Davinder Kaur *et al*, International Journal of Computer Science and Mobile Computing, Vol.3 Issue.5, May- 2014, pg. 552-556

Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 5, May 2014, pg.552 – 556

RESEARCH ARTICLE

An Improved min - min Algorithm for Job Scheduling using Ant Colony Optimization

Davinder Kaur[1], Sarpreet Singh[2]

[1]Department of Computer Science Engg., Sri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, India

[2]Department of Computer Science Engg., Sri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, India

Email id: davinder.gill301@gmail.com, ersarpreetvirk@gmail.com

Abstract: Grid computing is recognized as one of the most powerful vehicles for high performance computing for data-intensive scientific applications. Grid is alternative to traditional distributed computing. It addresses issues such as resource discovery, heterogeneity, fault tolerance and task scheduling. Scheduling is the one of the current issue in the complex heterogeneous environment. Job scheduling is used to schedule the user jobs to appropriate resources in grid environment. The most simple and well known scheduling algorithm is min-min algorithm that is used to minimize the makespan but it has some disadvantages. To remove the disadvantages of the min- min in this paper we proposed an improved min-min algorithm which uses the features of Ant Colony Optimization.

Keywords: Grid Computing; job scheduling; min-min algorithm; Ant Colony Optimization (ACO)

I. INTRODUCTION

Grid computing has emerged from distributed computing by its focus on large resource sharing, intensive applications and high performance orientation. The term grid was inspired by the analogy to power grids, which provide electricity to people, where the location of a electric power source is far away and unimportant to consumer. Similarly in grid computing, the user who submit their jobs need not be aware of the location of the resources that are used for executing their jobs. Grid is an infrastructure that involves the integrated and collaborative use of computers, networks, databases and scientific instruments owned and managed by multiple organizations. Foster and Kesselman defined the Grid as follows [1]: A computational Grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities.

There are number of problems, in the fields of science, engineering, and business, which are not tractable using the current generation of high-performance computers. In fact, due to their size and complexity, these problems required huge amount of resource (computational and data) intensive [2]. For such type of applications grid can be used. Grid Computing is also used in application areas like weather prediction, astrophysics, bioinformatics, earth quake research, ground water pollution and multi-particle physics.

Unlike traditional parallel and distributed systems, Grids address issues such as security, uniform access, scheduling, resource management, dynamic discovery, dynamic aggregation, and quality-of- services. Scheduling is considered one of the important issues in grid computing.

Grid scheduling is defined as the process of making scheduling decisions involving resources over multiple administrative domains. This process can include searching multiple administrative domains to use a single machine or scheduling a single job to use multiple resources at a single site or multiple sites[3]. Job scheduling is based on the necessity of a user who has a set of jobs to execute. Sometime the user's machine is not able to process the jobs either because of resource or hardware constraints, for such cases the user can use the grid system for running the job. The user submits the set of jobs to the job scheduler and the job scheduler splits the job depending on certain factors and gives it to the machines having available resources on the grid. The machines will complete the task and final result will be given to the user[4].

There are large number of scheduling algorithms in grid like heuristic algorithm, min-min and max-min static algorithm, ant colony optimization for load balancing and many more for effective and efficient job scheduling. Min-Min and Max-min are the two algorithms that work on the basis of execution and completion time of each task on the each available grid resource [5].

There are 2 phases [6] in Min-Min algorithm. In the first phase it finds the minimum execution time of all tasks by assigning them to the resources. Then in the second phase it chooses the task with the least execution time among all the tasks. The algorithm proceeds by assigning the task to the resource that produces the minimum completion time. The same procedure is repeated by Min-Min algorithm until all tasks are scheduled. The following are the some limitation of Min-Min algorithm:-

i. It chooses smaller tasks first which makes use of resource with high computational power. As a result, the schedule produced by Min-Min is not optimal when number of smaller tasks exceeds the large ones.
ii. One resource can execute only one job at a time.
iii. Size and number of resources are static and should be known in prior.[4]

To overcome the first limitation, there are so many algorithms that improve the performance of min-min algorithm. To overcome the second and third limitation of min-min algorithm, a new algorithm is proposed that uses the features of ACO with the min-min algorithm because it is better for load balancing. Thus a better load balancing is achieved and the total response time of the grid system is improved. The proposed algorithm applies the Ant Colony Optimization in the first phase and then Min-Min strategy for better result.

