ISSN 2348-375X

Unique Journal of Engineering and Advanced Sciences

Available online: www.ujconline.net

Research Article

NEIGHBOUR COVERAGE BASED PROBABILISTIC REBROADCAST REDUCTION OF ROUTING OVERHEAD AND CROSS TALK AVOIDANCE IN MANETS

Arunkumar T*

Asst. Prof, Department of Electronics & Communication Engineering, Surya Engineering College, Erode, Tamilnadu, India.

Received: 10-03-2014; Revised: 08-04-2014; Accepted: 07-05-2014

*Corresponding Author: **Arunkumar T**

[1]Asst.Prof, Department of Electronics & Communication Engineering, Surya Engineering College, Erode, Tamilnadu, India. E-mail: ece_ta@surya.ac.in

ABSTRACT

In this approach if there exists any cross talk while communicating two parties, the connection will be terminated as soon as the crosstalk has been identified. It also improved crosstalk avoidance by either re-routing the communication or by terminating it with the help of packet transmissions and overhead. In order to limiting the number of rebroadcasts can effectively optimize the broadcasting, a neighbor coverage-based probabilistic rebroadcast (NCPR) protocol, in order to effectively exploit the neighbor coverage knowledge. A novel rebroadcast delay to determine the rebroadcast order, and then obtain a more accurate additional coverage ratio and by keeping the network connectivity and reduce the redundant retransmissions, a metric named connectivity factor to determine how many neighbors should receive the route request (RREQ) packet. By combining the additional coverage ratio and the connectivity factor, introduce a rebroadcast probability, which can be used to reduce the number of rebroadcasts of the RREQ packet to improve the routing performance.

Keywords: Ad hoc On-demand Distance Vector Routing (AODV), Dynamic Source Routing (DSR), Neighbor coverage-based probabilistic rebroadcast (NCPR).

INTRODUCTION

A Mobile Ad-hoc network is a wireless ad-hoc network which is used to exchange information. Each node is willing to forward data to other nodes. It does not rely on fixed infrastructure. Many routing protocols, such as Ad hoc On-demand Distance Vector Routing (AODV)[1] and Dynamic Source Routing (DSR)[2], have been proposed for MANETs. The above two protocols are on demand routing protocols, and they could improve the scalability of MANETs by limiting the routing overhead when a new route is requested[3]. However, due to node mobility in MANETs, frequent link breakages may lead to frequent path failures and route discoveries, which could increase the overhead of routing protocols and reduce the packet delivery ratio and increasing the end-to-end delay[4]. Thus, reducing the routing overhead in route discovery is an essential problem. The conventional On-demand routing protocols use flooding to discover a route. They broadcast a Route Request (RREQ) packet to the networks, and the broadcasting induces excessive redundant retransmissions of RREQ packet and causes the broadcast storm problem[5],

Some methods have been proposed to optimize the broadcast problem in MANETs in the past few years. Williams and Camp[6] categorized broadcasting protocols into four classes: "simple flooding, probability-based methods, area based methods, and neighbour knowledge methods." For the above four classes of broadcasting protocols, they showed that an increase in the number of nodes in a static network will degrade the performance of the probability-based and area-based methods[6]. Kim et al, indicated that the performance of neighbor knowledge methods is better than that of area-based ones, and the performance of area-based methods is better than that of probability-based ones. We now obtain the initial motivation of our protocol: Since limiting the number of rebroadcasts can effectively optimize the broadcasting[6], and the neighbor knowledge methods perform better than the area-based ones and the probability-based ones[7], then we propose a neighbor coverage-based probabilistic rebroadcast (NCPR) protocol. Therefore, 1) in order to effectively exploit the neighbor coverage knowledge, we need a novel rebroadcast delay to determine the rebroadcast order, and then we can obtain a more accurate additional coverage ratio; 2) in order to keep the network connectivity and reduce the redundant retransmissions, we need a metric named connectivity factor to determine how many neighbors should receive the RREQ packet.

