3rd International Conference on Electrical & Computer Engineering
ICECE 2004, 28-30 December 2004, Dhaka, Bangladesh

ANT SYSTEM TO FIND THE SHORTEST PATH

Md. Mujibur Rahman Khan

Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology (BUET)
Dhaka, Bangladesh
E-mail: mujib1295@yahoo.com

ABSTRACT

An analogy with the way ant colonies function has suggested the definition of a new computation paradigm - known as Ant System. It is a viable new approach to stochastic combinatorial optimization. This paper presents a model of the Ant System (AS) that can efficiently be applied to solve the classical Shortest-Path problem. The main characteristics of the model are positive feedback, distributed computation, and the use of constructive greedy heuristic. Positive feedback accounts for rapid discovery of good solutions, distributed computation avoids premature convergence, and the greedy heuristic helps find acceptable solutions in the early stage of the search process.

1. INTRODUCTION

The intelligent behaviour of some insects like ants has been amazed the scientist for hundreds of years. The more research and observations are performed, the more interesting facts are coming out. In fact, research on the behavior of ants has greatly inspired the works of many scientists all over the world (see [1], [2], [4]). From the observation [3] proposed a new paradigm of computation – model of real ants – the Ant System. It is proposed as a viable new approach to stochastic combinatorial optimization. In this approach the search activities are distributed over so-called "ants", that is, agents with very simple basic capabilities, which to some extent, mimic the behavior of real ants. The main characteristics of this model are positive feedback, distributed computation, and the use of a constructive greedy heuristic [3]. Positive feedback accounts for rapid discovery of good solutions, distributed computation avoids premature convergence, and the greedy heuristic helps find acceptable solutions in the early stages of the search process. The AS and is used in solving many combinatorial problems. In this paper we present a solution to the classical Shortest Path Problem using the Ant System.

1.1 The Ant System

The AS is introduced and best described in [3]. Here we present a model to solve the Shortest Path problem. The model is derived from the study of real ant colonies. As the model is not intended to simulating ant colonies, but to use of artificial ant colonies as an optimization tool, our AS have some major differences with a real (natural) one:

- artificial ants have some memory,
- they are not be completely blind, and
- they live in an environment where time is discrete.

The basic idea is to leave a number of ants in the problem space, then let them wonder around with some given rules to find out the optimal solution. Each ant is a simple agent with the following characteristics:

- it chooses the vertex to go to with a probability that is a function of the vertex distance and of the amount of trail present on the connecting edge;
- to force the ant to make legal[1] tours, transitions to already visited vertices are disallowed until a tour is completed;
- when it completes a tour[2], it lays a substance called *trail* on each edge ***(i,j)*** visited.

1.2 The model

Let $b_i(t)(i = 1,2,...,n)$ be the number of ants in vertex i at time t and let $m = \sum_{i=1}^{n} b_i(t)$ be the total

[1] By legal tour we understand a path with no loop, i.e. no repetition of vertex.
[2] A tour is completed when an ant reaches to the destination from the source or vice versa.

ISBN 984-32-1804-4

number of ants. Let $\tau_{ij}(t)$ be the *intensity of trail* on edge *(i,j)* at time *t*. Each ant at time *t* chooses the next vertex, where it will be at time *t+1*. Therefore, in one *iteration* of the AS algorithm the *m* moves carried out by the *m* ants in the interval *(t, t+1)*. If any of the ants a completes a tour P_a after any iteration of the algorithm, the trail intensity is updated according to the following formula

$$\tau_{ij}(t+n) = \rho.\tau_{ij}(t) + \Delta\tau_{ij}, \qquad if\,(i,j) \in P_a \;... (1)$$

and $\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$

where, n is the tour length (in other words, the number of iterations required to complete this tour) and $(1-\rho)$ represents the evaporation of trail between time *t* and *t+n.* Here $\Delta\tau_{ij}^{k}$ is the quantity per unit of length of trail substance (pheromone in real ants) laid on edge *(i,j)* by the k-th ant between time *t* and *t+n* and equals to Q (a constant) if k-th ant uses edge *(i,j)* between time *t* and *t+n* Otherwise Zero. Thus we are following *ant-density* model proposed by [3]. Another factor is visibility η_{ij} the quantity $1/d_{ij}$, where d_{ij} is the length of edge *(i,j)*. The transition probability that an ant select the edge *(i,j)* is proportional to the trail and visibility and may be expressed as –

$$p_{ij}^{k} = \frac{[\tau_{ij}(t)]^{\alpha}.[\eta_{ij}]^{\beta}}{\sum[\tau_{ik}(t)]^{\alpha}.[\eta_{ik}]^{\beta}} \;\; ...... (2).$$

The parameters considered here are those that affect directly or indirectly the computation of the probability:

α: the relative importance of the trail, α≥0;
β: the relative importance of the visibility, β≥0;
ρ: trail persistence, 0≤ρ<1 (1-ρ can be interpreted as trail evaporation);
Q: a constant related to the quantity of trail laid by ants (used to compute $\Delta\tau_{ij}$).

