

A Novel Ant Colony Optimization Based Intelligent Routing Algorithm

**Pardeep Kumar^a, Saumya Mahajan^b, Pallavi Srivastava^b,
Karishma Sharma^b, Pradeep Kaur^b**

*^aDepartment of CSE, Assistant Professor,
Jaypee University of Information Technology, Solan, 173234, India*

*^bDepartment of CSE, B. Tech Students,
Jaypee University of Information Technology, Solan, 173234, India*

Abstract

Distance Vector Routing (DVR) has successfully tackled multiple routing problems, but it still suffers from certain drawbacks; slow convergence, limited scalability and routing loops. In this paper, we propose an adaptive routing algorithm which is loosely based on Original AntNet algorithm. From the paper, it shall be clear that our proposed algorithm exhibits better performance than the traditional routing algorithm as it will solve the problems encountered in distance vector routing like premature convergence, count to infinity problem, slow updating problem.

Keywords : Ant Colony Optimisation, Routing Algorithm, Swarm Intelligence.

Introduction

Due to wide advancement in technology and need for communication, there is a great demand for inter-network communication. The control and management of this communication takes-off with interaction among networks which is performed with the help of Routing. Routing serves as a basis for communication among different users in large computer networks. R&D work on Social insect and their behavior in various situations has provided computer engineers and scientists with effective methods for designing optimization algorithms. This serves as an inspiration for our proposed algorithm. Routing algorithm answers the question of how packets are transferred from one point to another point while simultaneously addressing and tackling features like simplicity, correctness, robustness, stability, fairness and optimality. Network information and management systems, communication and

telecommunication rely on a combination of routing algorithms, strategies and protocols to make sure the data sent is successfully transmitted and received while simultaneously choosing the best path, maximizing the efficiency and tackling the traffic problem. The growing size of networks and increasing demand on packet switched networks has created a demand of a new algorithm to tackle the current market needs. Routing is the task of directing units of data from their source to destination while optimizing various criteria. Optimization criteria can be the variability in delay undergone by data packets etc. An adaptive routing algorithm modifies its routing solution dynamically, in order to account for changes in the network, such as variations in the data traffic load or in the network topology.

Routing protocols in current scenario put on in widespread use (e. g. OSPF, RIP, EIGRP and BGP) are implemented through the information stored in routing tables available with each node in the network. Variations in a network are the result of change in traffic patterns, availability of new network resources or nodes and of removal, or failure of network resources. Adaptive routing algorithms for wired networks are traditionally developed using a *top-down approach*, where a well-known exact centralized algorithm for route calculation is adjusted to work in an adaptive, decentralized way [1]. A class of link-state routing protocol which includes widely applied OSPF protocol is an example of top-down approach. When each node locally monitors the current status of the links to all its neighbours it is said to be link-state routing. Periodically, it constructs a message packet which contains the local view, and forwarded it to all the other nodes in the network. A node receiving this message joins it with similar messages received from all other nodes, to obtain a complete view of the network. This complete view is then locally used to derive a weighted graph representation of the entire network and to calculate all the routes using Dijkstra's shortest path algorithm. This way, each node can perform routing based on a complete overview of the system, like a centralized computer system. Another example of the top-down approach is the class of *distance-vector* routing protocols, to which the Internet protocol RIP belongs. Distance-vector routing algorithm is based on the distributed bellman-ford algorithm [2]. This algorithm implements a distributed version of dynamic programming, which is a general solution method for optimization problems.

The basic idea behind dynamic programming is to recursively split a problem into sub problems, and to use the solutions to these sub problems to construct an optimal solution for the original problem [3]. Distance-vector routing implements dynamic programming in a distributed and asynchronous way. Nodes incrementally calculate estimates for the cost of the route to each possible destination based on estimates provided by neighbouring nodes. Periodically, each node sends its estimates out to its neighbours, so that they can update their own estimates again [4]. Each node periodically transmits its routing information to its neighbours and this information is used to maintain routing information for each possible destination node in the network at all times, this method of transmitting data is called dynamic or proactive approach. Another approach is of Reactive algorithms for adaptive routing as in this the routing information is gathered and maintained between pair of nodes that are the source and destination of nodes. In these algorithms, the search for routing information is

initiated by source nodes that have packets to send to various unknown destinations. Each node in the network acts independently, and the routing result is obtained from this combination of individual results. This approach is bottom-up design approach. This design approach would be our fundamental step for implementation of routing algorithm.

