Survey-ACO in Task Scheduling problem

Vinothina V
Department of Computer Science, Garden City College, Bangalore-39
Email: rvvino@yahoo.com
Dr.R.Sridaran
Faculty of Computer Applications, Marwadi Education Foundation's Group of Institutions, Rajkot, Gujarat, India.
Email: sridaran.rajagopal@gmail.com

ABSTRACT

Ant Colony Optimization (ACO) algorithms are probabilistic model inspired by the social behaviour of ants. Initially ACO is applied for travelling salesman problem. Later it is used for vector routing problem, scheduling problem, assignment problem, set problem, image processing, data mining, distributed information retrieval and others. By looking at the strength of ACO, they are the most appropriate for scheduling tasks. In this paper, we discussed the application of ACO in scheduling problem under different circumstances.

Keywords - Ant Colony Algorithm, Cloud Computing, Combinatorial Optimization Problem, Heterogeneous Environment, Real time systems, Resource Constrained problem, Scheduling.

I. INTRODUCTION

The three most successful ACO algorithms are Ant System, Ant Colony System and Max-Min Ant System.The first ACO algorithm was called the Ant system and it was aimed to solve the travelling salesman problem, in which the goal is to find the shortest round-trip to link a series of cities. Ant system (AS) was the first ACO algorithm to be proposed in the literature [4]. Its main characteristic is that the pheromone values are updated by *all* the ants that have completed the tour.

The first major improvement over the original ant system to be proposed was ant colony system (ACS), introduced by Dorigo and Gambardella in [5]. The first important difference between ACS and AS is the form of the decision rule used by the ants during the construction process. It is an algorithm for finding optimal paths that is based on the behavior of ants searching for food. At first, the ants wander randomly around their environment. 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 [2]". The social behaviors of ants are depicted in Fig.1.

The Ant Colony System algorithm is an example of an Ant Colony Optimization method from the field of Swarm Intelligence, Metaheuristics and Computational Intelligence. Ant Colony System is an extension to the Ant System algorithm and is related to other Ant Colony Optimization methods such as Elite Ant System, and Rank-based Ant System [3].

MAX-MIN ant system (MMAS) is another improvement, proposed by Stützle and Hoos [6], over the original ant system idea. MMAS differs from AS in that (i) only the best ant adds pheromone trails, and (ii) the minimum and maximum values of the pheromone are explicitly limited. The initial applications of ACO were in the domain of NP-hard combinatorial optimization problems and telecommunication networks. Concerning applications, the use of ACO for the solution of dynamic, multiobjective, stochastic, continuous and mixed-variable optimization problems is a current hot topic, as well as the creation of parallel implementations capable of taking advantage of the new available parallel hardware.

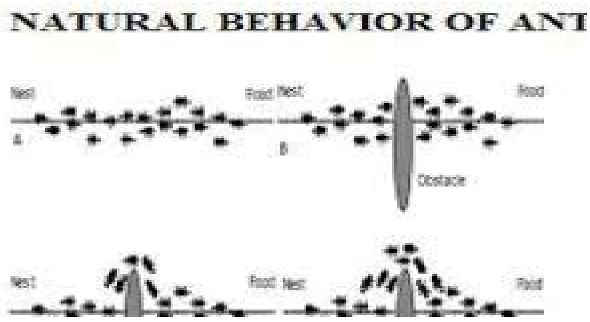

Fig.1 Social Behavior of ants

The Cloud computing is a kind of new computing paradigm that works for solving the new problem which combines the different computers to constitute a big computing system to execute some large tasks [1]. A major challenge in cloud computing is task scheduling and efficient use of resources. But the task scheduling is NP-hard and it is very difficult to attain optimal solution. The characteristics of cloud computing resources such as heterogeneous, dynamic and self-governing make the scheduling problems more complex. The nature of ACO algorithm such as robustness and self -adaptability can just match the characteristics of cloud computing. The ACO also used in grid computing task scheduling but doesn't get good performance.

The rest of the paper is organized as follows: In section II, motivation of this paper is presented. In section III, some of the scheduling strategies using ACO are addressed. Future directions of our ongoing research have been presented in section IV.

