International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 12 (2015) pp. 30801-30817

http://www.ripublication.com

Multipath On-Demand Routing for Large Scale Multi-Hop Networks

H.Santhi[1] and N.Jaisankar[2]

[1]*Assistant Professor (Senior), School of Computing Science and Engineering, VIT University, Vellore-632 014.*
[2] *Professor, School of Computing Science and Engineering, VIT University, Vellore-632 014.*
Emails: hsanthi@gmail.vit.ac.in, njaisankar@vit.ac.in

Abstract

Mobile ad hoc networks (MANETs) consist of a collection of wireless mobile nodes which dynamically exchange data without the need of fixed infrastructure or a wired backbone network. On-demand routing protocols use a flood-based discovery mechanism to find routes when required. Since each route discovery incurs high overhead and latency, the frequency of route discoveries must be kept low for on-demand protocols to be effective. On-demand routing protocols can achieve better performance and scalability by computing multiple routes in a single route discovery. Multipath routing is one of the applied approaches in mobile ad hoc networks to address their limited bandwidth and high route breakage rate. This paper proposed a novel approach called multipath on-demand routing (MPODRT), which reduces the routing overhead by using secondary paths by computing fail-safe multiple routes.

Keywords: Mobile ad hoc networks (MANET), Multipath routing, Primary path, Secondary path.

1. Introduction

Mobile Ad hoc Networks (MANETs) are autonomous networks, which operate without any fixed infrastructure or wired backbone. In MANETs [1], nodes typically communicate over multiple hops, while the intermediate nodes act as routers by forwarding data. Because of mobility and limited battery power of nodes topology of ad hoc network is highly dynamic. Hence routing protocols should adapt to such dynamic nature, and continue to maintain connection between the communicating nodes even if path breaks due to mobility and /or node failures.

The protocols are based on either link-state or distance-vector routing schemes are described in [2, 3]. These protocols compute routes to all the nodes in the network, and maintain them by periodically exchanging routing updates. Nodes also exchange triggered updates to maintain the consistent view of topology. These protocols cannot scale well because of their vast storage and communication overhead. Examples of such protocols are Destination Sequenced Distance Vector (DSDV) [6, 7, and 9] routing protocol for ad hoc networks, is an optimized version of the popular distributed Bellman Ford distance vector routing algorithm. In this protocol, nodes broadcast routing messages whenever they detect a topological change this leads to high overhead when nodes are moderate and highly mobile. And Fisheye State Routing (FSR) is based on link-state routing, in this nodes send link-state updates with high frequency to the nearest nodes and with lower frequency to far away nodes. These protocols become unmanageable in large networks as movement of any node in the network often trigger an update. Further, nodes possess invalid routing information to far away nodes, if they are highly mobile. This makes the route convergence difficult.

On-demand routing protocols [5] are well adapted to the dynamic environment of ad hoc networks, due to their low routing overhead and quick response to route disconnections. Nodes maintain routes only when needed. When a route to a destination is required, nodes flood a route request into the network. In the flooding process, if the destination or an intermediate has a route to the destination receives the request, the route is sent back to the source in a reply message. After the source node receives the reply, it starts sending data packets. Although, on-demand routing protocols show satisfactory performance in small networks, their performance degrades as the number of nodes in the network increases.

The protocol described in [5] maintains routes to active destinations by broadcasting a source-initiated query request. In any network, there may be more than one route to a destination. Single path routing protocols record only the most feasible (primary) path that was discovered earliest. Some on-demand single path routing protocols that have been proposed include Ad hoc On-demand Distance Vector (AODV)[8, 17] routing, Dynamic Source Routing (DSR), and Labelled Distance Routing (LDR). Dynamic Source Routing (DSR) has inherent inability to scale to larger networks due to its source routing. The need of flooding process of AODV has a significant impact on AODV’s performance in large networks. When networks grow to thousands of nodes, number of route breaks increases due to longer path lengths and mobility. A route error packet is sent to source for every route break, and the source node initiates a new route discovery to re-establish its connection to destination. Flooding the entire network each time a route is required consumes expensive bandwidth and limited processing power of nodes. This characteristic limits its ability to scale to larger networks.

The protocols presented in [6] reduce routing overhead by means of clustering. In [16] Hierarchical State Routing (HSR) nodes are grouped into clusters based on their geographical proximity, and a node in the cluster is elected as cluster-head to represent that cluster. These clusters are further grouped to form higher-level clusters and so on. Only cluster-head maintains routes to the nodes outside its cluster, by exchanging routing messages with respective cluster-heads. But HSR also lacks in scalability due to usage of flooding mechanism for finding routes to nodes outside the zone.

This paper is organized as follows. Section 2 recalls the related works. The novel scheme is proposed in Section 3. The details of the multipath routing framework are presented in Section 4. The analysis and performance evaluation are given in Section 5, 6. Finally we inwards conclusions and future work about the multipath routing protocols for MANETs.

2. Related Work

Mobile ad hoc networks (MANETs) [1, 6, and12] are characterized by a dynamic topology, limited channel bandwidth and limited power at the nodes. Because of these characteristics, paths connecting source nodes with destinations may be very unstable and go down at any time, making communication over ad hoc networks difficult. On the other hand, since all nodes in an ad hoc network can be connected dynamically in an arbitrary manner, it is usually possible to establish more than one path between a source and a destination. When this property of ad hoc networks is used in the routing process, then it is called multipath routing.

In most cases the ability of creating multiple routes from a source to a destination is used to provide a backup route. When the primary route fails to deliver the packets in some way, the backup is used. This provides a better fault tolerance and efficient recovery from route failures. Multiple paths [14] can also provide load balancing and route failure protection by distributing traffic among a set of paths. Multiple paths between a source and a destination can be disjoint in two ways: (a) link-disjoint paths and (b) node-disjoint paths. Node-disjoint paths do not have any nodes in common, except the source and destination hence they do not have any links in common. Link-disjoint paths, in contrast, do not have any links in common. They may, however, have one or more common nodes.

Many on-demand multipath routing protocols have been proposed for mobile ad hoc networks, including Split Multipath Routing (SMR)[4], Multipath Dynamic Source Routing (Multipath DSR)[5, 10, and13], Temporally Ordered Routing Algorithm (TORA), Routing On-demand Acyclic Multipath (ROAM), Ad hoc On-demand Multipath Distance Vector (AOMDV) [3], Ad hoc On-demand Distance Vector Backup Routing (AODV-BR) and Cooperative Packet Caching and Shortest Multipath (CHAMP). SMR and multipath DSR are based on source routing and are based on DSR while TORA, ROAM, AOMDV are distance-vector based. AODV-BR and AOMDV routing protocols are based on AODV.

