2012 International Conference on Industrial and Intelligent Information (ICIII 2012)
IPCSIT vol.31 (2012) © (2012) IACSIT Press, Singapore

Improved MACO approach for grid scheduling

Parisa Rahmani [1], Mehdi Dadbakhsh [2] and Soulmaz Gheisari[1+]

[1] Islamic Azad University Pardis Branch, Iran

[2] Applied Science and Technology University, Unit 25, Iran

Abstract. The grid computing system is a new trend in distributing computing systems. Grid scheduling is required in this system to achieve high performance. The goal of grid task scheduling is to achieve high system throughput and assigning the task to computing nodes. In general, there is an NP hard problem if we want to find an optimal scheduling by using the traditional sequential method. A near optimal solution can also be found by using heuristic approaches. Heuristic approaches are simpler than existing methods. The ant colony algorithm, which is one of the heuristic algorithms, suits well for the grid scheduling. In this paper we introduce a new task scheduling algorithm with load balancing based on multiple ant colony optimization (MACO). In the improved MACO approach, all colonies by using the repulsion mechanism construct their solution in parallel and find the optimal solution with a minimum execution time of task. According to experimental results, the proposed algorithm out performs the algorithms which are based on ACO.

Keywords: grid computing, task scheduling, multiple ant colony optimization, pheromone, load balancing

1. Introduction

Grid is defined as a type of parallel and distributed system that enables the sharing, selection and aggregation of geographically distributed autonomous resource dynamically at runtime depending on their availability, performance and users quality of service requirements [1]. In fact grid computing enables large scale resource sharing and provides a promising platform for execution task efficiently. Grid computing [2] is a new paradigm for solving the complex problems. In grid, we need to consider the condition such as network status and resources status. If network or resources are unstable, tasks would be failed or the total computing time would be very large. So we need on efficient task scheduling algorithm for these problems in the grid environment. Because the environment status may change frequently , traditional task scheduling algorithm such as " First Come First Server " (FCFS), "Shortest Job First " (SJF),etc . ,may not be suitable for the dynamic environment in grids. In grid, users may face hundreds of thousands of computers to utilize. Therefore grid task scheduling is a very important issue in grid computing. A good scheduling would adjust its scheduling strategy according to the changing status of the entire environment and the type of jobs. Therefore, a dynamic algorithm in task scheduling such as Ant Colony Optimization (ACO) [3], [4] is appropriate for grids. ACO is a heuristic algorithm with efficient local search for combinatorial problems. ACO imitates the behavior of real ant colonies in nature to search for food and to connect to each other by pheromone laid on paths traveled. The ACO approaches using a single colony system may suffer from stagnation problem because of its tendency to use the positive feedback mechanism of pheromone, multiple Ant Colonies Optimization (MACO) is employed to avoid this by using several ant colonies to solve combinatorial optimization problems cooperatively [5], [6] . MACO approaches have been explored for several optimization problems [6-11]. However, few researches have focused on solving task scheduling with load balancing in grid computing using MACO approaches.

In this paper, we proposed a multiple ant colony algorithm for task scheduling problem in grid computing and we add a new mechanism to the MACO approach to distributing the workload on computing nodes efficiently. This paper is organized as fallows. In section 2, the related work about many kinds of the new ACO algorithms to solve task scheduling problems are introduced. In section 3, our New MACO method is proposed. In section 4, the comparison of proposed algorithm with other existing methods is given. Finally, some concluding remarks are made in section 5.

2. Related work

2.1. Task scheduling algorithms in grids

The resource scheduling in grid is a NP complete problem. Various algorithms have been designed to schedule the tasks in computational grids. The most commonly used algorithm are OLB, FPLTF, WQR, MET, MCT, Min-Min, Max-Min, etc [12-14], that we will introduce some of them. The FPLTF (Fastest Processor to Largest Task First) [15] algorithm schedules tasks to resources according to the workload of task in grid system. If there are many tasks with heavy workload, its performance may be very bad. Dynamic FPLTF (DFPLTF) [16] is based on the static FPLTF, it gives the highest priority to the largest task. DFPLTF needs prediction information on processor speeds and task workload. The WQR (Work queue with Replication) is based on the work queue algorithm [16]. The WQR applies FCFS and random transfer to assign resources and it sets a faster processor with more task than a slower processor.

