Dynamically Adapting Mobile Ad Hoc Routing Protocols to Improve Scalability

Hwee Xian TAN[1, 2] Winston K G SEAH[2, 1]
{stuhxt, winston}@i2r.a-star.edu.sg

[1]Department of Computer Science
School of Computing
National University of Singapore
3 Science Drive 2, Singapore 117543

[2]Institute for Infocomm Research
Agency for Science Technology and Research
21 Heng Mui Keng Terrace
Singapore 119613

Abstract

While numerous routing protocols have been designed for mobile ad hoc networks, many of them have assumed certain network conditions and built in system parameters which are preset to suit these assumptions. We show that by simply changing certain system parameters dynamically, we are able to improve the performance of the network. In this paper, we demonstrate this concept using the Ad hoc On Demand Distance Vector (AODV) protocol, by adding simple algorithms to enable it to adapt its behaviour according to network characteristics. Focusing on just one aspect, we are able to improve network performance without changing the protocol architecture.

Keywords: wireless networks, mobile ad hoc networks, AODV, scalability

I. Introduction

A mobile ad hoc network (MANET) refers to a group of mobile nodes that are able to communicate with each other without the use of an existing infrastructure. Hence, no central administration is required to set up such a network, which is often useful in hostile terrains and areas of defense.

Current routing protocols that are used by MANETs can be classified into reactive and proactive protocols. Reactive protocols are those that compute routes on demand and do not maintain an overall view of the entire network topology. Hence, they usually incur longer end to end delay during the transmission of packets, because time is needed for route establishment if the required routes do not already exist. Such algorithms include the Dynamic Source Routing (DSR) protocol [1] and the Ad Hoc On Demand Distance Vector (AODV) protocol [2][3]. On the contrary, proactive protocols such as the Optimized Link State Routing (OLSR) protocol [4] and the Topology Based Reverse Path Forwarding (TBRPF) [5] are those that pre-compute the existing paths between nodes in a network and can therefore provide route information quickly as needed. However, any change in the routing information of a particular node may initiate widespread updates throughout the network, resulting in performance deterioration. There are also some protocols that fall in between these two categories – hybrid protocols that attempt to make use of the benefits of both reactive and proactive routing protocols to improve the efficiency and performance of the network. Prominent hybrid protocols that exist include the Zone Routing Protocol (ZRP) [6] and the Cluster Based Routing Protocol (CBRP) [7].

Due to the nature of MANETs, a node may experience a vast spectrum of network dynamics [8]. Such network conditions refer to varying network topology, data congestion, shared medium contention, varying traffic loads and varying traffic patterns, etc. However, many protocols have assumed certain network conditions and built in system parameters which are preset to suit these assumptions. Nevertheless, there has been some reported work in adaptive protocols, which include the following: Adapting to Route Demand and Mobility (ARM) protocol [9], Adaptive Reservation and Pre-allocation Protocol (ASAP) [10], SHARP Proactive Routing Protocol (SPR) [11] and Adaptive Distance Vector (ADV) protocol [12].

In this paper, we focus on how certain preset system parameters can be modified dynamically to adapt to specific network characteristics to improve performance. Using AODV as the example, we show that it can be done by varying the frequency of broadcasts of the Hello packets from each node, based on the mobility of the nodes. Specifically, we will evaluate the performance of the adaptive AODV as compared to the existing AODV, with respect to throughput, number of Hello packets that are being transmitted, end to end delay as well as the packet delivery ratio.

The rest of the paper is organized as follows: The next section provides an overview of the AODV routing protocol. Section III discusses our motivation and Section IV describes our adaptive algorithm. We present simulation results and analysis in Section V. Section VI concludes, together with the directions for future work.

II. AODV Routing Protocol

The AODV routing protocol is a dynamic routing protocol for use by mobile nodes in an ad hoc network. It avoids routing loops with the use of sequence numbers, and enables participating mobile nodes to adapt quickly to changes in network topology as well as link breakages. It also provides lower byte overheads in relatively static networks.

There are 3 types of control messages used by AODV: RREQ (Route Request) messages are initiated from the source node when it needs to send data to a destination node which it does not have a valid or existing path. Each node that receives the broadcasted RREQ message will update its routing table to reflect a route back to the source. RREP (Route Reply) messages may be initiated by either the target node or intermediate nodes if the latter has a valid route to the destination that is "fresh enough", based on the sequence numbers. RERR (Route Error) messages are used to notify the other nodes which use routes that have broken links, by means of a precursor list. The precursor list contains the addresses of nodes that have received or forwarded RREPs, and which may therefore be forwarding packets along the valid routes in the routing table.

