An ACO Algorithm for Scheduling Data Intensive Application with Various QOS Requirements

S.Aranganathan and K.M.Mehata
Department of CSE
B.S. Abdur Rahman University
Chennai – 600048, Tamilnadu, India

ABSTRACT

Grid computing is rapidly growing in the distributed heterogeneous environment for utilizing and sharing large scale resources to solve complex scientific problems. The main goal of grid computing is to aggregate the power of widely distributed resources and provide non trivial QOS services to the users. To achieve this goal, an efficient grid scheduling algorithm is required. The problem of scheduling on data intensive application in terms of QOS requirements is challenging and it significantly influences the performance of grids. The existing algorithms for scheduling the data intensive application can only tackle the problems with either system centric or application centric. This paper aims to propose a new algorithm based on ant colony optimization to schedule the data intensive application which combines both application centric and system centric benefits. We formulate the problem and simulation results demonstrate the effectiveness of proposed scheduling algorithm.

Key words: Data intensive scheduling, ant algorithm, Pheromone intensity

1. INTRODUCTION

The next generations of scientific applications are in diverse domains such as high-energy physics, molecular modeling, and earth sciences [1]. They involve the production of large data sets from simulations or large-scale experiments. Analysis of these data sets and their dissemination among researchers located over a wide geographic area requires high capacity resources such as super computers, high bandwidth networks and mass storage systems. Grid computing paradigm unites geographically distributed and heterogeneous computing, storage and network resources and provides a unified, secure and pervasive access to their combined capabilities. It enables sharing, exchange, discovery, selection and aggregation of distributed heterogeneous resources such as computers, databases and scientific instruments. It creates virtual organizations by allowing geographically distributed communities to pool resources in order to achieve the common goal.

The data grid aims to combine high end computing technologies, with high performance networking and wide area storage management techniques. Data is distributed over sites in grid [2]; hence job scheduling for data grids must not only consider computational loads at each grid node but also the distribution of data required by the job. Realizing such Grids requires overcoming challenges in security, resource management, scheduling, and data management. However, an important issue in grid computing is to overcome the complexity of scheduling distributed resources [3]. It is very difficult to find an optimal resource allocation method that minimizes the cost and time. So the job allocation and scheduling is the fundamental issue for achieving high performance in the data grid environment.

The Grid users and resource providers have different motivations when the join the Grid. These incentives presented by objective functions in scheduling. The objective functions can be classified into two categories: application centric and system centric. The scheduling algorithm adopting an application centric objective functions aim to optimize the performance of each individual application[4]. Most of the current grid applications concerns makespan and economic cost. Makespan is time spent from beginning of the first task in the job to end of the last task of the job. The economic cost of an application is amount of payment for resource utilization. The scheduling algorithm adopting system centric objective functions aims to optimize the performance of system. System centric objectives are related to resource utilization and throughput. Resource utilization refers percentage time the resource is busy.

The rest of the paper is organized as follows: section 2 gives related work. Section 3 presents an ant algorithm with scheduling scheme in detail. Section 4 evaluates the performance of our scheduling scheme. Section 5 offers conclusion.

2. RELATED WORK

In Grid environment, resources are distributed, heterogeneous, dynamic and instability [5]. The objective of this work is to propose a scheduling algorithm to run in the grid simulation environment to increase the performance in the highly congested network. The ant algorithm introduced by Dorigo M. in 1996 [6] is a new heuristic algorithm and is based on the behavior of real ants. The ACO algorithm used to simulate the cooperative behavior of ant colonies. When blind insects like ants look for food, the moving ant lays a pheromone on the ground, thus marking the path it followed by a trail of this substance. While an isolated ant moves essentially at random, an ant encountering a previously laid trail can detect it and decide with high probability to follow it, thus reinforcing the trail with its own pheromone. A collective behavior emerges as the ants are guided by pheromone intensity. Hence the more ants following a trail, the attraction for that trail is increased. Above observations inspired a new type of algorithm called an ant algorithm or ant systems, presented by Dorigo, M. and Gambaradella L. in 1996.