The next algorithm is Min-Min, the Min-Min [17] set the tasks which can be completed earliest with the highest priority. The main idea of Min-Min is that it assigns tasks to resource which can execute tasks the fastest. Max-min [17] set the tasks which has the maximum earliest completion time with highest priority . The main idea of Max-Min is that it overlaps the task with long running time with the tasks with short running time. The RR(round-robin) algorithm focuses on the fairness problem. RR uses the ring as it queue to store jobs .The advantage of R.R algorithm is that each job will be executed in turn and don't have to wait for the pervious one to complete . There are the traditional job scheduling algorithm such as FCFS, SJF, etc. may not be suitable for the dynamic environment in grids .In this section We have reviewed a number of Grid scheduling algorithms that weren't heuristic.

2.2. task scheduling algorithms in grid with ACO approach

The term heuristic is used for algorithms which find solutions among all possible ones. These algorithms, usually find a solution close to the best one and they find it fast and easily. The ant colony optimization algorithm (ACO), is a heuristic algorithm for solving computational problems which can be reduced to finding good paths through graphs. Ant colony optimization algorithm [dorigo 1996] are multi agent system, which consist of agents with the collective behavior (stigmergy) of ants for finding shortest paths . Ant colony algorithm was inspired by the observation of real ant colonies. One application of the ACO algorithm is the task scheduling in grid system. The original ant algorithm is shown below that its the basis for other methods based on ACO approach.

```
1. Procedure ACO
2. Begin
3. Initialize the pheromone
4. While    stopping criterion not satisfied     do
5. Position each ant in a stating node
6. Repeat
7. For   each ant     Do
8.choose next node by applying the state transition rate
9. end   for
10. until    every ant has build a solution
11. update the pheromone
12.end    while
13.end
```

Fig. 1 : Pseudo code for original ant algorithm

In the following we introduce several new methods for task scheduling in grid computing based on ACO approach. For the shortcoming of ant colony algorithm, Zhu and Wei proposed an improved ant colony algorithm (IACO). First initialize the pheromone with Min-min algorithm to makeup the lack of initial pheromone, and then improved the parameters of the ant colony algorithm to avoid falling into the part of local optimum. Results show IACO algorithm has a smaller Makespan (Makespan, namely, the completion time that all of tasks are assigned to the corresponding node by scheduling program) , resource load balancing [18]. Mathiyalagen in [19] proposed the modified ant colony algorithm that is used to solve large complex problems . It required grid scheduling to achieve high performance .The basic ant algorithm involved transition probability and pheromone updating rule. The modified ant colony algorithm , use to achieve better scheduling to improve rule of original ant colony algorithm . The new job scheduling algorithm that is based on the basic ACO algorithm proposed by smith jha [20] for dependent task scheduling . In this algorithm a Directed acyclic graph (DAG) given for a set of dependent task. The DAG is divided into n number of groups containing independent set of tasks. The procedure scheduling ant colony (Group no) apply for each group and thus all tasks are scheduled. The next ant colony algorithm for job scheduling in the grid environment combines the techniques from ant colony system and Max-Min ant system. The algorithm focused an local pheromone trial update and trail limit value . A matrix is used to record the status of the available resources .the new algorithm has been implement in the grid system architecture which consist of four main components namely the grid information server , grid resource broker , jobs and resources . The algorithm works as follows. User will send request to process a job. Grid resource broker starts to calculate the relevant parameter to schedule the job after receiving the message from the user. The information server also provides the resource information to grid resource broker. The largest entry in the pheromone value (PV) matrix will be selected by proposed technique. Then local pheromone update is performed after a job is assigned to a resource. A global pheromone update is performed after a resource completed processing a job . This algorithm provides a significant decrease in the Makespan [21].

