Ad Hoc On-Demand Distance Vector Routing protocol and the effects of flooding mechanism on its performance

Devendra Singh Chauhan[1]*, Jitendra K. Mishra[2], Nagesh Salimath[3]

[1,3] Department of Computer Science, Madhyanchal Professional University, Bhopal-462044

[2] Department of Electronics and Communication Enginnering, Madhyanchal Professional University, Bhopal-462044

Email: devendra.chouhan2005@gmail.com

Abstract - Throughout the past few years, there have been a significant increase in the number of wireless communication devices. Since a growing number of users desire to be able to connect with one another at any time and from any location, without depending on any centralised access point or current infrastructure, this has led to new kinds of technological requirements. An ad hoc network is made up of numerous mobile nodes working together to create such a network. Ad hoc networks do not require any existing infrastructure because each node serves as both a host and a router. One of the created protocols that allows routing with constantly changing topologies is the ad-hoc on demand distance vector routing protocol (AODV). Because AODV is reactive, it only creates routes when those routes are actually required. Although it attempts to increase the overall bandwidth available by limiting the use of any periodic ads, it uses heavy flooding of messages when determining routes. The effectiveness and precision of the deployed routing protocols have come under scrutiny due to the growing popularity of these on-the-fly networks. This study examines the AODV protocol's scalability and performance in both small and large networks.

Keywords - AODV, Routing protocol, flooding mechanism

- X -

JUSTIFICATION FOR THE DECISION TO DEVELOP A NEW ALGORITHM

There are various broadcast schemes that most popular ad hoc networks including AODV use to transmit messages. The most widely used is the "simple flooding" scheme in which the source node of the network produces a route request packet. This packet is then distributed to all the further network nodes impulsively. Since the distribution is not monitored, there seem to be several repetitive transmissions which are not required. As a result, both the packet collision as well as channel contention in the network increase exponentially in a small amount of time. This process of increase is also known as a broadcast storm. This phenomenon is undoubtedly disadvantageous because the power is high and at the same time the network and transmission become less potent. Therefore, the aim is always to be least in protocol overhead by reducing the flooding. This will also decrease routing load and eventually the power consumption.

AODV PROTOCOL

Ad hoc On-demand Distance Vector (AODV) [1-2] is a reactive routing protocol that creates a pathway starting from the source node to the required destination. A reactive routing protocol means that the transmission only occurs when the source node needs the data packets to be transferred forward. Keeping up with the latest knowledge in all the routes of the network means high communication overload for every node. Therefore, 'reactive' routing protocol in mobile ad hoc networks becomes very advantageous. Broadly, two procedures complete the working of AODV, namely, Route discovery and Route maintenance. When the destination node calls for information, the source node needs to forward the information. However, an authentic route is not fixed, therefore, the source node uses the simple flooding method [3], thus sending the route request packet to all the possible nodes. As a result, it gains knowledge on which route is the most efficient and uses it. This process is called route discovery. Route maintenance on the other hand is keeping the linkage of transfer intact by detecting breaks and fixing them with the help of route error packets (RERRS).

ROUTE REQUEST MECHANISM (RREQ)

Route Request (RREQ), Route Error (RERR) and Route Reply (RREP) are the three types of messages broadcasted in AODV [4]. The storage of the message takes place under concrete status information. These messages are stored because they do not directly go to the destination. instead, the interested destinations call for the messages and are then sent forward. The RREQ packet contains three identifiers, namely (i)Source identifiers, (ii)broadcast identifier and (iii)destination identifier. Then are two sequence numbers stored, namely,(i) the source sequence number and(ii) the destination sequence number. Apart from these, the RREQ also contains time-to-live.

Information related to the destination such as its serial number contained in the entry present in the routing table. This information is constantly updated to keep it in the latest form. For example, the serial number discussed above, also known as the destination sequence number, is updated every time some new information is gained by the node about the destination. This information is provided by RREQ, RREP or RERR. Mathematically, N denotes every node in the network, m is the number of nodes and sq(N_i) denotes the sequential number of an arbitrary node denoted by i.

