Multiple Feasible Paths in Ant Colony Algorithm for mobile Adhoc Networks with Minimum Overhead

Jitendra Prithviraj[1] and Jitendra Prithviraj[2]

[1] SRIT

Received: 10 January 2011 Accepted: 3 February 2011 Published: 15 February 2011

Abstract

Mobile ad-hoc networks are infrastructure-less networks consisting of wireless, possibly mobile nodes which are organized in peer-to-peer and autonomous fashion. The highly dynamic topology, limited bandwidth availability and energy constraints make the routing problem a challenging one. Ant colony optimization (ACO) is a population based meta-heuristic for combinatorial optimization problems such as communication network routing problem. In real life, ants drop some kind of chemical substances to mark the path that they used. Then on their way, back they choose the path with the highest pheromones which becomes the shortest path. But Ant net Algorithms may cause the network congestion and stagnation. Here, multiple optimal paths are proposed with negligible overhead in spite of single optimal path in Ant net routing algorithm, so that the problem of stagnation can be rectified. This paper proposes an improved Multiple Feasible Paths in Ant Colony Algorithm for mobile Ad-hoc Networks with Minimum Overhead.

***Index terms*—**

1 INTRODUCTION

oday with the fast growth of Internet, everybody wants to get connected to the internet. Millions of people use the Internet for daily business all over the world. Today the Internet has become very large, complex and dynamic. Failures and challenges occur at every step because of traffic flow from one part of network to another part. Routing is the process of selecting paths in a network to send traffic. Routing is an important aspect of network communication, which affects the performance of any network, since other characteristics of the network like throughput; reliability and congestion depend directly on it. In packet switching networks when a packet travels from a source to destination, it has to pass through a number of networks with varying characteristics. So an ideal routing algorithm is one which is able to deliver the packet to its destination with minimum amount of delay. It must be adaptive and intelligent enough to make the decisions. The growing size and increasing demands of Internet impelled the study of more powerful routing algorithms, which can optimize the flow of traffic.

The routing algorithms currently in use (e.g. OSPF, RIP, DSR,ADODV and BGP) are not sufficient to tackle the increasing complexity of such networks. They are not adaptive, intelligent and fault intolerant. The routing tables in them are updated by exchanging routing information between the routers. Different routing protocols use different approaches to exchange the routing information. There are mainly two approaches for routing algorithms, distance vector algorithms and linkstate algorithms. Distance vector algorithms use the Bellman algorithm. This approach assigns a number, the cost, to each of the links between each node in the network. In link-state algorithms for example, Open Shortest Path First (OSPF), the routers exchange linkstate information by flooding the link state packets. The link state updates are generated only when the link status changes. Once a node has obtained topology information of the entire network, Dijkstra's algorithm is generally used to compute the shortest path. The two main performance metrics that are affected by the routing algorithm are throughput and average packet delay. Nodes and links can fail, and congestion can arise in some areas. Thus, the routing algorithm needs to modify its routes, redirecting traffic and updating databases very quickly and adaptively.

In recent years, a new kind of routing protocols influenced by software agents called Ant Based routing is developed. S. Appleby and S. Steward were the first ones to introduce the concept of software agents used for control in telecommunication networks .The research process continued and it was applied to connection oriented networks [3]. Ant based routing was then applied to packet based connection less systems [4].This agent based approach was further researched and was modified for adaptive routing [5].Swarm intelligence provides a promising alternative to traditional routing algorithms by utilizing mobile software agents for network management. Although, an ant [1], [2] is a simple and unsophisticated creature, collectively a colony of ants can perform useful tasks such as building nests, and foraging (searching for food) [1], [2], [3].What is interesting is that ants are able to discover the shortest path to a food source and to share that information with other ants through stigmergy [1]- [5]. Stigmergy is a form of indirect communication used T by ants in nature to coordinate their problemsolving activities.

Ants achieve stigmergic communication by laying a chemical substance called pheromone.

