Mobility Prediction and Routing in Ad Hoc Wireless Networks

William Su, Sung-Ju Lee, and Mario Gerla
Wireless Adaptive Mobility Laboratory
Computer Science Department
University of California
Los Angeles, CA 90095-1596
(wsu,sjlee,gerla)@cs.ucla.edu

Abstract

Wireless networks allow a more flexible model of communication than traditional networks since the user is not limited to a fixed physical location. Unlike cellular wireless networks, an ad hoc wireless network does not have any fixed communication infrastructure. For an active connection, the end host as well as the intermediate nodes can be mobile. Therefore routes are subject to frequent disconnections. In such an environment, it is important to minimize disruptions caused by the changing topology for critical application such as voice and video. This presents a difficult challenge for routing protocols, since rapid reconstruction of routes is crucial in the presence of topology changes. By exploiting non-random behaviors for the mobility patterns that mobile users exhibit, we can predict the future state of network topology and perform route reconstruction proactively in a timely manner. Moreover, by using the predicted information on the network topology, we can eliminate transmissions of control packets otherwise needed to reconstruct the route and thus reduce overhead. In this paper, we propose various schemes to improve routing protocol performances by using mobility prediction. We then evaluate the effectiveness of using mobility prediction via simulation.

I. Introduction

An ad hoc network [10], [13] is a dynamically reconfigurable wireless network with no fixed infrastructure. Each host acts as a router and moves in an arbitrary manner. Ad hoc networks are deployed in applications such as disaster recovery and distributed collaborative computing, where routes are mostly multihop and network hosts communicate via packet radios. In such a network, it is critical to route the packets to destinations effectively without generating excessive overhead. This presents a challenging issue for protocol design since the protocol must adapt to frequently changing network topologies in a way that is transparent to the end user.

Various routing schemes have been proposed for ad hoc networks [19], [23], [24], [25], [27], [29], [22], [6], [11]. Two schemes specifically have been introduced to make use of location information of mobiles to improve routing protocol performance. Location-Aided Routing (LAR) [15] is a protocol that takes advantage of location information. LAR uses location information obtained from the Global Positioning System (GPS) [14], [8] to limit the propagation region of ROUTE REQUEST packets. Distance Routing Effect Algorithm for Mobility (DREAM) [3] is another location based routing protocol. DREAM is different from LAR in that it performs routing (location) table updates periodically. DREAM partially floods data to nodes in the general direction of the destination. The location table contains the coordinates for every destination in the network. Location information exchange is done on a periodic basis.

In typical mobile networks, nodes exhibit some degree of regularity in the mobility pattern. For example, a car traveling on a road is likely to follow the path of the road and a tank traveling across a battlefield is likely to maintain its heading and speed for some period of time before it changes them. By exploiting a mobile user's non-random traveling pattern, we can predict the future state of a network topology and thus provide a transparent network access during the period of topology changes. In this paper we present various enhancements to unicast and multicast routing protocols using mobility prediction. The enhancements proposed in this paper utilizes GPS location information. However, GPS information is used differently from LAR and DREAM. In our enhancements, GPS position information is used to estimate the expiration time of the link between two adjacent mobiles. Based on this prediction, routes are reconstructed before they expire. Our goal is to provide a seamless connection service by reacting before the connectivity breaks.

The remainder of the paper is organized as follows. Section II presents the method used to predict the link expiration time (LET) and various ways to enhance unicast and multicast routing protocols using mobility prediction. Section III describes the simulation environments used to evaluate the performance of the enhancements. The results from the simulation are presented in Section IV. Concluding remarks are made in Section V.

II. Mobility Prediction Mechanisms

A. Methods of Predicting Link Expiration Time

In this section, we introduce our mobility prediction method utilizing the location and mobility information provided by GPS.[1] In our initial approach, we assume a free space propagation model [26], where the received signal strength solely depends on its distance to the transmitter. We also assume that all nodes in the network have their clock synchronized (e.g., by using the NTP (Network Time Protocol) [20] or the GPS clock itself). Therefore, if the motion parameters of two neighbors (e.g., speed, direction, radio propagation range, etc.) are known, we can determine the duration of time these two nodes will remain connected. Assume two nodes i and j are within the transmission range r of each other. Let (x_i, y_i) be the coordinate of mobile host i and (x_j, y_j) be that of mobile host j. Also let v_i and v_j be the speeds, and θ_i and θ_j ($0 \leq \theta_i, \theta_j < 2\pi$) be the moving directions of nodes i and j, respectively. Then, the amount of time two mobile hosts will stay connected, D_t, is predicted by:

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

where

$a = v_i \cos\theta_i - v_j \cos\theta_j$,
$b = x_i - x_j$,
$c = v_i \sin\theta_i - v_j \sin\theta_j$, and
$d = y_i - y_j$.

[1] Mobility speed and heading information can be obtained from GPS or the node's own instruments and sensors (e.g., campus, odometer, speed sensors, etc.).

Note that when $v_i = v_j$ and $\theta_i = \theta_j$, D_t becomes ∞. The predicted time is the link expiration time (LET) between the two nodes.

Since GPS may not work properly in certain situations (e.g., indoor, fading, etc.), we may not always be able to accurately predict the link expiration time for a particular link. However, there is an alternative method to predict the LET. This method is based on a more realistic propagation model and has been proposed in [1] and [21]. Basically, transmission power samples are measured periodically from packets received from a mobile's neighbor. From this information the mobile can compute the rate of change for a particular neighbor's transmission power level. Therefore, the time that the transmission power level drops below the acceptable value (i.e., hysteresis region) can be computed. We plan to investigate the efficiency of this option further in our future work. However, the main effect of using power attenuation instead of GPS for distance estimation is a reduction in accuracy. The impact of accuracy will be addressed in this paper, and should apply also to power attenuation based on measurements.

B. Applying Mobility Prediction to Unicast Protocols

B.1 The On Demand Approach

In this section we describe a protocol that uses mobility prediction, the Flow Oriented Routing Protocol (FORP) [28], for routing real time flow traffic (e.g., voice and video) in ad hoc networks using the mobility prediction. Our proposed scheme is inspired by on-demand schemes; we only maintain routes for "active" source/destination pairs. When the sender has a flow to send, it constructs a route to the destination on demand and injects the flow. The destination predicts the change in topology ahead of time and determines when the flow needs to be rerouted or "handoffed" based on the mobility information contained in the data packets. We make the assumption that a given mobile will be able to predict the link disconnection time of any of its one-hop neighbors based on the method described in Section II.A. The basic concept of the enhancement is that if we can predict the LET along each hop on the route, we will be able to predict the Route Expiration Time (RET).

When a source needs to send a flow to the destination, it will first check its routing table to see if it has an unexpired route to the destination. If it does, it will send data packets using the route immediately. Otherwise it broadcasts a FLOW-REQ message to find a route to the destination. This FLOW-REQ message is processed similarly to the ROUTE REQUEST message in DSR [11], or the QRY message in LMR [6]. The message contains a sequence number, source ID, destination ID, and the list of node IDs appended by the intermediate nodes while the message travels through the network. A neighbor, upon receiving this message, will forward it to all of its neighbors if (a) the message contains a higher sequence number for any of the previously received source and destination ID pair, or (b) the message contains the same sequence number but the message arrives from a better route (e.g., the route has a larger RET with the same or fewer number of hops). When a node is forwarding the FLOW-REQ message, it also appends its own ID and the LET for the

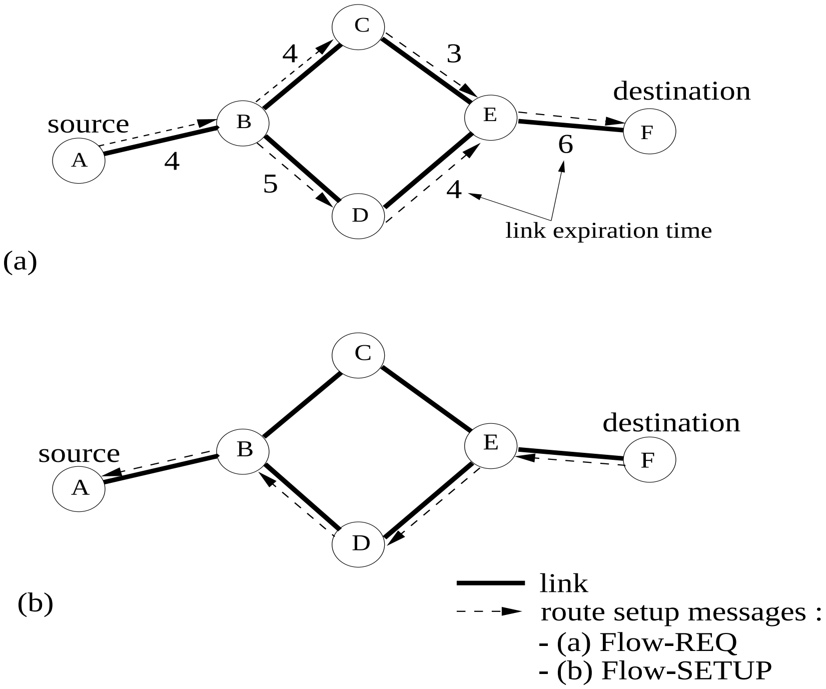


Fig. 1. The flow setup process: (a) the propagation of FLOW-REQ messages on their way to the destination node F; (b) the intermediate nodes setup their flow states when they receive the FLOW-SETUP message.

last link of the message. When a FLOW-REQ message arrives at the destination, it contains the list of nodes along the route it has traveled and the LETs for each hop along the route. The destination can then determine the RET for the route by using the minimum of the set of LETs along the route. The rationale is that if a single link on a path is disconnected, the entire path will be disconnected. We assume that all nodes have a common time reference (e.g., using GPS). If the received route is more stable than the one currently in use, the destination sends a FLOW-SETUP message back to the source along the chosen route. The intermediate nodes set up the flow state when they receive the FLOW-SETUP message. This flow setup process is shown in Figure 1.

