MJRET
Open Access
Multidisciplinary Journal of Research in Engineering and Technology
Journal homepage: www.mjret.in
ISSN:2348 - 6953

NEIGHBOUR-AWARE ROUTING PROTOCOL IN MANET TO MINIMIZE ROUTING OVERHEAD

Akshay Shah, Danish Nizami, Isha Chopra, Navneet Bhagasra
BE, Computer Engineering,
MIT Academy of Engineering,Pune, India

Abstract: *Mobile ad hoc networks (MANETs) are collections of wireless mobile nodes, constructed dynamically without the use of any existing network infrastructure or centralized administration. Due to the limited transmission range of wireless network interfaces, multiple hops may be needed for one node to exchange data with another one across the network. MANETs are characterized by limited power resource, limited transmission range, high mobility and limited bandwidth. Routing in MANETs can be accomplished through either single-path or multiple paths. Due to unpredictable behavior of wireless mobile nodes, frequent route failure occurs in the MANET, which results in performance degradation. In existing mechanisms of routing protocol, due to special control messages, route request storm, inefficient route discovery and duplicate packet processing, it generates many control packets in the network which leads to increase in overall routing overhead of the network. An efficient routing mechanism is needed to improve the performance of the network by reducing the routing overhead during route discovery process. Particular attention can be given on routing table size of intermediate nodes in dynamic route discovery mechanism.*

Keywords—NCPR, reducing routing overhead, dynamic routing discovery

1. INTRODUCTION

The maturity of wireless transmissions and the popularity of portable computing devices have made the dream of “communication anytime and anywhere” possible. Users can move around, while at the same time still remaining connected with the rest of the world.

This phenomenon is called mobile computing or nomadic computing. It has received intensive attention recently. Generally, most of the nomadic computing applications today require single hop connectivity to the wired network. This is the typical cellular network model that supports the needs of wireless communications by installing base stations or access points. In such networks, communications between two mobile hosts completely rely on the wired backbone and the fixed base stations. Nevertheless, the wired backbone infrastructure may be unavailable for use by mobile hosts for many reasons, such as unexpected natural disasters and radio shadows. Also, it might be infeasible to construct sufficient fixed access points due to cost and performance considerations; for instance, having fixed network infrastructure in wilderness areas, festival grounds, or outdoor assemblies, outdoor activities is sometimes prohibitive. In emergency search-and-rescue or military maneuvers, a temporary communication network also needs to be deployed immediately. In the above situations, a mobile ad hoc network (MANET) can be a better choice. A MANET consists of a set of mobile hosts operating without the aid of the established infrastructure of centralized administration (e.g., base stations or access points). Communication is done through wireless links among mobile hosts through their antennas. Due to concerns such as radio power limitation and channel utilization, a mobile host may not be able to communicate directly with other hosts in a single hop fashion.

Various routing protocols used in ad-hoc communication are known as Ad-hoc routing protocol. An ad hoc routing protocol is a convention, or standard, that controls how nodes decide which way to route packets between computing devices in a mobile ad-hoc network. In ad hoc networks, nodes are not familiar with the topology of their networks. Instead, they have to discover it: typically, a new node announces its presence and listens for announcements broadcast by its neighbour. Each node learns about others nearby and how to reach them, and may announce that it too can reach them.

The following is a list of some ad hoc network routing protocols:

a. Table-driven(Proactive) routing
b. On-demand(Reactive) routing
c. Hybrid(both Proactive and Reactive) routing
d. Hierarchical routing protocols

1.1 Table-driven Routing:

This type of protocol maintains fresh lists of destinations and their routes by periodically distributing routing tables throughout the network. Examples of proactive algorithms are:

1. Optimized Link State Routing(OLSR) Protocol RFC 3626
2. Babel RFC 6126
3. Destination Sequence Distance Vector(DSDV)

1.2 On-demand Routing:

This type of protocol finds a route on demand by flooding the network with Route Request Packets. Examples of on-demand algorithms are:

1. Ad hoc On-demand Distance Vector(AODV) RFC 3561
2. Dynamic Source Routing(DSR) RFC 4728
3. Flow state in the Dynamic Source Routing
4. Power-aware DSR-based

1.3 Hybrid(both Proactive and Reactive) routing:

This type of protocol combines the advantages of proactive and reactive routing. The routing is initially established with some proactively prospected routes and then serves the demand from additionally activated nodes through reactive flooding. The choice of one or the other method requires predetermination for typical cases.

