

Study of Impact of Hello Interval on Network Parameters (Throughput and Routing overhead) in AODV using NS-2

Gangu Vijay Kumar

Department of Computer Science and Engineering
AITAM, Tekkali, AP, India.

Pantula Muralidhar

Department of Computer Science Engineering
AITAM, Tekkali, AP, India.

Abstract:

Ad-hoc On-Demand Distance Vector, a routing protocol for ad hoc mobile networks with large numbers of mobile nodes. The protocol's algorithm creates routes between nodes only when the routes are requested by the source nodes, giving the network the flexibility to allow nodes to enter and leave the network at will. Routes remain active only as long as data packets are traveling along the paths from the source to the destination. Every node sends hello packet to its neighbor node in transmitting data packets. Depending on the number of packets may increase in the network overhead. This paper studies, Impact on Network overhead and throughput by varying Hello_Interval in AODV. We have used ns2 simulator in analyzing the results.

Keywords:

AODV; HELLO PACKETS; NETWORK LOAD; NS2 SIMULATOR.

I. INTRODUCTION TO MANETS:

A mobile ad-hoc network is a collection of mobile nodes forming an ad-hoc network without the assistance of any centralized infrastructure. These networks introduced a new art of network establishment and can be well suited for an environment where either the infrastructure is lost or where deployment of infrastructure is not very cost effective. The popular IEEE 802.11 "WI-FI" protocol [1] is capable of providing ad-hoc network facilities at low level, when no access point is available. However in this case, the nodes are limited to send and receive information but do not route anything across the network. Mobile ad-hoc networks can operate in a standalone fashion or could possibly be connected to a larger network such as the Internet. Mobile ad-hoc networks can turn the dream of getting connected "anywhere and at any time" [2] into reality.

Typical application examples include a disaster recovery or a military operation. Not bound to specific situations, these networks may equally show better performance in other places. It is an interconnected collection of wireless nodes where there is no networking infrastructure in the form of base stations, devices do not need to be within each other's communication range to communicate, the end-users devices also act as routers, nodes can enter and leave over time, data packets are forwarded by intermediate nodes to their final destination.

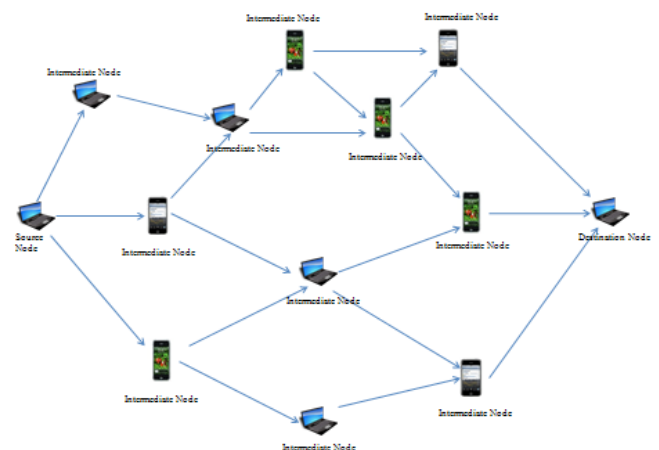


Figure 1: Mobile Ad-hoc Network

II.TAXONOMY OF ROUTING PROTOCOLS:

Taxonomy of routing protocols in mobile ad hoc network can be done in many ways; the routing protocols can be categorized as Proactive (Table Driven), Reactive (on-demand) and Hybrid depending on the network structure.

A.Proactive Routing Protocol:

The routing information about all the nodes is build and maintained by the proactive protocols. The proactive routing protocols are independent of whether or not the route is needed.

Control messages are transmitted with periodical intervals. Even if there is no data flow still control messages are transmitted. Because of these control messages proactive routing protocols are not bandwidth efficient. There are many advantages and disadvantages of proactive routing protocols. One of its advantages is that the nodes can easily get routing information and it easily starts a session.

The disadvantages are, too much data kept by the nodes for route maintenance, when there is a particular link failure its reform is too slow. Proactive routing protocols mandate that nodes in a MANET should keep track of routes to all possible destinations so that when a packets needs to be forwarded, the route is already known and can be routed. Proactive routing protocols experience minimal delay as it will need to select the route from the routing table [3].