In Figure 1(a), the source node A sends a FLOW-REQ message for destination node F. Nodes B, C, D, and E will forward the FLOW-REQ messages and append information such as their own node ID and the LET of the last link that the message was received from. Therefore in our example, two FLOW-REQ messages arrive at node F. One contains path (A,B,C,E,F) with LETs = (4,4,3,6), and the other contains path (A,B,D,E,F) with LETs = (4,5,4,6). Since RET is the minimum of the set of LETs for the route, node F can then obtain the RET for both routes. In our example, path (A,B,D,E,F) is more stable than path (A,B,C,E,F) since it has larger RET of value 4. Therefore it is chosen as the route to setup the flow. As shown in Figure 1(b), node F then sends the FLOW-SETUP message and the intermediate nodes will setup the flow states. Note that in the route selection phase, other criterion can be used to determine the optimum route to set up the flow such as route QoS (Quality of Service), load, etc. It is a good idea to take a weighted average of the criteria into consideration when selecting a route. We will explore these other factors in future

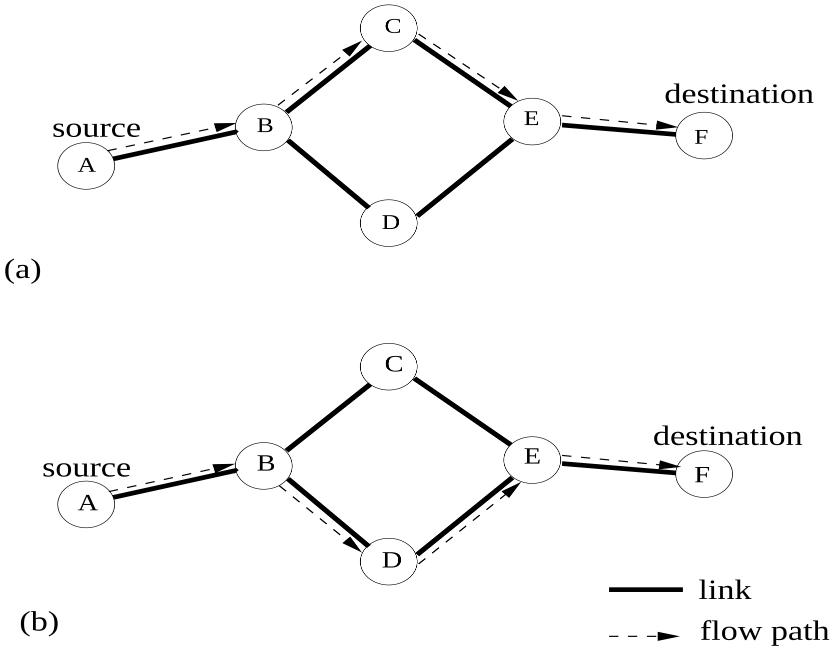


Fig. 2. The handoff process: (a) the original route, with nodes A and F being the source and the destination; (b) the new route after the handoff is completed

work and focus on the stability of routes in this paper.

During the connection, intermediate nodes append LETs to each data packet, so the destination continues to receive RET predictions from data packets. When the destination has determined that the "critical time," or the time that the route is about to expire, is reached, a Flow-HANDOFF message is generated by the destination and propagated via flooding in the same way as the FLOW-REQ message. After the source receives a Flow-HANDOFF message, it determines the best route on which to handoff the flow based on the information contained in the Flow-HANDOFF message (e.g., RET, number of hops, etc.). The source then sends a FLOW-SETUP message along the new route. Note that this FLOW-SETUP message is the same as the one used in the FLOW-SETUP phase described previously in Figure 1, except now it travels from the source to the destination.

The handoff process is shown in Figure 2. In Figure 2(a), the old route is along the route of node (A-B-C-E-F) and it is about to expire. In Figure 2(b), just before the "critical time" is reached, the handoff is performed and the new route is now (A-B-D-E-F).

In FORP, we define the "critical time" T_c by

$$T_c = RET - T_d$$

where T_d is the delay experienced by the last packet which has arrived along the route. Since network load conditions will change from time to time during the connection, the delay will also change accordingly. By using the latest arrived packet to calculate T_c, the scheme is adaptive to changing network conditions and the source will be correctly informed in a timely manner for a handoff so the packet losses can be minimized

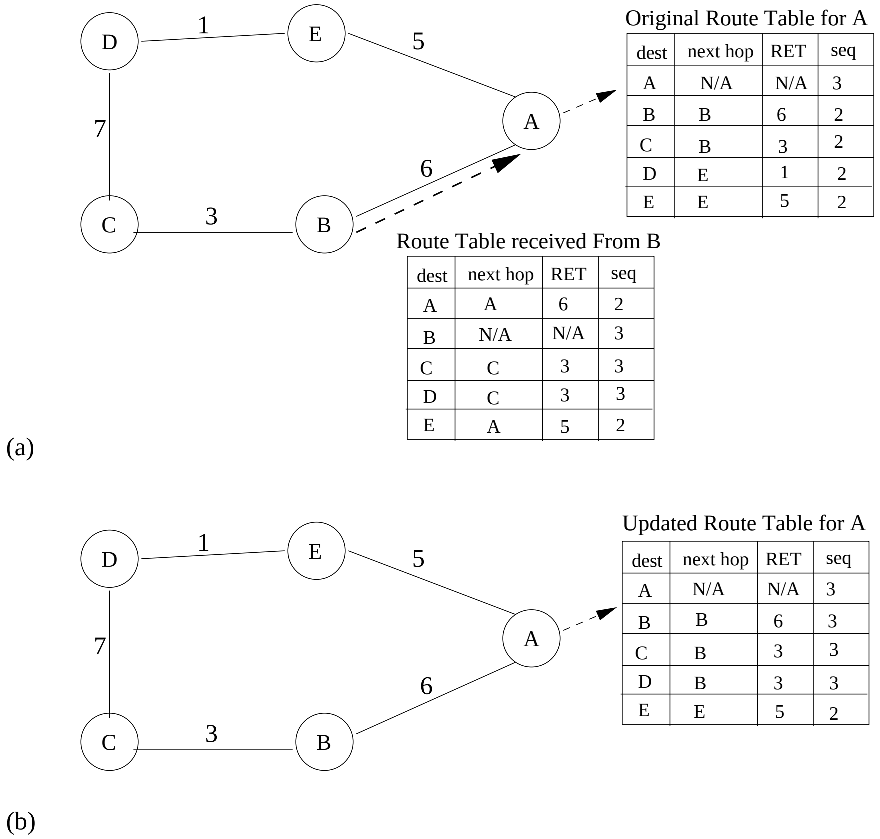
B.2 The Distance Vector Approach

Distance vector routing protocols maintain the most recent routing information by exchanging routing tables (RT) with neighbor nodes. Distance vector protocols have two kinds of RT updates, namely periodical updates and triggered updates. Triggered updates are generated whenever a node detects any link change for its neighbors. One of the strengths of distance vector protocols is that the waiting for route discovery is not necessary since routes are maintained proactively. Distance vector protocols are also attractive for networks with large number of senders, since the routing overhead is constant regardless of the number of senders in the network. On the other hand, when using an on-demand based protocol, the flooding of route discovery packets increases as the number of senders increases. This can lead to serious performance degradation when there are a significant number of active source-destination pairs in the network.

However, the performance of distance vector protocols is very sensitive to the periodic routing table update interval. In high mobility conditions, the routes need to be updated more often and the update interval must be shortened to improve performance. However, the short update interval will cause the routing overhead to increase. In slow mobility conditions, the frequent generation of routing tables may create excessive and unnecessary overhead since the routes do not need to be updated as often. In an ad hoc network, it is very difficult to select a routing update interval that will work for every scenario since the mobility condition can be very dynamic. DSDV and WRP which are modifications to distance vector algorithm, react to changing topology by generating triggered updates when connectivity changes are detected on top of the regular periodic updates. However, the responsiveness of the triggered updates depends on the interval of the periodic hello beacons used to detect connectivity changes. Moreover, using triggered update may lead to severe network congestion when the network is very dynamic.

In this section we propose a distance vector protocol with mobility prediction enhancements. The protocol uses route expiration time as the route selection metric in the route table. Therefore, data packets are sent over the most stable route. The use of triggered updates is also eliminated, since the routes are established based on stability. This means that the routing update interval can be relaxed and frequent updates are no longer required. In addition, using more stable routes minimizes the disruption caused by mobility since a different route with greater expiration time is used prior to a given route becomes invalid. This can be extremely effective when sending real-time data (i.e., voice, video, etc.) across a dynamic mobile network. The ultimate goal of using prediction is to make the routing protocol more robust to mobility and more bandwidth efficient. The proposed protocol requires a periodic broadcast of routing table by each node. Unlike the on demand approach, no route acquisition time is needed since routes are maintained proactively and the routing overhead remains constant regardless of the number of senders in the network.

In our enhancement distance vector method, to utilize the information obtained from prediction (i.e., LET, RET), mobility vector field for a node must be appended to the route table when it is being propagated. Also, the RET (Route Expiration Time) metric is added to the routing table for each entry. The route table is broad-



Original Route Table for A

dest	next hop	RET	seq
A	N/A	N/A	3
B	B	6	2
C	B	3	2
D	E	1	2
E	E	5	2

Route Table received From B

dest	next hop	RET	seq
A	A	6	2
B	N/A	N/A	3
C	C	3	3
D	C	3	3
E	A	5	2

Updated Route Table for A

dest	next hop	RET	seq
A	N/A	N/A	3
B	B	6	3
C	B	3	3
D	B	3	3
E	E	5	2

Fig. 3. An example of routing table update.

casted by each node periodically. A sequence number is used by each node when generating updates, and it is incremented after each routing table broadcast. The sequence number is then associated with routing table entries for the particular origin of the sequence number. This is to ensure that the latest routing information for a particular source gets updated correctly as the information propagates away from the source. In the proposed scheme, if node A receives a route table from its neighbor node B, the mobility vector contained in the routing table is used to calculate the LET between node A and node B. The LET and the routing table received from node B is then used to update node A's own routing table by with the following rules:

- If an entry for destination D with a better RET is received in the incoming routing table and the sequence number received is greater than or equals to the old entry, node A's entry for destination D is updated.
- If A's current next hop for destination D is node B and the incoming routing table entry for node D contains a higher sequence number than node A's entry for node D, then node A's routing table is updated.

The process is shown in Figure 3. In Figure 3, the LETs are shown beside each link in the network topology. In Figure 3 (a), node A's next hop to node D is node E and the RET through node E is 1. After node A receives the route update packet from node B, it will update its next hop for destination D to node B as shown in Figure 3 (b) because the route via node B possesses a higher RET.

There is a trade off between route optimality and route stability. A route that has the larger RET will remain active longer, but is might not necessarily be the route with the smaller hop count and/or delay.