2. LITERATURE SURVEY

Several pieces of key literature in the field of MANET routing protocols are identified which highlight existing protocols as well as the current thinking within the field and the directions in which the researchers are moving in the future.

As stated in the paper ‘A Neighbour Coverage-Based Probabilistic Rebroadcast for Reducing Routing Overhead in Mobile Ad Hoc Networks’ [1], the author(s) propose a Neighbour coverage-based probabilistic rebroadcast protocol for reducing routing overhead in MANETs. In order to effectively exploit the neighbour coverage knowledge, they propose a novel rebroadcast delay to determine the rebroadcast order, and then they obtained the more accurate additional coverage ratio by sensing neighbour coverage knowledge. They define a connectivity factor to provide the node density adaptation. By combining the additional coverage ratio and connectivity factor, they set a reasonable rebroadcast probability. Their approach was to combines the advantages of the neighbour coverage knowledge and the probabilistic mechanism, which can significantly decrease the number of re-transmissions so as to reduce the routing overhead, and can also improve the routing performance.

According to ‘A greedy-based stable multi-path routing protocol in mobile ad-hoc networks ‘ [2], emphasis is given on selecting a back-up route which is more stable and congestion-free. They have focused their discussion on topology stability routing which one of the classification of stable routing. It addresses route instability caused by link dynamics. They have also presented a Greedy - based Backup Routing (GBR) protocol to achieve high route stability.

The paper “ Neighbour Coverage: A Dynamic Probabilistic Route Discovery for Mobile Ad Hoc Networks” [3] states that conventional on-demand route discovery methods in MANET employs blind flooding, where a mobile node blindly rebroadcasts received Route Request (RREQ) packets until a route to a particular destination is established. This can potentially lead to high channel contention, causing redundant retransmissions and thus excessive packet collisions in the network. Such a phenomenon induces what is known as broadcast storm problem, which has been shown to greatly increase the network communication overhead and end-to-end delay. In this paper, we show that the deleterious impact of such a problem can be reduced if measures are taken during the dissemination of Route Request (RREQ) packets We propose two new probabilistic methods for on-demand route discovery, that is simple to implement and can significantly reduce the overhead involved in the dissemination of Route Request (RREQ) packets. Our analysis reveals that equipping AODV with probabilistic route discovery can result in significant performance improvements on all the standard metrics.

The Routing overhead minimization using the Dynamic source routing protocol is explained in the paper “A New Technique for Reducing Routing Overhead in MANET Using NCPR” [4].Here a new technique is proposed in which the NCPR Method is used to reduce the overhead in routing. In order to efficiently develop the formation of nearby coverage, the method NCPR is to determine rebroadcast delay in the rebroadcast order, and get the additional accurate average proportion by sensing neighbour coverage information. By using the NCPR technique in MANET, because of less redundant rebroadcast, the proposed protocol mitigates the network collision and contention, so as to increase the packet delivery ratio.

3. PROPOSED SYSTEM

Mobile Ad Hoc Network (MANETs) consist of a collection of mobile nodes which can move freely. These nodes can be dynamically self-organized into arbitrary topology networks without a fixed infrastructure. MANETs are highly dynamic network because nodes may join and leave the network at any time. It has limited network capacity. Quality of Service routing in such networks is usually limited by the network breakage due to either energy depletion or node mobility of the mobile nodes. Mobility of nodes may cause frequent link breakages which lead to frequent path failures and route discoveries. The overhead of a route discovery cannot be neglected. In this project, we have discussed a probabilistic rebroadcast for reducing routing overhead in a real-time MANET environment.

Also, we have proposed a new technique for reducing routing overhead in MANETs using NCPR. In order to efficiently develop the information of nearby coverage, the method NCPR is to determine rebroadcast delay in the rebroadcast order, and get the additional accurate coverage proportion by sensing Neighbour coverage information.

4. DETAILED DESIGN

NCPR is Neighbour Coverage-based Probabilistic Rebroadcast Routing Protocol. In this section, we discuss the detailed design of NCPR protocol. We calculate the rebroadcast delay and rebroadcast probability of the proposed protocol. We use the upstream coverage ratio of a Route Request (RREQ) packets received from the previous node to calculate the rebroadcast delay, and use the additional coverage ratio of the Route Request (RREQ) packets and the connectivity factor to calculate the rebroadcast probability in our protocol, which requires that each node needs its 1-hop neighbourhood information. In this section, we describe the algorithm for NCPR in detail.

