Study and Design of New Reactive Routing Protocol Advance AODV for Mobile Ad hoc Networks

Anurag Porwal[1], B.L.Pal[2], Rohit Maheshwari[3], Gaurav Kakhani[4]
[1]Associate Lecturer at Mewar University, Gangrar
[2]Asst.Professor at Mewar University,Gangrar
[3]Asst.Professor at Mewar University,Gangrar
[4] Associate Lecturer at Mewar University,Gangrar

Abstract

A mobile ad-hoc network is a collection of mobile nodes forming an ad-hoc network without the assistance of any centralized structures. These networks shows a new way of network establishment and these are well suited for an environment where either the infrastructure is lost or where deploy an infrastructure is not very cost effective. We have presented the overview of Ad hoc network routing protocols. In this paper we worked to solve the problem of intermediate route building in Ad hoc on demand distance vector routing protocol (AODV) and proposed scheme that enhances the performance of AODV protocol. The scheme proposed by us is Advance-Ad hoc on demand distance vector (AAODV) routing protocol. It consists of the use of Ad-hoc On demand Distance Vector with Backup routing (AODV-BR) and concept of local recovery with limited TTL value in case of failure of local recovery in first attempt i.e. if the neighboring node of the node that find the link break do not have path to destination in its alternate routing table.

I. INTRODUCTION

Ad hoc network can be considered as a special type of wireless mesh networks which is a collection of mobile wireless nodes formed without any infrastructure or any standard services.

Mobile Ad hoc Networks (MANETs) [1] are decentralized and mobile nodes act as router and also as host. Mobile nodes can transmit the packets to the nodes which are in its proximity. If a mobile node has to send the packet to other mobile nodes which are out of its range then the nodes within its range forwards packets to the next hop until packets reaches intended destination. Thus MANETs are also called mobile multihop wireless networks. MANETs can be setup between few nodes or can be extended by connecting to fixed network.

A Mobile ad hoc network is illustrated in Figure 1.1 consists of three wireless mobile nodes A, B and C. Transmission range of a node represented by dotted circle. Mobile node A is not within the transmission range of C and vice versa. If A wants to establish communication with C. Node B which in the transmission range of A and C forwards the packets so that A and C are able to communicate each other successfully.

The fundamental difference between fixed networks and MANET is that the computers in a MANET are mobile. Due to the mobility of these nodes there are some characteristics that are only applicable to MANET.

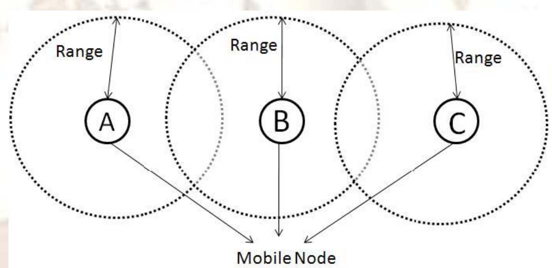

Figure 1.1: A Mobile Ad hoc network

1.2 The Protocol Stack

In this section the protocol stack for mobile ad hoc networks is described. This gives a comprehensive picture of, and helps to better understand, mobile ad hoc networks. Figure 1.2, shows the protocol stack which consists of five layers: physical layer, data link layer, network layer, transport layer and application layer. It has similarities to the TCP/IP protocol suite. As can be seen the OSI layers for session, presentation and application are merged into one section, the application layer.

On the left of Figure 1.2, the OSI model is shown. It is a layered framework for the design of network systems that allows for communication across all types of computer systems.

In the middle of the Figure 1.2, the TCP/IP suite is illustrated. Because it was designed before the OSI model, the layers in the TCP/IP suite do not correspond exactly to the OSI layers. The lower four layers are the same but the fifth layer in the TCP/IP suite (the application layer) is equivalent to the combined session, presentation and application layers of the OSI model.

On the right, the MANET protocol stack-which is similar to the TCP/IP suite-is shown. The main difference between these two protocols stacks lies in the network layer. Mobile nodes use an ad hoc routing protocol to route packets. In the physical and data link layer, mobile nodes run protocols that have been designed for wireless channels. Some options are the IEEE standard for wireless LANs, IEEE 802.11, the European ETSI standard for a high-speed wireless LAN, HIPERLAN 2, and finally an industry approach toward wireless personal area networks, i.e. wireless LANs at an even smaller range, Bluetooth.