1. 1 Swarm Intelligence

The application of the SI paradigm which concerns here is the bottom-up design of system that showcases the organizational characteristics of original natural system. This organizational feature of natural system serves as a stem of inspiration. Swarm Intelligence is the discipline that deals with natural and artificial systems which is composed of individuals that coordinate using self-organization and hence eventually leading to decentralized control. In particular, the discipline focuses on the collective behaviours that result from the local interactions of the individuals with each other and with their environment [5]. Many systems in nature have been studied and analysed for many years some of which are like colonies of ants, schools of fish, flocks of birds, colonies of termites etc. Swarm intelligence can be classified according to different criteria.

Our Concern of study shall be to combine both the aspects of swarm intelligence i. e. natural and engineering. The analysis of the natural behaviour can be defined as the foraging behaviour of ants combined with the engineering aspect of swarm based data analysis, to be discussed in the later section.

Properties of a Swarm Intelligence System which make it suitable to work for the routing algorithms can be enumerated as : it is composed of many individuals; the individuals are relatively homogeneous (i. e., they are either all identical or they belong to a few typologies). The interactions among the individuals are based on simple behavioural rules that exploit only local information that the individuals exchange directly or via the environment (stigmergy). The overall behaviour of the system results from the interactions of individuals with each other and with their environment.

The characterizing property of a swarm intelligence system is its ability to act in a coordinated way without the presence of a coordinator or of an external controller. Many examples can be observed in nature of swarms that perform some collective behaviour without any individual controlling the group, or being aware of the overall group behaviour.

1. 2 Application of Ant Colony Optimization (ACO) to adaptive routing algorithm

Swarm Intelligence can make a significant impact on efficiency of routing algorithm. The SI way of proceeding emphasizes a bottom-up modular design that relies on self-organization, redundancy and stochasticity in order to obtain the desired global response of the system. SI technique emphasizes on adding “intelligence” to the routers without affecting packets. A social insect metaphor provides a ground in which routing table and data packet exists but, in addition to that, new control packets are introduced that interact to keep the contents of routing table up to date. Our main focus will be on the Ant Colony Optimization (ACO) in the domain of swarm

intelligence. ACO takes inspiration from the pheromone-mediated ability of ant colonies to find the shortest paths between their nest and sources of food to define a meta-heuristic for combinatorial optimization based on the use of ant-like agents and stigmergic communication of artificial pheromone information.

The rest of the paper is organized as follows : Section 2 provides the view of work done so far. Section 3 describes the existing ant colony optimization approach. Section 4 discusses our proposed approach and intelligent Distance Vector Routing. Section 5 provides comparative analysis of existing approach and our proposed approach. The chapter is finally concluded in Section 6.

Work done till date

The first swarm-based approaches to network management were proposed in 1996 by Schoonderwoerd et al [6] and in 1998 by Di Caro and Dorigo [7]. Schoonderwoerd et al. proposed *Ant-based Control (ABC)*, an algorithm for routing and load balancing in circuit-switched networks; Di Caro and Dorigo proposed AntNet, an algorithm for routing in packet-switched networks. While ABC was a proof-of-concept, AntNet, which is an ACO algorithm, was compared to many state-of-the-art algorithms and its performance was found to be competitive especially in situation of highly dynamic and stochastic data traffic as can be observed in Internet-like networks [8]. First we will look at the original AntNet algorithm then we'll work on the proposal to implement it on the routing algorithm which is the main problem to be faced while working with ant colony optimization.

Existing Ant Colony Optimization

An Ant Colony Optimization algorithm (ACO) is essentially a system based on agents which simulate the natural behavior of ants, including mechanisms of cooperation and adaptation. In the use of this kind of system as a new meta-heuristic was proposed in order to solve combinatorial optimization problems. This new meta-heuristic has been shown to be both robust and versatile – in the sense that it has been successfully applied to a range of different combinatorial optimization problems [9]. Stigmergy is the inspiration of ant colony optimization arose. Stigmergy is the process of locally sensing and modifying the environment [10]. Stigmergy is at the core of most of all the amazing collective behaviors exhibited by the ant colonies.