II. MOTIVATION

In paper [7], Marco Dorigo et.al reviewed some convergence results on ACO algorithms and discussed relations between ant colony optimization algorithms and other approximate methods for optimization. Finally, they focused on some research efforts directed at gaining a deeper understanding of the behavior of ant colony optimization algorithms. Zhu Qingdao et.al analyzed the convergence of ACO algorithm for travelling sales man problem and shown that the algorithm converged under the condition $0< q_0 <1$.Many algorithms have been proposed using ACO for task scheduling in heterogeneous and resource constrained environment. Resource Allocation in cloud is NP-Hard Problem. ACO has been proved to perform well in combinatorial problem and other complex problems [8]. Hence the following section focuses on Scheduling strategy using ACO.

III. SCHEDULING STRATEGIES USING ACO

In computing, scheduling is the method by which threads, processes or data flows are given access to system resources. This is usually done to load balance and share system resources effectively or achieve a target quality of service. The need for a scheduling algorithm arises from the requirement for most modern systems to perform multitasking and multiplexing. The scheduler is concerned mainly with:

- Throughput
- Latency
    - Turnaround time
    - Response time.
- Fairness - Equal CPU time to each process
- Waiting Time

In practice, these goals often conflict (e.g. throughput versus latency), thus a scheduler will implement a suitable compromise [9]. Preference is given to any one of the concerns mentioned above, depending upon the user's needs and objectives. In real-time environments, such as embedded systems for automatic control in industry and scientific applications, the scheduler also must ensure that processes can meet deadlines; this is crucial for keeping the system stable. Scheduled tasks can also be distributed to remote devices across a network and managed through an administrative back end.

A major advantage of ACO over other meta-heuristic algorithms is the problem instance may change dynamically. The decisions made by all ants are purposeful and output of all ants is utilized in each iteration to construct the new optimal solution. Fig.2 represents the outline of ACO algorithm.

The idea of ant colony algorithm comes from the study of searching capability of ant populations. The best solution is determined by judging changes in pheromone concentration. Ling Ding et.al in [10] proposed scheduling algorithm using ACO aimed at improving system availability and maintaining an ideal response time of submitted tasks. They introduced the concept of computational heterogeneity and availability heterogeneity. The computational weight of class i on node j is defined as the ratio between its service rate on node j and the fastest service rate in the system. The computational heterogeneity can be measured by the standard deviation of the computational weights. The heterogeneous availability is measured by the standard deviation of the availability offered by all the nodes in the system.

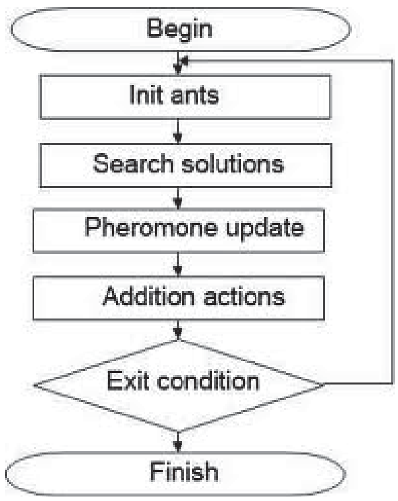

Fig.2Flowchart of ACO algorithm

Weifeng Sun et.al [1] proposed an independent task scheduling algorithm Period based ACO (PACO). It takes the scheduling period into consideration. They used two pheromones values namely real-time pheromone intensity τ_j (t) and innate pheromone intensity η_j. The ant chooses a resource for a task with minimum probability according to (1).

$$p^k{}_j(t) = \frac{[\tau_i(t)\ ]^{-\mathrm{A}} \bullet [\eta_j\ ]^{\beta}}{\sum_s [\tau_s(t)\ ]^{\alpha} \bullet [\eta_s(t)\ ]^{\beta}}$$ s ε available resource (1)

When a resource is selected, the pheromone intensity of it will be updated according to (2)

$$\tau_j(t) = \tau_j(t) - \theta K \quad (2)$$

K is the length of the task assigned on to resource j at time t, and θ (0<θ<1) is also a variable parameter. After all tasks are assigned, compute the total execution time and check whether it is the shortest execution time in current. If so, modification of the real time pheromone intensity according to (2) will be cancelled and then update the real-time pheromone intensity according to (3).

$$\tau_i(t) = (1-\rho)\tau_i(t) + \rho\Delta\tau_i \quad (3)$$

In PACO, when a task is assigned onto resource, the pheromone intensity of the selected resource will be reduced. That will lower the chance that other tasks select the resource and promote the load balance.

