IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 6, Ver. IV (Nov.-Dec. 2016), PP 59-62
www.iosrjournals.org

Proposed Efficient Scheduling Algorithm in Grid Computing Using Gap Filling Approach

Haruna Abdu[1], Abubakar Aliyu[2], Faisal Murtala[3]
[1](Department of Computer Science, Federal University Lokoja, Kogi State, Nigeria)
[2](Department of Computer Science, Umaru Musa Yar'adua University Katsina, Nigeria)
[3](Depertment of Computer Science, Al-Qalam University Katsina, Nigeria)

Abstract: *Computer grids are systems containing heterogeneous, autonomous and geographically distributed nodes that are capable of executing user's applications (job). The proper functioning of a grid depends mainly on the efficient management of grid resources to carry out the various jobs that users send to the grid. In grid computing system management of resources and scheduling of jobs are the most crucial problem, a lot of scheduling policies are in to existence such as, First Come First Serve (FCFS) which is a simple policy used to improve efficiency of the scheduler, even though it have been widely used, it therefore end up with invoking low system utilization as a result of the gaps that exists in between the jobs submitted to the scheduler that makes the resources to be idle before scheduling or executing later job.To improve system utilization gap filling approach is used,this approach allows tasks, jobs to fill in those gaps. This paperproposes an efficient scheduling algorithm based on gap filling approach. The proposed algorithm uses intelligent agents in the scheduler to perform scheduling in a collaborative and coordinated way. The algorithm uses gap filling strategy to optimize priority rule algorithms in grid scheduling system.*

Keywords: *Grid Computing, Algorithm, Architecture, Gap filling, Intelligent Agent.*

I. Introduction

The term grid is referring to a distributed computing paradigm, able to provide and share resources based on the users need[1][2]. With the advancement of computing paradigms, Grid computing has emerged as a promising attractive distributed computing. However, Computational Grids aim to aggregate the power of heterogeneous, geographically distributed, multiple-domain-spanning computational resources to provide high performance or high-throughput computing [3]. Grid computing technology provides users with promising potentials such as; Resources balancing, Exploiting underutilized resources, Collaboration, Reliability and Scalability.To achieve the promising potentialsof computational Grids, an effective and efficient scheduling of jobs and resource management is immensely needed.Grid scheduling system is unlike traditional scheduling system due toits heterogeneous and dynamic nature [4].The information service plays a highly important role with regards to grid scheduling system.

The grid scheduling problem deals with assigning resources to a set of tasks that enter the grid through different nodes at any instant of time, considering availability (dynamic and autonomous) and computing capacity (heterogeneous), among other things [5]. The different parameters and requirements relevant to the grid's clients and its resources must also be considered to ensure the quality of the services for the different actors in the grid.

This paper focuses on the design of scheduling algorithm using gap filling approach for computational grids to increase the efficiency of the scheduler by eliminating gaps that exist between the jobs in the queue. Section II of the paperhighlight the literature reviewed, gaps and limitations of the existing researches,Section III discusses on how a submitted job is processed or executed in grid, Section IV highlight on scheduling system, Section V is the proposed algorithm,and section VI conclude the paper. This research is of important to enhance the promising potentials that grid technology provides to both clients and grid service providers. However, it will also help researchers who have interest in research on grid scheduling or grid computing at large.

II. Review of Related Works

Over the past several years, a lot of research has been conducted to study the problem of job scheduling and resource management in grid environment, this leads to the development of many schedulingalgorithms by different researchers to tackle such problems based on some specific application domains. Gap filling techniques plays an important role in grid computing environment for scheduling tasks.

Some of the related literatures reviewed are summarized in table 1, which highlight the works done by some researchers andtheir limitations.

