An Appraisal of Energetic Rebroadcast Probability using Neighbor-Coverage Knowledge for Reducing Routing Problem in MANET

Gowri. A
M.E – Computer Science and Engineering,
Kongunadu College of Engineering and Technology,Trichy, India

Lavanya. M
Assistant professor, Department of CSE,
Kongunadu college of engineering and technology,Trichy, India

ABSTRACT
Broadcasting is a vital process in ad hoc network whereby a source node sends the same packet to all the nodes in the network. A mobile node blindly rebroadcasts the first established route request packet unless it has a route to the endpoint or target resource, and thus it causes the broadcast storm delinquent. This broadcast storm problem will create the serious severance, disputation, and collision in a route discovery. Many approaches have been proposed method a Dynamic rebroadcast probability using neighbor coverage knowledge, the rebroadcast delay to determine the order and probability is set by merging the surplus coverage ratio and connectivity factor. The neighbor based coverage approach combines the advantages of the neighbor coverage knowledge and probable mechanism, which can expressively decrease the number of retransmission process so as to condense the routing overhead, and can also the routing performance.

Keywords
Broadcast storm, neighbor coverage, dynamic rebroadcast probability, rebroadcast delay.

1. INTRODUCTION
MANETs is formed dynamically by an autonomous system of mobile nodes that are linked via wireless network link without using a prevailing fixed network infrastructure or centralized administration. The nodes are free to move randomly and organize themselves arbitrarily; thus, the network's wireless topology may change promptly and arbitrarily. Nodes in MANETs act as end points and sometimes router to forward packets in a wireless multi-hop environment. One of the fundamental challenges in the design of dynamic routing protocol can efficiently establish routes to deliver the data packet between mobile nodes with minimum communication overhead compared to other network communication while ensuring high throughput and low end-to-end delay. Many routing protocols ,such as Ad-hoc On-demand Distance Vector Routing(AODV) and Dynamic Source Routing (DSR),have been proposed for MANETs. The above two protocols are on-demand routing protocols, and they could improve the scalability of MANETs by controlling the routing overhead when a new route is requested .The conventional on-demand routing protocols have used Flooding to realize a route. They broadcast a Route Request (RREQ) packet to the networks, and the broadcasting encourages extreme level of redundant retransmissions of RREQ packet and causes the broadcast storm problem , which leads to a considerable number of packet collisions, particularly in dense networks. Therefore, it is requisite elevate the broadcasting mechanism .Some of the methods have been optimized the broadcast problem in MANETs for past few years. Broadcasting protocols categorized into four classes: simple flooding, probability based methods, area-based methods, and neighbor knowledge methods .These four protocols are only suitable for static network , when the nodes are increased , it degrades the performance of the probability-based and area-based methods. showed that the performance of neighbor knowledge methods is better than that of area-based and probability-based ones.

In this paper, Dynamic Rebroadcast Probability Using Neighbor-Coverage Knowledge is proposing to limit the number of rebroadcasts and effectually achieve the route discovery without any conflict in the network communication. It combines the advantage of neighbor coverage knowledge and the probable mechanism. Therefore, 1) in order to efficiently achieve the neighbor coverage knowledge, and need a novel rebroadcast delay to determine the rebroadcast order, and then obtain more accurate result of the additional coverage ratio; 2) in order to keep the network connectivity as well as reduce the redundant retransmissions, and connectivity factor to determine how many neighbors should receive the RREQ packet. After the process etsablishment, 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.

2. PRELIMINARIES
2.1 Broadcasting in a MANET
MANET consists of a set of mobile hosts that may communicate with one another from time to time. No base stations are supported. Each host is equipped with a CSMA/CA (carrier sense multiple access with collision avoidance) transceiver. In such environment, a host may communicate with another directly or indirectly. In the latter case, a multi hop scenario occurs where the packets are instigated from the source host and transmitted by several intermediate hosts before reaching the destination or end point of the communication.