The sequence number of the destination node is incremented when the following situations occur:

The sequence number is increased prior to the beginning of the new route search. The reason for this increase is that it prevents any conflict that may occur with already-defined routes to the source.

$$sq(N_i) = sq(N_i) + 1 \quad (1)$$

The sequence number has to be changed before it sends a new RREP. This number has to the changed to the highest of its present sequence number $sq(N_i)$ and the destination sequence number $s\quad(N_j)$ in the RREQ packet.

$$\text{if } sq(N_i) \le sq(N_j) \text{ then}$$

$$sq(N_i) = sq(N_j)\ i,j{=}1\ldots m \quad (2)$$

$$\text{if } sq(N_j) \le sq(N_i) \text{ then}$$

$$sq(N_j) = sq(N_i)\ i,j{=}1\ldots m \quad (3)$$

A destination node can increment the sequence number sq(N_j) by one if one of the three following statements is true:

$$seq(N_i) = seq(N_j) \ + 1\begin{cases} Ni \neq Nj \\ New\ seq\ (Nj) \\ No\ valid\ path \end{cases} \quad (4)$$

The routing table entries and lists of adjacent nodes

A node does the process of finding justifiable entries when a destination requires packets of information. This takes place in three cases when a path needs to be altered for a destination, there is a need to create a new path altogether, and if the neighbor sends a control packet (routing packet). The creation of a new record in the table occurs when the record is absent. The sequence number is directly related to the information which the control packet possesses.

The sequential number is updated in the following cases:

Mathematically speaking,

$$sq(N_i)' > sq(N_i) \text{ then } sq(N_i)' = sq(N_i), \quad (5)$$

when it receives a new number. Here, i is an arbitrary node, sq(N_i) represents the sequence number of i and sq(N_i)' is the new node. This occurs when the value of the sequential number is higher than that in the routing table.

When the value of sequential number and the value in the table are identical. If the value given in the table is still larger than the sum of hops (required steps). That is:

$$\textstyle\sum_{j=i+1}^{k} hopC(Ni,Nj) < \sum_{j=i+1}^{i} hopC(Ni,Nj), k \le l \le m \quad (6)$$

When the sequential number is unknown, then

$$sq(N_i) = sq(N_i) + 1 \quad (7)$$

Two things are simultaneously stored. First, the valid routes in the routing table and second, a list of precursors. In other words, intermediate nodes help in taking the packets along with the route. This helps the node to notify the precursors in case the route link is lost.

ROUTE REQUEST QUERY

Whenever a destination has a route request, an RREQ packet is distributed by the node. This occurs in two cases if there is an invalid or inactive path, or in advance when there is an unknown destination. The routing table copies the destination sequential number value and is stored in a routing table along with its association with the last number received. As soon as the RREQs send a message, the exact number is increased.

For a destination node j, If $sq(N_j)$ is its sequence number, then:

$$sq(N_j) = sq(N_j) + n_{priority} \qquad (8)$$

where $n_{priority}$ is the priority number of the received RREQ message.

Expect for the last $RREQ_{ID,}$ which the current node uses, all the other $RREQ_{ID}$ field is increased from a value of 1. A single and separate $RREQ_{ID}$ is maintained by every node. This $RREQ_{ID}$ is used by the source node. For PATH_DISOCVERY_TIME, the IP address of the RREQ source is also taken by the source node. When a node has already sent a response for an RREQ message, they have to be prevented from responding to the same RREQ message again. For this, the PATH_DISCOVERY_TIME procedure of the RREQ mechanism is deployed.

$$\text{If } (RREQ_{ID})_i = 1 \text{ then}$$

$$(RREQ_{ID})_{n+1} = (RREQ_{ID})_n + 1, \text{ n=i,..1, } 1 \leq n \qquad (9)$$

where i represents the value of the current node. Current node is calculated as it will be used along with the $RREQ_{ID}$ field.