ACO has been used by researchers in grid computing for addressing load balancing [7], job scheduling [8] and related problems. Ant colony optimization (ACO) was first introduced by Marco Dorigo as his Ph.D. thesis and was used to solve the TSP problem [9]. The ant colony algorithm for job scheduling in grid aims at submitted jobs to resources based on the processing ability of jobs as well as the characteristics of the jobs. Ant colony algorithm is the bio-inspired heuristic algorithm, which is derived from the social behavior of ants. Ants work together to find the shortest path between their nest and food source. When the ants move, each ant will deposit a chemical substance called pheromone. Using this pheromone, the shortest path is found. The same concept is used to assign jobs in grid computing. When a resource is assigned a job and completes its job then its pheromone value will be added each time. If a resource fails to finish a job, it will be punished by reducing pheromone value. The issue here is the stagnation, where there is a possibility of jobs being submitted to same resources having high pheromone value[4].

In the ant colony algorithm [10], the load balancing method is proposed to solve the issue of stagnation. The algorithm is as follows:-

i. The user will send request to process a job
ii. The grid resource broker will find a resource for the job
iii. The resource broker will select the resource based on the largest value in the pheromone value matrix
iv. The local pheromone update is done when a job is assigned to a resource.
v. The global pheromone update is done when a resource completes a job
vi. The execution result will be sent to the use

When the resource broker select a particular resource for a job j, jth column of the Pheromone Value matrix will be removed and jobs will be assigned to other resources. Thus the load balancing is achieved[4].

II. RELATED WORK

T. Kokilavani et al., [5] have proposed a new scheduling algorithm named Load Balanced Min-Min (LBMM) Algorithm for Static Meta-Task scheduling in grid computing to overcome the limitations of Min-Min algorithm. It is performed in two-phases. It uses the advantages of Max-Min and Min-Min algorithms and covers their disadvantages. In the first phase the traditional Min-Min algorithm is executed and in the second phase the tasks are rescheduled to use the unutilized resources effectively. The LBMM algorithm reduces the makespan and increases the resource utilization.

HE. X et al., [11] have proposed a new algorithm based on the conventional Min-Min algorithm to achieve high throughput computing in grid environment. This proposed algorithm named as QoS guided Min-Min for heuristic grid task scheduling, schedules tasks requiring high bandwidth before the others. Therefore, if the bandwidth required by different tasks varies highly, the QoS guided Min- Min algorithm provides better results than the traditional Min-Min algorithm. Whenever the bandwidth requirement of all of the tasks is almost the same, the QoS guided Min-Min algorithm acts similar to the traditional Min-Min algorithm. Furthermore, it also tolerates inaccurate execution estimations.

Singh. M et al., [12] present a QoS based predictive Max-Min, Min-Min Switcher algorithm for scheduling jobs in a grid. The algorithm makes an appropriate selection among the QoS based Max-Min or QoS based Min-Min algorithm on the basis of heuristic applied, before scheduling the next job. The effect on the execution time of grid jobs due to non-dedicated property of resources has also been considered. The algorithm uses the history information about the execution of jobs to predict the performance of non-dedicated resources.

Salman Meraji et al., [13] have proposed a new algorithm which is called best-min algorithm in order to overcome the disadvantage of min-min algorithm that is schedule produced by min-min is not optimal with respect of load balancing and max-min's relative time to finish assigning tasks is too high. It is a two phase algorithm. The best-min algorithm uses min- min to get the makespan in first step and reschedule the tasks in the second phase in order to reduce the makespan. There is a condition in algorithm that best-min should consider all the resources in grid environment and this is caused to maximize resource utilization as well.