The main contributions of this approach are as follows:

1. To calculate the rebroadcast delay. The rebroadcast delay is to determine the forwarding order. The node which has more common neighbours with the previous node has the lower delay. If this node rebroadcasts a packet, then more common neighbors will know this fact.
2. To calculate the rebroadcast probability. The scheme considers the information about the uncovered neighbors (UCN), connectivity metric and local node density to calculate the rebroadcast probability. The rebroadcast probability is composed of two parts: a. additional coverage ratio, which is the ratio of the number of nodes that should be covered by a single broadcast to the total number of neighbors; and b. connectivity factor, which reflects the relationship of network connectivity and the number of neighbors of a given node.

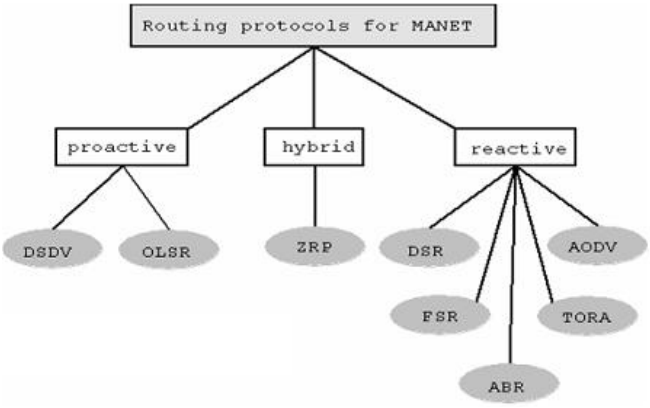

Figure 1: Routing Protocols in Ad Hoc Network

Related Works

Broadcasting is an effective mechanism for route discovery, but the routing overhead associated with the broadcasting can be quite large, especially in high dynamic networks[9].Ni et al, studied the broadcasting protocol analytically and experimentally, and showed that the rebroadcast is very costly and consumes too much network resource. The broadcasting incurs large routing overhead and causes many problems such as redundant retransmissions, contentions and collisions[5]. Thus, optimizing the broadcasting in route discovery is an effective solution to improve the routing performance. Haas et al, proposed a gossip based approach, where each node forwards a packet with a probability. They showed that gossip-based approach can save up to 35 percent overhead compared to the flooding.

However, when the network density is high or the traffic load is heavy, the improvement of the gossip-based approach is limited[9]. Kim et al, proposed a probabilistic broadcasting scheme based on coverage area and uses the neighbor confirmation to guarantee reachability. Peng and Lu[11] proposed a neighbor knowledge scheme named Scalable Broadcast Algorithm (SBA). This scheme determines the rebroadcast of a packet according to the fact whether this rebroadcast would reach additional nodes. Abdulai et al, proposed a Dynamic Probabilistic Route Discovery (DPR) scheme based on neighbor coverage. In this approach, each node determines the forwarding probability according to the number of its neighbors and the set of neighbors which are covered by the previous broadcast. This scheme only considers the coverage ratio but the previous node, and it does not consider the neighbors receiving the duplicate RREQ packet. Thus, there is a room of further optimization and extension for the DPR protocol.

Several robust protocols have been proposed in recent years besides the above optimization issues for broadcasting. Chen et al, proposed an AODV protocol with Directional Forward Routing (AODV-DFR) which takes the directional forwarding used in geographic routing into AODV protocol. While a route breaks, this protocol can automatically find the next-hop node for packet forwarding. Keshavarz-Haddad et al, proposed two deterministic timer-based broadcast schemes: Dynamic Reflector Broadcast (DRB) and Dynamic Connector-Connector Broadcast (DCCB).

They pointed out that their schemes can achieve full reachability over an idealistic lossless MAC layer, and for the situation of node failure and mobility, their schemes are robustness. Stann et al, proposed a Robust Broadcast Propagation (RBP) protocol to provide near-perfect reliability for flooding in wireless networks, and this protocol also has a good efficiency.

Uncovered Neighbors Set and Rebroadcast Delay

In this phase the calculation of coverage ratio and rebroadcast delay should takes place. When node ni receives an RREQ packet from its previous node s, it can use the neighbor list in the RREQ packet to estimate how many its neighbors have not been covered by the RREQ packet from s. In order to sufficiently exploit the neighbor knowledge and avoid channel collisions, each node should set a rebroadcast delay.