The last algorithm is based the MACO approach [22] . In the MACO approach, multiple ant colony work together and exchange information to collectively find solutions with a two –fold objective of minimizing the execution time of task and degree of imbalance of computing nodes. In this algorithm all colonies construct their solution in parallel, and interaction mechanisms are designed for sharing experiences among colonies. Pheromone is used as the interaction mechanism not only between the ants of the same colony but also among ant colonies. The pheromone evaluation mechanism averages the pheromone values off all colonies, which represents information of all colonies. Based on the average of the available experiences of ants for all colonies, an ant will decide how to choose an edge. This method greatly reduces the Makespan that it is desirable. In this paper we present a new algorithm based on MACO approach. The new method by adding new mechanisms has a better performance than the previous methods.

3. The framework of new algorithm

In this proposed algorithm we use the MACO approach for task scheduling in grid computing. We add a new mechanism to the MACO approach to distribute the workload on computing nodes efficiently. In MACO, more than one colony of ants are used to search for optimal solutions, and each colony of ants deposits a different type of pheromone. Although ants in each colony respond to pheromone from its own colony, MACO is augmented with a repulsion mechanism [23] that prevents ants from different colonies to choose the same optimal solution. According to traffic variations, using the MACO approach and repulsion mechanism will lead us to different solutions and finally the best solution is chosen. While in ACO approach, there is only one solution which isn't certainly optimal. In our proposed algorithm the Makespan is minimized by using the load balancing that is established in our algorithm.

Assumption:

- The scheduling algorithm is a non preemptive.
- Tasks are independent.

- The communication overloads are negligible.
- Each colony has its own pheromone.

3.1. Description of algorithm

At the first stage, ants don't have any knowledge of grid environment. They move through computing nodes randomly and assign the task to the nodes. The ants choose the next node based on the repulsion mechanism which will be considered in the next section. Local pheromone updating is implemented after each ant in each colony assigns all tasks and completes its tour. Finally, when all ants of all colony construct their solutions, global pheromone updating is done by best ant. The best ant is the ant that finds the best solution.

3.1.1. Repulsion mechanism

As we have mentioned in this method we use multiple ant colony. Each colony has its own pheromone. By adopting the MACO approach, it may be possible to reduce the likelihood that all colonies establish connections using only the optimal solution. If colony M_1 selects the optimal solution A , the idea of repulsion may increase the probability that colony M_2 will select the B solution(?).In addition, due to repulsion, an ant is less likely to prefer computing nodes with (higher concentration of) pheromone from other colonies. Moreover, it is reminded that the degrees of attraction and repulsion are determined by two weighting parameters as mentioned in the next section[23].

Assume that M ant colonies would be used for the scheduling problem and each colony contains N ants C_j is j^{th} computing node. We denote by ant(m , n) the n^{th} ant in m^{th} colony. Ant (m , n) moves through computing nodes and assigns task Ti to node Cj . In the repulsion mechanism the ant (m ,n) chooses one of the computing nodes as follows :

- The weight of attraction of computing node j for ant (m ,n) is calculate by (1)

$$a_{ij}^{mn} = \frac{\left[\tau_j^m\right]^\alpha \left[\eta_{ij}^{mn}\right]^\beta}{\sum_{j\in[1,q],k\neq j}\left[\tau_k^m\right]^\alpha \left[\eta_{ik}^{mn}\right]^\beta} \quad (1)$$

Where a_{ij}^{mn} is the attraction weight of the node C_j for ant (m,n) . τ_j^m Is intensity of the pheromone node C_j in the m^{th} colony. The α and β are the weights of the influence of pheromone and heuristic information, respectively, and η_{ij}^{mn} is the heuristic value for ant (m,n) to assign task T_i to node C_j . This parameter considers the load of computing nodes as the heuristic value and is given by (2)

$$\eta_{ij}^{mn} = \frac{l_{avg}}{l_j^{mn}} \quad (2)$$

Where l_{avg} is the average load among all the computing nodes and $l_j^{m,n}$ is the load of node C_j for a schedule of ant (m ,n). l_{avg} and $l_j^{m,n}$ are defined as (3)(4)

$$l_j^{mn} = \sum_{i\in[1,p]} the\ execution\ \ time\ of\ task\ T_i\ \ on\ node\ c_j\ for\ ant\ (m,n) \quad (3)$$

$$l_{avg} = \frac{l_{max}-l_{min}}{2} \quad (4)$$

- The weight of repulsion computing node is calculated by (5). The weight of repulsion of the node C_j is depended to the pheromone value of the other colonies.

