RLAR: Robust Link Availability Routing Protocol for Mobile Ad Hoc Networks

Xueyuan Su, Sammy Chan
Department of Electronic Engineering
City University of Hong Kong
Kowloon
Hong Kong
Email: {xysu, schan}@ee.cityu.edu.hk

King-Sun Chan
Department of Electrical and Computer Engineering
Curtin University of Technology
Western Australia 6845
Australia
Email: chank@ece.curtin.edu.au

***Abstract*— Many previously proposed routing metrics and algorithms for ad hoc networks work well in static networks, however, when nodes are moving and wireless links may fail from time to time, these routing metrics and algorithms are prone to poor performance. A cross-layer based routing protocol for mobile ad hoc networks, called Robust Link Availability Routing (RLAR) protocol, is proposed in this paper. RLAR consists of two modules, which estimates the link reliability using physical layer information and deals with robust optimal routing, respectively. Based on the optimal tree backbone, a mesh structure is formed for reliability enhancement. Through simulations, RLAR is proved to be able to increase the packet delivery ratio and reduce the frequency of reroutings for dynamic network topologies.**

***Index Terms*— Mobile ad hoc networks, robust routing, cross-layer design, link reliability**

I. INTRODUCTION

There has been very active research on routing in mobile ad hoc networks (MANETs) and many routing metrics and protocols were proposed in the past few years, for both unicast and multicast transmissions. However, the reliability of transmission remains a problem. Due to the mobility of nodes, the topology of a MANET changes dynamically, which poses challenges for reliable transmission. Because of the resource-limited nature of MANETs, too frequent rerouting will cause large overhead and packet loss, which degrade the end-to-end reliability.

As [2] indicates, static and mobile wireless networks can present two very different sets of challenges, and solutions that work well in one setting are not guaranteed to work just as well in another. It also has been shown that many recently proposed routing metrics, such as ETX [17], can only outperform the simple hop count metric in static topologies. When nodes are mobile and network topology changes dynamically, although using hop count metric does not lead to good performance, yet its performance is better than such link-aware metrics. At this standpoint, specially designed link-aware metrics for mobile environment are in urgent need.

In this paper, we propose a new routing metric called *normalized link availability* (NLA) which measures the reliability of a link. NLA is based on the link availability defined in [15] that indicates the probability of a link being continuously available during a given period of time. Compared to the previously proposed reliability metrics, such as link lifetime [8][9], NLA is more accurate since besides historical information, it is also based on predictions. Then, we propose a new robust routing protocol called Robust Link Availability Routing (RLAR) protocol for both unicast and multicast. Based on a cross-layer approach, one of the modules of RLAR calculates NLA from estimated physical layer information. The other module of RLAR takes charge of the route discovery and maintenance based on this reliability metric. The objective of our work is to increase the packet delivery ratio and reduce the frequency of reroutings. The performance of RLAR protocol is evaluated with promising results.

The rest of this paper is organized as follows. Section II reviews some other reliable multicast protocols for MANETs. Section III depicts the framework of RLAR. Then Section IV and Section V describe the two modules of RLAR, respectively. Performance evaluation is conducted in Section VI and Section VII concludes this paper.

II. RELATED WORK

Generally speaking, there are two different approaches to improve the reliability of multicast, one focuses on the MAC level [3][4], i.e., reliable transfer of data across single-hop wireless links; the other one is targeted at the network level. More specifically, the second approach achieves reliable multicast mainly from two aspects, i.e., loss recovery [5][6] and reliable routing [7][8][9]. Loss recovery ensures reliable end-to-end delivery of unreliable multicast packets through acknowledgments and retransmission. Reliable routing aims to build multicast routes with reliable links.

[3] presents an extension to the IEEE 802.11 MAC layer protocol called 802.11MX to provide link level reliability. It adopts NAK/NCTS and dual busy tones to reduce packet collisions and deal with packet loss recovery. Similar to [3], [4] also uses the busy tones to realize multicast reliability. It proposes RMAC to achieve full reliability which uses positive feedback ACKs. The limitation of RMAC is that it has to deal with multiple feedbacks.

The work described in this paper was supported by a grant from City University of Hong Kong (Project No. 7002017).

MAODV is proposed in [10], which is an extension of the well-known unicast routing protocol AODV. It is a tree-based routing protocol which finds the multicast route with minimal cost, such as number of hops, based on the distance vector approach. However, it does not specifically consider multicast routing from the reliability perspective. [5] implements anonymous gossip over MAODV to improve the reliability performance. When packet loss occurs in one receiver, a request for a copy of the lost packet is sent to a randomly chosen member of this multicast group. In [6], based on the multicast tree, FAT (family ACK tree) is proposed. When a link of the multicast tree fails, the fragmented parts of the tree will be glued back to the tree using the FAT protocol. Automatic repeat request (ARQ) is used for the recovery of lost packets. [7] proposes a method called Independent-tree ad hoc Multicast Routing which, based on the concept of alternate path routing, simultaneously constructs more than one multicast trees with minimal overlap. Since the trees are independent, if one of them fails, packet may be delivered via an alternative tree without invoking rerouting. However, since the tree construction does not take into account the reliability metric, this method fails to rule out the possibility that all the trees in the set have weak links and therefore are not reliable. This leads to the situation that the alternate paths also fail when they are needed. In addition, with multiple independent trees constructed, the size of the routing table in each node grows significantly. [8] also constructs more than one multicast trees for reliable enhancement, however, different from [7], the multiple multicast trees are of high correlation, since they share some most reliable links as the backbones.