Table 1: Summary of related literatures

S/N	Author(s)	Year	Title of the Research	Work done	Limitation
1.	[6]	2011	Proposed A New Approach To Solve The Decentralized Constrained Dynamic Multi-project Scheduling.	The author Develop Multi-agent System for Dynamic Multi-project Scheduling.	The algorithm useful only in project constrained environment
2.	[7]	2013	An efficient Resource Management and Scheduling Technique for Fault Tolerance in Grid Computing.	The presented techniques are based on queues based polices and are easy to implement.	This method is not very effective in mapping jobs to resources.
3.	[8]	2002	Eight Agent Based Algorithms for Solving Multimode Resource Constrained Project Scheduling Problem.	The work presented is based on small number of agents.	The algorithm is restricted to work with only eight agents
4.	[9]	2011	An Improved Backfilling Algorithm: SJF-BF.	Proposes An Improved Backfilling Algorithm Based On SJF.	The Algorithm works only with SJF policy.
5.	[10]	2013	Intelligent Agent Based Grid Resource Management System.	Agents are used in each computing node for scheduling	This work uses easy backfilling algorithm.
6.	[11]	2014	An efficient Job scheduling Approach in Grid Environment.	This work uses EGDF Method to fill biggest gap.	This work is used to fill earliest biggest gap only.
7.	[12]	2012	Agent Based Approach To Grid Scheduling Problem.	This work proposes communication mechanism between two or more agents	The algorithm is developed and used in Auvergrid only.

III. Process of Job execution in Grid Computing

Computer grids are systems containing heterogeneous, autonomous and geographically distributed nodes that are capable of executing jobs by the distributed nodes.

To illustrate how jobs are executed or processed in a grid environment, consider a grid system with five computing resources (c_1, c_2, c_3, c_4, *and* c_5), four distributed nodes (n_1, n_2, n_3, *and* n_4) and a job *S*, arrived at the Grid Information System (GIS) is divided into sub jobs, s_1, s_2, s_3, *and* s_4; then the grid information system identifies the interrelationships between the subtasks and their requirements in terms of computing resources and data access needs. Once these are identified, a network of distributed nodes and computing resources available is formed based on dependencies.

GIS will look for the available nodes to execute these subtasks by individually sending them to the distributed nodes in form of injective mapping (one-to-one) to increase efficiency. Some jobs would be executed immediately after receiving by a particular node and some might wait until some other jobs finished execution to serve as the input to the pending jobs for execution. The Fig. 1 below shows that subtask s_3, requires the result obtained as input after execution of subtask s_1 and subtask $s_2$and s_4 requires as input data the results obtained after s_3 is executed. Subtask s_1 is executed by node n_1, to execute subtask s_2 two distributed nodes are in contribution n_2 and n_3 to complete the task, execution of subtask s_3 does not need access to any node and subtask s_4 requires access to node n_4. The scheduler of the GIS assign subtask s_1 to computing resources (present in the distributed nodes) c_3,s_2 toc_1, s_3 to c_4, s_4 to c_2 and no any subtask is assign to c_5.

