IJCTA, 9(27), 2016, pp. 85-91 ISSN: 0974-5572

Enhanced Min-Min Algorithm for Meta-task Scheduling in Cloud Computing

D.I. George Amalarethinam* and S. Kavitha**

ABSTRACT

With the emergence of Cloud technologies, the problem of scheduling meta-task in heterogeneous systems has been stirring attention. Task scheduling is a NP-complete problemand it is more intricate under the Cloud environment. To better use tremendous capabilities of cloud system, effective and efficient scheduling algorithms are needed. In this paper,Enhanced Min-Min algorithm based on Min-Min algorithm is proposed. The strategy also considers the overall performance of resources to decide the scheduling of meta-task. The experimental results show that the proposed algorithm, Enhanced Min-Min, schedules tasks with lower makespan and better resource utilization.

Keywords: Cloud computing, meta-task, Min-min algorithm, resource utilization, makespan.

1. INTRODUCTION

Today is the epoch of smart computing. Everybody wants to use resources instantaneously. So researchers started to think about a technology, which can serve up anywhere anytime. Mark Weiser solved this problem by an intelligent design of computing, known as ubiquitous computing. In Morden era, this is called as Cloud Computing. Cloud computing is a new form of ubiquitous computing which delivers nearly all the IT services through the internet in virtualized manner [1]. Some of the issues of the cloud computing are data centre network structure expansibility, energy conservation, replica policies, security and scheduling mechanism [2][3][4]. The primary objective of this paper is optimizing the scheduling polices.

Task scheduling algorithm is to map thetasks to appropriate resources available in cloud environment in such a way that the total response timeis minimized [4]. In cloud environment,it is difficult to schedule a set of submitted tasks called meta-task, from different users on a set of computing resources to minimize the overall completion time.Scheduling is the set of policies to control theorder of work to be performed by a computersystem.There have been different types of scheduling algorithmexisting in cloud computing system;the main advantage of scheduling algorithm is to attain a high performance computing and the best system throughput [5]. Scheduling manages availability of resourcesand scheduling policy in order to give minimum makespan and maximumutilization of resources.

2. RELATED WORK

Intensive research has been conducted in cloudmeta-taskscheduling, to solve the problem of mapping a set of tasksto a set of resources. It has been originate that the scheduling problem is a NP complete problem. Many heuristicalgorithms are proposed because of lack of precisesolution. The mapping heuristics can be classified into twomodes: online mode and batch mode. In online mode the task is mapped to a resource as soon as it arrives to thescheduler. Task is considered only once for mapping and the mapping never changed once it is done. This mode

* Dean of Science & Director (MCA), Jamal Mohamed College, Trichy, Tamil Nadu, India, *Email: di_george@ymail.com*

** Assistant Professor, Department of Computer Science, SaradhaGangadharan College, Puducherry, India, *Email: kavi_dharan2000@yahoo.com*

of mapping is useful when arrival rate of tasks is low. In batchmode, tasks are collected into a set and called meta-task. Mapping of meta-task is performed at prescheduled times called mapping events. Mapping of each task is performed at every mapping event until it begins its execution. This mode can make better decisions becausenumber of tasks, number of resources and execution time of each task in each resource are known in advance.

Braun et al [6] have studied the relative performance of MET and MCT heuristic algorithms for task scheduling in grid computing. The Minimum Execution Time (MET) algorithm which assigns tasks to the resources based on their minimum expected execution time without considering the availability of the resource and its current load. This algorithm improves the makespan to some extent but it causes a severe load imbalance.

Minimum Completion Time (MCT) assigns tasks to the resources based on their minimum completion time. The completion time is calculated by adding the expected execution time of a task on that resource with the resource's ready time. The machine with the minimum completion time for that particular task is selected. But this algorithm considers the task only one at a time.

T. Kokilavaniet al [7] have studied about the performance of Min-Min algorithm which starts with a set of all unmapped tasks. The machine that has the minimum completion time for all tasks is selected. Then the task with the overall minimum completion time is selected and mapped to that resource. Compared to MCT this algorithm considers all tasks at a time. So it produces a better makespan.

Pardeep Kumar et al [8] have observed the performance of Max-Min algorithm which is similar to Min-Min algorithm. The machine that has the minimum completion time for all tasks is selected. Then the task with the overall maximum completion time is selected and mapped to that resource. The ready time of the resource is updated. This process is repeated until all the unmapped tasks are assigned. The idea of this algorithm is to reduce the wait time of the large jobs.