$$b_{ij}^{mn} = \frac{\left[\tau_j^h\right]^\alpha \left[\eta_{ij}^{mn}\right]^\beta}{\sum_{k\in[1,q],k\neq j}\left[\tau_k^h\right]^\alpha \left[\eta_{ik}^{mn}\right]^\beta} \quad (5)$$

Where τ_j^h is the sum of the pheromone values of all colonies except the m[th] colony on node C_j . In fact this value display the repulsion weight of node C_j that is defined by (6)

$$\tau_j^h = \sum_{z\in[1,M],z\neq m} \tau_j^z \quad (6)$$

- Based on the attraction and repulsion weights of the node C_j which were calculated , the parameter γ is defined by (7)

$$\gamma_{ij}^{mn} = \frac{(a_{ij}^{mn}/b_{ij}^{mn})}{\sum_{k\in[1,q]}(a_{ik}^{mn}/b_{ik}^{mn})} \quad (7)$$

Where K is the number of computing nodes, That We want to select one of these computing nodes to assign task T_i to it. γ increases as the attraction Weight increases and γ decreases as the repulsion weight increases .

Based on (7) the ant (m ,n) selects the computing node which has the maximum value of the γ . It means that the next node for assigning the task is selected based on the γ value. By using this mechanism, ants from different colonies construct different solution. These solutions are constructing based on the load balancing factor. The selection solution among different solutions is the best solution with more likelihood.

3.1.2. The frame work of the new MACO approach

In this approach we have multiple colonies which construct their solutions in parallel. In general, the approach can be briefly sketched as follows.

Step 1: At first , ants are distributed on computing nodes and the initial pheromone value is the same in all nodes .

Step2: for each ant in each colony do step 3 and 4

Step3: In this step, ant (m ,n) in m[th] colony selects the next computing node base on the repulsion mechanism and assign s task T_i to it .

Step4: local pheromone updating is implemented after each ant completes its tour and is applied on the visited nodes. We can denote that an ant construct the solution when all tasks are assigned . The local pheromone update is given by:

$$\tau_j^m = (1-\rho)\tau_j^m + \rho \quad (8)$$

Where ρ is the evaporation rate (0<= ρ<=1) andτ_j^m Is the pheromone value of m[th] colony on the node C_j , $\Delta\tau_j^{mn} = 1/T^{mn}$,T^{mn} is the latest completion time among all tasks which are assigned by the ant(m ,n) in other wordT^{mn} is Markspan for the schedule of ant(m ,n) .

Step5: global pheromone updating is done by the best ant of each colony after all ants of all colonies complete their tours and construct their solutions. The ant finding solution with the minimum Makespan will deposit an amount of pheromone on edge of its path according to the global updating

$$\tau_j^m = (1-\rho)\tau_j^m + \rho\Delta\tau_j^{mn} \quad (9)$$

Where $\Delta\tau_j^{mn} = 1/T_{op}$, T_{op} is the optimal Makespan that is achieved by the best ant . The best ant is the ant that construct the best solution.

Step6: The approach terminate when the steps repeat until the best solution is not improved in successive number of iteration.

4. EXPRIMENTAL RESULT

In this section, the proposed algorithm based on MACO is compared with the other grid scheduling algorithms. It is shown that performance of the algorithm is improved by applying new mechanisms. The interconnection structure is an irregular mesh that is interesting because it makes traffic management a complex and difficult task. The computing nodes are initialized randomly. We assume that there are ten computing nodes, ten colonies and ten ants in each colony. Tasks are created based on Poisson distribution and are assigned to resources. The interval time of tasks is generated based on exponential distribution. For simplicity we assume that the evaporation rate (ρ), and α and β are equal to one.

