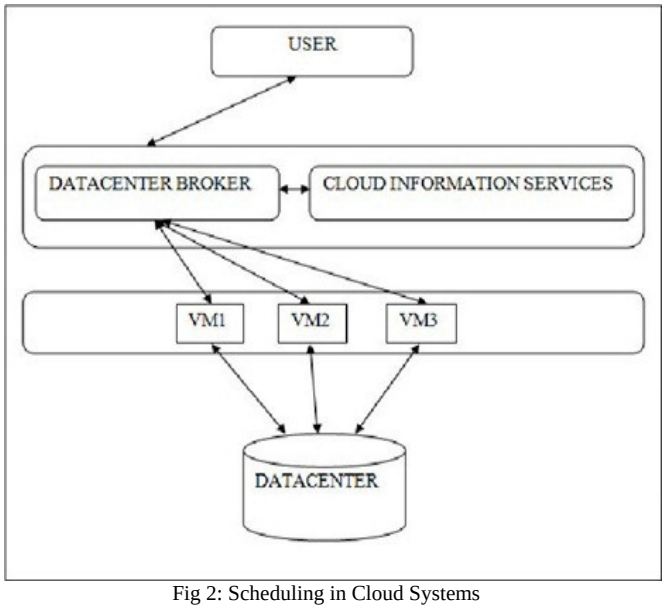

Fig 2: Scheduling in Cloud Systems

II. RELATED WORKS

There are various ways by which the available resources can be selected in cloud systems. The selection of available cloud resources can be either greedy, round robin, random or by any other means. The selection of arriving tasks which are to be scheduled for the execution, can be done on the basis of SJF (shortest job first), FCFS (first come first served), coarse grained or priority based task grouping strategies etc. The work of scheduling algorithm is to select the executable jobs and the corresponding available cloud resources where the execution of the tasks will take place. Each selection strategy has some advantageous aspects which reduce the short-comings of previous strategies and make new selection strategy better than previous ones.

If we talk about the benefits of existing scheduling strategies than we can say that these are beneficial either to clients/users/customers or to the service providing bodies. No algorithm is there which takes care of both together. Each strategy has its own plus points and negative points. For example, the user feels happy with priority based and greedy strategies. If the better utilization of cloud resources becomes prime parameter than grouping based strategy is a good option. The 'long waiting time' is the biggest short-coming of priority based scheduling strategies for the tasks having low priority. From the service provider bodies' perspective, the QoS parameters are not fulfilled whereas from the users'/clients' perspective, the cloud resources get wasted while the greedy strategies are considered into the cloud systems. Similarly some disadvantages such as considerable task completion time are also associated with task grouping strategy due to various formations of groups. Thus it is obvious from the study that there are some strategies which work either for the welfare of service provider bodies or for the welfare of cloud users/clients/customers. Hence there is an essential need to provide an efficient optimized scheduling algorithm which looks for the welfare of both ends i.e. for service provider bodies and for users/clients/customers.

III. PROPOSED FRAMEWORK

A. *Task Grouping*

The term 'grouping' refers a collection of various components based on homogeneous behaviour or attributes. In the cloud scenario, the term 'grouping' refers the similar type of arriving tasks under one head that are to be scheduled, operated and executed under the same computing criteria [2]. It can be said that the grouping is a behaviour which helps to make a bunch of similar type of tasks by considering homogeneous attributes. This proposed framework groups the arriving tasks on the basis of associated constraints such as minimum cost or deadline. After the grouping, the tasks can be measured for their priority levels and further scheduled as per priority levels.

B. *Prioritization*

The importance factor of the associated element/component is decided by its priority level. In the case of task scheduling the priority level of the task tells the execution order of that task based on certain parameters undertaken about its processing [3]. In this framework, the tasks which are based on deadline are prioritized on the basis of given deadline. First of all the shorter dead lined tasks are taken into account so they are assigned higher priority level in scheduling sequence. The task-list is continuously updated in ascending order of deadline for their efficient execution by considering the minimum time constraint. The prioritization of cost based tasks is assigned on the basis of their associated profit in descending order. If the tasks having higher profit, execute on the machines having minimum cost, than they may produce optimum profit as result.

C. *Greedy Allocation*

Greedy algorithms are useful for such resource environments which are heterogeneous and dynamic in nature and are connected to the scheduler using homogeneous communication medium [4]. The greedy algorithm is defined as follows:-

"*A greedy algorithm always makes the choice that looks best at that moment. That is, it makes a locally optimal choice in the hope that this choice will lead to a globally optimal solution*" [5].