The broadcast problem refers to the sending of a message to other hosts in the network. The problem considered here has the following characteristics.

i. **The broadcast is unprompted or impulsive.** Any mobile host can concern a broadcast operation at any time. The reason has hosted mobility and the lack of synchronization, preparing any kind of global topology knowledge as prohibitive (in fact this is at least as hard as the broadcast problem). Little or no local evidence of information may be collected in advance.

ii. The **transmission is untrustworthy** .No acknowledgement mechanism will be used. Nevertheless, the network should be attempted and distribute the broadcast message to as many hosts as possible without paying too much effort. The motivations to make such an assumption are (a) a host may blunder or missing the broadcast message because it is off-line, it is provisionally isolated from the network, or it experiences monotonous collisions, (b) acknowledgements may cause serious medium contention (and thus another "storm") surrounding the sender, and (c) in many applications (e.g., the route discovery in), a 100% reliable broadcast is unnecessary.

2.2 Broadcast Storm Caused by flooding

A straightforward approach to perform the broadcast by flooding the request packet. The host receiving a broadcast message for the first time, and the obligation to rebroadcast the message. Clearly, this costs n transmissions in a network of it hosts. In a CSMAICA network, drawbacks of flooding include:

2.2.1 Redundant rebroadcasts

When a mobile host decides to rebroadcast a broadcast message to its neighbors, all its neighbors already have the message.

2.2.2 Contention

After a mobile host broadcast a message, if many of its neighbors decide to rebroadcast the message, these transmissions (which are all from nearby hosts) may severely contend with each other.

2.2.3 Collision

Because of the deficiency of back-off mechanism, the lack of RTS/CTS dialogue, and the absence of CD, collisions are more likely to occur and cause more damage.

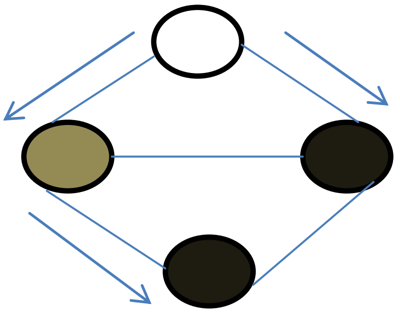
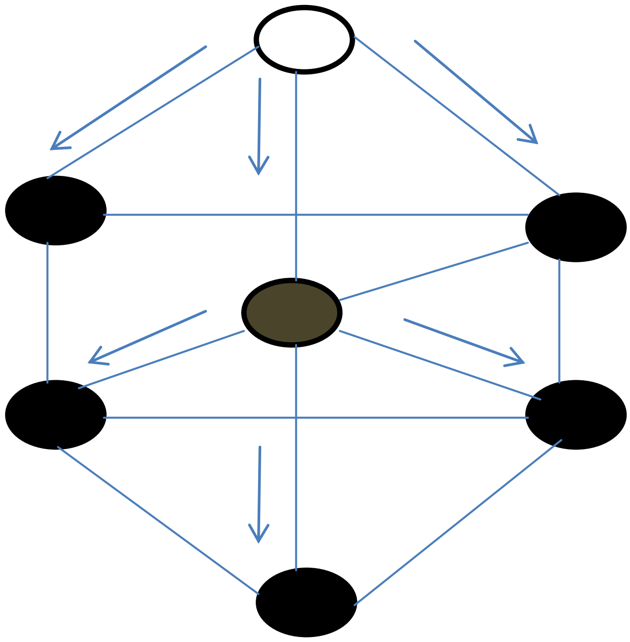


Figure 1: Two optimal broadcasting schedules in MANETs.

Connectivity between hosts is represented by links. White nodes are source hosts, and gray nodes are relay hosts. Collectively, we refer to the above phenomena as the broadcast storm problem. The following discussion shows how serious the storm is through analyses.

2.3 Ad Hoc On-demand Distance Vector Routing



The mobile ad hoc networks (MANETs), do not have fixed infrastructure. On Demand Distance Vector routing protocol is one of the most widely used routing protocols for an ad hoc network is the Ad hoc, abbreviated as AODV. In the AODV routing protocol, the source node forwards RREQ (Route Request) packet to discover the path to the destination node. The intermediate node having less lifetime, also forwards RREQ. As lifetime or energy expires after some time, the node goes failure; it could not forward RREP (Route Reply) on the reverse path. Therefore, source node has resumed RREQ rebroadcast to communicate with destination, which results in needless RREQ rebroadcast, less Packet Delivery Ratio (PDR) as well as more end to end delay and throughput.

