International Journal of Latest Trends in Engineering and Technology
Special Issue SACAIM 2017, pp. 099-102
e-ISSN:2278-621X

DESIGN AND IMPLEMENTATION OF NEW CLOUD SCHEDULING ALGORITHM

Santhosh B[1], Melisha Vaz [2] & Vandana BS[3]

Abstract- Scientific workflows are represented by using directed non-cyclic graphs (DAGs) model. Since tasks are subject to each other it requires an efficient task scheduling algorithm. In the cloud, dynamically receives jobs and submits to the data center for the execution. Here jobs are submitted based on their conditions. It submits the jobs only when all of its parent jobs completed execution successfully. These jobs will be listed in the scheduler. The scheduler will have the list of jobs which can be executed immediately. Since there is a limitation of the number of resources available it requires an efficient task scheduling mechanism. In this paper, we have compared several tasks scheduling algorithm with respect to make span of jobs in the workflow. The algorithms analyzed using WorkflowSim simulator.
Keywords– scheduling, workflow, make span, level

1. INTRODUCTION

Cloud Computing is the latest trend in today's world. It provides on-request services like hardware, software, platform, infrastructure, and storage etc. dynamically to the user according to the "pay per use" model by using virtualized resources over the internet. Cloud computing has different applications such as business, social networks, and scientific applications.
While Cloud computing provides different services like IaaS, PaaS and SaaS etc. to end users but due to the novelty of cloud computing, it also experiences many types of research issues such as security, performance, database management, virtual machine migration, server consolidation, fault tolerance and workflow scheduling etc. Among these workflow scheduling is real issue for scientific applications

2. RELATED WORKS

2.1 RASA Algorithm –

In task scheduling algorithm RASA [1], the distribution and scalability characteristics of grid resources, is proposed. The algorithm is built through a study and analysis of well-known task scheduling algorithm, Min-min and Max-min. RASA utilizes the advantages of the both algorithms and covers their drawbacks. To accomplish this, RASA firstly estimates the completion time of the tasks on each of the available grid resources and after that applies the Max-min and Min-min algorithms, alternatively. In this regard, RASA utilizes the Min-min strategy to execute small tasks before the large ones and applies the Max-min strategy to avoid delays in the execution of large tasks and to support concurrency in the execution of large and small tasks. RASA on scheduling independent tasks within grid environments explains the applicability of RASA to achieve schedules with comparatively lower makespan.

2.2 Hybrid Min-Min Max-Min Algorithm –

The high cost of supercomputers and the requirement for expansive scale computational resources has led to the development of network of computational resources known as Grid. To better utilize enormous capabilities of this large scale distributed system, effective and efficient scheduling algorithms are required. Hybrid Min-Min Max-Min Algorithm based on two conventional scheduling algorithms, Min-Min and Max-Min, to utilize their cons and in the meantime, overcome their pros. This heuristic scheduling algorithm, called min-min min-max selective, is assessed utilizing a grid simulator called GridSim by comparing to its performance against the two basic heuristics which it comes from. The simulation results show that the new heuristic can lead to significant performance gain for a variety of scenarios [2].

2.3 Enhanced Max-Min Task Scheduling Algorithm –

Cloud Computing is the utilization of computing resources (Hardware and Software) that are delivered as a service over a network (typically the internet). It supplies a high performance computing based on protocols which permit shared computation and capacity over long distances. In cloud computing, there are many tasks required to be executed by the

[1] Department of Software Technology, Aloysius Institute of Management and Information Technology (AIMIT), Beeri, Mangalore, Karnataka, India
[2] Department of Software Technology, Aloysius Institute of Management and Information Technology (AIMIT), Beeri, Mangalore, Karnataka, India
[3] Department of Software Technology, Aloysius Institute of Management and Information Technology (AIMIT), Beeri, Mangalore, Karnataka, India

accessible resources to accomplish best performance, minimal total time for completion, shortest response time, utilization of resources etc. Because of these diverse aims. In this algorithm, a remarkable adjustment of Enhanced Max-min task scheduling algorithm [3] is proposed. The algorithm is assembled based on comprehensive study of the impact of Improved Max-min task scheduling algorithm in cloud computing. Enhanced Max-min is based on the expected execution time instead of completion time as a selection basis. Enhanced (Proposed) Max-min is also based on the expected execution time instead of completion time as a selection basis but the only difference is that Improved Max-min algorithm assign task with Maximum execution time (Largest Task) to asset produces Minimum completion time (Slowest Resource) while Enhanced Max-min relegate task with average execution time (average or Closest greater than average Task) to resource produces Minimum fulfillment time (Slowest Resource).