The AODV-BR [2, 15] protocol uses the route discovery process as AODV [8]. When a source needs a route to a destination, and there is no route to that destination in its route cache, it searches a route by flooding a route request (RREQ) packet. Each of these packets has a unique ID so intermediate nodes can detect and drop duplicates. When an intermediate node receives a RREQ, it records the previous hop and the source node information and then broadcasts the packet or sends a route reply (RREP) packet back to the source if a route to the desired destination is known. The destination sends a RREP via the selected route when it receives the first RREQ or later RREQs that traversed a better route (with fewer hops).

The alternate route creation part is established during the RREP phase, and uses the nature of wireless communications. When a node that is not part of the selected route overhears a RREP packet not directed to it, it records the sending neighbour as the next hop to the destination in its alternate route table. In this way a node may receive numerous RREPs for the same route, select the best route among them and insert it into the alternate route table.

When an RREP finally reaches the source of the route, a primary route between that source and destination has been established. All the nodes that have an alternate route to the destination in their alternate route table form a fish bone. The properties of AODV-BR are is an extension of AODV. Floods RREQs with unique ID so duplicates can be discarded. Each node maintains backup route(s) in an alternate table. No multiple complete routes available. No multiple route(s) information known at source.

Like AODV-BR, the AOMDV [3, 9] uses the basic AODV route construction process. In this protocol some extensions are made to create multiple loop-free, link-disjoint paths. The main idea in AOMDV is to compute multiple paths during route discovery. It consists of two components: (i) A route update rule to establish and maintain multiple loop-free paths at each node. (ii) A distributed protocol to find link-disjoint paths. In AODV, when a source needs a route to a destination, it initiates a route discovery process by flooding a RREQ for destination throughout the network. RREQs should be uniquely identified by a sequence number so that duplicates can be recognized and discarded. Upon receiving a non-duplicate RREQ, an intermediate node records previous hop and checks whether there is a valid and fresh route entry to the destination in routing table. If such case, the node sends back a RREP to the source if not rebroadcasts the RREQ by incrementing the hop count. A node updates its routing information and propagates the RREP upon receiving further RREPs only if a RREP contains either a larger destination sequence number (fresher) or a shorter route found.

In AOMDV [17] each RREQ, respectively RREP arriving at a node potentially defines an alternate path to the source or destination. Just accepting all such copies will lead to the formation of routing loops. In order to eliminate any possibility of loops the “advertised hop count” is introduced. The advertised hop count of a node *i* for a destination *d* represents the maximum hop count of the multiple paths for *d* available at *i*. The protocol only accepts alternate routes with hop count lower than the advertised hop count, alternate routes with higher or the same hop count are discarded. The advertised hop count mechanism establishes multiple loop-free paths at every node. These paths still need to be disjoint.

In AOMDV duplicate copies of a RREQ are not immediately discarded. Each packet is examined to see if it provides a node-disjoint path to the source. For node-disjoint paths all RREQs need to arrive via different neighbour of the source. This is verified with the first hop field in the RREQ packet and the first hop list for the

RREQ packets at the node. At the destination a slightly different approach is used, the paths determined are link-disjoint or node-disjoint. In order to do this, the destination replies up to k copies of the RREQ, regardless of the first hops. The RREQs only need to arrive via unique neighbours.

In the SMR [4, 10] protocol, it provides way of determining maximally disjoint paths. Paths are maximally disjoint when they are node disjoint, but when there are no node-disjoint paths available, the protocol minimizes the number of common nodes. Multiple routes are discovered on demand, one of which is the path with the shortest delay. The routes established by the protocol are not necessarily equal in length.

When a source needs a route to a destination but no information is available, it floods a RREQ message on the entire network. Because of this flooding, several duplicates that traversed through the network over different routes reach the destination. The destination then selects multiple disjoint paths, and sends RREP packets back via the chosen routes. Since the destination needs to select disjoint paths, source routing is used. The complete route information is in the RREQ packets. Furthermore, intermediate nodes are not allowed to send RREPs, even when they have route information to the destination. If nodes reply from their cache ad in DSR and AODV, it is very difficult to determine maximally disjoint multiple routes since the destination does not receive enough RREQs and will not know the information of routes formed from intermediate nodes cache.

In SMR a new packet-forwarding approach is introduced. Instead of dropping all duplicate RREQs, intermediate nodes only forward the packets, which used a different incoming link than the first received RREQ and whose hop count is not larger than that of the first received RREQ. The SMR algorithm is optimized when the destination selects the two routes that are maximally disjoint. One of the two routes in the route with the shortest delay; the path taken by the first RREQ that destination receives. This path is used to minimize route acquisition latency required by on-demand protocols. When the first RREQ is received, the destination sends a RREP to the source via this path. The RREP includes the entire path so the intermediate nodes can determine where to forward the packet.

After sending the first RREP, the destination waits a certain amount of time to receive more RREQs and determine all possible routes. Because the destination then knows route information from all possible routes it can determine the maximally disjoint route to the already replied route. In the case where there is more than one maximally disjoint route, the shortest hop distance is used to select the desired route. When even that still leaves more than one route, the path that delivered the RREQ the fastest is chosen. The destination then sends a second RREP to the source along the path maximally disjoint to the first path.

3. Overview of the Proposed System

The ultimate aim of the paper is to develop multiple routes in order to improve scalability. Spontaneously, finding multiple paths in a single route discovery reduces the routing overhead incurred in maintaining the connection between source and destination nodes. Secondary paths can be used to transmit data packets, in case the primary path fails due to node mobility or battery failure, which avoids extra overhead generated by a fresh route discovery. These multiple paths are more advantageous in larger networks, where he number of route breaks are high.

3.1 Functionality
When a source node needs to send data to destination and does not have a valid path to destination, it starts a timer and relays a route request (RREQ) for destination with unique route request identifier. When source node receives a feasible reply for the destination it updates its route table and starts sending a data packet. In between if the timer expires then source node increments the route request identifier and initiates a new request for the destination.

Procedures involved in route discovery by MPODRT are:
- Initiate Route Request
- Relay Route Request
- Initiate Route Reply Destination
- Initiate Route Reply Intermediate Node
- Update/Add Route Entry
- Relay Route Reply
- Route Errors
- Route Data Packet