Shan et.al [11] hybridized particle swarm and Ant Colony algorithms for multi-mode resource constrained project scheduling problem with minimum time lag. The related parameters in ACO algorithms are adjusted by using PSO algorithm to make ACO algorithm achieve the best performance. The solution construction ends after each ant has completed a scheduling, that is, after each ant has constructed a sequence of activity in some process model. Next the pheromone trails are updated. In ACO, this is done by first lowering the pheromone trails by a constant factor and then allowing each ant to deposit pheromone on the arcs belong to its scheduling:

$$\tau_{ij}(t+1) = (1-\alpha)\tau_{ij}(t) + \sum_{k=1}^{m} \Delta\tau_{ij}{}^k(t) \ \ \forall (i,j) \quad (4)$$

$\Delta\tau_{ij}{}^k(t)$ is the amount of pheromone ant k deposits on the arcs: it is defined as

$\Delta\tau_{ij}{}^k(t) = \{\ 1/D^k(t)i$ if arc (i, j) is used by an ant K

0 otherwise

PSO‘s fitness values is the makespan of the best scheduling computed by an ACO using the related parameters in the given instance. They considered better of those parameters that minimize the makespan of the scheduling and secondly the time of computing.

In paper [12], authors used Ant Colony System and MinMin Ant System called MMACS to solve RCPSP. The algorithm contains four key modules: the behaviour of searching path, heuristic information and the mechanism of updating pheromone, schedule generation scheme. The pseudorandom rule [15] is used for choosing next activity. They applied priority rule LFT to calculate heuristics information. The pheromone is updated by only best ant so for according to the formula (5):

$$\tau_{ij}(t+n) = \tau_{ij}(t) + \Delta\tau_{ij}^{bs}(t, t+n) \quad (5)$$

This algorithm guarantees the precedence constraints between activities needn't calculate the location of successors and improves the time efficiency.

Apurva shah et.al proposed ACO based algorithm for real-time operating system. This algorithm is required to execute when a new task arrives or presently running task completes. It constructed the tour of different of ants and produced the task execution sequence. The generated task sequence are analysed and pheromone value is updated. Then based on the probability of each task, tasks are selected for execution. The probability is found using equation (6).

$$p_i(t) = \frac{(\tau_i(t)\ )^{\alpha} * (\eta_i(t))^{\beta}}{\sum_{l \in R_1} (\tau_l(t))^{\alpha} * (\eta_l(t))^{\beta}} \quad (6)$$

Pheromone is updating on each node is done in two parts as pheromone evaporation to forget the bad journey and pheromone Laying for two best journey.

Ant colony optimization used along with Monte-Carlo sampling for estimating the objective of combinatorial optimization problems with deterministic constraints. The pheromone updating mainly decides the ant construction solution [15]. Walter et.al had shown that on including that of linear increment of the sample size, the algorithm converges to the global optimal solution with probability one.

Diversity control is a crucial aspect of meta-heuristics and becomes even more important when continuous adaptiveness to dynamic problems is required. The author of paper [15] investigated the influence of α value and the critical phase of the convergence behaviour in ACO algorithm. They used dynamic α value and dynamic restart method to maintain diversity throughout the search in problem space which proved to effectively explore the search space in vicinity of the optimum.

The first paragraph under each heading or subheading should be flush left, and subsequent paragraphs should have a five-space indentation. A colon is inserted before an equation is presented, but there is no punctuation following the equation. All equations are numbered and referred to in the text solely by a number enclosed in a round bracket (i.e., (3) reads as "equation 3"). Ensure that any miscellaneous numbering system you use in your paper cannot be confused with a reference [4] or an equation (3) designation.