2.4 Dynamic Source Routing protocol (DSR)

The Dynamic Source Routing protocol (DSR) is a simple as well as efficient routing protocol. Mostly same as AODV routing protocol .The rebroadcast possibility would be small while the number of nearby nodes is high which means the host is in opaque area . The possibility will be large when the amount neighbor nodes are small which means a host is in a low area.

3. RELATED WORK

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].The broadcasting protocol analytically and experimentally are 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. Thus, optimizing the broad- casting in route discovery is an effective solution to improve the routing performance. Haas et al. [7] 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.

Kim et al. [5] Proposed a probabilistic broadcasting scheme based on coverage area and neighbor confirmation. This scheme uses the coverage area to set the rebroadcast probability, and uses the neighbor confirmation to guarantee reachability. [8] 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. 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 by 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 to the DPR protocol.

4. DYNAMIC REBROADCAST PROBABILITY USING NEIGHBOR-COVERAGE KNOWLEDGE

4.1 Discovery of Neighbor-Coverage Knowledge

The node receives a RREQ packet from its previous node S, we use the neighbor list in the RREQ packet to estimate how many its neighbors have not been covered by the RREQ packet from Source(s). If the node has more neighbors uncovered by the RREQ packet from us, which means that if node a rebroadcasts the RREQ packet, the RREQ packet can reach more additional neighbor nodes. To quantify of the Uncovered Nimbus' (UCN) set up (a) of node a as follows

$U(u_i) = N(u_i) - [N(u_i) \cap N(s)] - \{s\}$----> (1)

where $N(s)$ and $N(u_i)$ are the neighbors sets of node s and u_i, respectively. s is the node which sends an RREQ packet to node u_i. According to Eq. (1),

We obtain the initial UCN set. Due to broadcast characteristics of a RREQ packet, the node we can receive the duplicate RREQ packets from its neighbors. Node u_i could further adjust the $U(u_i)$ with the neighbor knowledge.

In order to sufficiently exploit the neighborhood 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 neighbor coverage knowledge. When a neighbor receives a RREQ packet, it could calculate the rebroadcast delay according to the neighbor list in the RREQ packet and its own neighbor list. The rebroadcast delay Td(ui) of node ui is defined as follows:

$$T_p(u_i) = 1 - |N(s) \cap N(u_i)||N(s)|$$

$$T_d(u_i) = \text{MaxDelay} \times T_p(u_i) \text{ ,----------->(2)}$$

where $T_p(u_i)$ is the delay ratio of node ui, and MaxDelay is a small constant delay. | · | Is the number of elements in a set.

The above rebroadcast delay is defined with the following reasons: Firstly, the delay time is used to determine the node transmission order. To sufficiently exploit the neighbor coverage knowledge, it should be disseminated as quickly as possible. When node s sends a RREQ packet, all its neighbors a, I = 1, 2, · · · , |N (s) | receive and process the RREQ packet. We assume that node u_k has the largest number of common neighbors with node s, according to Eq. (2), node u_k has the lowest delay. Once node u_k rebroadcasts the RREQ packet, there are more nodes to receive it, because node u_k has the largest number of common neighbors. Then there are more nodes which can exploit the neighbor knowledge to adjust their UCN sets. Of course, whether node u_k rebroadcasts the RREQ packet depends on its rebroadcast probability calculated in the next subsection. The objective of this rebroadcast delay is not to rebroadcast the RREQ packet to more nodes, but to disseminate the neighbor coverage knowledge more quickly. After determining the rebroadcast delay, the node can set its own timer.

4.2 Adjustment of uncovered neighbor set

The node which has a more rebroadcast delay might listen to RREQ packets from the nodes, which have lesser one. For example, if node u_i receives a duplicate RREQ packet from its neighbour v_j , it knows that how many its neighbours have been covered by the RREQ packet from v_j . Thus, node ui could further adjust its UCN set according to the neighbor list in the RREQ packet from v_j . Then the $U(u_i)$ can be adjusted as follows:

$$U(u_i) = U(u_i) - [U(u_i) \cap N(v_j)] \text{----------->(3)}$$

After adjusting the $U(u_i)$, the RREQ packet received from v_j is discarded. Do not need to adjust the rebroadcast delay because the rebroadcast delay is used to determine the order of disseminating neighbor coverage knowledge to the nodes which receive the same RREQ packet from the upstream node. Thus, it is determined by the neighbors of upstream nodes and its own.