2 II. ANT NET COLONY OPTIMIZATION

ACO [2] is a met heuristic in which a colony of artificial ants cooperates in finding good solutions to discrete optimization problems. Each ant of the colony exploits the problem graph to search for optimal solutions. An 'artificial ant', unlike natural counterparts, has a memory in which it can store information about the path it follows. Every ant has a start state and one or more terminating conditions. The next move is selected by a probabilistic decision rule that is a function of locally available pheromone trails, heuristic values as well as the ant's memory. Ant can update the pheromone trail associated with the link it follows. Once it has built a solution, it can retrace the same path backward and update the pheromone trails. ACO algorithm is interplay of three procedures as described in [2].

1. Construct ant solutions: This procedure manages a colony of ants that concurrently and asynchronously visit adjacent states of the considered problem by moving through neighboring nodes of the solution space of the problem's construction graph. 2. Update pheromones: It is the process by which pheromone trails are modified. The trail value can either increase, as ants deposit pheromone on the components or connections they use, or decrease, due to pheromone evaporation. Net increase/decrease in pheromone value at a given location on trail is determined by difference of deposition and evaporation. 3. Daemon actions: This procedure is used to implement centralized actions which cannot be performed by single ants.

3 III. ANT NET DATA STRUCTURES

AntNet is an ACO algorithm for data network routing proposed by Gianni Di Caro and Marco Dorigo [2]. Mobile agents (artificial ants) act concurrently and independently, and communicate in an indirect way (stigmergically), through the pheromones they read and write locally on the nodes. Each network node k stores two data structures:

4 1) Routing table Tk

For each possible destination d and for each neighbor node n, Tk stores a probability value Pnd expressing the goodness of choosing n as next node when the destination node is d:? P nd = 1, d ? [1, N], N k = neighbors (k) n?N k

Probability value P nd represents the pheromone concentration along the link from node k to neighbor node n for destination node d.

5 IV. ANT NET ROUTING ALGORITHM

AntNet algorithm, as proposed by Di Caro and Dorigo, is as follows 1. At regular intervals Î?”t from every network node s, a forward ant F s?d is launched toward a destination d to discover a feasible, low-cost path to that node and to investigate the load status of the network along the path. If f sd is a measure (in bits or in number of packets) of the data flow s?d, then the probability of creating at node s a forward ant with node d as destination isP sd = (f sd) N ? f si i=1

While traveling toward their destination nodes, the forward ants keep memory of their paths and of the traffic conditions found. The identifier of every visited node i and the time elapsed since the launching time to arrive at this i-th node are stored in a memory stack.

2. At each node i, each forward ant headed toward a destination d selects the node j to move to, with a probability P ijd computed as normalized sum of the pheromone ? ijd with a heuristic value ij taking into account the length of the j-th link queue of the current node i :P ijd = ? ijd + ?? ij 1+ ? (| N i | -1)

The heuristic value ? ij is a normalized value function of the length q ij of the queue on the link connecting the node i with its neighbor j:? ij = 1-(q ij) |Ni| ? q i1 i=1

The value of ? weighs the importance of the heuristic value with respect to the pheromone values.

3. If a cycle is detected, the cycle's nodes are removed and all the memory about them is deleted. When an ant reaches a node that is already in its memory, a cycle is detected and all the nodes until this recurrent node are deleted from the ant's memory.

4. When the destination node d is reached, the agent F s?d generates backward ant B d?s , transfers to it all of its memory, and is deleted.

5. The backward ant takes the same path as that of its corresponding forward ant, but in the opposite direction.

6. Arriving at a node i coming from a neighbor node, the backward ant updates the local model of the traffic Mi and the pheromone matrix Ti, for all the entries corresponding to the destination node d.

6 1) Update traffic model M i

The estimated mean and variance are updated as follows:µ id ? µ id + ? (o i?d -µid) ? 2 id ? ? 2 id + {? (o i?d -µid) 2 -? 2 id}

where o i?d is the observed ant's trip time from node i to destination d. The factor ? weighs the number of most recent samples that will really affect the average

