

INTERNATIONAL JOURNAL OF  
RESEARCH IN COMPUTER  
APPLICATIONS AND ROBOTICS  
ISSN 2320-7345

# AN EFFICIENT SECURE DATA TRANSMISSION FOR ADAPTIVE HELLO MESSAGING SCHEME IN MANET

S. M. Dhivya, P. Narendran<sup>2</sup>

<sup>1</sup>M.Phil Research Scholar, PG & Research Dept. of Computer Science  
Gobi Arts & Science College (Autonomous), Gobichettipalayam - 638 453  
Swtdivya006@gmail.com

<sup>2</sup>Head of Comp. Science, PG & Research Dept. of Computer Science  
Gobi Arts & Science College (Autonomous), Gobichettipalayam - 638 453  
narendranp@gmail.com

**Abstract:** In mobile ad-hoc networks, a hello protocol is generally used to maintain neighbour relationship between the nodes. According to this, each node broadcasts hello packets at regular intervals to discover its intermediate nodes to forward a packet. When the node moves into an inactive region, the transmission of hello packets to those nodes get wasted which results in energy loss. In this paper, we implement a dynamic hello protocol to suppress those inessential hello packets without affecting the link failure detection. Our simulation result implies that the proposed design decreases the energy consumption and delay without any major difference in throughput.

**Keywords:** Hello protocol, Mobile ad-hoc network, Neighbour relationship, Energy consumption.

## I. INTRODUCTION

Mobile Ad-hoc Networks (MANET) does not depend on a fixed infrastructure for its operation. It is an autonomous transitory association of mobile nodes which communicates with each other over wireless links. Nodes that rely within each other's send range can communicate directly and are responsible for dynamic route discovery. To enable connection between several nodes which are not in range, neighbouring nodes act as routers. These nodes are often energy constrained. In this energy constrained, dynamic distributed multi-hop network, nodes organize themselves dynamically to provide route establishment. Even though mobile devices are in off condition, the energy is consumed by No sleep energy bugs [7]. Similarly in MANET, the energy of nodes get drained when they move out of range. Many energy conserving hello messaging schemes were introduced [5] following the dynamic network topology [6]. Those schemes doesn't describe the time of execution [5]. Ad-hoc On Demand Distance Vector (AODV) is used for route discovery and maintenance in which the control packets of route request (RREQ) and route reply (RREP) are exchanged between the nodes. It leads to excessive bandwidth and energy consumption. Two hello protocols were discussed to restrain the inefficient hello packets [4]: a reactive hello protocol and an event based hello protocol. Reactive hello protocol exhibits hello messages only to the demanding nodes using control packets (req-reply) which leads to network overhead. Event based hello protocol provides hello messages to the nodes which are in communication range based on threshold time. High threshold results in less delay whereas low threshold results in neighbour information loss. This paper proposes a dynamic hello interval for route establishment by restraining the hello messages. It regulates the hello interval conservatively without reducing the probability of link failure detection. We implement an average event interval i.e. the time gap between two continuous events (transmission or reception of packets) to notice the communication between nodes. The hello packets get wasted when a node doesn't participate in communication. Our proposed system uses a constant risk level for suppression

of hello messages. If we are increasing the event interval, the hello interval can also increase without increasing the risk of sending a packet through an unavailable link. Therefore hello messages get restrained. Simulation results show that our proposed scheme reduces energy consumption without any overhead.

## II. OVERVIEW OF AODV

AODV is an on-demand routing protocol where routes are established only as needed. When a route is needed, [2] it uses a route discovery process to understand a route. Once it determines a route, it is maintained as long as it is required by a maintenance procedure. It uses a software state approach to maintain routes: if a route is not in use, it expires after a specified time. It may use one of the two methods for link failure detection: link layer feedback or hello messages. Due to difficulty in obtaining link layer feedback, it uses hello messages.

### A. Hello messages

Network connectivity may be determined through the reception of broadcast control messages. Any broadcast control message also serves as a hello message, indicating the presence of a neighbour as shown in figure 1. When a node receives [3] a hello message from its neighbour, it creates or refreshes the routing table entry to the neighbour. To maintain connectivity, if a node has not sent any broadcast control message within a specified interval, a hello message [9] is locally broadcast. This results in at least one hello message transmission during every time period. Failure to receive an hello message from a neighbour for several time intervals indicates that neighbour is no longer within transmission range and connectivity has been lost. Two variables control the determination of connectivity using hello messages: HELLO\_INTERVAL specifies the maximum time interval between the transmissions of hello messages ALLOWED\_HELLO\_LOSS specifies the maximum number of periods of HELLO\_INTERVAL to wait without receiving a hello message before detecting a loss of connectivity to a neighbour. The recommended value for HELLO\_INTERVAL is one second and for ALLOWED\_HELLO\_LOSS is two. In other words, if a hello message is not received from a neighbour within two seconds of the last message, a loss of connectivity to that neighbour is determined.

