# Improved QoS in Ad-Hoc On-Demand Routing Protocol for MANET

**Nima Manoharan[1] Cibile K Kanjirathumkal[2]**
[2]Professor
[1,2]Department of Electronics & Communication Engineering
[1]RIT, Kottayam [2]Govt. Engineering College, Kannur

*Abstract*— Mobile ad-hoc Networks (MANETS) is a collection of wireless nodes without any infrastructure support. Quality of service (QoS) provisioning in MANETs is very important in order to support real-time communications such as audio and video. But, provisioning of QoS over wireless networks is far more challenging than for wired networks because of variability of wireless links, node mobility, and lack of central coordination authority for QoS. Ad-hoc on demand distance Vector (AODV) algorithm is a quick and efficient routing protocol designed for MANETS. The major drawback of AODV is the shortage of the quality of service (QoS) provisions. This paper proposes modifications to AODV protocol to improve the quality of service by combining three different routing approaches namely adaptive hello messaging, traffic aware route discovery and local link failure recovery. The modified protocol provides better QoS than the conventional AODV protocol.

***Key words:*** MANET, AODV, QoS Improvement, Adaptive Hello Messaging, Traffic Aware Route Discovery, Local Link Failure Recovery

## I. INTRODUCTION

A temporary network, formed by a collection of wireless mobile hosts, without any centralized administration or standard support services can be called a mobile ad-hoc network (MANET) [1]. In this type of network each node acts as a host as well as a router with an arbitrary mobility. In the absence of a centralized administration the source and destination communicate through multiple hops. Continuously changing topology of MANET demands very efficient routing scheme and leads to evolution of various routing protocols.

The conventional AODV protocol is an example of on demand routing protocol that focuses on discovering the shortest path between two nodes [2]. It has characteristics like simplicity, low computational complexity and low processing overhead. The major drawback of the algorithm is the shortage of quality of service (QoS) provisions. In the conventional AODV routing protocol, network reliability and QoS are not considered for packet transmission to destination node. Incorporating QoS in MANETs is difficult due to its broadcast and dynamic nature.

The new multimedia applications (videoconferencing, VoIP, IP-TV etc) and real time applications require high throughput and reduced delays [8]. The protocols used by non real time applications, usually transmission control protocol (TCP), are tolerant to some amount of loss because of their re-transmission capabilities. Real time applications based on the user defined protocol (UDP) are significantly less tolerant to packet loss. AODV cannot perform well for real time applications which require certain QoS guarantees.

Given the growing importance of QoS in wireless networks, over the last few years, a number of works have been proposed to improve QoS in an ad-hoc wireless environment. AODV protocol is a comparatively mature on-demand routing protocol in mobile ad hoc networks. Hence most of the works proposed are extension to AODV protocol in spite of its limitations in term of QoS requirements. Many modifications have been suggested towards improvement in AODV performance to meet QoS challenges through focusing on bandwidth, packet delivery ratio, energy and overhead mechanism.

AODV backup routing (AODV-BR) discussed in [4] establishes multiple routes to destination. If the primary route breaks, alternate route can be initiated to carry out data transmission. The ad-hoc QoS on demand routing (AQOR) proposed in [5] deals with the bandwidth and end to end delay requirements. In QoS enabled AODV (QoS-AODV) in [6], the original AODV is extended by adding new fields including maximum delay extension and minimum bandwidth extension. In order to provide QoS, extensions can be added to these messages during the route discovery process. Quality of service AODV (QAODV) discussed in [7] is designed with a few modifications to traditional AODV protocol, QAODV focuses on two efficient route recovery mechanisms for QoS routing.

All the protocols discussed above perform well in certain environment, but not in all network situations and environment. So there is a lack of universality of application. Mixing different routing approaches to obtain desired performance is a common practice though choosing and combining them is a complex task. This proposed work makes use of an approach where modification employed in three stages of AODV routing protocol. This Provides significant QoS improvement in real time packet transmission. Also the proposed protocol results in improved performance with reduced complexity.

The paper is organized as follows: an overview to conventional AODV protocol is given in section II. The Proposed modifications in 3 stages of AODV routing protocol are included in Section III. This is followed by Section IV, where simulation environment needed for the experiments using network simulator is described. A comparison of performance analysis of modified protocol with conventional AODV for different QoS parameters is also covered in the same section. The paper is concluded in Section V.

