RESEARCH ARTICLE OPEN ACCESS

# Optimized Implementation of Large Data on Grid Environment using Multiple Ant Colony Algorithm

Ashraf Siddiqui [1], Kiran Bhandari [2], Veena Kulkarni [3]
ME Student [1], Department of Computer Science and Engineering [2] & [3]
Mumbai University/ Thakur college of engineering & Technology
Mumbai - India

## ABSTRACT

Grid computing is now widely used in various applications that are beyond distribution and sharing of resources. The Grid is the collection of a large volume of data and varieties of data such as structured, semi-structured, and unstructured or collection of all at different locations. The two major problems in a grid environment are data storage and processing**.** One major issue associated with the efficient utilization of heterogeneous resources is task scheduling. For effective processing of data and resources, various scheduling algorithms and techniques are used. In this paper, we discuss the optimized multiple ant colony algorithm for effective scheduling of resources to optimize the large data**.**

***Keywords :***— ACO, MACO, Gridlet, resource, makespan, iteration, I-MACO, GridSim.

## I. INTRODUCTION

Grid computing is the advance of distributed computing. The aim of grid computing is how to develop strong power processing and storage resources by many small and weak resources integration. Gird computing is a collection of internally connected multiple resources which provide excellent and powerful capabilities. Grid computing is a combination of many resources from different resources and specifications not like regular Distributed Processing which based on similar resources [1]. The user of the grid can use any of these interconnected resources in the environment to solve his problems which cannot be solved by locally owned resources capabilities. Jobs scheduling problem is the problem that is raised when a resource in the grid submits tasks and the manager of the grid would like to find a suitable resource to host these tasks. For job scheduling in grid, various well-known algorithms are proposed such as First-Come-First-Served (FCFS) Earliest-Deadline-First (EDF) and Backfilling. But these traditional scheduling gives poor grid schedulers and that is why these traditional scheduling algorithm cannot be applied in the grid computing [2]. Therefore much research has also been devoted in job scheduling in a grid environment.

## II. ANT COLONY ALGORITHM (ACO)

Ant colony algorithm is invented by the real ants finding the shortest path between the source and food. By the use of artificial ant colony algorithm, we can solve the grid scheduling problem similarly as the real ants colony finding the shortest path. Natural ant colonies are finding the shortest path with the use of a chemical substance called as pheromones. The ant colony work in following steps to find the shortest path [4].

1. Initialization of ants.
2. Finding the next move.
3. Updating the pheromone’s values
4. Sharing the pheromone’s values among all ants
5. Select the optimal path with the maximum pheromone’s values.
6. Finally, construct a solution as the shortest path

## III. MULTIPLE ANT COLONY ALGORITHM (MACO)

As ant colony algorithm is used for scheduling the resources in a grid environment. The grid is nothing but varieties of data in distributed format at different location. So using a single ant colony algorithm take more time for computation of grid data which will be not so effective. So the concept of multiple ant colony (MACO) is proposed and which perform better result as compared to the ant colony algorithm. Now MACO algorithm is widely used for grid scheduling problem. In MACO algorithm M number of ant colonies working together to find out the better results of scheduling of the resources in grid environment by finding the shortest path with the updated values of pheromones [3]. The multiple ant colony work in the following steps.

1. Initialize M number of ant colonies
2. N is the number of ants in each ant colony.
3. Calculating the pheromone values in two different ways known as local and global pheromones.
4. Sharing of pheromones values by each ant in the same colony that is called local pheromone sharing and sharing of pheromones value between the ant colonies known as global pheromones
5. Finding the shortest path with the local and global pheromone values.
6. Construct the shortest path that will be optimal path.
7. Construct the final solution by using the shortest path.

There are four main steps
Step 1: Initialization of algorithm
Step 2: Interaction of multiple ant colony system
Step 3: Pheromone updating
Step 4: Solution Construction

**Initialization of Algorithm**

Assume that ten number of ant colonies are used to search the optimal solution of the scheduling problem and N is the number of ants in each colonies. This can be denoted by ant (M, N) i.e. the $N^{th}$ ant in $M^{th}$ colony. In the beginning, ants are randomly distributed on a computing node and the value of pheromone would be (τjm) of the $M^{th}$ colony at node $C_j$ is initialized as low or a small value.