One distinct aspect of the AODV protocol is its ability to provide connectivity information via the use of Hello messages. These are RREP packets with their Time-To-Live (TTL) set to 1, and are broadcasted locally to all neighbouring nodes within the vicinity of any particular node. A node uses Hello messages only if it is part of an active route. Every HELLO_INTERVAL milliseconds, the node checks if it has sent a broadcast within the last HELLO_INTERVAL. If not, it may broadcast a Hello message to its neighbours, which can then determine the local network connectivity.

A node that receives a Hello message from a neighbour should ensure that it has an active route to that neighbour, and create one as necessary. If the route already exists, then the lifetime of the route should be increased to be at least ALLOWED_HELLO_LOSS × HELLO_INTERVAL. Currently, the default value of HELLO_INTERVAL is set to be 1000 milliseconds, as specified in RFC 3561.

III. Motivation

The motivation of our work is not to invent a new routing protocol as there are already many well-tested ones available, e.g. those adopted by the Internet Engineering Task Force, viz., AODV, DSR, OLSR and TBRPF. However, we argue that the current designs are not flexible enough to achieve optimal performance. Taking the example of AODV, each forwarding node in an ad hoc network has to maintain information of its connectivity with other active next hops and neighbouring nodes. The use of Hello messages provides local network connectivity information, especially in the absence of link layer notifications such as the RTS/CTS (Request to Send/Clear to Send) dialogues. These are handshakes that are established between the transmitting and receiving nodes prior to the actual transmission, and provide some degree of performance improvements over traditional CSMA (Carrier Sense Multiple Access) protocols.

However, use of Hello messages also contributes to the overall network traffic and affects performance. As Hello messages are being broadcasted locally by the nodes, they contend with the data packets for bandwidth. They may also increase the probability of collisions with data packets or other control messages in the network, which can lead to MAC backoff and buffer overflows at the interface queues. These factors amongst others, result in an overall reduction in network utilization, throughput and data delivery rate, as well as increase in the packet loss ratio.

As such, it is important to ensure that the number of Hello messages propagating in the network is sufficient to provide local connectivity information without degrading the overall network performance.

IV. Adaptive AODV Algorithm

We propose an adaptation of the AODV routing protocol by varying the frequency of broadcasts of Hello packets from each node, according to the node mobility of its neighbours.

```
During initiation of each Hello packet
  Calculate node mobility, N_m
  If (N_m < LOW_THRESH)
    Deviation_Frac = LOW_FRAC;
  Else if (N_m > HIGH_THRESH)
    Deviation_Frac = HIGH_FRAC;
  Else Deviation_Frac = 1;
  Timer for next Hello message =
    HELLO_INTERVAL*Deviation_Frac;
  Update previous neighbour table;

To calculate node mobility, N_m
  For any particular node
    Compare the current and previous neighbour table
    New_x = number of new neighbours
    Left = number of neighbours that have moved away
    Node mobility, N_m = New_x + Left;
```

Figure 1: Dynamic Hello interval for adaptive AODV

Figure 1 shows the pseudocode for our adaptive AODV scheme. During the initiation of each Hello packet, we calculate the node mobility, N_m, of that particular node. This is done by comparing the differences between the current neighbour table of the node and the previous neighbour table that was obtained in the last time period. The node mobility is then utilized to set a different deviation fraction for the HELLO_INTERVAL, which determines when the next Hello message should be initiated. Basically, where there is low node mobility, we increase the time interval between the sending of the

HELLO messages to reduce network congestion and bandwidth contention, since the neighbours are less likely to change. During high node mobility, the rate of change of neighbours is higher and may lead to more link breakages, hence the reduced HELLO_INTERVAL in between broadcasts. After we set the timer for the next Hello broadcast, we then update the previous neighbour table for the node. This is done by copying all the contents of the current neighbour table in the former table.

By reducing the number of unnecessary broadcasts of Hello messages, network control traffic is reduced. This has a number of chain effects, which includes less contention for bandwidth with the data packets and other control packets such as RREQ, RREP and RERR messages. The reduction in network overhead will also lead to lower packet loss, higher throughput and better overall utilization of network resources.