Neighbor Knowledge Probability

The node which has a larger rebroadcast delay may listen to RREQ packets from the nodes which have lower one. For example, if node ni receives a duplicate RREQ packet from its neighbor n_j , it knows that how many its neighbors have been covered by the RREQ packet from n_j. Thus, node n_i could further adjust its UCN set according to the neighbor list in the RREQ packet from n_j .

Table-Driven (or Proactive)

The nodes maintain a table of routes to every destination in the network, for this reason they periodically exchange messages. At all times the routes to all destinations are ready to use and as a consequence initial delays before sending data are small. Keeping routes to all destinations up-to-date, even if they are not used, is a disadvantage with regard to the usage of bandwidth and of network resources.

On-Demand (or Reactive)

These protocols were designed to overcome the wasted effort in maintaining unused routes. Routing information is acquired only when there is a need for it. The needed routes are calculated on demand. This saves the overhead of maintaining unused routes at each node, but on the other hand the latency for sending data packets will considerably increase.

Route Discovery

If node A has in his Route Cache a route to the destination E, this route is immediately used. If not, the Route Discovery

protocol is started. Node A (initiator) sends a Route Request packet by flooding the network.

If node B has recently seen another Route Request from the same target or if the address of node B is already listed in the Route Record, Then node B discards the request. If node B is the target of the Route Discovery, it returns a Route Reply to the initiator. The Route Reply contains a list of the "best" path from the initiator to the target. When the initiator receives this Route Reply, it caches this route in its Route Cache for use in sending subsequent packets to this destination. Otherwise node B isn't the target and it forwards the Route Request to his neighbors (except to the initiator).

Route Maintenance

In DSR every node is responsible for confirming that the next hop in the Source Route receives the packet. Also each packet is only forwarded once by a node (hop-by-hop routing). If a packet can't be received by a node, it is retransmitted up to some maximum number of times until a confirmation is received from the next hop.

Only if retransmission results then in a failure, a Route Error message is sent to the initiator that can remove that Source Route from its Route Cache. So the initiator can check his Route Cache for another route to the target. If there is no route in the cache, a RouteRequest packet is broadcasted.

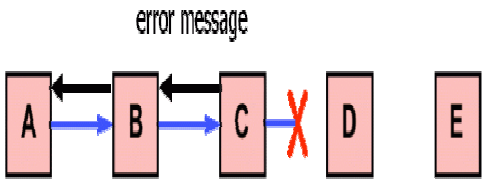

Figure 2: Routing Maintain

If node C does not receive an acknowledgement from node D after some number of requests, it returns a Route Error to the initiator A.

As soon as node receives the Route Error message, it deletes the broken-link-route from its cache.

If A has another route to E, it sends the packet immediately using this new route. Otherwise the initiator A is starting the Route Discovery process again.

Algorithm Description

The formal description of the Neighbor Coverage-based Probabilistic Rebroadcast for reducing routing overhead in route discovery is shown in algorithm 1.

Definitions

RREQv : RREQ packet received from node v,
Rv.id : the unique identifier (id) of RREQv,
N(u) : Neighbor set of node u,
U(u,x) : Uncovered neighbors set of node u for RREQ whose id is x,
Timer (u,x): Timer of node u for RREQ packet whose id is x.

a) It modify the source code of AODV in NS-2 (v2.30) to implement our proposed protocol.
b) Note that the NCPR protocol needs Hello packets to obtain the neighbor information, and also needs to carry the neighbor list in the RREQ packet.
c) Therefore, in this implementation, some techniques are used to reduce the overhead of Hello packets and neighbor list in the RREQ packet.