## II. OVERVIEW OF AODV PROTOCOL

AODV algorithm enables dynamic, self-starting, multi hop routing between participating mobile nodes which can maintain an ad-hoc network [3]. AODV algorithm allows mobile nodes to obtain routes quickly for new destinations, and does not require nodes to maintain routes to destinations that are not currently participating in active communication. Further this scheme allows mobile nodes to respond to link breakages and changes in network topology in a timely manner.

 

Subsection A, B, C gives the three mechanisms currently used in the conventional method of AODV protocol.

### A. Route Discovery

By broadcasting a route request (RREQ) packet a route discovery process is initiated when a node requires communication with another node which has no route. RREQ packet contains source IP address, current sequence number of source, broadcast-ID, destination IP address, last sequence number of destination, and hop-count. Hop count is initially 0 and is incremented by each node as it forwards the RREQ towards the destination.

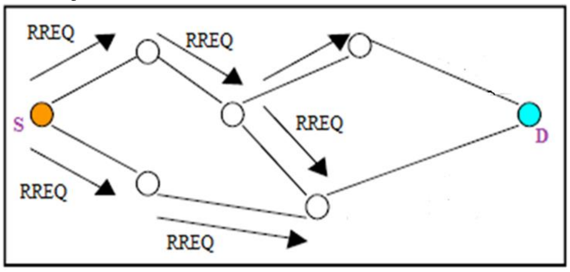


Fig. 1: Propagation of the RREQ from Source to Destination

The broadcast ID is incremented each time the source node initiates RREQ. the source node initiates RREQ. Sequence numbers are used to determine the timeliness of each data packet. An intermediate node upon receiving RREQ, first checks whether RREQ is received over a bi-directional link. Then the node compares with the already processed RREQ. If it is same, the packet is discarded, otherwise the node checks if it has a route entry for the destination. If it has a routing table entry for the destination then it replies to the source only if the destination sequence number in RREQ is greater than the destination sequence number in its route table. Otherwise it rebroadcasts the RREQ packet. Use of destination sequence number helps to select fresh routes and avoids looping in route [3].

When the destination node or an intermediate node with a route to the destination receives the RREQ, it creates the RREP and unicast the same towards the source node using the node from which it received the RREQ as the next hop.

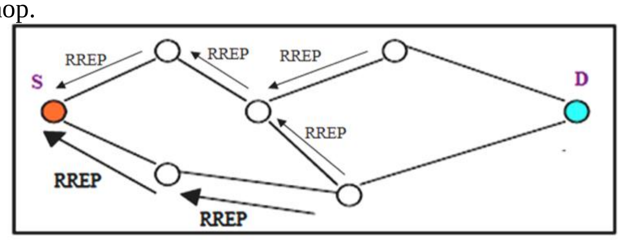


Fig. 2: Propagation of the RREP from Destination to Source

It is possible that the source node will receive more than one RREP from its neighbors. In this case it uses first RREP that it receives and upon the reception of another reply it checks if the later packet contains a greater destination sequence number or if it has a smaller hop count. Greater destination number indicates a fresher route and smaller hop count means shortest path. In this case it updates the route entry with new values, otherwise reply packet is discarded.

The following flow chart illustrates the route discovery process in conventional AODV.