Marjan Kuchaki Rafsanjani et al [9] proposed the New Heuristic Approach begins same as Min-Min heuristic. The aim behind this heuristic is that to select pair machines and tasks from the first step, then the machine can execute its corresponding task effectively with a lower execution time comparing to other machines.

Arash Ghorbannia Delavar et al [10] proposed the SHIS algorithm uses the parameters such as resource access rate, quality of service and load balancing. This proposed method tries to obtain an optimal tasks mapping on resources by minimizing completion time of tasks or, completion time of the last task (makespan).

Jinquan Z et al [11] proposed the Weighted Mean Time algorithmwhich employs weighted mean execution time as heuristic and then assigns the taskwhich has maximum weighted mean execution time to the machine with minimum earliest completion time.

Sameer Singh Chauhan et al [12] proposed the WMTS Scheduling strategy which considers the performance of resources and decides the sequence of assignment of tasks based on the overall performance of resources.

Among all the algorithms stated the Min-Min algorithm is simple.But it considers the shortest tasks first so it fails to utilize the resources efficiently which leads to a load imbalance. The aimof this work is to overcome the drawback of the Min-Min algorithm. So a new enhanced Min-Min algorithm is proposed. This proposed algorithm reduces the makes pan and gives better resource utilization.

3. ENHANCED MIN-MIN SCHEDULING STRATEGY

In this study, the proposed algorithm has been developedunder a set of assumptions:

- The applications to be executed are acollection of indivisible tasks, usually referred to as meta-task.
- Estimates of expected task execution times on eachResource are known in advance
- Resources are considered to be consistent.
- The scheduling process is to be performed statically.

- Deadlines and Priorities are not to be associated with the tasks.
- The number of the meta-tasks and the number ofresources in the cloud computingenvironment is known.
- Each resource executes a single task at a time in the order in which the tasks are assigned.

The proposedscheduling strategy, Enhanced Min-Min, considers the speed of all resources and decides the sequence of assignment of tasksbased on the speed of resources. Initially the resources are sorted in ascending order based on their speed. The task t_i having minimum execution time is selected andassigned to the fastest resource which gives the minimum completion time,and the task t_i is removed from the set. Again the task havingminimum execution time is selected and is assigned to the second fastest resourceand so on. While assigning,the total completion time of that particular resource is comparing with the makespan of Min-Min algorithm each time by using following equation.

$$\text{makespan}(Rj) < \text{MSMM} \tag{1}$$

where, MSMM is makespan of Min-Min algorithm

If it is lesser the task t_i is removed from the problem set. Otherwise the task t_i is reassigned to the fastest resource which gives minimum completion time. This process will be continued until all the tasks in meta-task set are to be assigned.

The completion time of all tasks on each resource is computed by using the following equation:

$$CT_{ij} = ET_{ij} + r_j \tag{2}$$

where ET_{ij} is execution time of task T_i in resource R_j and r_j is waiting time of the task T_i

Makespan of the proposed algorithm is computed by using the following equation:

$$\text{Makespan} = \max(CT_i) \tag{3}$$

where, CT_i is completion time of task T_i and T_iõ MT,where MT is meta-task set

The proposed algorithm gives minimum makespan comparing to Min-Min algorithm for each problem set. And the proposed algorithm gives better resource utilization. The pseudo code for proposed algorithm is shown in Table 1.

Table 1
Pseudo code for Enhanced Min-Min Algorithm

```
1.  for all task ti in MT
2.      for all resources mj
3.  CTij←–ETij + rj
4.  sort the resources in ascending order according to the speed
5.  for each task ti in MT
6.  if rj==0 for all j
7.  CTij←–findmin() //find minimum CTij and the resource mj that obtained it
8.  assign ti to mj
9.  else
10. CTji←–findnextmin() //find next minimum CTij and the resource that obtained it
11. assign ti to mj
12. if (makespan(mj)<MSMM)
13. delete ti from MT
14. update rj
15. update CTij for all i
16. else
17. assign ti to m1
18. delete ti from MT
19. update rj
20. update CTij for all i
```

4. EXAMPLE WITH ILLUSTRATION

A cloud environment with three resources R1, R2 and R3 and a meta-task group MTwith five tasks T1, T2, T3, T4 and T5 is considered. The cloud scheduler is supposed to schedule all the tasks within MTon the available resources R1, R2 and R3. Since Min-Min algorithm is simple and produces a better makespan than the other algorithms discussed in the literature, the proposed algorithm executes with unique variation of Min-Min algorithm which gives the better makespan and also better resource utilization. In this problem the execution time of all tasks in each resource are known prior. They are represented (in secs) in Expected Time to Compute (ETC) table. Table-2 represents the execution time of the tasks on each resource.