Table.1: Parameter Setting

Parameters	Number of colony	Number of ant in each colony	Number of nodes	Number of task	ρ	α	β
Values	10	1-10	10	10-500	0.1	1	1

Table.2: Comparison of scheduling algorithms in Makespan factor

Number of task	Makespan(sec)						
	Min-Min	FCFS	BACO	IACO	Modify ACO	MACO	NEW MACO
20	11.2	12.01	13.6	21.2	41.1	12.1	13.3
40	25	22.7	30.08	29	65.2	27.34	27.5
60	39.3	41.92	45.78	45	50	40.54	41.2
80	57.4	58.28	67.34	65.3	55.4	60.28	58.2
100	64.5	63.2	68.3	65.9	63.5	61.1	49.9
200	150	143.5	150.4	152.1	143.2	136.7	112.22
300	221.6	219.6	228.9	220.4	229	202.7	160.14
400	332.7	321.4	336.7	337.3	309.1	301.4	240.2
500	390.4	374.3	379.6	370.1	341.7	358.5	290.5

In table1, the new algorithm based on MACO and the other algorithms based on ACO such as, BACO, IACO, Modify ACO and MACO are compared. The new algorithm improved the performance of the task scheduling algorithm by using the repulsion mechanism and providing the load balancing. From the experimental results that are given in table 1, the performance of the proposed algorithm is better than the other algorithms based on ACO. The new algorithm also is compared with the other algorithms which aren't heuristic such as FSFC and Min-Min.

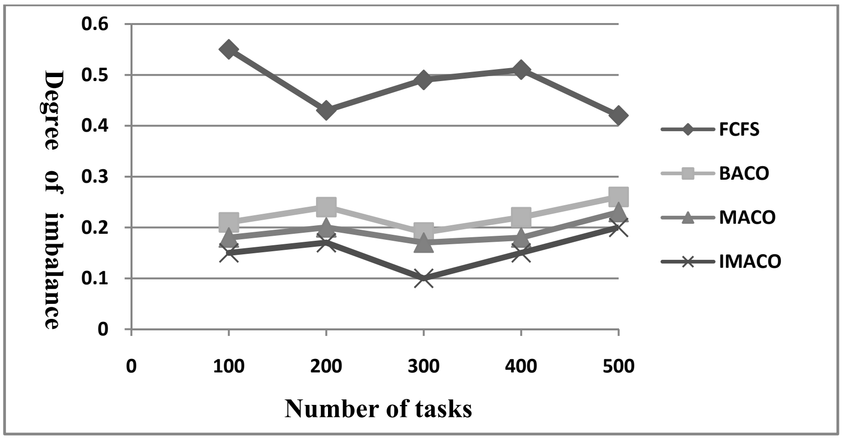

Fig. 2 : Comparison LOAD BALANCING factor of IMACO algorithm with MACO algorithm

In figure 2, the load balancing factor of the new algorithm and the other algorithms are compared. To measure the load balancing, we define a new factor which is called the degree of imbalance (DI) and is given by:

$$DI = \frac{L_{max} - L_{min}}{L_{avg}} \tag{10}$$

By minimizing the degree of imbalance, the workload is distributed effectively an each computing node. As it is shown the figure 1, the load distribution of the new algorithm is better than the MACO, BACO and FCFS algorithms.

5. Conclusion

In this paper we have proposed a new method based on MACO approach for achieving task scheduling with load balancing .Most of the methods that are based on the ACO approach are involved with the stagnation problem but in the new algorithm by using the multiple ant colony and repulsion mechanism the stagnation problem can be avoided. This algorithm can adopt the system environment freely at runtime and by applying the dynamic solutions which are optimal can distribute the load in the grid system. Experimental result shows that the new algorithm increases scheduling efficiency and improves overall performance of the grid task scheduling. Also it is capable of achieving solutions effectively and efficiency.

6. References

[1] P.Mathiyalagan ,S.Suriya , Dr.S.N.Sivanandam .”*Modified Ant Colony Algorithm for Grid Scheduling* “,in: International Journal on computer Science and Engineering ,Vol.02,no.2010,(IJCSE)

[2] Fangpeng Dong and Selim G. Akl “*Schedul Algorithms for Grid Computing: State of the Art and Open Problems*”, Technical Report No. 2006-504

[3] M. Dorigo, C. “*Blum, Ant colony optimization theory*”: A survey, Theoretical Computer Science 344(2-3) (2005) 243-278 .

[4] M. Dorigo, “ *Ant colony optimization*”, http://www.aco-metaheuristic.org.

[5] Jong J, Wiering M.” *Multiple ant colony system for the bus-stop allocation problem”* [C]. in: Proceedings of the Thirteenth Belgium-Netherlands Conference on Artificial Intelligence (BNAIC'01).2001. 141-148.

