Scheduling Algorithm design for grid computing

Ravika Goel
Department of Computer Science & Engineering
GJUS&T, Hisar, India

Davinder Singh
Department of Computer Science & Engineering
GJUS&T, Hisar, India

Minakshi
Department of Computer Science & Engineering
GJUS&T, Hisar, India

Abstract - A grid computer is multiple numbers of same classes of computers clustered together. A grid computer is connected through a super fast network. Geographically distributed resources cooperate to solve big problems, is called grid computing. Grid computing, is distributed computing model that is provides easy access to heterogeneous resources that are geographically dispersed. Today, due to heterogeneous grid resources that belong to different organizations and locations with different access policies and terms of workload dynamics are inherent; the use of this type in grade sharing, selection and gathering resources computing has become popular. Scheduling in grid computing systems that are normally non-concentrated is important in military, mobile medical and laboratory control systems.

Scheduling in grid computing is an inconclusive issue, so cannot used be certain of the algorithms to improve scheduling. In traditional scheduling approaches at grid computing, scheduling time to complete tasks is considered as the most important parameter, while the timing of the economic schedulers should also implement time jobs, cost of resource use is considered. The algorithm proposed in this paper that called pseudo code algorithm, the execution time required for all jobs using all three algorithms(Shortest Job First, First Come First Serve, Round Robin) is plotted which is less than the execution time required for all jobs using only one algorithm (Shortest Job First).

Keywords: Grid computing, schedule, pseudo code algorithm.

I. INTRODUCTION

A grid computing infrastructure hardware and software that provided access to high level computational capabilities into a reliable, consistent, pervasive and inexpensive offers [1]. Grid is a shared environment that implemented with the establishment of lasting service and standards, these services support create and share resources to distribute. Computational grid environment are suitable for solving problems that are long and heavy computation. In this environment, resources are geographically distributed, but in terms of logical are seen as a source [2]. The main goal of grid is providing services with high reliability and lowest cost for large volumes of users and support group work and the most important issue in grid computing are resource management and control, reliability and security. Today increased efficiency of grid is an important issue. To increase the efficiency of grid a properly and useful scheduling is needed. Unfortunately, the dynamic nature of grid resources and the demands of different users are causing complexity in the scheduling grid [3, 5]. Scheduling process in grid can be organized in three stages: resource discovery, resource selection.

II. RELATED WORK

A dynamic job grouping-based scheduling algorithm [6, 7] groups the jobs according to MIPS (Million Instructions per Second) of the available resources. The proposed job scheduling strategy takes into account: (i) the processing requirements for each job.
(ii) The grouping mechanism of these jobs, known as job grouping, according to the processing capabilities of available resources.
(iii) The transmitting of the job grouping to the appropriate resource.
Pseudocode of the Algorithm:
1. Groupedjobi:=0;
2. Sort(JobList_size) in Ascending order according to MI and

assign ID.
3. Resources selected by HST;
4. For i:=0 to ResourceList size-l Do
5. (Groupedjobi)MI:= 0;
6. Ri MI := ResourceListi MIPS * Granularity size;
7. Ri BW:= baud Rate * Tcomm
8. For j:=0 to JobList_size-1
9. while (j <=Joblist_ size-1)
10. {
11. Groupedjob:= Groupedjob+ Jobj ;
12. if(((Groupedjob)MI <= Ri MI) &&(Groupedjob)MS <= Ri MS)

III. PRESENT WORK

1. Implemented three scheduling algorithm (Shortest Job First, First Come First Serve, Round Robin) for jobs and send to the queue.
2. Implemented Shortest Job First algorithm along with all three algorithms for performance analysis.

Algorithm:

1. Find available parallel computing resources from the defined parallel configuration (job manger).
2. Create job object in the scheduler and in the client.
3. Create new task in job.
4. Calculating the minimum MI among the jobs.
5. Sort the jobs in array with minimum MI and the Shortest time job is executed first
6. The time slot is defined for all the jobs to be executed and the Job with higher MI above time slot allocated to it executed using round robin algorithm.
7. The time required for each job using all three algorithms is plotted on the graph
8.The execution time required for all jobs using all three algorithms(Shortest Job First, First Come First Serve, Round Robin) is plotted which is less than the execution time required for all jobs using only one algorithm (Shortest Job First).