The original AntNet algorithm, Ants are only capable of simple stochastic decisions influenced by the availability of previously laid stigmergic trails. The chemical denoting a stigmergic trail is subject to decay over time and reinforcement proportional to the number of ants taking the same path. Trail building is naturally a bidirectional process, ants need to reach the food (destination) and make a successful return path, in order to significantly reinforce a stigmergic trail. Moreover, the faster the route, the earlier the trail is reinforced. An ant on encountering multiple stigmergic trails will probabilistically choose the route with the greatest stigmergic reinforcement. Naturally, this will correspond to the 'fastest' route to the food (destination). The probabilistic nature of the decision, however, means that ants are

still able to investigate routes with a lower stigmergic trail.

Pseudo-code of AntNet algorithm [11]

```

Procedure ACO_meta-heuristic ()
  While (termination_criteria_not_satisfied)
    schedule_activities
      ants_generation_and_activity ();
      pheromone_evaporation ();
    end schedule_activities
  end while
end procedure
procedure ants_generation_and_activity ()
  while (available_resources)
    schedule_the_creation_of_new_ant ();
    new_active_ant ();
  end while
end procedure
procedure new_active_ant ()
  initialize_ant ();
  M = update_ant_memory ();
  while (current_state != target_state)

  A = read_local_ant_routing_table ();
  P = compute_transition_probabilities (A, M, d );
  next_state = apply_ant_decision_policy ( P, d );
  next_state = apply_ant_decision_policy (P, d);
  move_to_next_state (next_state);
  if (online_step_by_step_pheromone_update)
    deposit_pheromone_on_the_visited_arc ();
    update_ant_routing_table ();
  end if
  M = update_internal_state ();
end while
if (online_delayed_pheromone_update)
  for each visited_arc (belong to) Solution do
    deposit_pheromone_on_the_visited_arc ();
    update_ant_routing_table ();
  end for each
end if
die ();
end procedure

```

In AntNet, it is assumed that routing tables, T_k , exist at each node, k , in which a routing decision is made. Tables consist of L rows, one row for each neighbouring node/link. As far as a normal data packet is concerned, if the destination, d , from current node, k , is a neighbour then the routing is still a stochastic decision. In all

other cases, a route is selected based on the neighbour node probabilities.

The mathematics involved :

The transition rule which includes the probability.

The state transition rule used by ant system, called a random-proportional rule, is given by Eq. (1), which gives the probability with which ant k in node r chooses to move to the node s .

$T(r, s)$ is the amount of pheromone currently on the path that goes directly from node r to node s .

$H(r, s)$ is the heuristic value of that link- this is chosen to be $1/\text{distance}(r, s)$ -i. e. shorter the distance higher will be the heuristic value.

$P_k(r, s)$ is the probability that the ant k will chose the link from node r to nodes.

The rule is :

$$p_k(r, s) = \frac{T(r,s) * H(r,s)^\beta}{\sum_{c \in \text{allallowednode}} T(r,c) * H(r,s)^\beta} \quad (1)$$

We simulate the behaviour of real ant on the artificial ant by taking the probabilistic routing decision according to the above rule, we model the artificial ants by assigning it to preferentially move in the direction of high pheromone intensity. This method of pheromone trail and intensity give rise to stochasticity that provides adaptability. The heuristic value allows for dynamic adaptations based on traffic variations.

In ant system, the updating of pheromone trail is done with the following rule :

$$\tau(r, s) \leftarrow (1 - \alpha) * \tau(r, s) + \sum_{k=1}^m \Delta \tau_k (r, s) \quad (2)$$

While travelling to d , the forward ant records the delays it experiences. Once it reaches d , it becomes a *backward ant*, which returns to the source node s tracing back the path followed by the forward ant. At each node i along this path, the backward ant updates the entries in i 's tables for destination d . First, the estimates in the traffic statistics table are updated using the trip time t experienced by the ant. Then the pheromone entry d_{ij} is updated, with j being the neighbour over which the backward ant arrived in i . To this end, a reinforcement value r is first calculated, which reflects how good t is compared to the information about previous trip times that is stored in the statistics table. Then, d_{ij} is updated with r using a moving average. The same process is repeated for all visited nodes in the path from i to d .

