A Stable Power-Aware Multicast Routing Protocol for Mobile Ad Hoc Networks

Nen-Chung Wang and Yu-Li Su
Department of Computer Science and Information Engineering
Chaoyang University of Technology, Taichung 413, Taiwan, R.O.C.
Email: ncwang@mail.cyut.edu.tw

Abstract

A mobile ad hoc network (MANET) is a dynamically reconfigurable wireless network that does not have a fixed infrastructure. Due to the high mobility of nodes, the network topology of MANETs changes very fast, making it more difficult to find the routes that message packets use. Because mobile nodes have limited battery power, it is therefore very important to use energy in a MANET efficiently. In this paper, we propose a power-aware multicast routing protocol (PMRP) with mobility prediction for MANETs. In order to select a subset of paths that provide increased stability and reliability of routes, in routing discovery, each node receives the RREQ packet and uses the power-aware metric to get in advance the power consumption of transmitted data packets. If the node has enough remaining power to transmit data packets, it uses the global positioning system (GPS) to get the location information (i.e. position, velocity and direction) of the mobile nodes and utilizes this information to calculate the link expiration time (LET) between two connected mobile nodes. During route discovery, each destination node selects the routing path with the smallest LET and uses this smallest link expiration time as the route expiration time (RET). The destination nodes collect several feasible routes and then selects the path with the longest RET as the primary routing path. Then the source node uses these routes between the source node and each destination node to create a multicast tree. In the multicast tree, the source node will be the root node and the destination nodes will be the leaf nodes. Simulation results show that the proposed PMRP outperforms MAODV.

Keywords: Global positioning system, mobile ad hoc networks, power-aware routing, multicast routing protocol, tree-based routing, wireless networks.

1. Introduction

The mobile ad hoc network (MANET) [11] is a self-organizing and dynamically reconfigurable wireless network with no fixed infrastructure or central management. Two nodes communicate directly if they are within transmission range of each other. Due to the limited radio propagation range of wireless devices, routes are often “multi-hopped.” Every node in a MANET must be able to function as a router which forwards data packets to the other nodes. When applications must send the same data to more than one destination, multicasting is often used. Multicasting reduces the communication costs for applications that send the same data to multiple recipients. Instead of sending via multiple unicasts, multicasting minimizes link bandwidth consumption, sender and router processing, and delivery delay. The standard multicast routing protocols used in fixed networks or infra-structured mobile networks cannot be used in MANETs. Unicast routing is one special form of multicast routing and some proposed multicast routing protocols support both unicast and multicast routing. Many multicast routing protocols for MANETs have been proposed [1, 2, 3, 4, 5, 9, 10, 13, 15].

Multicast routing protocol in MANETs can be classified into tree-based routing protocols [1, 2, 5, 10, 13, 12], mesh-based routing protocols [3, 4, 9], and hybrid routing protocols. Tree-based routing protocols build a tree structure that connects all multicast members and provide one path between a pair of source node and destination node. Examples of tree-based routing protocols include the ad hoc multicast routing protocol (AMRoute) [2], the multicast ad hoc on demand distance vector routing protocol (MAODV) [15], the core-based trees (CBT) protocol [1], the reliable multicast algorithm (RMA) for mobile ad hoc networks [5], the dynamical construction of a core-based group-shared multicast tree in mobile ad hoc networks [10], and the efficient routing protocol for mobile ad hoc networks with neighbor awareness and multicasting (NAMP) [13]. Mesh-based protocols build a mesh structure that connects the mobile nodes between the source node and the destination node to each other. Because the mesh-based multicast protocols use redundant paths between two nodes, they provide alternative paths, and therefore a link failure need not trigger the re-computation of a mesh, but instead increases network overhead by flooding through a mesh. Examples of mesh-based protocols include the core-assisted mesh protocol (CAMP) [4], the on-demand multicast routing protocol (ODMRP) [9], the dynamic core based multicast routing protocol for ad hoc wireless networks (DCMP) [3]. In tree-based routing protocols, there is one path between the source node and destination node. However, tree-based routing protocols are not necessarily suitable for multicasting in MANETs, where network topology changes frequently.

In such an environment, mesh-based routing protocols seem to outperform tree-based routing protocols due to the availability of alternative paths which allow multicast datagram to be delivered to the receivers even if a link fails. On the other hand, mesh-based protocols have higher overhead than tree-based protocols. To resolve these difficulties, hybrid routing protocols can be used. Hybrid routing protocols use a combination of tree-based and mesh-based routing strategies. This kind of structure usually builds a mesh-based structure first, and then builds the tree-based structure afterwards.