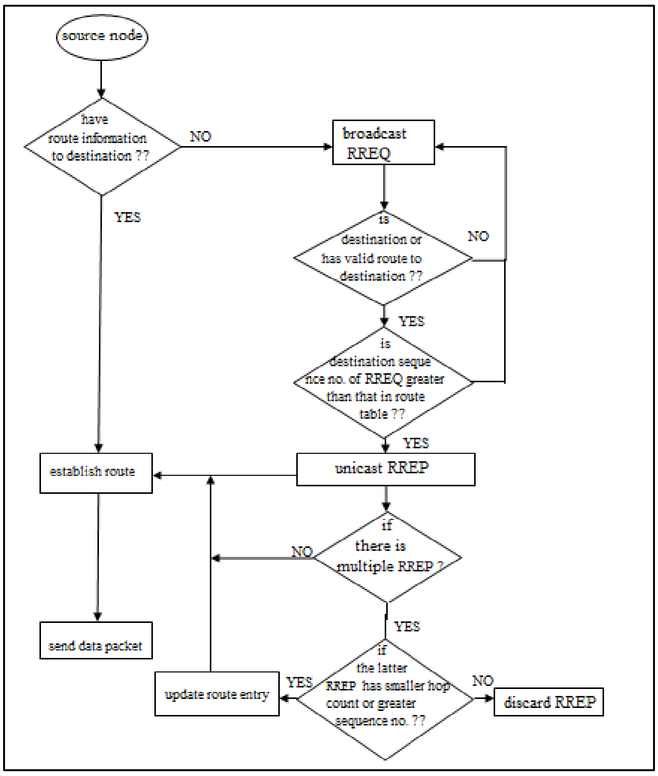


Fig. 3: Route Discovery Process

### B. Route Maintenance

Once the route between the source and the destination node is established it is maintained for the source node as long as it remains active. If the source node moves during an active session, it can simply reinitiate a route discovery process and establish a new route to the destination and continue communication. However, if either the destination or an intermediate node moves a route error (RERR) packet is sent to the source affected nodes.

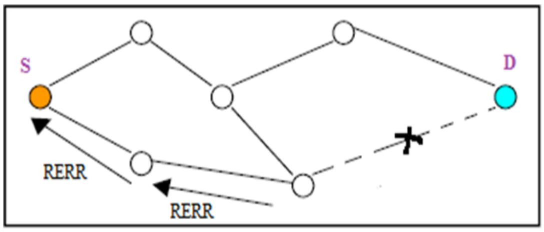


Fig. 4: Propagation of RERR message to source node

The RERR message is initiated by the node upstream of the link failure which is closer to the source. If the node upstream of the break has listed more than one node as a precursor node for the destination, it broadcasts the REER to these neighbors. When the neighbor nodes receive the RERR packet they mark the route to the destination as invalid by setting the distance to this destination node to infinity. When the RERR reaches the source node it can re initiate a route discovery if forwarding is required.

### C. Connectivity Determination

Network connectivity may be determined through the reception of broadcast control messages [9]. Any broadcast control message can serve as a hello message, indicating the presence of a neighbor as shown in fig.5. When a node receives a hello message from its neighbor, it creates or refreshes the routing table entry to the neighbor. To maintain connectivity a hello message is locally broadcasted, if this node has not sent any broadcast control message within a specified interval. This results in at least one hello message transmission during every time period. Failure to receive a hello message from a neighbor for several time intervals indicates that neighbor is no longer within transmission range and connectivity has been lost.

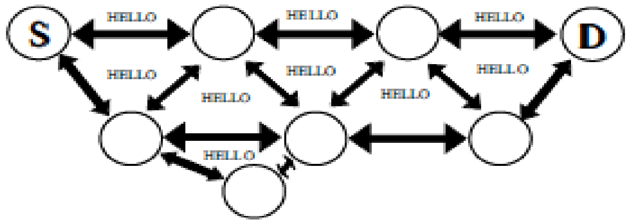


Fig. 5: Broadcast of Hello messages

The flow chart in Fig. 6 summarizes the action of connectivity determination and route maintenance in AODV.

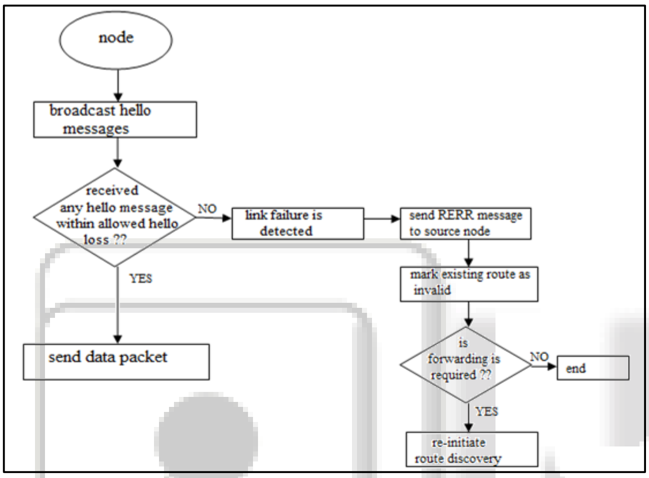


Fig. 6: Route maintenance and connectivity determination

## III. PROPOSED MODIFICATION

In this proposed work, the intention is to improve the QoS of AODV protocol to make it suitable for real time packet transmissions. The improvements made in the existing AODV protocol are discussed in this chapter.

