A Comprehensive Study on Ant Colony Optimization

[1]Nicy Johnson [2]Shejina N M [3] Dr. S Brilly Sangeetha
[1] Assistant Professor, Computer Science and Engineering [2] Assistant Professor, Computer Science and Engineering [3] Associate Professor, Computer Science and Engineering

Abstract*— All networks tend to become more and more complicated. They can be wired, with lots of routers, or wireless, with lots of mobile nodes. The problem remains the same: in order to get the best from the network, there is a need to find the shortest path. The more complicated the network is, the more difficult it is to manage the routes and indicate which one is the best. The Nature gives us a solution to find the shortest path. The ants, in their necessity to find food and brings it back to the nest, manage not only to explore a vast area, but also to indicate to their peers the location of the food while bringing it back to the nest. Thus, they know where their nest is, and also their destination, without having a global view of the ground. Most of the time, they will find the shortest path and adapt to ground changes, hence proving their great efficiency toward this difficult task. The purpose of this project is to provide a clear understanding of the Ants-based algorithm, by giving a formal and comprehensive systematization of the subject. The simulation developed in Java will be a support of a deeper analysis of the factors of the algorithm, its potentialities and its limitations.

Index Terms—*Double bridge experiment, Metaheuristic*

I. INTRODUCTION

Ant colonies are distributed systems that present a highly structured social organization. As a result of this organization, ant colonies can accomplish complex tasks that in some cases far exceed the individual capabilities of a single ant. Initially, real ants' behavior is observed and derived "ant algorithms" studies models which can be used for the solution of optimization and distributed control problems.

II. FROM REAL ANTS TO ARTIFICIAL ANTS

A. Foraging Behavior of Ants

Pheromones are the chemicals used by the ants which are used for most of the communication among individuals and the environment on ant colonies. A specific type of pheromone called trail pheromone is used by some ant species such as Lasius niger or the Argentine ant Iridomyrmex humilis for making paths on the ground, for example, paths from food sources to the nest. Foragers can follow the path to food discovered by other ants by sensing pheromones trails. The source of Ant colony optimization is inspired by collective trail-laying and trail-following behavior whereby an ant is influenced by a chemical trail left by other ants.

B. Double Bridge Experiments

Double bridge experiment was designed by Deneubourg and colleagues who used a double bridge connecting a nest of ants of the Argentine ant species I. humilis and a food source. l_l was the length of the longer branch and l_s the length of the shorter branch. They ran experiments varying the ratio $r = l_l / l_s$ between the length of the two branches of the double bridge.

The bridge had two branches of equal length ($r = 1$) in the first experiment. Initially ants were left free to move between the nest and the food source and the percentage of ants that chose one or the other of the two branches were observed over time. The outcome was that although random choices occurred in the initial phase, eventually all the ants used the same branch. When a trial starts there is no pheromone on the two branches. Hence, the ants do not have a preference and they select with the same probability any of the branches. A few more ants will select one branch over the other after some time. The larger number of ants on a branch, the larger will be the amount of pheromone on that branch; this larger amount of pheromone in turn stimulates more ants to choose that branch again, and so on until finally the ants converge to one single path. This autocatalytic process is an example of a self-organizing behavior of the ants.

The length ratio between the two branches was set to $r = 2$ in the second experiment. In this case, in most of the trials, after some time all the ants chose to use only the short branch. When they start moving they start accumulating pheromones. The ants choosing the short branch are the first to reach the food and to start their return to the nest. Therefore, pheromone starts to accumulate faster on the short branch, and finally most of the ants start using the shorter branch.

III. MINIMUM COST PATHS FROM ARTIFICIAL ANTS

Consider a static, connected graph $G = (N,A)$, where N is the set of $n = |N|$ nodes and A is the set of undirected arcs connecting them. Source and destination nodes are the two points between which we want to establish a minimum cost path.