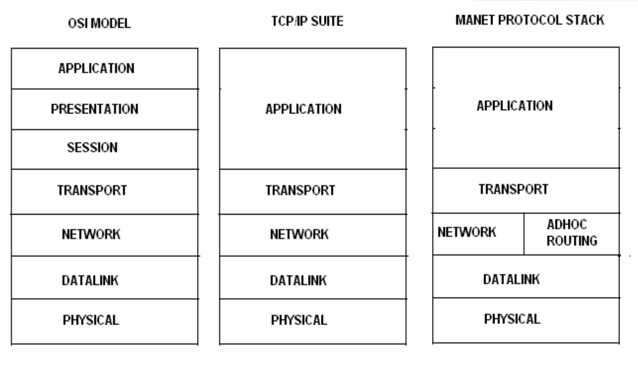

Figure 1.2: Protocol stack of OSI, TCP/IP and MANET

This paper focuses on ad hoc routing which is handled by the network layer. The network layer is divided into two parts: Network and Ad Hoc Routing. The protocol used in the network part is Internet Protocol (IP) and the protocols which can be used in the ad hoc routing part are Destination Sequenced Distance Vector (DSDV), or Ad hoc On Demand Distance Vector (AODV) etc .

Applications of MANET

The emerging field of mobile and nomadic computing, with its current emphasis on mobile IP operation, should gradually broaden and require highly-adaptive mobile networking technology to effectively manage multihop, ad hoc network clusters which can operate autonomously or, more than likely, be attached at some point(s) to the fixed Internet. Some applications of MANET technology could include industrial and commercial applications involving cooperative mobile data exchange. In addition, mesh-based mobile networks can be operated as robust, inexpensive alternatives or enhancements to cell-based mobile network infrastructures.

Characteristics of MANETs

(1) Dynamic topologies: Nodes are free to move arbitrarily; thus, the network topology--which is typically multihop--may change randomly and rapidly at unpredictable times, and may consist of both bidirectional and unidirectional links.

2) *Bandwidth*-constrained, variable capacity links: Wireless links will continue to have significantly lower capacity than their hardwired counterparts.

3) ***Energy-constrained operation***: Some or all of the nodes in a MANET may rely on batteries or other exhaustible means for their energy. For these nodes, the most important system design criteria for optimization may be energy conservation.

4) ***Limited physical security****:* Mobile wireless networks are generally more prone to physical security threats than are fixed-cable nets. The increased possibility of eavesdropping, spoofing, and denial-of-service attacks should be carefully considered.

MANET Routing Protocol Performance Issues

The following is a list of quantitative metrics that can be used to assess the performance of any routing protocol.

(1) *Average end-to-end delay of data packets* —It is caused by buffering during route discovery latency, queuing at the interface queue, retransmission delays at the MAC, and propagation and transfer times. This metric describes the packet delivery time, the lower the end-to-end delay the better the application performance.

(2) *Throughput: This* is the measure of how soon an end user is able to receive data. It is determined as the ratio of total data received to required propagation time. A higher throughput will directly impact the user's perception of the quality of service (QoS).

3) *Packet delivery Fraction (PDF)* — it is the ratio of the data packets delivered to the destinations to those generated by the CBR sources. The PDF shows how successful a protocol performs delivering packets from source to destination. The higher for the value give use the better results.

4) *Data Packet Loss (Packet Loss)* — Mobility-related packet loss may occur at both the network layer and the MAC layer. A packet is dropped in two cases: the buffer is full when the packet needs to be buffered and the time that the packet has been buffered exceeds the limit.

II. OVERVIEW AD-HOC NETWORK MOBILE AD HOC ROUTING PROTOCOLS

Existing Ad Hoc Routing Protocols

Since the advent of Defense Advanced Research Projects Agency (DARPA) packet radio networks in the early 1970s [2], numerous protocols have been developed for ad hoc mobile networks. Such protocols must deal with the typical limitations of these networks, which include high power consumption, low bandwidth, and high error rates. As shown in Fig. 3.1, these routing protocols may generally be categorized as:

• Table-driven
• Source-initiated (demand-driven)

Table-Driven Routing Protocols

Table-driven routing protocols attempt to maintain consistent, up-to-date routing information from each node to every other node in the network. These protocols require each node to maintain one or more tables to store routing information, and they respond to changes in network topology by propagating updates throughout the network in order to maintain a consistent network view. The areas in which they differ are the number of necessary routing-related tables and the methods by which changes in network structure are broadcast. The following sections discuss some of the existing table-driven ad hoc routing protocols. Example of table-driven protocol is DSDV.

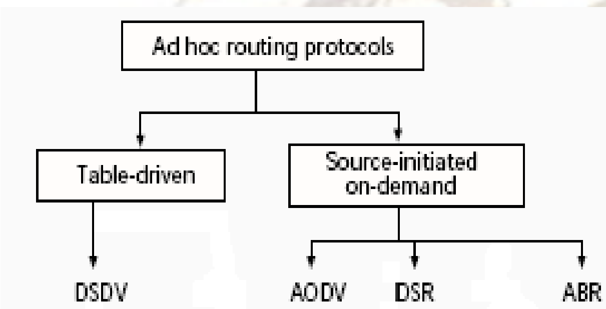

Fig 2.1: Types of Ad hoc routing protocols

Source-Initiated On-Demand Routing

A different approach from table-driven routing is source-initiated on-demand routing. This type of routing creates routes only when desired by the source node. When a node requires a route to a destination, it initiates a route discovery process within the network. This process is completed once a route is found or all possible route permutations have been examined. Once a route has been established, it is maintained by a route maintenance procedure until either the destination becomes inaccessible along every path from the source or until the route is no longer desired. Example of source –initiated on-demand protocol is AODV.