In mobile ad hoc networks, each mobile node has limited battery power. In order to maximize the lifetime of ad hoc networks, traffic should be sent via routes that can be avoid nodes with low power while minimizing the total transmission power. Power consumption in a battery-powered node generally falls into one of two categories: communication related power and the non-communication related power. The non-communication related power is very dependent upon hardware implementation, routing, and link protocol design. Examples of power-aware routing protocols include the minimum total transmission power (MTPR) [16] routing, and the minimum drain rate (MDR) protocol [8], and the min-max battery cost routing (MMBCR) [17], and the conditional max-min transmission battery capacity routing (CMMBCR) [18].

In this paper, we propose a power-aware multicast routing protocol (PMRP) with mobility prediction. In this scheme, we consider the power parameter and determine the duration of time between two connected mobile nodes using the global positioning system (GPS). We select the routing paths with the longest duration of time for transmission to increase routing reliability. When a link on a routing path is broken, the routing path will be disconnected. We also propose a GPS-aided route reconstruction process that selects a backup path for route maintenance

The rest of this paper is organized as follows. Section 2 presents the preliminaries of this work. The proposed scheme is presented in Section 3. Section 4 describes the experimental results. Finally, Section 5 gives the conclusions.

2. Preliminaries

In this section, we first describe the technology used in the GPS and the mobility prediction mechanisms. Then we describe the power-aware metric to computes the power consumption. Finally, we present the multicast ad hoc on demand distance vector (MAODV) routing protocol.

2.1. Global Positioning System (GPS)

GPS [7] is the only system today able to show the exact position of an object or person anywhere on earth at anytime in any weather. It is a satellite-based, radio navigation system. The satellites are continuously monitored by ground stations located worldwide. The satellites transmit signals that can be detected by anyone with a GPS receiver. Using the receiver, one can determine the location of an object or person with great precision.

GPS consists of three segments: the space segment, the user segment, and the control segment. The space segment consists of 24 satellites, each in its own orbit 11,000 nautical miles above the Earth. The user segment consists of receivers which one can hold in one's hand or mount in one's car. The control segment consists of ground stations (five ground stations, located around the world) that make sure the satellites are working properly.

2.2. Mobility Prediction Mechanisms

In this section, we introduce the mobility prediction method. This method uses the location information provided by GPS. We assume a free space propagation model [14] in which the signal strength depends solely on the distance to the transmitter. We also assume that all nodes have their clocks synchronized using the GPS clock. If we know the motion parameters of two nodes, we can calculate the duration of time these two nodes remain connected. These parameters include speed, direction, and radio range and can be obtained from GPS.

We assume that two nodes A and B are within the same transition range r of each other. We let (x_1, y_1) be the coordinate for mobile node A and (x_2, y_2) be the coordinate for mobile node B. We let v_1 and v_2 be the mobility speeds and q_1 and q_2 $(0 \le q_1, q_2 < 2p)$ be the moving directions. We can obtain the duration of time D_t by using the following equation [14]:

$$D_t = \frac{-(ab+cd)+\sqrt{(a^2+c^2)r^2-(ad-bc)^2}}{a^2+c^2} \qquad (1)$$

Note that $a = v_1 \cos q_1 - v_2 \cos q_2$, $b = x_1 - x_2$, $c = v_1 \sin q_1 - v_2 \sin q_2$, and $d = y_1 - y_2$. Note also that the equation cannot be applied when $v_1 = v_2$ and $q_1 = q_2$, and when D_t is ∞. In order to get and utilize the information from GPS, the packets must include extra fields. When a source node sends a request packet, the packet appends its location, direction, and speed. The next hop neighbor of the source node receives the request packet to predict the duration of time between itself and the source node. If node B is the next hop of the packet for node A, node A will insert its location information in the packet so node B will be able to compute the duration of time between node A and node B.

2.3. Power Aware Metric

In a MANET, the power consumption of each mobile node becomes an important factor that affects network performance. A lack of sufficient hosts can result in a partition of the network, causing interruptions in communications. In a simple radio model [6], E_{elec} = 50 nJoule/bit is dissipated to run the transmitter or receiver circuitry. E_{elec} is the power consumption of the circuit itself. Assuming d^2 energy loss, where d is the distance between nodes, a transmission amplifier at the sender node further consumes further $E_{amp}d^2$, where E_{amp} = 100 pJoule/bit/m^2. E_{amp} is the power consumed by the amplifier to transmitting packets. These parameters are slightly better than the parameters used in current state-of-the-art in radio design. We also assume an r^2 energy loss due to channel transmission. Thus, to transmit a k-bit message a distance d using the radio model, the radio expends:

$$E_{Tx}(k, d) = E_{elec}\times k + E_{amp}\times k\times d^2 \tag{2}$$

and to receive this message, the radio expends:

$$E_{Rx}(k) = E_{elec}\times k \tag{3}$$

Receiving a message is not a low cost operation using these parameter values. Protocols should thus try to minimize not only the transmission distances but also the numbers of transmission and reception operations for each message. We can generalize about the total transmission consumption as follows:

$$E_{total}(k) = (E_{elec}\times k + E_{amp}\times k\times d^2) + (E_{elec}\times k) \tag{4}$$

Fig. 1 shows the power-aware metric of a first order radio model.