3.2 System Preliminary Design

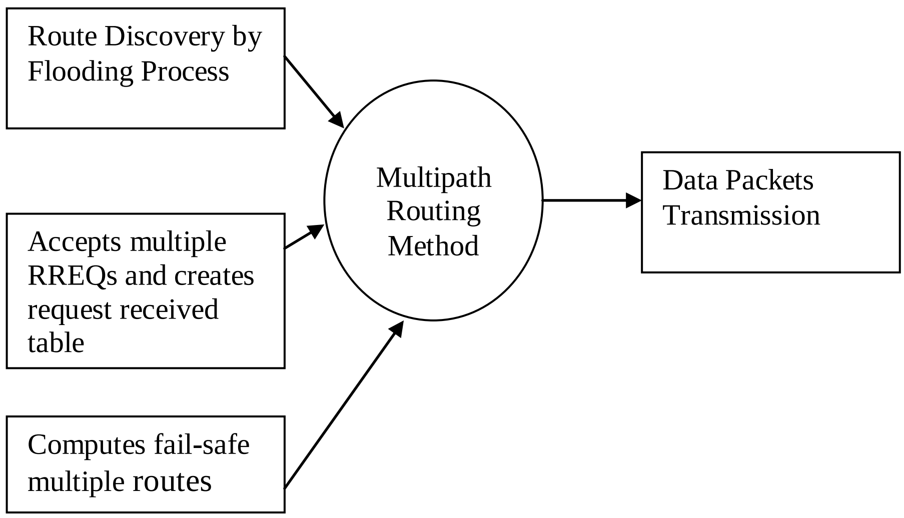

Fig.1 Preliminary Design

4. Design of the Proposed System
4.1 Architectural Design

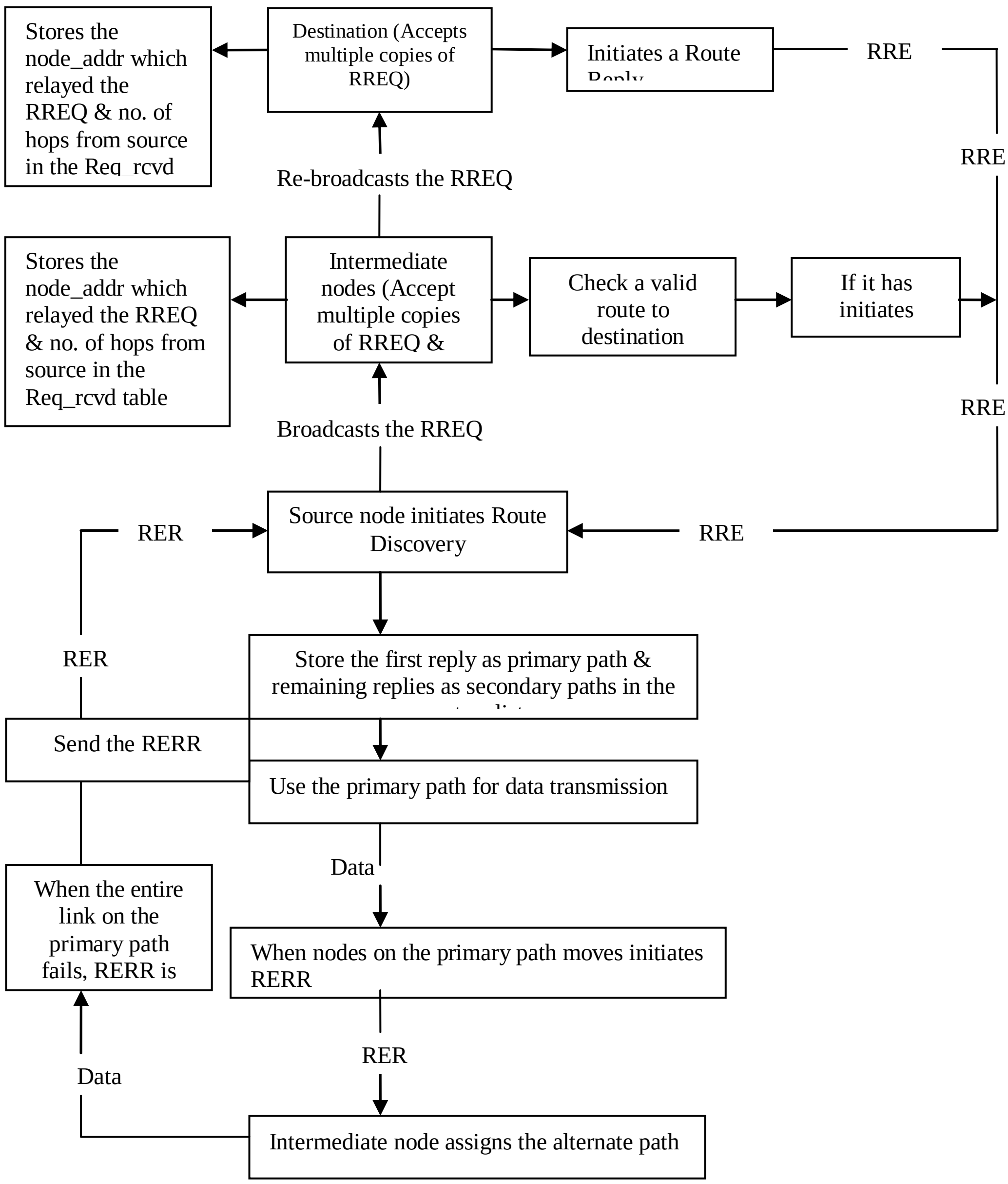

Fig.2 Overall System Architecture

4.2 Overall operation of MPODRT

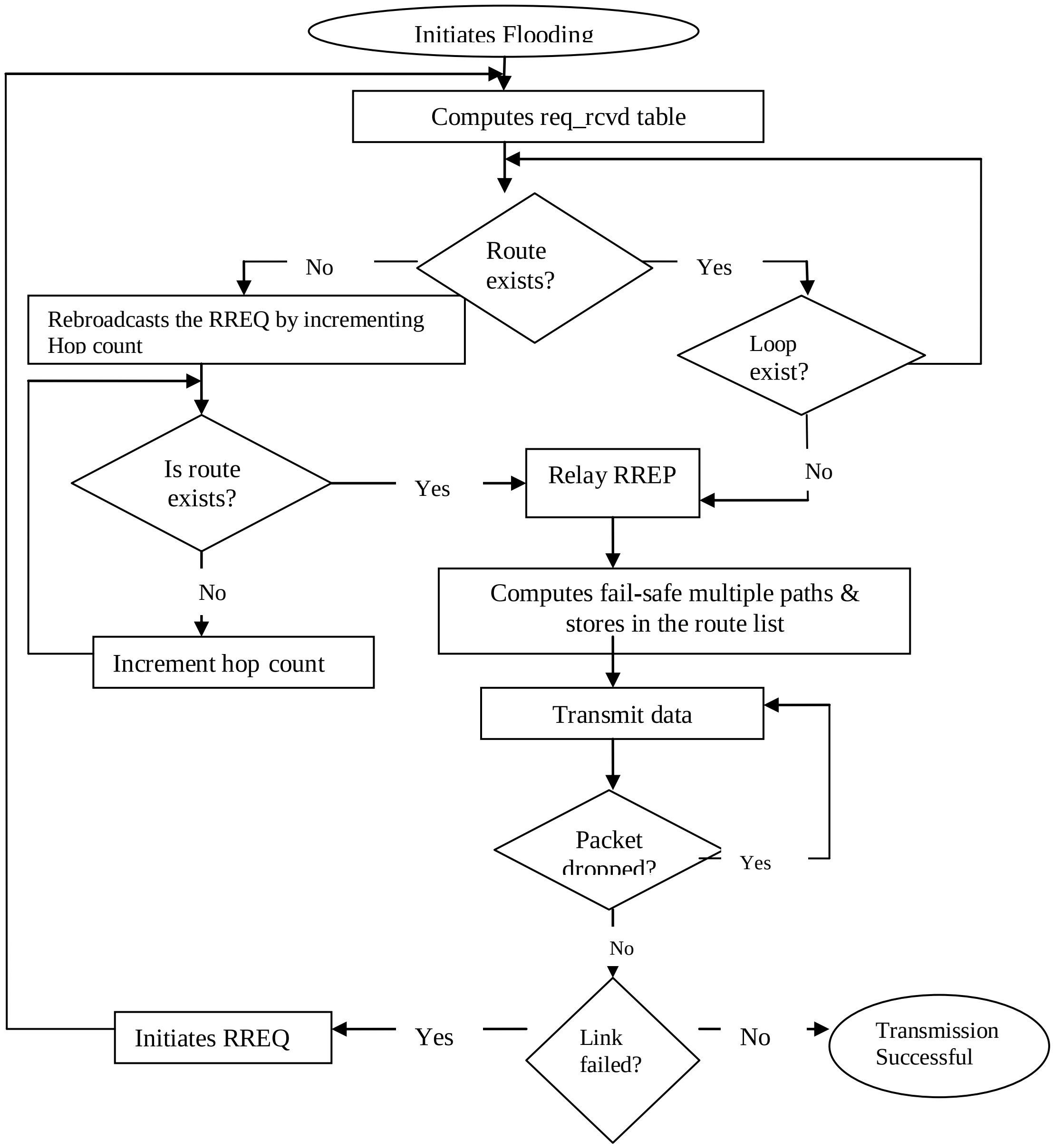

Fig.3 Flowchart of Overall Architecture

4.3 Process of control packets at each node

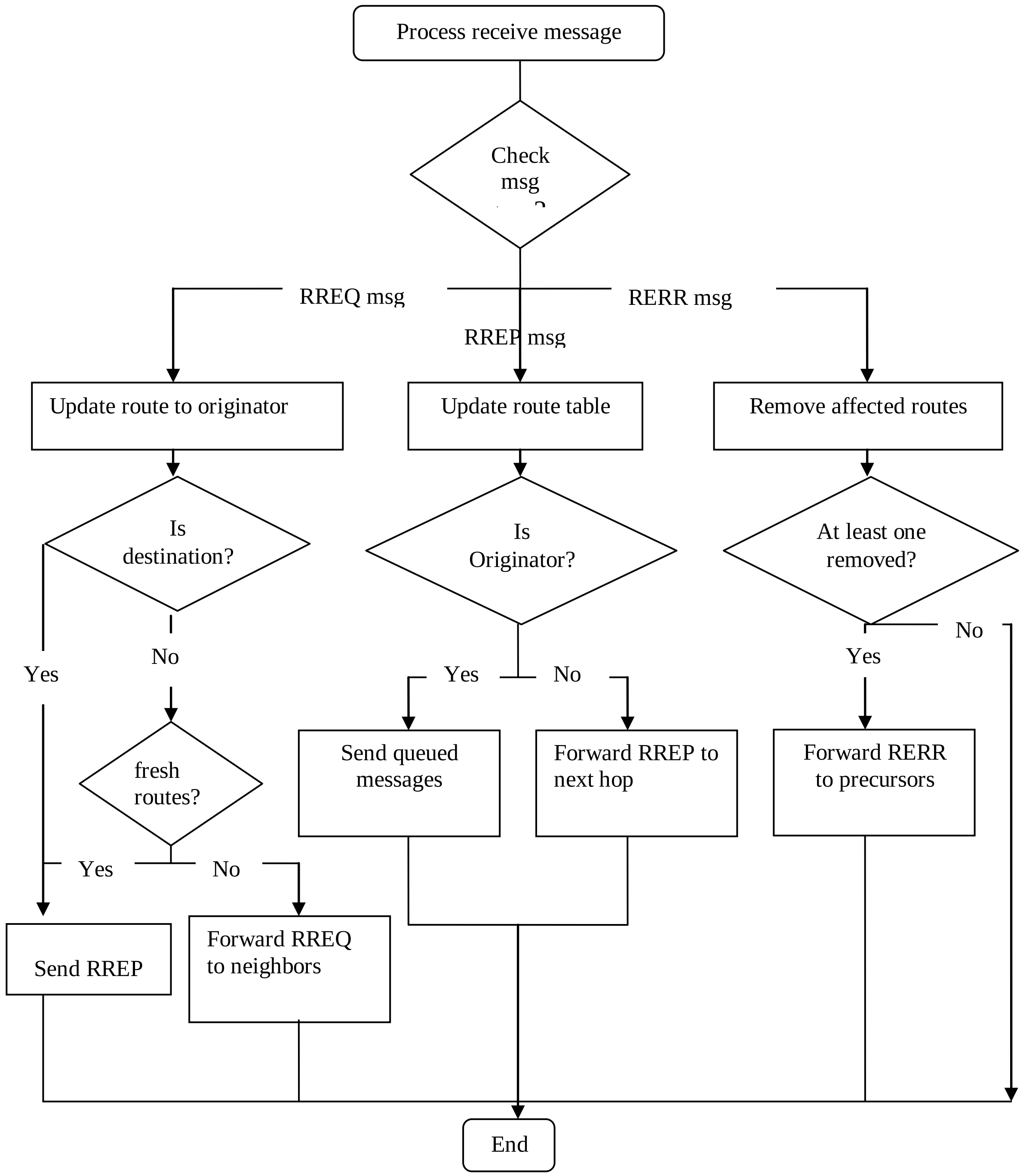

Fig. 4 Flowchart shows Operation of Route Discovery Process

5. Implementation

5.1 Design of Network Topology

We have implemented MPODRT using the GLOMOSIM. The simulation environment consists of different number of nodes in a rectangular region of varying size. The nodes are randomly placed in the region and each of them has a radio range of 150 meters. Five sessions of Constant Bit Rate flows are developed for data transmission. The random waypoint model is chosen as the node mobility model. Simulation time is 300 seconds. Each scenario is simulated five times and an average is taken for the performance analysis. The random waypoint model is chosen as the node mobility model. All data packets are 512 bytes. Table *1* shows the simulation parameters used.