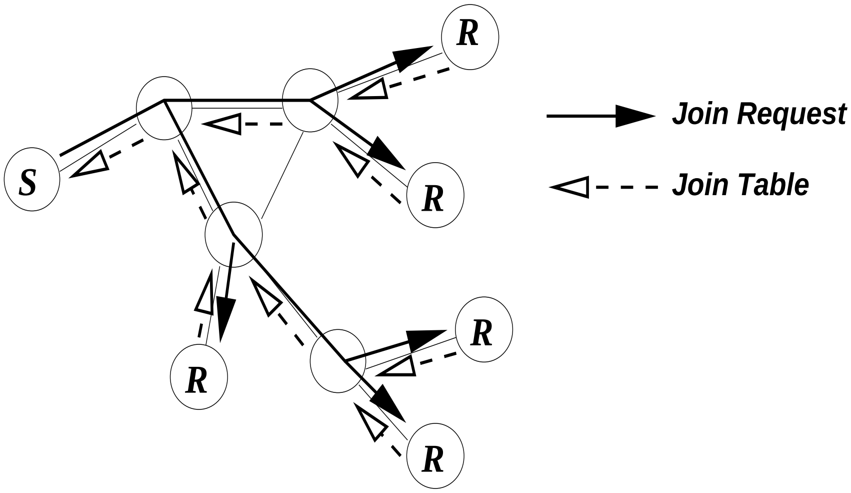


Fig. 4. On-demand procedure for membership setup and maintenance.

C. Applying Mobility Prediction to Multicast Protocols

C.1 ODMRP Overview

In this section, we use On-Demand Multicast Routing Protocol (ODMRP) [16], [17] as a protocol applying our mobility prediction method. In ODMRP, group membership and multicast routes are established and updated by the source *on demand*. Similar to on-demand unicast routing protocols, a request phase and a reply phase comprise the protocol (see Figure 4). While a multicast source has packets to send, it floods a member advertising packet with data payload piggybacked. This packet, called JOIN DATA, is periodically broadcasted to the entire network to refresh the membership information and update the routes as follows. When a node receives a non-duplicate JOIN DATA, it stores the upstream node ID (i.e., backward learning) into the routing table and rebroadcasts the packet. When the JOIN DATA packet reaches a multicast receiver, the receiver creates and broadcasts a JOIN TABLE to its neighbors. When a node receives a JOIN TABLE, it checks if the next node ID of one of the entries matches its own ID. If it does, the node realizes that it is on the path to the source and thus is part of the forwarding group. It then sets the FG_FLAG (Forwarding Group Flag) and broadcasts its own JOIN TABLE built upon matched entries. The JOIN TABLE is thus propagated by each forwarding group member until it reaches the multicast source via the shortest path. This process constructs (or updates) the routes from sources to receivers and builds a mesh of nodes, the *forwarding group*.

We have visualized the forwarding group concept in Figure 5. The forwarding group is a set of nodes which is in charge of forwarding multicast packets. It supports shortest paths between any member pairs. All nodes inside the “bubble” (multicast members and forwarding group nodes) forward multicast data packets. Note that a multicast receiver also can be a forwarding group node if it is on the path between a multicast source and another receiver. The mesh provides richer connectivity among multicast members compared to trees. Flooding redundancy among forwarding group helps overcome node displacements and channel fading. Hence, unlike trees, frequent reconfigurations are not required.

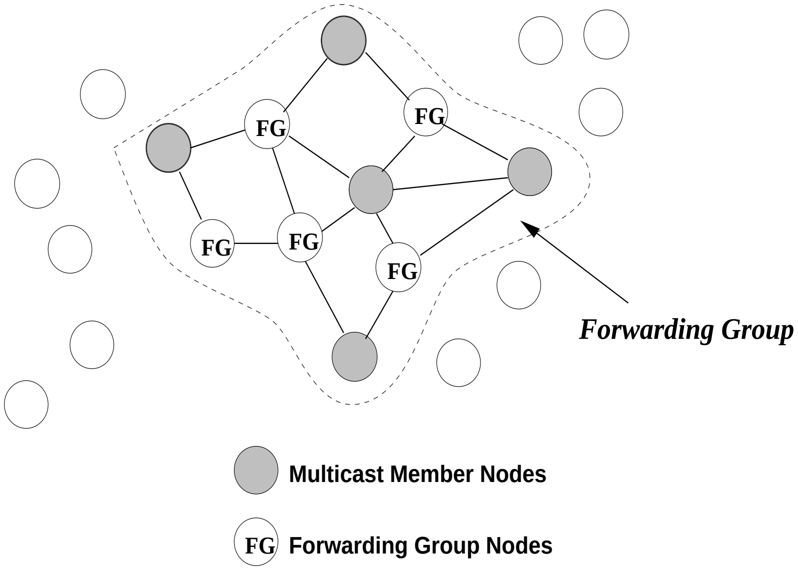


Fig. 5. The forwarding group concept.

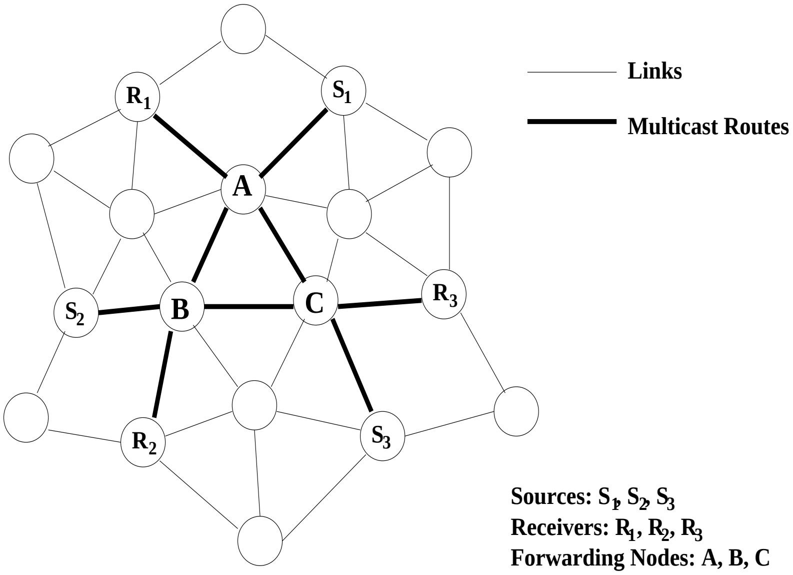
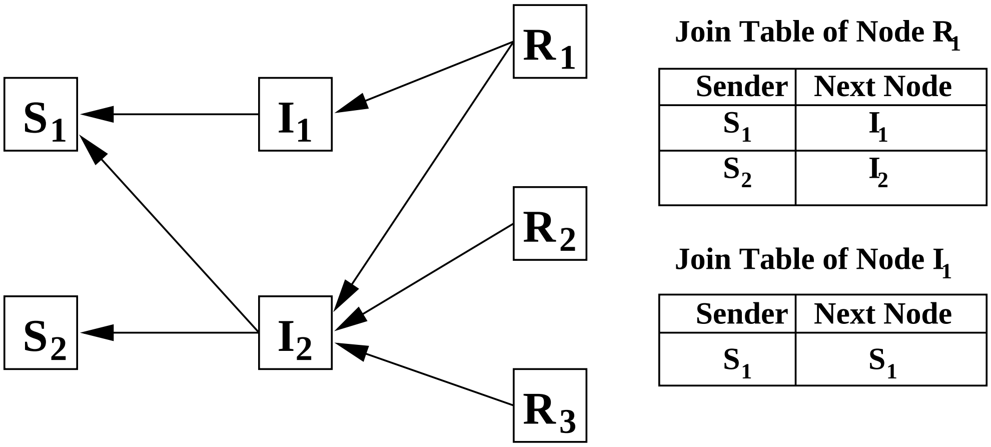


Fig. 6. Why a mesh?

Figure 6 is an example to show the robustness of a mesh configuration. Three sources (S_1, S_2, and S_3) send multicast data packets to three receivers (R_1, R_2, and R_3) via three forwarding group nodes (A, B, and C). Suppose the route from S_1 to R_2 is (S_1-A-B-R_2). In a tree configuration, if the link between nodes A and B breaks or fails, R_2 cannot receive any packets from S_1 until the tree is reconfigured. ODMRP, on the other hand, already has a redundant route (e.g., (S_1-A-C-B-R_2)) to deliver packets without going through the broken link between nodes A and B.



Join Table of Node R_1

Sender	Next Node
S_1	I_1
S_2	I_2

Join Table of Node I_1

Sender	Next Node
S_1	S_1

Fig. 7. An example of a Join Table forwarding.

C.2 Example

Let us consider Figure 7 as an example of a Join Table forwarding process. Nodes S_1 and S_2 are multicast sources, and nodes R_1, R_2, and R_3 are multicast receivers. Nodes R_2 and R_3 send their Join Tables to both S_1 and S_2 via I_2. R_1 sends its Join Table to S_1 via I_1 and to S_2 via I_2. When receivers send their Join Tables to next hop nodes, an intermediate node I_1 sets the FG_FLAG and builds its own Join Table since there is a next node ID entry in the Join Table received from R_1 that matches its ID. Note that the Join Table built by I_1 has an entry for sender S_1 but not for S_2 because the next node ID for S_2 in the received Join Table is not I_1. In the meantime, node I_2 sets the FG_FLAG, constructs its own Join Table and sends it to its neighbors. Note that even though I_2 receives three Join Tables from the receivers, it broadcasts the Join Table only once because the second and third table arrivals carry no new source information. Channel overhead is thus reduced dramatically in cases where numerous multicast receivers share the same links to the source.

C.3 Reliability

The reliable transmission of Join Tables plays an important role in establishing and refreshing multicast routes and forwarding groups. Hence, if Join Tables are not properly delivered, effective multicast routing cannot be achieved by ODMRP. The IEEE 802.11 MAC (Medium Access Control) protocol [18], which is the emerging standard in wireless networks, performs reliable transmission by retransmitting the packet if no acknowledgment is received. However, if the packet is broadcasted, no acknowledgments or retransmissions are sent. In ODMRP, the transmission of Join Tables are often broadcasted to more than one upstream neighbors since we are handling multiple sources (e.g., see the Join Table from node R_1 in Figure 7). In such cases, the hop-by-hop verification of Join Table delivery and the retransmission cannot be handled by the MAC layer. It must be done indirectly by ODMRP.