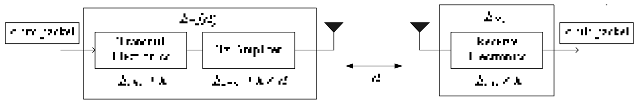

Fig. 1. Power first order radio model.

2.4. Multicast Ad Hoc on Demand Distance Vector (MAODV) Routing Protocol

In MAODV [15], the researchers directly followed the unicast AODV protocol [12] and discovered multicast routes on demand using a broadcast route discovery mechanism that employs the same route request (RREQ) and route reply (RREP) packets that exist in the unicast AODV protocol.

When a mobile node wishes to join a multicast tree or has data to send to a multicast group but has no route it can use, it will broadcast RREQ packet. Only a member of the desired multicast group may respond to an RREQ. If an intermediate node receives a join RREQ for a multicast group of which it is not a multicast member, or if it does not have a route to the group, it will rebroadcast the RREQ to its neighbors. If a node receives a RREQ for a multicast tree, it may reply if it is a member of the multicast tree and its recorded sequence number for the multicast group is as great as that contained in the RREQ. The responding node of the multicast tree unicasts a RREP back to the source node after it receives an RREQ packet. As nodes along the path to the source receive the RREP, they add both a route table and a multicast route table entry for the node from which they received the RREP, as shown in Fig. 2.

After the source node broadcasts a RREQ packet to a multicast group, it often receives more than one RREP packet. For a period of time, the source node keeps the received route with the greatest sequence number and shortest hop count to the nearest multicast member of the multicast tree, and disregards the other routes. At the end of this period, it enables the selected next hop node in its multicast route table, and sends a unicast activation message (MACT) to the selected next hop node. On receiving the message, the next hop node enables the entry for the source node in its multicast routing table. The next hop node does not propagate the message any further if it is a member of the multicast tree. However, if the intermediate node is not a member of the multicast tree, it will have received several RREPs from its neighbors. It keeps the best next hop node for its route to the multicast group, unicasts MACT to that next hop node, and enables the corresponding entry in its multicast routing table. This process continues until the node that originated the chosen RREP is reached. MACT ensures that the multicast tree does not have multiple paths to any tree node. The intermediate node forwards data packets only along the activated routes.

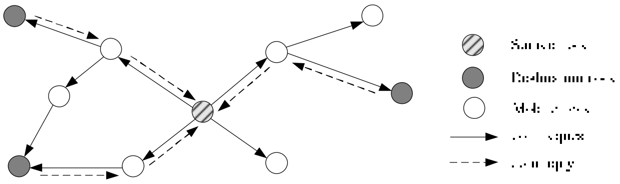

Fig. 2. Route discovery of the MAODV protocol.

3. The Power-Aware Multicast Routing Protocol (PMRP) with Mobility Prediction

In this section, we propose a power-aware multicast routing protocol (PMRP) with mobility prediction. In PMRP, we create a more multicast tree for transmitting data packets. The proposed protocol includes the route discovery process and the route maintenance process.

3.1. Route Discovery Process

In PMRP, the member nodes and multicast routes are established and updated by the source “on demand.” Similar to on-demand unicast routing protocols, PMRP contains the request phase and reply phase. Since PMRP is an on-demand protocol, it sets up routes when a source has data to send.

The RREQ packet is broadcast by the source node when the source node needs to send data to the destination nodes and has no route information in its routing table. When each destination node can finds a suitable path, the destination node sends a RREP packet to the source node. The beacon packet is broadcast by each node frequently to determine the neighboring nodes and the distance between that node and the neighboring nodes.

In our proposed PMRP, we define three parameters: the total power consumption of transmitted data packets ($P_{prediction}$), the link expiration time (LET) and the route expiration time (RET). The $P_{prediction}$ represents the total power consumption of transmitted data packets. It is calculated by each node, when a node receives a RREQ packet using Equation (4). The LET represents the link duration time between two nodes. When a node forwards a RREQ packet, an intermediate node receives this packet and calculates the LET by using Equation (1). The RET is equal to the minimum of the set of LETs for the routing path. Thus, the RET is the time that the route is expected to be stable. The larger the RET is the more reliable routing path.