V. Simulation Results and Analysis

We verify the correctness of our design by running simulations on GloMoSim [13], which provides a scalable simulation environment. Each simulation was run for a period of 300 seconds, with the underlying MAC protocol being IEEE 802.11 wireless LAN. Nodes are uniformly distributed and constant bit rate (CBR) traffic with data packets of size 512 bytes are transmitted at randomly chosen intervals of 100 milliseconds for different number of data sources. Each scenario is also run with different seed numbers and the measurements are averaged out to minimize any arbitrary randomness.

Using LOW_THRESH = 1, LOW_FRAC = 1.25, HIGH_THRESH = 5 and HIGH_FRAC = 0.75, we study our proposed scheme under two mobility models [14]:

1. Random Waypoint mobility model, where nodes move towards a random destination with speeds between $10ms^{-1}$ to $20ms^{-1}$ and stay there for a specified pause time.
2. Reference Point Group Mobility model (RPGM), where group movements are based upon the path traveled by a logical centre.

We compare the performance of our adaptive AODV scheme against that of the original AODV protocol by varying the following parameters: terrain size, node density, number of CBR data sources and mobility speeds. The following performance measures are then observed in our study: throughput, number of hello packets that are being transmitted, and packet loss ratio.

In Figures 2-5, we present the performance results under the random waypoint mobility model with a pause time of 30s and a terrain size of 2000×2000 metres. As can be observed, our adaptive AODV using dynamic Hello intervals performs better than the original AODV routing protocol with respect to the overall throughput (given as the total number of kilobytes being delivered to all the destinations), packet delivery ratio (total number of packets being received at the destinations as a fraction of the total number of packets originated from the sources) and end to end delay.

This improvement for throughput is more significant in a smaller network with about 50 to 100 nodes. With the same volume of network traffic distributed among fewer nodes, the probability of contention is higher and our adaptive scheme has shown that it can effectively improve the network performance.