Ad hoc On demand Distance Vector (AODV) Routing Protocol

The Ad Hoc On-Demand Distance Vector (AODV) routing protocol described in [5] builds on the DSDV algorithm previously described. 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. The authors of AODV classify it 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 [5].

Routing Table in AODV

AODV maintains the following fields in its routing table for each routing table entry.

1. Destination IP Address
2. Destination Sequence Number
3. Valid Destination Sequence Number flag
4. Other state and routing flags (e.g., valid, invalid, repairable, being repaired)
5. Hop Count (number of hops needed to reach destination)
6. Next Hop
7. Network Interface
8. List of Precursors
9. Lifetime (expiration or deletion time of the route)

Sequence Numbers

Many distance vector routing protocols suffer from a condition called Count to infinity [6]. This problem can be solved in AODV by using sequence numbering scheme which is derived from DSDV. Each AODV node maintains a monotonically increasing sequence number which is independent of other nodes. In AODV sequence numbers represent the freshness of the routing information. Nodes increment its sequence number when it generates a new route request or when it generates a route reply. If a node gets multiple route replies for the destination then it will always selects the route to the destination with greatest destination sequence number. This ensures that selected route is the recent one. If destination sequence numbers of route replies are same then node will selects the route which has less number of hops to destination.

Routing Table Management

Routing table management determines whether a route is still active using primary parameters: source sequence numbers, destination sequence numbers, route request expiration timer and route caching timeout. The route request expiration timer is used to invalidate all the entries of those nodes that do not lie on the path from the source to destination. The expiration time depends on the size of network. The route caching timeout is the time beyond which a route is no longer considered to be valid. For each valid route maintained by a node as a routing table entry, the node also maintains a list of precursors that may be forwarding packets on this route. These precursors will receive notification from the node in the event of detection of the loss of the next hop link. The list of precursors in a routing table entry contains those neighboring nodes to which a route reply was generated or forwarded.

Every routing table entry contains the following information: Destination address, Next hop, Number of hops, Destination sequence number, Precursor list, expiration timer. With this information each node in AODV can determine whether its neighbor is considered active for the particular destination. The criterion for being active is determined if the neighbor originates or relays at least one packet for a destination within the most recent active route timeout period. This enables all active source nodes to become informed if a link along a path to destination breaks. Each time a route entry is used to transmit data, the expiration time is updated to the current time plus the active route timeout.

Message Types in AODV

In AODV there are four different message formats [7] they are:

1. Route Request (RREQ)
2. Route Reply (RREP)
3. Route Error (RERR)
4. Route Reply Acknowledgment (RREPACK)

Route Discovery In AODV

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. 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" route to the destination is located. Figure 2.2a illustrates the propagation of the broadcast RREQs across the network. AODV utilizes destination sequence numbers to ensure all routes are loop-free and con- contain the most recent route information. Each node maintains its own sequence number, as well as a broadcast ID. The broadcast ID is incremented for every RREQ the node initiates, and together with the node's IP address, uniquely identifies an RREQ. Along with its own sequence number and the broadcast ID, 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. 2.2b). 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. Associated with each route entry is a route timer which will cause the deletion of the entry if it is not used within the specified lifetime. Because the RREP is forwarded along the path established by the RREQ, AODV only supports the use of symmetric links.

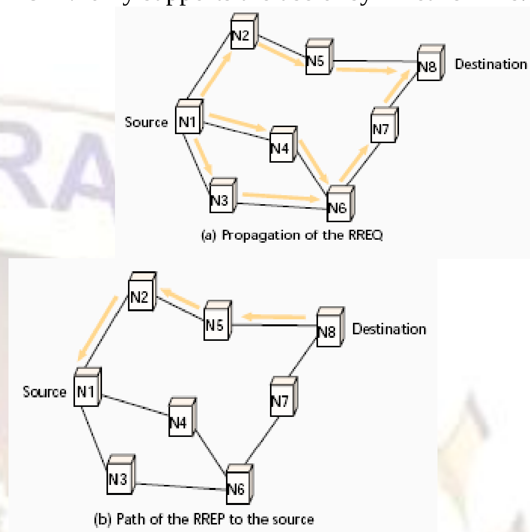

Figure 2.2 AODV route discoveries

Local Connectivity

In MANETs, links between the nodes can break due to node mobility, restricted range and capacity of wireless channel. A mechanism must exist for nodes to determine when a link to a neighbor along an active path is broken. One method for obtaining such connectivity information is by using Hello messages. Hello messages are locally broadcast RREPs that indicate the existence of the sending node. The time to live (TTL) of the RREP is set to one, so that only the node's immediate neighbors receive the message. The hello message includes the node's address, its current sequence number, and a lifetime for the link. A node generates Hello message for every HELLO INTERVAL. If a node does not receive hello messages from its neighbors during interval of ALLOWED HELLO LOSS * HELLO INTERVAL seconds, then it expires the routing table entry. All the nodes in the precursor list are notified about the link failure.