In PMRP, each node frequently broadcasted the beacon packet to find out the neighboring nodes, when a neighboring node receives a beacon packet, it calculates the distance between two nodes, and sent back a reply packet. When a source node wants to multicast information to some multicast destination nodes, in the beginning, the source node will be the root node of the multicast tree and will initiate a path discovery process by broadcasting a RREQ packet to its neighbors. The RREQ packet has to include the location information, that is, the node's location, velocity, and direction. Once the intermediate nodes receive the RREQ, it first checks the broadcast ID to determine whether the entry is its own ID. If not, it will calculate the $P_{prediction}$ to its neighboring nodes. If the power remaining on the node is lower than the $P_{prediction}$ ($P_{remain} < P_{prediction}$), it means that the intermediate node does not have enough power to transmit data packets. It will not rebroadcast the RREQ packet and will discard the packet. If the power remain on the node is higher than the $P_{prediction}$ ($P_{remain} > P_{prediction}$), the node will calculate the LET between the next-hop node using the location information and then write the data into the LET field of the RREQ packet and its routing table. It will then rebroadcast the packet to its neighbors. By the time a request packet arrives at the destination node, it has recorded all the nodes along the routing path it has traversed and the duration of time of each link along the route. Then the destination node will determine the RET by looking in the packet to find the minimum number of LETs along the path of the route request packet. Each destination node waits a period of time to collect RREQs. We assume that each destination node has many routing paths that can be used. Each destination node then selects the primary routing path with the maximum RET and sends a RREP packet back to the multicast source node along the decided path. The intermediate nodes between each source and destination pair may need to act as routers for the tree. We employ the routing paths with the longest route duration of time to establish a reliable multicast tree.

Let us consider the example shown in Fig. 3. In Fig. 3(a), each node in the network broadcasts a beacon packet to obtain the information of the neighboring nodes' information (i.e. the neighboring nodes' ID and distance). Source node *A* wants to send data to destination nodes *J*, *K*, *L*, *M*, and *N*. It broadcasts a RREQ packet to its neighboring nodes. Nodes *B*, *C*, …, *H*, and *I* will append their own information, such as their own ID and the duration of time, to the request packet and then forward the request packet. In this example, when source node *A* broadcasts the RREQ packet, intermediate node *B* receives the request packet and calculates $P_{prediction}$. $P_{prediction}$ of the node *B* is 30 and P_{remain} of the node *B* is 50. Because P_{remain} is higher than the $P_{prediction}$, node *B* has enough power to transmit data packets. Then it will utilize the location information to get the LET between node *A* and *B*, write the information into the RREQ packet and then rebroadcast this packet. $P_{prediction}$ of nodes *D* and *H* are higher than P_{remain}. It means that the nodes do not have enough power to transmit data packets, and they will discard these RREQ packets. Finally, destination node *K* receives three request packets. The first packet contains path (*A*, *C*, *F*, *K*) with the duration of time of LETs = (6, 4, 3). The second packet contains path (*A*, *C*, *E*, *K*) with LETs = (6, 7, 5). The third packet contains the path (*A*, *B*, *E*, *K*) with LETs = (6, 4, 5). Node *K* can obtain the RET evoked from the minimum LET. In this case, the RET of path (*A*, *C*, *F*, *K*) is 3, the RET of path (*A*, *C*, *E*, *K*) is 5, and the RET of path (*A*, *B*, *E*, *K*) is 4. Thus, the RET of path (*A*, *C*, *E*, *K*) is larger than that of path (*A*, *C*, *F*, *K*) and path (*A*, *B*, *E*, *K*). Thus, path (*A*, *C*, *E*, *K*) is more stable than the other two paths. Similarly, nodes *J*, *L*, *M*, and *N* will select paths (*A*, *C*, *E*, *J*), (*A*, *G*, *L*), (*A*, *G*, *M*), and (*A*, *G*, *I*, *N*) with the RETs equal to 5, 5, 5, and 4, respectively. If the P_{remain} of the destination node is lower than the $P_{prediction}$, the routing discovery process will fail.

In Fig. 3(b), each destination node selects the primary routing path with the longest duration of time for multicast routing and then sends its RREP packet back to the source node. Source node *A* receives all the destination nodes' reply packets and delivers the data packet along these routing paths. Finally, as shown in Fig. 3(c), we can use these routing paths to build a multicast tree, the multicast tree not only has enough power to transmit data packet but also has the longest route lifetime.

