Grouping-Based Job Scheduling Model In Grid Computing

Vishnu Kant Soni, Raksha Sharma, Manoj Kumar Mishra

Abstract—Grid computing is a high performance computing environment to solve larger scale computational applications. Grid computing contains resource management, job scheduling, security problems, information management and so on. Job scheduling is a fundamental and important issue in achieving high performance in grid computing systems. However, it is a big challenge to design an efficient scheduler and its implementation. In Grid Computing, there is a need of further improvement in Job Scheduling algorithm to schedule the light-weight or small jobs into a coarse-grained or group of jobs, which will reduce the communication time, processing time and enhance resource utilization. This Grouping strategy considers the processing power, memory-size and bandwidth requirements of each job to realize the real grid system. The experimental results demonstrate that the proposed scheduling algorithm efficiently reduces the processing time of jobs in comparison to others.

Keywords—Grid computing, Job grouping and Job scheduling.

I. INTRODUCTION

GRID computing, originally motivated by wide-area sharing of computational resources [1], has evolved to be mainstream technologies for enabling large-scale virtual organization [2]. In the late1990's, a complex computation environment, grid computing, had come out. The aspiration to share processing resources between many organizations to resolve large scale problems has provoked computational grids [3].The term "Grid" refers to systems and applications that integrate resources and services distributed across multiple control domains [4].Computational grids provide large-scale resource sharing, such as personal computer, clusters, MPPs, Data Base, and online instructions, which may be cross-domain, dynamic and heterogeneous [5].

In a Grid computing environment, scheduler is responsible for selecting the suitable machines or computing resources in the Grid for processing jobs to achieve high system throughput, but there exist several applications with a large number of lightweight jobs [6]. Job scheduling with light weight gives low performance in terms of processing time and communication time. So to achieve high performance, jobs are to be scheduled in groups instead of

Authors are with School of Computer Engineering, Kalinga Institute of Industrial Technology, University, Bhubaneswar, India
(e-mail:vishnu.soni132, rakshasharma10, manojku.mishra05}@gmail.com)

light weight jobs. This paper mainly focuses on light weight job scheduling, how they are grouped and allocated to resources in dynamic environment. This paper is organized as follows. Section II, discusses related work, Section III the proposed model, Example and Algorithm, Section IV presents experimental evaluation and Section V gives conclusion and future work and lastly, the references.

II. RELATED WORK

In this section, some of the representative research works on job scheduling in Grid computing have been surveyed. A dynamic job grouping-based scheduling algorithm, group jobs according to MIPS of the resource. This model reduces the processing time and communication time of job, but the algorithm doesn't take the dynamic resource characteristics into account and the grouping strategy can't utilize resource sufficiently [6]. Scheduling framework for Bandwidth-Aware Job Grouping-Based strategy that groups the jobs according to MIPS and Bandwidth of the resource, but the groping strategy does not utilize resource sufficiently and consideration of bandwidth strategy is not efficient to transfer the job [7]. A Bandwidth-Aware Job Grouping-Based scheduling strategy that schedules the jobs according to MIPS and bandwidth of the resource and the model sends grouped jobs to the resource whose network bandwidth has highest communication or transmission rate, but this does not ensure that the resource having a sufficient bandwidth will be able to send the grouped jobs within the required time [8]. Grouping-based fine-grained job scheduling algorithm presents job scheduling algorithm that schedules the groups of jobs to resources according to MIPS and Bandwidth of resources. But the disadvantage of the algorithm is that preprocessing scheduling time of job scheduling is high, time complexity of the scheduling algorithm is high and finally, it does not pay any attention of the memory file-size [9]. To address some of the above mentioned problems, Grouping-Based Job scheduling model is presented in this paper.

III. PROPOSED MODEL

This study presents and evaluates an extension from Computational-Communication to Computational-Communication-Memory based Grouping Job Scheduling strategy. This strategy maximizes the utilization of Grid resources, reduces processing time of jobs and network delay to schedule and execute jobs on the Grid. The model converts

light weight jobs into coarse-grained job or grouped job according to the requirement jobs and resource capability.

A. Grouping Strategy

Grouping strategy is based on processing capability (in MIPS), bandwidth (in Mb/s), and memory-size (in Mb) of the available resources. Jobs are grouped according to the capability of the selected resource. Therefore, the following conditions must be satisfied:

$$\text{Groupedjob\_MI} \leq \text{Resource\_MIPS} * \text{Granularity size} \tag{1}$$

$$\text{Groupedjob\_MS} \leq \text{Resource\_MS} \tag{2}$$

$$\text{Groupedjob\_MS} \leq \text{Resource\_baud\_rate} * \text{Tcomm} \tag{3}$$

Where, MI (Million Instruction) is job's required computational power, MIPS (Million Instruction Per Second) is processing capability of the resource and Granularity size is user defined time which is used to measure total number of jobs that can be completed within a specified time, Groupedjob_MS is required Memory Size of group job, Resource_MS is the amount of Memory available at resource, Baud_rate is the bandwidth capacity of resource, and T_{comm} is the job's communication time.