D. *Deadline Constrained Based*

The greedy algorithm is used to improve the task completion time, targeting to minimize the turning around of individual tasks so that the overall improvement in completion time may be achieved.

$$\text{Turnaround Time} = \text{Resource Waiting Time} + \text{Task Length / Proc. Power of Resource}$$

Fig 3: Calculation of Turnaround Time

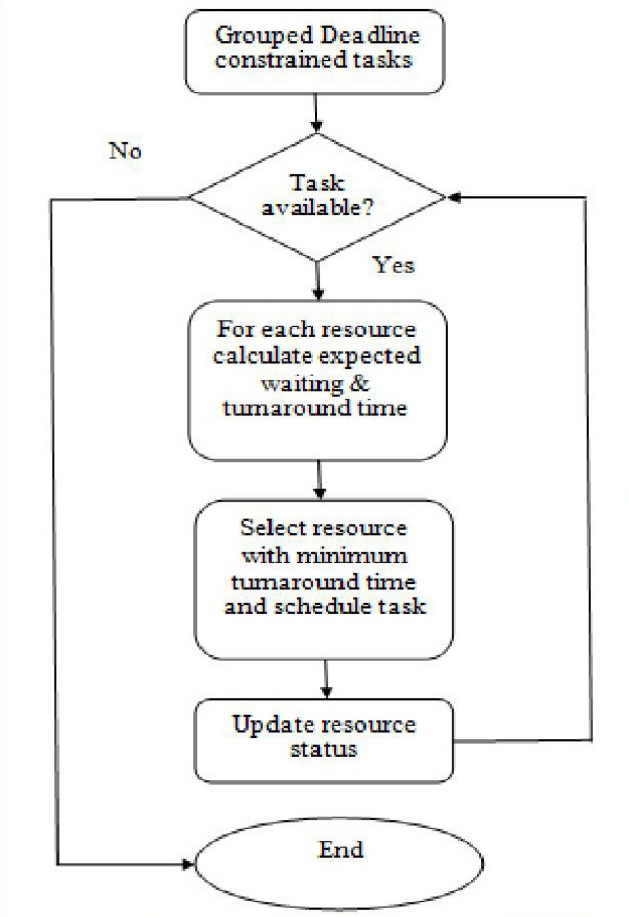

Fig 4: Scheduling of Deadline Constrained Tasks

The tasks are executed on those resources which have minimum turnaround time after calculating the individual turnaround time for each cloud resource. This is the responsibility of scheduler to select the best fit cloud resource to minimize the turnaround time. The expected completion time of a task, forecasts about the involved turnaround time. When the scheduler fills a task on the cloud machine, the cloud resource remains involved in that task for some amount of time to do its processing. After completing the execution of task, the status of cloud resource gets updated and the resource becomes available to execute new tasks.

E. Minimum Cost Based

The tasks are scheduled continuously on that resource which has the minimum cost until the capacity of that resource gets over. After executing each task, the status of cloud resource is continuously updated.

$$\text{Cost of Task} = \frac{\text{Task length / Proc. Power of Resource}}{*} \quad \text{Resource Cost}$$