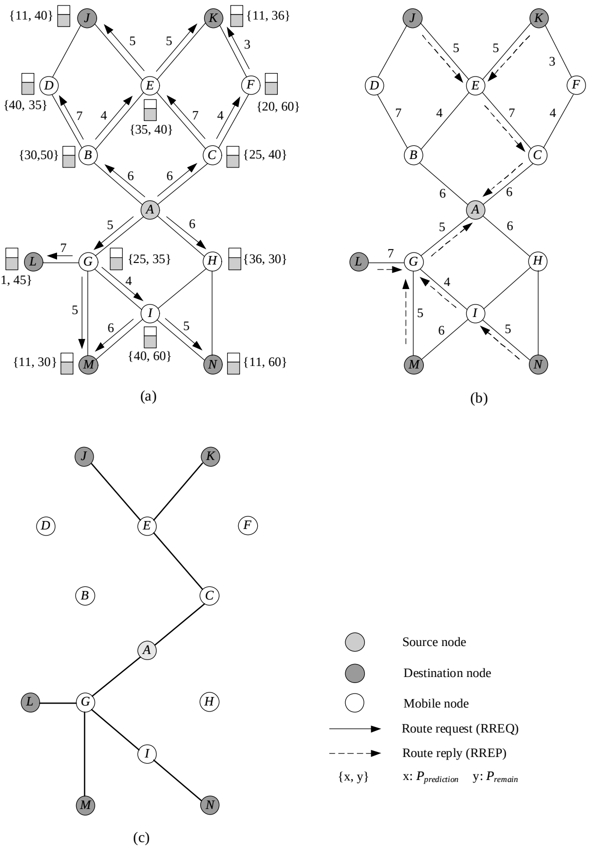

Fig. 3. Routing process. (a) The route discovery process. (b) The route relay process. (c) The reliable multicast tree.

3.2. Route Maintenance Process

In the following, we present the route maintenance process that includes multicast join operation, node pruning operation, and broken link maintenance.

3.2.1. Multicast Join Operation

When a node wants to join a multicast tree, it broadcasts a join request packet across the networks. Only a node that is a member of a multicast tree (i.e., a router for the group) may respond. If a node receives a join request packet for a multicast group of which it is not a member, or if it receives a join request packet and does not have a route to that group, it creates a reverse route entry to the prospective node and then broadcasts the join request packet to its neighbors. Each member node of the multicast tree waits for a period time to collect the join request packet, chooses the longest duration of time route from the prospective node, and sends back the join reply packet. The join reply packet must add the LET information between the prospective node and the source node. The prospective node will select the longest duration of time route to join the multicast tree. Fig. 4 shows the node join operation.

In Fig. 4(a), prospective node *L* broadcasts a join request packet to its neighbors. When nodes *D*, *F* and *G* receive the packet, they will calculate the LET with the prospective node, put the information into the packet, and then rebroadcast it, repeating this process until the multicast group member receives the packet. The member nodes collect these join request packets. Nodes *A*, *B*, *C*, *E*, and *I* will select the route with the longest link expiration time and return the join reply packet to the prospective node, as shown in Fig. 4(b). The node waits for a period of time to collect these join reply packets and then chooses the most reliable route to the tree, as shown in Fig. 4(c).

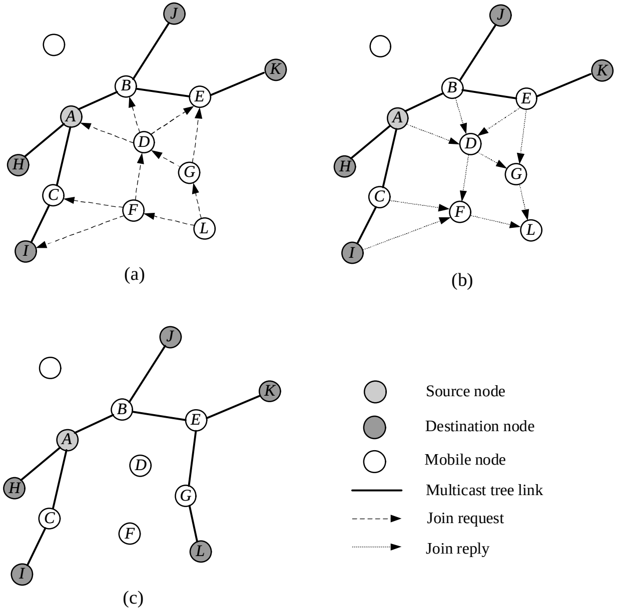

Fig. 4. Multicast join operation. (a) Join request packet propagation. (b) Join reply sent back to source. (c) Multicast tree branch addition.

3.2.2. Node Pruning Operation

When a node is removed from a multicast tree, the pruned node sends to its upstream node a quit_request packet. When the upstream node receives the quit request packet, it will remove the corresponding entry from its multicast routing table. If the upstream node becomes a leaf node (because it is a router for the tree and not the tree receiver as a result of this removal, and because it is not interested in multicast traffic from this tree), it can further prune itself from the tree and in turn send a quit request packet to its upstream node.

For example, as shown in Fig. 5(a), node *I* decides to leave the multicast tree. It sends a quit request packet to node *C*. When node *C* receives the packet and deletes node *I* from its list of next hops, it discovers that it is a leaf node. But because it is just a router for the multicast tree and not a multicast member, it will in turn send a quit request packet to node *A*. Fig. 5(b) illustrates the new multicast tree.

