eduction

Mean-Min task scheduling algorithm for cloud environment

S. Vaaheedha Kfatheen[1*]

Abstract
Cloud computing is a form of distributed computing which provides various services to the users. Because it of its dynamic and distributed services it has attracted the users and it is a fast growing technology in the field of distributed computing. It is highly scalable and it allows on-demand services. Millions of user are using the cloud environment because of its pay-as-you-go service. The cloud environment must be able to meet the user requirements and provide quality of service with high performance. The performance of cloud computing will be better if an efficient task scheduling algorithm is used for mapping the tasks to the available resources. An efficient scheduling algorithm increases the resource utilization thereby minimizing the execution time and cost. This paper proposes a Mean-Min Task Scheduling algorithm for Cloud environment based on the minimum execution time of the tasks submitted.

Keywords
Cloud Computing, Task Scheduling, Completion Time, Execution Time.

[1,2]*Department of Computer Science, Jamal Mohamed College, Tiruchirappalli–620020, Tamil Nadu, India.*
Corresponding author: [1] svaaheedha73@gmail.com

Contents

1. Introduction

Cloud computing is one of the form of utility computing in which the user need not own any infrastructure or software or platform needed for them. Instead the user can make use of the service provided by the cloud and they can pay as per use [1]. Cloud provides dynamic services using the concept of virtualization and scalability. Cloud can be defined as an execution environment in elastic fashion which involves a metered service for multiple varied users [2]. Some of the services provided by cloud environment are Software as a service (Saas), Platform as a service (Paas) and Infrastructure as a service (Iaas). Cloud technology is the pioneer in implementing commercial form of using the computer science by public users. The concept of virtualization is used to share

the available resources among the users who requested the service. The cloud provides a high performance by dividing the workloads among all the available resources efficiently which reduces the execution time, waiting time and throughput [3].The distributed computing environments provide various services out of which job scheduling is a major activity. The performance can be increased by efficient task scheduling which can also balance the load among the resources.

Scheduling is the process of mapping the jobs submitted by users to the best available resource so as to minimize the execution time, waiting time and cost. Scheduling maps the user jobs based on user objectives [4]. Scheduling can be done at job level or task level. Job scheduling is nothing but mapping and executing the whole job in a single resource whereas task scheduling is dividing the jobs into smaller tasks so that they can be executed in multiple resources simultaneously to reduce the total execution time. Task scheduling can be done in static mode or dynamic mode. In static scheduling, once the tasks are mapped to the resources they cannot be migrated from one resource to another one during run time whereas in dynamic scheduling the tasks scheduled in one resource can be migrated to another resource during run time. Thus static scheduling is termed as compile time scheduling while dynamic scheduling is termed as run time scheduling. The subdivided tasks can be dependent or independent. Dependent

tasks should follow the priority while independent tasks need not follow the hierarchy and they can be executed in any order.In the cloud computing environment virtual machines are considered as the resources and it should follow the policies to distribute the tasks among the virtual machines[3].This paper proposes a static independent task scheduling algorithm for cloud computing environment which reduces the makespan produced by the min-min algorithm.

2. Literature Review

Cloud Computing refers to operating, organizing, and accessing the services and applications online. The end user is not aware of the devices or virtual machines used for their service. The cloud scheduling algorithms can be of three types namely, Batch Mode, On-line Mode or Dependency Mode [5].Scheduling in cloud environment is highly challenging as the resources in data centers are heterogeneous. Min-Min, Max-Min, Tabu Search, Genetic Algorithm, Ant Colony Optimization, Hill Climbing and Particle Swarm Optimization are some of the popular algorithms used for cloud scheduling.

Panda et al [6] proposed an algorithm based on the popular benchmark algorithms Min-Min and Max-Min. The proposed algorithm gives better results for resource utilization and makespan.

Hussin et al [7] proposed a hybrid algorithm based on Particle Swarm Optimization, Best-Fit and Tabu-Search algorithms called BFPSOTS. The initial population for PSO has been generated using Best-Fit instead of random generation. To improve the local search Tabu-Search has been used. The authors have implemented the BFPSOTS using Cloudsim. To show the performance of the algorithm three different parameters execution time, resource utilization and cost have been considered. Through experiments the authors have shown that the proposed algorithm gives better results than the PSO algorithm.