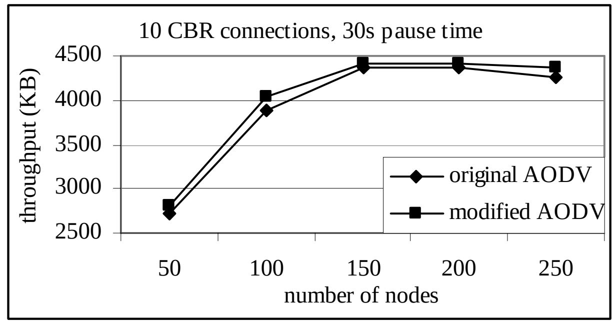

Figure 2: Throughput vs number of nodes

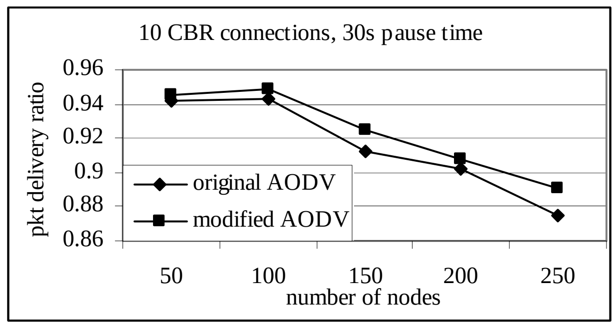

Figure 3: Packet delivery ratio vs number of nodes

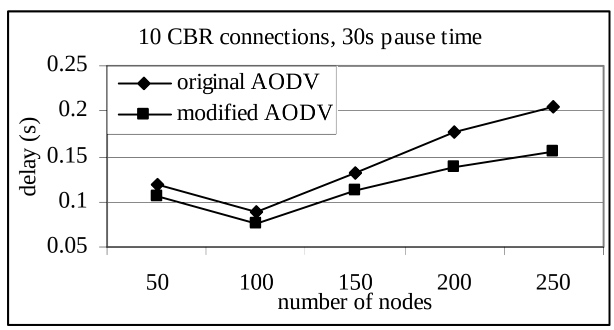

Figure 4: Delay vs number of nodes

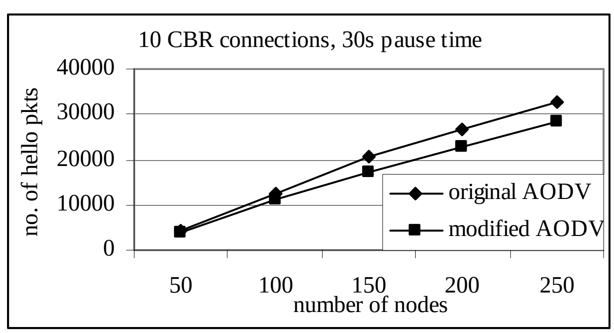

Figure 5: No. of hello packets transmitted vs number of nodes

Figure 5 also shows a significantly lower number of Hello packets released into the network by our adaptive AODV scheme, especially in a network with large number of nodes. This will allow the network to support more data traffic.

We also simulated the two protocols under the random waypoint mobility model with 0s pause time, which represents continuous motion. The results show the same trend as those which were obtained earlier using pause times of 30s, in that there are performance improvements for throughput, packet delivery ratio, end to end delay as well as control overhead caused by Hello packets.

We next analyze the effect of data traffic on our adaptive AODV, by varying the number of data connections. 50 nodes are placed in a terrain size of 2000×2000 metres, and Random Waypoint is used to simulate node movement with the same parameters as before.

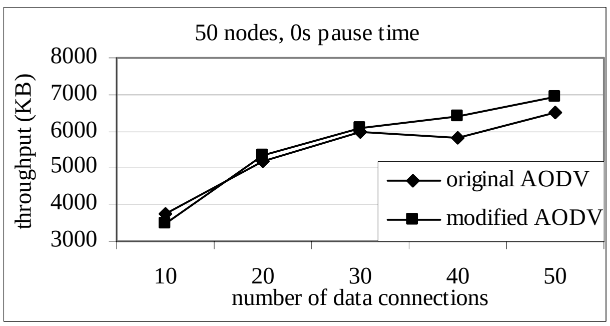

Figure 6: Throughput vs number of data connections

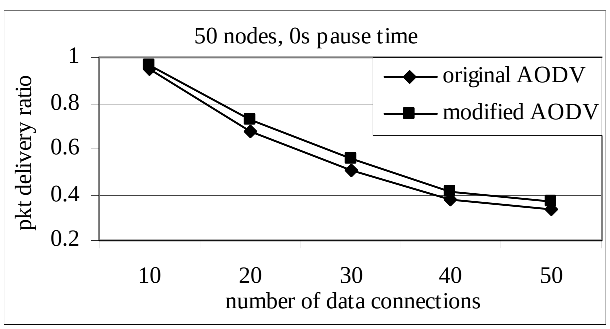

Figure 7: Packet delivery ratio vs number data connections

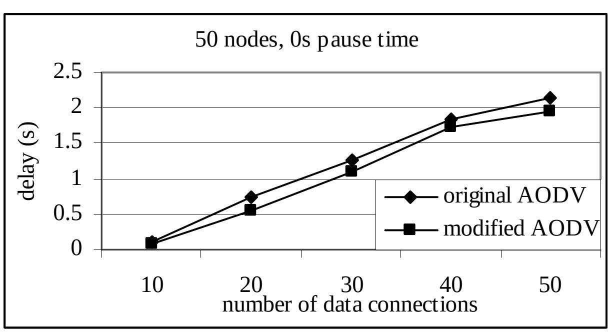

Figure 8: Delay vs number of data connections

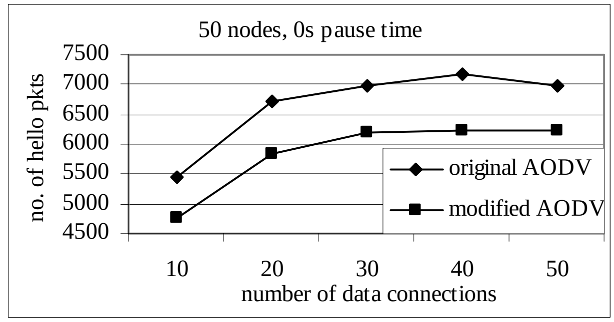

Figure 9: No. of hello packets transmitted vs number of data connections

Our results in Figures 6-9 show that the modified AODV scheme performs better than the original AODV protocol, especially for higher number of data connections. We also performed the simulations using a pause time of 30s and achieved consistent improvements as like the case of 0s pause time.