### *A. Adaptive Hello Messaging Scheme*

In traditional AODV each node periodically exchanges hello messages between its neighbors. When a node is not involved in any communication for a given period, it does not need to maintain the status of the link, as the hello packets broadcasted during this period are unnecessary. It is obvious that unnecessary hello messaging can drain batteries while mobile devices are not in use. Increasing time gap between sending hello packets will decrease the number of hello messages but it can increase the risk of sending a packet through an unavailable link. Hence it is important to suppress unnecessary hello messages without reduced detectability of broken links.

The basic idea behind the adaptive hello messaging involves setting up a timer to broadcast hello messages. The timer is based on the activeness of the nodes. This timer must be selected such that the nodes which are actively taking part in communication, is given maximum time to update their neighbor i.e. they are given more time period for hello messaging. Whereas nodes which are less active are updated less frequently or they have less time to update their neighbor table. This means that the timer is set dynamically based on the activeness of the node. If the timer is set too low that results in loss in info and if it set too high it will result in no reduction in overhead.

This work uses an adaptive hello messaging scheme, in which hello interval is dynamically adjusted and does not increase the risk that a sender will transmit a packet through a broken link that has not been detected by hello messaging. For that instead of using constant hello interval we use constant risk level. This risk level is the probability of failure of detection of an unavailable link $P_{F\ D}$. To estimate this risk, we exploit an average event interval, that is, an average time gap between two consecutive events such as sending/receiving a data packet on a node. By monitoring the event intervals, we can estimate how actively a node is involved in sending or forwarding. As the event interval increases, the hello interval can also increase without increasing risk level. If the event interval is extremely large (node is inactive for long time), the hello messaging interval is also correspondingly large. In such cases hello interval is practically suppressed.

The basic principle behind the local link connectivity shows that when a node moves out of range, its intermediate nodes will detects its link failure before the transmission of packet. Fig. 8 depicts the time analysis

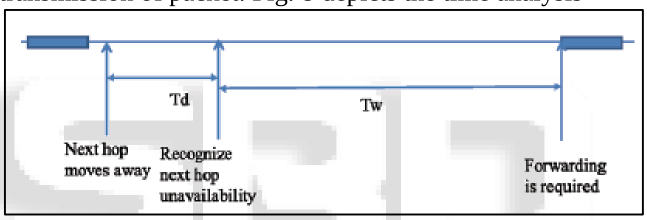


Fig. 7: Time analysis of local link connectivity

In the fig.7 $T_d$ is the link refresh period. It represents the time for link failure detection based on periodic Hello packet messaging. $T_w$ is the time gap between the link failures is detected and the link is needed.

The link refresh period $T_d$ for a single node is selected as in [2].

$$T_d = (\text{allowed hello loss} - 0.5\ ) * \text{hello interval} \quad (3.1)$$

In conventional scheme link availability information is updated as soon as it recognizes a link failure with one of its neighboring nodes. However, link failure information does not actually need to be updated until its neighbor is involved in communication such as forwarding a packet. That is link failure information is needed to update only within $T_d + T_W$. If this information is updated within $T_d$ , during $T_W$ multiple hello packets are broadcasted unnecessarily. Moreover, if a node has moved into an area where no active nodes are in its neighborhood and keeps broadcasting hello packets, energy is consumed unnecessarily. To avoid this problem, Hello packets should be suppressed. The proper solution, however, depends on determining the correct Hello interval.

Let us consider an event interval whose transmission of packets occurs between two senders and its neighbor. Two possible situations emerge when the neighbor node moves out of range. In case 1 the sender is in need to transfer packet and in case 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 immediately.

 

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. In case 1 hello interval should be small so that the node detect link failure before transmission of packet, but in case 2 longer hello interval can be used as the node is inactive and does not require the route information immediately.

The default hello interval is 1 second [3]. The hello interval needs to be increased only when the event interval of node is greater than one. As given in [10] all traffics in MANET are bounded by the exponential distribution where $x > 1$. When the distribution of event intervals is selected as exponential, for a given x , the CDF of event intervals is given by

$$F(x, \beta) = 1 - e^{x/\beta} \qquad (3.2)$$

Where $\beta$ is the average event interval of a node.

Considering $x$ as link refresh period in (3.1) $F(x, \beta)$ can be equated to probability of failed detection $P_{FD}$. Mathematically,

$$P_{FD} = 1 - e^{T_d/\beta} \qquad (3.3)$$

$$T_d = -\beta \ln(1 - P_{FD}) \qquad (3.4)$$

Note that, the link refresh period $T_d$ increases linearly as increases. The neighboring node in the aforementioned two cases can use for calculating the appropriate Hello interval to maximize $T_d$ and minimize $T_w$ in Fig 7. Once $T_d$ is determined using (3.4), the hello messaging interval of the neighboring node can be calculated using (3.1). Here allowed hello loss can be set to an arbitrary value. In the proposed scheme, allowed hello loss is also set as 2 as in conventional AODV [3] for comparison.