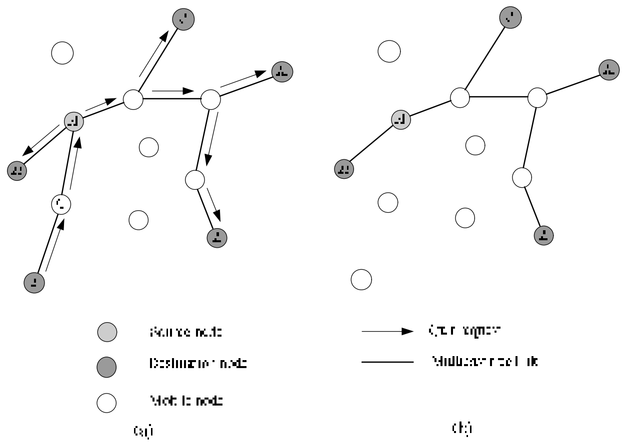

Fig. 5. Node pruning operation. (a) Quit request packet propagation. (b) Multicast tree after pruning.

3.2.3. Broken Link Maintenance

In a MANET, because of the mobility of the mobile nodes, a link between two nodes will break easily. In PMRP, we use the location information to get the LETs and RETs. We can predict the time that the link will break. The maintenance process will be started before the link breaks. The upstream node of the link will broadcast the route request_repair (RREQ_R) packet to the downstream node. The RREQ_R packet must add information about the remnants of the data packets that need to be sent. An intermediate node receives the packet, and uses the remnant data information to calculate the $P_{prediction}$, if P_{remain} is lower than $P_{prediction}$, it drops this RREQ_R packet. If P_{remain} is higher than $P_{prediction}$, the node calculates the LET between the next-hop node using the location information, writes the LET into the link expiration time field of the RREQ_R packet, and rebroadcasts the RREQ_R packet to its neighbors until the request packet arrives at the downstream node. When the downstream node receives the RREQ_R packet, it determines the RET by looking in the packet to find the minimum number of LETs along the path of the RREQ_R packet. The downstream node waits a period of time to collect more RREQ_Rs and selects the alternative routing path with the maximum RET. Then it sends the route reply_repair (RREP_R) packet back to the upstream node. The upstream node will use the alternative path to transmit data before the original link breaks. Moreover, the upstream node defines a threshold $T_{timeout}$. The upstream node will drop the RREP_R packets after the threshold $T_{timeout}$ counts down to 0. Then it sends an error packet to the source node and restarts the route discovery process.

An example of the link broken maintenance is shown in Fig. 6(a). Source node *A* sends a data packet to the destination nodes *H*, *I*, *J*, *K*, and *L*. We assume that the link between node *B* and node *E* will break. When node *B* detects that the link between node *B* and node *E* will break, node *B* broadcasts a RREQ_R packet to discover a backup path. In this example, node *B* broadcasts a RREQ_R packet to discover the backup routing path. When node *D* and node *J* receive the RREQ_R packets, they update their routing tables and utilize the power first order radio model to get $P_{prediction}$. Because P_{remain} of node *J* is lower than $P_{prediction}$, node *J* will discard the RREQ_R packet. Because P_{remain} of node *D* is higher than $P_{prediction}$, node *D* will use the location information to calculate the LET and write the LET into the link expiration time filed of the RREQ_R packet, and then rebroadcast the RREQ_R packet. We assume the RREQ_R packet arriving at node *E* is via node *D*. Node *E* selects the RET by looking in the RREQ_R packet to find the minimum number of LETs. Finally, node *E* sends back a RREP_R to node *B*. When the link between node *B* and node *E* is broken, node *B* delivers data along the backup path (*B*, *D*, *E*) to the destination nodes *K* and *L*, as shown in Fig. 6(b). If we cannot find a replacement node, we use the last node of the broken path to send the error packet back to the source node. We utilize the error packet to restart the route discovery.

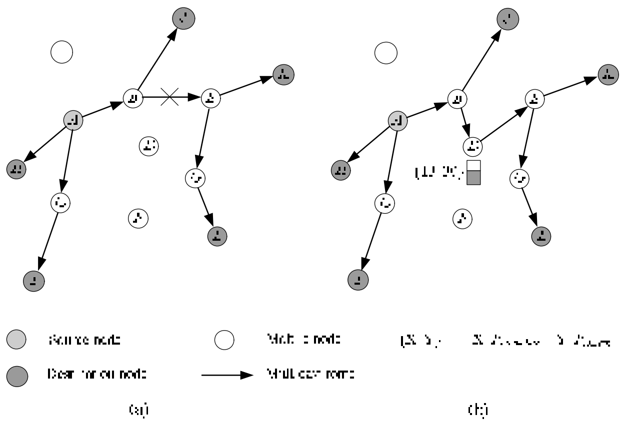

Fig. 6. Repair of a broken tree link. (a) Link break (b) Repaired multicast tree.

4. Experimental Results

In this section, we will introduce the software and the parameters used in our simulation. In addition, we will compare the performance of the proposed PMRP with that of the MAODV [15].