4.1 Uncovered Neighbour Set and Rebroadcast Delay:

Uncovered Neighbour Set, as the name indicates, is a set of Neighbour maintained by each node during route discovery process. When node *ni* receives an RREQ packet from its previous node *s*, it can use the neighbour list in the Route Request (RREQ) packets to estimate how many its neighbours have not been covered by the Route Request (RREQ) packets from *s*. If node *ni* has more neighbours uncovered by the Route Request (RREQ) packets from *s*, which means that if node *ni* rebroadcasts the RREQ packet, the RREQ packet can reach more additional neighbour nodes. To quantify this, we define the *Uncovered neighbours set U (ni)* of node ni as follows:

$$U(ni) = N(ni) - [N(ni) \cap N(s)] - \{s\}$$

Where *N(ni)* and *N(s)* are the neighbours sets of node *ni* and *s*, respectively. *s* is the node which sends an Route Request (RREQ) packets to node *ni*.

According to above equation, we obtain the initial UCN(Uncovered neighbour set) set. Due to broadcast characteristics of a Route Request (RREQ) packets, node *ni* can receive the duplicate Route Request (RREQ) packets from its neighbours. Node *ni* could further adjust the *U (ni)* with the neighbour knowledge. In order to sufficiently exploit the neighbour knowledge and avoid channel collisions, each node should set a rebroadcast delay. The choice of a proper delay is the key to success for the proposed protocol because the scheme used to determine the delay time affects the dissemination of neighbour coverage knowledge. When a neighbour receives an Route Request (RREQ) packets, it could calculate the rebroadcast delay according to the neighbour list in the Route Request (RREQ) packets and its own neighbour list. The *rebroadcast delay Tp(ni)* of node *ni* is defined as follows:

$$Tp(ni) = 1 - \frac{|N(s) \cap N(ni)|}{|N(s)|}$$

$$Td(ni) = MaxDelay * Tp(ni)$$

Where *Tp(ni)* is the delay ratio of node *ni*, and MaxDelay is a small constant delay. | . | is the number of elements in a set.

The purpose of 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 neighbours will know this fact. Therefore, this rebroadcast delay enables the information that the nodes have transmitted the packet spread to more neighbours, which is the key to success for the proposed scheme.

4.2 Neighbour Knowledge and Rebroadcast Probability:

The node which has a larger rebroadcast delay may listen to Route Request (RREQ) packets from the nodes which have lower one. For example, if node *ni* receives a duplicate Route Request (RREQ) packets from its neighbour *nj*, it knows that how many its neighbours have been covered by the RREQ packet from *nj*. Thus, node *ni* could further adjust its UCN(Uncovered neighbour set) set according to the neighbour list in the Route Request (RREQ) packets from *nj*. Then, the *U(ni)* can be adjusted as follows:

$$U(ni) = U(ni) - [U(ni) \cap N(nj)]$$

After adjusting the *U(ni)*, the Route Request (RREQ) packets received from *nj* is discarded.

We do not need to adjust the rebroadcast delay because the rebroadcast delay is used to determine the order of disseminating neighbour coverage knowledge to the nodes which receive the same RREQ packet from the upstream node. Thus, it is determined by the neighbours of upstream nodes and its own.

When the timer of the rebroadcast delay of node *ni* expires, the node obtains the final UCN(Uncovered neighbour set). The nodes belonging to the final UCN(Uncovered neighbour set) set are the nodes that need to receive and process the RREQ packet. Note that, if a node does not sense any duplicate Route Request (RREQ) packets from its neighbourhood, its UCN(Uncovered neighbour set) set is not changed, which is the initial UCN(Uncovered neighbour set) set. Now, we study how to use the final UCN(Uncovered neighbour set) to set the rebroadcast probability.We define the *Rebroadcast Probability P(ni)* of node *ni* as:

$$P(ni) = F(ni) * R(ni)$$

$$F(ni) = \frac{Nc}{|N(ni)|}$$

$$R(ni) = \frac{|U(ni)|}{|N(ni)|}$$

Where *F(ni) and R(ni)* are the *Connectivity Factor* and *Additional Coverage Ratio* respectively, *Nc = 5.1774 log n*, and *n* is the number of nodes in the network.