### B. Modified Route Discovery and Transmission Technique

In this modification the route discovery and packet forwarding in AODV is modified so that throughput is increased and end-to-end delay is decreased for real time applications. The respective modification adopted in the proposed work is described in next section.

#### 1) Traffic Aware Route Discovery:

In scenarios where the source node wants to communicate with another node having no routing information, the route discovery process starts by broadcasting RREQ message to its neighbors. The packet format of RREQ message is modified to carry the traffic load status information of each node in the route. The non real time packets also referred as best effort packets. Number of best effort (BE) packets among other type of packets in the queue is accounted at each node. The succeeding nodes compute the ratios by considering the respective ratio forwarded by previous nodes also.

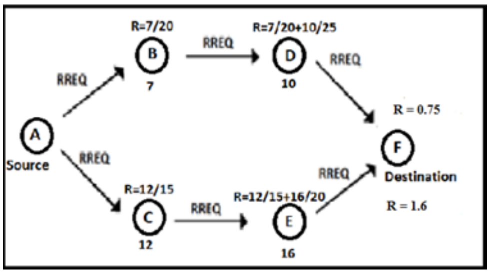


Fig. 8: Route request process

In figure 3.1, the value below the node is indicating the number of BE packets in the queue of that node. The ratio of BE packets to the rest of the packets is denoted as R.

$$R = \frac{BE\ Packets}{Total\ number\ of\ packets}$$

The computation process of R is repeated at each node through which the RREQ message is passed. The value of R is forwarded to the destination, where the final value is calculated. A reply message, RREP, is sent back to source node with this load information regarding R [11]. In fig.5 on receiving multiple RREP s the source node chooses the path which has minimum hop count as well as minimum R value. On receiving RREP from path A-B-D-F and A-C-E-F with same hop count source node chooses the path which has smaller R value. In the given example it is the path A-B-D-F. This way of path selection ensure that, real time traffic contains minimum BE packets. This enhances the throughput of overall system.

#### 2) Modified Packet Forwarding Procedure:

Increasing QoS for real time traffic has to ensure that the real time packets reside for minimum time in a queue. Normally, each node checks packets in its sending buffer for a particular destination in the network. If a packet is found in the queue, then it forwards the packet one by one without checking its type. In the proposed solution BE packets and real time packets are treated separately. Here each node in the route first checks its sending buffer for the arrival as well as type of packets. If the packet is real time (RT) then it is transmitted immediately. The same procedure will be followed by all the intermediate nodes in the route from source to the destination. By following this procedure, the destination will receive more real time packets in reduced time than normal scheduling procedure [11]. This work gives more transmission opportunity to real time data packets where the RT data packets require more bandwidth. The Possibility of dropping of real time data packets in case of congestion in network is reduced and improved end-to-end delay connectivity is achieved. Further enhancement in throughput of real time traffic is achieved by incorporating type selection (checking whether packet is RT or BE) in packet forwarding at each node.

### C. Local Link Failure Recovery (LLFR)

When a link failure occurs due to faint signal between nodes, the route has to be re-configured and repaired spontaneously so that there is no data loss. When a link failure is detected by a node, LLFR mechanism of node find an alternate path from that intermediate node [12]. The LLFR then updates the alternate path to source and sends the data packets to the destination. Instead of dropping the whole route as in the conventional AODV, a new route is discovered to destination. The over head of the system is not increased as the end-to-end control packets are substantially reduced. The packet delivery ratio increases, as preventive measures for safe landing of data packets to the destination are taken in the new route, by considering the signal strength of neighboring nodes.

When a link failure is detected the intermediate node act as the source node as it receives RERR message. It immediately triggers the LLFR to explore an alternate route to the destination simultaneously having a vigil on the signal strength of successive links.