In general, the grid scheduling is divided into three phases namely resource recovery, scheduling and executing [7]. The second phase aims at finding the best match between the set of jobs and available resources which is a NP-hard (Non deterministic polynomial time) problem. For example, mapping of 25 jobs into 10 resources produces 10^{25} possible mappings. The ACO algorithm, which is population based approach, has been successfully applies to many NP hard optimization problems [8]. We utilize the characteristics of ant algorithm mentioned above to schedule the job. We can carry on new job scheduling by experience depending on the result in the past job scheduling. It is very helpful within the grid computing for scheduling the data intensive application.

The data intensive grid has dynamic and unpredictable characteristics due to the following reasons.

- Data is distributed over sites in data grids and size is in tera bytes and peta bytes.
- Computational performance of each resource varies from time to time.
- The network connection may be unreliable. Bandwidth may increase or decrease at any time.
- The resources may join or leave the grid at any time.
- The resource may become unavailable without any notification.

The execution time of a job on different machines will have different range of completion time. So the scheduling of the grid resources, to minimize the job execution time is a major issue. In Grid scheduling, decisions are often made on the basis of being data intensive or computation intensive. In data intensive situation, jobs may be pushed to the data and computation intensive situation data may be pulled to the job. This kind of scheduling in which there is no consideration of network characters can lead to performance degradation in the grid environment and may result in large processing queues and job processing delays due to site overhead. Mcclatchey and Anjum et. al [9], [10] in their paper describe a data intensive network aware meta-scheduling approach which takes into account the processing power and network characteristics while making the scheduling decisions across multiple sites. Through a practical implementation on a grid testbed it is shown that queue length and execution times of data intensive jobs can be significantly reduced with their DIANA schedulers.

Kavitha Ranganathan et al. [11] and Xu, L et al [12] describe scheduling considerations in data grid environment, they describe a scheduling frame work which both address the job and data scheduling. Their simulation results show that job scheduling and data scheduling handled with separately for achieving better performance.

Jia Yu and Rajkumar Buyya et al [13] propose a novel QOS – based workflow scheduling algorithm that minimizes the cost of execution while meeting deadline. It emphasizes more on QOS constraints of jobs, such as deadline and budget.

Xu, Zh., Hou, X at al [14] propose a task scheduling algorithm based on ant algorithm in grid computing. The algorithm achieves good performance results in terms of response time, average resource utilization. However, it emphasizes the scheduling of jobs that are computation intensive and is not suitable for scheduling data intensive application.

The QOS based algorithm for job allocation and scheduling [15] takes resource failure into consideration and provides re-allocation mechanism. The qualities of the resources like processing power and communication ability are expressed in this paper. They propose a new instance of ant algorithm to calculate and update the resource quality. However, the algorithm works well in stable conditions, the performance will be degraded when the network condition become unstable.

Srikumar venugopal et al[16] have presented a model for executing jobs on data grids which takes in to account both processing and data transfer costs. They have also presented an algorithm which creates a resource set, consisting of both compute and data resources, that promises the least cost with least time.

3. ACO ALGORITHM FOR JOB SCHEDULING

The Ant Colony Optimization (ACO) has been shown to be an effective strategy for several problems closely related to scheduling jobs in data intensive application, and it seems to be an effective strategy in this domain. The ants build their solution using both information encoded in the pheromone trail and also problem-specific information in the form of a heuristic.

1. **Initialization of algorithm**
   All the pheromone values and parameters are initialized at the beginning of the algorithm.
2. **Solution construction**
   The N artificial ants are used in the algorithm. The N ants set out to build N solutions to the problem based on pheromone and heuristic values using the selection rule.
3. **Pheromone updating**
   After all ants have completed their solution at the end of the each iteration, the pheromone values are updated.

3.1 Defining pheromone and heuristic Information

The following notions are used for formulate the mathematical model

T_i – deadline given by the user i
B_i - budget of the grid user i
P_j – price unit of resource j
W_j – total workload of resource j
C_j – current workload of j
TR_i – time required completing the job at resource j

When a resource j join the grid system, it should submit his quality factors in set S

$$\tau_j(0) = \sum_{i=1}^{n} s_i \cdot f_i \cdot \quad \sum_{i=1}^{n} f_i = 1 \qquad (1)$$

Where

τ_j (O) – Innate performance of resource j
f_i – Intensive weightage factors
s_i = system centric parameters

In any ACO algorithm we must first determine what information we will encode in the pheromone trail, which will allow the ants to share useful information about good solutions. The pheromone value τ_j (t) represent the favorability of scheduling a particular job i onto a particular resource j at time t.