The ants, while building a solution, may generate loops. Loops tend to become more and more attractive and ants can get trapped in them as a consequence of the forward pheromone trail updating mechanism. An ant can escape such loops. Even then the overall pheromone trail distribution becomes such that short paths are no longer favored and the mechanism that in the simpler double bridge situation made the ant choose the shortest path with higher probability does not work anymore. It might seem that the simplest solution to this problem would be the removal of the forward updating mechanism, because this problem is due to forward pheromone trail updating. In this way ants would rely only on backward updating. For allowing the ants to efficiently build solutions to the minimum cost path problem, the ants can implement a number of useful behaviors via the use of memory. Those behaviors are (1) probabilistic solution construction biased by pheromone trails, without forward pheromone updating; (2) deterministic backward path with loop elimination and with pheromone updating; (3) evaluation of the quality of the solutions generated and use of the solution quality in determining the quantity of pheromone to deposit and (4) pheromone evaporation.

IV. Experiments with S-ACO

So many experiments were run to evaluate the importance of some aspects of S-ACO: evaporation, number of ants, and type of pheromone update (function of the solution quality or not). In the experiments presented in the following the behavior of S-ACO is judged with respect to convergence toward the minimum cost (shortest) path, in a way similar to what was done for the outcome of the simulation experiments of Deneubourg et al. and for the experiments with the discrete model. As the algorithm runs for an increasing number of iterations, the ants' probability of following the arcs of a particular path increases—in the limit to a point where the probability of selecting the arcs of this path becomes arbitrarily close to 1 while for all the others, it becomes arbitrarily close to 0.

A. Number of Ants and Types of Pheromone Update: Experiments with the Double Bridge

We ran a first set of experiments in which we studied the influence that the number of ants used and the way the amount of pheromone to be deposited is determined by ants have on the behavior of S-ACO. The experiments were run using the double bridge. The choice of the double bridge was due to the desire of comparing the results obtained with S-ACO to those obtained with the model of real ants' behavior. A major difference between that model and S-ACO is that all other equations describe the average behavior of the system, whereas in S-ACO a fixed number of ants move autonomously on the graph. Intuitively, an increasing number of ants in S-ACO should approximate better and better the average behavior given by other equations.

B. Pheromone Evaporation: Experiments with the Extended Double Bridge

In a second set of experiments, we studied the influence that pheromone trail evaporation has on the convergence behavior of S-ACO. Experiments were run using the extended double bridge graph. In these experiments the ants deposit an amount of pheromone that is the inverse of their path length; also, before depositing it, they eliminate loops using the procedure. To evaluate the behavior of the algorithm we observe the development of the path lengths found by the ants. In particular, we plot the moving averages of the path lengths after loop elimination (moving averages are calculated using the most recent paths found by the ants, where m is the number of ants). In other words, in the graph of a point is plotted each time an ant has completed a journey from the source to the destination and back (the number of journeys is on the x-axis), and the corresponding value on the y-axis is given by the length of the above-mentioned moving average after loop elimination.

V. ACO - Metaheuristic

A set of algorithmic concepts that can be used to define heuristic methods applicable to a wide set of different problems is called metaheuristic. Metaheuristic has significantly increased the ability of finding very high quality solutions to hard, practically relevant combinatorial optimization problems in a reasonable time.

Optimization problems involve finding values for discrete variables such that the optimal solution with respect to a given objective function is found. Common examples are the shortest-path problems, finding a minimum cost plan to deliver goods to customers, an optimal assignment of employees to tasks to be performed, a best routing scheme for data packets in the Internet, an optimal sequence of jobs which are to be processed in a production line, an allocation of flight crews to airplanes etc.

The combinatorial optimization problem is either a maximization or a minimization problem which has associated a set of problem instances. The problem means the general question to be answered, usually having several parameters or variables with unspecified values. The instance means problem with specified values for all the parameters. The traveling salesman problem (TSP) is an example which is the general problem of finding a minimum cost Hamiltonian circuit in a weighted graph, while a particular TSP instance has a specified number of nodes and specified arc weights.