Ibrahim Attiya et al [8] have designed a modified Harris hawks optimization (HHO) algorithm based on simulated annealing (SA) algorithm for job scheduling in cloud computing. The proposed algorithm uses SA for local search to improve the convergence rate and the quality of solution obtained from the HHO algorithm.

Deepika Saxena et al [9] have developed an algorithm for task scheduling named Dynamic fair priority task scheduling algorithm for cloud computing. The newly developed algorithm groups the submitted tasks into two groups deadline based tasks and cost based tasks. Then these tasks are arranged in sorted order and mapped into three priority queues low, middle and high. Then the tasks are selected from these queues for execution.

AoFeng Zhou [10] has proposed a resource scheduling algorithm which combines the features of global search method of genetic algorithm and convergence mechanism of ant colony optimization algorithm. The newly proposed algorithm gives better accuracy and convergence speed than the other algorithms.

Hongyan Cui et al [11] have proposed a task scheduling algorithm based on Genetic Algorithm and Ant Colony Optimization. The newly proposed algorithm is named as Genetic Algorithm-Chaos Ant Colony Optimization (GA-CACO) which gives better results in terms of convergence speed and performance.

Mohammed A. S. Mosleh et al [12] have designed a Cost-based Task Scheduling (ACTS) which gives data access to the virtual machines within the user specified deadline by minimizing the cost. In this algorithm low priority tasks are allocated minimum cost path and high priority tasks are allocated fast access path.

M.Zhu et al [13] have designed a delay-constrained optimization algorithm to increase the utilization of resources. They have developed a two-step workflow scheduling technique to reduce the cloud overhead within the specified time bound.

Ritu Kapur [14] has developed a cost effective algorithm to reduce the cost of resource. They state that the cost can be decreased by balancing the load among resources.

3. Proposed methodologymmts

In the proposed methodology, users submit the tasks with its execution time. Then the ETC (Expected Time to Complete) matrix is calculated. The completion time CTij of each task is calculated using the following equation:

$$CT_{ii} = ET_{ii} + R_i \qquad (3.1)$$

Where ET_{ij} is the expected execution time of ith task on j^{th} virtual machine (VM)and R_i is the ready time of thVM. The ready time is also called as the availability time of VM_j and it is calculated from the completion time of previously assigned tasks. The min-min has two phases namely, Resource Discovery and Resource Selection. In the resource discovery phase, minimum ETC for each task is found. Then the task with overall minimum ETC is selected and mapped to the VM that produces it. This second phase is termed as resource selection.

In MMTS, all the tasks are sorted in ascending order based on the minimum ETC and placed in a task queue TQ. Similarly the virtual machines are also sorted based on their speed and partitioned into two sets high speed resources and low speed resources. Then the mean of METs of all tasks in TQ is determined using the formula given in equation 3.2.

$$\text{mean} = \sum_{i=1}^{n} \frac{MET(T_i)}{n} \qquad (3.2)$$

Where n is the number of tasks to be scheduled.If the MET of a task is greater than the mean then it is scheduled in high speed resources set whereas if the MET of a task is less than the mean then it is scheduled in low speed resources set. The pseudocode for the proposed Mean-Min Task Scheduling (MMTS) algorithm is shown in figure 1.

```
Read the input ET of tasks
Read the number of VMs available and its capacity
Calculate ETC
Find the MET of each task and the VMj producing it.
Sort the tasks in ascending order based on
MET and Place them in TQ
Calculate mean for the tasks in TQ
While TQ not Empty
If MET(Ti)> mean then
Schedule task Tiin high speed resource set.
Update the ready time of VM.
Remove the task Ti from TQ
Else
Schedule task Tiin low speed resource set.
Update the ready time of VM.
Remove the task Ti from TQ
End If
End While
```