Equation (1) required computational power of grouped job shouldn't exceed to the resource's processing capability. Eq (2) Memory-size requirement of grouped job shouldn't exceed to the resource's memory-size capability. In Eq (3) Memory-size of the grouped job shouldn't exceed to resource's transfer capability within a specific time period. These are the main factors in job grouping strategy that influences the way job grouping is performed to achieve the minimum job processing time and maximum resource utilization of the Grid system.

B. Grouping-Based Job Scheduling Model

The job scheduler is a service that resides in a user machine figure 1. Therefore, when the user creates a list of gridlets or jobs in the user machine, these jobs are sent to the job scheduler for scheduling arrangement. The job scheduler obtains information about the available resources from the Grid Information Service (GIS). Based on the information,

the job scheduling algorithm is used to determine the job grouping and resource selection for grouped jobs. The size of a grouped job depends on the processing requirement length expressed in MI, Bandwidth expressed in Mb/s and Memory-size requirement expressed in Mb. When the jobs are put into a group according to the selected resources, the grouped job is dispatched to resources for computation. Figure 2 depicts the architecture of the job scheduler used in the system. The system accepts total number of user jobs with specification such as Job_ID, Job_MI, Job_Memory-size and the available grid resources are specified by Resource_ID, Resource_MIPS, Resource_Bandwidth, Resource_Memory-size. Granularity-size and communication time are user defined in the grid environment (step 1-3). After gathering the details of user jobs and the available resources, the system

periodically selects Jobs in First come first serve order after sorting it into ascending order (step 4).
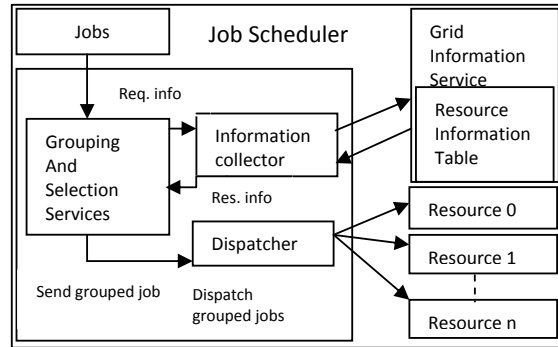

Fig. 1 Grouping-Based Job scheduling Model

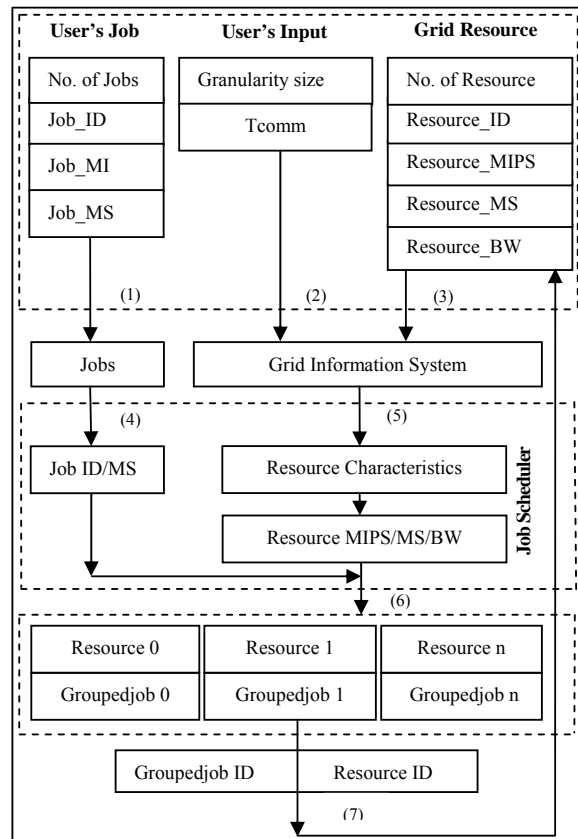### C. Architecture of the Grouping-Based Job Scheduler