B. Reactive Routing Protocol:

Reactive routing protocols are called on-demand routing protocols so these routing protocols are called when they are needed and the routes are built. These routes can be acquired by sending route requests through the network. Disadvantage of this algorithm is that it offers high latency in searching a network.

Reactive Routing protocols employ a lazy approach whereby nodes only discover routes to destinations on demand. Reactive routing protocols consume less bandwidth than proactive routing protocols but the delay to determine route can be significantly high and experience a long delay for discovering a route to a destination [3].

III. AODV:

Ad Hoc On-Demand Distance Vector, a routing protocol for ad hoc mobile networks with large numbers of mobile nodes. The protocol's algorithm creates routes between nodes only when the routes are requested by the source nodes, giving the network the flexibility to allow nodes to enter and leave the network at will. Routes remain active only as long as data packets are traveling along the paths from the source to the destination. AODV supports both unicast and multicast [4].

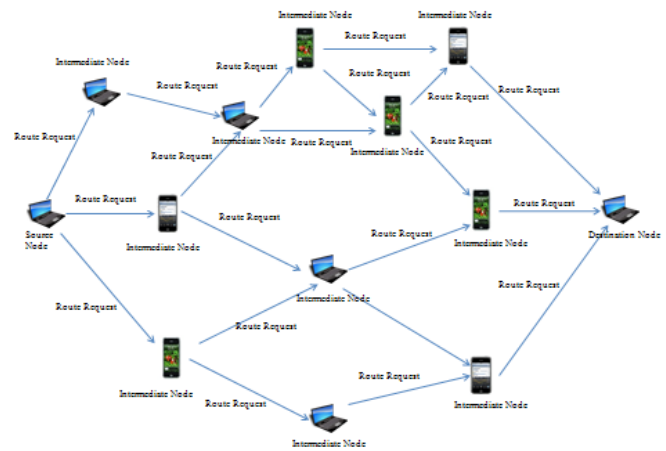


Figure 2: Route Request from Source to Destination

A. Unicast AODV :

AODV builds routes using a Route Request / Route Reply query cycle. When a source node desires a route to a destination for which it does not already have a route, it broadcasts a route request (RREQ) packet across the network. Nodes receiving this packet update their information for the source node and set up backwards pointers to the source node in the route tables. In addition to the source node's IP address, current sequence number, and broadcast ID, the RREQ also contains the most recent sequence number for the destination of which the source node is aware. A node receiving the RREQ may send a route reply (RREP) if it is either the destination or if it has a route to the destination with corresponding sequence number greater than or equal to that contained in the RREQ [5]. If this is the case, it unicasts a RREP back to the source.

Otherwise, it rebroadcasts the RREQ. Nodes keep track of the RREQ's source IP address and broadcast ID. If they receive a RREQ which they have already processed, they discard the RREQ and do not forward it. As the RREP propagates back to the source, nodes set up forward pointers to the destination. Once the source node receives the RREP, it may begin to forward data packets to the destination. If the source later receives a RREP containing a greater sequence number or contains the same sequence number with a smaller hop count, it may update its routing information for that destination and begin using the better route. As long as the route remains active, it will continue to be maintained. A route is considered active as long as there are data packets periodically travelling from the source to the destination along that path.

Once the source stops sending data packets, the links will time out and eventually be deleted from the intermediate node routing tables. If a link break occurs while the route is active, the node upstream of the break propagates a route error (RERR) message to the source node to inform it of the now unreachable destination(s). After receiving the RERR [5], if the source node still desires the route, it can reinitiate route discovery.

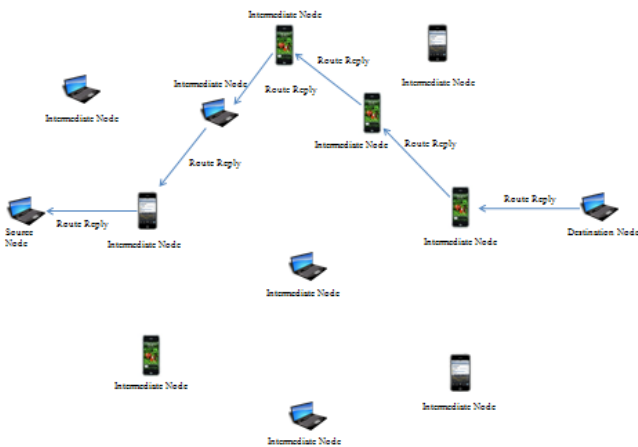


Figure 3: Route Reply from Destination to Source

B.Multicast AODV:

