International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970)
Volume-3 Number-4 Issue-13 December-2013

Network routing problem-A simulation environment using Intelligent technique

Vijayalaxmi[1], Chandrashekara S.Adiga[2], H.G.Joshi[3], Harish S.V[4]

Abstract

Ever since the internet became a necessity in today's technological world people have been looking for faster ways to connect one machine to another. However, when it comes to the internet as a whole and connecting the mass of people to their respective destinations, an efficient algorithm is more difficult to produce. One direction that researchers have gone to pursue this is to study the behaviour of ants for their techniques to find the shortest path between two points. In this paper, the Ant Colony Optimization Technique has been applied in different network models with different number of routers and structures to find the shortest path with optimum throughput. The performance measure taken here is the shortest path as well as time taken by the data packets from source to destination.

Keywords

Swarm Intelligence, Ant Colony Optimization, Network routing.

1. Introduction

The routing protocols play a very important role in selecting the relevant path for transferring the data from the source to the destination efficiently. There are already many accepted routing algorithms to find the shortest path and also to increase the throughput of the network. In this paper, we are proposing a routing algorithm which may be applied in large network structure with heavy load. The goal of every network routing algorithm is to direct the traffic from source to the destination maximizing the performance of the network [1-2].

Vijayalaxmi, Department of Electrical & Electronics, Manipal Institute of Technology, Manipal, India.

Chandrashekara S. Adiga, Department of Electrical & Electronics, Manipal Institute of Technology, Manipal, India.

H.G.Joshi, Department of Commerce, Manipal University, Manipal, India.

Harish S.V, Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal, India.

The performance measure that is taken into account is number of packets successfully reaching the destination in shortest path and minimum time form source to destination. Ant colony optimization (ACO) [2] is a metaheuristic inspired by the foraging behavior of ants [3]. In order to find the shortest path from the nest to a food source, ant colonies make use of positive feedback mechanism. They use a form of indirect communication called stigmergy [4]. This is based on the laying and detection of pheromone trails. In Ant Colony Optimization approach, a generic combinatorial optimization problem is encoded into a constrained shortest path problem. In this way the ants following the shorter path are expected to return earlier and hence increase the amount of pheromone deposit in its path at a faster rate than the ants following longer path.

ACO is basically inspired from the foraging behavior of the ants. These ants deposit pheromone on the ground in order to mark some favorable path that should be followed by other members of the colony. However, the pheromone is subjected to evaporation by a certain amount at a constant rate after a certain interval and therefore the paths visited by the ants frequently, are only kept as marked by the pheromone deposit, whereas the paths rarely visited by the ants are lost because of the lack of pheromone deposit on that path and as a result the new ants are intended to follow the frequently used paths only. Thus, all the ants starting their journey can learn from the information left by the previously visitor ants and are guided to follow the shorter path directed by the pheromone deposit.

In the subsequent sections, motivation behind the work, basic theory of Ant colony optimization and methodology used to tackle the network routing problem are explained.

2. Motivation

In any static routing algorithm packets are bound to follow a constant path. This causes following problems.

- Gives rise to the problem of congestion
- Packets may have to wait indefinitely.

These drawbacks can be very well managed using Ant Colony Optimization [3]. In case of ACO, the next node is selected dynamically and randomly, with the probability to choose the shortest path more. This increases the net throughput of the network when the number of packets in the network increases.

3. Ant Colony Optimization

Ant colony optimization is an iterative algorithm. At each iteration, a number of artificial ants are considered. Each of them builds a solution by walking from vertex to vertex on the graph with the constraint of not visiting any vertex that it has already visited in its walk. At each step of the solution construction, an ant selects the following vertex to be visited according to a stochastic mechanism that is biased by the pheromone: when in vertex i, the following vertex is selected stochastically among the previously unvisited ones (see Figure 1). In particular, if j has not been previously visited, it can be selected with a probability that is proportional to the pheromone associated with edge (i, j). At the end of an iteration, on the basis of the quality of the solutions constructed by the ants, the pheromone values are modified in order to bias ants in future iterations to construct solutions similar to the best ones previously constructed[4].