Route Maintenance

HELLO messages may be used to detect and monitor links to neighbors. In such case, each node broadcasts periodic HELLO messages to all its neighbors. When a broken link is detected, either by a MAC layer acknowledgment or by not receiving HELLO messages, the upstream node sends Route Error (RERR) message to all predecessor nodes that use the broken link to reach their respective destinations. The RERR packet is propagated towards the source and the route is deleted from the routing table.In Fig 2.3. Node B detects a link break and sends a RERR message to node A.

When a node receives a RERR, it first checks whether the node that sent the RERR is its next hop to any of the destinations listed in the RERR. If the sending node is the next hop to any of these destinations, the node invalidates these routes in its route table and then propagates the RERR back towards the source. The RERR continues to be forwarded in this manner until it is received by the source. Once the source receives the RERR, it can reinitiate route discovery if it still requires broken. Node B invalidates its route table entries for both nodes C and D (Fig 2.3), creates a RERR message listing these nodes, and sends the RERR upstream towards the source.

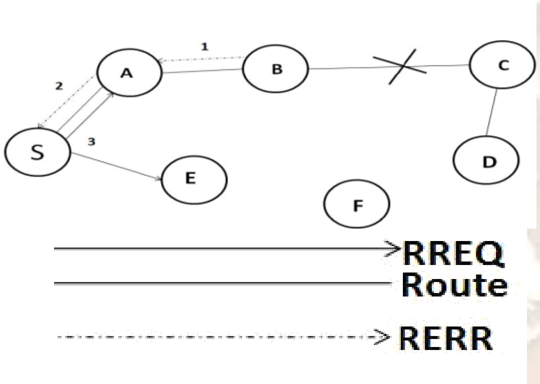

Fig. 3.2: Route Maintenance in AODV

Ad hoc on Demand Distance Vector-Backup Routing (AODV-BR)

AODV-BR [14] utilizes a mesh structure to provide multiple alternate paths to existing on-demand routing protocols without producing additional control messages. Having multiple alternate paths in ad hoc networks is beneficial because wireless networks are prone to route breaks resulting from node mobility, fading environment, signal interference, high error rate, and packet collisions. It is also important to generate multiple routes without propagating more control messages than when building only single route. Minimizing the number of packet transmissions is critical in ad hoc networks with limited bandwidth and shared wireless medium.

Route Construction in AODV-BR

AODV-BR is incorporated with reactive routing protocols that build routes on demand via a query and reply procedure. It uses the same RREQ structure as used by AODV protocol. When a source needs to initiate a data session to a destination but does not have any route information, it searches a route by flooding a ROUTE REQUEST (RREQ) packet. Each RREQ packet has a unique identifier so that nodes can detect and drop duplicate packets. An intermediate node, upon receiving a non-duplicate RREQ, records the previous hop and the source node information in its route table (i.e. backward learning). It then broadcasts the packet or sends back a ROUTE REPLY (RREP) packet to the source if it has a route to the destination. The destination node sends a RREP via the selected route when it receives the first RREQ or subsequent RREQs that traversed a better route (in AODV for instance, fresher or shorter route) than the previously replied route. The mesh structure and alternate paths are established during the route reply phase. We slightly modify the AODV protocol in this procedure. Taking advantage of the broadcast nature of wireless communications, a node promiscuously "overhears" packets that are transmitted by their neighboring nodes. From these packets, a node obtains alternate path information and becomes part of the mesh as follows. When a node that is not part of the route overhears a RREP packet not directed to itself transmit by a neighbor (on the primary route), it records that neighbor as the next hop to the destination in its alternate route table. A node may receive numerous RREPs for the same route if the node is within the radio propagation range of more than one intermediate node of the primary route. In this situation, the node chooses the best route among them and inserts it to the alternate route table. When the RREP packet reaches the source of the route, the primary route between the source and the destination is established and ready for use. Nodes that have an entry to the destination in their alternate route table are part of the mesh. The primary route and alternate routes together establish a mesh structure that looks similar to a fish bone (see Fig. 2.4).

