

A Reliable DTPR: Delay Tolerant Probabilistic Rebroadcast for Mobile Ad Hoc Networks*

Najiya A¹, Asst. Professor Reshma V. K²
Department of Computer Science
N.C.E.R.C, Pampady, Kerala

Abstract—High mobility of nodes in MANETs cause large routing overhead in route discovery. A Delay-Tolerant Probabilistic Rebroadcast has been proposed for reducing the routing overhead in MANETs. The DTPR mechanism calculates the rebroadcast delay which is used to determine the forwarding order and the rebroadcast probability can be used to determine the probable location of the nodes. But reliable data delivery in MANETs with high mobility of nodes remains an issue. If the data packet sent out is not relayed by the best forwarder within a certain period of time, then the neighbour nodes that have overheard the transmission will serve as forwarding candidates. This paper proposes such a type of in-the-air backup through which the communication is maintained without any interruption. Additional latency incurred by local route recovery can be greatly reduced and the duplicate relaying caused by packet reroute is also decreased.

Keywords—MANET; reliability of data ; routing overhead; latency;

I. INTRODUCTION

A mobile ad hoc network (MANET) is a self configuring infrastructure less network of mobile devices connected by wireless. Each node in a MANET is free to move independently in any direction, and will therefore change its location within network. As a result some stale routes are generated in the routing table which leads to unnecessary routing overhead. Topology-based traditional routing protocols in MANET are susceptible to node mobility. Quality of Service (QoS) routing in such networks is usually limited by the network breakage due to either node mobility or energy depletion of the mobile nodes. Routing overhead is the another issue that affects the QoS routing.

A delay-tolerant probabilistic rebroadcast is proposed to reduce the routing overhead in MANETs. The DTPR mechanism effectively exploit the neighbour coverage knowledge. The upstream coverage ratio of an RREQ packet received from the previous node is used to calculate the rebroadcast delay. The node which has more common neighbors with the previous node has the lower delay. The 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 neighbours and the connectivity factor which reflects the relationship of network connectivity and the number of neighbours of a given node is used to calculate the rebroadcast probability in the DTPR protocol.

The rebroadcast probability is used to reduce the number of rebroadcasts of the RREQ packet thus improving the routing performance.

Although the DTPR reduces much of the routing overhead in the dense network, it does not consider about the reliability of data packets. Due to the error prone wireless channel and the dynamic network topology, reliable data delivery in MANETs remains an issue. Existing topology-based MANET routing protocols are susceptible to node mobility. Traditional topology-based MANET routing protocols are quite susceptible to node mobility. The predetermination of an end-to-end route before data transmission is one of the main reason. It is very difficult to maintain a deterministic route due to the constantly and even fast changing network topology. The discovery and recovery procedures are also time and energy consuming. When the path breaks, data packets will be delayed for a long time or get lost until the route is reconstructed again, causing transmission interruption. In fact, due to the broadcast nature of the wireless medium, multiple reception occur as a result of a single packet transmission. The robustness of the routing protocol can be significantly enhanced by using such transmission as a backup.

In this paper, a novel Position-based Reliable Routing of the DTPR protocol is proposed. In this scheme the packet that has been received using MAC interception is cached by several forwarding candidates. In certain time slots if the best forwarder does not forward the packet, suboptimal candidates will take turn to forward the packet according to a locally formed order. In this way, the data transmission will not be interrupted, as long as one of the candidates succeeds in receiving and forwarding the packet.

The rest of this paper is organized as follows: Section 2 reviews the related work. Section 3 present the overview of the DTPR protocol. Section 4 discusses about the reliable routing mechanism incorporated into the DTPR protocol. Section 5 illustrates the performance of the proposed scheme by simulation and conclusion is given in section 6.