Fig. 2 Architecture of the Grouping-Based Job Scheduler

The scheduler will then acquires resources in First come first serve order and multiplies the Resource_MIPS with the given granularity size and Resource_BW with communication time and resulting value is saved in grid information system (step 5). The jobs are then gathered or grouped according to the resulting total MI, Memory_size and Bandwidth of the resource (step 6). After grouping the jobs, the scheduler

submits the grouped job to the selected resources for job computation (step 7).After execution the jobs result goes to the user and resource is again available to the Grid and ready to execute the another job. This process continues till all the jobs are grouped and executed in the grid system.
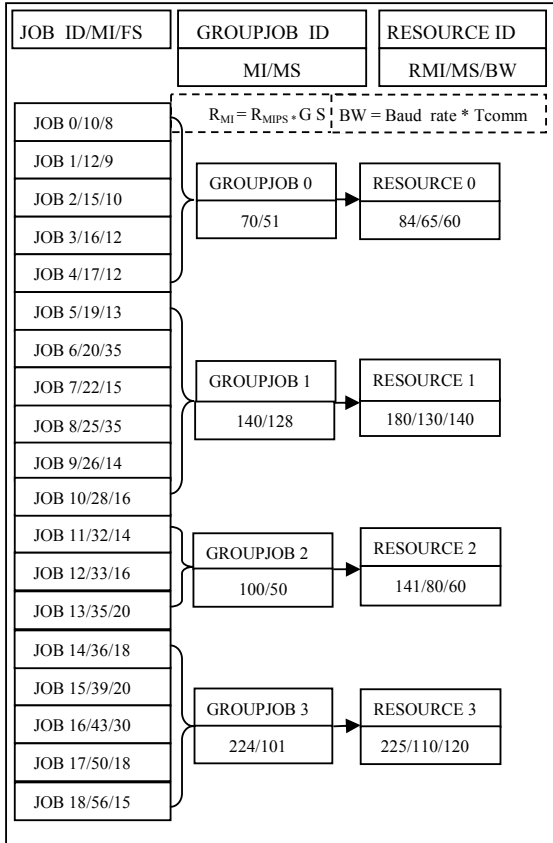
D. Example of the proposed model

Fig. 3 Example of the Job and scheduling strategy

E. Algorithm of the proposed model

1. Groupedjob$_i$:=0;

2. Sort(JobList_size) in Ascending order according to MI and assign ID;

3. Resource comes in FCFS order;

4.   For i:=0 to ResourceList_size-1 Do

5.   (Groupedjob$_i$)$_{MI}$:= 0;

6. $R_{i\ MI}$:= ResourceList$_{i\ MIPS}$ * Granularity size;

7. $R_{i\ BW}$:= baud_rate * T_{comm}

8.   For j:=0 to JobList_size-1 Do

9.   while (j $\leq$ Joblist_size-1)

10.   {

11.   Groupedjob:= Groupedjob+ Job$_j$;

12.   if(((Groupedjob)$_{MI}$ $\leq$ $R_{i\ MI}$) && (Groupedjob)$_{MS}$

13. $\leq$ $R_{i\ MS}$) && ((Groupedjob)$_{MS}$ $\leq$ $R_{i\ BW}$)))

13.   {

14.   j ++;

15.   }

16.   Else

17.   {

18.   Groupedjob:= Groupedjob - Job$_j$;

19.   j--;

20.   }

21.   Break;

22.   } // End while

23.   Create a new job with total MI less or equals to Resource MI;

24.   Assign a unique ID for the newly created Groupedjob;

25.   Place the Groupedjob$_i$ to Target ResourceList$_i$ for computation;

26.   Receive computed Groupedjob from ResourceList$_i$;

27.   i++;

28. Endfor;

29. End;

IV. EXPERIMENTAL EVALUATION

A. Experimental setup and comparison

GridSim [10] has been used to create the simulation environment. A simulation is conducted in heterogeneous environment where each cluster has machines with different characteristics such as MIPS, bandwidth and memory size. In this simulation, size of the granularity is taken as 10 seconds and Tcomm is 5 second for scheduling algorithm. The MIPS, bandwidth and memory size are the main factor taken to group the jobs according to the available capability of the selected resource. The processing time is taken into account to analyze the feasibility and to verify the improvement of proposed model over other scheduling strategies.