2.4 Improved Max-min Algorithm-

Improved Max-min algorithm [5] is the improvement over to the Enhanced Max-Min algorithm, In Enhanced Max-Min the largest task just greater than the average execution time is selected and is assigned to the slower resource then Max-min algorithm is followed. In Improved Max-Min algorithm, average execution time is calculated using the arithmetic mean. Every time instead of the largest job, job just greater than average execution time of all jobs in the job list is selected and assigned to the resource this gives minimum execution time.

3. PROPOSED ALGORITHM

The proposed algorithm is the improvement over to the Improved Task scheduling Algorithm based on Max-min for Cloud Computing [4]. The proposed algorithm first checks whether the tasks are uniformly distributed. The uniformity is calculated based on the length of each task. If the tasks are uniformly distributed select the task which is just greater than the Arithmetic mean average execution time else select the task just greater than the Geometric mean average execution time.

3.1 Proposed AvgTask-MaxMin Algorithm

1. For all submitted tasks in Meta-task; Ti
1.1. For all resources; Rj
1.1.1. Cij = Eij + rj
2. While Meta-task not Empty
2.1 if (all the tasks are uniformly distributed)
Find task Tk which is just greater than Arithmetic Average execution time.
Else // if the taks are not uniformly distributed
Find task Tk which is just greater than Geometric Average execution time.
2.2 Assign task Tk to resource Rj which gives minimum completion time.
2.3. Remove task Tk from Meta-tasks set.
2.4. Update rj for selected Rj.
2.6. Update Cij for all j.

4. EXPERIMENTAL RESULTS

WorkflowSim is an open source workflow simulator from University of Southern California. It provides workflow level support of simulation by extending the CloudSim. The simulator CloudSim supports only the static scheduling algorithms and workflowsim supports both static and dynamic scheduling algorithm.

4.1 Test Data 1 and Its Results

Simulation uses one data center, 4 vms with mips speed 300, 600, 700 and 800 respectively and datasets Montage_50 and Montage_100. The makespan of FcFs, MinMin, MaxMin, IMaxMin, RASA and proposed algorithm is calculated and shown, as in Table 1

DataSet→ / Scheduling Algorithm	Montage_50	Montage_100
FcFs	293.34	570.4
MinMin	253.14	495.87
MaxMin	251.76	494.06
IMaxMin	251.76	492.46
RASA	252.34	491.16
AvgTask-MaxMin	250.17	491.10

Table 1: Makespan in sec's for the datasets Montage_50 and Montage_100

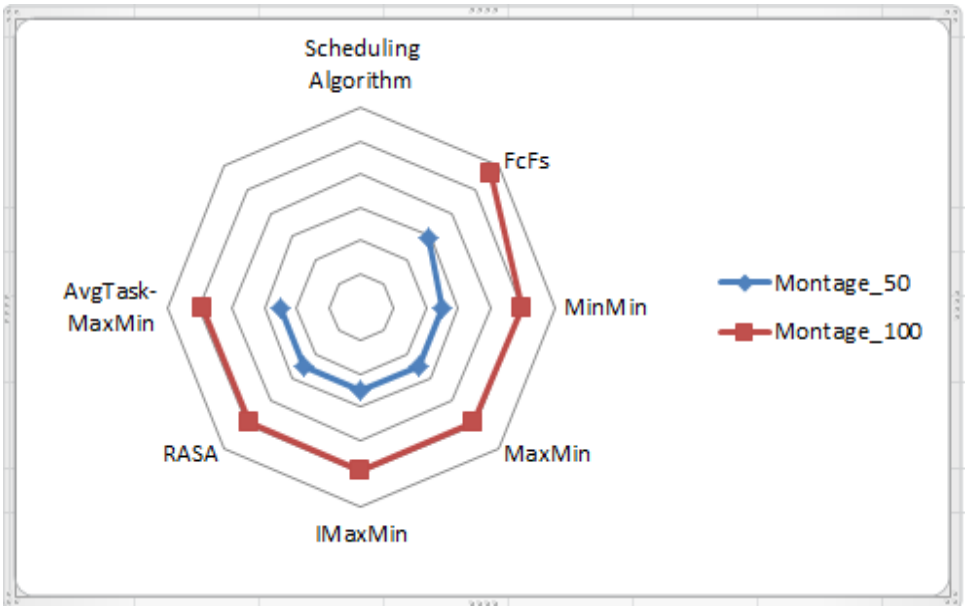

Fig 1. Graph Representation for the datasets Montage_50 and Montage_100

The above alogorithms are tested with 4 vms and mips speed of 300,600,700,800 along with the datasets Montage_50 and Montage_100 and with scheduling algorithms. Table 1 gives the results of makespan(in secs) and it shows that proposed algorithm is efficient than the other scheduling algorithms.