II. RELATED WORK

The most straightforward method is to provide some degree of redundancy to enhance a system's robustness. Existing

robust routing protocols for MANETs can be classified into two categories according to the degree of redundancy; one which uses the end-to-end redundancy for e.g., multipath routing, while the other uses the hop-by-hop redundancy which takes advantage of the broadcast nature of wireless medium and transmits the packets in an opportunistic or cooperative way. The PRR mechanism falls into the second category. Multipath routing, which is typically proposed to increase the reliability of data transmission [12] in wireless ad hoc networks, leads to the formation of multiple paths between the source and the destination. Existing multipath routing protocols are broadly classified into the following three types: 1) using alternate paths as backup (e.g., [10], [13], [14]); 2) packet replication along multiple paths (e.g., [13], [15]); and 3) split, reconstruction using some coding techniques [16] and multipath delivery. However, as discussed in [9], it may be difficult to find suitable number of independent paths. More importantly, due to constantly changing topology all paths may be broken with considerably high probability, making multipath routing still incapable of providing satisfactory performance, when the end-to-end path length is long..

In recent years, wireless broadcast is widely exploited to improve the performance of wireless communication. By using opportunistic overhearing, the connectivity between the mobile node and base station (BS) can be significantly improved in the context of infrastructure networks. The node that has higher packet delivery ratio to the destination will be preferred in relaying by assigning the higher priority relay a smaller contention window size.

III. DELAY TOLERANT PROBABILISTIC REBROADCAST PROTOCOL FOR MANETs

The rebroadcast delay is calculated by using the upstream coverage ratio of an RREQ packet received from the previous node. The rebroadcast probability in the DTPR protocol can be calculated by combining the additional coverage ratio of the RREQ packet and the connectivity factor.

A. REBROADCAST DELAY AND UNCOVERED NEIGHBOURS SET

When node n_i receives an RREQ packet from its previous node s , it can use the neighbour list in the RREQ packet to estimate how many of its neighbours have not been covered by the RREQ packet from s . If node n_i has more neighbours uncovered by the RREQ packet from s , it can reach more additional neighbour nodes. To quantify this, the Uncovered Neighbours set $U(n_i)$ of node n_i can be defined as follows:

$$U(n_i) = N(n_i) - [N(n_i) \cap N(s)] - \{s\} \quad (3.1)$$

Where $N(n_i)$ and $N(s)$ are the neighbours sets of node n_i and s respectively. The node s sends an RREQ packet to node.

According to (3.1), initial UCN set can be obtained. Due to the broadcast characteristics of an RREQ packet, node n_i can receive the duplicate RREQ packets from its neighbours. Node n_i could further adjust the $U(n_i)$ with the neighbour knowledge. Each node should set a rebroadcast delay in order to sufficiently exploit the neighbour

knowledge and avoid channel collisions. The key to success for the proposed protocol is the choice of a proper delay because the scheme determines the delay time affects the dissemination of neighbour coverage knowledge. It could calculate the rebroadcast delay according to the neighbour list in the RREQ packet when a neighbour receives an RREQ packet and its own neighbour list. The rebroadcast delay $T_{rd}(n_i)$ of node n_i is defined as follows:

$$T_p(n_i) = 1 - \frac{|N(s) \cap N(n_i)|}{|N(s)|}$$

(3.2)

$$T_{rd}(n_i) = \text{Max Delay} \times T_p(n_i),$$

Where $T_p(n_i)$ is the delay ratio of node n_i , and Max Delay is a small constant delay. $| \cdot |$ is the number of elements in a set.

The above rebroadcast delay is defined with the following reasons. The node transmission order can be determined by the delay time. When the node s sends an RREQ packet, all of its neighbours n_i , $i = 1, 2, 3, \dots, |N(s)|$ receive and process the RREQ packet. Assumed that node n_k has the largest number of common neighbours with node s , according to (3.2), node n_k has the lowest delay. Once this node rebroadcasts the RREQ packet, then more nodes can receive it, because node n_k has the largest number of common neighbours. Then, more number of nodes can exploit the neighbour knowledge to adjust their UCN sets. Of course, whether node n_k rebroadcasts the RREQ packet depends on its rebroadcast probability. The main objective of this rebroadcast delay is to disseminate the neighbour coverage knowledge more quickly. The node can set its own timer after determining the rebroadcast delay.