Algorithm (1) - NCPR

1: If n_i receives a new $RREQ_S$ from s then
2: Compute initial uncovered neighbors set $U(n_i, R_s.id \text{ for } RREQ_S)$
3: $U(n_i, R_s.id \text{ for } RREQ_S) = N(n_i) - [N(n_i) \cap N(s)] - \{s\}$
4: Compute the rebroadcast delay $T_d(n_i)$
5: $T_p(n_i) = 1 - [N(s) \cap N(n_i)]/[N(s)]$
6: $T_d(n_i) = MaxDelay * T_p(n_i)$
7: Set a timer $(n_i, R_s.id)$ according to $T_d(n_i)$
8: end if
9: while n_i receives a duplicate $RREQ_j$ from n_j before timer$(n_i, R_s.id)$ expires do
10: Adjust $U(n_i, R_s.id)$
11: $U(n_i, R_s.id) = (n_i, R_s.id) - [U(n_i, R_s.id) \cap N(n_j)]$
12: discard $(RREQ_j)$
13: end while
14: if timer$(n_i, R_s.id)$ expires then
15: Compute the rebroadcast probability $P_{re}(n_i)$
16: $R_u(n_i) = [U(n_i, R_s.id)]/N(n_i)]$
17: $F_c(n_i) = N_c / N(n_i)$
18: $P_{re}(n_i) = F_c(n_i) * R_u(n_i)$
19: if Random (0,1)$\leq P_{re}(n_i)$ then
20: broadcast $(RREQ_s)$
21: else
22: discard $(RREQ_s)$
23: end if
24: end if

Mobile Networking in NS2.35

This section describes the wireless model that was originally ported as CMU's Monarch group's mobility extension to NS2. The first section covers the original mobility model ported from CMU/Monarch group. In this section, we cover the internals of a mobile node, routing mechanisms and network components that are used to construct the network stack for a mobile node. The components that are covered briefly are Channel, Network interface, Radio propagation model, MAC protocols, Interface Queue, Link layer and Address resolution protocol model (ARP). CMU trace support and Generation of node movement and traffic scenario files are also covered in this section. The original CMU model allows simulation of pure wireless LANs or multihop ad-hoc networks. Further extensions were made to this model to allow combined simulation of wired and wireless networks. MobileIP was also extended to the wireless model.

The Basic Wireless Model in NS

The wireless model essentially consists of the Mobile node at the core, with additional supporting features that allows simulations of multi-hop ad-hoc networks, wireless LANs etc. The Mobile Node object is a split object. The C++ class Mobile Node is derived from parent class Node. A Mobile Node thus is the basic Node object with added functionalities of a wireless and mobile node like ability to move within a given topology, ability to receive and transmit signals to and from a wireless channel etc. A major difference between them, though, is that a Mobile Node is not connected by means of Links to other nodes or mobile nodes. In this section we shall describe the internals of Mobile Node, its routing mechanisms, the routing protocols dsdv, aodv, tora and dsr, creation of

network stack allowing channel access in MobileNode, brief description of each stack component, trace support and movement/traffic scenario generation for wireless simulations.

Mobile Node: Creating Wireless Topology

MobileNode is the basic nsNode object with added functionalities like movement, ability to transmit and receive on a channel that allows it to be used to create mobile, wireless simulation environments. The class MobileNode is derived from the base class Node. MobileNode is a split object. The mobility features including node movement, periodic position updates, maintaining topology boundary etc are implemented in C++ while plumbing of network components within MobileNode itself (like classifiers, dmux , LL, Mac, Channel etc) have been implemented in Otcl.

Protocol Implementation an Performance Evaluation

In order to reduce the overhead of Hello packets, not to be use periodical Hello mechanism. Since a node sending any broadcasting packets can inform its neighbors of its existence, the broadcasting packets such as RREQ and route error (RERR) can play a role of Hello packets. By use the following mechanism to reduce the overhead of Hello packets. Only when the time elapsed from the last broadcasting packet (RREQ, RERR, or some other broadcasting packets) is greater than the value of Hello Interval, the node needs to send a Hello packet. The value of Hello Interval is equal to that of the original AODV.