The Probability that task is allocated to resource j within a job is computed using the formula

$$P_j(t) = \frac{[\tau_j(t)]^{\alpha}.[\tau_j(0)]^{\beta}}{\sum_{\mu}[\tau_{\mu}(t)]^{\alpha}.[\tau_{\mu}(0)]^{\beta}} \qquad (2)$$

Where

τ_j (t) – is the pheromone intensity on the path from scheduler centre to resource j at a time t;
τ_j (O) – is the innate performance of resource j
α – is the importance of pheromone
ß – is the importance of resource innate attribute
µ – is the resource that is available for the job

The paper [17] describes a formula to update the pheromone intensity on the path from schedule center to corresponding resource as follows

$$\tau_j^{\,new} = \rho\, \tau_j^{\,old} + \Delta\tau_j \qquad (3)$$

Where

1. τ_j new is the change of pheromone on path from the schedule
   center to resource j.
2. ρ is evaporation of pheromone($0\leq\rho\leq1$).
3. When a task is allocated to resource j, $\Delta\tau_j$ = -K, K is the quality that the task consumed.
4. When a task is canceled and the data resource it uses is still in service, $\Delta\tau_j$ = K It is used for restoring the resource quality.
5. When a task successfully returns from resource j, $\Delta\tau_j$ = Ce · K, Ce ≥1, Ce is the encouraging factor.
6. Otherwise, if task fails in returning from resource j, $\Delta\tau_j$ = Cp ·
   K, Cp is the punishing factor and $0\leq Cp\leq1$.

The following steps will be very useful to find optimal values for these parameters. These steps are added in the algorithm to adjust automatically based on the resource status. The Environment variable B_j is calculated as follows

$$B_j = \quad ((Resource\ failure\ rate\ (\%) + Network\ stability(\%))/2)/100$$

Where

$$Network\ stability = \frac{current\ bandwidth}{Required\ bandwidth}$$

3.2 Procedure for adjusting the parameter

```
Procedure ParaAdj
Begin
Compute  Bj
  If (Bj >= 0.9)
  Assign ß = 0.5, α = 1 -  ß
    Else If (Bj >= 0.8 and Bj < 0.9)
    Assign ß = ß + 0.1, α = 1 -  ß
      Else If (Bj >= 0.6 and Bj < 0.8)
      Assign ß = ß + 0.2, α = 1-  ß
         Else If (Bj < 0.6)
        Assign ß = ß + 0.3, α = 1-  ß
        End if
      End if
    End if
  End if
End ParaAdj
```

3.3 Resource Allocation

When the user submits the job with relevant QOS parameters, such as cost, deadline, the job enters into job pool and lineup. The job is divided into several tasks. The task is allocated to the resource according to the formula (2) in ACO algorithm subject to following conditions. Deadline given by the user (Ti) is less than total expected execution of all task.

1.Budget given by the user (Bi) is less than cost associated with all the resources which is used to execute the tasks.
2.Total workload (Wj) is less than current workload(Cj).

The formula [3] is used to update the pheromone intensity.

4. EXPERIMENTAL TESTING

All the data intensive jobs are more depend on storage, network capacity and computation. When a job is submitted to grid scheduling system, the scheduling system has the responsibility of selecting the suitable resource and then to manage the job execution. The decision of which resource should be used is the outcome of scheduling algorithm. The following simulation parameters are used to evaluate the algorithm

Simulation Parameters

Simulation parameters	**Value**
Number of Grid sites	150
Number of nodes in each site	30
Bandwidth capacity	200Mbps to 2Gbps
Processing capacity (MIPS)	512 to 1024
Total Number of providers	300
Total Number of users	800
Budget of users (units)	100 to 1000
Price of resources (units)	100 to 2000
Number of file	100 to 15000
Size of each file	100 Mb to 4 Gb

The α determines the intensity of the pheromone information which is used by the ants to build their solution. Pheromone is critical for success of the algorithm and having experimented. α can be set to a higher value in high steady environment, or a smaller value in other environment. ß determines the importance of heuristic information which is used by the ants to build their solution. Again, different values were tested. The high ß value is necessary to get good solution in dynamic environment. The heuristic intensive weightage factors and parameters α and ß and how their variation affect the performance has been studied in detail. Their values are decided by stability of the network and resource condition. The quality and consistency of the networks are among the most important factors since the grid can be subject to failure if network does not perform well. The algorithm is modified such way that the performance is improved by varying the parameters.