TABLE I
COMPARISION BETWEEN GBJS WITH AFJS AND DJGBSDA

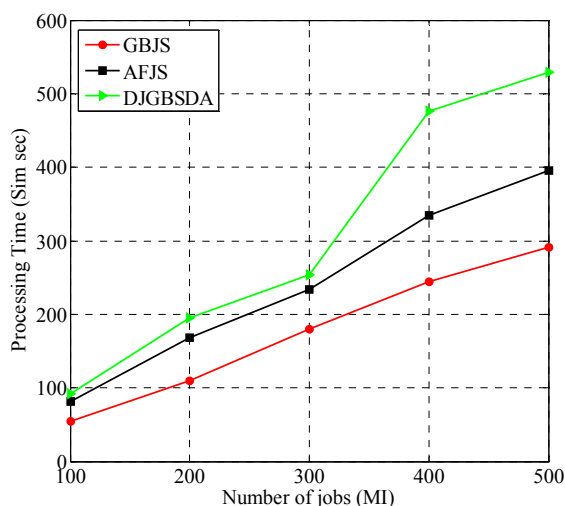| Number of jobs	Processing Time		
	GBJS	AFJS	DJGBSDA
100	55	81	92
200	109	168	195
300	180	234	254
400	244	335	476
500	291	396	529

Fig. 4 Processing Time of the algorithms

B. Experimental Result

Simulation is conducted to analyze and compare the processing time of jobs between proposed algorithm GBJS with AFJS [9] and DJGBSDA [6] algorithm. Result shows significant reduction in processing time of GBJS algorithm when number of jobs increases in comparison with other algorithms.

V. CONCLUSION AND FUTURE WORK

In order to reduce processing time and utilize grid resources sufficiently, Grouping-Based Job scheduling model has been proposed taking memory constraint into account. The simulation environment has shown that the proposed model is able to achieve the mentioned objectives in grid environment. Results of the comparative study shows that the proposed GBJS gives better performance than AFJS and DJGBSDA in terms of processing time. The proposed model provides a real time grid computing environment and reduces the waiting time of the grouped jobs.

In future, this work can be extended to design a parallel scheduler in grid system to realize its performance.

REFERENCES

[1] I. Foster and C. Kesselman, *The Grid: Blueprint for a New Computing Infrastructure*, Morgan-Kaufmann, 1998.

[2] I. Foster, C. Kesselman and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", *International Journal of Supercomputer Applications*, Vol. 15, No. 3, 2001.

[3] Berman, F., Fox, G. and Hey, A.: Grid Computing Making the Global Infrastructure a Reality. London,Wiley, 2003.

[4] Foster, I. and Kesselman, C. Computation Grids. Foster, I. and Kesselman, C. eds. The Grid: Blueprint for a New Computing Infrastructure, Morgan Kaufmann, 1999, 2-48.

[5] Ian Foster and Carl Kesselman, "The Grid: Blueprint for a New Computing Infrastructure," Elsevier Inc., Singapore, Second Edition, 2004.

[6] N. Muthuvelu, Junyan Liu, N.L.Soe, S.venugopal, A.Sulistio, and R.Buyya, "A dynamic job grouping-based scheduling for deploying applications with fine-grained tasks on global grids," in Proc of Australasian workshop on grid computing, vol. 4, pp. 41–48, 2005.

[7] Ng Wai Keat, Ang Tan Fong, "SCHEDULING FRAMEWORK FOR BANDWIDTH-AWARE JOB GROUPING-BASED SCHEDULING IN GRID COMPUTING", Malaysian Journal of Computer Science, Vol. 19, No. 2, pp. 117-126, 2006.

[8] T.F Ang, W.K.Ng, T.C Ling, "A Bandwidth-Aware Job Grouping-Based Scheduling on Grid Environment", Information Technology Journal, Vol .8, NO.3, pp. 372-377, 2009.

[9] Quan Liu, Yeqing Liao, "Grouping-based Fine-grained Job Scheduling in Grid Computing", IEEE First International Workshop on Educational technology And Computer Science, Vol.1, pp. 556-559, 2009.

[10] Buyya, R. and M. Murshed, 2002. GridSim: A toolkit for the modeling and simulation of distributed management and scheduling for grid computing. Concurrency COmput: Practice Exp., 14: 1175-1220.

Manoj Kumar Mishra, 2005 holds M.Tech Comp. Science from KIIT, University, Bhubneswar, India. His area of interest includes Grid Computing, Parallel and Distributed computing, Multicore and cluster computing.

Raksha Sharma, 2005 holds B.E (IT) degree from Govt. Engg. Collage, Bilaspur (C.G). She is currently pursuing her M.Tech Computer Science and Engineering from Kalinga Institute of Industrial Technology, Bhubaneswar (Odisha) India. Her area of interest includes Grid Computing, Distributed and Parallel Computing.

Vishnu Kant Soni, 2005 holds B.E (IT) degree from Kirodimal Institute of Technology, Raigarh (C.G). He is currently pursuing his M.Tech Computer Science from Kalinga Institute of Industrial Technology, Bhubaneswar (Odisha) India. His area of interest includes Grid Computing, Distributed and Parallel Computing, Network Security, Computer Network.