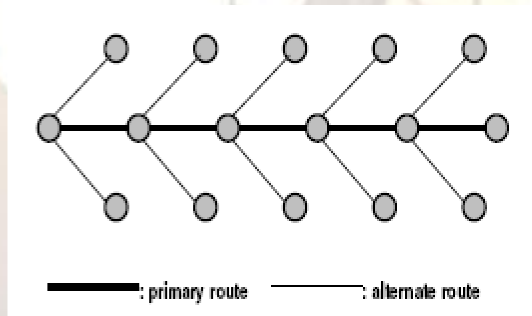

Fig 2.4. Multiple routes forming a fish bone structure

Route Maintenance and Mesh Routes

Data packets are delivered through the primary route unless there is a route disconnection. When a node detects a link break (for example, receives a link layer feedback signal from the MAC protocol, does not receive passive acknowledgments, does not receive hello packets for a certain period of time, etc.), it performs a one hop data broadcast to its immediate neighbors. The node specifies in the data header that the link is disconnected and thus the packet is candidate for "alternate routing." Upon receiving this packet, neighbor nodes that have an entry for the destination in their alternate route table, unicast the packet to their next hop node. Data packets therefore can be delivered through one or more alternate routes and are not dropped when route breaks occur. To prevent packets from tracing a loop, these mesh nodes forward the data packet only if the packet is not received from their next hop to the destination and is not a duplicate. The node that detected the link break also sends a ROUTE ERROR (RERR) packet to the source to initiate a route rediscovery. The reason for reconstructing a new route instead of continuously using the alternate paths is to build a fresh and optimal route that reflects the current network situation and topology. In AODV, a route is timed out when it is not used and updated for certain duration of time. AODV-BR uses the same technique for timing out alternate routes. Nodes that provide alternate paths overhear data packets and if the packet was transmitted by the next hop to the destination as indicated in their alternate route table, they update the path. If an alternate route is not updated during the timeout interval, the node removes the path from the table.

III. PROBLEM STATEMENT

Problem Description

In Ad hoc networks link break occurs frequently due to nodes mobility, greater error rates, interference of signals, fading environment etc. But an actual route break occurs due to mobility of nodes. Link breaks caused by other sources are factious. In ad hoc network On Demand routing protocols have one of the three choices to perform in case of link break. First, the source node will do nothing and gets timeout waiting for an acknowledgment from the destination. Then after timeout the source node may starts a new fresh route discovery cycle in case it wants further communication with the destination node. Second, the intermediate node which find the link break report the error to the source node by sending a route error (RERR) message immediately. The source node then again re-initiate a route discovery cycle for that destination if it required further communication with that destination .Third, some local recovery mechanism is used to bypass the link in error.

In multipath routing [13], multiple routes from source to destination are cached during a single route discovery cycle. In case of the occurrence of link break, any of the alternative routes is selected to forward the packets. The performance of multipath routing shows better utilization of network resources, but number of packet drops and delay is increased because alternative cached routes may become stale. On the other hand, a special route maintenance mechanism is used in a local recovery scheme, to repair the broken routes. In Ad-hoc On Demand Distance Vector (AODV) routing protocol [5], when a link break occurs, the upstream node decides either to propagate a route error (RERR) message back to the source node or to repair the route using limited TTL broadcast. This decision is based on the distance between the intermediate node (that find the link break) and the destination node. If the node that finds the link break is close to the destination, it performs local recovery by sending a route request (RREQ) message with limited time-to-live (TTL) value. Otherwise, RRER message is unicasted to the source node to give information about link break. The source node after receiving the RRER message may starts new route discovery cycle if required. After starting the local repair process, the intermediate node waits for a route discovery period. If the repair process fails, the node sends a RRER message back to the source node. Otherwise, the node updates its routing entry for that destination. But if local recovery is performed many times then there is a danger of using non-optimal route inspite of the existence of another optimal route.

These limitations of AODV motivated us to propose an efficient technique for route maintenance, Advance - Ad Hoc On-demand Distance Vector (AAODV), and Multipath Preemptive - Ad hoc On Demand Distance Vector (PM-AODV) which improves the performance of an existing on-demand routing protocols, specifically AODV.

IV. PROPOSED SOLUTION

4.1 Advance-Ad hoc On Demand Distance Vector (AAODV)

Our Proposed solution consists of the use of AODV-BR and concept of local recovery with limited TTL value in case of failure of local recovery in first attempt i.e. if the neighboring node of the node that find the link break do not have path to destination in its alternate routing table.

First we will discuss about the medications required to the existing AODV and later the approach in detail.

The changes made in RREQ packet include:

1. We have introduced a new ***flag k*** in RREQ packet, it indicates first look in alternate routing table if k is 1.

The changes made in RREP packet include:

1. ***Flag p*** inserted in the RREP packet. It indicates local recovery is performed.

The changes made in Routing table include:

1. A variable ***total*** is included in routing table that indicates total number of hops from source to destination.
2. A variable ***count*** is included whose value indicates number of times local recovery is performed, initially count=0.
3. A flag ***long*** is included in routing table whose value indicates hop metric increased in local recovery, initially long=0.