4.1. The Simulation Environment

We designed and implemented a simulator to act as an experimental platform for testing multicast operations in a MANET. We developed it using in Java language. The simulation modeled a network in a 1000 m × 1000 m area with 50 mobile nodes. A random waypoint model was used in the simulation. The mobile speed of each node was from 0 m/sec to 40 m/sec. The transmission range was 150 m. The data packet size was 1024 bytes. The data transmission rate was set to 2 Mb/sec. Each simulation was executed for 600 seconds of simulation time. The source and destination nodes were randomly chosen. Each node was randomly assigned an initial energy.

The performance metrics used were as follows:

1. Packet delivery ratio: The ratio of the average number of data packets received by the destination node to the number of data packets transmitted by the multicast source.
2. End-to-end delay: The time when a data packet is sent by the source to the time the data packet is received at the destination node.
3. Control overhead: The total number of control packets received by the destination node.

4.2. Performance Analysis

In the following, some simulations are conducted for packet delivery ratio, end-to-end delay, and control overhead, respectively.

4.2.1. Packet Delivery Ratio

Fig. 7 shows the performance of the average packet delivery ratio under various mobile speeds, which ranged from 0 m/sec to 40 m/sec. The packet delivery ratio decreased with increasing mobility due to more link breaks. This resulted in more multicast tree partitions for PMRP and MAODV. Notice that the number of packet deliveries was high when the nodes had low mobility. Note also that the multicast tree structure was mostly static and, therefore, the packet delivery ratio was high. At high speeds, the tree links broke down quite often, leading to constant branch reconstructions and larger packet losses. In PMRP, we used the power information and location information to select the more stable routing paths for multicasting. Thus, the delivery ratio of PMRP was higher than that of MAODV.

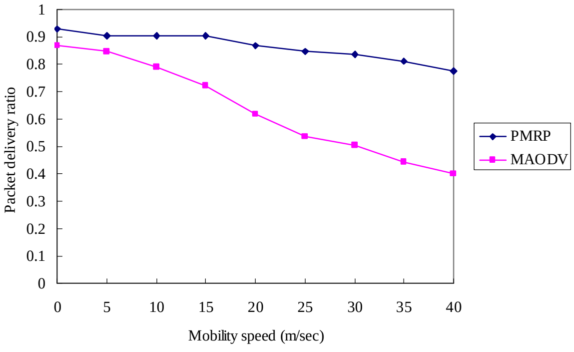

Fig. 7. Packet delivery ratio vs. mobility speed.

4.2.2. End-to-End Delay

Fig. 8 shows the performance of the end-to-end delay under various mobility speeds. PMRP had higher end-to-end delay than the MAODV because PMRP had a longer routing path from the source node to the destination node. However, PMRP can found the more stable path. This is because our reliable path scheme will increase the end-to-end delay.

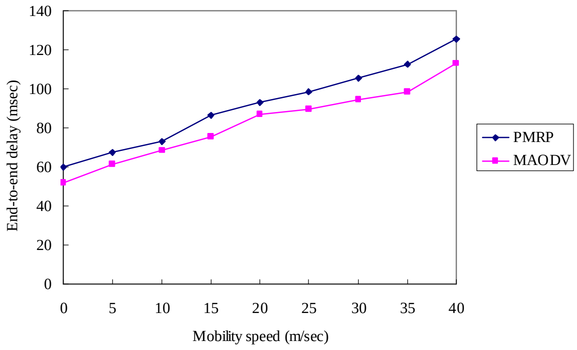

Fig. 8. End-to-end delay vs. mobility speed.

4.2.3. Control overhead

Fig. 9 shows the performance of the control overhead under various mobility speeds. As was expected, the control overhead increased as the mobile nodes became more mobile. The reason is that there were more chances for routes to break when the speed of the mobile nodes was faster. Thus, the number of rebroadcasts increased. Because our protocol eliminates inefficient nodes in order to decrease the number of control packets to be broadcast and selects the more stable route for data transmission, the number of route reconstructions was less. Therefore, PMRP had have a lower control overhead than MAODV. Our protocol can maintain routing paths in advance by using the proposed route maintenance process.

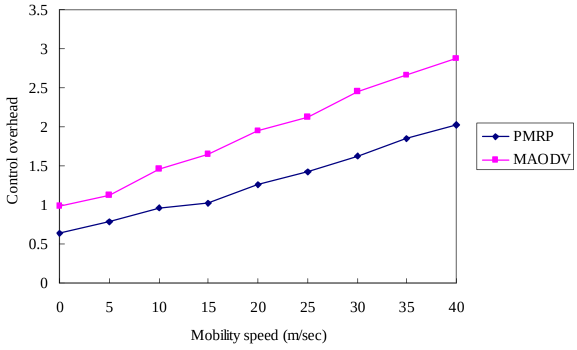

Fig. 9. Control overhead vs. mobility speed.