The RREQ packet, each node needs to monitor the variation of its neighbor table and maintain a cache of the neighbor list in the received RREQ packet modify the RREQ header of AODV, and add a fixed field num neighbors which represents the size of neighbor list in the RREQ packet and following the num neighbors is the dynamic neighbor list. In the interval of two close followed sending or forwarding of RREQ packets, the neighbor table of any node ni has the following three cases:

a) If the neighbor table of node ni adds at least one new neighbor nj, then node ni sets the num- neighbors to a positive integer, which is the number of listed neighbors, and then fills its complete neighbor list after the num neighborsfield in the RREQ packet. It is because that node nj may not have cached the neighbor information of node ni, and, thus, node nj needs the complete neighbor list of node ni.

b) If the neighbor table of node ni deletes some neighbors, then node ni sets the num neighbors to a negative integer, which is the opposite number of the number of deleted neighbors, and then only needs to fill the deleted neighbors after the num neighbors field in the RREQ packet.

c) If the neighbor table of node ni does not vary, node ni does not need to list its neighbors, and set the num neighbors to 0. The nodes which receive the RREQ packet from node ni can take their actions according to the value

Simulation Parameters

The Distributed Coordination Function (DCF) of the IEEE 802.11 protocol is used as the MAC layer protocol. The radio channel model follows a Lucent's Wave - LAN with a bit rate of 2 Mbps, and the transmission range is 250 meters. It consider constant bit rate (CBR) data traffic and randomly choose different source-destination connections. Every source sends four CBR packets whose size is 512 bytes per second. The mobility model is based on the random waypoint model in a field of 1000m × 1000m. In this mobility model, each node moves to a random selected destination with a random speed from a uniform distribution. After the node reaches its destination, it stops for a pause time interval and chooses a new destination and speed.

Table 1: Parameters

SIMULATION PARAMETER	VALUES
Simulator	NS-2 (V2.35)
Topology Size	1500 m × 1500 m
Number of nodes	50
Transmission range	250 m
Bandwidth	1 Mbps
Interference Queue Length	100
Traffic Type	CBR
Number of CBR connections	10
Packet Size	1024 bytes
Packet Rate	8 packets / s
Pause Time	0 second
Minimum speed	5 m/s
Maximum speed	10 m/s

Performance of Routing Protocols – Metrics

MAC collision rate: The average number of packets (including RREQ, route reply (RREP), RERR, and CBR data packets) dropped resulting from the collisions at the MAC layer per second.

Normalized routing overhead: The ratio of the total packet size of control packets (include RREQ, RREP, RERR, and Hello) to the total packet size of data packets delivered to the destinations. For the control packets sent over multiple hops, each single hop is counted as one transmission. To preserve fairness, use the size of RREQ packets instead of the number of RREQ packets, because the DPR and NCPR protocols include a neighbor list in the RREQ packet and its size is bigger than that of the original AODV.

Packet delivery ratio: The ratio of the number of data packets successfully received by the CBR destination to the number of data packets generated by the CBR sources.

Average end-to-end delay: The average delay of successfully delivered CBR packets from source to destination node. It includes all possible delays from the CBR sources to destinations. The experiments are divided to three parts, and in each part, its evaluate the impact of one of the following parameters on the performance of routing protocols:

Number of nodes: It vary the number of nodes from 50 to 300 in a fixed field to evaluate the impact of different network density. In this part, set the number of CBR connections to 15, and do not introduce extra packet loss.

Number of CBR connections: By change the number of randomly chosen CBR connections from 10 to 20 with a fixed packet rate to evaluate the impact of different traffic load. In

this part, set the number of nodes to 150, and also do not introduce extra packet loss.

Random packet loss rate: Use the Error Model provided in the NS-2 simulator to introduce packet loss to evaluate the impact of random packet loss. The packet loss rate is uniformly distributed, whose range is from 0 to 0.1. In this part, set the number of nodes to 150 and set the number of connections to 15.

Performance with Varied Number of Nodes

The effects of network density on the MAC collision rate shows in Fig. 3. In the IEEE 802.11 protocol, the data and control packets share the same physical channel. In the conventional AODV protocol. It is very important to reduce the redundant rebroadcast and packet drops caused by collisions to improve the routing performance. Compared with the conventional AODV protocol, the NCPR protocol reduces the MAC collision rate by about 92.8 percent on the average. Under the same network conditions, the MAC collision rate is reduced by about 61.6 percent when the NCPR protocol is compared with the DPR protocol. This is the main reason that the NCPR protocol could improve the routing performance.