Previous schemes have proposed many routing metrics. The three metrics evaluated in [2], i.e., expected transmission count (ETX), per-hop RTT, and per-hop packet pair, could only have good performance in static networks. Based on historical node staying time, [9] proposes a metric called “link lifetime”. [13] incorporates both signal stability and location stability to quantify the quality of the wireless link. The link quality is estimated by statistically analyzing the historical information of the link existence and no prediction is involved. Due to the dynamic property of mobile ad hoc networks, these metrics based on historical information may lead to misjudgement of link quality. [14] proposes a probabilistic link availability model to predict the accumulative availability probability during a period of time. In other words, in this estimation, a link is allowed to be broken down during one or more intervals. Therefore, it is not feasible for routing algorithms because usually after a link failure, a rerouting procedure will be performed. Based on the same assumptions as in [14], [15] tries to estimate the link availability by considering both the constant movement and possibly changing movement patterns. Although the link availability is indicated to be predicted from the continuous available time, how to generate this time value is not mentioned in [15].

III. THE FRAMEWORK OF RLAR PROTOCOL

The RLAR protocol is a hybrid protocol which combines pro-active routing with on-demand routing. Each node in the network maintains two tables. The first one is Link Quality Table (LQT) which stores the quality information of wireless links from the node to its one-hop neighbors. Due to the mobility of nodes in MANETs, the content of LQT should be refreshed from time to time. It is proposed here to use the beaconing mechanism for general update of LQT. With the use of the beacon signal, this part of RLAR is similar to the pro-active schemes. The other table is Routing Table (RT) as mentioned in other routing protocols. In RLAR, the information in RT is updated when a transmission session is initiated. In other words, the routing procedure uses the mechanism of on-demand routing. Therefore, RLAR consists of two modules: (1) link reliability estimation, and (2) route discovery and maintenance. Besides being a hybrid combination of both pro-active routing and on-demand routing, RLAR is also a cross-layer based protocol, since the link reliability is measured directly from physical layer information and forwarded to higher layers. The framework of cross-layer design is shown in Figure 1. In the next two sections, the two modules are discussed in detail respectively.

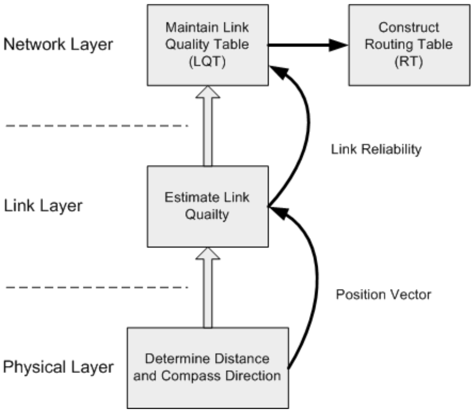


Fig. 1: Cross-layer design framework.

IV. MODULE I: LINK QUALITY ESTIMATION

We discuss the first module of RLAR in this section. For some existing routing algorithms, such as MAODV [10], it is assumed that the best route between two nodes is the one with minimum number of hops. This may be reasonable and convenient in wireline networks, however, in mobile ad hoc networks, due to the dynamic topology and node mobility, links are not always available. Low reliability route will lead to frequent rerouting and cause interruption of communication. This is intolerable in resource-limited wireless networks. Therefore, we give a high priority to route reliability. As it will be shown, the information from physical layer can be used to estimate the link quality between each pair of the neighboring wireless nodes. RLAR relies on it to update LQT and determine the optimal route for robust routing.

A. Physical layer information collection and position vector construction

We first describe how to collect information from physical layer and construct the position vector based on such information. Two assumptions are made as follows.

(1) The shadowing model [16] is adopted as the propagation model. Transmission channel could be modelled by the equation

$$P_r = \frac{P_t \times G_t \times G_r \times \lambda^2}{16 \times \pi^2 \times D^2 \times L} \tag{1}$$

In other words, the receiving power P_r is only related to the transmission power P_t, transmission gain G_t, receiving gain G_r, wavelength λ, transmission distance D and system loss L. Usually, P_t, G_t, G_r, λ and L are fixed values. Therefore, the only factor that affects the receiving power P_r is transmission distance D. Based on this assumption, a node can determine the distance to its neighbor from the receiving power according to Equation (1).

(2) The antenna of one node can determine the compass direction θ from which the other node's signal is arriving. This technology is known as AOA (Angle of Arrival) and has been shown feasible in GSM network positioning. Actually, the wireless channel in wireless ad hoc network is much better than that in GSM network, since the transmission range is much smaller (usually maximal 300m) and the channel suffers less from those problems such as channel fading and multi-path effect. Therefore, in wireless ad hoc networks, it is easier to determine the compass direction from which the signal is arriving. Due to limited space here, we omit the details of AOA. Interested readers are referred to [11][12].