DEST	NEXT HOP	……	***TOTAL***	***COUNT***	***LONG***

Table 4.1: Routing table of AAODV

The proposed solution of the defined problem consists of following concept:

- Route construction mechanism: It is same as of AODV-BR so that in case of link break route can be find quickly during local recovery.
- Route maintenance mechanism: When an intermediate node finds a link break, it first calculates its distance from destination. If it is closer to the destination, it prepares a RREQ packet by setting TTL=1 and some other flags to query its neighbors for connectivity information. The neighboring node on receiving RREQ packet, search its alternate routing table for route to the required destination. If it finds route in its alternate routing table it then copy that entry in its main routing table and sends a RREP packet back to the node from which it received RREQ. Before sending a RREP it set the flag in RREP to indicate local recovery is performed. On receiving RREP the node updates its routing table and informs the source about this new route. If a node that finds link breaks do not get reply from any of its neighbor than it starts local route recovery with specified TTL to limit the area of search. If any node finds the alternative route to the destination within the timeout period it updates its routing table (Fig. 5.1). Each node also makes an entry in its routing table about number of hops to reach the destination. The intermediate node performs route recovery in two cases. First, if it is performing recovery first time. Second, if the number of hops obtained in previous local recovery is less or equal to the number of hops known before recovery and if this node is closer to the destination. If node that finds link break is closer to the source node then it do not perform local recovery and sends RERR packet to the source to initiate a route discovery cycle again. The reason for reconstructing a new route instead of performing local recovery again is to obtain a fresh and optimal route that reflects the current network situation and topology. If a node that finds link breaks do not get reply from any of its neighbor than it starts local route recovery with specified TTL to limit the area of search. If any node finds the alternative route to the destination within the timeout period it updates its routing table (Fig. 4.1)

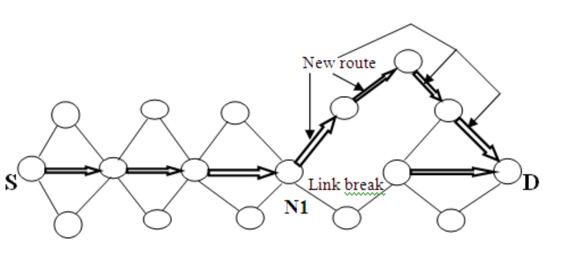

Fig. 4.1: New route in case of link break, when either neighbor is having any route to destination in its main routing table rather than in its alternate routing table or when recovery is performed with TTL=dest (can be tuned)

Route Maintenance Algorithm In AAODV

Data packets are send using the primary route unless the link breaks occurs. The operation after the intermediate node (say N1) has identified link break is presented below:

1. If((dist<total/2) && (count<5 || long==0))

// ***dist*** is the variable whose value is equal hope count field attribute in the routing table.

// ***total*** is routing table attribute that indicates total number of hops from source to destination.

// ***count*** is a field in routing table whose value indicates number of times local recovery is performed, initially count=0.

// ***long*** is a flag in routing table whose value indicates hop metric increased in local recovery, initially long=0.

2. {
N1 Sends RREQ with TTL=1 and by setting flag k of RREQ packet to 1;
}
3. Else {
goto step 17 ;
}
4. On receiving RREQ packet node do following:
If (flag k of RREQ ==1) goto step 5;

//we have introduced new flag k in RREQ packet, it indicates that first look in alternate routing table if k is1.

Else goto step 6;

5. The node looks up in its alternate route table. If it finds any node as its next hop to the destination it update its main routing table by making an entry to that destination. It then prepares RREP packet by setting its hop count field equal to the value of its dist field in its alternate route table. It than unicast it back to the node from which it receive RREQ. Goto step 7

6. The node looks up its main routing table to see whether it is having any next hop to that destination. If it finds any node as its next hop to the destination it then prepares RREP packet by setting its hop count field equal to number of hops from it to destination. It than unicast RREP packet back to the node from which it receive RREQ.

7. If RREP is received

8. { N1 updates its routing table with alternate route ;

9. N1 set count=count+1;

10. If(dist<n+1) // n is the value of hop count field in RREP

{ Set long=1;

}

11. Set dist1=dist; // dist1 is a local variable

Set dist=n+1;

Set total=total + (dist – dist1);

12. N1 make changes in received RREP as follows:

Set flag p=1;

// flag p indicates local recovery is performed p is a flag inserted in the RREP packet.

Set n=n+1; // increases hop count by one

After preparing RREP N1 send it towards source .N1 also prepares a special Gratuitous RREP for destination as follows:

{ Set flag G of RREP =1;

Hop count(n) = total;

Set flag p =1;

}

N1 now sends this Gratuitous RREP towards destination.

13. Each intermediate node on receiving RREP updates its route for that Destination and perform following action:

If (p==1)

{Set count=count+1;

}

Set dist1=dist; // dist1 is a local variable

Set dist=n+1;

Set total=total + (dist –dist1);

Set n=n+1; // n is the hop count value in RREP

14. Each intermediate node on receiving Gratuitous RREP updates its route for that Destination and perform following action:

If (p==1)

{

Set count=count+1 in its routing.

Set total = n // n is the hop count value in RREP.