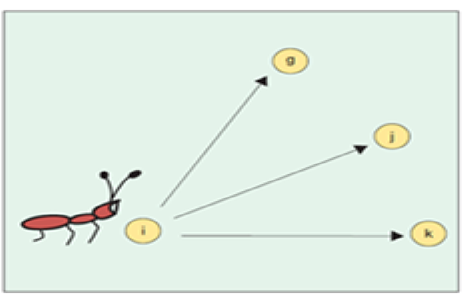

Fig 1: Ants' decision to take the shortest path

An important and interesting behaviour of ant colonies is their foraging behaviour, and, in particular, how ants can find the shortest paths between food sources and their nest. While walking from food sources to the nest and vice versa, ants deposit on the ground a substance called pheromone, forming in this way a pheromone trail. Ants can smell pheromone, and when choosing their way, they tend to choose, in probability, paths marked by strong pheromone concentrations. The pheromone trail allows the ants to find their way back to the food source (or to the nest). Also, it can be used by other ants to find the location of the food sources found by their nest mates [5]. The application of ACO is particularly interesting for i) NP- hard problems, which cannot be efficiently solved by more traditional algorithms. ii) Dynamic shortest path problems in which some properties of the problems change over time concurrently with the optimization process. Ant algorithms were inspired by the observation of real ant colonies. Ants are social insects, that is, insects that live in colonies and whose behaviour is directed more to the survival of the colony as a whole than to that of a single individual component of the colony. Social insects have captured the attention of many scientists because of the high structuration level their colonies can achieve, especially when compared to the relative simplicity of the colony's individuals. It has been shown experimentally by many researchers that this pheromone trail following behaviour can give rise, once employed by a colony of ants, to the emergence of shortest paths. [6-7] That is, when more paths are available from the nest to a food source, a colony of ants may be able to exploit the pheromone trails left by the individual ants to discover the shortest path from the nest to the food source and back. Ant Colony Optimization technique has different segments such as Ants generation & Activity, Pheromone Evaporation, daemon actions etc. [8].These segments needs to be designed based on the problem demand and nature. Rest of this paper talks about the pseudo code and algorithms which are needed to implement these techniques in network routing problems.

4. Methodology

ACO algorithms can be applied to the network routing problems to find the shortest path. In a network routing problem, a set of artificial ants (packets) are simulated from a source to the destination. The forward ants are selecting the next node randomly for the first time taking the information from the routing table and the ants who are successful in reaching the destination are updating the pheromone deposit at the edges visited by them by an amount (C/L), where 'L' is the total path length of the ant and C a constant value that is adjusted according to the experimental conditions to the optimum value [9]. The next set of ants can now learn from the pheromone deposit (feedback) left by the previously visited successful ants and will be guided to follow the shortest path. The probability of selecting a node j from node i is given by

$$p_{ij} = \frac{\tau_{ij}^{\alpha}\eta_{ij}^{\beta}}{\Sigma\Gamma_{ij}^{\alpha}\eta_{ij}^{\beta}}$$

(if a link exists between nodes i and j)
or

$$p_{ij} = 0$$

if there is no link between nodes i and j.
where, p_{ij} is the probability of selecting a node i from node j, τ_{ij} is the pheromone associated with the path joining node i to node j, $\eta_{ij} = \frac{1}{d_{ij}}$, where d_{ij} is the distance between the nodes i and j and α and β are parameters that controls the relative importance of the pheromone versus the heuristic information.
The main characteristic of the ACO is that, after each iteration, the pheromone values are updated by all the number of ants (packets) that have reached the destination successfully [8-9]. The pheromone value Γ_{ij} while travelling from node i to node j is updated as follows:

$$\Gamma_{ij} = (1-\rho)\tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$$ where ρ is the evaporation rate, m is the total number of successful ants (packets) and is the quantity of pheromone laid on edge (i,j) by packet k.