Based on the two assumptions made above, a node could collect information from physical layer to determine the distance D to another node and the compass direction θ from which the other node's signal is arriving. A *position vector* $\overrightarrow{L_{mn}}$ from the receiving node m to the transmitting node n can be derived from these two parameters, with $|\overrightarrow{L_{mn}}| = D$ and $\angle\overrightarrow{L_{mn}} = \theta$. The construction of position vector is shown in Figure 2.

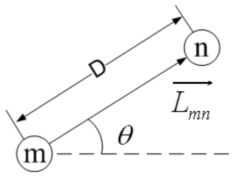


Fig. 2: Position vector.

In order to enable each node to collect accurate position vectors from one-hop neighbors, we propose a beaconing mechanism for each node to actively broadcast the beacon signal in the neighborhood. The objective of broadcasting beacon signal is only to let one-hop neighbors know the current position of the sender. Therefore, the beacon signal could be carried by a small beacon packet which is transmitted in the same frequency band as data packets, or by a out-of-band signal such as a narrowband tone. In order to avoid contention among beacon signals from many neighbors, the inter-transmission time of two beacon signals of each node is uniformly distributed in the range of $[0.5T, 1.5T]$, where T is the average inter-transmission time of beacon signals. In addition, in order to minimize the overhead caused by beacon signals, if a node has just sent out a data or control (ACK, RTS, and CTS) packet, this packet could be used for determining position vector and the next beacon signal could be skipped. In this way, the number of beacon signal transmissions is reduced. Due to the beaconing mechanism, the link quality information stored in LQT is refreshed actively.

B. Calculation of the continuous available time T_p

We now show how to calculate the continuous available time T_p via the position vectors constructed from the physical layer information. Without loss of generality, assume that at time $t = 0$, node m and node n are at positions m_0 and n_0, respectively. Let us consider nodes m and n are both in movement. As shown in Figure 3, after a short period of time τ, node m moves to $m^{'}$, and node n moves to $n^{'}$, both with constant velocity and direction.

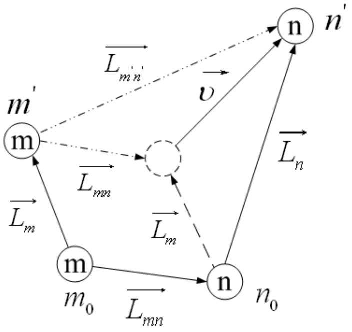


Fig. 3: Translation of relative movement vector.

The movement vectors are $\overrightarrow{L_m}$ and $\overrightarrow{L_n}$, respectively. Now we translate the two nodes' movements into one combined movement by fixing the position of one node, say, node m. From the viewpoint of node m, the relative movement of node n follows the vector $\overrightarrow{\nu}$, which we call *the relative movement vector*. It is obvious that $\overrightarrow{\nu} = \overrightarrow{L_n} - \overrightarrow{L_m}$. However, $\overrightarrow{L_n}$ and $\overrightarrow{L_m}$ are not known due to the distributed structure of MANETs. Instead, we can use another approach, i.e.,

$$\overrightarrow{\nu} = \overrightarrow{L_{m'n'}} - \overrightarrow{L_{mn}} \tag{2}$$

As discussed above, the position vectors $\overrightarrow{L_{m'n'}}$ and $\overrightarrow{L_{mn}}$ could be constructed from physical layer information at node m at $t = 0$ and $t = \tau$, respectively. By Equation (2), we obtain the relative movement vector from the position vectors. In this sense, the combined movement of both nodes could be interpreted as that while node m is fixed, node n is moving along the vector $\overrightarrow{\nu}$. Based on $\overrightarrow{\nu}$ and the assumption that both nodes maintain the directions and velocities of their movements, the trajectory of node n seen by node m is shown in Figure 4.

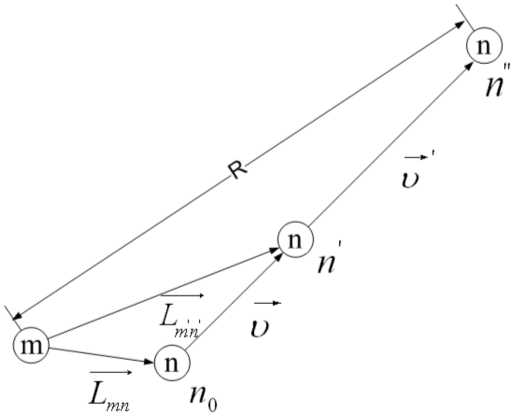


Fig. 4: Trajectory of node n relative of node m.