B. REBROADCAST PROBABILITY

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

$$U(n_i) = U(n_i) - [U(n_i) \cap N(n_j)] \quad (3.3)$$

After adjusting the $U(n_i)$, the RREQ packet received from n_j is discarded. The rebroadcast delay needs to be adjusted 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. The node obtains the final UCN set after expiring the timer of rebroadcast delay of node n_i . The final UCN set contains the nodes that need to receive and process the RREQ packet. If a node does not sense any duplicate RREQ packets from its neighbourhood, its UCN set is not changed, which is the initial UCN set.

The final UCN set can be used to set the rebroadcast probability. The additional coverage ratio ($R_{ac}(n_i)$) of node n_i can be calculated as follows:

$$R_{ac}(n_i) = \frac{|U(n_i)|}{|N(n_i)|} \quad (3.4)$$

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 n_i . The nodes that are additionally covered need to receive and process the RREQ packet. As R_{ac} becomes bigger, more nodes will be covered by the rebroadcast, and more nodes need to receive and process the RREQ packet. So the rebroadcast probability should be set to be higher.

If each node connects to more than $5.1774 \log n$ of its nearest neighbours, then the probability of the network being connected is approaching 1 as the number of nodes in the network n increases. Then, $5.1774 \log n$ can be used as the connectivity metric of the network. The ratio of the number of nodes that need to receive the RREQ packet to the total number of neighbours of node n_i is the connectivity factor $F_{cf}(n_i)$. In order to keep the probability of network connectivity approaching 1, have a heuristic formula

$$|N(n_i)| \cdot F_{cf}(n_i) \geq 5.1774 \log n.$$

The minimum $F_{cf}(n_i)$ can be defined as the connectivity factor, which can be found as follows:

$$F_{cf}(n_i) = \frac{N_{cv}}{|N(n_i)|} \quad (3.5)$$

where $N_{cv} = 5.1774 \log n$, and n is the number of nodes in the network. From (3.5), observed that when $|N(n_i)|$ is greater than N_{cv} , $F_{cf}(n_i)$ is less than 1. That means node n_i is in the dense area of the network, then only part of neighbours of node n_i forwarded the RREQ packet could keep the network connectivity. And when $|N(n_i)|$ is less than N_{cv} , $F_{cf}(n_i)$ is greater than 1. That means node n_i is in the sparse area of the network, then node n_i should forward the RREQ packet in order to approach network connectivity. The additional coverage ratio and connectivity factor can be combined to calculate the rebroadcast probability $P_{re}(n_i)$ of node n_i which can be defined as follows:

$$P_{re}(n_i) = F_{cf}(n_i) \cdot R_a(n_i) \quad (3.6)$$

where, if the $P_{re}(n_i)$ is greater than 1, then set the $P_{re}(n_i)$ to 1.

The above rebroadcast probability is defined with the following reason. The parameter R_a does not consider the relationship of the local node density and the overall network connectivity. The parameter F_{cf} is inversely proportional to the local node density. That means if the local node density is low, the parameter F_{cf} increases the rebroadcast probability which in turn increases the reliability of the DTPR in the sparse area. Thus, the parameter F_{cf} adds density adaptation to the rebroadcast probability.

Note that the calculated rebroadcast probability $P_{re}(n_i)$ may be greater than 1. It just shows that the node must forward the RREQ packet when the local density of the node is so low. Then, node n_i need to rebroadcast the RREQ packet received from s with probability $P_{re}(n_i)$.

C. ALGORITHM DESCRIPTION

The formal description of the Neighbour Coverage Based Probabilistic Rebroadcast for reducing routing overhead in route discovery is shown below.

Algorithm 1. DTPR

Definitions:

RREQq: RREQ packet received from node q .

$R_q.id$: the unique identifier (id) of RREQq.

$N(p)$: Neighbour set of node p .

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

Timer(p, x): Timer of node p for RREQ packet whose id is x .