**Interaction of multiple Ant colony system**

The pheromone have used as the communication and interaction mechanism between the ants of the same colony and with the other colonies also. The colony level communication or interaction is achieved by calculating the pheromones of all different colonies.

The calculated pheromone 'τj' on the node '$C_j$' in terms is evaluated as [3] below equation.

$$\mathbf{T_j = {}_{[1,M]}T_j^m/M} \qquad (1)$$

Where 'M' is the no. Of ant colonies and '$T_j$' is the calculated pheromones.

The resultant pheromone value is the average sum of all colonies pheromone values which are the experience of all colonies.

The resultant pheromone value is the average of experience of all colonies pheromone values and basis of this value next node is decided.

**Pheromone Updating**

In ACO the traditional pheromone updating method is used but in the MACO (M, N) pheromone updating in two different ways such as local pheromone updating [3]

**Local pheromone updating** The Pheromone value of each single ant colonies are treated as local pheromone values and are expressed by the equation (2) below.

$$\mathbf{\tau jm = (1-\rho)\ \tau jm + \Delta\ \rho\ \tau jmn} \qquad (2)$$

Evaporation Co-Efficient

**Global Pheromone Updating** When all the ant colonies calculate their pheromone values and find the best pheromone values among the best pheromone values among all the colonies is called as global pheromone updating. Which is also represented by equation (3).

$$\mathbf{\tau jm = (1-\lambda)\ \tau jm + \lambda\ \Delta\ \tau jmn} \qquad (3)$$

λ- Evaporation Co-Efficient

Thus the ant finding the minimum makespan by maximum intensity of the pheromone and its path.

**Solution Construction**

Solution Construction is the final phase of the process in this ant (M,N) moving through the computing nodes allocate the task τi to the node $C_j$ probabilistically in terms of the pheromone value and some heuristic information and complete all allocation [3].The probability $P^{mn}$ is for ant(M,N) to assign the defined node as

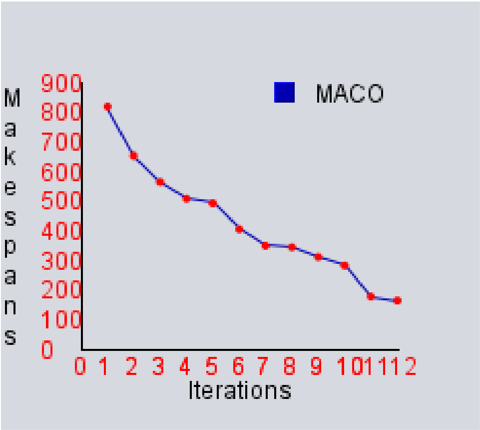
$$\mathbf{P\ mn\ ij = \tau j\ Ŋ\ mnij / \sum \tau j\ Ŋ\ mnij} \qquad (4)$$

The above equation (4) is the final calculation of all ant colonies. From the above equation, final optimal path will get and simulation results with the use of GridSim simulation tool. Which is in the form of tables and that table is shown makespan time with respect to the number of iterations when the gridlet and resources are configured.

Table 1

| Iteratio... | Gridlet | Resou... | Makes... |
|---|---|---|---|
| 1 | 1 | 1 | 818 |
| 2 | 1 | 2 | 652 |
| 3 | 1 | 3 | 563 |
| 4 | 1 | 4 | 506 |
| 5 | 2 | 1 | 491 |
| 6 | 2 | 2 | 404 |
| 7 | 2 | 3 | 348 |
| 8 | 2 | 4 | 343 |
| 9 | 3 | 1 | 309 |
| 10 | 3 | 2 | 280 |
| 11 | 3 | 3 | 176 |
| 12 | 3 | 4 | 163 |

Table of Makespan time




Makespans VS Iteration graph

## IV. PROPOSE I-MACO

The main Goal is to find the best or optimal way to process the resources with the minimum time to get the effective results in the grid environment. The scheduling is the best way to optimize the execution time and providing the best way to schedule the resource with the relevant data [6]. In this proposed idea we are trying to modify the MACO algorithm in terms of pheromone value update to get the result.

In the proposed idea we are considering the three operational steps which are performed by each of ant colonies as that is the basic function of MACO [3].

1) Initialization
2) Local and global pheromone updating.
3) Construction of final result.