Following the trajectory, node n can always communicate with node m until it reaches position n'', which is at a distance R from m', where R is the transmission range of each node. Let us denote the time taken by node n to move from n' to n'' as T_p, which is clearly the continuous available time of the wireless link between node m and node n from $t = \tau$. Since $\overrightarrow{L_{mn}}$, $\overrightarrow{L_{m'n'}}$, $\vec{\nu}$ and τ are all available, and that $|\vec{\nu}'|$ can be obtained from R and $|\overrightarrow{L_{m'n'}}|$, T_p can be calculated by

$$T_p = \frac{|\vec{\nu}'|}{|\vec{\nu}|} \times \tau \tag{3}$$

C. Link reliability estimation

Once T_p is known, we can calculate the probability that a link is available during the time period of T_p using the approach developed in [15]. Assuming a random walk-based mobility model, each node moves in a constant direction at a constant speed during a time interval called mobility epoch. The epoch length of each node is exponentially distributed with mean λ^{-1}. The direction is uniformly distributed over $[0, 2\pi)$ and the speed is also uniformly distributed in a given range. Also assuming that speed, direction and epoch length are uncorrelated, mobility of nodes are uncorrelated and links fail independently. Based on all the assumptions above, a conservative prediction of the link being available in the time period of T_p is given in [15] as

$$L(T_p) = \frac{1 - e^{-2\lambda T_p}}{2\lambda T_p} + \frac{\lambda T_p e^{-2\lambda T_p}}{2} \tag{4}$$

To take into account the reliability of a link when selecting routes, we need a metric to reflect this aspect and whose value should lie in the range [0,1]. Although $L(T_p)$ satisfies these criteria, it is not suitable for our purpose because it is associated with a time scale of T_p. That is, it provides link availability estimation for the time interval T_p. In general, each link will have a different T_p and thus it is impossible to aggregate the $L(T_p)$ of each link along a path to estimate the path availability. Instead of using $L(T_p)$, we start off with the term $T_p \times L(T_p)$, which is an estimation of average available time of a link. Assuming that estimation will be carried out regularly with period T_r, we can confine our interest of the estimation within T_r. In other words, we are more interested to know the ratio of $T_p \times L(T_p)$ to T_r. When this ratio is smaller than one, a link with a higher value is more reliable than a link with a lower value. On the other hand, when two links have $\frac{T_p \times L(T_p)}{T_r} \geq 1$, both of them can be regarded as always available in the time period T_r and there is no need to distinguish the availability of two such links. Based on the above reasoning, we propose a new routing metric which is referred to as *normalized link availability* (NLA) and its definition is as follows:

$$NLA = \min[\frac{T_p \times L(T_p)}{T_r}, 1] \tag{5}$$

Once NLA is obtained, it is then forwarded to network layer and stored in LQTs.

Note that T_r is a tunable system parameter and could be related to R/S, where S is the maximum speed at which nodes can move. As R/S is the minimum time that a node would take to move out of the transmission range of another node, we may set T_r to be a fraction of this value, say, $\frac{R/S}{10}$ or $\frac{R/S}{5}$, depending on how often we want to monitor the availability.

V. MODULE II: ROUTE DISCOVERY AND MAINTENANCE

Functionally, three phases are defined in the second module of RLAR: the initialization phase, the operation phase, and the recovery phase. When a unicast or multicast transmission is initiated, RLAR triggers the initialization phase to discover the optimal route based on the link quality information stored in LQTs. After the route is successfully constructed, operation phase begins and data packets are delivered to the receivers. When packet loss occurs, RLAR works in recovery phase and tries to perform local recovery first. If the situation keeps on becoming worse and more packets are lost, RLAR triggers rerouting procedure and enter the initialization phase. Next, we will discuss the route discovery in detail.

A. Formation of a robust route

Assuming that each wireless link fails independently, the reliability of a single-chain route P is then given by

$$R_e(P) = \prod_{l \in P} NLA_l, \tag{6}$$

where NLA_l is the normalized link availability of link l.

When a transmission session is initiated, the source node triggers the initialization phase to discover the optimal route. The route discovery phase works in two steps: (1) construct the optimal tree structure (for unicast, there is only one leaf in the tree) with routes of highest reliability; (2) according to a reliability threshold, make additional protections for some vulnerable links in the tree backbone, thus forming the ultimate mesh structure. Usually this is the partially connected mesh, rather than the fully connected one. By forming the mesh structure, packets may be forwarded through shorter path. In addition, the overall reliability is improved.

In order to simplify the demonstration of RLAR, we take a simple unicast scenario as an example. The network topology is shown in Figure 5. The value on each link is the NLA estimated from the first module. The source node S intends to send packets to destination node D. If no route information is available, it will trigger the route discovery.

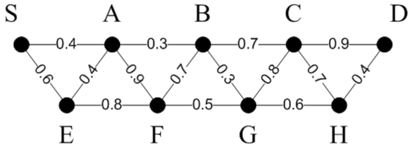


Fig. 5: Unicast network topology.

If the number of hops is used as the cost metric, the best route is the one with minimal number of hops, which is $S \rightarrow A \rightarrow B \rightarrow C \rightarrow D$, as shown in Figure 6. However, from the viewpoint of reliable transmission, this route is not the best one. According to Equation (6), the reliability of this route is $R_e = 0.4 \times 0.3 \times 0.7 \times 0.9 = 0.0756$.

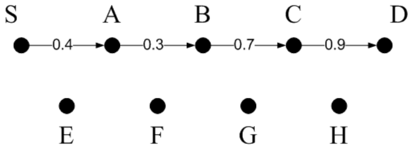
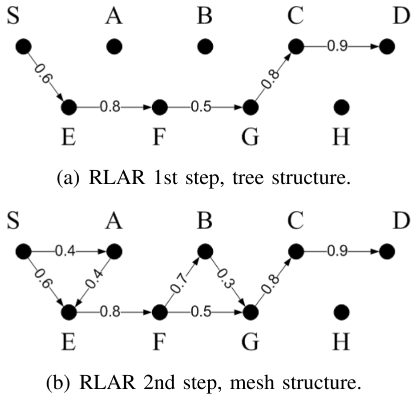


Fig. 6: Route with minimal number of hops.