Static scheduling of all tasks to machines according to Min-Min algorithm is shown in Figure1.In Min-Min algorithm all tasks are scheduled to resource R1 and R2. So the resource R3 remains idle. The makespan produced by Min-Min algorithm is 21 seconds.

Table 2
ETC matrix of 5 tasks and 3 resources

Tasks	*Resources*		
	R1	*R2*	*R3*
T1	2	4	6
T2	6	12	18
T3	3	6	9
T4	10	20	30
T5	4	8	12

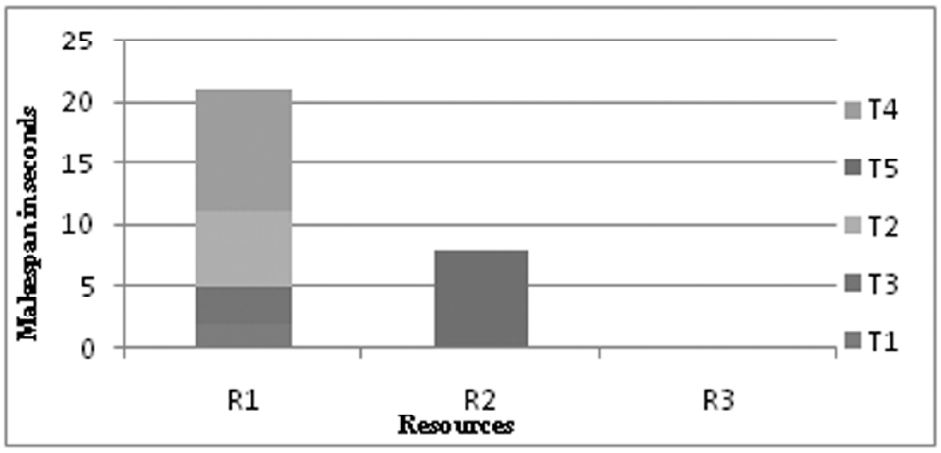

Figure 1: Makespan of Min-Min algorithm

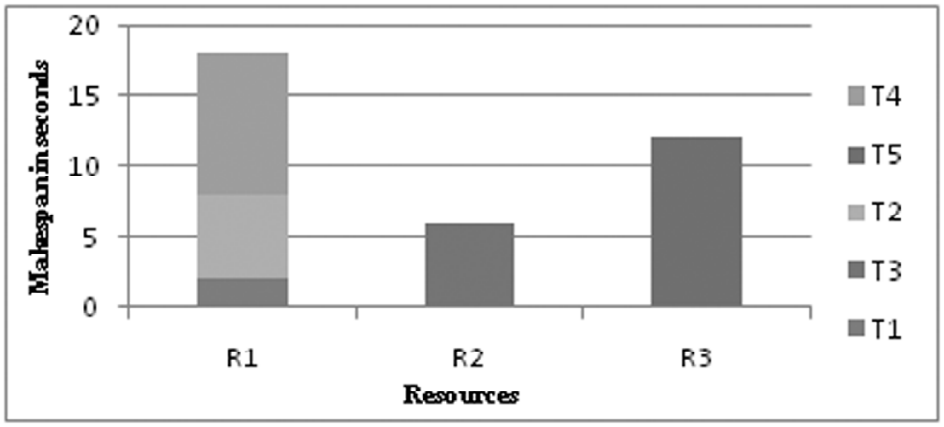

Figure 2: Makespan of Enhanced Min-Min algorithm

According to proposed algorithm, Enhanced Min-Min, the tasks T1, T2, T4 are assigned to the resource R1, T3 is assigned to R2 and T5 is assigned to R3. Makespan of proposed algorithm is 18 and all 3 resources are utilized. Scheduling of tasks based on proposed algorithm is shown in Figure 2.

5. RESULTS

To evaluate the efficiency of the proposed algorithm, problemshaving machine and task heterogeneity arecollected from various research papers, [7][8][12][13][14][15], and executed for both Min-Min and proposed Enhanced Min-Min algorithms.The results are evaluated on the basis of following performance metrics.

1. *Makespan*: Makespan is the measure of the throughput of the Cloud. It can becalculated using the equation (3). The results for the algorithms are tabulated in Table 3.

The proposed algorithm Enhanced Min-Min algorithm outperforms the existing Min-Min algorithm and the results are plotted in a graphshown in Figure 3, demonstrates that Enhanced Min-Min algorithm produces less makespan than the Min-Min algorithm for all problemsets.