Table 1.Simualtion parameters used

Parameter	**Description**
Terrain Area	Rectangular area is chosen. Area size varies with varying number of nodes.
Simulation time	300 seconds
Number of nodes	100 to 1000
Node placement	Uniform
Bandwidth	2Mbps
Receiver power threshold	-81.0 dBm
MAC layer protocol	IEEE 802.11
Propagation path loss model	Free-Space
Application type used	CBR (Constant Bit Rate)
Mobility model used	Random Waypoint Model

5.2 Implementation of MPODRT

Computes a fail-safe multiple routes between source and destination pair and maintains them in a route table as backup routes for data transmission. This task is achieved by extending the structure of route reply packet and the route table. It also extends the RREP control packet with three additional fields. But the RREQ packet structure is similar to AODV. Also creates two different tables Request Received Table and Route Table. Request Received Table entry is used for sending the RREP back to the source. Route Table maintains multiple route entries in a single discovery.

6. Performance Analysis

In order to evaluate and compare the performance of proposed technique MPODRT (Multipath On demand Routing), a most widely used unipath on demand protocol AODV is chosen.

- Throughput
- Number of Control packets transmitted

- Packet delivery ratio

6.1 Scenario – I: Keeping the mobility of a node constant

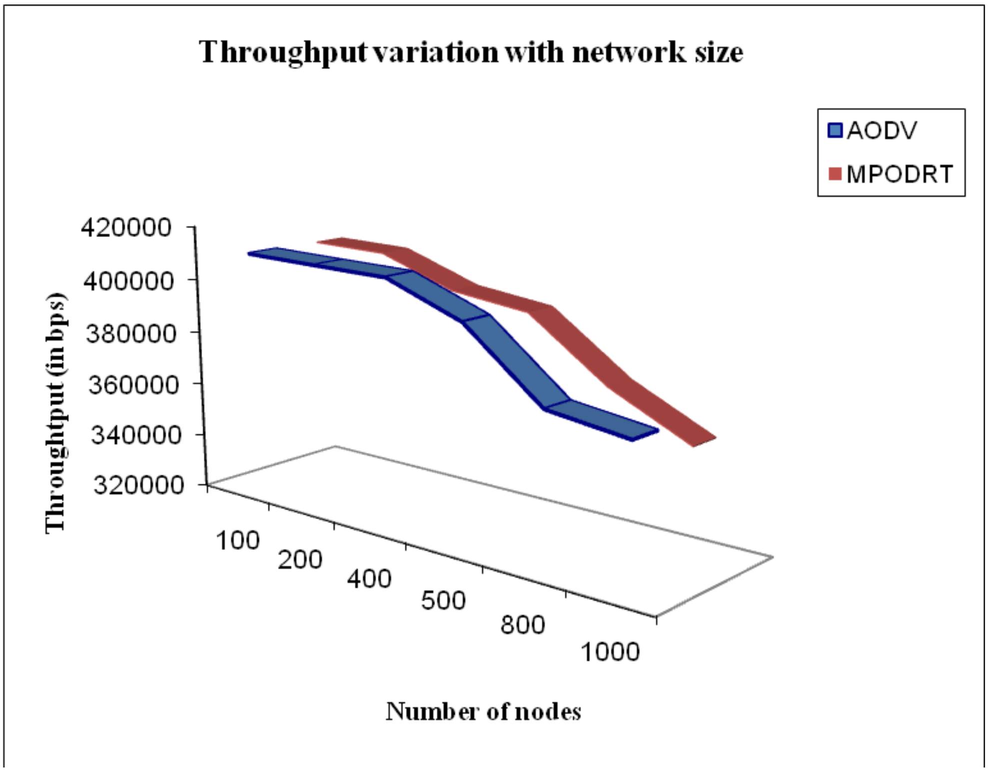

Fig. 5. Variation of throughput with network size.

Fig. 5 shows the throughput comparison of MPODRT and AODV. Packet delivery capacity of all these routing techniques decreases as the number of nodes in the network increases. This is due to the increasing number of route breaks as the size of network increases. However, MPODRT outperforms AODV in packet delivery capability for all sizes of network because most of the route breaks are corrected with secondary paths at intermediate nodes. This avoids packet drops at all the upstream nodes of the intermediate node that detected the route break. On the other hand, in AODV, all upstream nodes of the broken link drop packets to the disconnected destinations as they do not have secondary paths. Some of the packet drops are also due to the congestion caused by high routing overhead in AODV.

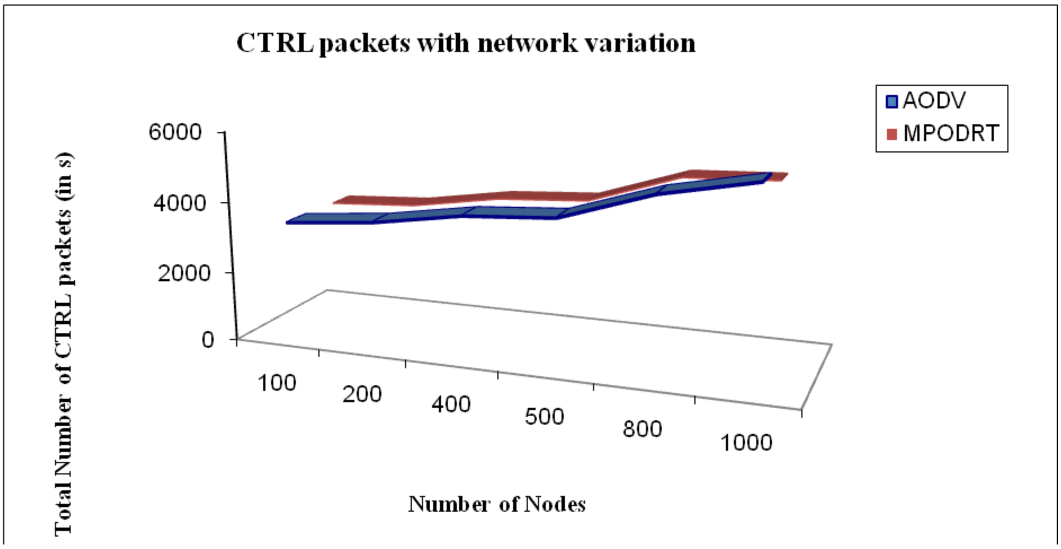

Fig. *6*. Variation of routing overhead with network size.

Fig. 6 shows the variation of routing overhead of two routing techniques. The value increases with network size because, the number of nodes communicating control packets and number of route computations increase as the network size increases. Number of route computations increase with network size because of increase in number of route breaks. AODV has higher routing overhead than MPODRT at all network sizes. This is because, AODV involves additional route computations and route error packet transmission for recovering route breaks. Where as in MPODRT route breaks can be resumed through the secondary paths and only a limited number of route breaks cause fresh route discoveries. Hence MPODRT has lower routing overhead that of AODV.