 

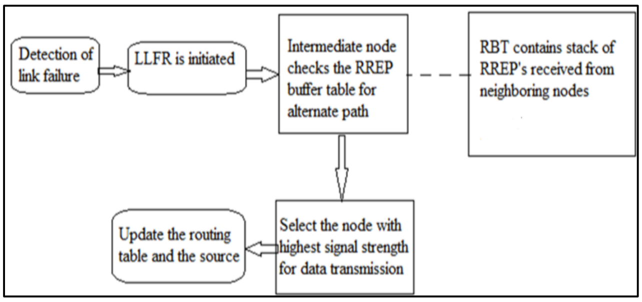


Fig. 9: LLFR Model

## IV. SIMULATION RESULTS AND DISCUSSIONS

This section discusses the effect of the proposed scheme on energy use, throughput, packet delivery factor, end to end delay, routing overhead with Ns-2 v.2.31 within various simulation parameters [13].

Table 1 contains simulation parameters used. The nodes move according to a random way point model with a velocity that allows a uniform distribution that ranges from minimum speed to maximum speed. Two traffic generation applications are used in this simulation process, CBR and TCP. CBR is used to simulate real time flows, and TCP to simulate best effort traffic.

Simulation is performed under two scenarios. In scenario 1 the throughput and packet delivery factor of conventional AODV and the modified protocol is compared by changing the mobility of nodes. In scenario 2, number of nodes is varied and effects in overhead and end-to-End delay are studied. Also the average energy consumption of nodes is evaluated.

| Parameter | Values |
|---|---|
| Simulation tool | NS-2.31 |
| Operating system | Ubuntu 10.04 |
| Simulation time | 200 s |
| Simulation area | 1000*1000 sqm |
| Number of nodes | 10,20,30,40,50 |
| Maximum speed of nodes | 5,10,15,20,25 m/s |
| Radio propagation model | Two ray ground |
| MAC protocol | MAC/802.11 |
| Routing protocol | AODV |

Table 1: Simulation Parameters

Each run of the simulator accepts (as input) a scenario file that describes the exact motion of each node using random way point mobility model. Hence, to evaluate the performance at a particular factor, 10 random simulation runs are considered to generate 10 random scenario patterns. The performance of the network according to the performance factors are computed from the average of these 10 outputs. In scenario 1 maximum speed of nodes varied as 5,10,15,20,25 m/s whereas the no of nodes, pause time , network size and simulation duration are fixed at 40, 20s, 1000X1000 sqm and 200s respectively. In scenario 2 number of nodes varied from 10 to 50 with an increment of 10 nodes whereas the pause time, network size and simulation duration are not changed.

Figure 10 compares the remaining energy in nodes after simulation in conventional AODV and modified AODV. In this simulation, initial energy of each node was set to 100 joules. The remaining energy determines the battery life time. As time duration is more, the initial supplied energy decreases.

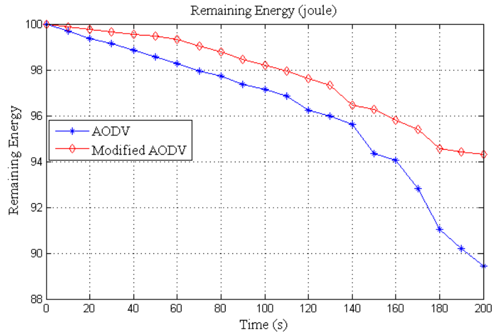


Fig. 10: Remaining energy in nodes.

The conventional AODV consumes energy at a faster rate due to periodic reception and transmission of hello packets. Further, re-routing in case of link failures also increases the energy consumption. As the proposed scheme reduces the number of unnecessary hello packets and since link failures are recovered locally, energy is saved due to reduced transmission events.

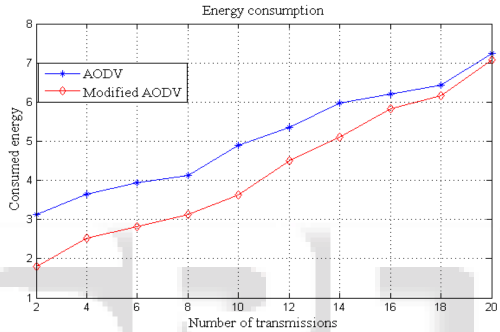


Fig. 11: Energy consumption.

Figure 11 compares the energy consumption between AODV and Modified AODV for variable packet transmissions. The effect of energy saving is high when the number of packet transmission is less than 10. As the number of packets increases, the effect of the energy saving decreases since more nodes will participate in forwarding data packets.