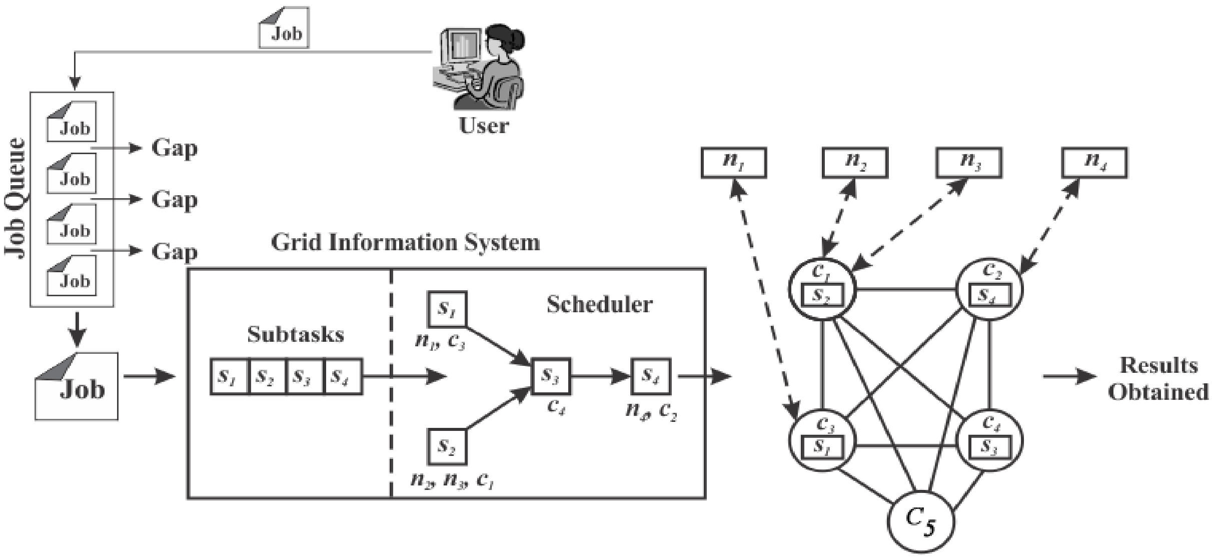

Figure1: Scheduling and job execution in grid system

The grid system illustrated in Fig. 1is usually very large and complex system, and the link connecting various elements of the network normally a virtual link over a long topological or geographical distance. When all subtasks that constitute the job are executed or processed, the result will be send to result collection center, and then turned to the user as a service requested.

IV. Scheduling in Grid System

Grid computing as a promising attractive distributed computing paradigm that aim at converging heterogeneous and geographically distributed computing resources to be shared among users, efficient scheduling is need to achieve the promising potentials of grid system. Compared to traditional systems for distributed environment, such as clustering computing, Grid scheduling systems have to take into account diverse characteristics of both various Grid applications and various Grid resources [3].

Grid environment contains a lot of computing resources that can be shared and used among users. The grid resources can be allocated or deallocated based on the availability of the resource; this is as a result of the uncertainty of the presence of a particular node present in the grid. A node can join and leave at any time, due to the dynamic nature of grid system resources availability, the grid management system gives higher priority to local service request and then consider the external service request for execution.

After client sent a job for execution it will remain in the queue, it's the function of grid information system to choose which node is suitable for execution of such a job. Since, the grid information system contains information about all the grid resources such as scheduler, the total number or resources available, CPU capacity, memory, bandwidth, current demand on every node etc. There are two kind of scheduler in grid environment:

- ***Local Scheduler:*** It's also called a scheduler; it's responsible for scheduling of jobs and managing resources at a particular node. This scheduler it's reside in a node that is assigned or chosen to execute a jobby the grid scheduler.
- ***Global Scheduler:*** It's also called Grid Scheduler, which is responsible for selecting appropriate local site and mapping of jobs on to the selected site or domain [13].

Scheduling can also be classified into static and dynamic scheduling. In static scheduling, before execution the jobs are assigned to the suitable machines and those machines will continue executing those jobs without interruption. In dynamic scheduling, the rescheduling of jobs is allowed [14]. The jobs executing can be migrated based on the dynamic information about the work load of the resources [15].

V. Propose Scheduling Algorithm

Algorithm: - Efficient Gap Filling Scheduling Algorithm.
Input: - Jobs <1,2,………,m>
Output: - Efficient scheduling through gap(s) elimination

Let A_r = Available resources
h_n = Total number of holes (in dynamic array)
j_n = Total number of jobs in the queue
if $A_r >= j_n$ then
map j_i to A_{ri} injectively
Using FCFS
Else
For i =1 to j_n-1
If there exist h_x between j[i] and j[i +1]
put h_x to h_n [i]
end for
end else
end if
for j = 1 to h_n -1
for i =1 to j_n – 1
if job_i.size can fit h_n[j] then
assign job_i to h_n [j]
else
i ++
end else
end if
end for
j ++

24. end for

The Algorithm works by checking the available number of resources that exist in the system and the total number of jobs in the queue pending to be allocated or executed; therefore, it checks if the available resources is the equals to the number of pending jobs then it will allocate the jobs to the available resources injectively (one-to-one). But if the resource cannot accommodate all the jobs available in the queue, then it will check the available gaps that exist in the job queue and store them in a dynamic array; then by considering the jobs in FCFS order then it compares each job with the stored gaps, until when the job find the gap that can accommodate it.

