ISSN 2320 – 2602
Volume 3, No.8, August 2014
International Journal of Advances in Computer Science and Technology
Available Online at http://warse.org/pdfs/2014/ijacst01382014.pdf

STUDY THE IMPACT OF PROACTIVE INTERVAL IN ANTHOCNET

V.Manohar[1], V.Narasimha Swamy[2], A.Durga Devi[3]
[1] P.G Research Scholar, J.N.T.U.A College of Engineering Pulivendula, INDIA, vankammanohar@gmail.com
[2] M.Tech Student, SRTS, Kadapa, INDIA, swamy554@gmail.com
[3] M.Tech Student, J.N.T.U.A College Of Engineering Pulivendula, INDIA, durga.arigala@gmail.com

ABSTRACT

A mobile ad-hoc network is a collection of self-organized mobile nodes. This network doesn't require existing infrastructure or central administration. As the nodes have mobility, the biggest challenge in this kind of networks is to find a path i.e., routing between the communications end points. Ant colony optimization is a technique to solve problems like routing in ad-hoc networks based on food searching behavior of ants. All ant colony algorithms are subset of Swarm Intelligence which means collective behavior of individual ants. All the ant based algorithms are mainly depended on pheromone concentration. Pheromone is a volatile chemical substance secreted by ants from nest to food source in order to influence other ants to follow them. The route will be discovered by the concentration of pheromone values.

In this project, To investigate the internal working of AntHocNet is focused on study the impact of Proactive Interval in AntHocNet at different parameters (different number of nodes, number of pause times, number of speeds and number of data rates) and various metrics (End to End Delay, Routing Overhead, Packet Delivery Ratio, Packet Loss, Throughput, Jitter), in UDP traffic type. Finally in reference to results obtained, Evaluation is done based on the contribution of each parameter in AntHocNet using the network simulator (ns-2.34).

Key words: AHWMN, ACO, AntHocNet

.

1. INTRODUCTION

Ant Colony Optimization routing was originally inspired by mechanisms found in biology: it is based on principles that are present in the foraging behavior of ants in nature, and on the ACO frame work for optimization that was derived from these principles. ACO routing algorithms work in a highly distributed way, and have properties such as adaptively, robustness and scalability. This makes them particularly interesting to deal with the challenges in adhoc routing. It has been observed that ants from e.g. the family of Argentine ants Linepithema Humile are able to find the shortest path between their nest and a food source. This is remarkable because each individual ant is a rather simple creature, with very limited vision and computing power, and finding the shortest among several available paths is certainly beyond its capabilities. The only way that this difficult task can be realized is through the cooperation between the individuals in the colony. The key behind the colony level shortest path behavior is the use of pheromone. This is a volatile chemical substance that is secreted by the ants in order to influence the behavior of other ants and of itself. Pheromone is not only used by ants to find shortest paths, but is in general an important tool that is used by many different species of ants (and also by a lot of other social animals) for a wide variety of tasks that involve coordinated behavior. The use of pheromone is an example of a form of indirect communication that is often referred to as stigmergy.

2. ANTHOCNET

AntHocNet is a multipath routing algorithm that combines both proactive and reactive components. It is based on Ant Net, designed for wired networks, with some modifications to be used on ad-hoc networks. For example, it does not maintain routes to all possible destinations at all times, but only for the open data sessions. This is done in a Reactive Route Setup phase, where reactive forward ants are sent by the source node to find multiple paths towards the destination node. Backward ants are used to actually setup the route. While the data session is open, paths are monitored, maintained and improved proactively using different agents, called proactive forward ants. The algorithm reacts to link failures with either a local route repair or by warning preceding nodes on the paths. AntHocNet reactively finds a route to the destination on demand, and pro actively maintains and improves the existing routes or explore better paths. In AntHocNet, ant maintains a list of nodes it has visited to detect cycles. The source node sends out forward ants and when it receive all the backward ants, one generation is completed. Each node i keeps the identity of the forward ants, the path computation, number of hops, of the ant from the source to node i, and the time the ant visited node i. Note that more than one ant may have reached node i and therefore the identity of the ant is important. When an ant arrives at a node, the node checks the ant's path computation and the time it reached node i. If the path computation and time are within a certain limit of those produced by another ant of the same generation then the ant is forwarded. Otherwise, the ant is discarded.