In the following sets of simulation results, we demonstrate that our design is flexible enough to work under different mobility models such as the Reference Point Group Mobility (RPGM) model. Nodes are simulated in a terrain size of 2000×2000 metres with a maximum and minimum speed of $20ms^{-1}$ and $10ms^{-1}$ respectively. The average number of nodes per group is set to 10, and the maximum distance from the group centre is set at 750 metres (i.e. three times the maximum transmission radius of a node.)

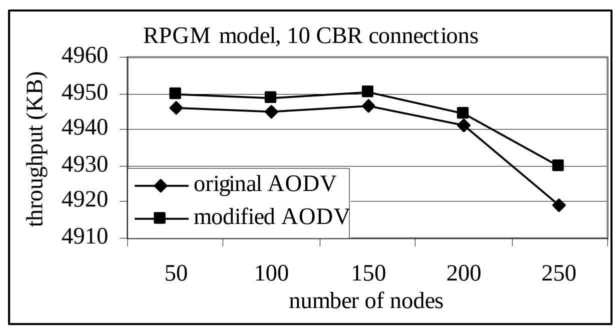

Figure 10: Throughput vs number of nodes

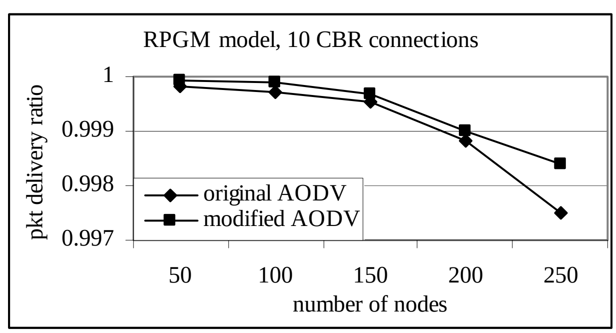

Figure 11: Packet delivery ratio vs number of nodes

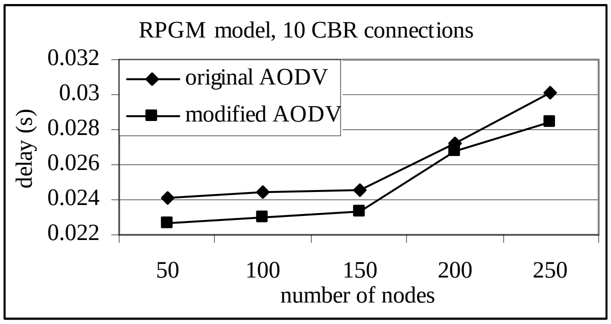

Figure 12: Delay vs number of nodes

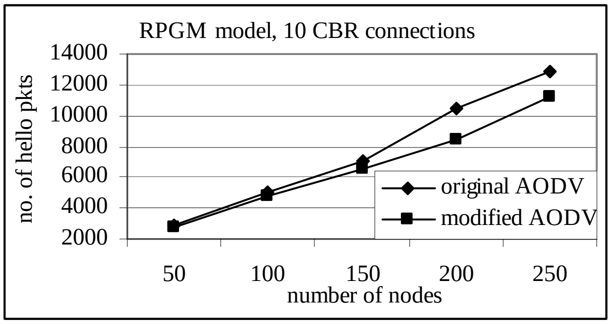

Figure 13: No. of hello packets transmitted vs number of nodes

Figures 10-13 show the performance comparison using 10 CBR connections and a pause time of 30s. Each CBR connection sends data packets of size 512 bytes at every 100 milliseconds interval. Figure 13 shows that there is a more significant drop in the number of Hello messages that are being released into the network (about an average of 20% decrease). This is expected because with group mobility, nodes do not change their neighbours as often as when under the Random Waypoint model, using the same speed and pause time parameters. This relative stability among some nodes (generally in clusters of 5-6 hops in diameter) also helps the original AODV achieve better throughput and the gain in throughput achieve by our adaptive scheme is less significant, in low traffic load conditions.

Figures 14-17 show the performance of the adaptive AODV as compared to the original AODV, under varying traffic loads. Here, packets of 512 bytes were originated from a different number of sources at a time interval of 100 milliseconds. We see that as like before, there are marked improvements in the modified AODV with respect to overall throughput, packet delivery ratio, end to end delay and the number of Hello packets that are being transmitted in the network. However, under heavy traffic loads (more than 40 data connections in a network size of 50 nodes), achieved by increasing the number of data connections, has caused the overall performance to degrade. This is due to the excessive contention of the bandwidth in the network, leading to higher number of collisions, lower throughput and higher numbers of packets being lost. Hence, the need to reduce unnecessary overheads, like Hello messages, is even more important and we have shown that our scheme, by doing that, is able to improve the network performance.