We adopt a scheme that was used in [13]. Figure 8 is shown to illustrate the mechanism. When node B

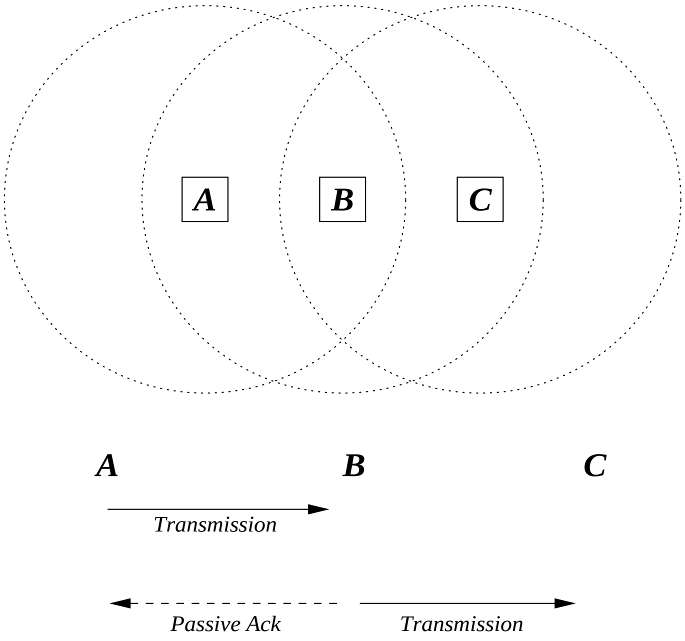


Fig. 8. Passive acknowledgments.

transmits a packet to node C after receiving a packet from node A, node A can hear the transmission of node B if it is within B's radio propagation range. Hence, the packet transmission by node B to node C is used as a *passive acknowledgment* to node A. We can utilize this passive acknowledgment to verify the delivery of a JOIN TABLE. Note that the source itself must send an active acknowledgment to the previous hop since it does not have any next hop to send a JOIN TABLE to unless it is also a forwarding group node for other sources.

If packet delivery cannot be verified after an appropriate number of retransmissions, the node considers the route to be invalidated. At this point, the most likely cause of route failure is the fact that a node on the route has failed or has moved out of range. An alternate route must be found "on the spot." The node thus broadcasts a message to its neighbors specifying that the next hop to a set of sources cannot be reached. Upon receiving this packet, each neighbor builds and unicasts the JOIN TABLE to its next hop if it has a route to the multicast sources. If no route is known, it simply broadcasts the packet specifying the next hop is not available. In both cases, the node sets its FG_FLAG. In practical implementations, this redundancy is sufficient to establish alternate paths until a more efficient route is established during the next refresh phase. The FG_FLAG setting of every neighbor may create excessive redundancy, but most of these settings will expire because only necessary forwarding group nodes will be refreshed in the next JOIN TABLE propagation phase.

C.4 Data Forwarding

After the group establishment and route construction process, a source can multicast packets to receivers via selected routes and forwarding groups. When receiving the multicast data packet, a node forwards it only when it is not a duplicate and the setting of the FG_FLAG for the multicast group has not expired. This procedure minimizes the traffic overhead and prevents sending packets through stale routes.

C.5 Soft State

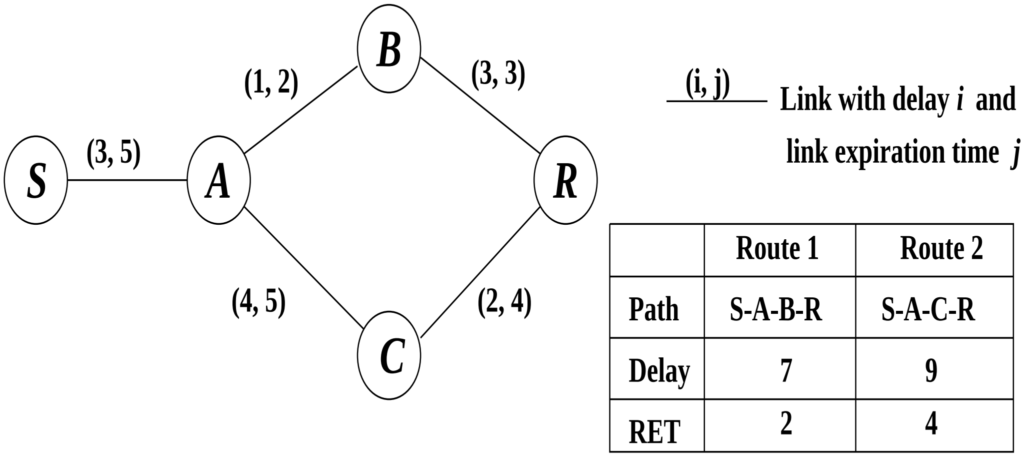
In ODMRP, no explicit control packets need to be sent to join or leave the group. If a multicast source wants to leave the group, it simply stops sending JOIN DATA packets since it does not have any multicast data to send to the group. If a receiver no longer wants to receive from a particular multicast group, it does not send the JOIN TABLE for that group. Nodes in the forwarding group are demoted to non-forwarding nodes if not refreshed (no JOIN TABLES received) before they timeout.

C.6 Applying Mobility Prediction

ODMRP requires periodic flooding of JOIN DATA to build and refresh routes. Excessive flooding, however, is not desirable in ad hoc networks because of bandwidth constraints. Furthermore, flooding often causes congestion, contention, and collisions. Finding the optimal flooding interval is critical in ODMRP performance. Here we propose a scheme that adapts the flooding interval to mobility patterns and speeds. By utilizing the location and mobility information provided by GPS (Global Positioning System) [14], we predict the duration of time routes will remain valid. With the predicted time of route disconnection, JOIN DATA are only flooded when route breaks of ongoing data sessions are imminent.

For our proposed enhancement, extra fields must be added into JOIN DATA and JOIN TABLE packets. When a source sends JOIN DATA, it appends its location, speed, and direction. It sets the MIN_LET (Minimum Link Expiration Time) field to the MAX_LET_VALUE since the source does not have any previous hop node. The next hop neighbor, upon receiving a JOIN DATA, predicts the link expiration time between itself and the previous hop using the equation given in section II.A. The minimum between this value and the MIN_LET indicated by the JOIN DATA is included in the packet. The rationale is that as soon as a single link on a path is disconnected, the entire path is invalidated. The node also overwrites the location and mobility information field written by the previous node with its own information. When a multicast member receives the JOIN DATA, it calculates the predicted LET of the last link of the path. The minimum between the last link expiration time and the MIN_LET value specified in the JOIN DATA is the RET (Route Expiration Time). This RET value is enclosed in the JOIN TABLE and broadcasted. If a forwarding group node receives multiple JOIN TABLES with different RET values (i.e., lies in paths from the same source to multiple receivers), it selects the minimum RET among them and sends its own JOIN TABLE with the chosen RET value attached. When the source receives JOIN TABLES, it selects the minimum RET among all the JOIN TABLES received. Then the source can build new routes by flooding a JOIN DATA before the minimum RET approaches (i.e., route breaks). Note that JOIN TABLES need not be periodically transmitted by multicast receivers. Since sources flood JOIN DATA only when needed, receivers only send JOIN TABLES after receiving JOIN DATA.

In addition to the estimated RET value, other factors need to be considered when choosing the flooding in-



	Route 1	Route 2
Path	S-A-B-R	S-A-C-R
Delay	7	9
RET	2	4

Fig. 9. Route selection example.

terval of JOIN DATA. If the node mobility rate is high and the topology changes frequently, routes will expire quickly and often. The source may propagate JOIN DATA excessively and this excessive flooding can cause collisions and congestion, and clogs the network with control packets. Thus, the `MIN_REFRESH_INTERVAL` should be enforced to avoid control message overflow. On the other hand, if nodes are stationary or move slowly and link connectivity remains unchanged for a long duration of time, routes will hardly expire and the source will rarely send JOIN DATA. A few problems arise in this situation. First, if a node in the route suddenly changes its movement direction or speed, the predicted RET value becomes obsolete and routes will not be reconstructed in time. Second, when a non-member node which is located remotely to multicast members wants to join the group, it cannot inform the new membership or receive data until a JOIN DATA is received. Hence, the `MAX_REFRESH_INTERVAL` should be set. The selection of the `MIN_REFRESH_INTERVAL` and the `MAX_REFRESH_INTERVAL` should be adaptive to network situations (e.g., traffic type, traffic load, mobility pattern, mobility speed, channel capacity, etc.).

In the basic ODMRP, a multicast receiver selects routes based on the minimum delay (i.e., routes taken by the first JOIN DATA received). A different route selection method is applied when we use the mobility prediction. The idea is inspired by the Associativity-Based Routing (ABR) protocol [29] which chooses associatively stable routes. In our new algorithm, instead of using the minimum delay path, we can choose a route that is the most stable (i.e., the one with the largest RET). To select a route, a multicast receiver must wait for an appropriate amount of time after receiving the first JOIN DATA so that all possible routes and their RETs will be known. The receiver then chooses the most stable route and broadcasts a JOIN TABLE. Route breaks will occur less often and the number of JOIN DATA propagation will reduce because stable routes are used. An example showing the difference between two route selection algorithms is presented in Figure 9. Two routes are available from the source S to the receiver R. Route 1 has a path of (S-A-B-R) and route 2 has a path of (S-A-C-R). If the minimum delay is used as the route selection metric, the receiver node R selects route 1. Route 1 has a delay of 7 ($3+1+3=7$) while route 2 has a delay of 9 ($3+4+2=9$). Since the JOIN DATA that takes route 1 reaches the receiver first, node R chooses route 1. If the stable route is selected instead, route 2 is chosen by the receiver. The route expiration time of route 1 is 2 ($\min(5, 2, 3) = 2$) while that of route 2 is 4 ($\min(5, 5, 4) = 4$). The receiver selects the route with the maximum RET, and hence route 2 is selected.

D. Effect of Prediction Accuracy on Performance

So far we assumed the mobility information obtained from GPS is always accurate and each node has a simple mobility pattern (e.g., no sudden changes of direction, each node's trajectory follows a straight line, constant velocity for all nodes, etc.). Under these conditions, the route disconnection time of a mobile can always be accurately predicted. However, this assumption cannot hold true in some scenarios. In the real world, the GPS reading obtained may not always be accurate due to various reasons (e.g., multipath fading, indoor conditions, etc.). Moreover, a mobile can accelerate, decelerate, and change its direction while it is traveling. All these factors can cause the route expiration time prediction to become inaccurate since such events are generally not predictable.

Incorrect prediction can be caused by GPS inaccuracies. GPS inaccuracies can range from ± 5 meters for differential GPS equipments [8] to ±100 meters for regular GPS equipments [14]. For our proposed mobility prediction scheme, we are using the GPS readings to predict link expiration time. We plan to investigate the effect on performance when the GPS inaccuracy is varied in our simulations.

III. SIMULATION ENVIRONMENT

The simulator was implemented within the Global Mobile Simulation (GloMoSim) library [30]. The GloMoSim library is a scalable simulation environment for wireless network systems using the parallel discrete-event simulation capability provided by PARSEC [2]. Our simulation modeled a network of 50 mobile hosts placed randomly within a 1000m × 1000m area. Radio propagation range for each node was 250 meters and channel capacity was 2 Mbits/sec. Each simulation executed for 600 seconds of simulation time. Multiple runs with different seed numbers were conducted for each scenario and collected data were averaged over those runs.