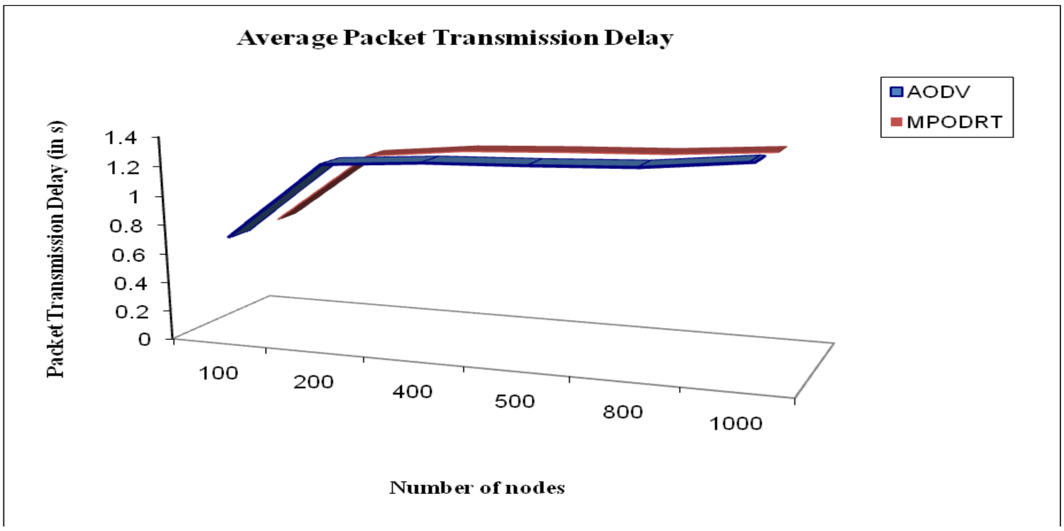

Fig. *7*. Variation of packet transmission delay with network size.

Fig. 7 shows the comparison of average packet transmission delay experienced by data packets for AODV and MPODRT. This metric reflects the delay involved in resuming the sessions after route breaks have occurred. The delay is high for AODV than MPODRT. But MPODRT has the lowest delay value at all network sizes, as it finishes the session with lowest number of route computations when compared to AODV.

6.2 Scenario – II: Varying the mobility speed

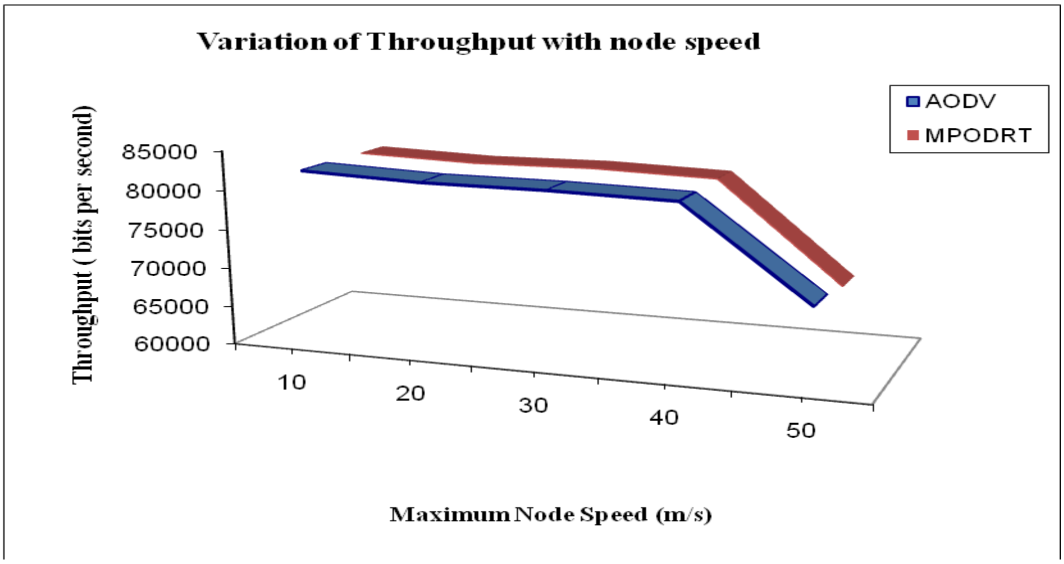

Fig. *8*. Variation of throughput with node speed.

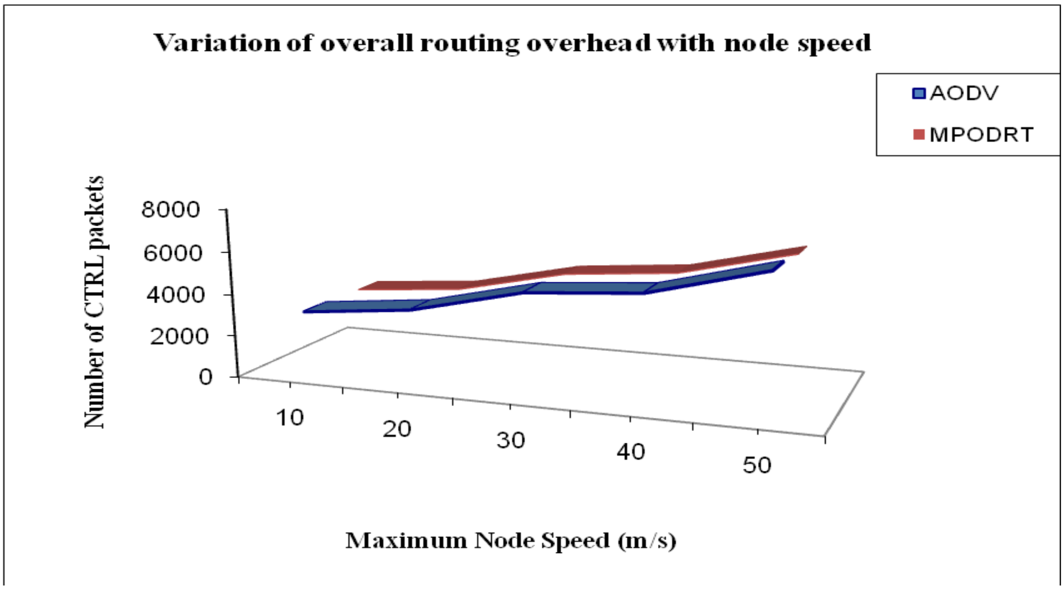

Fig. *9*. Variation of routing overhead with node speed.