2.1 The Impact Of Proactive Interval In AntHocNet

The proactive ant send interval is the time between successive proactive ants in the proactive path maintenance and improvement phase. It defines how often the algorithm looks for path improvements, and therefore how quickly it can adapt to changes. We made tests with send intervals of 0.5, 1, 2, 5, 10, 20, and 50 seconds. All tests show a similar pattern. A too low ant send interval leads to bad performance, because the network gets flooded by ants. At 2 seconds, there seems to be an optimal send interval. For frequencies lower than that, the performance decays because the algorithm is not sending enough ants to keep up with the changes in the network. For low speeds, this decay is slower since the network changes less fast. However, it is interesting to note that the best send interval value is independent of the node speed. We also did some tests keeping the speed constant on 10 m/s and 20 m/s varying the data tra□c load. Although higher tra□c load could be expected to leave less space for ants, also there the best ant send interval was always around 2 seconds. The general e□ect of the use of proactive actions was also evident the significant decrease in performance when the proactive mechanisms are switched on with respect to the case of using the same metric but switching proactivity on.

The proactive path update phase is useful in maintaining paths set up in the reactive path set up phase. This phase proactively improves the existing path. At some point in communication session the source node bring into play proactive forward ants to update the information about the currently used paths to the destination, and tries to find alternate paths. This Proactive mechanism is achieved with the bootstrapped information in the form of hello messages. The hello messages are periodically broadcasted. These are short messages broadcast every t hello seconds by the nodes (e.g. t hello = 1 sec). If a node k receives hello message from a node j, k assumes that j is its neighbor and expects to receive hello message from j every t hello seconds. If k misses certain number of hello messages from j, it assumes that j is no longer its neighbor. While broadcasting, the node construct the hello message by including routing information it has about active destinations. The node obtains information about active destinations from its pheromone table. If there are many active destinations at a node, it randomly selects number of destinations. When a node k receives the hello message from j, it will check if it has an entry for destination d over neighbor j. If there is no entry, it is an indication of a possible new path from k to d over j. If it has entry for d but over other intermediate node it is hint of potential alternate path to d. To build hello message the node consult its pheromone table and put together its bootstrapped pheromone value which is indicated as an alternative pheromone to the regular pheromone constructed on the reactive phase by the reactive backward ants. This virtual pheromone is placed in the Virtual Pheromone Routing Table to avoid mixing the regular pheromone values with virtual ones and create routing loops. Each node compares their regular pheromone with virtual pheromone. The regular pheromone is changed by the bootstrapped one if the virtual pheromone is considerably better than the regular one. In case if a node hasn't got any routing information for a destination and if it is available in the bootstrapped one, then this information will be used and a new route to the destination is activated. Proactive forward ants are unicast and uses values in Virtual Pheromone routing table as much as possible. If there is no entry in Virtual pheromone routing table, then only the regular pheromone table is used to make its way to the destination. At a node where there is no information available for destination d in both the tables, the proactive forward ant is simply discarded. The proactive forward ant is converted into proactive backward ant when it successfully reaches the destination and it traces back the intermediate nodes it visited to the source and at the same time it removes the entries from the virtual pheromone table into the regular pheromone routing table. Both the reactive path setup and proactive path maintenance phases create paths between the source and destination. Now using the pheromone table entries, the data is forwarded.

Metrics are calculated at different speeds, pause times, nodes and date rates by using different mobility models constant. The values fixed for different parameters are shown in Table 1 fixed values. Where the speeds are varied 10 and 20 m/sec and pause times 50 sec and data rates taken as 0.5Mbps and udp traffic type. The mobility model used here is random waypoint mobility model. When one of the parameters is varied the others are kept constant.

3. SIMULATION ENVIRONMENT

To test the performance of AntHocNet protocol, the network simulator NS-2.34, is used. The network model used in our simulation is composed by mobile nodes and links that are considered unidirectional and wireless. Each node considered as communication endpoint is host and a forwarding unit is router. In addition to NS-2, a set of tools, mainly Bash scripts and AWK filters, to post-process the output trace files generated by the simulator are developed. In order to evaluate the performance, multiple experiments were set up.

4. PERFORMANCE METRICS

Different performance metrics are used in the evaluation of routing protocols. They represent different characteristics of the overall network performance. In this report, we evaluate three metrics used in our comparisons to study their effect on the overall network performance. These metrics are Average end to end delay, Routing Overhead, packet delivery ratio, Packet Loss, Throughput, Jitter.

Average End-to-End Delay: This is defined as the average delay in transmission of a packet between two nodes. This metric describes the packet delivery time: the lower the end-to-end delay the better the application performance.

D = (Tr - Ts) Where Tr = receive time and Ts = sent time.

Routing Overhead: It is the total number of control or routing (RTR) packets generated by routing protocol during the simulation. All packets sent or forwarded at network layer are considered as routing overhead. This metric provides an indication of the extra bandwidth consumed data traffic.

Overhead = Total number of routing packets/number of packets received

Packet Delivery Ratio: It is the ratio of data packets delivered to the destination to those generated by the sources. It is calculated by dividing the number of packet received by destination through the number packet originated from source.