1: if n_i receives a new RREQs from s then

2: {Compute initial uncovered neighbours set $U(n_i, R_s.id)$ for RREQs}

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

4: {compute the rebroadcast delay $T_{rd}(n_i)$ }

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

6: $T_{rd}(n_i) = \text{Max Delay} \times T_p(n_i)$

7: Set a Timer($n_i, R_s.id$) according to $T_{rd}(n_i)$

8: end if

9: while n_i receives a duplicate RREQj from n_j before

Timer($n_i, R_s.id$) expires do

10: {Adjust $U(n_i, R_s.id)$ }

11: $U(n_i, R_s.id) = U(n_i, R_s.id) - [U(n_i, R_s.id) \cap N(n_j)]$

12: discard (RREQj)

13: end while

14: if Timer($n_i, R_s.id$) expires then

15: {Compute the rebroadcast probability $P_{re}(n_i)$ }

16: $R_{ac}(n_i) = \frac{|U(n_i, R_s.id)|}{|N(n_i)|}$

17: $F_{cf}(n_i) = \frac{N_{cv}}{|N(n_i)|}$

18: $P_{re}(n_i) = F_{cf}(n_i) \cdot R_{ac}(n_i)$

19: if Random(0, 1) $\leq P_{re}(n_i)$ then

20: broadcast (RREQs)

21: else

22: discard (RREQs)

23: end if

24: end if

IV. PROPOSED WORK

A. CONCEPT OF RELIABLE ROUTING

The design of Reliable Routing is based on reliable forwarding and neighbour coverage. In this scheme it is assumed that the nodes are aware of the positions of their own and their direct neighbours. The information of the

location of neighbourhood can be exchanged using one-hop beacon in the header of the data packet. Also assume that a location registration and a lookup service which maps the addresses of node to the corresponding location is available for the position of the destination as in [5]. To transmit a packet, a source node gets the location of the destination first and attaches it to the header of packet. The multihop path may diverge from the true location of the final destination and a packet would be dropped even if it has already been delivered into the neighbourhood of the destination due to the destination node's movement.

An additional check for the destination node is introduced to deal with such issue. The node that forwards the packet will check its neighbour list at each hop to see whether the destination is within its transmission range. If it is in the range, the packet will be directly forwarded to the destination. In conventional opportunistic forwarding, either IP broadcast or an integration of routing and MAC protocol is adopted, to have a packet received by multiple candidates. Because of the lack of collision avoidance support for broadcast packet in current 802.11 the former is susceptible to MAC collision. The latter one requires complex coordination and is not easy to be implemented. In PRR, we use similar scheme as the MAC multicast mode described in [13]. The packet is transmitted as unicast (the best forwarder which makes the largest positive progress toward the destination is set as the next hop) in IP layer and multiple reception is achieved using MAC interception. All the nodes within the transmission range of the sender can eavesdrop on the packet successfully with higher probability due to medium reservation and the use of RTS/CTS/DATA/ACK significantly reduces the collision.

Each of the data packets that are transmitted in a multicast-like form, is identified with a unique tuple (source_ip, seqn_no) where source_ip is the IP address of the source node and seqn_no is the corresponding sequence number. A gradually increasing sequence number is maintained by every node, and an ID_Cache to record the ID (source_ip, seqn_no) of the packets that have been recently received. The packet will be discarded if it is received again with the same ID. Otherwise, if the receiver is the next hop it will be forwarded at once. If it is received by a forwarding candidate, it is cached in a Packet List

When the time slots exceeded the waiting limit the packet in the Packet List will be sent out or discarded if the same packet is received again during the waiting period (this implicitly means a better forwarder has already carried out the task). The reliable routing scenario can be simply illustrated in Fig. 1. Without link break that is the normal situation, the packet is forwarded by the next hop node (such as nodes E, A) and the forwarding candidates (such as nodes C, B; nodes G, F) will be suppressed (i.e., the same packet in the Packet List will be dropped) by the transmission of next hop node. In the case of, node A fails to deliver the packet (either node A has moved out or it cannot receive the packet), the forwarding candidate with the highest priority, the node B, will relay the packet and

suppress the forwarding of the lower priority candidate (node C) as well as node S. Node S will remove node A from the neighbour list and select a new next hop node for the subsequent packets by using the feedback from MAC layer. The packets in the interface queue will be given a second chance to reroute taking node A as the next hop. The packet pulled back from the MAC layer will not be rerouted as long as node S overhears node B's forwarding.