The adaptive algorithm was simulated using GridSim[18] simulation tool kit and was found to be working efficiently. To obtain the good results under various loads of resources, varied set of jobs are submitted at different intervals. We employ the non pre-emptive approach in our adaptive scheduling algorithm and once a job starts execution we do not move it since check-pointing and restart are very expensive operation in data intensive application. Intensive weightage factors I1 and I2 added in the equation (1) for making good scheduling decision and prioritize the quality of service factors. The heuristic value needs to be driven by these intensive weigtage factors. These values gave optimum results and tested with various scenarios.

Results from the set of experiments which have been conducted with the adaptive scheduling using simulation to check the behavior of the algorithm is shown here. We have employed the network interface in the simulation to obtain the network performance information since it provides detailed dynamic information about the status of network. The results are in minutes.

We use two metrics to compare the performance of proposed algorithm with application centric algorithm proposed by sri kumar venugopal, and system centric algorithm proposed by Xiangang Zhao, Bai Wang et al. Economic cost, makespan. Economic cost is minimum cost spent for executing the job. Makespan is minimum time spent from the beginning of the first task in a job to end of the last task of the job. The experiment was conducted with different input values to study the performance of proposed algorithm.

Figure I. Number of jobs versus time

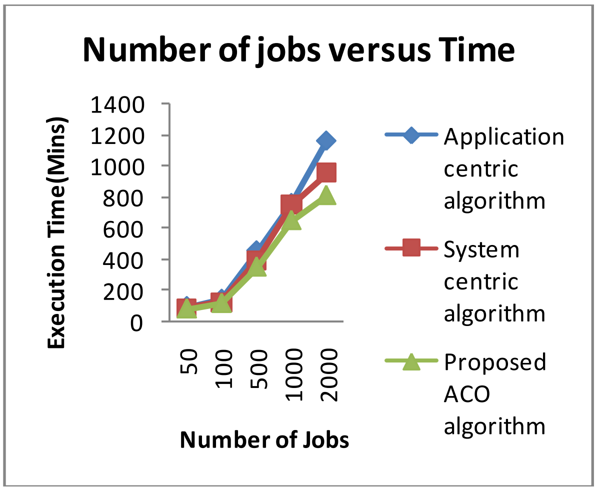
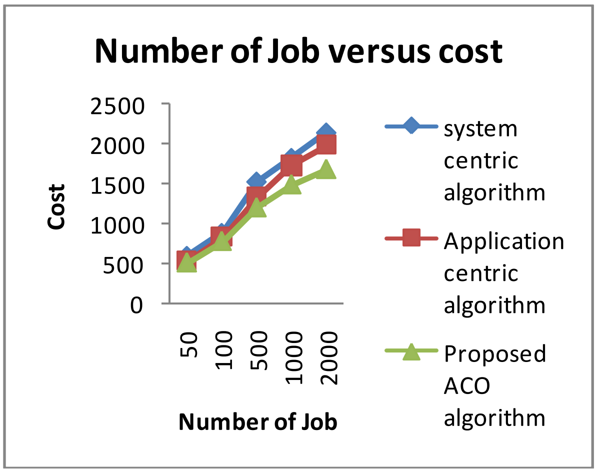


Figure II. Number of jobs versus cost



The Graph shows the proposed scheduling algorithm considerably improve the performance in reducing the time and cost with other two algorithms. The parameters α, ß and intensive weightage factors I1, I2 play vital role in getting the optimized results. Here we note that adaptive schedule is significant since as number of jobs increases it finds only those sites for job execution which have adequate network bandwidth and least loaded. The experimental results show that the algorithm performed to our expectations. It improves the efficiency and reliability especially even in unreliable resource and network conditions.

5. CONCLUSION

The adaptive resource scheduling has become a key technique for data intensive application to provide desired services with appropriate QOS. In this paper we propose a new scheduling algorithm that provides a solution which combines both application centric and system centric benefits. The results indicate that considerable optimization can be achieved using adaptive scheduling algorithm for data intensive applications. The formal description of adaptive scheduling algorithm is presented and shows that a scheduling approach significantly reduces the total execution time even in unreliable resource and network conditions. Intensive weightage factors, the parameters of heuristic value and pheromone intensity are the key elements, which are optimized in scheduling algorithm. More work could be carried out with algorithm described here, for example different QOS parameters. We also plan to explore with simulation the effect of changing the weightage factors and parameters used in the algorithm on application performance.

