International Journal of Engineering Sciences Paradigms and Researches (IJESPR)
(Vol. 21, Issue 01) and (Publishing Month: June 2015)
(An Indexed, Referred and Impact Factor Journal)
ISSN (Online): 2319-6564
www.ijesonline.com

Minimization of Cost and Execution Time using HRSA Ant Colony Algorithm

Priyanka[1], Jitendra Kumar[2] and Milan bajaj[3]

[1]M. Tech. Scholar,
Rayat BahraI Institute of Mgt. & Technology, Chidana,Gohana, DCRUST, Murthal
pinky.info.dd@gmail.com

[2]HOD, CSE Department,
Rayat BahraI Institute of Mgt. & Technology, Chidana,Gohana, DCRUST, Murthal
jitendrakumar.03@gmail.com

[3]Assistant Professor, CSE Department,
Rayat BahraI Institute of Mgt. & Technology, Chidana,Gohana, DCRUST, Murthal
bajajmilan@hotmail.com

Abstract

Ant colony algorithm is an optimization technique which can be used in many NP hard problems like Resource Scheduling problem, TSP, knapsack problem etc. In this paper researcher has optimize the resource scheduling problem by Ant Colony algorithm. In order to implement this scheduling problem, researcher proposed an algorithm, an application have been developed which uses the simulated Grid environment i.e. GridSim. Researcher has implemented the application using JAVA programming language over the GridSim. The performance of the Heuristic Resource Scheduling Algorithm (HRSA) is compared with the Random Resource Scheduling Algorithm (RRSA) which randomly selects resources from the list of available resources.

Keywords: *Ant Colony Optimization, Pheromones, Optimization, Resource Scheduling Problem.*

I. Introduction

Ant Colony Algorithm: Ant System was developed by Marco Dorigo (Italy) in his PhD thesis in 1992.Max-Min Ant System developed by Hoos and Stützle in1996. Ant Colony was developed by Gambardella Dorigo in 1997[5].This bio inspired algorithm is based on a population of ants that perform a cooperative search [4].The ant colony algorithm is an algorithm for finding optimal paths that is based on the behaviour of ants searching for food. At first, the ants wander randomly. When an ant finds a source of food, it walks back to the colony leaving "markers" (pheromones) that show the path has food. When other ants come across the markers, they are likely to follow the path with a certain probability. If they do, they then populate the path with their own markers as they bring the food back. As more ants find the path, it gets stronger until there are a couple streams of ants travelling to various food sources near the colony. Because the ants drop pheromones every time they bring food, shorter paths are more likely to be stronger, hence optimizing the "solution." In the meantime, some ants are still randomly scouting for closer food sources. A similar approach can be used find near-optimal solution to the travelling salesman problem. Once the food source is depleted, the route is no longer populated with pheromones and slowly decays. The ACO algorithms is look like[3].

```
procedure ACO
begin
Initialize the pheromone
while stopping criterion not satisfied do
Position each ant in a starting node
repeat
for each ant do
Chose next node by applying the state transition rate
end for
until every ant has build a solution
Update the pheromone
end while
end
```

II. Related work

A lot of research had already been done in the field of distributed environment related to resource scheduling and load balancing and many algorithms are developed. However, they do not fully handle the case of heterogeneous resource capacities. In paper [17] author proposed that ant algorithm can be improved using some form of local search algorithm. Local search algorithm can be applied to the output of the ant algorithm to find the optimal resource to schedule a job. In paper [18] author proposed an ant colony algorithm for dynamic job scheduling in Grid environment. The next resource selection depends on the pheromone value and the transition probability. Author improved the existing ant colony algorithm and tried to minimize the total tardiness time of the job. In paper [19] author described an ant colony optimization algorithm in combination with local and tabu search. The author suggested that in ant colony algorithm the ants build their solutions using both information encoded in the pheromone trail also specific information in the form of the heuristic. The pheromone value updation rule is taken from the Max-Min algorithm in which the pheromone is only updated by the bets ant. And for heuristic information the Min-Min heuristic is used, which suggests that the heuristic value of a particular job should be directly proportional to the minimum completion time of the job. The local search algorithm is applied to each of the solutions built by the ants before the pheromone updation stage to take ant solution to its local optimum. The tabu search algorithm performs number of trails or iterations on the solution build by the ant colony algorithm and after each iteration the solution gets improved.