B. SELECTION AND PRIORITIZATION OF FORWARDING CANDIDATES

The key problem related to this scheme is the selection and prioritization of forwarding candidates. The nodes which are located in the forwarding area [14] would be considered as the backup nodes. The sender and the next hop node determines the forwarding area. The node which is located in the forwarding area must satisfy the two following conditions: 1) it must make positive progress toward the destination; and 2) its distance to the next hop node should not exceed half of the transmission range of a wireless node (i.e., $R=2$) so that ideally all the forwarding candidates can hear from one another.

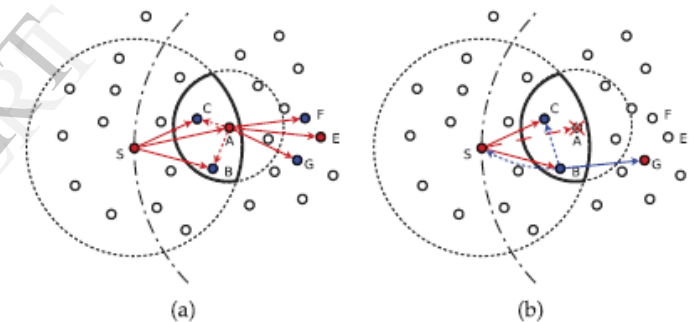


Fig.1. (a) Normal Operation. (b) Operation of routing when the next hop fails to receive the packet.

In Fig. 1, the bold curve which encloses the area is defined as the forwarding area. The nodes in this area, besides node A (i.e., nodes B, C), are potential candidates. The forwarding candidates are selected according to the required number of backup nodes. The forwarding candidate's priority is decided by its distance to the destination. The higher priority it will get when it is nearer to the destination. The node selects the next hop forwarder as well as the forwarding candidates among its neighbours when a node sends or forwards a packet. The next hop and the candidate list comprise the forwarder list. Algorithm 2 shows the procedure to select and prioritize the forwarder list. The packet header is attached with the candidate list and updated hop by hop. The forwarding candidates are the ones which are specified in the candidate list. The node has higher priority when the index of the node in the candidate list is lower.

Algorithm 2. Candidate Selection

```

ListNL : Neighbour List
ListCL : Candidate List, initialized as an empty list
DN: Destination Node
Breadth: Distance between current node and DN
if find(List NL,DN) then
    next_hop ← ND
    return
end if
for i ← 0 to length(List DN) do
    List NL[i].dist ← dist(List NL[i],DN)
end for
List NL.sort( )
next_hop ← List NL[0]
for i ← 1 to length(List NL) do
    if dist(List NL[i],DN) ≥ breadth (List CL)= N
    then
        break
    else if dist(list NL[i], list N[0] < R/2 then
        List CL.add(List NL[i])
    end if
end for

```

A forwarding table is maintained in every node for each flow of the packets which has forwarded or sent. A new forwarder list is calculated by looking the forwarding table to check whether the valid item for that destination is still available or not. During data packet transmissions the forwarding table is constructed and its maintenance is very easier than a routing table. Forwarding table takes much less time to be constructed since its establishment depends on local information. Therefore, an expire time can be set on the items maintained to keep the table relatively small. In other words, only the current active flows are recorded by the table. But in conventional protocols, when the time of route expires, it would require more resources to rebuild.

V. PERFORMANCE EVALUATION

A. SIMULATION ENVIRONMENT

Using the NS-2 simulator evaluated the performance of the proposed Reliable DTTPR and compared it with AODV AND DTTPR. The topology size taken for simulation is 1000m×1000m, in which 80 mobile nodes move for a simulation time of 900 seconds. The transmission range is 250 meters and the simulated traffic is Constant Bit Rate (CBR).

The parameters of simulation are summarized in table

TABLE 1

Parameter	Value
MAC Protocol	IEEE 802.11
Propagation Model	Two-ray Ground
Transmission Range	250 m
Mobility Model	Random Way Point (RWP)
Traffic Type	Constant Bit Rate (CBR)
Packet Size	256 Bytes
Number of Nodes	80
Simulation Time	900 sec

B. PERFORMANCE METRICS

The performance of routing protocols is evaluated using the following performance metrics:

- 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 destination
- Packet delivery ratio: The ratio of the number of data packets that are received by the CBR destinations 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.