A node wishing to join a multicast group broadcasts a RREQ with the destination IP address set to that of the multicast group and with the 'J'(join) flag set to indicate that it would like to join the group. Any node receiving this RREQ that is a member of the multicast tree that has a fresh enough sequence number for the multicast group may send a RREP [5]. As the RREPs propagate back to the source, the nodes forwarding the message set up pointers in their multicast route tables. As the source node receives the RREPs, it keeps track of the route with the freshest sequence number, and beyond that the smallest hop count to the next multicast group member. After the specified discovery period, the source node unicasts a Multicast Activation (MACT) message to its selected next hop. This message serves the purpose of activating the route. A node that does not receive this message that had set up a multicast route pointer will timeout and delete the pointer. If the node receiving the MACT was not already a part of the multicast tree, it will also have been keeping track of the best route from the RREPs it received. Hence it must also unicast a MACT to its next hop, and so on until a node that was previously a member of the multicast tree is reached.

IV.HELLO PACKETS:

HELLO Packets are used by routing protocols to check the network connectivity. HELLO Packet is a special packet consisting of initial parameters that are sent out periodically from the nodes to establish and confirm network. The source node will broadcast HELLO Packets only when there is a need to send data packets to the destination that decrease the routing load or overhead. HELLO packets are used as a keep alive mechanism. If a node failed to receive a HELLO Packet for few time intervals from its neighbor, it means that the node is no longer in a communication range of a neighbor node[11]. HELLO Packets is used to discover the neighbors and exchange the initial parameters (source address and sequence numbers). Hello packets ensure bidirectional communication.

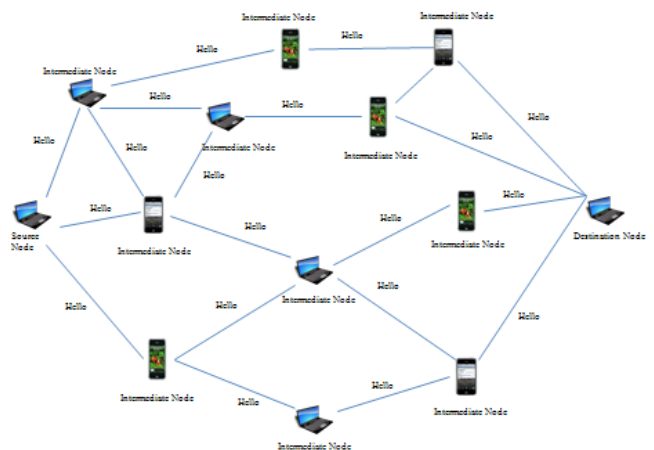


Figure 4: Hello Packet Broadcast

A.Hello Interval:

Hello_Interval specifies maximum time interval between the transmissions of hello messages [6].

B.Hello Loss:

Hello_Loss specifies maximum number of periods of Hello_Interval to wait w/o receiving a hello message before detecting a loss of connectivity to a neighboring [7].

V.NETWORK PARAMETERS:

Efficiency of the network can be computed based the network parameters.

A.Throughput:

Throughput is the rate of successful message delivery over a communication channel. This data may be delivered over a physical or logical link, or pass through a certain network node. The throughput is usually measured in bits per second (bit/s or bps), and sometimes in data packets per second or data packets per time slot [8].

B.Routing Overhead:

Overhead is any combination of excess or indirect computation time, memory, bandwidth, or other resources that are required to attain a particular goal [9].

VI.SIMULATION WITH NS-2

The Simulation of AODV was done by using discrete event simulator called NS2.The Simulation Scenario was given below:

Table 1. Parameters used in NS-2

| Constant Parameters | Values |
|---------------------|-----------|
| Simulation Time | 300 sec |
| Number of Nodes | 11 |
| Transmission Rate | 1Mb |
| Band width | 1Mb |
| Traffic Type | FTP |
| Packet Size | 512 bytes |
| Terrain Region | 500X500 |
| Propagation Model | Shadowing |
| Queue Length | 30 |
| Queue Type | Droptail |

A.Modification of CPP code in Ns2:

Initially the Hello Packets are deactivated in AODV. To activate the Hello Packets in AODV we have to comment the following two statements in aodv.cc file and recompile the aodv.cc file.

```

#ifndef AODV_LINK_LAYER_DETECTION
    htimer.handle((Event*) 0);
    ntimer.handle((Event*) 0);
#endif // LINK LAYER DETECTION

```

After successful compilation of aodv.cc, file executes the TCL program and you will find Hello Packets in trace file.