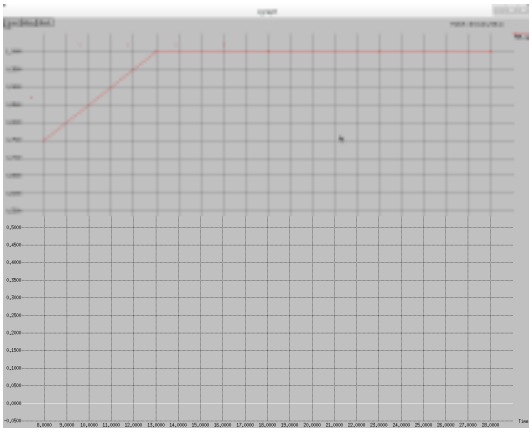

Figure 3: Packet delivery ratio

Performance With Varied Number Of CBR-Connections

Traffic load on the MAC collision rate shows in Fig. 4. Since the data and control packets share the same physical channel in the IEEE 802.11 protocol, as the number of CBR connections increases, the physical channel will be busier and then the collision of the MAC layer will be more severe. Both the DPR and NCPR protocols do not consider load balance, but they can reduce the redundant rebroadcast and alleviate the channel congestion, so as to reduce the packet drops caused by collisions. Compared with the conventional AODV protocol, the NCPR protocol reduces the MAC collision rate by about 95.2 percent on the average.

Performance With Varied Random Packet Loss Rate

The packet loss rate on the MAC collision rate shows in Fig.5. In our simulation parameters, we use both the Incoming Err Proc and Outgoing Err Proc options at the same time; thus, the packet error will be more often and the retransmissions caused by random packet loss at MAC layer will be more. Therefore, the MAC collision rate of all the three routing protocols increases as the packet loss rate increases. Both the DPR and NCPR protocols do not consider robustness for packet loss, but they can reduce the redundant rebroadcast and alleviate the channel congestion, thus, both of them have the lower packet dropscaused by collisions than the conventional AODV protocol. Compared with the conventional AODV protocol, the NCPR protocol reduces the MAC collision rate by about 92.8 percent on the average. In the same network density and traffic load but in different packet loss rate, the MAC collision rate is reduced by about 61.6 percent when the NCPR protocol is compared with the DPR protocol