The pseudo code for the proposed simulation study is as shown in Fig 2 [10-11]. Here termination criteria is number of iterations. Iteration is said to be completed when all sent packets reach the destination. Maximum number of packets select shortest path, as the pheromone count in that particular path tends to be maximum.

```
procedure ACO_Meta_heuristic( )
while(termination_criterion_not_satisfied)
schedule_activities
ants_generation_and_activity( );
pheromone_evaporation( );
daemon_actions( );{optimal}
end schedule_activities
end while
end procedure
```

Fig. 2: Pseudo code

Algorithm:
The Ant Colony Optimization Meta heuristic
Set parameters, initialize pheromone trails
while termination condition not met do
ConstructAntSolutions
ApplyLocalSearch (optional)
UpdatePheromones
Endwhile

Based on this algorithm two approaches are proposed, simulated and presented in this paper. In the first method, the order of routers in which a data packet should travel in order to start from a source router, reaching to the destination router by visiting all rest of the routers each only once and travelling with shortest path distance for the same is determined. No packets are allowed to visit a router that is already visited by it in its journey. It means that is no packets are allowed to make a loop in its path. While selecting the next router, it is checking that the router has already been visited by it or not. It updates shortest distance taken with the newly obtained value. If the router is an already visited one, then that router is discarded and checked for the other available routers. At last the shortest route from the source to all the available routers is determined. Fig 3 shows a typical communication network considered for the analysis.

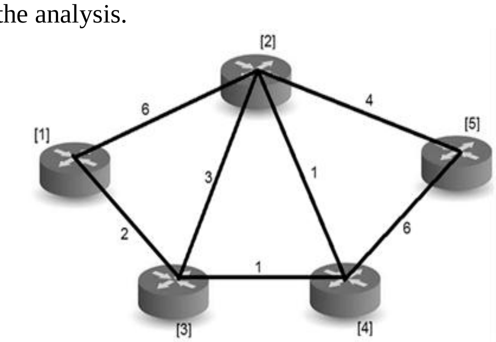

Fig. 3: Typical communication network

In an another approach, the packets are allowed to make loops in their paths and visit an already visited router provided that it will not visit only the last router visited by it[12-13]. In this method, after the completion of the journey of the successful packets, the paths of the packets are checked and the pheromone deposits of the router are updated. Here data packet does not depend on distance alone, travel time is equally important. Hence shorter distance with longer transmission time is less efficient than longer distance with short transmission time. Hence weight of the links represents time taken by data packets to travel. Here assumptions are slightly modified in order to get more throughput of the system. The next section summarizes the results obtained while simulating these two approaches.

5. Results

For the given inputs, the obtained results are tabulated in this section. C code has been developed for the algorithm and compiled using Turbo C++ compiler version 3.1. In the first method, the order of routers in which a data packet should travel in order to start from a source router, reaching to the destination router by visiting a particular router only once and travelling with shortest distance for the same is determined. Execution of the implemented code starts from the Main module. Other modules are called from this module and the final results are displayed. Table 1 gives the source and destination node as referred to Fig 3. When the code is executed the distance between each nodes are to be entered. It is as shown in Table 2.

Table 1: Inputs to ACO algorithm

Enter source	2
Enter Destination	1

Table 2: Inputs given to ACO algorithm (Distance)

SOURCE	DESTINATION	DISTANCE(Units)
One	Two	6
One	Three	2
Two	Three	3
Two	Four	1
Two	Five	4
Three	Four	1
Four	Five	6

This algorithm is generic in nature. This can be efficiently used to send the packets from any source node to any destination node. Apart from this, the number of nodes can also be increased depending on the problem. The algorithm proceeds as shown in Table 3. The output window and the obtained outputs are as shown in Fig 4 and Table 4 respectively.

From router 2 to router 1, the shortest path found is through router 4 and 3 as shown in Fig.5. The output produced has minimum circuit length. Another major observation is that only some routers have been traversed and that too only once. In the output, for each run the best path and its shortest execution are reported. (not shown here).