7 V. LIMITATIONS

Although Experiments of AntNet have shown very promising results, antnet has outperformed under different experimental conditions with respect to other dynamic routing algorithms e.g. RIP, OSPF. Still there are some problems with this adaptive algorithm. One of the major problems is that the network gets trapped because a node prefers a link with higher probability to a destination when choosing an outgoing link say no to send a packet. If link no keeps good condition for a long time, its probability to that destination will be very high. Such a condition may cause congestion of n o ; it also reduces probability of selecting other paths. Hence the node will stuck to this outgoing link and loose its adaptive ability. This is called the problem of "stagnation". Stagnation is reached when a node reaches its convergence. Stagnation is a very critical because 1. n o may lose its optimality if it gets congested 2. If the network gets fails at any time then the most preferred path no may become unavailable 3. Other non-optimal paths may become optimal due to changes in network topology Many researchers have tried to provide the solution for the same. These methods are

? Evaporation ? Aging ? Limiting and Smoothing Pheromone But all the methods are very complex and needed extra overhead. Here, multiple optimal paths are proposed so that the congestion and stagnation over single optimal path can be avoided.

8 VI. PROPOSED LOGIC

According to original Antnet Algorithm ,Routing

9 VII. A NETWORK SIMULATOR TOOL(Ns2)

We chose ns-2 for the implementation of AntNet because it is an event driven simulator which enables the generation and forwarding of packets according to the route logic. Also trace files provide nice details of packet flow that enables to check the correctness and accuracy of algorithm.

10 VIII. ANT NET IMPLEMENTATION

We could successfully extend network simulator ns-2 to implement AntNet simulation. Ant packets of two types representing forward ants and backward ants flow through the network according to the algorithm and keep a memory of the path traversed to discover optimal solution. We defined a routing agent called Antnet agent which is responsible to implement route logic and perform the protocols functionality by processing ant packets according to the flow of the algorithm. Routing agent is implemented through class Antnet derived from class Agent. We defined Tcl hooks that enable simulation of AntNet routing algorithm through Tcl script. Hence, it can be simulated for varying simulation time and time interval at which ants are launched.

11 1) Ant Packets

We have implemented artificial ants by defining a new packet type in ns-2. Besides information on packet source, destination, sequence number and length, the packet header also has a field that identifies it as forward ant or backward ant. A local data structure represents the memory of a packet, which stores node addresses and trip time to corresponding nodes that the packet traversed.

12 2) Antnet Routing Agent

We have defined class Antnet inherited from class Agent. This implements the Antnet routing agent to which a node can be associated. The routing table and local traffic model at a node are stored as local data structures in the Antnet agent. It is responsible for creating and forwarding forward ant packets and backward ant packets. The modules in the Antnet class implement the functionality of receiving ant packets, processing them according to the flow of the algorithm which includes to determine destination node and next hop node, and forwarding them to Antnet agent associated with next hop node. Whenever Antnet agent receives a forward ant, it adds the node address and trip time to the ant's memory, determines next node to which it has to be forwarded and sends it to Antnet agent associated with that node. When Antnet agent receives a forward ant destined to itself, it creates a backward ant packet and sends it along the reverse path as stored in ant's memory. When Antnet agent receives a backward ant packet, it updates the routing table and local traffic model and sends it further along the path that it has been retracing. Upon complete traversal of path, the backward ant packet is destroyed. The agent implements cycle detection and elimination, packet forwarding and update of traffic model and routing table according to the AntNet algorithm.

13 3) Results

We simulated AntNet on ns-2 for a number of topologies. It was observed that AntNet constructs probabilistic routing tables, wherein better paths among the available paths have higher pheromone concentration. Now follows the description of our experimental simulation on ns-2. Simulation parameters are as follows:

1. Simulation time = 11.1s. {This is the time which the algorithm takes to converge.} Fig. **??**. Network Topology used for simulation results on ns-2 2. Time interval at which forward ants are launched = 0.02s As an example for the explanation of results, consider a packet having source node 2 and destination node 1 in Fig. **??**.It can be observed that among all the neighbor nodes of node 2, node 0 provides the best path of only one hops for node 0.Nodes 3 provide next best paths, both of them being almost equally good. Node 3 is the worst of all possible paths, because it results in a cycle between nodes 1 and 2. The probability of selecting node 0 as the next hop is 0.817340, which is the highest among all neighbors. Next higher values are those corresponding to node 3 (0.040842) both of them being almost equal.

