OPTIMIZATION OF PATH RECOVERY TIME IN MULTIPATH AODV

Charushila Raskar [1], Shoba Krishnan [2]

[1,2] *Department of Electronics and Telecom. Engineering,*

Mumbai University, Mumbai, (India)

ABSTRACT

Ad-hoc network means a collection of wireless mobile nodes moving in some area without any backbone infrastructure such as Access point and the devices connected to it for communication. As the nodes moves randomly in any area the topology is also changing arbitrarily. The network disconnection occurs frequently. This project proposes a difference of the AODV protocol, which selects interior nodes using path's accumulation information, and then recognized multi paths based on interior-node's information. Proposed algorithm accumulates the reset time to build the new path and also reduces the reset control messages by using the interior-node information when the network gets disconnected.

Keywords: Multipath, Ad-hoc

I INTRODUCTION

Routing support for mobile hosts is presently being formulated as mobile IP technology when the mobile agent moves from its home network to a foreign (visited) network, the mobile agent tells a home agent on the home network to which foreign agent their packets should be forwarded. Routing protocol for ad-hoc networks can be group according to the way in which nodes obtain routing information and according to the type of information they use to compute preferred paths. In terms of the way in which nodes obtain information, routing protocols have been categorized as on-demand and table-driven. In on-demand routing protocols, nodes maintain path information for only those destinations. Examples of this approach are AODV [2], DSR [3], and TORA [4]. In table-driven routing protocols, each node maintains path information for each known destination in the network and updates its routing-table entries as needed. Examples of table-driven algorithms based on distance vectors are the routing protocols of the DARPA packet radio network [5], DSDV [6], and WRP [6].

II PRESENT WORK

The various steps used in the recent routing algorithm are discussed below.

2.1 Ad-hoc on-demand distance vector

The Ad Hoc on Demand Distance Vector (AODV) routing protocol builds on the DSDV algorithm. AODV is an improvement on DSDV because it typically minimizes the number of required broadcasts by creating routes on a demand basis, as opposed to maintaining a complete list of routes as in the DSDV algorithm. AODV classify as a pure on-demand route acquisition system, since nodes that are not on a selected path do not maintain routing information or participate in routing table exchanges. When a source node desires to send a message to some destination node and does not already have a valid route to that destination, it initiates a path discovery process to locate the other node.
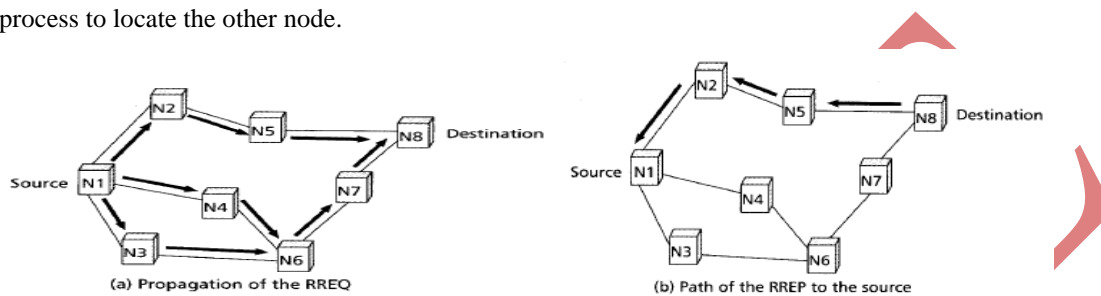
Fig. 1 Path in the AODV

Type	J	R	G	D	U	Reserved	Hop count
RREQ ID							
Destination IP Address							
Destination sequence Number							
Originator IP Address							
Originator sequence Number							