In the proposed idea we are trying to modify the step 2 that is the local pheromone updating method as equation (2) & global pheromone updating method as equation (3) by this modification we can improve the MACO algorithm performance in a grid environment. As in the MACO there are two pheromones updating one by single ant colony that is called as local pheromone value [3].

The value of the pheromone evaporation coefficient $\rho$ in equation (2) and $\lambda$ in equation (3) must be in between 0 and 1. In proposed idea, we are decreasing values of pheromone evaporation coefficient so we can speed up the process of searching for the shortest path by each ant colony. That is why we are replacing the value of $\rho$ with $\delta$ & $\lambda$ with $\psi$ i.e. local & global pheromone evaporation coefficient $\delta$ by $(\delta/ \delta+1)$ & $\psi$ $(\psi / \psi +1)$ in both equation (2) and equation (3).

The proposed new equation will become as, for local pheromone updating method is equation (5).

$$\mathbf{Tj^{m}=\{ 1-(\delta/\delta+1)\ \tau j^{m} \}+\{\delta/ (\delta+1)\}\Delta\ \tau j^{mn}} \qquad (5)$$

**(δ / δ +1)** is the evaporation coefficient.

Similarly, the global pheromone updating equation will become as equation (6)

$$\mathbf{Tj^{m}=\{1-(\psi / \psi +1)\ \tau j^{m\}} \}+\{ \psi / (\psi +1)\}\Delta\ \tau j^{mn}} \qquad (6)$$

**(ψ / ψ+1)** is the pheromone evaporation coefficient.

Due to the changes of $\delta$ to $(\delta / \delta +1)$ & $\psi$ to $(\psi / \psi +1)$ the evaporation rate will be increased and the ants colonies process of finding the shortest path. As speed, the pheromone updating method will faster than the constructs of solution by all ant colonies will also be faster.

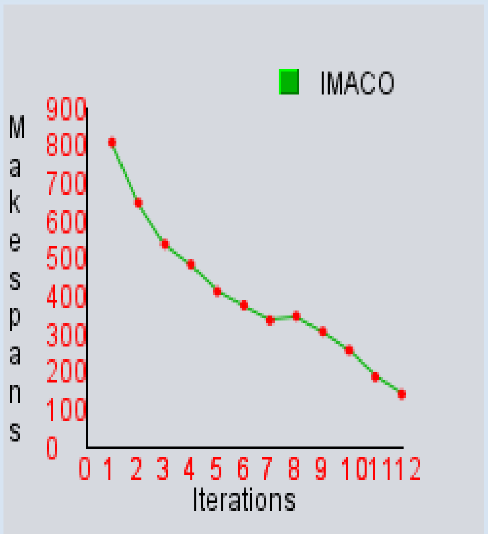
## V. EXPERIMENTAL RESULT OF I-MACO

GridSim is the simulation tool which provides the grid computing environment to run the MACO and I-MACO.by the use of this tool we get these simulated experimental result are shown in the tables[7].

Table 2

| Iteratio... | Gridlet | Resou... | Makes... |
|---|---|---|---|
| 1 | 1 | 1 | 807 |
| 2 | 1 | 2 | 709 |
| 3 | 1 | 3 | 537 |
| 4 | 1 | 4 | 483 |
| 5 | 2 | 1 | 412 |
| 6 | 2 | 2 | 375 |
| 7 | 2 | 3 | 335 |
| 8 | 2 | 4 | 346 |
| 9 | 3 | 1 | 305 |
| 10 | 3 | 2 | 256 |
| 11 | 3 | 3 | 185 |
| 12 | 3 | 4 | 139 |