RLAR algorithm regards NLA as the primary metric, so the best route is the one with the highest reliability, but not necessary the one with minimal number of hops. For this network topology, RLAR algorithm first constructs the optimal tree structure $S \rightarrow E \rightarrow F \rightarrow G \rightarrow C \rightarrow D$, which is shown in Figure 7(a). The reliability of this route is $R_e = 0.6 \times 0.8 \times 0.5 \times 0.8 \times 0.9 = 0.1728$, which is much better than that of the minimal-hop route.

Then, in the second step, RLAR examines each link in the tree structure. For some links, if the NLA is smaller than a given threshold L_r, e.g. $L_r = 0.8$, protections for such links are made. Redundant paths are set up to protect these relatively weaker links. Note that there is no resource reservation in the redundant paths, only the route information is stored in RTs of corresponding nodes. When delivering a packet, if the weak links in the tree backbone fail, the packet can be delivered through the protection path. After this step, the tree structure becomes a mesh structure, which is shown in Figure 7(b).

As a result, the reliability of this route is $R_e = [1-(1-0.6)(1-0.4\times 0.4)] \times 0.8 \times [1-(1-0.5)(1-0.7\times 0.3)] \times 0.8 \times 0.9 = 0.2314$, which is better than the one we obtain in the first step.



(a) RLAR 1st step, tree structure.

(b) RLAR 2nd step, mesh structure.

Fig. 7: Routing result of RLAR.

B. Implementation issues

In the previous section, we have demonstrated the procedure of route discovery. Here we discuss some related implementation issues.

(1) How to construct the optimal tree structure?

This can be achieved by simply adopting an flooding approach similar to that of DSR. Let us consider the network depicted in Figure 5. Node S broadcasts RREQ messages to A and E. Instead of the number of hops, the information included in each RREQ message is called *cumulative path reliability*, which is the product of the traversed link reliabilities. Node S broadcasts RREQ message to node A and node E. Node A then inserts (S-A 0.4) into the RREQ message and broadcast it, while Node E inserts (S-E 0.6) into the RREQ message. When the RREQ message from node A arrives at node E, E inserts (S-A-E 0.16) and compares the value with the previous one. It is obvious that $0.16 < 0.6$, so node E knows it is a worse path and discards the RREQ message from node A. All the intermediate nodes repeat the same procedure. When all the RREQ messages arrive at node D, it compares the values in each packet and chooses the maximal one. The path included in this packet is therefore part of the optimal tree structure which has the leaf as node D. Obviously, this approach does not introduce additional overhead compared to DSR and AODV.

(2) How to construct and maintain the mesh structure?

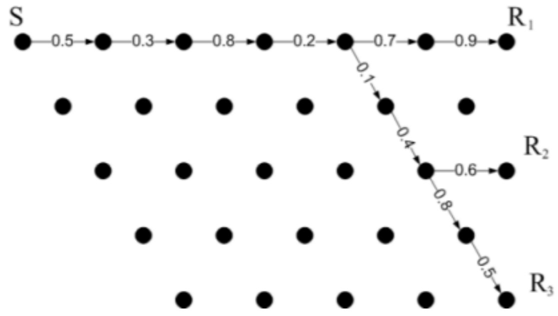
In addition to the optimal tree structure, additional protections are made for some vulnerable links, which leads to the mesh structure. First, an reliability threshold L_r is used to determine whether a link is reliable enough. If a certain link with reliability smaller than L_r, e.g., in Figure 5 link S-E with $NLA = 0.6 < L_r = 0.8$, the redundant path S-A-E is included to protect it. This is done by including corresponding route information in the RTs of node S and node A. Similarly, by including route information in the RTs of node F and node B, protection path F-B-G is provided for link F-G. The value of L_r determines the number of protected links and therefore affects the overall reliability as well as the requirement of memory storage for RTs. It can be tuned for different network topologies and node mobilities.

After the initiation phase, RLAR enters operation phase and data packets are delivered to the receivers. When packet loss occurs during the operation phase, RLAR will first try to perform local recovery. This is done by adjusting the protection links according to the up-to-date content of LQTs, while the backbone of the original tree structure will not be changed. When the situation becomes worse and more packets are lost, a rerouting procedure is performed and a new tree backbone will be constructed.

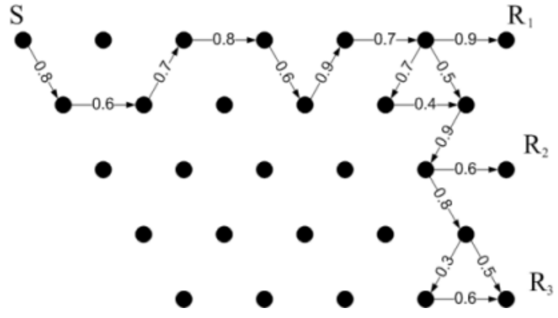
It should be noted that there are many different approaches to make the protections. In the above discussion, we adopt the most direct one - link protection. Some other approaches such as end-to-end protection and portion protection are still under study. The objective of these protection approaches is to draw a balance between the reliability enhancement and the increased requirement for memory storage of RTs.