2. THE ALGORITHM

2.1 Overview

For a weighted graph ***G***, the source ***S*** and the Destination ***D*** are given. All the edges of ***G*** are initially given a positive equal amount of trail. We put ***N*** = |***G***| ants in ***S*** and same number of ants in ***D***. That is the algorithm performs a bidirectional search to minimize the chance of local maxima. Each ant is associated with a list of vertices ***T*** that are already visited so that there is no loop. The ***i***th ant is associated with $\boldsymbol{T_i}$. If the two extreme vertices of the list of any ant are ***S*** and ***D*** (regardless of their order), the ant has completed its tour. At this time the trail intensity of all the edges are updated. Now the ant starts a new tour from its current location towards the other end perhaps following a different path.

2.2 Outline

The outline of our proposed algorithm is given below:

1. Initially take **2x*N*** ants. Initialize $\boldsymbol{T_i}$ by ***S*** for ***i = 1, 2, …, N*** and by ***D*** for ***i = N+1, N+2, …, 2N***. Set the ***cycle counter***[3] to zero. Set the length of ***local solution***[4] to the maximum possible value.

2. For ***i = 1 to 2N*** do
   Select the next vertex $\boldsymbol{v \notin T_i}$ with probability ***P*** using the formula (2);
   Append ***v*** to $\boldsymbol{T_i}$;

3.
   a. If any ant has completed the tour at this time, update the trail intensity for all the edges using the formula (1).
   b. Find the shortest path among the paths found by the ants at this time. Compare it with the shortest path found so far. Store the shorter one as the *local solution*.
   c. For each ant that has just completed a tour, remove all but the last vertex from its associated list.

4. Increment the *cycle counter* by 1. If it reaches the maximum cycle count or no further improvement for the last ***M***[5] cycles then return the *local solution* and stop; otherwise go to step 2.

The local solution returned by the algorithm from step 4 is the shortest path between ***S*** and ***D***.

3. EXPERIMENTAL RESULT AND DISCUSSION

We have implemented our algorithm using C++ language and MS VC++ environment under Windows XP platform in a PIII 730MHz, 128MB

[3] It will be used to count the number of ant-cycles.
[4] It will store the shortest path found so far. When the algorithm finishes, it will be holding the global solution.
[5] We have used ***3xN*** cycles as the value of ***M***. But we are trying to find out an optimal range value for ***M***.

RAM machine. We considered randomly generated graphs with 6 vertices to 25 vertices for our primary experimentation. For each of the cases, we have taken randomly generated source and destination. We also performed an exhaustive method to find the shortest path so that the performance of the AS can be verified. The results obtained from primary experimentation are found to be promising. The parameter values we used are $\alpha = 1, \beta = 5$, $\rho = 0.99$ and Q = 100 as per the general guideline given in [3]. But we are still conducting experimentations to identify the range of the parameters within which the AS performs best.

4. CONCLUSION

We presented a solution to the classical Shortest Path problem. We used the recently investigated and discovered paradigm of computation, known as the Ant System. We first presented a model of AS for the specific problem. Then we proposed an algorithm to solve the problem using our AS model. Further research could be carried on to find better definition of the model and the algorithm for this specific problem. The ranges of parameter values have a great influence on the performance and efficiency of the system. So finding the optimal ranges of the parameters are also of great importance.

REFERENCE

[1] Deneubourg, J.L., Pasteels, J.M. and Verhaeghe, J.C. "Probabilistic Behaviour in Ants: a Strategy of Errors?", Journal of Theoretical Biology, 105, 259–271, 1983.

[2] Deneubourg, J.L. and Goss, S. "Collective patterns and decision-making", Ethology, Ecology & Evolution, Vol.1, 295–311, 1989.

[3] Dorigo, M., Maniezzo, V. and Colorni, A. "The Ant System: Optimization by a colony of cooperating agents", IEEE Transactions on Systems, Man, and Cybernetics–Part B, Vol.26, No.1, 1996, pp.1-13

[4] Goss, S., Beckers, R., Deneubourg, J.L., Aron, S. and Pasteels, J.M. "How Trail Laying and Trail Following Can Solve Foraging Problems for Ant Colonies", in Behavioural Mechanisms of Food Selection, R.N.Hughes ed., NATO-ASI Series, vol. G 20, Berlin:Springer-Verlag, 1990.

[5] Horowitz, E. and Sahni, S., Fundamentals of Computer Algorithms, Computer Science Press, USA.

[6] Lawler, E.L., Lenstra, J.K., Rinnooy-Kan, A.H.G. and D.B.Shmoys eds., The Travelling Salesman Problem, New York:Wiley, 1985.

[7] Papadimitriou, CH. and Steiglitz, K., Combinatorial Optimization Algorithms and Complexity, Prentice-Hall Inc. USA.

[8] West, D.B., Introduction to Graph Theory, Prentice-Hall Inc. USA