Fig 5: Calculation of Task Cost

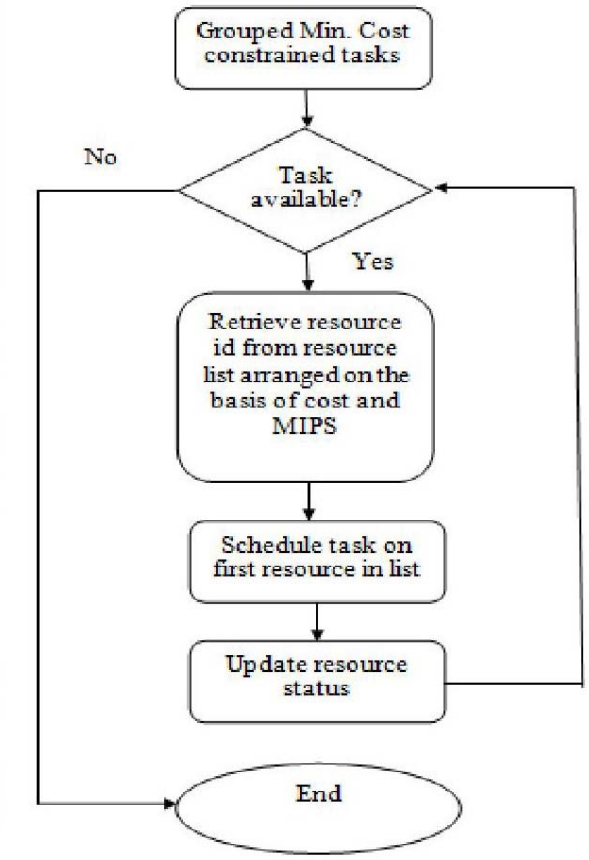

Fig 6: Scheduling of Cost Based Tasks

IV. PROPOSED ALGORITHM

In this section, a scheduling algorithm which gives the optimum results is proposed and implemented, which works as follows:-

1. The arriving tasks are distinguished on the basis of their type- low cost requirement or deadline constrained.
2. After the completion of their first grouping, the priority levels are assigned to tasks on the basis of their deadlines or profits. It is essential because the shorter dead lined tasks are to be scheduled first and similarly the most profitable tasks are to be scheduled on the machines having lower cost.
3. (a) In deadline constrained group:-
   For each prioritized task –
   i) Turnaround time at each resource is calculated taking following parameters into account.
      - Waiting time
      - Task length
      - Processing Power of virtual machine

   ii) The virtual machine with minimum turnaround time that is capable to execute the task is selected and task is scheduled for execution on that machine.

   iii) Waiting time and resource capacity of selected machine are updated accordingly.

Fig 7: Steps in Deadline Constrained Grouping

4. (b). In cost based group:-
For each prioritized task –

i) Virtual Machine are selected on the basis of processing power of machine and its cost

ii) For each virtual machine cloudlets from the group are scheduled till the resource capacity is permitted.

iii) Resource capacity and waiting time are updated accordingly.

Fig 8: Steps in Cost Based Grouping

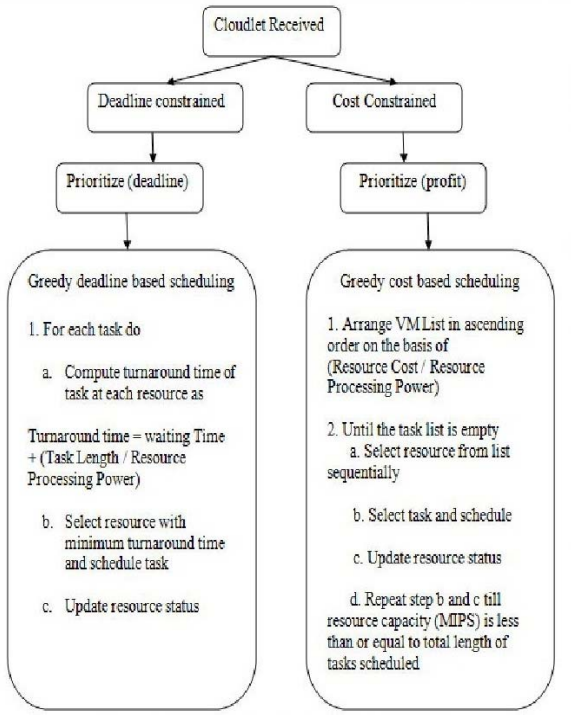

Fig 9: Proposed Algorithm

V. Simulation Results

The 'CloudSim 2.1.1' simulator has been used to simulate this work in the heterogeneous resource and communication environment [6] & [7]. This simulator also verifies the correctness and efficiency of proposed work by performing the experiments with simulator's default 'sequential assignment' and the 'proposed work'. The arrival of jobs is distributed randomly and uniformly to generalize the cloud scenario.

In the following figure, the configuration of cloud datacenter is shown:-

No. of PE (Processing Elements)-1

No. of Hosts-2

RAM(MB)	10240
Processing Power(MIPS)	110000
VM Scheduling	Time Shared

Fig 10: Configuration of Hosts

The VM-Configuration used in this proposed work, is shown below:-

Virtual Machines	VM 1	VM2
RAM(MB)	5024	5024
Processing Power(MIPS)	22000	11000
Processing Element(CPU)	1	1

Fig 11: Configuration of VMs

Cost Performance: -

The comparison of cost performance between sequential algorithm and proposed algorithm is shown in following figure which proves the improvement of proposed work:-