C. Multicast example

In this section, a multicast scenario is shown as an example. Based on the description in the previous section, the route obtained by MAODV is shown in Figure 8(a), while the one obtained by RLAR is shown in Figure 8(b). It is obvious that the latter one has higher reliability.



(a) Resultant route of MAODV.



(b) Resultant route of RLAR.

Fig. 8: Multicast example.

VI. PERFORMANCE EVALUATION

This section describes the simulations we have done to evaluate the performance of RLAR. We compare the performance of this protocol with MAODV and two other reliable routing protocols, i.e., ITAMAR [7] and RoMR [8]. How the accuracy of link reliability estimation will affect the performance of RLAR is also investigated.

A. Simulation setup

First the simulation setup for our experiments is described. In order to estimate the reliability performance of the algorithms, it is assumed that no congestion and collision is involved and all the packet losses are caused by link failures due to node mobility.

Firstly, there is a requirement for our proposed protocol that the node density is high enough in the neighborhood so that redundant protective links could be constructed. Note that this is also a common requirement for ITAMAR and RoMR. One multicast session with single source and multiple destinations is simulated. We use the shadowing model [16] as the propagation model in physical layer. There are 50 uniformly placed nodes in the wireless network, with distance between two adjacent nodes about 250 m. The transmission range is 300 m. The source sends data at the rate of 100 packets/second. The random walk-based mobility model is used to determine the movement of each node. $\lambda^{-1} = 10$ and the range from which the random speed is generated is tuned for different scenarios. The reliability threshold is $L_r = 0.7$. For each scenario, at first all the nodes have a velocity of 10 m/s. In the next epoch, the velocity is uniformly generated from a specific range. The moving direction is uniformly generated from $[0, 2\pi)$. The link reliability estimation and route discovery are performed at the beginning of the simulation. MAODV uses hop-count as the routing metric, while ITAMAR builds up two independent trees for each multicast session and RoMR uses link lifetime as the routing metric. After the source node sends out 1100 packets, the simulation is finished. In order to better estimate the reliability performance, during the simulation, no rerouting is allowed. Every scenario is simulated for 10000 times and the average result is adopted.

The performance metrics used in evaluation are:

- **Packet delivery ratio:** The number of packets successfully delivered to *all* the multicast receivers over the total number of data packets needed to be delivered. This metric directly measures how reliable a route is because no retransmission is assumed.
- **Average communication time before rerouting:** The average communication time before a rerouting procedure should be triggered, i.e., the continuous reliable communication time before route failure. Route failure followed by rerouting procedures would interrupt the data transmission and cause packet loss during the transient period. Therefore reliability and throughput would benefit from longer communication time before rerouting.
- **The number of successful communications:** The number of communications during which all the packets are successfully delivered to the destinations, i.e, no route failure occurs during the communication. This is another metric to reflect the reliability of a selected route.

The performance comparison between RLAR and other schemes are evaluated in each scenario. With any estimation method, certainly there is estimation error. In order to evaluate how the estimation error in T_p will affect the performance, we also perform some simulations of RLAR for the case with precise estimation of T_p. This is done by pre-defining the movement trajectory of each mobile node. In this way, an exact T_p is obtained directly. In the following plots, those results with exact T_p are represented by the curve labelled as "RLAR (no error)". Note that although an exact T_p is used in Equation (4) to estimate the link availability, the estimated availability still has error incurred by the model used in [15]. Also note that it is not possible in actual implementations to have precise T_p, however, some additional technologies such as GPS might help minimizing the estimation errors.

B. Simulation results

In this section, the simulation results are shown and analyzed. The performances of different protocols are drawn in the same figure for comparison. The horizontal axis of each figure is nodes' mobility speed, which represents the range from which the random speed is generated. For example, the value 5 means the range is $[0, 5]$, 10 represents $[5, 10]$, and so on.

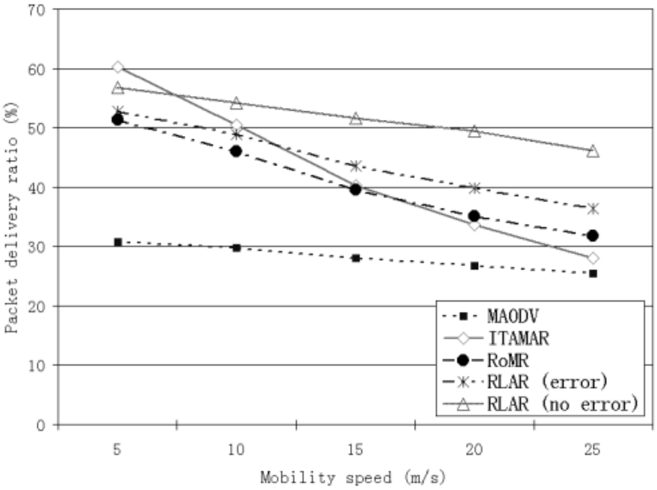


Fig. 9: Packet delivery ratio.