Fig 1. Pseudocode of MMTS

4. Results and Discussion

The efficiency and performance of proposed algorithm MMTScan be illustrated by conducting experiments with different number of tasks and virtual machines. In this work 5 tasks T1, T2, T3, T4 and T5 and 2 virtual machines VM1 and VM2 are considered.The virtual machines are assumed to have different speed.The computation time of 5 tasks and ETC of two virtual machines is shown in Table 1. Scheduling of tasks based on

Table 1. ETC matrix

| Task	Computation Time	VM1	VM2
T1	5	4.1	5
T2	3	2.6	3
T3	8	6.7	8
T4	10	9.2	10
T5	7	5.9	7

Min-Min algorithm is shown in Table 2 and the scheduling based on MMTS is shown in Table 3. Min-Min produces a makespan of 17.7 while MMTS produces a makespan of 15.9. From the results it is observed that MMTS produces less makespan than Min-Min algorithm. The schedule is also shown as Gantt chart in Figure 2 and Figure 3. The proposed MMTS algorithm effectively utilizes the resources.
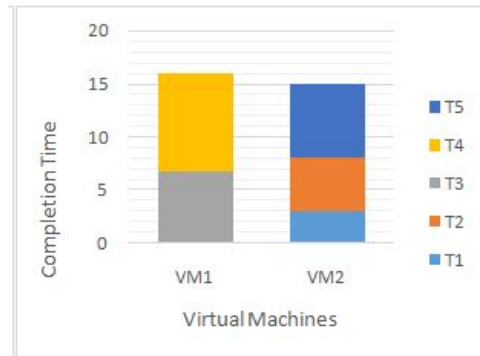
Table 2. Scheduling in Min-Min

Task	VM1	VM2
T2	2.6	
T1		5
T5	5.9	
T3		8
T4	9.2	
Completion	17.7	13
Time	17.7	13

Table 3. Scheduling in MMTS

Task	VM1	VM2
T1		3
T2		5
T3	6.7	
T4	9.2	
T5		7
Completion Time	15.9	15
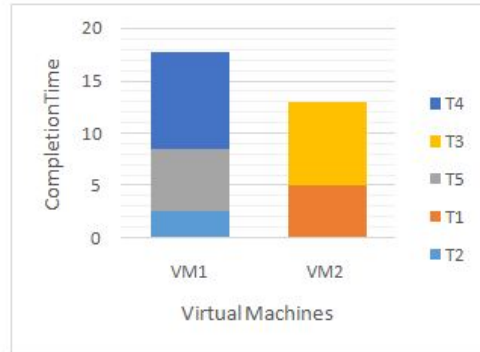
Fig 2. Gantt Chart of Min-Min

Fig 3. Gantt Chart of MMTS

The performance of the proposed algorithm is studied by varying the number of tasks and resources. The results are tabulated and shown in Table 4. From the results it is observed that MMTS produces less makespan than Min-Min algorithm. The graphical representation of the results are shown in Figure 4.

Table 4. Makespan produced by Min-Min and MMTS

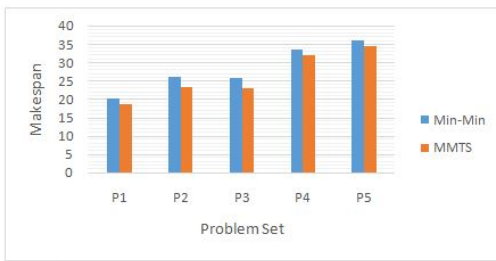| Problem Set	Min-Min	MMTS
P1	20.3	18.6
P2	26	23.4
P3	25.7	23
P4	33.5	31.8
P5	36	34.3

Fig 4. Comparison of Min-Min and MMTS

5. Conclusion and Future Work

Cloud computing is a form of distributed technology used by millions of users. The performance of cloud computing can be increased in terms of time and cost by effectively utilizing the resources. An efficient task scheduling algorithm is needed to reduce the time and cost of cloud computing environment. Min-Min algorithm schedules the tasks based on minimum completion time which gives a less makespan that is less completion time. But it does not produce a balanced schedule. To overcome this limitation a new Task Scheduling algorithm MMTS is proposed which produces less makespan than the min-min algorithm as well as balances the load by effectively utilizing the resources. In future, cost can be calculated and it can be shown that MMTS also reduces the cost.