Proposed Approach

The given map contains 12 paths on which the ant colony will move towards its food source.

1-3-8, 2-3-8, 1-4-5-7-8, 2-4-5-7-8, 1-4-5-7-9, 2-4-5-7-9, 1-4-6-7-9, 2-4-6-7-9, 1-4-6-7-9, 2-4-6-7-9, 2-3-9, 1-3-9

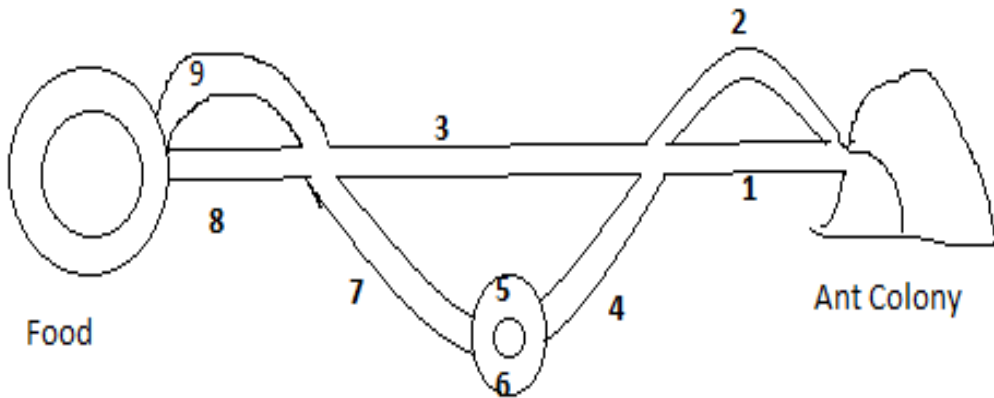


Fig 1 an Ant Colony Network

The shortest path out of the above 12 paths is 1-3-8. Though some ants will move through other paths but the pheromone trail evaporation on 1-3-8 path would be lower in rate as compared to other paths and hence the ant follow-rate on this path would be maximum. Since being the shortest path, the ants travelling on this path will return earlier and hence will make deep impression of pheromone trail faster and other ants will follow this shortest path with maximum pheromone amount.

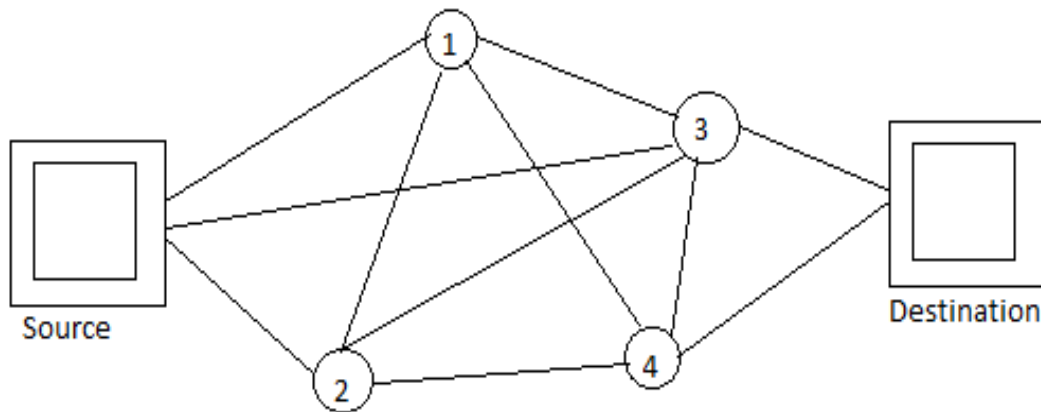


Fig 2 various paths a packet can travel from source to destination

Any data travelling from its source to reach its destination would need to travel a number of intermediary nodes (these nodes can be servers or any service units). This can be seen as being in similar fashion like ants travelling from their colony to food source. Our foremost priority here is to formulate a technique in a manner such that the natural phenomenon of trail (stigmergy) can be implemented artificially for our purpose. What we would follow for our data packets would be a proactive model. In this model, the data packet would not be a function of conditioning and conditions but

rather it would be a product of its choice, decision or self-awareness based on our implementation method of pheromone trail.