The simulation time of 10.8 seconds represents the saturation time required for the convergence of the algorithm with forward ant packets launched every 0.03 seconds. Since, routing tables generate the best paths; the time taken by packet to travel from source to destination must be the minimum possible.

14 IX. CONCLUSION

We successfully simulate AntNet on Network simulator, ns-2. Our experiments generated correct routing tables for all the topologies simulated. In this paper an improved version of the AntNet algorithm is proposed for mobile ad-hoc network. In the improved version, more than one optimal outgoing interfaces are identified as compared to only one path, which are supposed to provide higher throughput and will be able to explore new and better paths even if the network topologies gets changed very frequently. This will distribute the traffic of overloaded link to other preferred links. Hence the throughput of the network will be improved and the problem of stagnation will be rectified in mobile ad-hoc network. [1] [2] [3] [4]

2

Figure 1: 2)

[1] March 2011Multiple Feasible Paths in Ant Colony Algorithm for mobile Ad-hoc Networks with Minimum Overhead ©2011 Global Journals Inc. (US)

[2] March 2011©2011 Global Journals Inc. (US)

[3] March 2011Simulation Result ©2011 Global Journals Inc. (US)

[4] March 2011This page is intentionally left blank

? The difference d amongst the adjacent value is calculated and is compared to some threshold value say pm.
? If the difference d is less than pm then those values are selected and comparison amongst the adjacent values is continued until difference is greater than pm.
? Otherwise at the very first occurrence of difference greater than pm, the comparison is stopped and the corresponding value(s) in the array is (are) selected.
? The interfaces corresponding to these values are stored in a new routing table which will have the same structure as the original one, but obviously, the new table will have less number of rows.

Figure 2:

This algorithm is efficient for end to end delay and packets delivery ratio. We will simulate our proposed algorithm with one of network simulators and we compare the scheme performance with other routing algorithms and simulator.

[Baran and Sosa ()] 'A new approach for AntNet routing'. B Baran , R Sosa . *presented at the Proc.9th Int. Conf.Computer Communications Networks*, (Las Vegas, NV) 2000.

[Dorigo et al. ()] 'Ant algorithms for discrete optimization'. M Dorigo , G D Caro , L M Gambardella . *Artif. Life* 1999. 5 (2) p. .

[Caro and Dorigo ()] 'Ant colonies for adaptive in packet-switched communications networks'. G D Caro , M Dorigo . *Proc. 5th Int. Conf. Parallel Problem Solving from Nature*, (5th Int. Conf. Parallel Problem Solving from NatureAmsterdam, The Netherlands) Sept 27-30, 1998.

[Dorigo and Caro ()] *Antnet: A mobile agents approach to adaptive routing*, M Dorigo , G D Caro . 1997. University Libre de Bruxelles, IRIDIA (Tech Report)

[Caro and Dorigo ()] 'Antnet: Distributed stigmergetic control for communications networks'. G D Caro , M Dorigo . *Journal of Artificial Intelligence Research* 1998. 1998. 9 p. .

[Subbramanian et al. ()] *Ants and reinforcement learning: A case study in routing in dynamic networks*, D Subbramanian , P Druschel , J Chen . 1997. Proces Alto, Morgan Kauffman. p. .

[Schoonderwoerd et al. ()] 'Ants for Load Balancing in Telecommunication Networks'. O Schoonderwoerd , J Holland , L Bruten , Rothkrantz . HPL-96-35. *Hewlett Packard Lab* 1996. (Tech. Rep.)

[Appleby and Steward ()] 'Mobile software agents for control in telecommunication networks'. S Appleby , S Steward . *BT echnol* 1994. 12 (2) .

[Bonabeau and Suyers ()] 'Routing in telecommunication networks with smart like agents'. E Bonabeau , D Suyers . *Proceedings of LATA*, (LATA) 1998.

[Caro and Dorigo ()] 'Two ant colony algorithms for best-effort routing in datagram networks'. G D Caro , M Dorigo . *Proc. 10th IASTED Int. Conf. Parallel Distributed Computing Systems*, (10th IASTED Int. Conf. Parallel Distributed Computing Systems) 1998. p. .