This metric indicates the ratio of the number of nodes that are additionally covered by this rebroadcast to the total number of neighbours of node *ni*. The nodes that are additionally covered need to receive and process the RREQ packet. As *R* becomes bigger, more nodes will be covered by this rebroadcast, and more nodes need to receive and process the RREQ packet, and, thus, the rebroadcast probability should be set to be higher. Also, from the above equation of *F(ni)*, we can observe that when |*N(ni)*| is greater than *Nc*, *F(ni)* is less than 1. That means node *ni* is in the dense area of the network, then only part of neighbours of node ni forwarded the RREQ packet could keep the network connectivity. And when |*N(ni)*| is less than *Nc*, *F(ni)* is greater than 1. That means node *ni* is in the sparse area of the network, then node *ni* should forward the RREQ packet in order to approach network connectivity.

The above rebroadcast probability is defined with the following reason. Although the parameter *R* reflects how many next-hop nodes should receive and process the RREQ packet, it does not consider the relationship of the local node density and the overall network connectivity. The parameter *Fc* is inversely proportional to the local node density. That means if the local node density is low, the parameter *F* increases the rebroadcast probability, and then increases the reliability of the NCPR in the sparse area. If the local node density is high, the parameter *F* could further decrease the rebroadcast probability, and then further increases the efficiency of NCPR in the dense area. Thus, the parameter *F* adds density adaptation to the rebroadcast probability. Note that the calculated rebroadcast probability *P(ni)* may be greater than 1, but it does not impact the behaviour of the protocol. It just shows that the local density of the node is so low that the node must forward the RREQ packet. Then, node ni need to rebroadcast the RREQ packet received from s with probability *P(ni)*.

4.3 Algorithm:

The formal description of the Neighbour Coverage-based Probabilistic Rebroadcast for reducing routing overhead in route discovery is shown below:

Definitions:

RREQv: RREQ packet received from node *v*.

Rv.id: the unique identifier (id) of *RREQv*.

N(u): Neighbour set of node *u*.

U(u,x): Uncovered neighbours set of node *u* for RREQ whose id is *x*.

Timer (u,x): Timer of node *u* for Route Request (RREQ) packets whose id is *x*.

{Note that, in the actual implementation of NCPR protocol, every different RREQ needs a UCN(Uncovered neighbour set) and a Timer.}

1: **if** *ni* receives a new Route Request (RREQ) packets from *s* **then**

2: {Compute initial uncovered neighbours set *U(ni,Rs.id)*

For *RREQs* :}

3: $U(ni, Rs.id) = N(ni) - [N(ni) \cap N(s)] - \{s\}$

4: {Compute the *rebroadcast delay* $T_d(ni)$:}

5: $$Tp(ni) = 1 - \frac{|N(s) \cap N(ni)|}{|N(s)|}$$

6: $$Td(ni) = MaxDelay * Tp(ni)$$

7: Set a Timer (*ni*, *Rs.id*) according to $T_d(ni)$

8: **end if**

9:

10: **while** *ni* receives a duplicate $RREQ_j$ from *nj* before

Timer (*ni*, *Rs.id*) expires **do**

11: {Adjust *U(ni,Rs.id)*:}

12: $U(ni, Rs.id) = U(ni) - [U(ni) \cap N(nj)]$

13: discard (*RREQj*)

14: end **while**

15:

16: **if** Timer (*ni,Rs.id*) expires **then**

17: {Compute the *rebroadcast probability P(ni)*:}

18: $$R(ni) = \frac{|U(ni, Rs.id)|}{|N(ni)|}$$

19: $F(ni) = \frac{Nc}{|N(ni)|}$

20: $P(ni) = F(ni) * R(ni)$

21: **if** Random (0,1) <= *P(ni)* **then**

22: broadcast (*RREQs*)