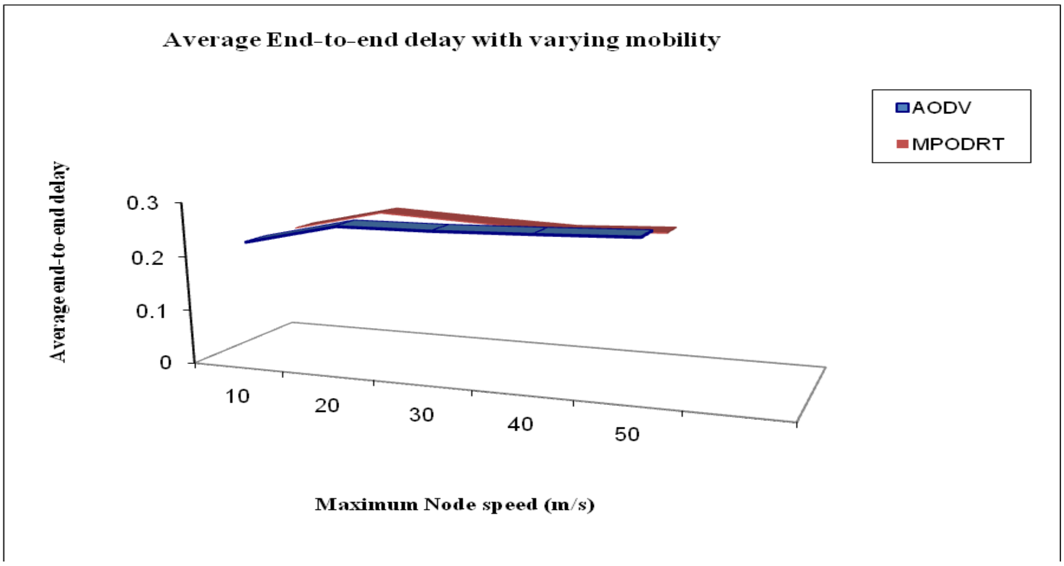

Fig.10 Variation of packet transmission delay with node speed.

The comparative results of throughput, control overhead and end-to-end delay are shown in Fig. 8, 9, and 10 respectively. As mobility increases, the protocol behaves as expected. Routing overhead and number of packet drops of these protocols increases with mobility, because of larger number of route breaks at higher speeds. But, MPODRT achieves improvement over AODV due to usage of secondary paths. Drastic increase of routing overhead in AODV at higher speeds show the need for methods to repair the route breaks with minimal routing overhead.

6.3 Scenario – III: Varying the network load

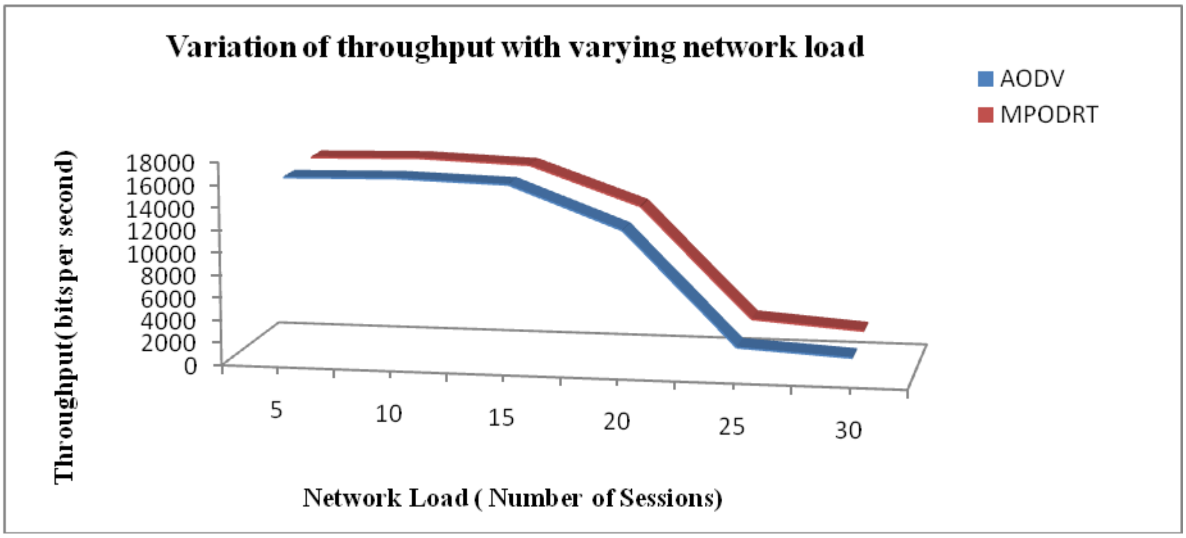

Fig.11 Variation of throughput with network load.

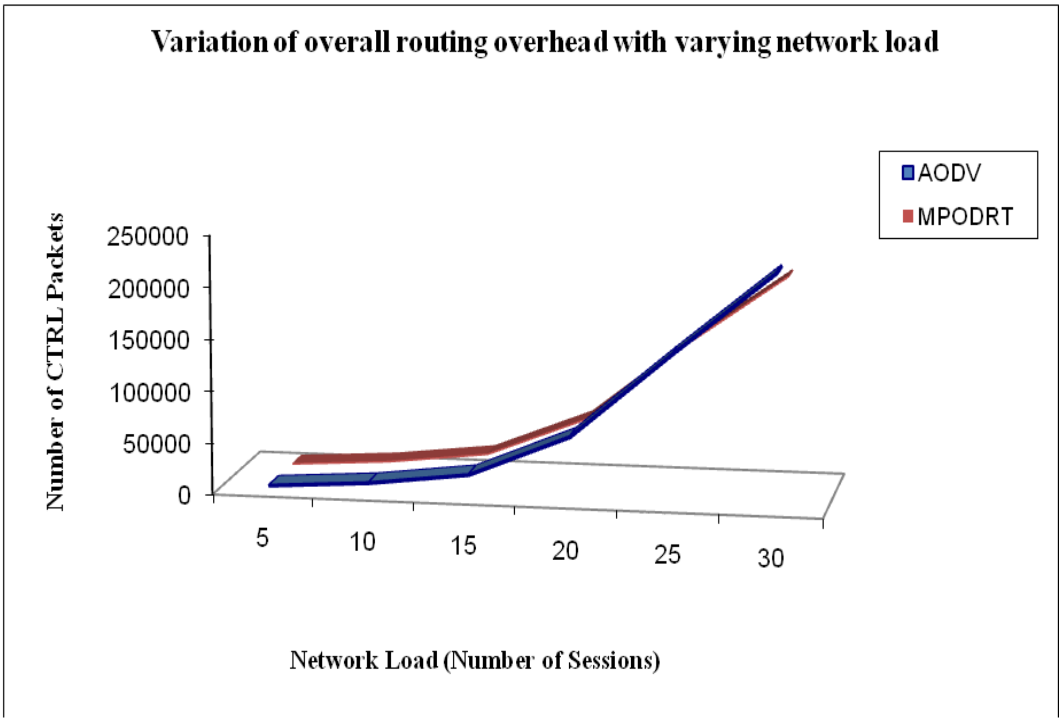

Fig.12 Variation of routing overhead with network load.