No. Of Cloudlets	Proposed Algorithm	Sequential Assignment
25	565.91	735.68
50	1131.82	1471.36
75	1697.73	2207.05
100	2263.6	2942.73

Fig 12: Comparison of Execution Cost

The analysis of execution cost of both algorithms is shown in following graph:-

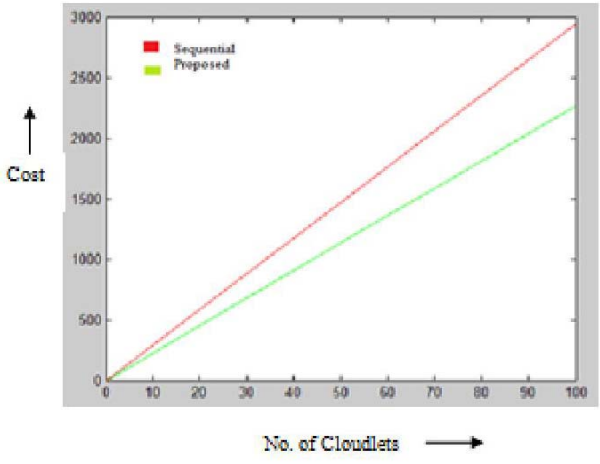

Fig 13: Analysis of Execution Cost

Time Performance: -

The comparison of time performance between sequential algorithm and proposed algorithm is shown in following figure which proves the improvement of proposed work:-

Cloudlets	Proposed Algo	Sequential Algo
25	565.91	735.68
50	1131.82	1471.36
75	1697.73	2207.05
100	2263.6	2942.73
125	910.04	997.99
150	1298.50	1439.75

Fig 14: Comparison of Task Completion Time

The analysis of execution time of both algorithms is shown in following graph:-

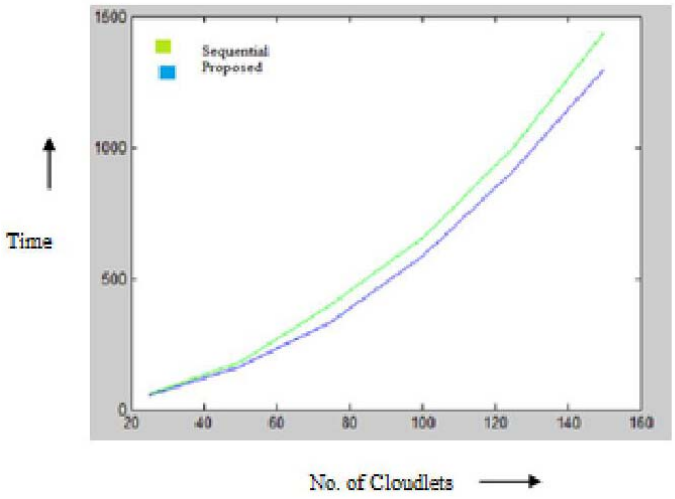

Fig 15: Analysis of Task Completion Time

.

VI. Conclusion And Future Work

It has been clearly observed that the cost and completion time of arriving tasks is improved and too much better by using proposed algorithm as compared to previous sequential algorithm. The cost and turnaround time of individual jobs are minimized hence the cost of all arriving tasks and average turnaround time are minimized in a time instance respectively. As the task count increases, the results improve accordingly. This proposed algorithm can become more optimized and improved by considering following concerns:-

- For reducing the communication overhead, the cost based tasks may be grouped before the allocation of resources after applying resource capacity measuring mechanism.
- Other factors such as, the length of task, type of task, should be considered to schedule the tasks properly.

References

[1] J. Geelan, "Twenty-one experts define cloud computing," *Cloud Computing Journal, vol. 4, pp. 1-5, 2009.*

[2] P. J. Wild, P. Johnson and H. Johnson, "Understanding task grouping strategies," *PEOPLE AND COMPUTERS, pp. 3-20, 2004*.

[3] Q. Cao, B. Wei and W. M. Gong, "An optimized algorithm for task scheduling based on activity based costing in cloud computing," *In International Conference on e-Sciences 2009, pp. 1-3.*

[4] 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.*

[5] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, "Introduction to Algorithms": *The MIT Press, 2001, pp. 16.*

[6] R. N. Calheiros, R. Ranjan, C. A. F. De Rose, and R. Buyya, "Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services," *Arxiv preprint arXiv:0903.2525*, 2009.

[7] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose and R. Buyya "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience*, vol. 41, no. 1, pp. 23-50, 2011