4.3 Dynamic Rebroadcast Probability

When the timer of the rebroadcast delay of node u_i expires, the node obtains the final UCN set. The nodes belonging to the final UCN set are the nodes that need to receive and process the RREQ packet. Note that, if a node does not sense any duplicate RREQ packets from its neighborhood, its UCN set is not changed, which is the initial UCN set. The additional coverage ratio $R_a(u_i)$.

$$R_a(u_i) = |U(u_i)| \text{ ----------> (4)}$$

$N(u_i)$

This metric indicates the ratio of the number of nodes that are additionally covered by this rebroadcast to the total number of neighbors of node ui. [10] Derived that if each node connects to more than 5:1774 $\log_n$ of its nearest neighbors, then the probability of the network being connected is approaching 1 as n increases, where n is the number of nodes in the network. Then, we can use 5:1774 $\log_n$ as the connectivity metric of the network.

$$F_c(u_i) = \frac{N_c}{|N(u_i)|} \text{ ---------->(5)}$$

Nc = 5.1774 $\log_n$, the n is the number of nodes inthe network. The Eq. (5), observe that when $|N(u_i)| > N_c$, Fc(ui) < 1. The means node ui is in the dense area of the network, then only part of neighbours of node u_i forwarded the RREQ packet could keep the network connectivity. And $|N(u_i)| < N_c$, $F_c(u_i) > 1$.

The means of node ui is in the sparse area of the network, then node ui should forward the RREQ packet in order to approach network connectivity. Combining the additional coverage ratio and connectivity factor, to obtain the rebroadcast probability Pre (ui) of node ui

Pre(ui)=Ra(ui).Fc(ui) -----------> (6)

Where, if the Pre (ui) is > 1, to set the Pre (ui) to 1.

4.4 Algorithm Description

The formal description of the Dynamic Rebroadcast Probability Using Neighbor-Coverage Knowledge For Reducing Storm Problem in route discovery is shown in Algorithm 1.

Algorithm 1

Definitions:
$RREQ_v$: RREQ packet received from node v.
R_v.id: the unique identifier (id) of $RREQ_v$.
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.
{Note that, in the actual implementation of NCPR protocol, every different RREQ needs a UCN set and a Timer.}

1. if u_i receives a new RREQs from s then
2. {Compute initial uncovered neighbors set $U(u_i,R_s.id)$ for $RREQ_s$:}
3. $U(u_i,R_s.id) = N(u_i) - [N(u_i) \cap N(s)] - \{s\}$
4. {Compute the rebroadcast delay $T_d(u_i)$:}
5. $T_p(u_i) = 1 - |N(s) \cap N(u_i)|\ |N(s)|$
6. $T_d(ui) = MaxDelay \times T_p(ui)$
7. Set a $Timer(u_i,R_s.id)$ according to Td(ui)
8. end if
9. while u_i receives a duplicate $RREQ_j$ from v_j before
10. $Timer(u_i,R_s.id)$ expires do
11. {Adjust $U(u_i,R_s.id)$:}
12. $U(u_i,R_s.id)= U(u_i,R_s.id) - [U(u_i,R_s.id) \cap N(v_j)]$
13. discard($RREQ_j$)
14. end while
15. if $Timer(u_i,R_s.id)$ expires then
16. {Compute the rebroadcast probability $P_{re}(u_i)$:}
17. $R_a(u_i) = |U(u_i,R_s.id)|\ |N(u_i)|$
18. $F_c(u_i)= N_c/|N(u_i)$
19. $P_{re}(u_i) = F_c(u_i) \cdot R(u_i)$
20. if Random(0,1) ≤ $Pre(u_i)$ then
21. broadcast(RREQs)
22. else
23. discard(RREQs)
24. end if
25. end if

4.5 Algorithm Implementation

This algorithm needs Hello packets to obtain the neighbor information, and also needs to carry the neighbor list in the RREQ packet. Therefore, in our implementation, some techniques are used to reduce the overhead of Hello packets and neighbor list in the RREQ packet, which are described as follows:

In order to reduce the overhead of Hello packets, we do not 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. We use the following mechanism [17] to reduce the overhead of Hello packets: Only when the time elapsed since the last broadcasting packet (RREQ, RERR, or some other broadcasting packets) is greater than the value of HelloInterval, the node needs to send a Hello packet. The value of HelloInterval is equal to that of the original AODV.