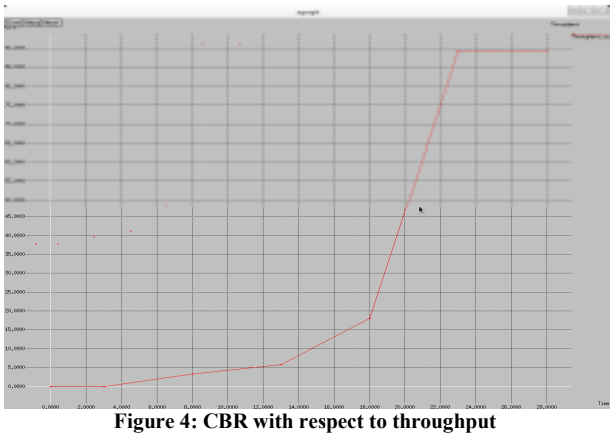

Figure 4: CBR with respect to throughput

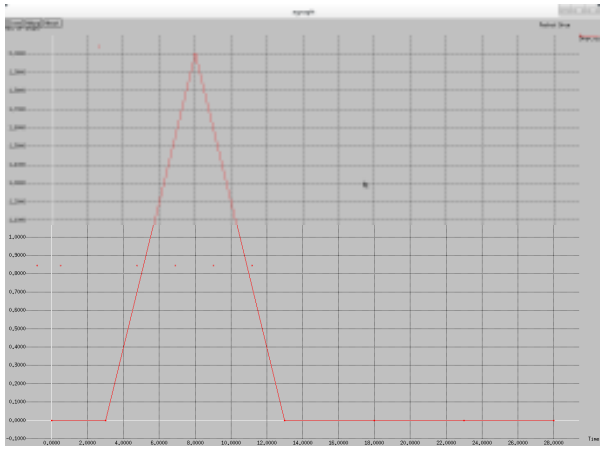

Figure 5: Performance of Random packet loss (or) drop

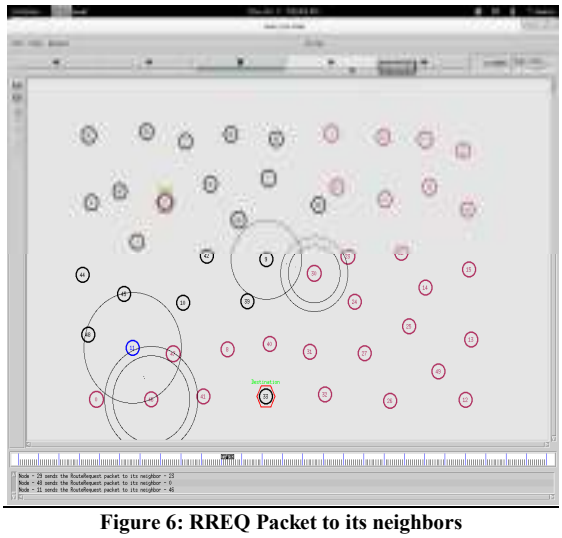

Figure 6: RREQ Packet to its neighbors

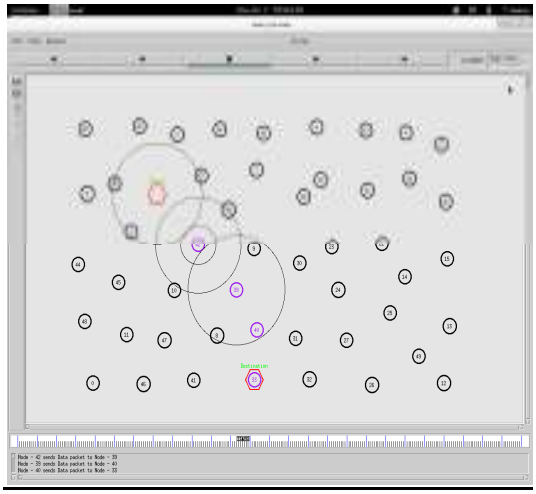

Figure 7: Source nodes send data packet to Destination nodes

CONCLUSION

It produces the complete performance of throughput which is in the routing protocol. It may be working in wired LAN protocol by the MAC function and avoid the cross-talk strategy in communication medium. It minimizes the packet dropping ratio and maximizes the end to end delivery. To reduce the reducing overhead in MANETs and insists a coverage ratio and connectivity factor. Dynamically calculate the re-broadcast delay and decrease the average end to end delay. It provides good performance when the network is in high traffic in heavy load. Easily detecting and avoiding the packet overhead in the one particular network. To avoid cross talk in the mobile communication by simulate the packet transmissions.

REFERENCES

1. Sze-Yao Ni, Yu-Chee Tseng, Yuh-Shyan Chen, and Jang-Ping Sheu, The Broadcast Storm Problem in a Mobile Ad Hoc Network, Milcom Conference Record, 1995; 1: 236-240.
2. Huda Al Aamri, On Optimising Route Discovery for Multi-Interface and Power-Aware nodes in Heterogeneous MANETs, 2010 Sixth International Conference on Wireless and Mobile Communications, 33rd IEEE Conference, 2008; 648–654.
3. Xianren WU, Sadjadpour HR and Garcia-Luna-Aceves JJ, Routing Overhead as A Function of Node Mobility: Modeling Framework and Implications on Proactive Routing IEEE, 1997; 3:1405–1413.
4. Tasneem Bano, Maulana Azad, Probabilistic: A Fuzzy Logic-Based Distance Broadcasting Scheme For Mobile Ad Hoc Networks,(IJACSA) International Journal of Advanced Computer Science and Applications, 2012; 3: 9.

Source of support: Nil, Conflict of interest: None Declared