A free space propagation model [26] with a threshold cutoff was used in our experiments. In the free space model, the power of a signal attenuates as $1/d^2$ where d is the distance between radios. In the radio model, we assumed the ability of a radio to lock on to a sufficiently strong signal in the presence of interfering signals, i.e., radio capture. If the capture ratio (the minimum ratio of an arriving packet's signal strength relative to those of other colliding packets) [26] was greater than the predefined threshold value, the arriving packet was received while other interfering packets were dropped. The IEEE 802.11 Distributed Coordination Function (DCF) [18] was used as the medium access control protocol. A traffic generator was developed to simulate constant bit rate sources. The size of data payload was 512 bytes. Each node moved constantly with the predefined speed. Moving direction was selected randomly, and when nodes reached the simulation terrain boundary, they bounced back and continued to move.

A. Scenarios

To assess the performance of our proposed enhancements, we simulated the following scenarios under various conditions.

A.1 Unicast Routing Evaluation

In this scenario, we examine the performance of unicast prediction enhancements. We simulated and compared the following schemes: FORP (Flow Oriented Routing Protocol), an on demand protocol with mobility enhancement that we described in Section II.B.1; LAR (Location Aided Routing) [15], an on demand routing protocol that uses GPS; DV-MP (Distance Vector protocol with Mobility Prediction that we proposed in Section II.B.2); and WRP (Wireless Routing Protocol) [19], a distance vector routing protocol for ad hoc networks).

All schemes were evaluated as a function of (i) speed and (ii) number of unicast UDP (User Datagram Protocol) data sessions. For simplicity (but, without loss of generality) we assume no reliable transport protocol, i.e., TCP (Transmission Control Protocol) which may carry out retransmission of lost packets. In the experiments in which we varied the mobility speed, number of data sessions was set to 5 and speed was varied from 0 km/hr to 72 km/hr. In the experiments that we varied the number of data sessions, mobility speed was set to 36 km/hr and the number of sessions was varied from 5 to 30. In all the experiments for this scenario, the total number of data packets were sent at the rate of 20 packets/sec.

A.2 Multicast Routing Evaluation

We measure the performance improvement made by mobility prediction in multicast routing protocols. The protocols that we simulated are : ODMRP-MP (ODMRP with Mobility Prediction enhancements as we have described in Section II.C), and ODMRP (ODMRP without mobility prediction). The network conditions that we vary in this scenario are (i) speed, (ii) multicast group density, and (iii) number of multicast source. In the experiments in which we varied the speed, a multicast group of size 10 with one sender is used. The sender sends CBR (Constant Bit Rate) data at the rate of 10 packets per second, and speed is varied from 0 km/hr to 72 km/hr. In the experiments where we varied the multicast group density, mobility speed is fixed at 36 km/hr while multicast group size is varied from 3 to 20. The group has one sender and it sends data at the rate of 10 packets per second. In the experiments that we varied the number of multicast source, number of group size is fixed at 10, while the number of senders for the group is varied from 1 to 10. The senders sends packets at the aggregate rate of 10 packets per second.

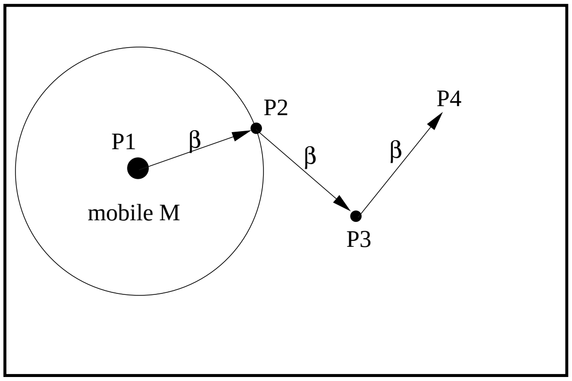


Fig. 10. The modified waypoint model, shown as M is at waypoint P1. β is the waypoint distance.

A.3 Effect of Prediction Error

In this scenario, we investigate the effect of prediction accuracy on routing protocol performance when mobility prediction is used. Our goal is to determine the enhancement's dependency on the accuracy of mobility prediction. We simulated DV-MP (Distance Vector protocol with Mobility Prediction we proposed in Section B.2) and DV (the pure Distance Vector protocol without mobility prediction). The conditions that we varied in this scenario are (i) GPS accuracy in meters, (ii) frequency of direction changes, and (iii) waypoint distance in the random waypoint mobility model [11]. Random waypoint mobility model randomly selects successive destination location points for the node to travel to. Upon reaching a destination point, a new target location is selected and the whole process is repeated again. When a destination position is reached, a mobile can pause for a certain period of time before moving towards the new waypoint again. In this experiment, we make modifications to the waypoint model by limiting the waypoint distance (i.e., distance between two successive random destinations) to β as shown in Figure 10. Therefore, when mobile M reaches waypoint P_1, a new waypoint P_2 is selected with distance β. This process is repeated when P_2 is reached, and so on. Sine mobile M changes its direction each time a new waypoint is reached, decreasing the waypoint distance β will increase the randomness of the mobility pattern. On one end of the spectrum, a very large waypoint distance will result in a straight line trajectory mobility model. The other extreme of having a very small waypoint distance will result in a Brownian-motion like pattern.

In the experiments that we varied the GPS accuracy, mobility speed is fixed at 18 km/hr for the low speed condition and 72 km/hr for the high speed condition. The GPS accuracy is varied from 5 to 150 meters. In the experiments where we varied the frequency of directional changes, mobility speed is fixed at 36 km/hr and direction changing frequency is varied from 0 times/second (straight trajectory) to 5 times/second. In the experiments in which we varied the waypoint distance, mobility speed is fixed at 18 km/hr and the waypoint distance is varied from 10 meters to 175 meters. For all experiments in this scenario, 5 pairs of data session are used and each session sends data at the rate of 10 packets per second. The routing update interval for DV and DV-MP is fixed at 1.5 second.

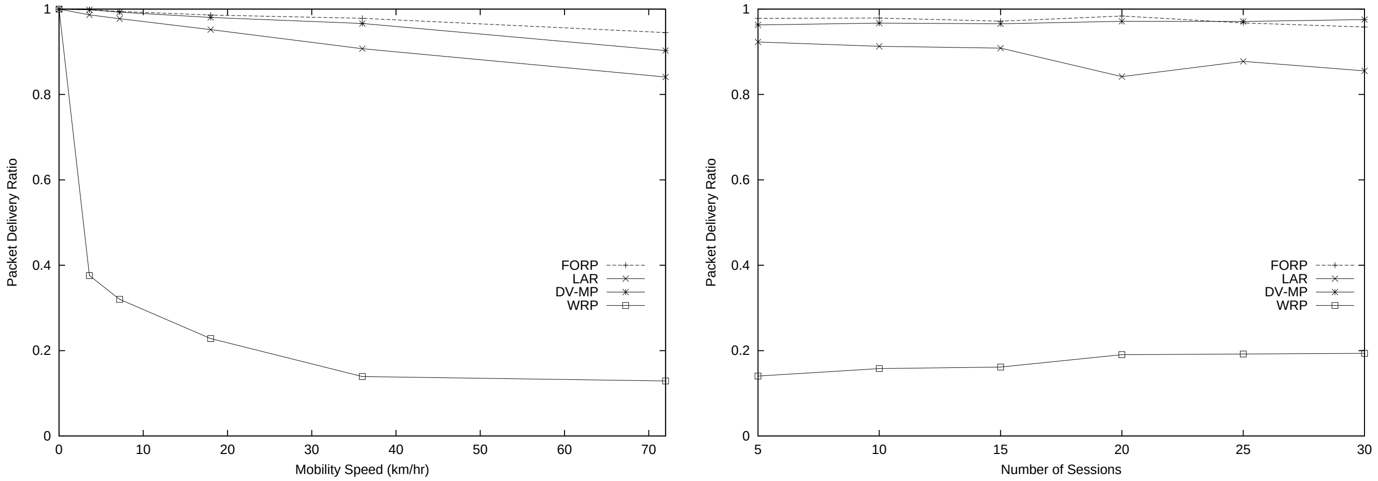


Fig. 11. Packet delivery ratio as a function of: (a) speed, (b) number of Sessions.

B. *Metrics of Interest*

For all scenarios that we simulated, the metrics of interest are:

- **Packet delivery ratio**: The number of data packets received by destinations over the number of data packets supposed to be received by destination nodes.
- **Number of control bytes transmitted per data byte delivered**: Instead of the pure control overhead, we use a ratio of control bytes transmitted and data byte delivered to investigate how efficiently control packets are utilized in delivering data. Only bytes of the data payload contributes to the data bytes delivered. Accordingly, data packet header as well as control packets are included in the control byte overhead.
- **Number of total packets transmitted per data packet delivered**: The number of all packets (i.e., data and control packets) transmitted divided by the number of data packet delivered to destinations. This measure shows the efficiency in terms of channel access and is very important in ad hoc networks since link layer protocols are typically contention-based.

IV. Simulation Results

A. *Scenario 1 - Unicast Enhancements*

A.1 Packet Delivery Ratio

The packet delivery ratio as a function of mobility speed and the number of data sessions is shown in Figure 11. We can see from Figure 11(a) that as speed increases, the routing effectiveness of WRP degrades rapidly compared to the other schemes. As nodes move faster, link connectivity changes more often and more update messages are triggered. For each triggered update, neighbor nodes are required to send back an acknowledgment. Moreover, temporary loops were being formed because the network view converged slowly, with many changes needing to be absorbed and propagated. Loops, triggered updates, and ACKs

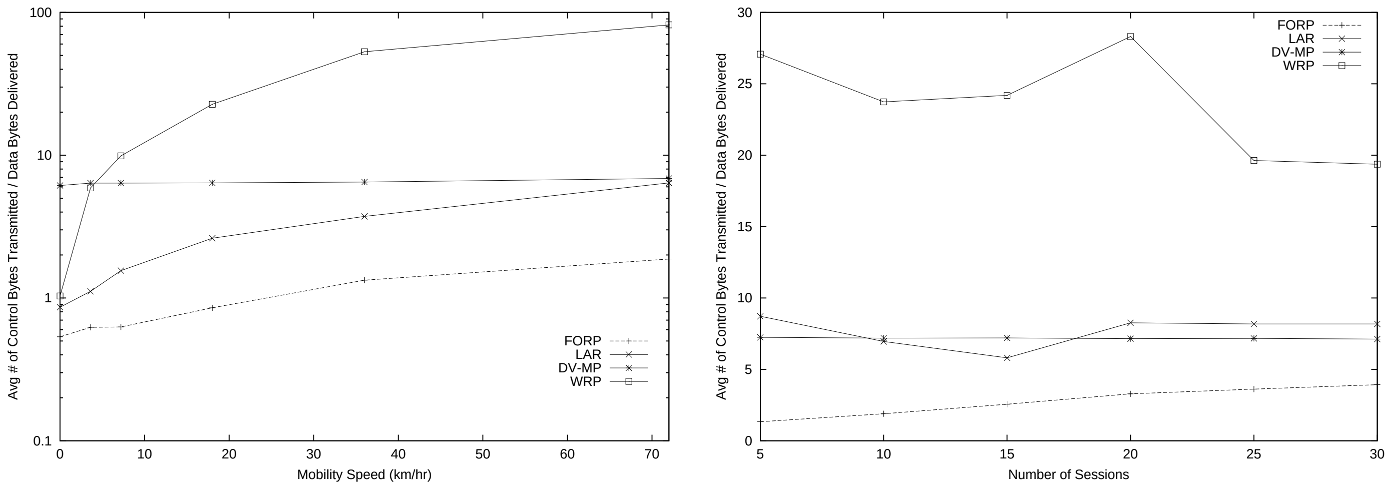


Fig. 12. The number of control bytes transmitted per data byte delivered as a function of: (a) speed, (b) number of sessions.

created an enormous amount of packets, contributing further to collisions, congestion, contention, and packet drops. FORP and DV-MP are the schemes that are the least affected by mobility, since they are able to maintain delivery ratio above 0.9 for all mobility speeds in our experiments. Using mobility prediction to perform rerouting prior to route disconnection and to send data over more stable routes minimized packet losses.