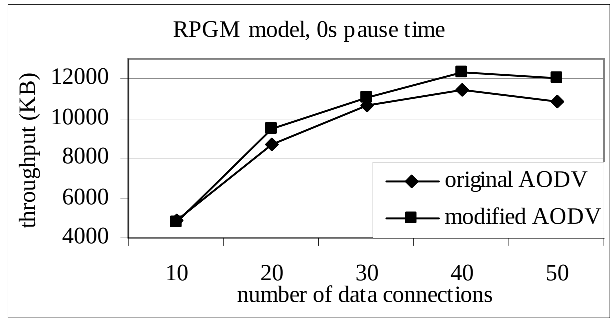

Figure 14: Throughput vs number of data connections

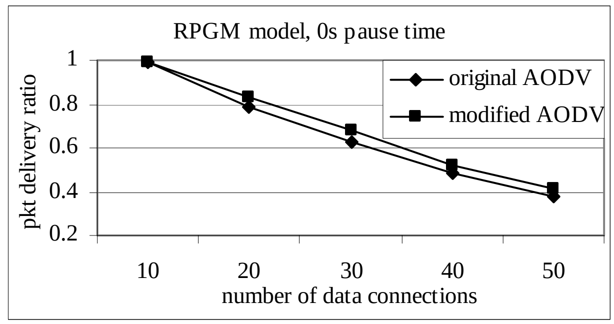

Figure 15: Packet loss ratio vs number of data connections

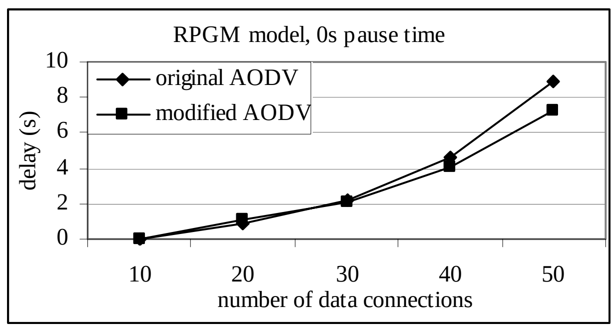

Figure 16: Delay vs number of data connections

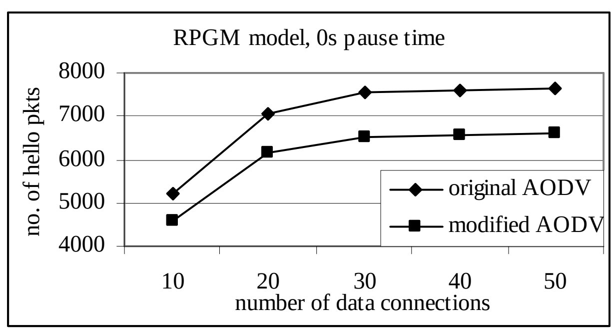

Figure 17: No. of hello packets transmitted vs number of data connections

VI. Conclusion and Future Work

There is a growing trend towards adaptive protocols for MANETs that can improve performance over a wide range of network dynamics, as compared to conventional routing protocols. In this paper, we have shown that by dynamically tuning certain system parameters, the network performance can be improved, and used AODV as an example to demonstrate this concept. The AODV protocol is a reactive routing protocol that is commonly used in ad hoc networks because of its scalability and quick adaptation to node movements and link breakages. It is also loop free and has lower overhead as compared to other routing protocols such as DSR. However, despite its benefits, there are several modifications that can be made to the AODV routing protocol to make it more adaptive to changes in node mobility and other network characteristics.

We have proposed a scheme to dynamically adjust the time interval in between Hello message broadcasts from active nodes in the network. These Hello message broadcasts are used by nodes to transmit local connectivity information, and have been statically configured to have a broadcast interval of 1000 milliseconds. Our adaptive algorithm computes the node mobility of the network environment by calculating the rate of change of neighbours of a particular node, before adjusting the time interval for the next Hello broadcast.