}

15. Source node on receiving RREP update its route table and start sending data using new route.

}

16. Else

If(w==1)

// Node N1 sends RERR message to source node to start global route discovery.

Goto step 17

Else

N1 sends RREQ with TTL=dist (can betuned);

Set flag k=0;

w=1;

// w is a local variable it is used so that local recovery is performed only two times

Goto step 6;

17. Node N1 sends RERR message to source node to start global route discovery.

18. Stop

Performance Analysis

Various cases are considered for evaluating the performance of the proposed scheme and it is also compared with AODV [5], AODV-BR [14] and Bypass-AODV [15]:

Case 1: AAODV finds alternate route in case of link break and continue to use it until link break occurs again (as in Fig. 4.1). While in AODV-BR and Bypass-AODV alternate route is used only to forward buffered data. In AAODV source node perform route discovery less often than AODV, AODV-BR and Bypass-AODV. Hence AAODV reduces global flooding and increases bandwidth utilization compare to AODV, AODV-BR and Bypass-AODV.

Case 2: In AAODV there is no danger of using long route in case of frequent link breaks and performing more local discoveries. Local recovery is performed only if the previous route recovery results in equal or less number of hops. Otherwise source node perform route discovery.

Case 3: Data packet drops in AAODV during link break are less than AODV, AODV-BR and Bypass-AODV because it performs local route discovery twice (if it fails to find route in first attempt, with TTL=1) with two different TTL values. AODV simply drops data packets when routes are disconnected. AODV-BR also has some packet drops because alternate paths may also break as the primary route because of mobility.

Case 4: In AAODV the probability of finding alternate route in case of link break is more than any other on-demand routing protocol. Hence, AAODV is more reliable.

***Comparison of the AAODV and AODV Routing Protocol*:**

The simulation results are revealed in the following section in the form of line graphs. Graphs illustrate comparison between the protocols by varying different numbers of sources on the basis of the above-mentioned metrics as a function of pause time.

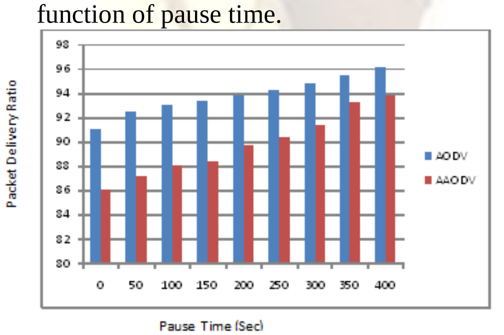

Fig 5.1 : Packet delivery fraction vs. Pause time for 40-node model with 40 sources

Above graph shows Packet delivery function gets increases as we increase the pause time AODV protocol's PDF is grater then AAODV Protocols packet delivery ratio.

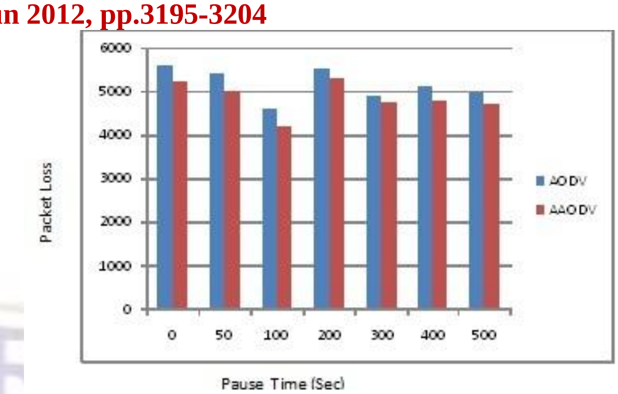

Fig5.2: Packet Loss vs. Pause time for 40-node model with 40 sources

Packet loss with respect to pause time is shown as pause time is increases packet loss varies according to pause time.

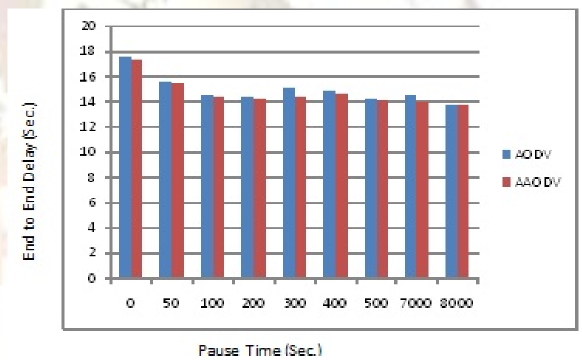

Fig 5.3: end to end delay vs. Pause time for 40-node model with 40 sources

The comparison show that AAODV Gives better performance then AODV and it reduce the route maintenance time when the link break is occurs so it improves the performance of AODV routing protocol.

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed two schemes AAODV and PM-AODV for route maintenance. These protocols are proposed to increase the performance of AODV routing protocol in case of link break.