Saeed Parsa et al.,[14] have proposed a new task scheduling algorithm in Grid Environment called RASA, considering the distribution and scalability characteristics of grid resources. RASA uses the advantages of the min-min and max-min algorithms and outcomes their disadvantages . RASA uses 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 first apply min- min and then max-min algorithm.

Yagoubi. B et al., [15] have offered a model to demonstrate grid architecture and an algorithm to schedule tasks within grid resources. The algorithm tries to distribute the workload of the grid environment amongst the grid resources, fairly. Although, the mechanism used here and other similar strategies which try to create load balancing within grid resources can improve the throughput of the whole grid environment, the total makespan of the system does not decrease, necessarily.

Siriluck Lorpunmanee et al., [16] have developed a general framework of grid scheduling using dynamic information and an ant colony optimization algorithm to improve the decision of scheduling. The performance of various dispatching rules such as First Come First Served (FCFS), Earliest Due Date (EDD), Earliest Release Date (ERD), and an Ant Colony Optimization (ACO) are compared. It is found that the scheduling system using an Ant Colony Optimization algorithm can efficiently and effectively allocate jobs to proper resources. It is also observed that the different number of machines have different significant impact on schedule lengths.

D.Maruthanayagam et al., [17] proposes an improved ant colony scheduling algorithm combined with the concept of Resource Aware Scheduling Algorithm (RASA). The first step for this to select a set of computers and a network connection (switching, routers, Ethernet, Myrinet Etc.,) for an application. The RASA algorithm first estimates the completion time of the tasks on each of the available grid resources and then applies the Max-min and Min-min algorithms. The improved ant colony algorithm is evaluated using the simulated execution times for a grid environment. Before assigning the tasks to machines, the expected execution time for each task on each machine must be estimated and represented by an Expected Time calculation. And also this algorithm can find an optimal processor and network for each machine to allocate a job that minimizes the tardiness time of a job when the job is scheduled in the system. The proposed scheduling algorithm is designed to achieve high throughput computing in a grid environment.

Kumar Nishant et al., [18] proposed an algorithm for load distribution of workloads among nodes of a cloud or grid by the use of Ant Colony Optimization (ACO). The main aim of the algorithm is load balancing of nodes. This algorithm has an edge over the original approach in which each ant build their own individual result set and it is later on built into a complete solution. However, in this algorithm the ants continuously update a single result set rather than updating their own result set. The algorithm uses the both forward and backward movements for overload and underload.

III. PROPOSED WORK

Scheduling is the most important issue in grid computing. Min-Min is the most simple and well known algorithm for efficient job scheduling as it minimize the makespan .But still there are some disadvantages of min-min algorithm. In the proposed work to overcome the above mention limitations of min- min algorithm, a new algorithm is proposed that uses the features of min –min algorithm and ant colony optimization. At an end we will study the execution time of the new approach and also compare the result with the previously designed min-min algorithm and ant colony optimization algorithm.

IV. CONCLUSION

Grid computing usually consists of one computer that distributes the information and tasks to a group of networked computers to accomplish a common goal. To make grid more efficient there is a need of better job scheduling strategies. In job scheduling, users job's are scheduled on the appropriate resources in grid environment. There are many job scheduling algorithms in the grid computing. Min-min algorithm is the most commonly used job scheduling algorithm but it has some disadvantages. In the min-min one resource can execute only one job at a time and resources are known in prior To overcome these disadvantages a new algorithm is proposed using the features of min- min and ant colony optimization.

REFERENCES

[1] I. Foster and C. Kesselman. "The Grid: Blueprint for a New Computing Infrastructure*"*. Morgan Kaufmann, San Francisco, CA, July 1998.