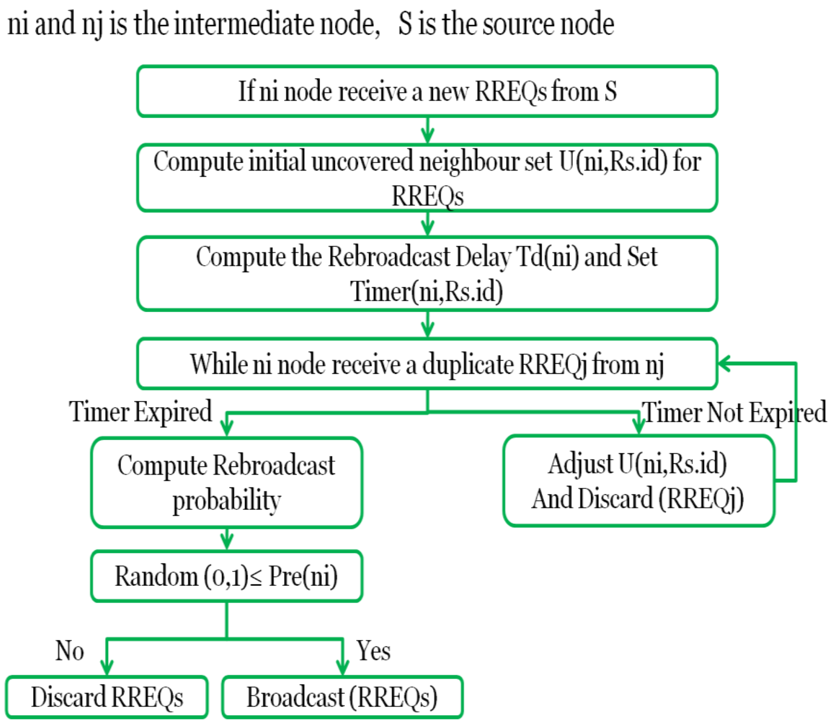
23: **else**

24: discard (*RREQs*)