AAODV scheme is proposed to improve the route maintenance phase of AODV routing protocol needed in case of link break. AAODV uses a combined strategy of AODV and AODV-BR and a new concept of repeating local route recovery next time only in case of getting equal or improve hop metric in previous local recovery. AAODV minimized the routing overhead because source node performs route discovery less often. Also by performing local recovery only in case of getting better metric in previous local discovery, we always get a fresh and optimal route that reflects the current network topology.

In future we will compare its performance with other routing protocols like TORA, DSR etc.

REFERENCES:

[1] S. Corson, J. Macker, RFC 2501 - Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, January 1999.

[2] J. Jubin and J. Tornow, “The DARPA Packet Radio Network Protocols,” Proc. IEEE, vol. 75, no. 1, 1987, pp. 21–32.

[3] C. E. Perkins and P. Bhagwat, “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers,” Comp. Commun.Rev., Oct. 1994, pp. 234–44.

[4] L. R. Ford Jr. and D. R. Fulkerson, Flows in Networks, Princeton Univ. Press, 1962.

[5] C.E. Perkins and E.M. Royer, “Ad-Hoc On Demand Distance Vector Routing,” Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA), New Orleans, LA, February 1999, pp. 90-100.

[6] A.Tanenbaun, Computer Networks, Fourth Edition.

[7] C. Perkins, E. BeldingRoyer and S. Das, Ad hoc On Demand Distance Vector (AODV) Routing, http://www.ietf.org/rfc/rfc3561.txt, July 2003.

[8] D. Kim, J. Garcia and K. Obraczka, “Routing Mechanisms for Mobile Ad Hoc Networks based on the Energy Drain Rate”, IEEE Transactions on Mobile Computing. Vol 2, no 2, 2003, pp.161-173

[9] WaveLAN/PCMCIA Card User’s Guide – Lucent Technologies.

[10] www.isi.edu/nsnam/ns/tutorial Marc Greis tutorial on ns2

[11] Matthias Transier “ Ns2 tutorial running simulations ”

[12] J. B. Andersen, T. S. Rappaport, and S. Yoshida. Propagation measurements and models for wireless communications channels. IEEE Communication Magazine, 33(1):42–49, Jan. 1995.

[13] A.Nasipuri,R. Castañeda, and S.R.Das.2001.Performance of multipath routing for on-demand protocols in mobile ad hoc networks. ACM/Baltzer Mobile Networks and Applications (MONET) Journal-6, 4, 339-349.

[14] Sung-Ju Lee and Mario Gerla 2000. AODV-BR: Backup Routing in Ad hoc Networks. Wireless Communications and Networking Conference, 2000. WCNC. 2000 IEEE, Vol. 3 (2000), pp. 1311-1316 vol.3.

[15] Cigdem, Sengul and Robin, Kravets. 2006. Bypass routing: An on-demand local recovery protocol for ad hoc networks. Ad Hoc Networks. 4, 3 (January 2006), 380-397.

[16] D. Johnson, D. Maltz, Y. Hu, and J. Jetcheva. The dynamic source routing protocol for mobile ad hoc networks. Internet Draft, Internet Engineering Task Force, Mar. 2001. http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-05.txt.

[17] S. S. Rappaport. Wireless Communication Systems. Prentice Hall, 1996.

[18] T. Goff, N. B. AbuGhazaleh, D. S. Phatak and R.Kahvecioglu, Preemptive Routing in Ad Hoc Networks, Proc. of ACM MobiCom, 2001.

[19] Preemptive Multipath—Adhoc on Demand Distance Vector Routing Protocol 2011 by manoj kumarr sing

AUTHORS:

Anurag Porwal is an Associate lecturer in Department of Computer Science and Information Technology at Mewar University, Chittorgarh (Rajasthan). He has completed his B.Tech.(I.T.) form Rajasthan University and M.Tech.(CSE) from Mewar University Chittorgrah **(**Raj.). His areas of interest are Ad-hoc Networks, Network Security and Cloud Computing.

B.L.Pal is an Asst. Professor in Department of Computer Science and Information Technology at Mewar University, Chittorgarh (Rajasthan) .He has completed his B.Tech. (IT***)*** from AAI_DU Allahabad U.P. (2002-06) and M.Tech. (SIT) from DAVV Indore. M.P. (2007-09). His research interests are in the field of GIS, Spatial Database Management System and N/W Securities.

Rohit Maheshwari is an Asst. Professor in Department of Computer Science and Information Technology at Mewar University, Chittorgarh (Rajasthan). He has completed his B.E. (Hons.) in I.T. form Rajasthan University (2005) and M.Tech. (CSE) from Rajasthan Technical University, Kota**.** His research interests are in the field of Network Security, Cloud Computing and Algorithms.

Gaurav Khakani is an Associate lecturer in Department of Computer Science and Information Technology at Mewar University, Chittorgarh (Rajasthan). He has completed his B.Tech. (I.T.) form Rajasthan University and M.Tech. (CSE) from Mewar University, Chittorgrah. His areas of interest are Ad-hoc Networks and Computer Networks.