2. *Average resource utilization rate*: Average resource utilization rateru[11] of allresource is calculated through following equation :

$$ru = \frac{\sum_{j=1}^{m} ru_j}{m} \tag{4}$$

Here, ru_j is the average resource utilization rate of resource r_j. It can be calculated by the following equation:

Table 3
Comparison of Min-Min and Enhanced Min-Min

Problem set	*Min-Min*	*Enhanced Min-Min*
P1	10	8
P2	11	9
P3	21	18
P4	39	36
P5	60	50
P6	94	80
P7	107.67	84.31
P8	354	285

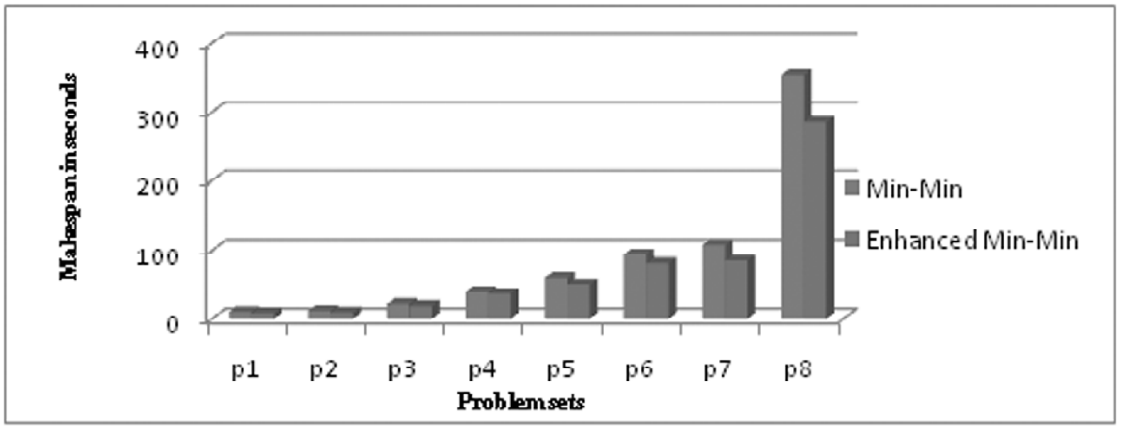

Figure 3: Comparative Study of Min-Min and Enhanced Min-Min

$$ru_j = \frac{\sum(te_i - ts_i)}{T} \tag{5}$$

where, te_i is the end time of executing task t_i on resource m_j, ts_i is the start time of executing task t_i on resource m_j and T is the total application time so far, it can be calculated using following equation:

$$T = \max(te_i) - \min(ts_i) \tag{6}$$

The results of resource utilization rate are shown in figure 4.

This figure shows that Enhanced Min-Minincreases resource utilization than the Min-Min algorithm for all problemset.