25: **end if**

26: **end if**

4.4 Algorithm(Diagram):



4.5 Data Flow Diagram:

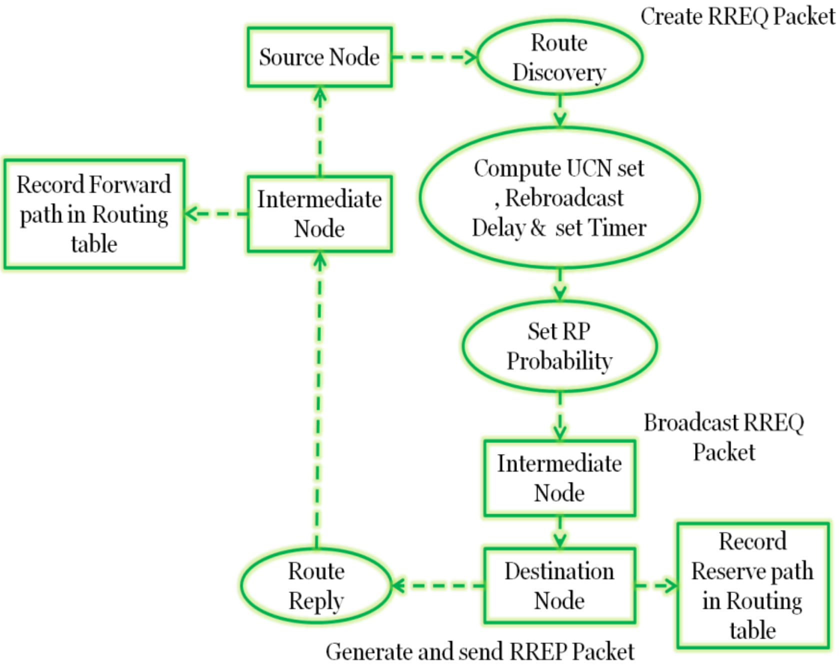


Fig.2 Data Flow Diagram

4.6 Architectural Design:

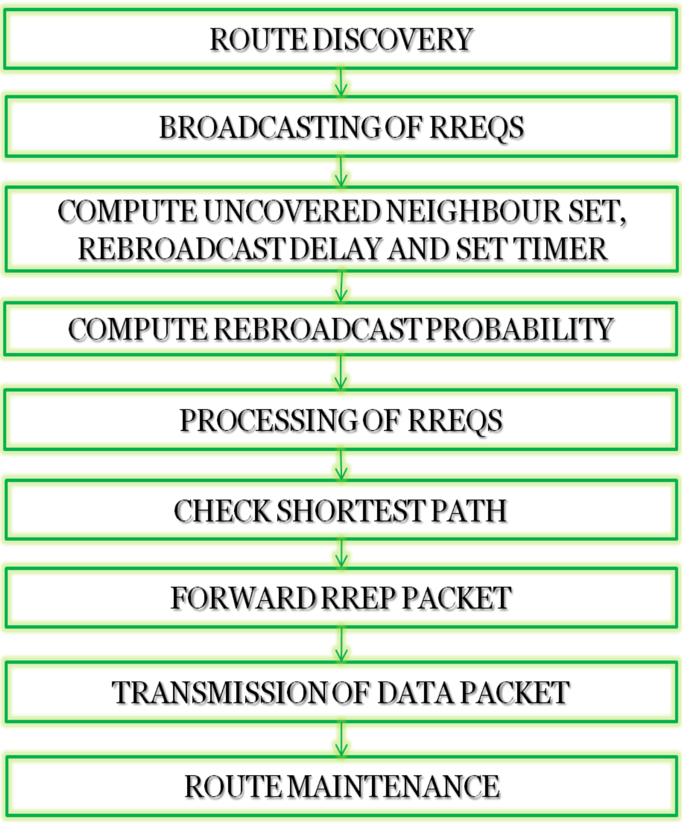


Fig.3 Architectural Design

4.7 PerformaNce Metrics:

We evaluate the performance of routing protocols using the following performance metrics:

1. ***MAC collision rate:*** The average number of packets (including route request (RREQ), route reply (RREP), route error (RERR), and CBR data packets) dropped resulting from the collisions at the MAC layer per second.
2. ***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, we use the size of Route Request (RREQ) packets instead of the number of Route Request (RREQ) packets, because the DPR and NCPR protocols include a neighbour list in the RREQ packet and its size is bigger than that of the original AODV.
3. ***Packet delivery ratio:*** The ratio of the number of data packets successfully received by the CBR destinations to the number of data packets generated by the CBR sources.
4. ***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 we evaluate the impact of one of the following parameters on the performance of routing protocols:

1. ***Number of nodes.*** We vary the number of nodes from 50 to 300 in a fixed field to evaluate the impact of different network density. In this part, we set the number of CBR connections to 15, and do not introduce extra packet loss.
2. ***Number of CBR connections***. We vary 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, we set the number of nodes to 150, and also do not introduce extra packet loss.
3. ***Random packet loss rate.*** We use the Error Model provided in the NS-3 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, we set the number of nodes to 150 and set the number of connections to 15.

5. CONCLUSION

In this paper, we proposed a probabilistic rebroadcast protocol based on neighbour coverage to reduce the routing overhead in MANETs. This neighbour coverage knowledge includes additional coverage ratio and connectivity factor. We proposed a new scheme to dynamically calculate the rebroadcast delay, which is used to determine the forwarding order and more effectively exploit the neighbour coverage knowledge. Because of less redundant rebroadcast, this protocol helps achieving our objectives, i.e., decrease the average end-to-end delay, reduces the routing overhead and also helps to increase the *Packet Delivery Ratio*.

ACKNOWLEDGMENT

The authors would like to thanks all the people associated with the project directly or indirectly. You all contributed to our work in some way. In addition, I would like to thank the authors of all the papers, journals and books that we have referred while compiling our work. Your work means a lot to us and we have shown that in our paper.

REFERENCES

[1] Xin Ming Zhang, Member, IEEE, En Bo Wang, Jing Jing Xia, and Dan Keun Sung, "A Neighbour Coverage-Based Probabilistic Rebroadcast for Reducing Routing Overhead in Mobile Ad Hoc Networks" Senior Member, IEEE. IEEE Transaction On Mobile Computing Vol. 12, No. 3, March 2013

[2] Wenjing Yang, Xinyu Yang, Shusen Yang, Dongxu Yang,"A greedy-based stable multi-path routing protocol in mobile ad-hoc networks", Deptt. of Computer ScieNce and Technology, Xi'an Jiaotong University, China, Elsevier, Ad Hoc Networks 9(2011) 662-674

[3] J.D. Abdulai, M. Ould-Khaoua, L.M. Mackenzie, and A. Mohammed, "Neighbour Coverage: A Dynamic Probabilistic Route Discovery for Mobile Ad Hoc Networks," Proc. Int'l Symp. PerformaNce Evaluation of Computer and Telecomm. Systems (SPECTS '08), pp. 165-172, 2008.

[4] R.Ranjith kumar, "A New Technique for Reducing Routing Overhead in Manet Using NCPR", in International Journal of Scientific & Engineering Research, Volume 4, Issue 3, March-2013, pp:1-5.