6. REFERENCES

[1] Foster, I. and Kesselman, C. (eds.): The Grid 2: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers, San Francisco (2003).

[2] Stockinger, H., Donno, F., Puccinelli, R. and Stockinger, K.: Data Grid Tutorials with Hands-on Experience. In: IEEE International Symp. on Cluster Computing and the Grid (2004),152-159.

[3] Park, S. M., Kim, J. H.: Chameleon: "A Resource Scheduler in A Data Grid Environment". In: Proceedings of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID'03).

[4] Li, K.: "Job Scheduling for Grid Computing on Metacomputers" In: Proceedings of the 19th IEEE International Parallel and Dist. Processing Symp (IPDPS'05).

[5] Buyya, R., Abramson, D. and Venugopal, S.: "The Grid Economy". In: proceedings of the IEEE, vol. 93, NO. 3, March 2005, 698-714.

[6] Dorigo M V Maniezzo A Colorni "The Ant System : Optimization by colony of cooperating agent" IEEE Transaction on Systems, and Cybernetics Vol 26 No 1, pp 29-41, 1996.

[7] Aggarwal, A. K., Kent, R. D.: "An Adaptive Generalized Scheduler for Grid Application". In: Proceedings of the 19th International Symposium on High Performance Computing Systems and Applications (HPCS'05).

[8] Fan, H., Hua, Zh., Li, J. and Yuan, D.: "Solving a Shortest Path Problem by Ant Algorithm". In: Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai, 26-29 August 2004, 3174-3177.

[9] Richard Mccltchey, Ashiq Anjum, Heinz Stockinger, Arshad Ali, Ian willers "Bulk Scheduling with the

DIANA Scheduler" IEEE Transaction on Nuclear Science Vol 53 No 6 Dec. 2006.

[10] Richard Mccltchey, Ashiq Anjum, Heinz Stockinger, Arshad Ali, Ian willers, Michael Thomas " Data intensive and Network aware (DIANA) Grid scheduling" J Grid comp.(2007) 5: 43-64.

[11] kavitha Ranganathan,K Foster I : "Decuopling computation and data scheduling in distributed intensive application" In : International Symposium on High performance distributed Computing(HPDC-11)Scotland,2002.

[12] Xu, L., Wang, B. and Ai, B.: "A Strategy for Data Replication in Data Grids. In: Current Trends in High Performance Computing and Its Applications". Springer- Verlag, Berlin Heidelberg (2005), 557-562.

[13] Jia Yu, Rajkumar Buyya and Chen Khong Tham, "QoS-based Scheduling of Workflow Applications on Service Grids", Proceedings of the 1st IEEE International Conference on e-Science and Grid Computing (e-Science 2005, IEEE CS Press, Los Alamitos, CA, USA), Dec. 5-8, 2005, Melbourne, Australia.

[14] Xu, Zh., Hou, X., Sun, J.: "Ant algorithm-based task scheduling in grid computing". CCECE 2003 – CCGEl 2003, Montreal, May/mai 2003, 1107-1110.

[15] Xiangang Zhao, Bai Wang et al "QOS – Based Algorithm for Job Allocation and Scheduling in Data Grid" GCCW'06 IEEE international conference 2006.

[16] Srikumar Venugopal and Rajkumar Buyya, "A Deadline and Budget Constrained Scheduling Algorithm for eScience Applications on Data Grids", 6th International Conference on Algorithms and Architectures for Parallel Processing, Melbourne, Australia.

[17] H.Yan, X.Shen,Xli and MWu,"An Improved ant algorithm for job scheduling in Grid computing" In proceeding of fourth international conference of Machine Learning and cyber netics,18-21 August 2005.

[18] Anthony Sulistio, Uros Cibej, Borut Robic and Rajkumar Buyya "A Tool for Modelling and Simulation of Data Grids with Integration of Data Storage, Replication and Analysis", Technical Report, GRIDS-TR-2005-13, Grid Computing and Distributed Systems Laboratory, University of Melbourne, Australia, Nov. 8, 2005.