In paper [20] the author proposed heuristic scheduling algorithm that works in two phases. In the first phase heuristic begins with the set of all unmapped tasks then the set of minimum completion time is found like Min-Min heuristic. In the second phase there are two choices based on the threshold value (represents the percentage balance of load in the system).The idea is to select the pair of machine and task from the first phase so that the machine can execute its corresponding task effectively with a lower execution time in comparison with other machines and by comparing workload in the second phase. In paper [21] an improvement in the ant colony algorithm is proposed to reduce the make span and to converge towards the optimal solutions in a very faster manner. The algorithm proposed used the basic ant pheromone updating rule and modified transition updating rule. The proposed algorithm uses a new pheromone updating rule and probability matrix calculation formula in order to increase the efficiency the existing ant colony algorithm. The proposed algorithm is simulated with large data set and results show the improved make span.

In paper [22] the author proposed an ant colony algorithm with the modification in existing ant pheromone updating rule proposed by Marco and Dorgio in 1992 in his Ph.D. thesis. The proposed algorithm takes in to consideration the free time of resources and the execution time of the jobs to achieve better resource utilization and better scheduling. The proposed and the existing algorithm are simulated with GridSim toolkit with different number of tasks and it is found that the existing algorithm takes less time to execute the tasks.

III. Problem Statement

A computational Grid is a type of parallel and distributed system that enables the sharing, selection and aggregation of geographically distributed autonomous and heterogeneous resources dynamically at runtime depending on their availability, capability, performance, cost and user’s quality of service requirements. Scheduling a task to a particular resource in such a way so as to improve the make span and flow time is the foremost problem in the Grid computing environment because of the heterogeneous and dynamic nature of the resources. For mapping a task to a resource there is a Grid scheduler that receives the applications from the Grid users, selects the feasible resources for the applications according to acquired information from the Grid information service module and submits the resource to the selected resource. The Grid scheduler does not have control over the resources and also on the submitted jobs. Any machine can execute the any job, but the execution time differs. As compared with the expected execution time, the actual time may be varied at the time of running the jobs to allocated resource. The Grid scheduler’s aim is to allocate the jobs to the available nodes. The best match must be found from the list of available jobs to the list of available resources. The selection is based on the predictions of the computing power of the resource. So, lots of problem needed to be solved in this area. The Grid scheduler must allocate the job to the resources efficiently. The efficiency depends on two parameters one is execution time of the job and second is cost of using the resource. To minimize these parameters the workload has to be evenly distributed over all nodes which are based on their processing capabilities. It raises a new problem called Load Balancing.

The load balancing problem is closely related to the scheduling and resource allocation. It is concerned with all the techniques allowing an evenly distribution of the workload among the available resources in the Grid environment .The main objective of a load balancing consists primarily to optimize the average response time of applications by equal distribution of the load among available resources.

So we can summarize the problem statement as “Map the best resource to a task so that overall execution time of the task and cost of the resource can be minimized.”

IV. Proposed Algorithm (Heuristic Resource Scheduling Algorithm)

Proposed algorithm (HRSA) is inspired on an analogy with real life behaviour of a colony of ants when looking for food, and is effective algorithm for the solution of many combinatorial optimization problems. Investigations describe that ant has the ability of finding an optimal path from nest to food. On the way of ants moving, they lay some pheromone on the ground. 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. The probability of ant chooses a way is proportion to the concentration of a way's pheromone. To a way, the more ants choose, the way has denser pheromone, and the denser pheromone attracts more ants. Through this positive feedback mechanism, ant can find an optimal way finally.

In HRSA, the pheromone is associated with resources rather than path and ants are treated as tasks. The main objective of algorithm is reduction in total cost and execution time.

Heuristic Resource Scheduling Algorithm_HRSA (T, A, Q)

Let the number of tasks (ants) in task set T maintained by task agent A and the number of registered resources (pheromone) is in set Q.

Step 1: [Initialize Pheromone Value of Each Resource]

For every new resource R_i registered to Grid Information Server, the initial pheromone $I_p(0)$ is given as:

$$I_p(0) = [(N \times M) + C_s]/R_C \text{(Eq-1)}$$

Step 2: [Select Task]

Repeat 3 to 5 While ($T \neq \Phi$)

Select task *t*from task set *T*.