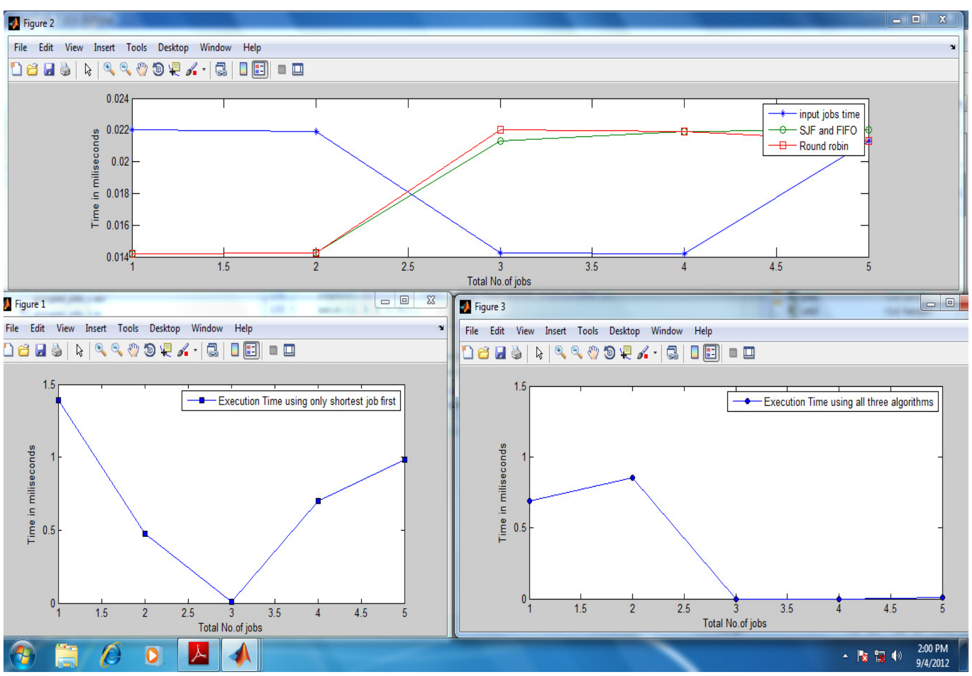

Figure 1: Execution time using only shortest job first, this graph is only for performance test purpose.

Figure2: Individual times of all three algorithms for each job, the blue line indicate the input jobs times, red line is for round robin algorithm, green is for SJF AND FIFO

Figure3: Execution time required for all three algorithm, compared with shortest job first and figure 2 (Final result)

IV. CONCLUSIONS AND FUTURE SCOPE

In future, this work can be extended to design a high performance parallel scheduler for grid system to realize a real grid environment.

Optimization approach to scheduling tasks in grid environments, in addition to the time schedule parameters should is considered execution time used in the resource scheduling process.

In most scheduling algorithms, the user should be to determine one limitations of time for the scheduler. That this story, although the direction is good, but for the user is not able to provide a suitable area without the knowledge of market conditions, will create a problem.

In this paper, a scheduling algorithm is proposed considering the dependencies between tasks and execution time between tasks in grid environments.

REFERENCES

[1] Foster and C. Kesselman (editors), "The Grid: Blueprint for a Future Computing Infrastructure", Morgan Kaufmann Publishers, USA, 1999.

[2] Baker M., Buyya R., and Laforenza D., "The Grid: International Efforts in Global Computing", Proc. Of International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet, Rome, Italy, 2000.

[3] Li, X.L.: A New Intelligent Optimization-Artificial Fish Swarm Algorithm. Doctor thesis,Zhejiang University of Zhejiang, China ,2003.

[4] Jiang, M.Y., Yuan, D.F.: Wavelet Threshold Optimization with Artificial Fish Swarm Algorithm. Proc. of IEEE International Conference on Neural Networks and Brain, Beijing China, 2005,pp.569 -572.

[5] Xiao, J.M., Zheng , X.M., Wang , X.H.: A Modified Artificial Fish-Swarm Algorithm. Proc. of IEEE the 6th World Congress on Intelligent Control and Automation, Dalian China, 2006,pp.3456-3460.

[6] Zhang, M.F., Cheng, S., Li, F.C.: Evolving Neural Network Classifiers and Feature Subset Using Artificial Fish Swarm. Proc. of IEEE International Conference on Mechatronics and Automation, Luoyang China, 2006, pp.1598-1602.

[7] Shan, X.J., Jiang, M.Y.: The Routing Optimization Based on Improved Artificial Fish Swarm Algorithm. Proc. of IEEE the 6th World Congress on Intelligent Control and Automation, Dalian China, 2006, pp.3658-3662.