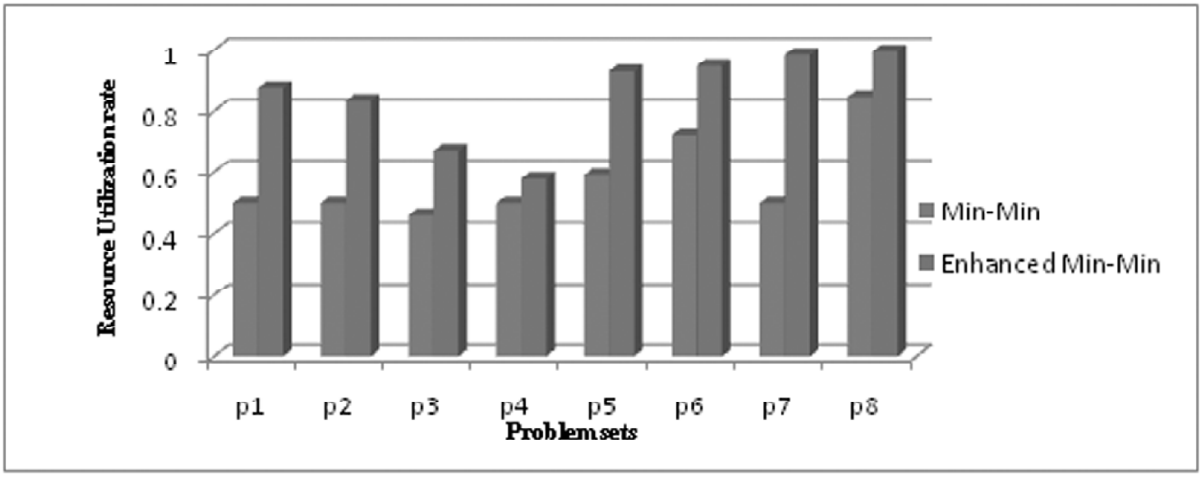

Figure 4: Resource Utilization rate

6. CONCLUSION AND FUTURE WORK

Min-Min algorithm is applicable in small scaledistributed systems. If the number of the small tasks is increased comparing to the large tasks in a meta-task, the Min-Minalgorithm cannot schedule tasks efficiently, and themakespan of the system gets relatively large. To overcome thelimitations of Min-Min algorithm, a new task schedulingalgorithm, is proposed. It includes theadvantages of Min-Min algorithm and overcomes the disadvantages. The experimental resultsof the proposed algorithm Enhanced Min-Min algorithm for various problems provethat it outperforms the existing Min-Min algorithm in terms of makespan and resource utilization. Thisstudy is only concerned with the number of the resources andtask execution time. The study can be further extended byconsidering the priority of tasks and cost factor in cloud environment.

REFERENCE

[1] Shekhar Singh, Mala Kalra, "Task Scheduling Optimization of Independent Tasks inCloud Computing Using Enhanced Genetic Algorithm", *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, ISSN 2319 – 4847, **3(7)**, July 2014

[2] Anu Gupta, "Cloud Computing Growing Interest and Related Concerns", *IEEE 2nd International Conference on Computer Technology and Development (ICCTD)* 2010.

[3] KresimirPopovic, ZeljkoHocenski," Cloud computing security issues and challenges",*MIPRO*, 24-28, 2010.

[4] Pham Phuoc Hung, Mui Van Nguyen, Mohammad Aazam, Eui-Nam Huh, "Task Scheduling for Optimizing Recovery Time in Cloud Computing", IEEE, 2014.

[5] HarpreetKaur,Maninder Singh, "Review of Various Scheduling Techniques in Cloud Computing", *International Journal of Networking & Parallel Computing*, **1(2)**, 2012.

[6] Braun, T.D., Siegel, H.J., Beck, N., Boloni, L.L., Maheswaran, M., Reuther, A.I., Robertson, J.P., et al. "A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems", *Journal of Parallel and Distributed Computing*, **61**, 810–837, 2001.

[7] T. Kokilavani Dr. D.I. George Amalarethinam, "Load Balanced Min-Min Algorithm for Static Meta-Task Scheduling in Grid Computing" *International Journal of ComputerApplications*,**20**, 2011.

[8] Pardeep Kumar, AmandeepVerma,"Independent Task Scheduling in Cloud Computing byImproved Genetic Algorithm", *International Journal of Advanced Research in Computer Science and Software Engineering*, 2**(5)**,2012.

[9] MarjanKuchaki Rafsanjani, Amid KhatibiBardsiri, "A New Heuristic approach for Scheduling Independent Tasks on Heterogeneous Computing System", *International Journal of Machine and Computing.***2,** 2012.

[10] ArashGhorbanniaDelavar, Yalda Aryan, "A Synthetic Heuristic Algorithm for Independent Task Scheduling in Cloud System", *International Journal of Computer Science Issues*, **8(6)**,2011.

[11] Jinquan Z, Lina N, Changjun J, "A Heuristic Scheduling Strategy for Independent Tasks on Grid", *Proceedings of the Eighth International Conference on High-Performance Computing in Asia-Pacific Region (HPCASIA '05)*, 2005.

[12] Sameer Singh Chauhan, R. C. Joshi, "A Weighted Mean Time Min-Min Max-Min Selective Scheduling Strategy for Independent Tasks on Grid", *Advance Computing Conference (IACC), IEEE*4-9,2010.

[13] KaddaBeghdadBey, FaridBenhammadi, RédhaBenaissa, "Balancing Heuristic for Independent Task cheduling in Cloud Computing", *Programming and Systems (ISPS), 12th International Symposium, INSPEC Accession Number:15438512*, 1-6, 2015.

[14] PreritChawda, ParthaSarathi Chakraborty, "An Improved Min-Min Task Scheduling Algorithm for Load Balancing in Cloud Computing", *International Journal on Recent and Innovation Trends in Computing and Communication*, ISSN: 2321-8169, **4(4)**, 60-64, 2016.

[15] SoheilAnousha, Mahmood Ahmadi, "An Improved Min-Min Task Scheduling Algorithm in Grid Computing", *Research Gate*,LNCS 7861, 103–113,2013.