Step 3: [Select Resource]

Determine the resource $R_i$for task *t* having higher transitions probability *P* (higher pheromone intensity) Such as:

[Initially C_{pj}=I_{pj}]

$$P_j(t) = Max\left[\frac{[C_{pj}(t)^a]\times[I_{pj}(t)^b]}{(\sum_r[C_{pj}(t)]^a\times[I_{pj}(t)]^b)\times R_c}\right]\text{(Eq-2)}$$

[*j*, rare available resources.]

Step 4: [Schedule Task to Selected Resource]

Schedule task tto$R_j$and remove it from *T*

T=T-{t}

Step 5: [Update Pheromone]

For every resource R_j assigned a task pheromone is

[1- P_d = Pheromone Permanence]

$$C_{pj}^{new} = P_d \times C_{pj}^{old} + \Delta_j \text{(Eq-3)}$$

[Pheromone Variance $\Delta_j = -$Complexity of task (*C*)]

If (Task_Status= Succesful_Complete)

$$\Delta_j = \varphi \times C \text{(Eq-4)}$$

If (Task_Status= Failure)

$$\Delta_j = \Theta \times C \text{(Eq-5)}$$

Step 6: EXIT.

In the given algorithm the Step 1 initialize the pheromone value associated with each resource which depends on the number of processing elements (*N*) a resource have, execution speed (*M*) in terms of millions of instruction per second, the communication speed ($C_{s)}$ of the resources and inversely proportional to the cost ($R_{c)}$ of using a resource as shown in Equation 1.Step 2 selects the tasks t from the task set *T* until the task set gets empty. In Step 3 transition probability is used to select the resource having the maximum probability or pheromone value as shown in Equation 2 .The probability (*P*) of a resource to get selected is depends on the current pheromone($C_{p)}$ value, historic information i.e. initial pheromone (I_p) value and inversely proportional to the cost of using that particular resource. Task is assigned a resource R_j (selected in Step 3) in Step 4. In Step 5 the value of the pheromone is updated according to Equation 3, if task returns successfully then pheromone value is increased by encouragement argument (Φ) otherwise it is decreased by punishment argument(Θ) according to the Equation 4 and Equation 5 respectively. Same algorithm is repeated for each task t in task set. For simulation of the above algorithm the parameters and their values are given in next section.

V. Implementation

In this section, the Total Execution Time of Heuristic Resource Scheduling Algorithm (HRSA) is compared with Random Resource Scheduling Algorithm (RRSA) with the following parameters. Result are shown with the help of table.

Resource Allocation Policy=TIME_SHARED
Number of Resources =25
Number of Tasks = 10 to 50

Table 1

Sr. No.	Number of Tasks	Execution Time using HRSA(Sec.)	Execution Time using RRSA(Sec.)	Average Improvement %
1	10	5181	8910	71.97
2	20	18182	29597	62.78
3	30	38747	67431	74.02
4	40	66593	116923	75.57
5	50	101353	180192	84.82

The Table 1 show the result of Number of Tasks Vs. Execution Time in TIME_SHARED Allocation. And Table 2 shows the Number of Tasks Vs. Execution Cost in TIME_SHARED Allocation. Average Improvement in Total Execution Cost is = 29.55 %.Figure 2 shows that as the number of tasks increases the difference between execution cost the task acquired by two algorithms increases.

Table 2

Sr. No.	Number of Tasks	Execution Cost using HRSA(G/$)	Execution Cost using RRSA(G/$)	Average Improvement %
1	10	457	565	23.63
2	20	1416	2062	45.62
3	30	3334	4452	33.53
4	40	6077	7823	28.73
5	50	9383	10907	16.24

VI. Conclusion And Future Scope

To find the optimal resource for a job, we proposed an efficient Ant Colony Based Heuristic Resource Scheduling algorithm. The proposed HRSA (Heuristic Resource Scheduling Algorithm)and existing RRSA (Random Resource Scheduling Algorithm) are simulated using the open source JAVA based toolkit GridSim4.0.We used resource allocation policies i.e. TIME_SHARED (Round-Robin) to implement both the proposed and existing algorithm. Results confirms that whether we use TIME_SHARED resource allocation to implement the proposed algorithm, it will always take less execution time and cost in comparison of the existing algorithm(RRSA).