[6] Kawamura H, Yamamoto M, Suzuki K, et al. “ *Multiple ant colonies algorithm based on colony level interactions*” [J]. IEICE Trans. Fundamentals, 2000,E83-A(2)

[7] Chen C-H, Ting C-J.” *Applying Multiple Ant Colony System to Solve Single Source Capacitated Facility ocation Problem*” [C]. in: ANTS 2006. 508-509.

[8] Chan F T S, Kumar N. Effective allocation of customers to distribution centers:” *A multiple ant colony optimization approach*”[J].Robotic and Computer-Integrated Manufacturing, 2009,25: 1-12.

[9] Ellabib I, Calamai P, Basir O. “*Exchange strategies for multiple Ant Colony System”*[J]. Information Sciences, 2007,177: 1248-1264.

[10] MIDDENDORF M. “ *Multi Colony Ant Algorithms”*[J]. Journal of Heuristics, 2002,8: 305-320

[11] Sim K M, Sun W H. “*Multiple Ant Colony Optimization for Load Balancin”*. in: IDEAL 2003,.2003. 467-471

[12] M. Maheswaran, S. Ali, H.J. Siegel, D. Hensgen, R. Freund, “ *Dynamic matching and scheduling of a class of independent tasks onto heterogeneous computing system*”, Journal of Parallel and Distributed Computing 59 (1999) 107-131

[13] Braun, T. D., Siegel, H. J., Beck, N., B¨ol¨oni, L. L., Maheswaran, M.,Reuther, A. I., Robertson, J. P., Theyr, M. D.,Yao, B., Hensgen, D.,and Freund, R. F. (2001). *A comparison of eleven static heuristics for mapping a class of*

independent tasks onto heterogeneous distributed computing systems. Journal of Parallel and Distributed Computing,61(6):810-837

[14] G. Ritchie and J. Levine, " *A hybrid ant algorithm for scheduling independent jobs in heterogeneous computing environments*". University of Strathclyde.

[15] D. Saha, D. Menasce, S. Porto, "*Static and dynamic processor scheduling disciplines in heterogeneous parallel architectures*", Journal of Parallel and Distributed Computing 28 (1) (1995) 1-18

[16] D. Paranhos, W. Cirne, F. Brasileiro," *Trading cycles for information: Using replicationto schedule bag-of-tasks applications on computational grids*", in: International Conference on Parallel and Distributed Computing (Euro-Par), in: Lecture Notes in Computer Science, vol. 2790, 2003, pp. 169-180.

[17] M. Maheswaran, S. Ali, H.J. Siegel, D. Hensgen, R. Freund, " *Dynamic matching and scheduling of a class of independent tasks onto heterogeneous computing system*", Journal of Parallel and Distributed Computing 59 (1999) 107-131

[18] Youchan Zhu and Qiujuan ." *An Improved ANT Colony Algorithm for Independent Tasks Scheduling of Grid* " ,Conference IEEE 2010

[19] P.Mathiyalagan ,S.Suriya , Dr.S.N.Sivanandam ."*Modified Ant Colony Algorithm for Grid Scheduling* ",in: International Journal on computer Science and Engineering ,Vol.02,no.2010,(IJCSE)

[20] Mrs.Smitha Jha ," *Balanced Ant Colony Algorithm for Scheduling DAG to Grid Heterogeneous System* " International Journal of Scientific &Engineering Research Volume 2,Issue 6,June-2011

[21] Ku Ruhana Ku-Mahamud,H.Jamal Abdul Nasir ,"*Ant Colony Algorithm for Job Scheduling in Grid computing* " , Fourth Asia Conference on Mathematic /Analytical Modelling and Computer Simulation,2010

[22] Liang Bai ,Yan-Li ."*Task Scheduling with Load Balancing using Multiple Ant Colonies Optimization in Grid Computing* ",Sixth International conference on Natural Computation (ICNC 2010)

[23] N. Varela and M. C. Sinclair, "*Ant colony optimization for virtual-wave- length-path routing and wavelength allocation*," in Proc. Congress Evo- lutionary Computation, Washington, DC, July 1999, pp. 1809-1816.

.