IV. CONCLUSION

ACO algorithm is used in task scheduling by considering different parameters for pheromone updating. Through this paper, we can know that pheromone updating equation and visit probability equation are the most important in ACO algorithm. Therefore, the values of related parameters have great impact on the performance of ACO algorithm. It also proved that both performance and convergence of ACO algorithms is improved with Particle Swarm Optimization algorithm. Hence it is significant that theoretically extending the ACO algorithms parameters instead finding the application of ACO algorithms.

REFERENCES

[1] Weifeng Sun, Ning Zhang, Haotian Wang, wenjuan Yin, Tie Qiu, PACO: A Period ACO_based Scheduling Algorithm in Cloud Computing, *International conference on Cloud Computing and Big Data*.482-486,2013.

[2] Macura, Wiktor K. "Ant Colony Algorithm." From MathWorld--A Wolfram Web Resource, created by Eric W. Weisstein. http://mathworld.wolfram.com/AntColonyAlgorithm.html .

[3] Jason Brownlee “Clever Algorithms: Nature-Inspired Programming Recipes” http://www.cleveralgorithms.com/nature-inspired/swarm/ant_colony_system.html.

[4] M. Dorigo, V. Maniezzo, and A. Colorni. Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics – Part B, 26(1):*29–41, 1996.

[5] M. Dorigo and L. M. Gambardella. Ant Colony System: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation, 1(1):*53–66, 1997.

[6] T. Stützle and H. H. Hoos. MAX–MIN Ant System. *Future Generation Computer Systems, 16(8)*:889–914, 2000.

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

[8] A.Colorni, M.Dorigo et V.Maniezzo, “Distributed Optimization by ant Colonies”, *actes de la premiere conference europeenne sur la vie artifficicllc, Paris, France and Elsevier Publishing*, 134-142,1991.

[9] http://en.wikipedia.org/wiki/Scheduling_(computing)

[10] Ling Ding, Ping Fan, Bin wen, “A Task Scheduling algorithm for Heterogeneous System using ACO”, *2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation*, 2013.

[11] Shan Miyuan, Wu Juan, Peng Danni, “Particle Swarm and Ant Colony Algorithms Hybridized for Multi-mode Resource –constrained Project Scheduling Problem with Minimum Time Lag”, IEEE.2007.

[12] Yumiao Zhou, Qingshun Guo, Rongwei Gan, “Improved ACO algorithm for Resource –Constrained Project scheduling Problem”, *International Conference on Aritificial Intelligence and computational Intelligence*, 2009.

[13] Apurva Shah, Ketan Kotecha, “Scheduling algorithm for Real-Time Operating Systems using ACO”, *International conference on computational Intelligence and communication networks*, 2010.

[14] Walter J.Gutjahr “A Converging ACO algorithm for stochastic combinatorial optimization”, *SAGA Stochastic Algorithms: Foundations and Applications*, 2003.

[15] Bernd Meyer, “On the convergence behavior of ant colony search”, *Complexity International*, 2005.
[16] Zhu Qingbao, Wang Lingling, “The analysis of the convergence of ant colony optimization algorithm”, *Front.Electr.Electron.Eng.China 268-272*, 2007.

Biographies and Photographs

Ms.Vinothina M.Sc, is currently pursuing her Ph.D. in an emerging area of Cloud Computing. She has done her Post Graduation and Under Graduation in Computer Science. She has also completed M.B.A. with specialization in Systems and M.Phil. in Image Processing. She has presented papers in various conferences, attended many workshops and Seminars. Her areas of academic interest includes Object Oriented Programming, Web Technologies and Cloud computing.

Dr. R. Sridaran is a Ph.D in Computer Science from Madurai Kamaraj University. He has published 25+ research papers in leading journals and presented in many conferences. He is presently guiding eight research scholars in the areas of Cloud Computing, e-learning and Software Engineering. He has got 20 years of academic experience and served in leading educational institutions at different capacities. He is currently the Dean of Faculty of Computer Applications, Marwadi Education Foundation, Rajkot.