In general, high node mobility is characterized by a high rate of change of neighbours and this may lead to frequent changes in network topology and link breakages. To minimize the effect of route errors, the HELLO_INTERVAL in between broadcasts is set to a lower deviation fraction so that such link failures due to movement of nodes can be detected more quickly. With low node mobility, nodes do not move around that often and links between nodes are more stable. As such, the HELLO_INTERVAL is set to a higher deviation fraction to reduce the number of Hello packets propagating in the network, thus reducing the network control overhead. Consequently, this leads to lower network congestion, lower rate of bandwidth contention between Hello packets and the data packets, lower packet loss ratio and overall higher throughput. All these are crucial performance metrics in a mobile ad hoc network.

Our simulation results have verified the correctness of the adaptive AODV design by the reduction of Hello messages in the network, the increased throughput, increased packet delivery ratio and decreased end to end delay.We are currently working on other ways to improve the performance of MANET routing protocols by developing schemes to dynamically adapt them according to other network characteristics, including link stability, mobility models, network size, traffic characteristics, traffic patterns, etc. The objective is not to change the protocol architecture, and ensure that nodes running an adaptive version of the protocol can still communicate with nodes using the basic protocol.

References

[1] David B. Johnson and David A. Maltz, Dynamic Source Routing in Ad Hoc Wireless Networks, *Mobile Computing* (Kluwer Academic Publishers, pages 153-181, 1996).

[2] Charles E. Perkins, Elizabeth M. Royer and S. Das, Ad Hoc On-Demand Distance Vector (AODV) Routing, *IETF RFC3561, July 2003.*

[3] Elizabeth M. Royer and Charles E. Perkins, An Implementation Study of the AODV Routing Protocol, *Proc. IEEE Wireless Communications and Networking Conference*, Chicago, IL, September 2000.

[4] Thomas Clausen and Philippe Jacquet, Optimized Link State Routing Protocol (OLSR), *Internet Draft, draft-ietf-manet-olsr-11.txt, July 2003.*

[5] R. Ogier, F. Templin and M. Lewis, Topology Dissemination Based on Reverse-Path Forwarding (TBRPF), *Internet Draft, draft-ietf-manet-tbrpf-11.txt, October 2003*.

[6] Zygmunt J. Haas and Marc R. Pearlman, Determining the Optimal Configuration for the Zone Routing Protocol, *IEEE JSAC, Special Issue on Ad-Hoc Networks, 17*(8), August 1999.

[7] M. Jiang, J. Li and Y.C. Tay, Cluster Based Routing Ptotocol (CBRP), *Internet Draft, draft-ietf-manet-cbrp-spec-01.txt, August 1999.*

[8] Boleng J, Navidi W, and Camp T., Metrics to Enable Adaptive Protocols for Mobile Ad Hoc Networks, *Proc. The 2002 International Conference on Wireless Networks (ICWN '02)*, Las Vegas, Nevada, USA, June 2002, 293-298.

[9] Sungjoon Ahn and A. Udaya Shankar, Adapting to Route-Demand and Mobility in Ad Hoc Network Routing, *Computer Networks 38*(6), April 2002, 745-764.

[10] Jianbo Xue, Patrick Stuedi and Gustavo Alonso, ASAP: An Adaptive QoS Protocol for Mobile Ad-Hoc Networks., *Proc. 14th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC2003)*, Beijing, China, 2003.

[11] Venugopalan Ramasubramanian, Zygmunt J. Haas and Emin Gun Sirer, SHARP: A Hybrid Adaptive Routing Protocol for Mobile Ad Hoc Networks, *Proc. ACM Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc)*, Annapolis, Maryland, June 2003.

[12] Rajendra V. Boppana and Satyadeva P Konduru, An Adaptive Distance Vector Routing Algorithm for Mobile, Ad Hoc Networks, *Proc. 20th Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Infocom 2001)*, Anchorage, Alaska, March 2001.

[13] Lokesh Bajaj, Mineo Takai, Rajat Ahuja, Ken Tang, Rajive Bagrodia and Mario Gerla, GloMoSim: A Scalable Network Simulation Environment, *UCLA Computer Science Department Technical Report 990027*, 1999.

[14] T. Camp, J. Boleng, and V. Davies, A survey of mobility models for ad hoc network research, *Wireless Comm. and Mobile Computing (WCMC), 2*(5), 2002, 483-502.