To implement pheromone trail, we propose the “method of count”. Every node on its routing table will contain an additional column which is called “count”. This count column would represent the number of packets passing through that node, which would increment with every packet passing through that node and going to the same destination as registered by the previous node. Another packet passing through the same node but not having same destination as other would have a different count identifier. Any data packet passing a particular node would leave a unique number such that the same packet after reaching its destination would return an acknowledgement permitted by the destination; this acknowledgement would be identified by the same unique number as left by its parent data packet. The complexity involved in this technique is that more the number of different destinations packet passing through a particular node more will be the count identifiers.

Proposed Pseudo code for Ant Colony Optimization

```

initialize all nodes to initial pheromone level (count)  $\tau_0$ ;
place each ant on any random node;
for each iteration do :
do while each ant has not completed its tour :
for each ant do :
move ant to next node by the probability function;
increment the counter for each node;
end;
end;
end;
for each iteration;
do while ant has not completed its tour;
for each ant do;
move ant to previous node by probability function;
increment counter;
end;
end;
for each ant with a complete tour do :
initialize timer to 0;
evaporate pheromones;
apply pheromone update;
if (ant k's tour is shorter than the global solution)
update global solution to ant k's tour
decrement counter;
end;
end;

```

Pseudo Code for proposed distance vector routing protocol on our proposed algorithm

- R[d]. link is the outgoing link that the router uses to forward packets towards destination d.

- R[d]. pheromone is the sum of the metrics of the links that compose the shortest path to reach destination d.
- R[d]. time is the timestamp of the last distance vector containing destination d.

Every N seconds :

```

v =Vector ()
for each iteration do #do while each packet has not completed its tour;
{ for each destination=d in R[] {
  v. add (Pair (d, R[d]. pheromone)); } }
for each interface {
Send (v, interface) # send vector v on this interface
  move packet to next node by probability function; #using Eq (1)
increment counter for R[d];}
Received (Vector V[], link l)
{ # received vector from link l for each destination=d in V[]
{ if not (d is in R[])
{ # new route R[d]. pheromone=V[d]. pheromone + l. pheromone;
R[d]. link=l;
  R[d]. time=now;
} else { if ( ( V[d]. pheromone+l. pheromone) > R[d]. pheromone) or ( R[d]. link ==
l) )
{ # Better route or change to current route R[d]. pheromone=V[d]. pheromone+l.
pheromone;
  R[d]. link=l; #apply pheromone update usingEq (2)
  R[d]. time=now; } }
if packet reaches its destination {
send (ack); }
for each packet with complete tour
{ intialise timer to 0;
evaporate pheromones; #before the timer expires
decrement counter;
}

```

5. Comparison

Distance vector routing protocols tend to suffer from a number of significant shortcomings. The disadvantages tend to magnify significantly as the internetwork grows. First, routing tables in a large internetwork can be correspondingly large, and they can consume significant bandwidth in the exchange process. Given that most distance vector routing protocols generate periodic updates at intervals of 10 to 90 seconds, this can result in the routing protocol consuming a significant amount of bandwidth [12].

Table-1 Comparative Analysis of Distance Vector Routing Protocol and our Proposed Algorithm for Distance Vector Routing

Problem Type	Distance Vector Routing	Proposed Algorithm
Type	This algorithm is based on destination vector stored in the routing table i. e. it is destination based.	It is a packet based algorithm. In which pro-activity is the main property showcased by each packet.
Stochasticity	The stochasticity property is not an option in distance vector routing as the shortest path computed by algorithm is the only path packets can choose.	Path to be taken by the packet is completely random which is by virtue of stochasticity property, hence there is possibility of generation of new solutions.
Input	The solution is obtained by interaction of routing table at routers which contains the destination address and distance path (shortest path).	The solution is obtained by interaction of many packets while finding the destination i. e. a multiple –interaction property.
Aim	It works to obtain a single shortest path instead of multiple path hence does not show autonomous property.	It supports multiple paths so load balancing can be achieved along with the shortest path.
Convergence Problem	Shows premature convergence.	Premature convergence is not a problem as it makes use of distributed computation.