Table 3: Determination of shortest path in the algorithm

Present router	Possible routers	Distance	circuit length & path
2	1	6	-
	3	3	-
	4	1*	1,2-4
	5	4	-
4	3	1*	2,2-4-3
	5	6	-
3	1	2*	4,2-4-3-1

Fig 4: Output Window

Table 4: The Output (shortest distance)

Shortest distance taken is:		4
PATH IS:		
Two	Four	1
Four	Three	1
Three	One	2

It can be inferred from the Table 4 that the optimal path was found by this algorithm. Though different paths were present but data packets explored and converged to the optimal path

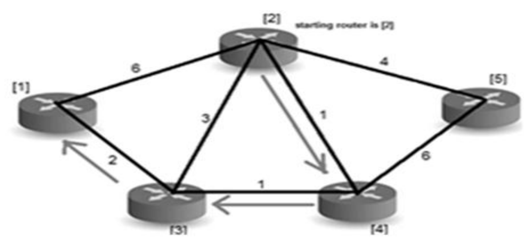

Fig 5: The shortest path taken by the data packet

The second approach emphasizes on minimal time.
In this approach, the order of routers in which a data packet should travel from a source router, reaching to

the destination router in minimum time is determined. The packets are allowed to make loops in their paths and visit an already visited router provided that it will not visit only the last router visited by it.

Table 5: Inputs (time)

SOURCE	DESTINATION	TIME(Units)
1	1	0
1	2	7
1	3	4
1	4	0
2	1	0
2	2	0
2	3	3
2	4	0
3	1	2
3	2	2
3	3	0
3	4	5
4	1	0
4	2	3
4	3	3
4	4	0

Execution of the implemented code starts from the Main module. Other modules are called from this module and the final results are displayed. The input to this algorithm assumes a prior knowledge of number of routers and time taken by data packets to travel from one router to another. The inputs are tabulated as shown in Table 5.

Table 6: The output (minimum time)

Source Router 1	Path	Time taken
Router 2	1-3-2	6
Router 3	1-3	4
Router 4	1-3-4	9

The obtained outputs are shown in Table 6.Only a few outputs are tabulated here. This algorithm was tested for different sources and destinations and works satisfactorily.

The output produced has minimum circuit travel time and the final selected path details. Another major observation is that only some routers have been traversed. In the output table, for each run the best path and its shortest execution are reported. It can be inferred from the Table 4, that the optimal path was found by this algorithm. Though different paths were present, data packets explored and converged to the optimal path. An effort is made to compare the obtained results with the Hamiltonian circuit.

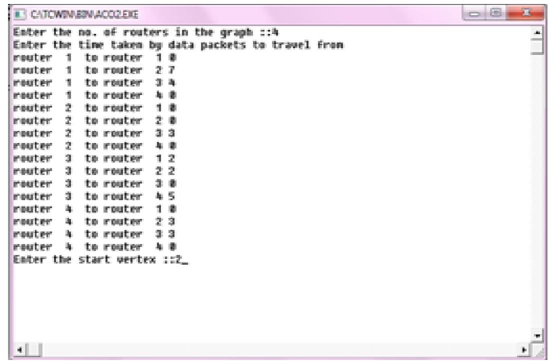

Fig 6: The Output window

Hamiltonian circuit consumes more time as it has to visit all the available routers to transmit data packet. The proposed algorithm is better because in this case a router can be visited more than once (but not the last visited router). Also, data packet does not depend on distance alone, travel time is equally important. Hence shorter distance with longer transmission time is less efficient than longer distance with short transmission time. Hence weight of the links represents time taken by data packets to travel.

6. Conclusions

In this simulation environment, the main objective was to determine the shortest path between any two routers. The shortest path is obtained by the simulation from the source router to the destination router. Each router is visited only once. It is observed from the output table that the routers connected via the edges with shortest distance are selected as the final solution or the best path. More the pheromone deposit, shortest the route is. This is depicted in Fig.7

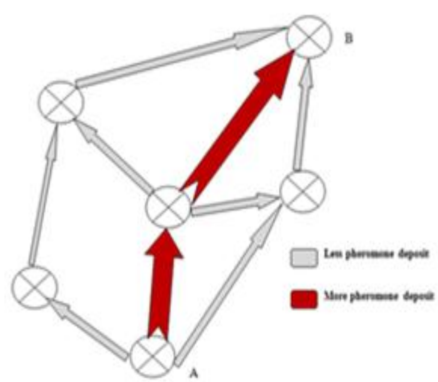

Fig.7: Shortest route using ACO

The algorithm was further modified by considering time into account. In this algorithm there was a scope to visit the routers more than once to reduce some longer paths. But in larger networks an ant (packet) may fall in infinite loop as small loops are allowed. This demerit can be taken as future scope of work. To avoid this situation after a certain interval of time if an ant (packet) has not reached the destination, the packet must be marked as unsuccessful and its journey must be forcefully stopped.