References

[1] Sindhu S., Mukherjee S, "Efficient Task Scheduling Algorithms for Cloud Computing Environment", High Performance Architecture and Grid Computing. HPAGC 2011. *Communications in Computer and Information Science*, 169. Springer, Berlin, Heidelberg.

[2] Athokpam Bikramjit Singh, Sathyendra Bhat J., Ragesh Raju, Rio D'Souza, "A Comparative Study of Various Scheduling Algorithms in Cloud Computing", *American Journal of Intelligent Systems*, 7(3)(2017), 68-72.

[3] Tahani Aladwani, "*Types of Task Scheduling Algorithms in Cloud Computing Environment*", Scheduling Problems - New Applications and Trends, Mecca, Saudi Arabia, April 2020.

[4] T. Kokilavani, D.I. George Amalarethinam, "Reduced Makespan Task Scheduling Algorithm for Grid Computing", *I J C T A,* 9(27)(2016), 71-76.

[5] Priyanka Sharma , Dr. Sanjay Shilakari, Uday Chourasia, Priyanka Dixit, Alpana Pandey, "A Survey On Various Types Of Task Scheduling Algorithm In Cloud Computing Environment", *International Journal Of Scientific & Technology Research,* 9(01)(2020) ISSN 2277-8616, 1513 – 1521.

[6] S. K. Panda and P. K. Jana, "An efficient task scheduling algorithm for heterogeneous multi-cloud environment," 2014 International Conference on Advances in Computing, *Communications and Informatics (ICACCI),* IEEE Xplore: 01 Delhi, India, 2014, 1204-1209.

[7] Hussin M. Alkhashai, Fatma A. Omara, "An Enhanced Task Scheduling Algorithm on Cloud Computing Environment", *International Journal of Grid and Distributed Computing*, 9(7)(2016), 91-100, ISSN: 2005-4262 IJGDC.

[8] Ibrahim Attiya, Mohamed Abd Elaziz, Shengwu Xiong, "Job Scheduling in Cloud Computing Using a Modified Harris Hawks Optimization and Simulated Annealing Algorithm", *Computational Intelligence and Neuroscience*, 2020, Article ID 3504642, 17 pages, 2020.

[9] Deepika Saxena, Dr. R.K. Chauhan, "A Review On Dynamic Fair Priority Task Scheduling Algorithm In Cloud Computing", International Journal of Science, *Environment and Technology*, ISSN 2278-3687 (O), 3(3)(2014), 997-1003.

[10] AoFeng Zhou, "*Genetic Ant Colony Algorithm Improves Resource Scheduling in Cloud Computing*", ICISS 2020: Proceedings of the 2020 The 3rd International Conference on Information Science and System, March 2020, Pages 106–109.

[11] Hongyan Cui, Xiaofei Liu, Tao Yu, Honggang Zhang, Yajun Fang, Zongguo Xia, "Cloud Service Scheduling Algorithm Research and Optimization", *Security and Communication Networks*, 2017, Article ID 2503153, 7 pages, 2017.

[12] Mohammed A. S. Mosleh, G. Radhamani, Mohamed A. G. Hazber, Syed Hamid Hasan, "Adaptive Cost-Based Task Scheduling in Cloud Environment", *Scientific Programming*, 2016, Article ID 8239239, 9 pages, 2016.

[13] M. Zhu, Q. Wu and Y. Zhao, "A cost-effective scheduling algorithm for scientific workflows in clouds," 2012 IEEE 31st *International Performance Computing and Communications Conference* (IPCCC), Austin, TX, 2012, 256-265.

[14] Ritu Kapur, "A Cost Effective approach for Resource Scheduling in Cloud Computing", *IEEE International Conference on Computer, Communication and Control* (IC4-2015), 2015.

[15] Cao, J., Spooner, D.P., Jarvis, S.A., Nudd, G.R.: Grid Load Balancing Using Intelligent Agents. *Future Generation Computer Systems*, 21(1)(2005), 135–149.