Fig. 2 RREQ Packet Format

It broadcasts a route request (RREQ) packet to its neighbors, which then forward the request to their neighbors, and so on, until either the destination or an intermediate node with a fresh enough routes to the destination is located. Figure 1(a) illustrates the propagation of the broadcast RREQs across the network. AODV utilizes destination sequence numbers to ensure all routes are loop free and contain the most recent route information. The source node includes in the RREQ the most recent sequence number it has for the destination. Intermediate nodes can reply to the RREQ only if they have a route to the destination whose corresponding destination sequence number is greater than or equal to that contained in the RREQ. During the process of forwarding the RREQ, intermediate nodes record in their route tables the address of the neighbor from which the first copy of the broadcast packet is received, thereby establishing a reverse path. If additional copies of the same RREQ are later received, these packets are discarded. Once the RREQ reaches the destination or an intermediate node with a fresh enough route, the destination intermediate node responds by unicasting a route reply (RREP) packet back to the neighbor from which it first received the RREQ(Fig.4). As the RREP is routed back along the reverse

path, nodes along this path set up forward route entries in their route tables which point to the node from which the RREP came. These forward route entries indicate the active forward route.

III PROTOCOL CONCEPT

Algorithm reduces the packets which carry the control information in a network like RREQ, RRER, RREP packets. It selects the Core nodes to forward the RREQ packets to establish the connection to the destination node. This reduces the additional load on the source node and also saves the lot of time to send the error message to the source node. It means that the Core-node reinitiate the path without knowing to the source node. If link break occur the core node sends the data through the secondary path.

3.1 Route Discovery Process

AODV drops the packet if it gets the duplicate packets in the network but this algorithm flood the packet instead of dropping it. Before sending the packet to the next node the control packets pass through the node, nodes ID is attached to packets payload. Figure shows the packet send to the next node.

Type	J	R	G	D	U	Reserved	Hop count
RREQ ID							
Destination IP Address							
Destination sequence Number							
Originator IP Address							
Originator sequence Number							
Accumulation Path							

Fig. 3 Modified RREQ Packet Format

In modified format, RREQ packet stores the information for data transfer route. If link break occur, the network removes the loop by using stored node ID's information. From this accumulated node information, algorithm selects the CORE-nodes that forward a RREQ packet for multi-path. As shown in Figure 4, two packets are sent at the same time towards a destination node. Those nodes store path in Accumulation PATH variable of RREQ packet. Using the information accumulated in the path of two packets, the destination node D selected node 5 and node 12 as CORE-nodes.
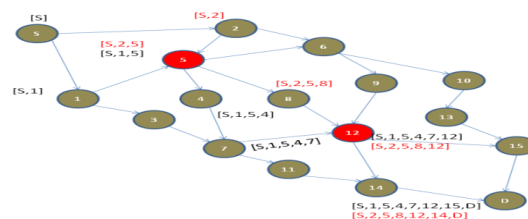

Fig. 4 Path creations from source to destination

As shown in figure there are multiple path from the source node to the destination node. At the end the multiple paths has found to the destination node having same source node, then it selects the two path with min length among them. Path which is having minimum length among two acts as a primary path and other is secondary path. Reply packet (RREP) is send through primary as well as secondary path to the sender node by unicast.

3.2 Route Recovery Process

When the link break occurs to the path AODV does the route recovery with the help of control packet. Upstream node detects that it is Core-node. If the upstream node is Core-node it starts the transmission through the secondary path and restarts the data transfer. If the upstream node is a Core-node then the control packet is transfer to the Core-node with the error (RRER) message to the Core-node. It tells the Core-node about the link break. Then the Core-Node detects the previous broken link's information and Core-node selects node of the secondary paths and restarts the data transfer. So the algorithm does not require flooding the control message so to reduce the route recovery latency.
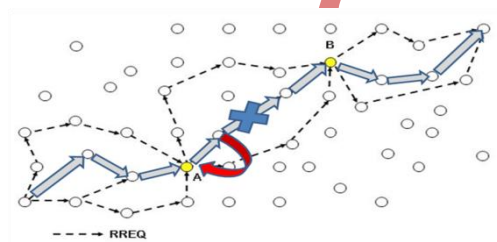
⇒ Primary path   ------ Secondary Path   ⤺ - RERR