References

[1] David Corne, Marco Dorigo, Fred Glover, "New Ideas in Optimization" "Advanced Topics in Computer Science" Mcgraw-Hill Series, 1999.

[2] Marco Dorigo,Gianni Di Caro,Luca M.Gambardella, "Ant algorithms for discrete optimization", Artificial Life archive. Volume 5, Issue 2, 1999.

[3] Debasmita Mukherjee and Sriyankar Acharyya, "Ant Colony Optimization Technique Applied in Network Routing Problem", International Journal of Computer Applications (0975 - 8887), Volume 1 – No. 15, Pg no. 66-73, 2000.

[4] Marco Dorigo, Mauro Birattari, and Thomas St¨utzle,"Ant Colony Optimization", IEEE computational intelligence magazine, November 2006.

[5] M. Dorigo, V. Maniezzo & 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.

[6] Gianni Di Caro and Marco Dorigo, "An adaptive multi-agent routing algorithm inspired by ants behavior, IRIDIA – Université Libre de Bruxelles – Belgium.

[7] Gianni Di Caro and Marco Dorigo, "Ant Colonies for Adaptive Routing in Packet-Switched Communications Networks" Lecture Notes In Computer Science, 1998.

[8] Devika Subramanian, Peter Druschel, Johnny Chen.,"Ants and reinforcement learning: A case study in routing in dynamic networks" 1997, Proceedings of the Fifteenth International Joint Conf. on Arti Intelligence,1997.

[9] Marco Dorigo and Thomas Stutzle, "Ant Colony Optimization", MIT Press Edition, ISBN-81-203-2684-9.

[10] Beckers R., Deneubourg J.L. and S. Goss, "Trails and U-turns in the selection of the shortest path by the ant Lasius niger", Journal of theoretical biology, 159, 397-415,1992.

[11] Verstraete, Vincent, Matthias Strobbe, Erik Van Breusegem, Jan Coppens, Mario Pickavet, and Piet Demeester. "AntNet: ACO routing algorithm in practice." In Proceedings of the 8e INFORMS Telecommunications Conference. 2006.

[12] Rutger Claes,.Tom Holvoet.,"Ant Colony Optimization applied to Route Planning using Link Travel Time", IEEE International Parallel & Distributed Processing Symposium, 2011.

[13] Goss, Simon, Serge Aron, Jean-Louis Deneubourg, and Jacques Marie Pasteels. "Self-organized shortcuts in the Argentine ant." Naturwissenschaften 76, no. 12 (1989): 579-581.

Mrs.Vijayalaxmi is working as Assistant Professor in the department of Electrical and Electronics Engineering at MIT Manipal. Her research area includes Computational Intelligence a nd Digital system Design. She is the life member of ISTE,GSTF and SSI. She has around 8 research publications.

Chandrashekara Adiga S obtained his Ph.D from Manipal University in the year 2006. Currently he is serving as Professor in the department of Electrical and Electronics Engineering at MIT Manipal. His research areas include Illumination Technology, Power Systems, Artificial Intelligence. He is the life member of various bodies like ISLE, ISTE,IE and SSI. He has more than 20 research publications in various International Conferences and Journals.

H.G.Joshi is Professor in the Department of Commerce, Manipal University, Manipal. He is the Chair Person of Dr. TMA Pai Chair for Social Entrepreneurship, Manipal University. He has 12 research publications. His range of expertise includes enterprise setting and entrepreneurship development

Harish S.V obtained his Ph.D from N.I.T.K Suratkal in the year 2011. Currently he is serving as Professor in the department of Computer Science and Engineering at MIT Manipal.He is the life member of various bodies like ISTE,SSI and CSI. He has 10 research publications in various International Conferences and Journals.