The source and the destination communicate in a twofold manner. This implies that for the communication to be successful, there ought to be a route to the source for the destination also. Thus, an intermediate node creates an RREP along with information about how to reach the source.

The RREP packet has the following main parts as shown in figure 1:

Source Identifier	Destination Number* (Destination IP Address)	Destination Sequence Number	Hop Count	Lifetime

Figure 1: Sample of RREP packet.

*Destination Number is the same as the Destination IP address

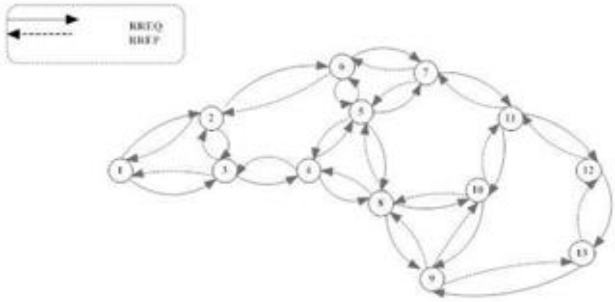

Figure 2: A typical AODV network of nodes

Two routes, first the forward route for the RREQ packet and the second, a route back to the source created by every intermediate note. To create the reverse route, the next hop information has to be imprinted in its routing table. This reverse route also helps the RREQ by allowing it to have a path to send an RREP to the source. This RREP is sent by the RREQ when it finds either an intermediate node or the destination node. However, the intermediate route has to have a valid route. To determine this, the sequence number of the intermediate node and the destination sequence number are compared with each other. All the nodes save the next hop information in the routeing table. This is done so that every node that forwards an RREP also makes a forward route for the destination. There are several nodes between the source and destination which do not need to have information about other nodes. They only require the information of the next hop node. Other than all the processes discussed above, route maintenance is another process that comes under AODV. This process is done after route discovery and the use of the route chosen. Once a route is in use, the node in between the source and destination creates and maintains a list of all the neighboring one-hops. This list is maintained using “hello” packets which are exchanged regularly to keep the list up-to-date. One example of the use of this process is when a route stops working. This can happen in case of a dead battery, in which case the network is informed by an activated timer which informs the network about the route expiry. In case of a link breakage, the routing agent becomes aware of an active route and thus an RERR is generated to send to the appropriate nodes. As soon as the source node gets the RERR packet, a new route discovery process is deployed to find a new route without any breakage.

For route discovery, AODV employs a straightforward flooding technique. In this technique, every node on proximity to the source node receives a message sent by it. All the nodes check for the same message if it has been sent prior or not. If so, the the message is discarded, otherwise the same process of broadcasting repeats. This procedure keeps on until the message reaches every node. Since this process is very repetitive, it consumes a lot of batter and at the same time so many broadcasts increase the traffic. A probabilistic message forwarding system (a forwarding technique that uses a probability to choose the number of nodes to send the messages) may be used to reduce the routing message overhead and, consequently, the power consumption of AODV. By decreasing redundant broadcasting from the nodes with a specific probability, this may be accomplished. The forwarding probability is the key element in this approach.

FLOODING ON AODV PROTOCOL

The protocol employs flooding [5], sometimes known as blind flooding, to find a path from a source to a destination node. To do this, the source node broadcasts the RREQ packet to all one-hop nodes. In order to confirm the target node and the quickest path to it, the RREP packet is subsequently transmitted utilizing the reverse route after the destination node has been located. If a route cannot be found, an RRER packet is issued to inform the source node that there is no practical way to reach the destination node. All three packets are together referred to as control packets. The flooding technique results in [6] overhead because control packets are constantly rebroadcast, which congests the network and utilizes more energy.