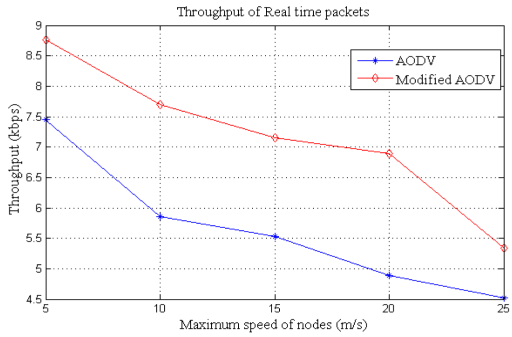


Fig. 12: Throughput of real time packets

In Fig. 12, Throughput of real time packets are plotted across maximum speed of the nodes to study the effect. Throughput is the number of bits successfully transmitted per unit second over a communication channel. Here the packet size is 512 bytes therefore throughput in kbps can be computed as

$$\text{Throughput} = \frac{\text{recieved packets} * 512 * 8 *}{\text{simulation time} * 1000}$$

When the speed of nodes increases throughput decreases. Results show that after the modification in route discovery process and packet forwarding procedure (which selects a route with less traffic and give more transmission probability to real time packets), the throughput of real time packets get increased compared to conventional AODV protocol.

Figure 13 compares the packet delivery factor of real time transmission of modified AODV protocol with the existing one by varying maximum speed of the nodes. Packet delivery factor (PDF) is the ratio of the numbers of packets received by the destination node to the number of packets sent by the source node.

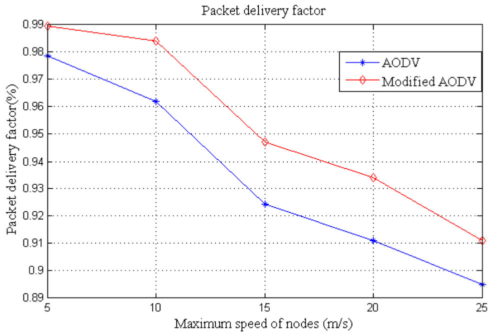


Fig. 13: Packet Delivery Factor

The packet delivery ratio of modified AODV protocol shows significant improvement when compared to conventional AODV routing protocol. The new route discovery mechanism selects routes for transmission that has lesser BE traffic on it. Also real time packets resides less time in queue. In case of link failures, the quick recovery of broken links reduces the packet drop. Hence the modified protocol results in higher packet delivery factor.

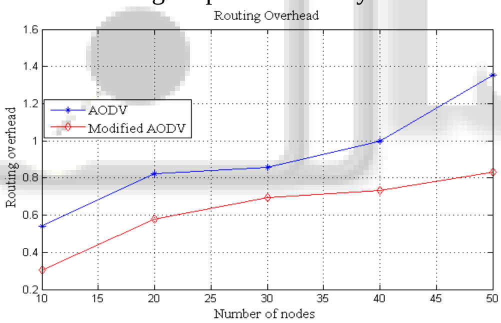


Fig. 14: Routing overhead

It can be seen from the fig. 14 that the number of routing packet increases as the number of nodes increases. This is mainly due to the number of nodes increases, more nodes will be flooding the network with RREQs and consequently more nodes will be able to send RREPs as well. This increases the number of control packets hence the overhead.

The modified protocol has reasonably lesser overhead when compared to AODV. In conventional AODV, mobile nodes respond to link failures with numerous messages that are flooded across the network to maintain an active route in AODV. In addition periodic hello messages broadcasted during each hello interval increases the routing overhead. The routing Protocol with LLFR has the best overhead performance because of its uniqueness in spontaneously responding to link failures. By making hello interval adaptive and by suppressing unnecessary hello messages, the modified routing protocol reduces the overhead significantly.

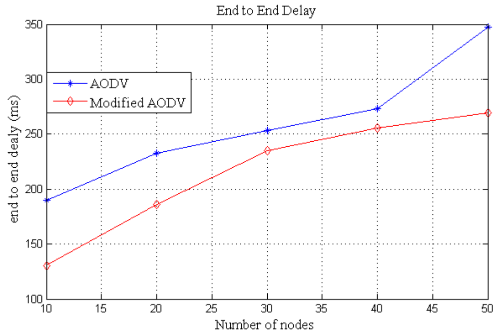


Fig. 15: End-To End Delay

Figure 15 compares the end-to end delay in conventional AODV and modified protocol. End to end delay (EED) is the time taken for an entire message to completely arrive at the destination from the source. As the number of nodes increases each node receives more number of route reply and the processing time also increases. This will increase end-to-end delay as the number of nodes increases.