B.Tables:

Table.2 Throughput values by varying Hello_Interval

| Hello Interval | Throughput |
|----------------|------------|
| 1 | 560.09 |
| 2 | 558.34 |
| 3 | 540.97 |
| 4 | 555.57 |
| 5 | 551.02 |
| 6 | 529.73 |
| 7 | 544.5 |
| 8 | 537.64 |
| 9 | 534.31 |
| 10 | 550.45 |



Figure 5. Graph represents throughput values for the values of Hello_Interval

Table 3. Network Overhead values for Hello_Interval

| <i>Hello Interval</i> | <i>Overload</i> |
|-----------------------|-----------------|
| 1 | 0.102 |
| 2 | 0.052 |
| 3 | 0.033 |
| 4 | 0.024 |
| 5 | 0.021 |
| 6 | 0.019 |
| 7 | 0.015 |
| 8 | 0.014 |
| 9 | 0.014 |
| 10 | 0.010 |

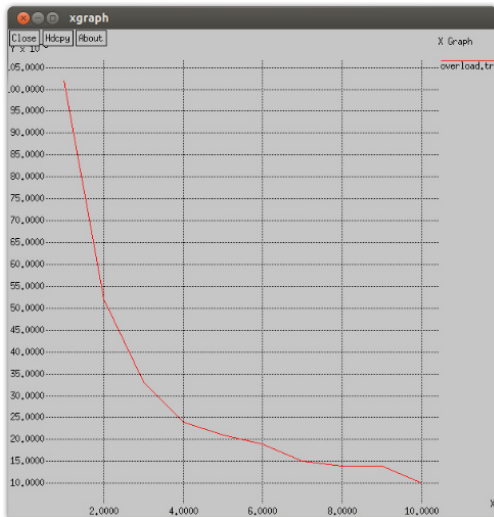


Figure 6: Graph represents the values of Network Overhead for Hello_Interval

C.ANALYSIS:

According to Table.1 the Throughput follows zigzag mechanism i.e., when Hello_Interval increases the throughput decreases at first from 560.09 kbps and there is a sudden raise in the value of throughput at Hello_Interval values 4,7 and 10. When we observe Table.2 network overhead decreases with increase in the values of Hello_Interval.

VII.CONCLUSION:

From obtained tables and graphs, we observe that the network overhead decreases with increase in the Hello_Interval and Throughput appears as zigzag with Hello_interval increase because of failure in detection of link establishment between neighbor nodes.

A network must result high throughput and low network overhead. The future work can extend with the concept optimizing the network with dynamically sending the Hello Packets.

References:

[1]http://en.wikipedia.org/wiki/Mobile_ad_hoc_network.

[2]Handbook of Mobile AD HOC Networks for Mobility Models by Radhika Ranjan Roy.

[3]Mehran Abolhasan, Tadeusz Wysocki, Eryk Dutkiewicz “A review of routing protocols for mobile ad hoc networks” in www.elsevier.com/locate/adhoc ,2004,pp. 1-22.

[4]<http://www.webopedia.com/TERM/A/AODV.html>.

[5]<http://moment.cs.ucsb.edu/AODV/aodv.html>.

[6]http://www.juniper.net/techpubs/en_US/junos12.3/topics/reference/configuration-statement/hello-interval-edit-protocols-ospf.html.

[7]<http://tools.ietf.org/wg/manet/draft-ietf-manet-cbrp-spec/draft-ietf-manet-cbrp-spec-01-from-00.wdiff.html>.

[8]<http://en.wikipedia.org/wiki/Throughput>.

[9][http://en.wikipedia.org/wiki/Overhead_\(computing\)](http://en.wikipedia.org/wiki/Overhead_(computing))..

[10]<http://www.isi.edu/nsnam/ns/>

[11]Krunal Patel, Tejas Vasavada, Mayur Vegad “Effect of Hello Message on Performance of Ad-hoc On-demand Routing Protocol” in National Conference on Recent Trends in Engineering & Technology.