In Figure 11(b), FORP and DV-MP outperform the other schemes again. DV-MP shows no performance degradation when the number of sessions is increased and the degradation of FORP is very negligible. LAR also shows a high delivery ratio. The delivery ratio for WRP is significantly lower than other schemes because the mobility speed was relatively high (36 km/hr).

A.2 Number of Control Bytes Transmitted per Data Byte Delivered

Figure 12(a) shows the number of control bytes transmitted per data byte delivered as a function of mobility speed for each protocol. Remember that the transmission of control packets in DV-MP is periodically triggered without adapting to mobility speed. Hence, the ratio for DV-MP does not increase as the mobility speed increases. On the other hand, the overhead of FORP becomes larger as mobility speed increases. Since mobility prediction is applied to adapt to mobility speed, more FLOW-REQ and FLOW-SETUP are sent when mobility is high, resulting in more overhead. In WRP, route updates are produced more frequently in high mobility since there are more link changes. Therefore, WRP has the highest ratio in mobile situations because of the small number of delivered data packets and the large number of triggered updates. LAR also shows more overhead as mobility speed increases because more route breaks occur and they invoke route reconstruction procedures.

In Figure 12(b), we can see that FORP and DV-MP have better ratios than WRP and LAR. The ratios for DV-MP remains constant but the one for FORP grows as the number of session is increased. This is expected

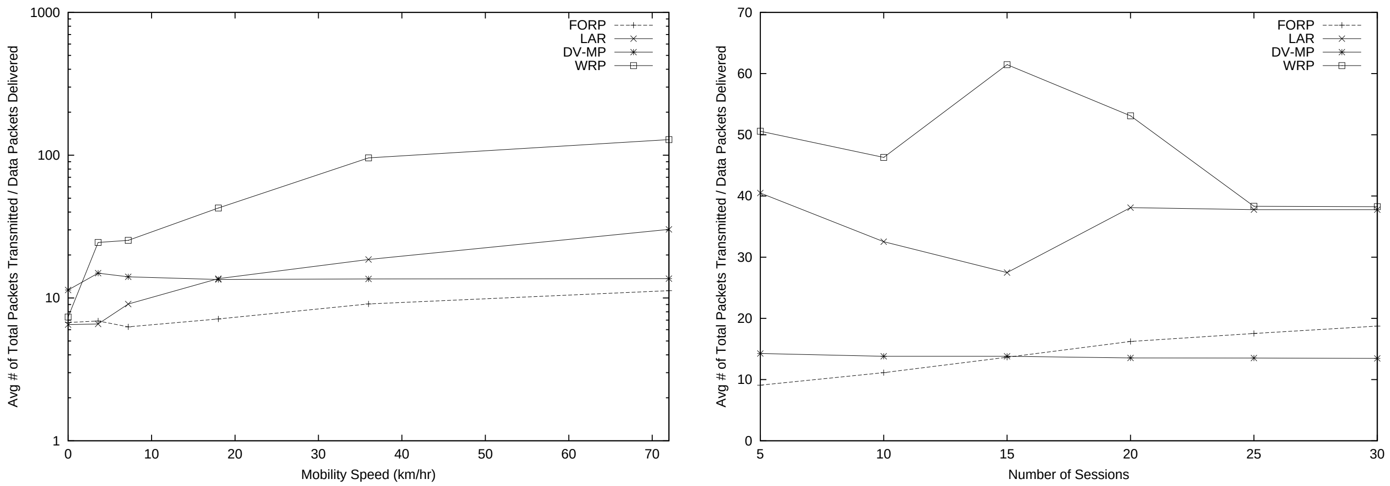


Fig. 13. The number of total packets transmitted per data packet delivered as a function of: (a) speed, (b) number of sessions.

since increasing number of senders will increase control message traffic for an on-demand scheme such as FORP.

A.3 Number of Total Packets Transmitted per Data Packet Delivered

The number of total packets (i.e., control packets, data packets) transmitted per data packet delivered is presented in Figure 13. We have mentioned previously that this measure indicates the channel access efficiency. From Figure 13(a) we can see that the numbers for FORP and DV-MP remain relatively constant, with FORP's ratio being lower than that of DV-MP. WRP has the highest ratio due to the same reasons described in Section IV-A.2. In Figure 13(b), FORP again has the best performance compared to other schemes when there are a small number of sources. However, DV-MP has better channel access efficiency over FORP as the number of sessions increases. We can conclude that DV-MP is more scalable than FORP in terms of number of sessions.

B. Scenario 2 - Multicast Enhancements

B.1 Effect of Mobility Speed on Performance

Moving now to the multicast scenarios, we show in Figure 14 the packet delivery ratio as a function of mobility speed. As speed increases, the routing effectiveness of ODMRP degrades rapidly compared to ODMRP_MP. ODMRP_MP has a very high delivery ratio (over 90%) regardless of speed. As the routes are reconstructed in advance of topology changes, most data are delivered to multicast receivers without being dropped. In ODMRP, on the other hand, JOIN DATA and JOIN TABLES are transmitted periodically without adapting to mobility speed and direction. At high speed, routes that are taken at the JOIN DATA phase may already be broken when JOIN TABLES are propagated.

The number of control bytes transmitted per data byte delivered as a function of speed is shown in Fig-

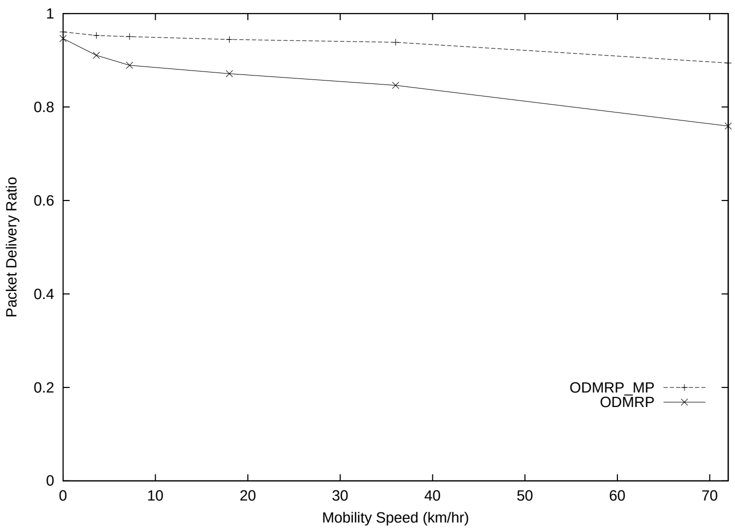


Fig. 14. Packet delivery ratio as a function of mobility speed.

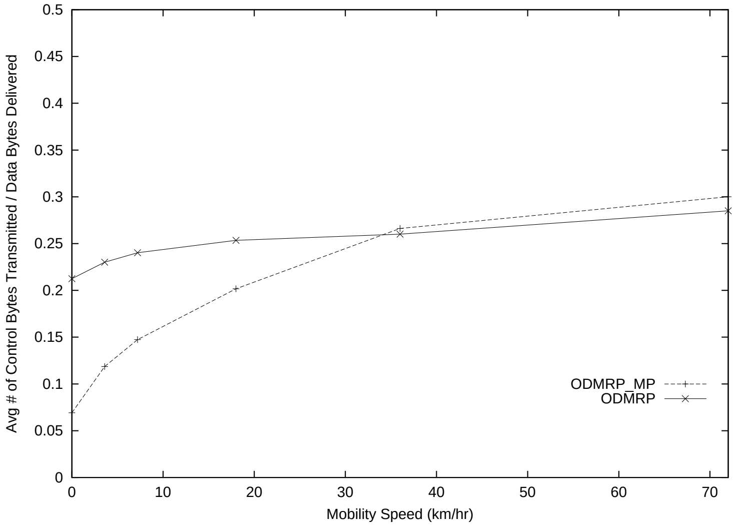


Fig. 15. Average number of control bytes transmitted per data byte delivered as a function of mobility speed.

ure 15. From slow to moderate mobility speeds, ODMRP_MP has significantly less control overhead compared to ODMRP because it only sends control packets when necessary. As mobility increases, more control packets are transmitted to adapt to mobility speed and construct the routes by ODMRP_MP. Thus, its ratios gradually become similar to those of ODMRP.