Packet delivery Fraction = (total no. of data packets delivered / total no. of data packets generated)

Packet Loss: The difference between the total number of packets send by the source and number of packets received to the destination.

Packet Loss = number of packets sent – number of packets received.

Throughput: The number of successful delivered data packets per unit time in the network. In other words, the total number of received packets at the destination out of total transmitted packets. This can be calculated in Number of Bytes/second.

Throughput= (Total number of received packets at destination *Packet size) Total simulation time

Jitter: This is the variation in the time interval between the arrivals of subsequent packets.

5. RESULTS AND ANALYSIS

5.1 Varying the Proactive Ant Send Interval at Speed 20m/Sec:

a. End- to- End Delay

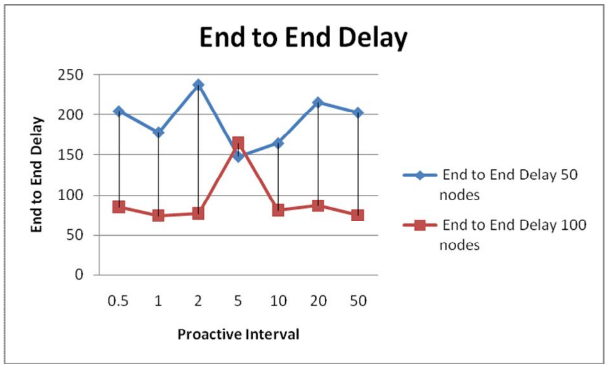

Figure 1: End- to- End Delay for AntHocNet

From the above figure 1: It is clear that AntHocNet having less Average End- to -End Delay when Proactive interval is 5sec in small networks(i.e. 50 nodes) and 1sec in large networks(i.e. 100 nodes). This is due to proactive route maintenance process.

b. Routing Overhead

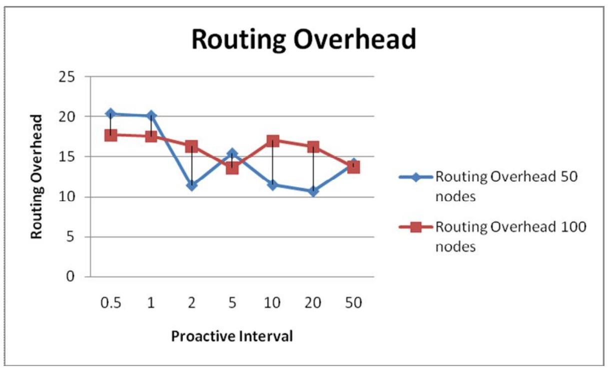

Figure 2: Routing Overhead for AntHocNet

From the above figure 2: It is clear that AntHocNet having less Routing Overhead when Proactive interval is 20sec in small networks(i.e. 50 nodes) and 5sec in large networks(i.e. 100 nodes). This is due to proactive route maintenance process.

c. Packet Delivery Ratio

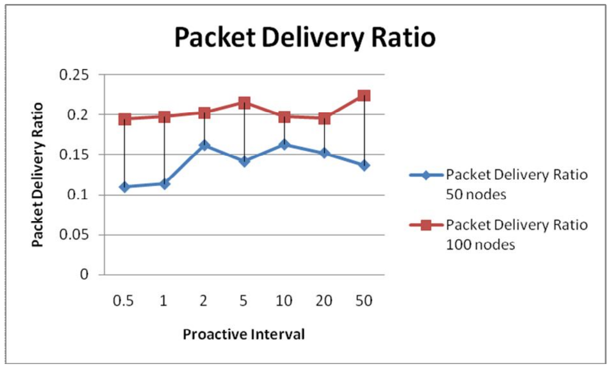

Figure 3: Packet Delivery Ratio for AntHocNet

From the above figure 3: It is clear that AntHocNet having high Packet Delivery Ratio when Proactive internal is 10sec in small networks(i.e. 50 nodes) and 50sec in large networks(i.e. 100 nodes). This is due to proactive route maintenance process.

d. Packet Loss

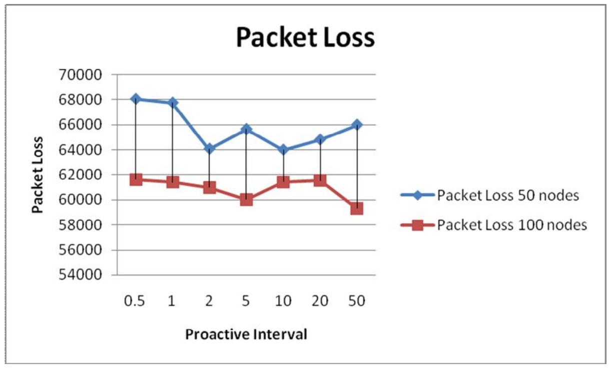

Figure 4: Packet Loss for AntHocNet

From the above figure 4: It is clear that AntHocNet having less Packet Loss when Proactive interval is 10sec in small networks(i.e. 50 nodes),50sec in large networks(i.e. 100 nodes). This is due to proactive route maintenance process.