References

[1] Marco Dorigo and Thomas Stützle,"Ant Colony Optimization"
[2] K. Kousalya and P. Balasubramanie, "To Improve Ant Algorithm's Grid Scheduling Using Local Search". IJCC, Vol. 7, No. 4, December 2009.
[3] StefkaFidanova and MariyaDurchova ,"Ant Algorithm for Grid Scheduling Problem"
[4] Edson Flórez, Wilfredo Gómez and MSc. Lola Bautista,"AN ANT COLONY Optimization Algorthim For Job Shop Scheduling Problem" International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 4, No. 4, July 2013.
[5] AnirudhShekhawat, Pratik Poddar, Dinesh Boswa,"Ant colony Optimization Algorithms :Introduction and

Beyond"Indian Institute of Technology Bombay,Artificial Intelligence Seminar 2009.

[6] Baker M., Buyya R., and Laforenza D., "Grids and Grid Technologies for Wide-Area Distributed Computing".Software: Practice and Experience, vol. 32(15), pp. 1437-1466, Dec 2002.

[7] Berman F., Hey A. J. G., and Fox G., "Grid Computing: Making the Global Infrastructure a Reality". John Wiley & Sons, 2003.

[8] Foster I. and Kesselman C., "The Grid: Blueprint for a New Computing Infrastructure".Morgan Kaufmann, 2004.

[9] Gibbins H., Nadiminti K., Beeson B., Chhabra R., Smith B., and RajkumarBuyya, "The Australian BioGrid Portal: Empowering the Molecular Docking Research Community".APAC Conference and Exhibition on Advanced Computing, Grid Applications and eResearch (APAC 2005), Sept. 2005.

[10] RajkumarBuyya and Srikumarvenugopal, "A Gentle Introduction to Grid Computing and Technologies". CSI Communications, vol. 29(1), pp. 9-19, 2005.

[11] Krauter K., R. Buyya R., and Maheswaran M., "A Taxonomy and Survey of Grid Resource Management Systems for Distributed Computing". International Journal of Software: Practice and Experience, vol. 32, pp. 135-164, 2002.

[12] Abramson D, Giddy J, Kotler L. "High performance parametric modeling with Nimrod/G: Killer application for the global grid?",Proceedings of the 14th International Parallel and Distributed Processing Symposium (IPDPS2000).

[13] Buyya R, Abramson D, and Giddy J. "Nimrod/G: An architecture for a resource management and scheduling system in a global computational Grid". Proceedings of the International Conference on High Performance Computing in Asia–Pacific Region (HPC Asia 2000), 2000.

[14] Hoscheck W, Jaen-Martinez J, Samar A, Stockinger H, Stockinger K. "Data management in an international data Grid project". Proceedings 1st IEEE/ACM International Workshop on Grid Computing (Grid 2000), pp. 77–90, Dec 2000.

[15] Chervenak A, Foster I, Kesselman C, Salisbury C, Tuecke S. "The data Grid: Towards architecture for the distributed management and analysis of large scientific datasets." Journal of Network and Computer Applications, vol. 23(3), pp. 187–200.

[16] Nahrstedt K, Chu H, Narayan S. "QoS-aware resource management for distributed multimedia". Journal of High-Speed Networking (Special Issue on Multimedia Networking); vol. 7(3), pp. 227–255, Dec. 2000.

[17] K. Kousalya and P. Balasubramanie, "To Improve Ant Algorithm's Grid Scheduling Using Local Search". IJCC, Vol. 7, No. 4, December 2009.

[18] SiriluckLorpunmanee, Mohd Noor Sap, Abdul Hanan Abdullah, and Chai Chompoo-inwai, "An Ant Colony Optimization for Dynamic Job Scheduling in Grid Environment". International Journal of Computer and Information Science and Engineering, vol. 1(5), pp. 207-214, 2008.

[19] Ritchie,G.and Levine .J, " A hybrid ant algorithm for scheduling independent jobs in heterogenous computing envoirnment". in proceedings of the 23 rdworkshop of the UK Planning and scheduling special interest group,UNSPECIFIED,2004.

[20] Amid KhatibiBardsiri,MarjanKuchaki Rafsanjani, "A New Heuristic Approach based on Load Balancing for Grid Scheduling problem". Journal of Convergence Information Technology,Volume 7,Number 1,Janurary 2012.

[21] J. Rajesh Kumar,M.Kowsigan, " Efficient Scheduling in Computational Grid with an Improved Ant Colony Algorithm".International Journal of Computer Science and Technology,Vol 2,Issue 4,Oct-Dec,2011.

[22] P.Mathiyalagan ,S. Suriya,Dr. S.N Srivanadam,"Modified ant Colony Algorithm For Grid Scheduling ".IJCSE, VOl 2,No.02,2010.