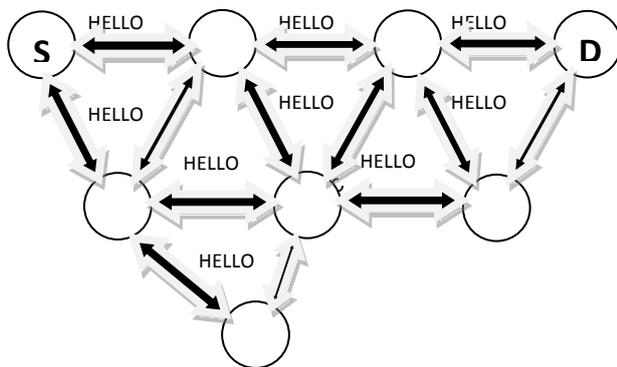


Fig 1. Broadcast of Hello messages

### B. Route Discovery

When a source needs to send packets to a destination, it first must determine a path for communication. The source node begins route discovery by broadcasting a route request (RREQ) message containing the IP address of the destination. When an intermediate node receives the RREQ, it records the reverse route toward the source and checks whether it has a route to the destination. If a route to the destination is not known, the intermediate node rebroadcasts the RREQ. When the destination or an intermediate node with recent information about a route to the destination, receives the RREQ, a route reply (RREP) is generated. The RREP is unicast back to the source using the reverse route created by the RREQ. For example, in this figure two nodes have recent information about the destination because hello messages are being used. These two nodes unicast a RREP to the source. As the RREP propagates toward the source, a *forward route* to the destination is created at each intermediate hop. When a RREP reaches the source, the source records the route to the destination and begins sending data packets to the destination along the discovered path. If more than one RREP is received by the source, the route with the lowest hop count to the destination is selected.

### C. Route Maintenance

When a link breaks along an active path, the node upstream of the break detects the break and creates a route error (RERR) message. The RERR message lists all destinations that are now unreachable, due to the link break. The node then sends the RERR message toward the source. Each intermediate hop deletes any broken routes and forwards the RERR packet toward the source. When the source receives the RERR packet, it determines whether it still needs the route to the destination. If so, the source creates a RREQ and begins the route discovery process again.

## III. TIME ANALYSIS OF LINK CONNECTIVITY

The basic principle behind the local link connectivity [1] shows that when a node moves out of range, its intermediate nodes will detect its link failure before the transmission of packet. Traditional scheme updates its routing information through the intermediate nodes immediately after detecting a link failure. But there is no need of update until its intermediate node is in need to transmit a packet. During, multiple hello packets are broadcasting.

$T_d$  - Time for detection of unavailable link

$T_w$  - Time gap between the link failure is detected and the link is needed

The average  $T_d$  is given as

$$T_d = (\text{ALLOWED\_HELLO\_LOSS} * 0.5) * \text{HELLO\_INTERVAL} \quad (1)$$

Also if a node moves beyond the coverage area where its neighbouring nodes are not in active communication and still broadcasting hello packets continuously, the battery of the nodes get drained. To overcome this, those hello packets should be suppressed by finding an exact hello interval.

### A. Design of dynamic hello interval

Let us consider an event interval whose transmission of packets occurs between two senders and its neighbour. Two possible situations emerge when the neighbour node moves out of range. (1) The sender is in need to transfer packet. (2) The sender is not in need to transfer. Case (1) causes a link failure. In case (2), there is no need for updating the routing information. To overcome the link failure, the sender must have the routing information before forwarding a packet. The HELLO\_INTERVAL should be set by referring to the event interval to reduce the risk of packet transmission through unavailable route. Analyze the distribution of packet transmission of 50 nodes. We find many event intervals are less than a constant HELLO\_INTERVAL (1sec) and the hello interval is not changed, if it is less than constant HELLO\_INTERVAL. The Cumulative Distributed Function of 50 nodes is shown in the figure. In that, all the traffics follows an exponential distribution where  $x > 1$ .