C. RESULTS

In the first experiment we compare the routing overhead with varied number of nodes, in which the Reliable DTTPR yields 74.9% less overhead than the existing AODV and NCPR.

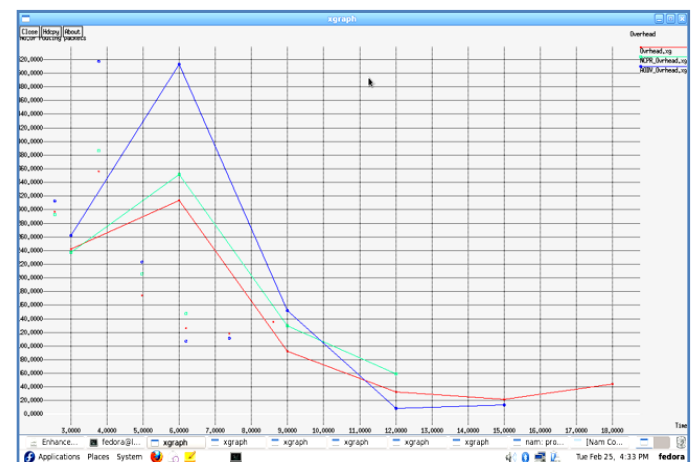


Fig.2.Comparison of routing overhead with AODV and NCPR

Secondly we compare the packet delivery ratio with a varied number of nodes which increases the packet delivery ratio higher than the AODV and NCPR protocol.

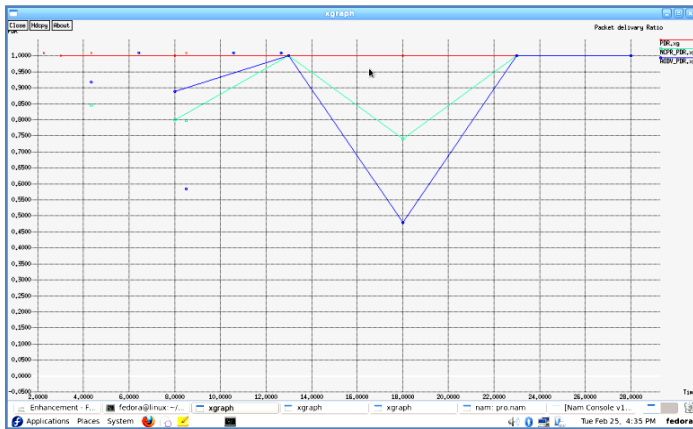


Fig.3. Packet Delivery Ratio

The throughput of Reliable DTPR is very much increased when compared to the AODV and NCPR approach, which is illustrated in the following figure no:4.

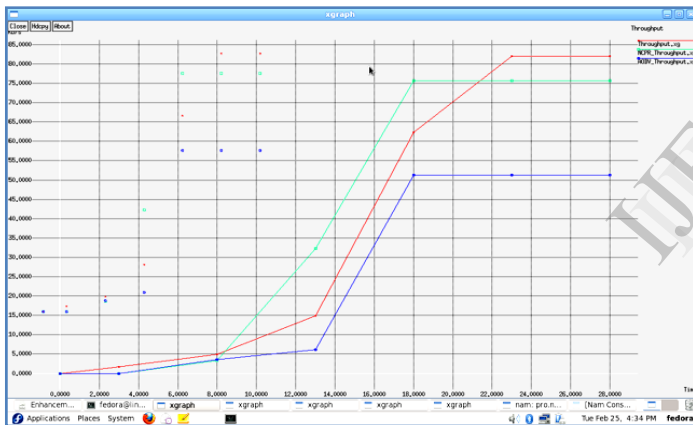


Fig.4.Enhanced throughput compared to AODV and NCPR

VI. CONCLUSION

In this paper, the problem of reliable data delivery in dynamic mobile ad hoc networks is addressed. Conventional ad hoc routing protocols incapable of providing satisfactory performance due to the constantly changing network topology. In the case of frequent link break age due to node mobility, substantial data packets would either experience long latency before restoration of connectivity or get lost. Hence proposed a novel reliable routing DTPR mechanism for mobile ad hoc networks which takes advantage of the neighbour coverage routing and the broadcast nature of wireless medium. Several forwarding candidates are also explicitly specified in case of link break, besides selecting the next hop. Leveraging on such natural backup in the air, the route which is broken can be recovered in a timely manner. The efficiency of the involvement of forwarding candidates against node mobility, as well as the overhead of forwarding is analyzed. The

effectiveness and efficiency of the Reliable DTPR mechanism is confirmed through simulation. High packet delivery ratio is achieved while the delay and overhead are lower.