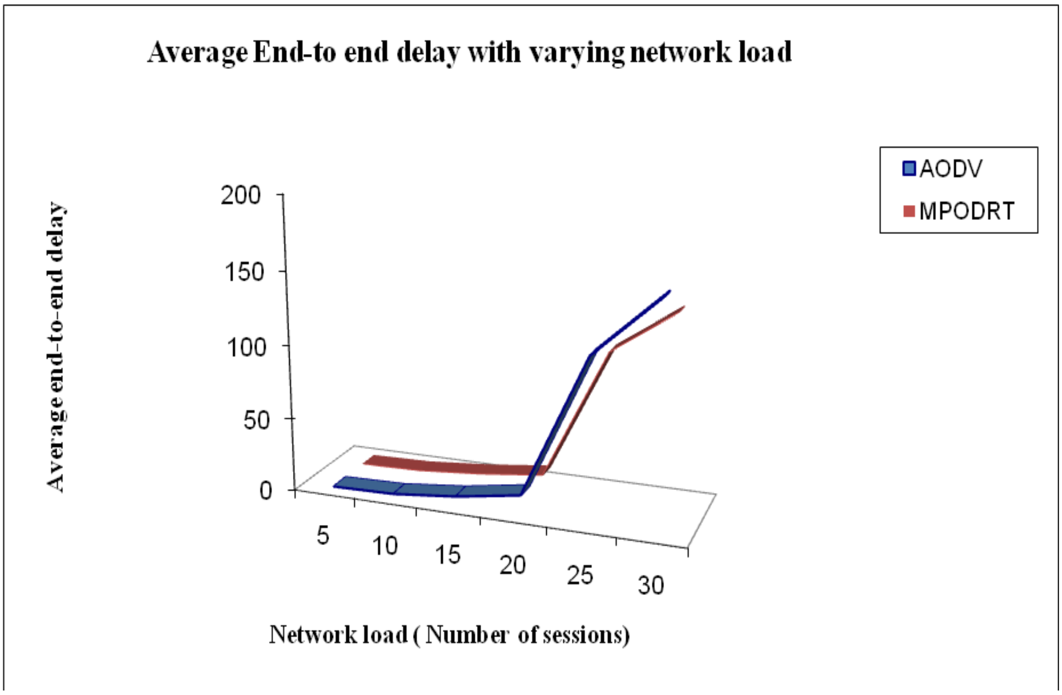

Fig.13 Variation of packet transmission with network load.

Figs. 11, 12, and 13 shows throughput, routing overhead, and end-to-end delay variation with offered load respectively. This slightly increases congestion at those nodes and causes some packet drops. At higher loads, number of false route breaks increases due to congestion created by more number of active sessions. False route breaks occur as nodes falsely assume that a route break as occurred, when there are lots of packet drops due to collisions created by congestion is intact. So, AODV's overhead increases as it initiates fresh route discovery for every route break. MPODRT outperforms AODV by using secondary paths to repair route breaks.

Conclusion and Future Work

The paper proposed a multipath routing, in order to improve scalability. Simultaneously, finding multiple paths in a single route discovery reduces the routing overhead incurred in maintaining the connection between source and destination nodes. Multipath routing can provide load balancing and reduce the frequency of on-demand route discovery. These benefits make multipath routing appear to be an ideal routing approach for MANETs. However, these benefits are not explored easily because multiple paths will interfere with each other's transmission and the cost of searching for proper multiple paths is usually larger than a single path. The performance analysis shows that the frequency of an on-demand route discovery for multipath routing is less than that for single path routing. The on-demand multipath routing can gain some improvement of end-to-end delay in a shared channel MANET. The network load can be distributed more evenly in multipath routing. Mobility can also contribute to the network load balancing. The initial selection of the multiple paths with different factors can control the average end-to-end delay when mobile speed is low.

In future, how multipath routing influences the power exhaustion and how to efficiently find the multiple diverse paths without the knowledge of the whole network topology can be considered.

References

[1] C.E. Perkins, E.M. Royer, "Ad-hoc On-Demand Distance Vector Routing", Proceedings of IEEE WMCSA, Feb 1999.

[2] S.J.Lee, M.Gerla, "AODV-BR: Backup Routing in ad hoc networks", IEEE WCNC, September 2000.

[3] Mahesh K. Marina Samir R. Das, "AOMDV: On-demand Multipath Distance Vector Routing in Ad Hoc Networks", 2000.

[4] Sung-Ju Lee and Mario Gerla, "Split Multipath Routing with Maximally Disjoint Paths in Ad hoc Networks", Proceedings of IEEE International Conference on Communications, June 2001.

[5] S. Murthy and G. L. Aceves, "An Efficient Routing Protocol for Wireless Networks, " *Mobile Networks and Applications*, 1996, Vol. 1, No. 2.

[6] C. E. Perkins and P. Bhagwat, “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, ” *Proceedings of ACM SIGCOMM*, 1994, pp. 234-244.

[7] 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*, 1999.

[8] D. B. Johnson and D.A. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks, ” Mobile *Computing,* 1996, pp. 153-181.

[9] S. Mueller, R. P. Tsang and D. Ghosal, “Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges, ” *InPerformance Tools and Applications to Networked Systems, Vol. 2965 of LNCS*, 2004.

[10] S. Mueller and D. Ghosal, “Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges, ” *Lecture Notes in Computer Science,* Springer, Berlin, 2004, pp. 209-234.

[11] Z. Ye, V. Srikanth and K.Tripathi, “A Framework for Reliable Routing in Mobile Ad hoc Networks, ” *IEEEINFOCOM 2003,* IEEE, San Francisco, 2003, pp. 270-280.

[12] K. Ishida, Y. Kakuda, M. Nakamura and T. Kikuno, K. Amano, “A protocol synthesis method for fault-tolerant multipath routing, ” *Information and Software Technology*, *Volume 41, Issues 11-12,* 1999, pp. 745-754.

[13] W. Lei, “Multipath source routing in wireless Ad hoc networks, ” *The 2000 IEEE Canadian conference on Electrical and Computer Engineering*, IEEE, Halifax, 2000, pp. 479-483.

[14] Yongwei Wang, Venkata C. Giruka and Mukesh Singhal, “Truthful multipath routing for ad hoc networks with selfish nodes, ” *Journal of Parallel and Distributed Computing, Vol. 68, No. 6*, 2008, pp. 778-789.

[15] S. J. Lee and M. Gerla, “AODV-BR: Backup routing in ad hoc networks, ” *Wireless Communications and Networking Conference*, IEEE, Chicago, 2000, pp. 1311-1316.

[16] Carlos T. Calafate, José Oliver, Juan-Carlos Cano, Pietro Manzoni and Manuel P. Malumbres, “A distributed admission control system for MANET environments Supporting multipath routing protocols, ” *Microprocessors and Microsystems*, *Volume 31, Issue 4,* 2007, pp. 236-251.

[17] M. K. Marina and S. R. Das, “On-demand multipath distance vector routing for ad hoc networks, ” *The International Conference for Network Protocols*, IEEE, California, 2001, pp. 14-23.