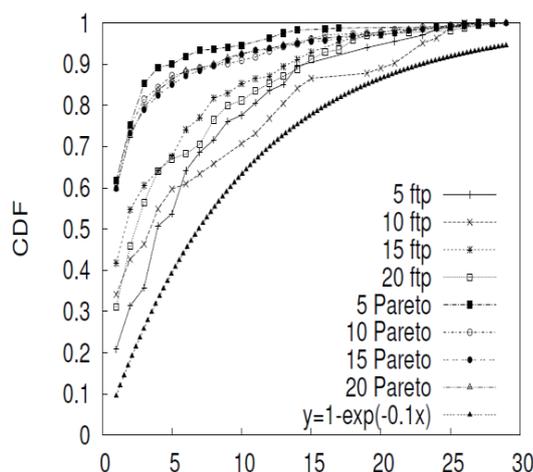


Fig 2. Distribution of event interval

The cdf of  $x$  can be written as  $F(x, \beta) = 1 - e^{-x/\beta}$  where  $x > 1$  (2) By using exponential distribution, we can predict the probability that an interval is larger than a given interval. This result provides an upper bound. So the probability that an event occurs after a given interval can be neglected. This analysis can be used to adjust the hello interval conservatively. Traditional hello protocol uses a default the probability of failure of detection of an unavailable link

( $P$ ) which can be varied with respect to  $\beta$ . Here we use  $PFD$  as a fixed value ( $=P$ ) and make it into a variable. Thus the hello interval can be adjusted with respect to the event interval dynamically. We can equate  $F(x,\beta) = PFD$ .

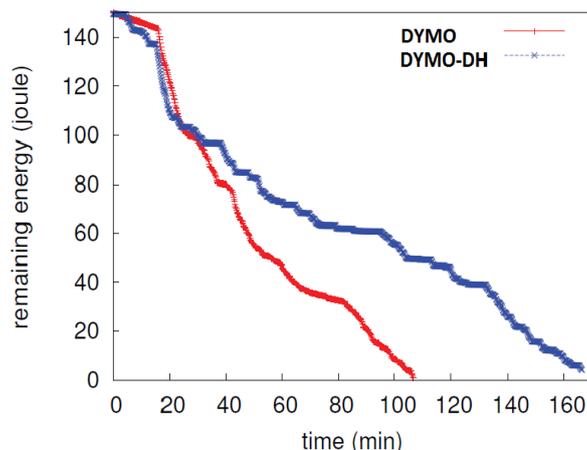
Therefore,  $1 - e^{-x/\beta} = PFD$  (3)

It can be written as  $x = -\beta \ln(1 - PFD)$  (4)

As the  $\beta$  increases, the event interval also increases. We can use this  $\beta$  to estimate the correct hello interval to minimize the unwanted hello packets.  $PFD$  will be smaller in equation (4) since it is the upper bound of an exponential distribution.

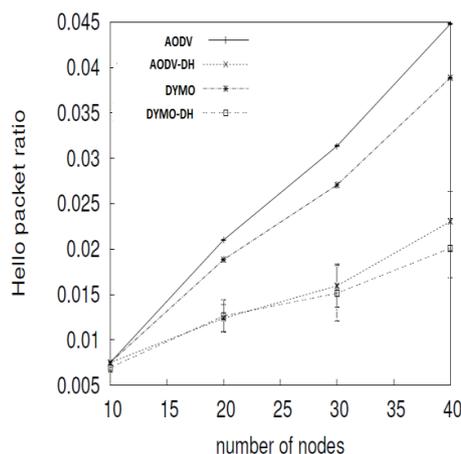
#### IV.EVALUTION

Our proposed scheme modified the AODV and DYMO. We call this as AODV with dynamic hello (AODV-DH) and DYMO with dynamic hello (DYMO-DH). This section depicts the effect of the proposed scheme [8] on energy use, throughput, Hello packet ratio within various parameters such as node intensity, mobile speed and node density.



**Fig 3. Energy comparison**

Figure 3 shows the comparison of energy consumption between DYMO and DYMO-DH. Each node initially has 150 joules. After certain time, DYMO consumes energy quickly. Since the proposed scheme reduces the number of inessential hello messages, energy is saved due to less transmissions and receptions.



**Fig 4. Hello packet overhead**

Figure 4 shows the Hello packet ratio. As the number of nodes increases, the hello packet ratio also increases since the number of hello packet broadcasters and the number of received packets increases.