Fig. 6 Route maintenance process

If links break successively, there are no alternate paths between Core-nodes. In figure 7 Route recovery process must restart in divided sections by CORE-node A and CORE-node B. Also, there will be CORE-nodes (C, C') between the CORE-node A and CORE-node B. And there are many CORE-nodes in the section where a link break frequently occurs and the length of the path is close. Therefore, proposed algorithm unlike the AODV, it doesn't need to flood control messages so reduces the route recovery latency. If a link breaks successively, there are no alternate paths between CORE-nodes. Route recovery process must restart in
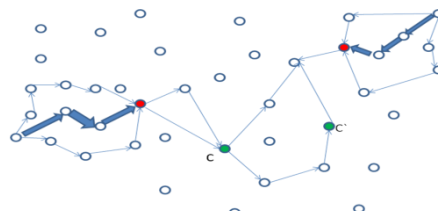

→ = Primary Path → Second Path

Fig. 7 Route Discovery after Link-Break.

divided sections by CORE-node A and CORE-node B. Also, there will be CORE-nodes (C, C') between the CORE-node A and CORE-node B. And there are many CORE-nodes in the section where a link break frequently occurs and the length of the path is close.

IV SIMULATION SET UP & EXPERIMENTAL RESULTS

Figure 8 shows the comparison graph of AODV vs multipath AODV. Graph has interval vs path_establish_time for both AODV and multipath AODV. Graph has different values as No. of packet send in one unit interval and the time require for path establishment.
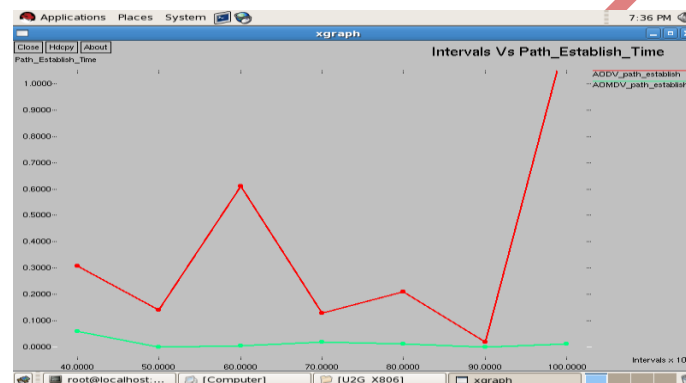
Fig. 8 Path establish time AODV vs AOMDV

Multipath AODV with core node require less time to the route request, route reply and route error packet as compare to the AODV protocol. The role of Core-Node in multipath AODV is to maintain route on be-half of source node. If link break occurs in between intermediate links, node which is connected to link failure node does not get reply message from next node so the link breakage or error message sends to the upstream node/Core-Node instead of source node. But if it happens with AODV node protocol sends the link failure message to source node only. With this AOMDV performs better while doing the link recovery process. After recovery of link failure core node sends the same data from the secondary path which has been store in the beginning of AOMDV route establishment.

V CONCLUSION

Path in the wireless network could change frequently as the time progress. As the number of nodes increased in the ad-hoc network flow of the data may increase rapidly. Multipath AODV helps to minimize the number of control packets in the network as the Core node i.e. intermediate node helps to provides alternate path to the destination when a link is broken. This will reduce the recovery latency because source node does not need to reinitiate a route discovery process.

REFERENCES

[1] C. Perkins, "*RFC 3561 : Ad hoc On-Demand Distance Vector Routing*", July, 2003

[2] D.Johnson and D.Maltz, "*Protocols or Adaptive Wireles and Mobile Networking*", IEEE pers, Vol.3, No.1, February, 1996

[3] D.B.Johnson, "*Dynamic Source Routing in Ad-Hoc Wireless Networks in Mobile Computing*", 1996

[4] J.Jubin, "*The DARPA Packet Radio Network Protocols*", Proceedings of IEEE, Vol.75, NO.1, January 1987

[5] C.Perkinsand P.Bhagwat "*Highly Dynamic Destination–Sequenced Distance-Vector Routing (DSDV) for Mobile Computers*", Proc. ACM 94, London, UK, October 1986

[6] S.Murthy, "*An Efficient Routing Protocol for Wireless Networks*," ACM Mobile Networks and Applocations Journal, Specail issue on Routing in Mobile Communication Networks, 1996.