Table of makespan time

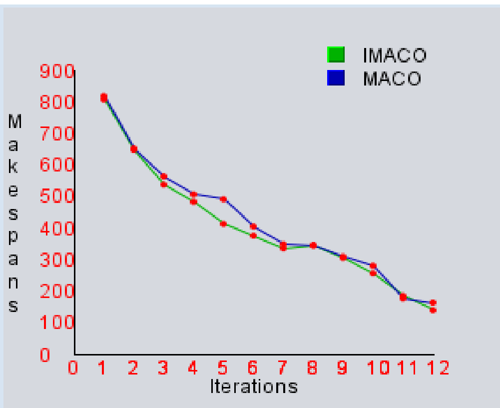



Makespans VS Iteration graph

Table 3

| Iterat... | Gridlet | Res... | MAC... | IMA... |
|---|---|---|---|---|
| 1 | 1 | 1 | 818 | 807 |
| 2 | 1 | 2 | 652 | 647 |
| 3 | 1 | 3 | 563 | 537 |
| 4 | 1 | 4 | 506 | 483 |
| 5 | 2 | 1 | 491 | 412 |
| 6 | 2 | 2 | 404 | 375 |
| 7 | 2 | 3 | 348 | 335 |
| 8 | 2 | 4 | 343 | 346 |
| 9 | 3 | 1 | 309 | 305 |
| 10 | 3 | 2 | 280 | 256 |
| 11 | 3 | 3 | 176 | 185 |
| 12 | 3 | 4 | 163 | 139 |

Comparative result of MACO and I-MACO




Makespans VS Iteration graph

## VI. CONCLUSION

Grids are the very large data storage of different types and scheduling of the Gridlets to the resources in the way so that task execution time should be very less is a big challenge. For effective scheduling, various algorithms are used and MACO is one of them. In the MACO, local and global pheromone evaporation are slower. Because of this, the moment of all ants of the same colony and the colonies are slower and ants required more time to compute the shortest path. In the proposed IMACO the speed evaporation of the pheromones are increase so the moment of ants and ant colonies are faster and will take less time to get the shortest path and construction solution that is the purpose.

## VII. ACKNOWLEDGMENT

The authors are grateful to the professor. Kiran Bhandari and professor.Veena Kulkarni for his valuable guidance regarding the contents which are included in the paper.

## REFERENCES

[1] Homer Wu, Chong-Yen Lee , Wuu-Yee Chen, Tsang-Yean Lee Proceedings of the First International Conference on Complex, Intelligent and Software Intensive Systems (CISIS'07) 0-7695-2823-6/07

[2] Dipti Sharma , Mr. Pradeep Mittal Volume 3, Issue 5, May 2013 ISSN: 2277 128X

[3] Fangpeng Dong and Selim G. 2006 Scheduling Algorithms for Grid Computing State of the Art and Open Problems. School of Computing, Queen's University Kingston, Ontario.

[4] S. Umarani et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (2), 2012,3388-3393 SNS College of Technology, Coimbatore Bannari Amman Institute of Technology, Erode.

[5] Vijay Subramani, Rajkumar Kettimuthu, Srividya Srinivasan, P. SadayappanR 1994. Distributed Job

Scheduling on Computational Grids using Multiple Simultaneous Requests, Department of Computer and Information Science. The Ohio State University

[6] Amr Rekaby1 and Mohamed Abo Rizka 2013 A COMPARATIVE STUDY IN DYNAMIC JOB SCHEDULING APPROACHES IN GRID COMPUTING, Egyptian Research and Scientific Innovation Lab (ERSIL), Egypt Arab Academy for Science, Technology and Maritime Transport College of Computing & Information Technology, Cairo, Egypt

[7] Dr. Rahul Malhotra*Prince Jain Volume 3, Issue 9, September 2013 ISSN: 2277 128X

[8] Jun Zhang and Chris Phillips, Job-Scheduling with Resource Availability Prediction for Volunteer-Based Grid Computing, Queen Mary, University of London.

[6] A. K. Aggarwal and R. D. Kent, 2005 An Adaptive Generalized Scheduler for Grid Applications, in Proc. of the 19th Annual International Symposium on High-Performance Computing Systems and Applications (HPCS’05), pp.15-18, Guelph, Ontario Canada.

[7] H. Kawamura, M. Yamamoto, K. Suzuki, and A. Ohuchi, 2000 "Multiple ant colonies algorithm based on colony level interactions", IEICE Trans. Fundamentals, Vol. E83-A, No.2, pp. 371-379.

[8] J. Jong, and M. Wiering, 2001 "Multiple ant colony system for the bus-stop allocation problem", Proc of the Thirteenth Belgium Netherlands Conference on Artificial

Intelligence BNAIC’01, Amsterdam, Netherlands, pp. 141–148.

[9] Kruger .F, Middendorf .M, and Merkle. D,” 1998 Studies on a Parallel Ant System for the BSP Model” Unpub. Manuscript.