4.2 Test Data 2 and Its Results

Simulation uses one data center, 6 vms with mips speed 300, 600, 700, 800, 900 and 550 respectively and datasets CyberShake__50 and CyberShake__100. The makespan of FcFs, MinMin, MaxMin, IMaxMin, RASA and proposed algorithm is calculated and shown, as in Table 3.2

Data Sets → / Scheduling algorithms	CyberShake_50	CyberShake_100
FcFs	485.75	953.84
MinMin	503.04	1237.22
MaxMin	483.81	881.38
IMaxMin	471.56	876.81
RASA	468.95	891.75
AvgTask-MaxMin	460.30	864.45

Table 2: Makespan in sec's for the datasets CyberShake_50 and CyberShake_100

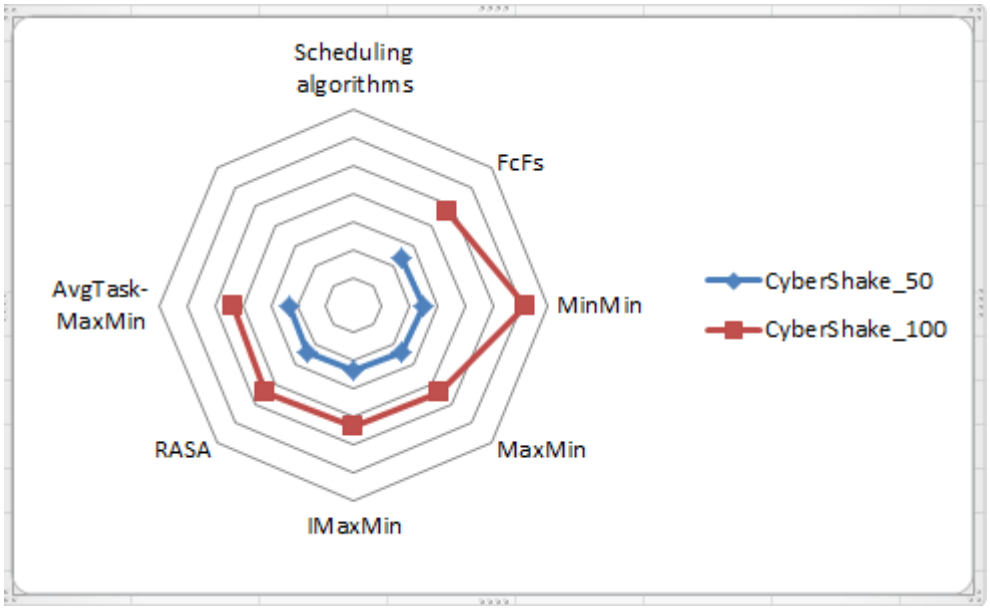

Fig 2. Graph Representation for the datasets CyberShake_50 and CyberShake_100

The above algorithms are tested with 6 vms and mips speed of 300,600,700,800,900 and 550 along with the datasets Montage_50 and Montage_100 and with scheduling algorithms. Table 2 gives the results of makespan(in secs) and it shows that proposed algorithm is efficient than the other scheduling algorithms.

5.CONCLUSION

This paper presents the design of AvgTask- Max-Min Algorithm for task scheduling in cloud computing environment. The proposed algorithm and other variations of MaxMin algorithms are tested with different datasets. Here the efficiency of the algorithm is tested with respect to the makespan. The experimental results demonstrate that proposed AvgTask- Max-Min Algorithm is efficient than Improved Max-Min, RASA and other traditional scheduling algorithms.

6. REFERENCES

[1] Saeed Parsa and Reza Entezari-Maleki, “RASA: A New Task Scheduling Algorithm inGrid Environment”, World Applied Sciences Journal 7 (Special Issue of Computer & IT): 152-160, 2009 ISSN 1818.4952.

[2] Kobra Etminani , Mahmoud Naghibzadeh , Noorali Raeeji Yanehsari, " A HYBRID MIN-MIN MAX-MIN ALGORITHM WITH IMPROVED PERFORMANCE," Department of Computer Engineering, Ferdowsi University of Mashad, Iran.

[3] Upendra Bhoi, Purvi N. Ramanuj, " Enhanced Max-min Task Scheduling Algorithm in Cloud Computing," P.G.Student, Department of Computer Science & Technology, L.D.College of Engineering, Gujarat Technological University, Ahmedabad.

[4] O. M. Elzeki, M. Z. Reshad, M. A. Elsoud " Improved Max-Min Algorithm in Cloud Computing ", International Journal of Computer Applications (0975 – 8887) Volume 50 – No.12, July 2012 ..

[5] B. Santhosh and D. H. Manjaiah, "An improved task scheduling algorithm based on max-min for cloud computing", International Journal of Innovative Research in Computer and Communication Engineering, vol. 2, no. 2, pp. 84-88, 2014