VI. Conclusion

Grid environment contains a lot of computing resources that can be shared and used among users. The grid resources can be allocated or deallocated based on the availability of the resource; this is as a result of the uncertainty of the presence of a particular node present in the grid. In this paper we highlight how job is schedule and process in grid system,and to make Scheduler efficiently works we proposed and algorithms that fill in the gaps that exist between the jobs in the job queue.This research is of important to enhance the promising potentials that grid technology provides to both clients and grid service providers. However, it will also help researchers who have interest in research on grid scheduling or grid computing at large.

Reference

[1]. Angulo, D., Foster, I., Liu, C., and Yang, L. (2002). Design and Evaluation of a Resource Selection Framework for Grid Applications, *Proceeding of IEEE International Symposium on High Performance Distributed Computing (HPDC-11)*, Edinburgh, and Scotland.

[2] GeorgianaM. (2011) Grid Computing Technology, *Database Systems Journal* vol. II, No. 3/2011

[3] Yanmin Zhu, Lionel M. Ni, (2013) *A Survey on Grid Scheduling Systems*, Technical Report SJTU_CS_TR_200309001, Department of Computer Science and Engineering, Shanghai Jiao Tong University, 2013.

[4] Suganya G. (2014) An Efficnet Job Scheduling Approach in Grid Environment Using Biggest Hole Priority Algorithm. International Journal of Computer & Organization Trends – Volume 9 Number 1 – Jun 2014

[5] Blythe, J., Deelman, E., Gil, Y., Kesselman, C., Agwal, A., Mehta, G., Vehi, K., (2003) The Role of Planning in Grid Computing, *in Proceeding of international conference on AI planning and Scheduling, (ICAPS)*. Treanto (Italy).

[6] Adhau, S., and Mittal, M. L. (2012) A multi-agent based approach for dynamic multi-project scheduling, *International Journal of Advanced Operations Management,* vol. 3, No. 3/4, pp 230–238

[7] Deeptanoy et al (2013) An efficient Resource Management and Scheduling Technique for Fault Tolerance in Grid Computing *in Cracow Grid Workshop '05 Proceedings. Academic Computer Centre CYFRONET AGH,* Cracow, Poland.

[8] Knotts et al (2002) Eight Agent Based Algorithms for Solving Multimode Resource Constrained Project Scheduling Problem.*IEEE Trans. Vol. 22(2), pp 187-189.*

[9] Mishra S., and Kushwah, D.S. (2011) An Improved Backfilling Algorithm: SJF-BF. *Int. J. on Recent Trends in Engineering & Technology*. Vol. 05 (1), pp. 8-16.

[10] Muthuchelvi, S. and Anandha, R. (2013) Intelligent Agent Based Grid Resource Management System. *Proceedings of the 8th International Conference of Distributed Computing Systems,* pages 104-111.

[11] Suganya (2014) An efficient Job scheduling Approach in Grid Environment*, in Proceedings of the 25th IASTED International Multi-Conference, Parallel and Distributed Computing and Networks*. Vol. 6(7), pp 22-34. Innsbruck, Austria.

[12] Victor, P., Solar M., Rojas J., Mendoza M., and Raul, M. (2012) A Multiagent Based Approach To Grid Scheduling Problem.*Clei Electronic Journal*, vol 15, no. 2, p5, pp 190-105.

[13] Foster, I., Jennings, N., Kesselman, C. (2004), Brain Meets Brawn: Why Grid and Agents Need Each Other?, *Autonomous Agents and Multiagent Systems, InternationalJoint Conference on,* Vol.1(8) pp–15-31.

[14] Khushboo Y., Deepika J. and Ramandeep S. (2013), Job Scheduling in Grid computing. *International Journal of Computer Applications,* Vol-69- No. 22.

[15] Chtpen, (2005), Dynamic Scheduling in grids system. Six Firw, PhD Symposium, Faculty of Engineering, Ghent University.