5. Conclusions

In this paper, we proposed a power-aware multicast routing protocol (PMRP) with mobility prediction for MANETs. In this scheme, we use the power-aware metric to compute power consumption in advance. We also use the result to discover the efficient nodes and decrease the numbers of control packets that need to be broadcast. Then we use the global positioning system (GPS) to get the location information of the mobile nodes and utilize this information to calculate the LET between two connected mobile nodes. The destination node selects the routing path with the smallest LET and uses this smallest link expiration time as the RET. Each destination node collects several routing paths and selects the primary routing path with the longest RET for data transmission. We also proposed a route maintenance process to increase the data delivery ratio and decrease the control overhead. Finally, the experimental results showed that PMRP outperformed MAODV, especially in a high mobility environment.

Acknowledgements

This work was supported by the National Science Council of Republic of China under grants NSC-93-2218-E-324-001 and NSC-94-2213-E-324-025.

References

[1] A. Ballaradie, J. Crowcroft, and P. Francis, "Core Based Tree (CBT) – An Architecture for Scalable Inter-Domain Routing Protocol," *Proceedings of the 1993 ACM SIGCOM*, pp. 85-89, October 1993.

[2] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade, "AMRoute: Ad Hoc Multicast Routing Protocol," *Internet draft*, August 1998.

[3] S. K. Das, B. S. Manoj, and C. S. R Murthy, "A Dynamic Core Based Multicast Routing Protocol for Ad hoc Wireless Network," *Proceeding of the third ACM international symposium on mobile ad hoc networking & computing*, pp.24-35, June 2002.

[4] J. J. Garcia-Luna-Aceves and E. L. Madruga, "The Core-Assisted Mesh Protocol," *IEEE JSAC*, pp. 1380-1394, August 1999.

[5] T. Gopalsamy, M. Singhal, D. Panda, and P. Sadayappan, "A Reliable Multicast Algorithm for Mobile Ad Hoc Networks," *Proceedings of the 2002 IEEE International Conference on Distributed Computing Systems*, pp. 563-570, July 2-5, 2002.

[6] W. R. Heinzelman, A. Chandrakasan and H. Baladrishnan, "Energy-efficient routing protocols for microsensor networks," *Proceedings of the 33rd Hawaii International Conference on System Sciences*, Vol. 8, pp. 1-10, January 2000.

[7] E. D. Kaplan. "Understanding GPS: Principles and Applications," Artech Hourse, Boston, MA, 1996.

[8] D. Kin, J. J. Garcia-Luna-Aceves, and K. Obraczka, "Power-Aware Routing Based on the Energy Drain Rate for Mobile Ad Hoc Networks." *Proceedings of the IEEE International Conference on Computer Communications and Networks*, pp. 565-569, October 2002.

[9] S. J. Lee, M. Gerla, and C. C. Chiang, "On Demand Multicast Routing Protocol," *Proceedings of the 1999 IEEE WCNC*, pp. 1298-1302, September 1999.

[10] B. H. Liu, M. J. Tsai and W. C. Ko, "Dynamical Construction of a Core-Based Group-Shared Multicast Tree in Mobile Ad Hoc Networks," *19th International Conference on Advanced Information Networking and Applications*, Vol. 1, pp. 90-95, March 2005.

[11] J. P. Macker and M. S. Corson, "Mobile Ad Hoc Networking and the IETF," *ACM SIGMOBILE Mobile Computing and Communications Reviews*, Vol. 2, No. 2, pp. 9-14, January 1998.

[12] C. E. Perkins and E. Royer, "Ad-Hoc On-Demand Distance Vector Routing," *Proceedings of the Second IEEE Workshop on Mobile Computing System and Application*, New Orleans, LA, USA, pp. 90-100, February 1999.

[13] A. –S.K. Pathan, M. M. Alam, M. M. Monowar and M. F. Rabbi, "An efficient routing protocol for mobile ad hoc networks with neighbor awareness and multicasting," *E-Tech 2004*, pp. 97-100, July 2004.

[14] T. S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice-Hall, Upper Saddle River, NJ, 1995.

[15] E. M. Royer and C. E. Perkins, “Multicast Operation of the Ad Hoc On-Demand Distance Vector Routing Protocol,” *ACM MOBICOM*, pp. 207-18, August 1999.

[16] K. Scott and N. Banmbos, "Routing and Channel Assignment for Low Power Transmission in PCS," *Proceedings of the International Fifth ICUPC Conference on Universal Personal communications*, vol. 2, pp. 368-369, October. 1996.

[17] S. Singh, M. Woo, and C. S. Raghavendra, "Power-Aware with Routing in Mobile Ad Hoc Networks," *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 181-190, October 1998.

[18] C. K. Toh, "Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks," *IEEE Communications Magazine*, Vol. 39, No. 6, pp. 138-147, June 2001.