Our proposed algorithm exhibits Inherent parallelism as ants work autonomously in finding the shortest path to reach their destination. It is based on a pro-active approach in which each ant reaches its destination independently. In our algorithm, Positive Feedback results in effective and quick discovery of better solutions as the ant traverses the path from source to destination (vice versa) on the basis of probability function and pheromone trail. It overcomes the disadvantage of Distance Vector Routing by overcoming premature convergence through distributed computation. Also it can to some extent overcome the count to infinity problem.

Many different simulators are available like NS2, NS3, OMNet++, QualNet, Opnet etc. Here we will be using OMNet++ as our simulator tool. OMNeT++ is an object-oriented modular discrete event network simulation framework [14]. An OMNeT++ model consists of modules that communicate with message passing. The active modules are termed simple modules; they are written in C++, using the simulation class library. Simple modules can be grouped into compound modules and so forth; the number of hierarchy levels is unlimited. We would be judging our network performance based on the proposed algorithm using specific parameters which will determine the effectiveness of a network model. These performance metrics criteria are : Throughput : the ratio of packets received by the destination to

that produced by the source; Average delay : route buffering delays, queuing time delay, retransmission need and delay, propagation and transfer times; No. of packets dropped : no. of packets unable to reach their destination; Routing check packets / overhead : sent for route discovery or maintenance.

6. Conclusion

A good algorithm would be the one which shows the first parameter i. e. throughput as high and other parameters with a lesser value. A comparison of reactive and proactive model also needs to be analysed.

In this paper, we studied the performance of distance vector routing protocol over defined communication networks. The drawbacks of distance vector routing have already been explained above. We use our proposed algorithm which is loosely based on original AntNet algorithm to overcome these shortcomings and subsequently optimize the routing protocol using its characteristics of stochasticity and stigmergy. Hence, our proposed algorithm is better than distance vector routing protocol in terms of throughput and performance. Concerning the future impact to the network, scalability cannot be overlooked. We also need to consider the fact that increase in packet size effect the performance of the network. For our future work, we will work on Congestion control like error control and flow control.

References

- [1] "Principles and applications of swarm intelligence for adaptive routing in telecommunications networks", Frederick Ducatelle, Gianni A. Di Caro, Luca M. Gambardella, "DalleMolle" Institute for Artificial Intelligence Studies (IDSIA)
- [2] "On a routing problem. *Quarterly of Applied Mathematics*", 16 (1) : 87–90, 1958, R. Bellman
- [3] Principles and applications of swarm intelligence for adaptive routing in telecommunications networks Frederick Ducatelle, Gianni A. Di Caro, Luca M. Gambardella
- [4] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence : From Natural to Artificial Systems*. Oxford University Press, New York, NY, 1999.
- [5] "The Ant Colony based routing algorithm for MANETs", MesutGunes, UdoSorges, ImedBouazizi
- [6] "AntNet : Distributed stigmergic control communications networks", DiCaro, Dorigo
- [7] [http : //iridia. ulb. ac. be/mbiro/paperi/DorBir2007sp. pdf](http://iridia.ulb.ac.be/mbiro/paperi/DorBir2007sp.pdf)
- [8] M. Dorigo, G. Di Caro and L. M. Gambardella, "Ant algorithms for discrete optimization," *Artificial Life*, vol. 5, no. 2, pp. 137-172, 1999.
- [9] M. Dorigo, E. Bonabeau, and G. Theraulaz. Ant algorithms and stigmergy. *Future Generation Computer Systems*, 16 (8) : 851–871, 2000.

- [10] Ant Colony Optimization : A New Meta-Heuristic Marco Dorigo IRIDIA Universitt Libre de Bruxelles Gianni Di Caro IFUDIA Universitt Libre de Bruxelles
- [11] “Computation of pheromone values in AntNet algorithm”, Anuj K. Gupta, Harsh Sadarwati, Anil K. Verma
- [12] Yih-Chun Hu, David B. Johnson, Adrian Perrig, “SEAD : secure efficient distance vector routing for mobile wireless ad hoc networks”, Ad Hoc Networks 1 (2003) 175–192, elsevier.
- [13] Comparison and Performance Analysis of AntNet and Distance Vector Routing Protocol in Telecommunication Networks Abdusy Syarif Harris Simaremare Pascal Lorenz
- [14] [http : //www. omnetpp. org/doc/omnetpp/Manual. pdf](http://www.omnetpp.org/doc/omnetpp/Manual.pdf)