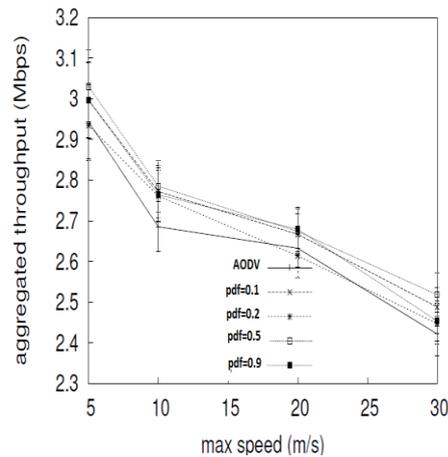


Fig 5. Throughput

Figure 5 depicts the throughput for various speeds and  $P$ . Since the number of affected links is small, high  $PFD$  does not make any major changes in throughput. Successive packets will be forwarded through a valid link. Due to quick reset to a default value, there is no throughput drop compared to the existing scheme. Thus high throughput can be obtained.

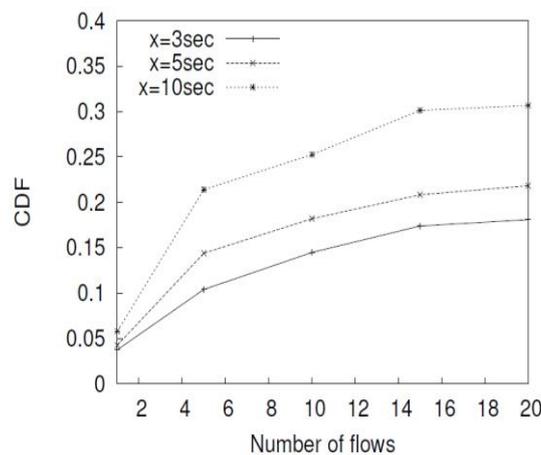


Fig 6. Cummulative distribution

Figure 6 shows the comparison of cdf over number of flows. The cdf of active nodes during  $x$  seconds are shown. When the number of flows is 10 and  $x=3$  sec, 85% of nodes do not involve in communication in the recent 3 sec. This reveals that large number of nodes use longer hello interval.

## V. CONCLUSION

This paper aims at reducing the energy consumption by restraining the inessential hello messages. Considering the event interval, the hello interval can be increased without reducing the probability of link failure detection, which reduces the delay and hidden energy consumption.

## REFERENCES

- [1] Ian D. Chakeres, Elizabeth M. Belding-Royer, "The Utility of Hello Messages for Determining Link Connectivity" in Proc. April 2003 Wireless Personal Multimedia Communications, 2003. The 5th International Symposium, vol. 4, pp. 245-260.
- [2] E. Belding-Royer and S. D. C. Perkins, "Ad hoc on-demand distance vector (AODV) routing," Proc. 2nd IEEE Workshop on Mobile Computer Systems and Applications, July 2003, vol 5, pp, 368-410.
- [3] C. Gomez, M. Catalan, X. Mantecon, J. Paradells, and A. Calveras, "Evaluating performance of real ad-hoc networks using AODV with hello message mechanism for maintaining local connectivity," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 2, pp. 1327–1331, May 2005.
- [4] V. C. Giruka and M. Singhal, "Hello protocols for ad-hoc networks: overhead and accuracy tradeoffs," in *Proc. Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks*, vol 1, pp. 354–361, January 2012.
- [5] C.-M. Chao, J.-P. Sheu, and I.-C. Chou, "An adaptive quorum-based energy conserving protocol for IEEE 802.11 ad hoc networks," *IEEE Trans. Mobile Computing*, vol. 5, no. 5, pp. 560–570, May 2006.
- [6] R. Oliveira, M. Luis, L. Bernardo, R. Dinis, and P. Pinto, "The impact of node's mobility on link-detection based on routing hello messages," in *Proc. IEEE Wireless Communications and Networking Conference*, vol 2, pp.1–6, June 2010.
- [7] A. Pathak, A. Jindal, Y. C. Hu, and S. P. Midkiff, "What is keeping my phone awake? Characterizing and detecting no-sleep energy bugs in smartphone apps," in *Proc. International Conference on Mobile Systems, Applications, and Services*, vol 7, pp. 267–280, July 2012.
- [8] Teresa Alberó-Alberó, Salvador Santonja-Climent, Víctor-M.Sempere-Payá, Jordi Mataix-Oltra, "AODV Performance Evaluation and Proposal of Parameters Modification for Multimedia Traffic on Wireless Ad hoc Networks", *IEEE Communications Magazine*, vol. 45, Iss. 4, pp 78--86. (2007) .
- [9] Ian D. Chakeres and Elizabeth M. Belding-Royer. "The Utility of Hello Messages for Determining Link Connectivity". Proceedings of the 5<sup>th</sup> International Symposium on Wireless Personal Multimedia Communications (WPMC) 2002, Honolulu, Hawaii, October 2002.