The number of total packets (i.e., JOIN DATA, JOIN TABLES, Data, and active acknowledgments) transmitted per data packet delivered is presented in Figure 16. The number for ODMRP remains relatively constant after an initial increase. Since the number of data packets delivered and the amount of control bytes transmitted both decrease as mobility increases, the number for ODMRP remains almost unchanging. The measures for ODMRP_MP increase with mobility speed. Since ODMRP_MP deliver a high portion of the data to destinations regardless of speed, more control packets must be sent in order to adapt to the increasing speed. Thus the total number of packets transmitted increases with speed. Additionally, since ODMRP_MP uses longer,

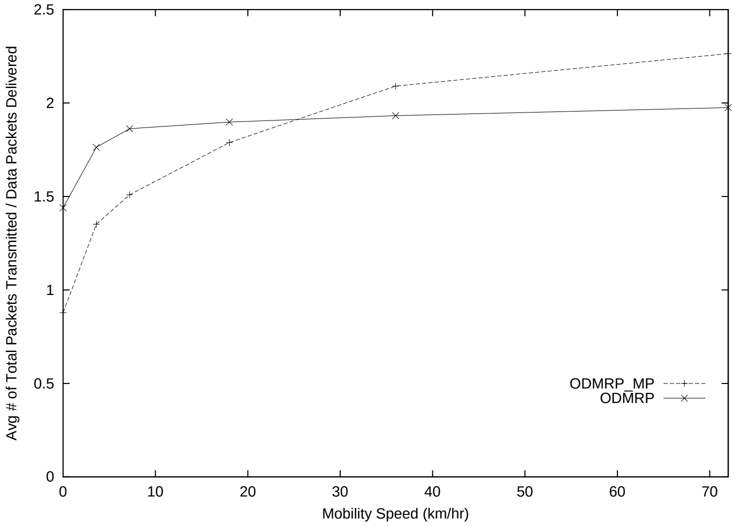


Fig. 16. Average number of total packets transmitted per data packets delivered as a function of mobility speed.

more stable routes compared to ODMRP, ODMRP_MP sends data over a longer hop length than ODMRP and therefore more data packets are transmitted.

B.2 Effect of Multicast Group Density on Performance

The results for both schemes when multicast group density is varied are presented in this section. In Figure 17 the packet delivery ratio as a function of group density is shown. ODMRP_MP is able to maintain a fairly high level of ratio above 90% regardless of group density while the performance for ODMRP improves slightly as group density increases. When the group is sparse, a fragile mesh with few redundant routes is built, thus more data packets are dropped when the primary route is broken. As group density increases, the portion of the network that becomes the forwarding mesh grows. This mesh provides more redundant routes and subsequently more data packets are delivered. Decreasing group density has no significant effect on ODMRP_MP performance since data are delivered over stable routes even when the group is sparse (group size ≤ 5).

The number of control bytes transmitted per data byte delivered as a function of group density is presented in Figure 18. Both protocols show decreasing value as the group becomes dense since more data packets are delivered with little or no increase of overhead. The result indicates that ODMRP_MP uses overhead more efficiently than ODMRP when the multicast group is relative sparse. This is consistent with results from Figure 17 since ODMRP delivers less data packets when the group is sparse. As group density increases, ODMRP_MP's ratios gradually approach those of ODMRP.

Figure 19 shows the results for number of total packets transmitted per data packets delivered as a function of group size. The result is very similar to Figure 18, with both schemes having decreasing curves when group density is increased. ODMRP_MP uses the channel more efficiently when the group is sparse due to the same reason discussed in Figure 18. However, when the group is dense (group size ≥ 15), ODMRP_MP's

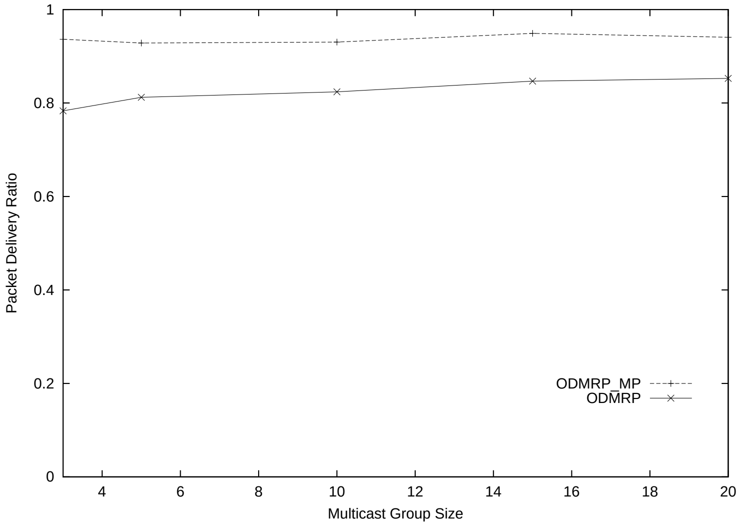


Fig. 17. Packet delivery ratio as a function of group size.

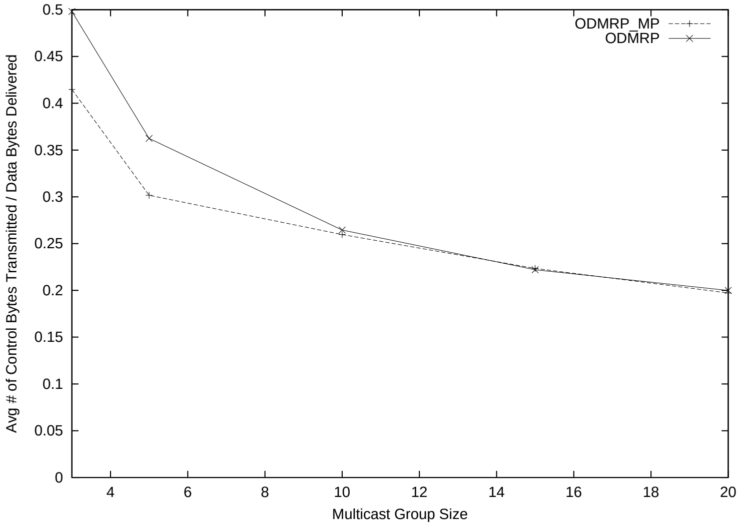


Fig. 18. Average number of control bytes transmitted per data byte delivered as a function of group size.

value is slightly larger than ODMRP's. This is also expected since ODMRP_MP uses more stable routes which may be longer in hops than ODMRP, so more data packets are transmitted.

B.3 Effect of Number of Multicast Source on Performance

In this section we examine the results when the number of senders in the multicast group is varied. We define α to be $\frac{N_s}{N_g}$, where N_s is the number of senders in the multicast group and N_g is the multicast group size. The case $\alpha = 0.1$ can represent the scenario when a lecture is given by an instructor to a group of students, while $\alpha = 1.0$ represent the scenario of a video conferencing session. In Figure 20, the packet delivery ratio as a function of α for both high and low mobility scenario is shown. We can see from Figure 20(a) and (b) that ODMRP_MP maintains high level of packet delivery ratio for all range of α. However, the delivery ratio for ODMRP degrades when α is decreased. This degradation is more significant in the high mobility case.

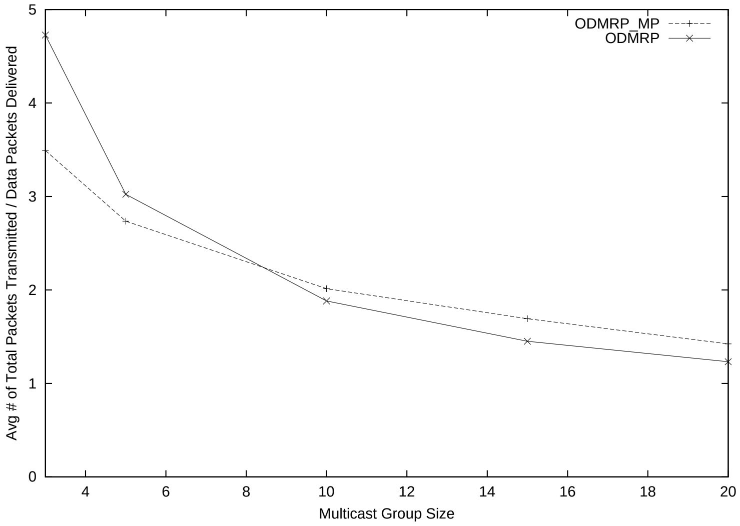


Fig. 19. Average number of total packets transmitted per data packets delivered as a function of group size.

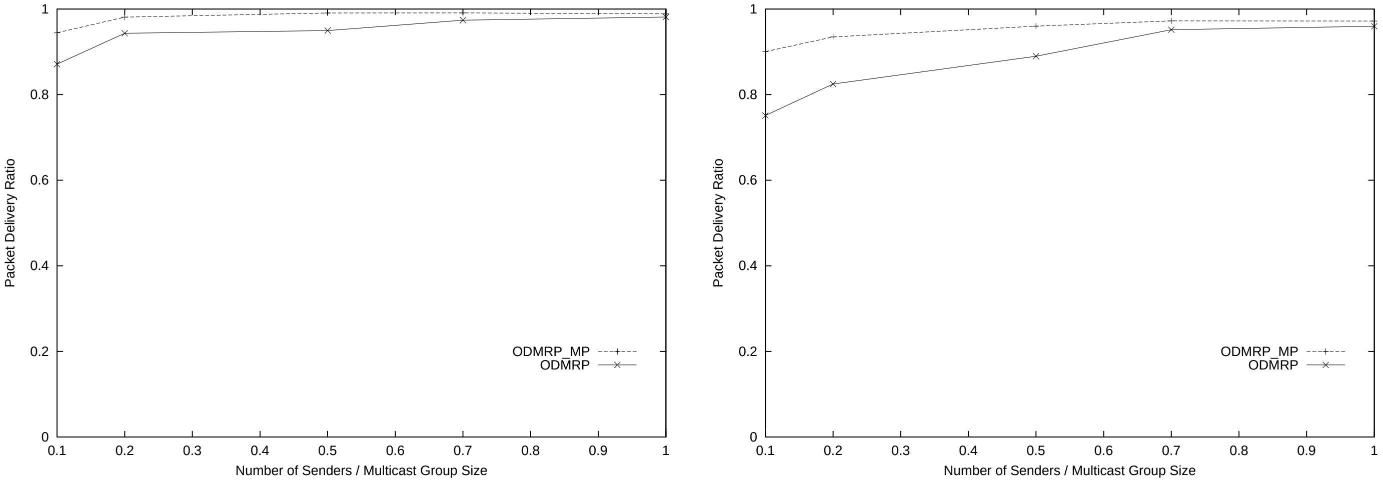


Fig. 20. Packet delivery ratio as a function of α for (a) low mobility case, (b) high mobility case.

As α increases, the number of forwarding group nodes becomes larger for ODMRP. Since member nodes are usually dispersed throughout the network, increasing the number of senders enlarges the portion of network that gets created as the mesh. Note that since ODMRP_MP uses the same mechanism to set up the mesh, it also exhibit the same behavior. However, ODMRP_MP's values are more stable in terms of α due to the use of more durable mesh.

The number of control bytes transmitted per data byte delivered as a function of α is presented in Figure 21. For both schemes the value increases as α increases since more control packets (i.e., JOIN TABLES, JOIN DATA, and active acknowledgments) are transmitted as the number of source goes up. ODMRP_MP is more efficient than ODMRP at low mobility because it is able to adapt to the slower mobility speeds and reduce the transmission of control packets. At high mobility, ODMRP_MP refreshes the routes almost at the same rate as ODMRP, and the number of control bytes transmitted are nearly identical.