In order to reduce the overhead of neighbor list in 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. We modify the RREQ header of AODV, and add a fixed field num neighbor 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 ui has the following 3 cases:

i. If the neighbor table of node ui adds at least one new neighbor vj , then node ui 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 neighbors field in the RREQ packet. It is because that node vj may not have cached the neighbor information of node ui, and, thus, node vj needs the complete neighbour list of node ui;

ii. If the neighbor table of node ui deletes some neighbors, then node ui 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;

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

iv. if the num neighbors is a positive integer, the node substitutes its neighbor cache of node ui according to the neighbor list in the received RREQ packet;

v. if the num neighbors is a negative integer, the node updates its neighbor cache of node ui and deletes the deleted neighbors in the received RREQ packet;

vi. if the num neighbors is 0, the node does nothing. Because of the two cases 2) and 3), this technique can reduce the overhead of neighbor list listed in the RREQ packet.

5. CONCLUSION

Because of high mobility of the nodes in MANETS, always there is a greater chance of frequent link breakages between nodes. These frequent link failures will cause a number of rebroadcasts between nodes and broadcast storm problem .In this paper a Dynamic Rebroadcast Probability Using Neighbor-Coverage Knowledge For Reducing Storm Problem In MANET is discussed. This neighbor coverage knowledge includes additional coverage ratio and connectivity factor. Because of less redundant rebroadcast, the proposed protocol mitigates the network collision and contention, so as to increase the packet delivery ratio and decrease the average end-to-end delay.

6. REFERENCES

[1] C. Perkins, E. Belding-Royer, and S. Das, Ad Hoc On-Demand Distance Vector (AODV) Routing, IETF RFC 3561, 2003.

[2] D. Johnson, Y. Hu, and D. Maltz, The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR) for IPv4, IETF RFC 4728, vol. 15, pp. 153-181, 2007

[3] H. AlAamri, M. Abolhasan, and T. Wysocki, “On Optimizing Route Discovery in Absence of Previous Route Information in MANETs,” Proc. IEEE Vehicular Technology Conf. (VTC), pp. 1-5, 2009

[4] S.Y. Ni, Y.C. Tseng, Y.S. Chen, and J.P. Sheu, “The Broadcast Storm Problem in a Mobile Ad Hoc Network,” Proc. ACM/IEEE MobiCom, pp. 151-162, 1999.

[5] J. Kim, Q. Zhang, and D.P. Agrawal, “Probabilistic Broadcasting Based on Coverage Area and Neighbor Confirmation in Mobile Ad Hoc Networks,” Proc. IEEE GlobeCom, 2004.

[6] J.D. Abdulai, M. Ould-Khaoua, and L.M. Mackenzie, “Improving Probabilistic Route Discovery in Mobile Ad Hoc Networks,” Proc. IEEE Conf. Local Computer Networks, pp. 739-746, 2007.

[7] Z. Haas, J.Y. Halpern, and L. Li, “Gossip-Based Ad Hoc Routing,” Proc. IEEE INFOCOM, vol. 21, pp. 1707-1716, 2002.

[8] W. Peng and X. Lu, “On the Reduction of Broadcast Redundancy in Mobile Ad Hoc Networks,” Proc. ACM MobiHoc, pp. 129-130, 2000.

[9] J.D. Abdulai, M. Ould-Khaoua, L.M. Mackenzie, and A. Mo- hammed, “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.

[10] F. Xue and P.R. Kumar, “The Number of Neighbors Needed for Connectivity of Wireless Networks,” Wireless Networks, vol. 10, no. 2, pp. 169-181, 2004.