[2] Parvin Asadzadeh, Rajkumar Buyya1, Chun Ling Kei, Deepa Nayar, and Srikumar Venugopal, "Global Grids and Software Toolkits: A Study of Four Grid Middleware Technologies" Grid Computing and Distributed Systems (GRIDS) Laboratory Department of Computer Science and Software Engineering The University of Melbourne, Australia

[3] Jennifer M. Schopf, " TEN ACTIONS WHEN GRID SCHEDULING" The User as a Grid Scheduler, Mathematics and Computer Science Division, Argonne National Laboratory

[4] G. Jaspher W. Kathrine and Mansoor Ilaghi U,"Job Scheduling Algorithms in Grid Computing – Survey*", International Journal of Engineering Research & Technology (IJERT)* Vol. 1 Issue 7, September - 2012 ISSN: 2278-0181

[5] T. Kokilavani, Dr. D.I. George Amalarethinam, "Load Balanced Min-Min Algorithm for Static Meta-Task Scheduling in Grid Computing" *International Journal of Computer Applications* (0975 – 8887) Volume 20– No.2, April 2011

[6] G. K. Kamalam and V. Murali Bhaskaran, "New Enhanced Heuristic Min-Mean Scheduling Algorithm for Scheduling Meta-Tasks on Heterogeneous Grid Environment", in European Journal of Scientific Research, Vol.70 No.3 (2012), pp. 423-430.

[7] H. Jamal, A. Nasir, K. Ruhana, K. Mahamud and A.M. Din, Load Balancing Using Enhanced Ant Algorithm in Grid Computing, Proceedings of the Second International Conference on Computational Intelligence, Modelling and Simulation, pp. 160-165, 2010.

[8] K.u Ruhana, K. Mahamud, H. Jamal and A. Nasir, Ant Colony Algorithm for Job Scheduling in Grid Computing, Proceedings of the Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation, pp. 40-45, 2010.

[9] K. Kousalya and P. Balasubramanie,"To Improve Ant Algorithm's Grid Scheduling Using Local Search". (HTTP://WWW.IJCC.US), VOL. 7, NO. 4, DECEMBER 2009.

[10] Ku Ruhana Ku-Mahamud and Husna Jamal Abdul Nasir. "Ant Colony Algorithm for Job Scheduling in Grid Computing", in 2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation, IEEE Computer Society 2010.

[11] HE. X, X-He Sun, and Laszewski. G.V, "QoS Guided Min- Min Heuristic for Grid Task Scheduling," *Journal of Computer Science and Technology*, Vol. 18, pp. 442-451, 2003.

[12] Singh. M and Suri. P.K, QPS A QoS Based Predictive Max-Min, Min-Min Switcher Algorithm for Job Scheduling in a Grid, Information Technology Journal, Year: 2008, Volume: 7, Issue: 8, Page No.: 1176-1181.

[13] Salman Meraji*, M. Reza Salehnamadi, "A Batch Mode Scheduling Algorithm for Grid Computing," *Journal of Basic and Applied Scientific Research*, Year :2013, Volume: 3, Issue: 4, pp. 173-181, ISSN 2090-4304

[14] Saeed Parsa and Reza Entezari-Maleki, "RASA: A New Task Scheduling Algorithm inGrid Environment," World Applied Sciences Journal,7(Special Issue of Computer & IT),pp 152-160, 2009

[15] Yagoubi. B, and Slimani. Y, "Task Load Balancing Strategy for Grid Computing," Journal of Computer Science, Vol. 3, No. 3, pp. 186-194, 2007.

[16] Siriluck Lorpunmanee, Mohd Noor Sap, Abdul Hanan Abdullah, and Chai Chompoo-inwai, "An Ant Colony Optimization for Dynamic Job Scheduling in Grid Environment" *world Academy of Science, Engineering and Technology International Journal of Computer*, Information Science and Engineering Vol:1 No:5, 2007

[17] D.Maruthanayagam, Dr. R.Uma Rani, "Improved Ant Colony Optimization for Grid Scheduling" *IJCSET* |November 2011 | Vol 1, Issue 10, 596-604

[18] Kumar Nishant, Pratik Sharma, Vishal Krishna, Chhavi Gupta and Kunwar Pratap Singh et al., "Load Balancing of Nodes in Cloud Using Ant Colony Optimization" 14th *International Conference on Modelling and Simulation*, 2012