Figure 9 shows the packet delivery ratio. Note that during the simulation, no rerouting is allowed, therefore it is reasonable that the packet delivery ratios of other protocols may seem to be smaller than those reported in the literature. It is obvious that MAODV has the worst performance since it does not consider the reliable delivery of packets. Also, due to the less accurate estimation of link reliability, RoMR cannot choose the optimal route and thus has lower packet delivery ratio than RLAR. When nodes are static or quasi-static, ITAMAR has higher delivery ratio because of two independent alternative paths. However, as the mobility speed increases, the performance of ITAMAR drops dramatically. This is mainly due to the fact that ITAMAR does not estimate link reliability for routing metric. When nodes are moving fast, the probability of both independent trees having weak links is high and the performance of the algorithm is degraded. Although ITAMAR can reduce this probability by building up more independent trees, the requirement for memory storage puts too much overhead to the nodes. As can be seen from the curve labelled as "RLAR (no error)", if we can eliminate the estimation error, the performance of RLAR will be improved and the performance degradation due to speed increase will also be reduced.

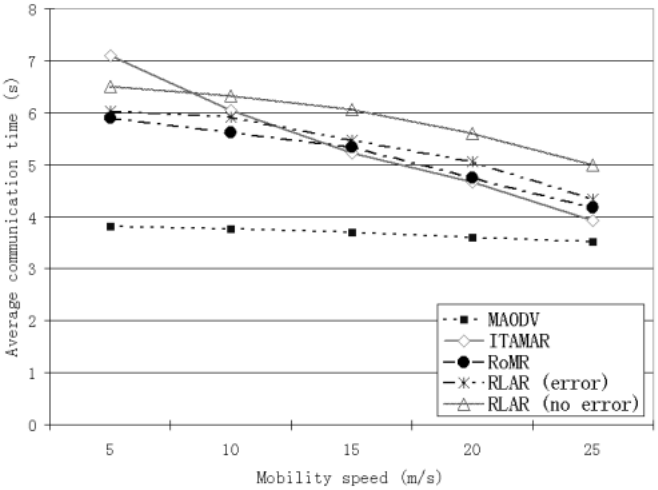


Fig. 10: Average communication time before rerouting.

Figure 10 shows the average communication time before rerouting. This metric is obtained by recording the simulation time before any route failure. Once a route fails, the packet cannot be delivered to the destination via the original route, thus a rerouting should be performed. Due to the resource-limited characteristic of wireless networks, frequent rerouting will cause large overhead and degrade the performance of the network. In addition, the packet in delivery during the rerouting procedure may be lost. Therefore, if the communication time before rerouting is longer, the original route is more reliable and the frequency of rerouting will be decreased. From Figure 10, we can see that RLAR always has longer average communication time than MAODV and RoMR. ITAMAR can achieve longer time with low mobility speed at the cost of more memory storage of RTs. When mobility speed increases, the average communication time achieved by ITAMAR has the similar decreasing trend as the packet delivery ratio.

Figure 11 shows the number of successful communications. A communication is regarded as successful if no route failure happens during the whole simulation period, i.e, no rerouting procedure is needed in the simulation time. This metric is related to the reliability of the original route and the nodes' mobility speed. As shown in Figure 11, even when speed is low, MAODV has very few successful communications, while the other three reliable routing protocols have much more. It

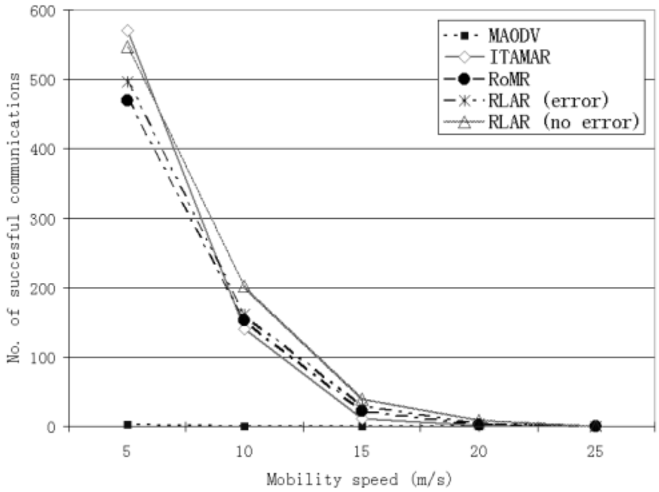


Fig. 11: Number of successful communications.

is also noted that as speed increases, the number of successful communications drops dramatically for all protocols. This implies that in highly dynamic scenarios, rerouting is necessary.

In summary, RLAR has its efficiency and effectiveness when compared to some previous multicast routing protocols. The reliability metric estimated by RLAR from physical layer information is more accurate than the historical data used by RoMR, and it requires much less memory storage than ITAMAR with comparable performance at low mobility speed, while much better performance at high mobility speed.