The average end to end delay is reduced considerably in the modified protocol when compared to conventional AODV routing protocol in conditions of node failure. This has been achieved by allowing the intermediate node to spontaneously choose the alternate route during the link failure. Here, the data transmission time after failure is reduced, as the route buffer table has the RREP with highest signal strength.

## V. CONCLUSION

Routing in MANETs are challenging issues to be addressed for improving performance of transmission systems. AODV protocol, specially designed for MANET has the disadvantage of lack of QoS provisions. This research aims to improve the performance of AODV protocol. To achieve this goal three different routing approaches are combined in the conventional AODV. Unnecessary connecting checks are suppressed using adaptive hello messaging scheme. Route discovery and packet forwarding is modified so that protocol chooses route with less traffic. Real time packets reside for reduced time in the queue as the link failures are recovered locally at the point of link breakage.

Here, the performance of modified AODV routing protocol is compared with traditional AODV in terms of energy consumption, throughput, packet delivery ratio, routing overhead, and average end-to-end delay. The simulation results obtained confirms the improved efficiency by the significant improvement in QoS of the proposed modified AODV routing protocol. This makes the proposed modifications more suitable for real time communications

## REFERENCES

[1] M Kumar, R Mishra, “An overview of MANET: History, Challenges and application, ”Indian Journal of Computer Science and Engineering (IJCSE) vol. 3, no. 1, Feb-Mar 2012.

[2] Chakeres, Ian D, and Elizabeth M. Belding-Royer. “AODV routing protocol implementation design,” In Distributed Computing Systems Workshops, 2004.

 

Proceedings. 24th International Conference on. IEEE, pp. 698-703, 2004.

[3] Perkins, Charles, Elizabeth Belding-Royer, and Samir Das. “Ad hoc on-demand distance vector (AODV) routing (RFC 3561).” IETF MANET Working Group (August.2003).

[4] Lee, Sung-Ju, and Mario Gerla. “AODV-BR: Backup routing in ad hoc networks,” In Wireless Communications and Networking Conference, 2000. WCNC. 2000 IEEE, vol. 3, pp. 1311-1316. IEEE, 2000.

[5] Xue, Qi, and Aura Ganz. “Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks,” Journal of parallel and distributed computing 63, no. 2, pp. 154-165, 2003.

[6] Enzai, Nur Idawati Md, Farhat Anwar, and Omer Mahmoud. “Evaluation study of QoS-enabled AODV,” In Computer and Communication Engineering, 2008. ICCCE 2008. International Conference on, pp. 1254-1259. IEEE, 2008.

[7] Sarma, Nityananda, Sukumar Nandi, and Rakesh Tripathi. “Enhancing Route Recovery for QAODV Routing in Mobile Ad Hoc Networks,” In Parallel Architectures, Algorithms, and Networks, 2008. I-SPAN 2008. International Symposium on, pp. 39-44. IEEE, 2008.

[8] Wang and J. Crowcroft, “Quality-of-service routing for supporting multi-media applications,” IEEE Journal on Selected Area in Communications, vol. 14, no. 7, pp. 1228-1234, Sep. 1996.

[9] Chakeres, Ian D., and Elizabeth M. Belding-Royer. “The utility of hello messages for determining link connectivity,” In Wireless Personal Multimedia Communications, 2002. The 5th International Symposium on, vol. 2, pp. 504-508. IEEE, 2002.

[10] Seon Yeong Han, Member, IEEE, and Dongman Lee, Member, IEEE “An Adaptive Hello Messaging Han, Seon Yeong, and Dongman Lee. “An adaptive hello messaging scheme for neighbor discovery in on-demand MANET routing protocol,.” Communications Letters, IEEE 17, no. 5 (2013): 1040-1043.

[11] Iftikhar Ahmad, Humaira Jabeen, Faisal Rias “Improved Quality Of Service Protocol for Real Time Trafic in MANET ”, International Journal of Computer Networks and Communications (IJCNC) vol.5, no.4, July 2013.

[12] P.R. Jasmine jeni, A. Vimala Julie and A. Messiah Bose “An enhanced route failure recovery model for mobile ad hoc networks ”journal of computer science 10 (8): 1561-1568, 2014 ISSN: 1549-3636.

[13] The network simulator; NS-2 http://www.isi.edu/nsnam/ns.

 