REFERENCES

1. J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," Proc. ACM MobiCom, pp. 85-97, 1998.
2. M. Mauve, A. Widmer, and H. Hartenstein, "A Survey on Position-Based Routing in Mobile Ad Hoc Networks," IEEE Network, vol. 15, no. 6, pp. 30-39, Nov./Dec. 2001.
3. D. Chen and P. Varshney, "A Survey of Void Handling Techniques for Geographic Routing in Wireless Networks," IEEE Comm. Surveys and Tutorials, vol. 9, no. 1, pp. 50-67, Jan.-Mar. 2007.
4. D. Son, A. Helmy, and B. Krishnamachari, "The Effect of Mobility Induced Location Errors on Geographic Routing in Mobile Ad Hoc Sensor Networks: Analysis and Improvement Using Mobility Prediction," IEEE Trans. Mobile Computing, vol. 3, no. 3, pp. 233-245, July/Aug. 2004.
5. B. Karp and H.T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," Proc. ACM MobiCom, pp. 243-254, 2000.
6. S. Biswas and R. Morris, "EXOR: Opportunistic Multi-Hop Routing for Wireless Networks," Proc. ACM SIGCOMM, pp. 133-144, 2005.
7. S. Chachulski, M. Jennings, S. Katti, and D. Katabi, "Trading Structure for Randomness in Wireless Opportunistic Routing," Proc. ACM SIGCOMM, pp. 169-180, 2007.
8. E. Rozner, J. Seshadri, Y. Mehta, and L. Qiu, "SOAR: Simple Opportunistic Adaptive Routing Protocol for Wireless Mesh Networks," IEEE Trans. Mobile Computing, vol. 8, no. 12, pp. 1622-1635, Dec. 2009.
9. A. Balasubramanian, R. Mahajan, A. Venkataramani, B.N. Levine, and J. Zahorjan, "Interactive WiFi Connectivity for Moving Vehicles," Proc. ACM SIGCOMM, pp. 427-438, 2008.
10. K. Zeng, Z. Yang, and W. Lou, "Location-Aided Opportunistic Forwarding in Multirate and Multihop Wireless Networks," IEEE Trans. Vehicular Technology, vol. 58, no. 6, pp. 3032-3040, July 2009.
11. S. Das, H. Pucha, and Y. Hu, "Performance Comparison Of Scalable Location Services for Geographic Ad Hoc Routing," Proc. IEEE INFOCOM, vol. 2, pp. 1228-1239, Mar. 2005.
12. R. Flury and R. Wattenhofer, "MLS: An Efficient Location Service for Mobile Ad Hoc Networks," Proc. ACM Int'l Symp. Mobile Ad Hoc Networking and Computing (MobiHoc), pp. 226-237, 2006.
13. E. Felemban, C.-G. Lee, E. Ekici, R. Boder, and S. Vural, "Probabilistic QoS Guarantee in Reliability and Timeliness Domains in Wireless Sensor Networks," Proc. IEEE INFOCOM, pp. 2646-2657, 2005.
14. D. Chen, J. Deng, and P. Varshney, "Selection of a Forwarding Area for Contention-Based Geographic Forwarding in Wireless Multi-Hop Networks," IEEE Trans. Vehicular Technology, vol. 56, no. 5, pp. 3111-3122, Sept. 2007.
15. N. Arad and Y. Shavitt, "Minimizing Recovery State in Geographic Ad Hoc Routing," IEEE Trans. Mobile Computing, vol. 8, no. 2, pp. 203-217, Feb. 2009.
16. Y. Han, R. La, A. Makowski, and S. Lee, "Distribution of Path Durations in Mobile Ad-Hoc Networks - Palm's Theorem to the Rescue," Computer Networks, vol. 50, no. 12, pp. 1887-1900, 2006.