e. Throughput

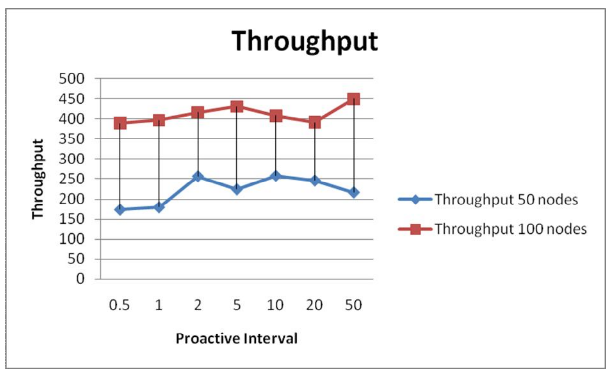

Figure 5: Throughput for AntHocNet

From the above figure 5: it is clear that AntHocNet having high throughput when Proactive interval 10sec in small networks (i.e.

50 nodes), 50sec in large networks (i.e. 100 nodes). This is due to proactive route maintenance process.

f. Jitter

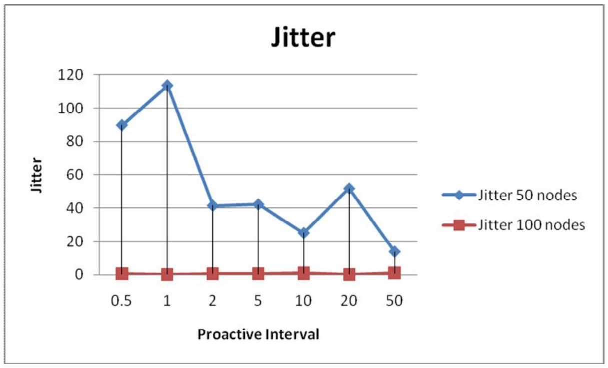

Figure 6: Jitter for AntHocNet

From the above figure 6: it is clear that AntHocNet having less Jitter when Proactive interval is 50sec in small networks(i.e. 50 nodes),1sec in large networks(i.e. 100 nodes). This is due to proactive route maintenance process.

6. CONCLUSION

The investigation of AntHocNet is to focus on study the proactive ant send interval for various parameters like pause times, speeds and different number of nodes for AntHocNet are calculated by using the performance metrics. A too low ant send interval leads to bad performance, because the network gets flooded by ants. At 2 seconds, there seems to be an optimal send interval. For low speeds, this decay is slower since the network changes less fast. However, it is interesting to note that the best send interval value is independent of the node speed. From the results observe that some tests keeping the speed constant on 10 m/s and 20 m/s are varying the data tra□c load. Although higher tra□c load could be expected to leave less space for ants, also there the best ant send interval was always around 2 seconds.

REFERENCES

[1]. Prof. P. Chenna Reddy and T. Nishitha. Bio inspired Routing in Ad-hoc networks, *Journal of engineering , computers and applied sciences*, ISSN : 2319-5606, Vol.1, no.1, October 2012.

[2]. R. M. Sharma. Performance Comparison of DSR, AODV and AntHocNet Protocols, *International Journal of Computer Science and Technology*, Vol.1, Issue 1, Sept 2010.

[3]. Gianni Di Caro, Frederick Ducatelle and Luca Maria Gambardella. AntHocNet: An Adaptive Nature-Inspired Algorithm for Routing in Mobile Ad Hoc Networks, *Published online in Wiley InterScience [www.interScience.wiley.com]*, Vol.16, pp - 443-455, May 2005.

[4]. Gambardella L.M., Ducatelle F., Di Caro G.A.. AntHocNet: an ant-based hybrid routing algorithm for mobile ad hoc networks, *Proceedings of PPSN VIII - Eight International Conference on Parallel Problem Solving from Nature*, Sept 18, 2004.

[5]. The VINT project. The ns Manual (formely ns note and documentation). *http://www.isi.edu/nsnam/ns/ns-documentation.html*, November 2011H.

[6]. David A. Maltz, Jorjeta Jetcheva , David B. Johnson, Yih-Chun Hu, Josh Broch. Performance comparison of Multi-Hop wireless routing protocols, *In the Proceedings of Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Dallas, Texas, USA, Oct 1998.

[7]. Shiyou Yang H. C. Wong, and S. L. Ho. An Improved Ant Colony Optimization Algorithm and Its Application to Electromagnetic Devices Designs.*IEEE TRANSACTIONS ON MAGNETICS*, Vol.41, issue:5, pages: 1764-1767, May 2005.