VII. CONCLUSION AND FUTURE WORK

RLAR is a new robust routing protocol for mobile ad hoc networks. It combines pro-active routing with on-demand routing. The two modules of RLAR are responsible for pro-active estimation of link reliability and on-demand optimal routing, respectively. One strength of RLAR is that it accurately estimates link reliability from physical layer information, therefore it works well with dynamic network topologies. Based on estimated link reliability, the optimal routing tree is constructed. Additional protective links are also added to the tree backbone to form the ultimate mesh structure so as to further enhance reliability. From simulations, RLAR is proved to successfully reduce the number of rerouting and improve the average communication time before a rerouting procedure is triggered. The packet delivery ratio is also remarkably increased. In addition, it has been shown that the performance of RLAR can be further improved by more precise estimation of link reliability.

RLAR proposed here provides an inspiration and direction for considering physical link reliability when dealing with the routing problem in MANETs, which is usually ignored in previous works. However, this paper is just the first step towards a comprehensive solution for robust routing in MANETs and many open issues need to be addressed in the future. The first open problem is the estimation precision of the first module. In this paper, we adopt the simple shadowing model [16] as the propagation model in physical layer and assume the channel is perfect, i.e., no other channel impairments and collisions are involved. Based on this simplification, we use received signal strength indication (RSSI) and compass direction to estimate the link availability. We notice that the shadowing model over-simplifies the actual physical layer characteristics. New approaches of estimating the link availability based on more accurate physical layer models are being studied.

Another open problem is that a robust routing metric needs to consider not only physical layer aspects but also MAC layer aspect. In this paper, in order to highlight the importance of considering the physical link reliability, we directly use the link reliability as the routing metric. However, as we have indicated earlier, a good routing metric should also reflect the MAC layer conditions, for example, data rate, interference and congestion. Our next step is to combine the physical layer and MAC layer aspects together, and thus developing a comprehensive solution to the robust routing problem.

REFERENCES

[1] M. S. Corson, J. P. Macker and G. H. Cirincione, "Internet-based mobile ad hoc networking," *IEEE Internet Computing*, vol.3, issue 4, pp. 63-70, 1999

[2] R. Draves, J. Padhye, and B. Zill, "Comparisons of Routing Metrics for Static Multi-Hop Wireless Networks," *Proceedings of SigComm 2004*, pp. 133-144, 2004

[3] S. Gupta, V. Shankar, and S. Lalwani, "Reliable multicast MAC protocol for wireless LANs," *Proceedings of ICC 2003*, pp. 93-97, 2003

[4] W. Si and C. Li, "RMAC: A Reliable Multicast MAC Protocol for Wireless Ad Hoc Networks," *Proceedings of ICPP 2004*, pp. 494-501, 2004

[5] R. Chandra, V. Ramasubramanian and K. Birman, "Anonymous Gossip: improving multicast reliability in mobile ad-hoc networks," *Proceedings of ICDCS 2001*, pp. 275-283, 2001

[6] M. Jiang and W. Liao, "Family ACK tree (FAT): a new reliable multicast protocol for mobile ad hoc networks," *Proceedings of ICC 2002*, vol.5, pp. 3393-3397, 2002

[7] S. Sajama and Z.J. Hass, "Independent-tree ad hoc multicast routing (ITAMAR)," *Proceedings of VTC 2001*, vol.2, pp. 600-604, 2001

[8] G. H. Lynn and T. F. Znati, "RoMR:A Robust Multicast Routing Protocol for Ad-Hoc Networks," *Proceedings of IEEE Conference on Local Computer Networks*, pp. 260-268, 2001.

[9] T. Gopalsamy, M. Singhal, D. Panda and P. Sadayappan, "A Reliable Multicast Algorithm for Mobile Ad hoc Networks," *Proceedings of ICDCS 2002*, pp. 563-571, 2002

[10] E. M. Royer and C. E. Perkins, "Multicast Operation of the Ad-hoc On-Demand Distance Vector Routing Protocol," *Proceedings of MobiCom*, pp. 207-218, 1999

[11] J. Yli-Hietanen, K. Koppinen, J. Astola, "Low-complexity angle of arrival estimation of wideband signals using small arrays," *Proceedings of the 8th IEEE Signal Processing Workshop on Statistical Signal and Array Signal Processing*, pp. 109-112, 1996

[12] J. Yli-Hietanen, K. Koppinen, J. Astola, "Time-Delay Selection for Robust Angle of Arrival Estimation," *SIP 1999*, pp. 81-83, 1998

[13] R. Dube, C. Raia, K-Y Wang and S. Tripathi, "Signal stability based adaptive routing (SSA) for ad hoc networks," *IEEE Personal Communications*, pp. 36-45, 1997

[14] A. B. McDonald and T. F. Znabi, "A path availability model for wireless ad hoc networks," *Proceedings of WCNC 1999*, pp. 35-40, 1999.

[15] S. Jiang, D. He and J. Rao, "A prediction-based link availability estimation for mobile ad hoc networks," *Proceedings of INFOCOM 2001*, vol. 3, pp. 22-26, 2001.

[16] T. S. Rappaport, "Wireless Communications, Principles and Practice," Prentice Hall, 1996.

[17] D. De Couto, D. Aguayo, J. Bicket and R. Morris, "High-throughput path metric for multi-hop Wireless Routing, *Proceedings of MobiCom*, 2003.