A. Computational Complexity

A straightforward approach to the solution of combinatorial optimization problems would be exhaustive search, that is, the enumeration of all possible solutions and the choice of the best one. Unfortunately, in most cases, such a naive approach becomes rapidly infeasible because the number of possible solutions grows exponentially with the instance size n, where the instance size can be given, for example, by the number of binary digits necessary to encode the instance. For some combinatorial optimization problems, deep insight into the problem structure and the exploitation of problem-specific characteristics allow the definition of algorithms that find an optimal solution much quicker than exhaustive search does. In other cases, even the best algorithms of this kind cannot do much better than exhaustive search.

B. Solution Methods for NP-Hard Problems

Two classes of algorithms are available for the solution of combinatorial optimization problems: exact and approximate algorithms. Exact algorithms are guaranteed to find the optimal solution and to prove its optimality for every finite size instance of a combinatorial optimization problem within an instance-dependent run time. In the case of NP-hard problems, exact algorithms need, in the worst case, exponential time to find the optimum. Although for some specific problems exact algorithms have been improved significantly in recent years, obtaining at times impressive results, for most NP-hard problems the performance of exact algorithms is not satisfactory. So, for example, for the quadratic assignment problem (QAP), an important problem that arises in real-world applications and whose goal is to find the optimal assignment of n items to n locations, most instances of dimension around 30 are currently the limit of what can be solved with stateof- the-art exact algorithms. Despite the small size of the instance, the computation time required to solve it is extremely high. In addition to the exponential worst-case complexity, the application of exact algorithms to NP-hard problems in practice also suffers from a strong rise in computation time when the problem size increases, and often their use quickly becomes infeasible. If optimal solutions cannot be efficiently obtained in practice, the only possibility is to trade optimality for efficiency. In other words, the guarantee of finding optimal solutions can be sacrificed for the sake of getting very good solutions in polynomial time. Approximate algorithms, often also loosely called heuristic methods or simply heuristics, seek to obtain good, that is, near-optimal solutions at relatively low computational cost without being able to guarantee the optimality of solutions. Based on the underlying techniques that approximate algorithm use, they can be classified as being either constructive or local search methods.

CONCLUSION

This project tried to cover the state-of-the-art studies about Ant Colony Optimisation (ACO) algorithm and its application to routing protocols. It has been a great pleasure to study these papers. At the beginning of this project we developed our own application which simulates the ACO, which gave me the best understanding of the algorithm and its issues. Thus, the applications to the routing protocols are easier to understand since the main ideas behind them have always been inspired by the ants. Ant Colony Optimization is a well defined and good performing metaheuristic technique that is applied to solve the complex combinatorial problems. Ant Colony Optimization is a population- based metaheuristic which exploits an indirect form of memory of previous performance. In this paper, we have reviewed the ideas of this approach that lead from the biological inspiration to ACO metaheuristic. Most of the existing approaches have been described. The main difference between the various Ant System extensions consist of the techniques used to control the search process.

REFERENCES

[1] S. Kirkpatrick; C. D. Gelatt; M. P. Vecchi, “Optimization by Simulated Annealing,” Science- New Series, Vol. 220, pp. 671-680, 1983.

[2] E. H. L.Aarts and J. K. Lenstra, “Local Search in Combinatorial optimization,” John Wiley & Sons, Chichester, 1997.

[3] D. E. Goldberg, “Genetic Algorithms in Search, Optimization and Machine Learning,” Addison –Wesley, 1989.

[4] S. Voss, S. Martello and I.H. Osman, “Meta-Heuristics: Advances and trends in Local Search Paradigms for Optimization,” Kluwer Academic Publishers, Boston, MA, 1999.

[5] F. Glover and M. Laguna,,“Tabu Search,” Kluwer Academic Publishers, Boston, MA, 1997.

[6] M. Dorigo, V. Maniezzo, and A. Colorni, “The Ant System: Optimization by a colony of cooperating agents,” IEEE transcations on Systems, Man, and Cybernetics, vol. 26, pp. 29-41, 1996.

[7] M.Dorigo and T. Stutzle, “Ant Colony Optimization,” MIT Press, Cambridge, MA, 2004.

[8] E. Bonabeau, M. Dorigo, and G. Theraulaz, “Swarm Intelligence: From Natural to Artificial Systems,” Oxford University.