Numerous research projects [6]–[9] have been conducted to find solutions for the issues brought on by the flooding process. The goal of all of these research projects has been to provide methods for selecting a group of nodes with minimal forwarding overhead. According to the (Enhanced Ad hoc On demand Distance Vector routing) EAODV in [6] presented by Shobha and Rajanikanth, nodes that are moving quickly may have more recent routes than nodes that are moving slowly. This is possible because the EAODV employs two mobility requirements. By using the neighbouring nodes to broadcast the RREQ in accordance with their mobility rate and recent involvement, this method eliminates the usage of flooding mechanisms.

The authors in [7] provide an effective on-demand routing strategy that makes use of directional flooding (DF) and is appropriate for MANETs. By employing a constrained directional flooding methodology, this method interferes with the route discovery process and lowers the quantity of route request (RREQ) packets broadcast.

Another strategy for effective flooding has been put forth by Karthikeyan et al. in [8], and it involves a density-based flooding mechanism that makes sure that control packets are delivered from a source node to every node in the network with the least amount of MAC load, overhead, and power use possible.

A cluster-based approach was also put out by Karthikeyan et al. [9] in an effort to simplify the flooding broadcasting problem. The method notifies nodes of the least time for each transmission and adjusts to the topology dynamically. It promises to transfer the messages in a certain amount of time. The programme attempts to resolve any delay latencies and therefore avoids the flooding's negative impacts by using information about the numerous nodes that are situated at the same location.

CONCLUSION

The rationale behind selecting the AODV routing protocol for further development over other existing protocols has been discussed in this chapter. The way the AODV routing protocols operate and how the flooding mechanism impacts overhead have also been discussed in depth. energy use, MAC load, and routing load. The two suggested algorithms, the AODV EXT and AODV EXT BP, are described in the following chapters.

REFERENCES

[1] L. Xia, Z. Liu, Y. Chang, P. Sun, " An Improved AODV Routing Protocol Based on the Congestion Control and Routing Repair Mechanism", International Conference on Communications and Mobile Computing, pp. 259-262, Vol. 2, Jan. 2009, China.

[2] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," IETF Mobile Ad Hoc Networking Working Group INTERNET DRAFT, RFC 3561,July 2003, http://www.ietf.org/rfc/rfc3561.txt. Experimental RFC, retrieved in October 2007.

[3] "AODV Routing", http://www.ietf.org/rfc3561.txt, 2003

[4] M. J. N.Karthikeyan, Dr.V.Palanisamy, and Dr.K.Duraiswamy, "Performance Comparison of Broadcasting methods in Mobile Ad hoc Network" Int. Jour. of Future Generation Communication and Networking, Vol. 2, No. 2, pp. 47-58, June 2009

[5] G. S. Tomar, M. Dixit, S. Verma, " AODV Routing Protocol with Selective Flooding", International Conference of Soft Computing and Pattern Recognition, pp. 682-689, Malaca, December 2009.

[6] K. R. Shbha, K. Rajanikanth, "Adaptive AODV Routing Protocol for Mobile Ad Hoc Networks", Int. Journal of Ad Hoc, Sensor & Ubiquitus Computing (IJASUC), Vol. 2, No. 1, March 2011.

[7] K. Dong-Won, L. Ahn-Ngoc, C. You-Ze, T. Keong, L. In-Soo, "An Efficient On- Demand Routing Approach with Directional Flooding

for Wireless Mesh Networks", IEEE Journal, Vol. 12, Iss. 1, pp. 67-73, February 2011.

[8] N. Karthikeyan, V. Palanisamy, K. Duraiswamy, "Optimum Density Based Model for Probabilistic Flooding Protocol in Mobile Ad hoc Network", European Journal of Scientific Research, Vol. 39, No. 4, pp. 577-588, 2010.

[9] N. Karthikeyan, V. Palanisamy, K. Duraiswamy, "Reducing Broadcast Overhead Using Clustering Based Broadcast Mechanism in Mobile Ad hoc Network", Journal of Computer Science, pp. 548-556, Vol. 5, No. 8, 2009.

Corresponding Author

Devendra Singh Chauhan*

Department of Computer Science, Madhyanchal Professional University, Bhopal-462044