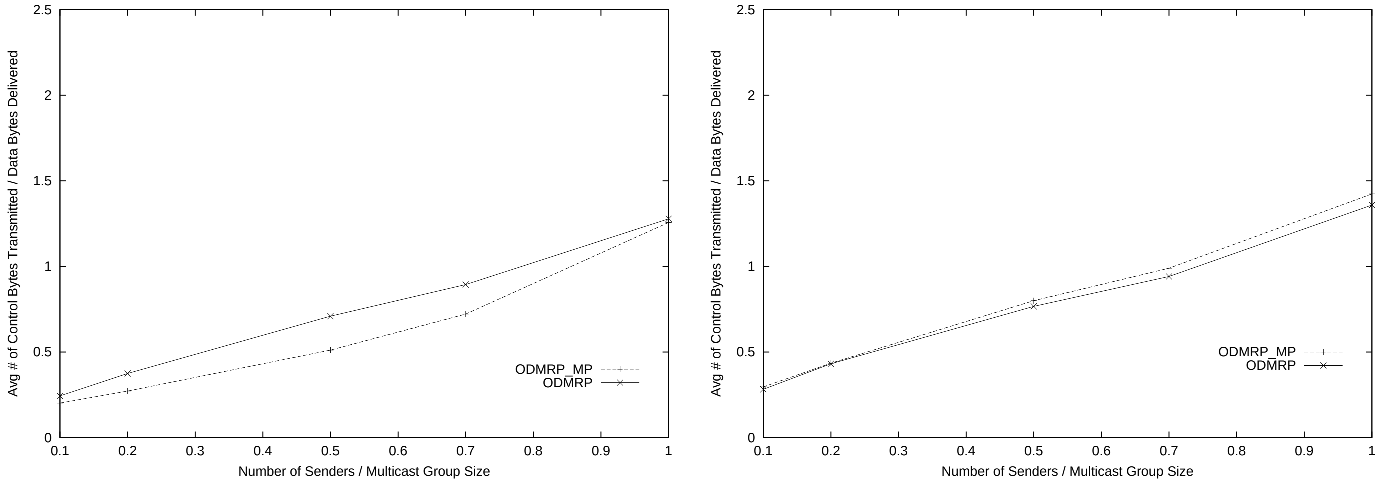


Fig. 21. Average number of control bytes transmitted per data byte delivered as a function of α for (a) low mobility case, (b) high mobility case.

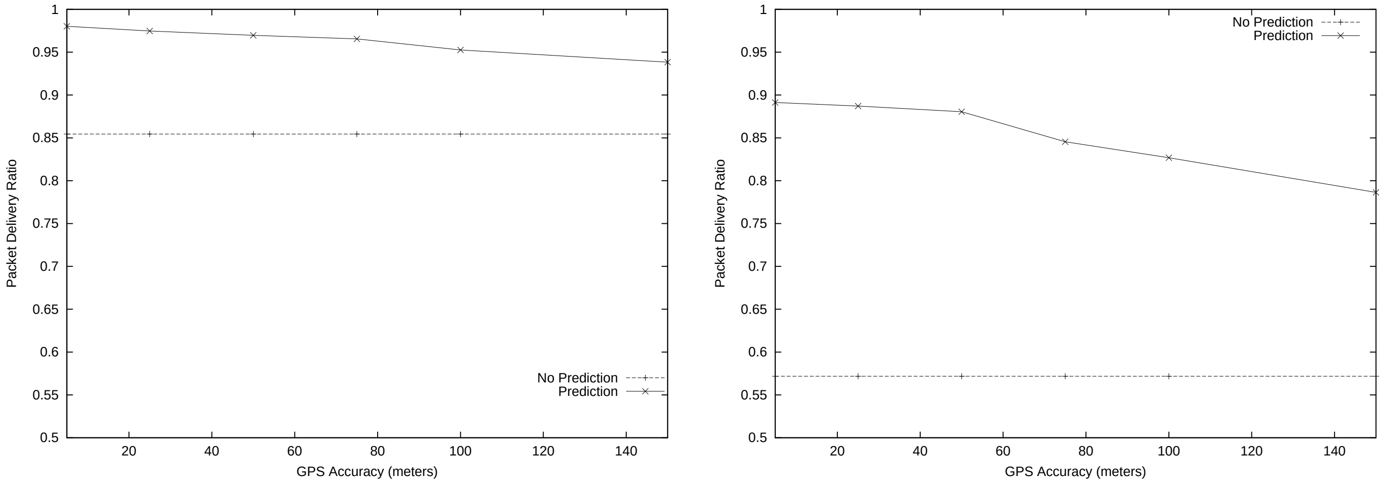


Fig. 22. Packet delivery ratio as a function of GPS accuracy for (a) low mobility case, (b) high mobility case.

C. Scenario 3 - Effect of Prediction Error

C.1 Effect of GPS Accuracy on Performance

Figure 22 shows the packet delivery ratio under different GPS accuracy conditions for the low and high mobility case. Recall that we are comparing the Distance Vector protocol and Distance Vector with Mobility Prediction. The delivery ratio for using prediction degrades when the GPS deviation increases. Moreover, at high mobility the delivery ratio for using prediction degrades at a faster rate as GPS deviation increases. Although using prediction will cause performance to degrade, we can see from the result that there is still significant performance improvement for using prediction, especially under high mobility condition ($\approx$ 20 % when using an accuracy of ± 150m).

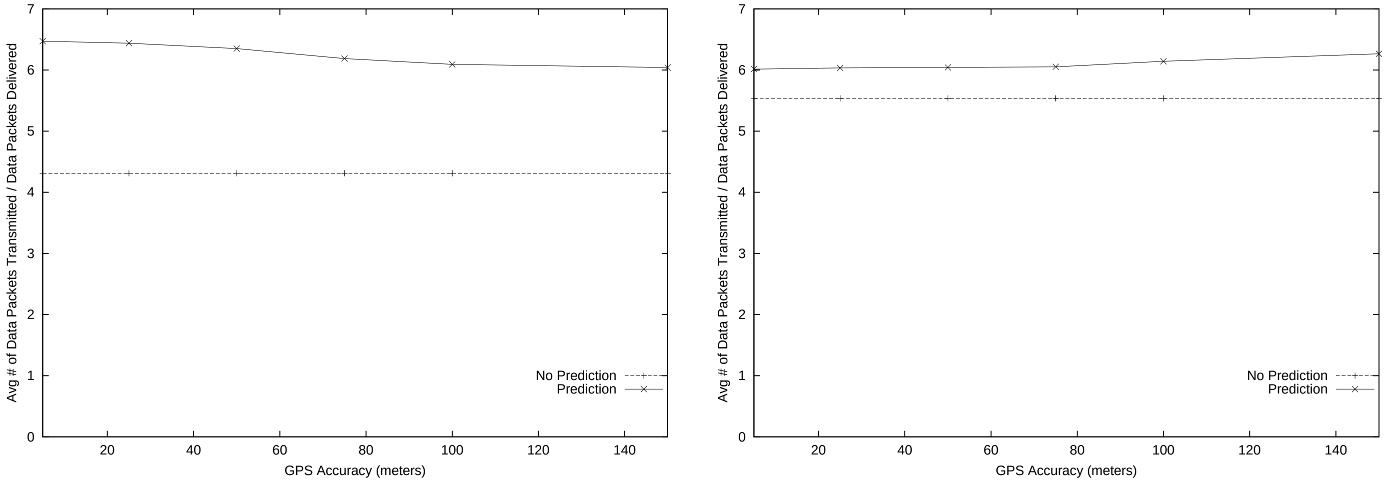


Fig. 23. Average number of data packets transmitted per data packet delivered as a function of GPS accuracy for (a) low mobility case, (b) high mobility case.

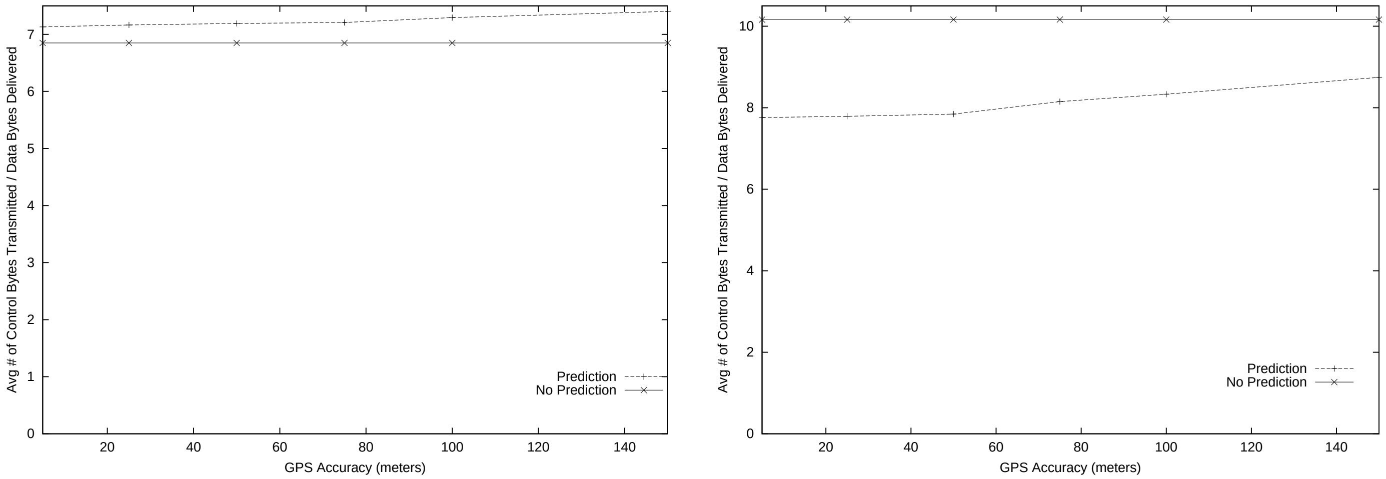


Fig. 24. Average number of control bytes transmitted per data byte delivered as a function of GPS accuracy for (a) low mobility case, (b) high mobility case.

Figure 23 shows the number of data packets transmitted per data packets delivered as a function of GPS accuracy. The numbers for the prediction method are higher than the no prediction approach. From Figure 23(b) the ratio for prediction increases when the accuracy range is increased at high mobility. This is due to the fact that more packets are being mislead toward their destination when the GPS deviation is increased, which led to a larger hop length for the packets. The distance vector protocol with prediction uses the route disconnection time as the metric for route selection. At low mobility, since the nodes are relatively stationary, increasing the deviation range randomly leads to shorter hop length in some cases. This is consistent with the results shown in Figure 23(a) which shows that the ratios for the prediction scheme decrease as accuracy becomes worse.

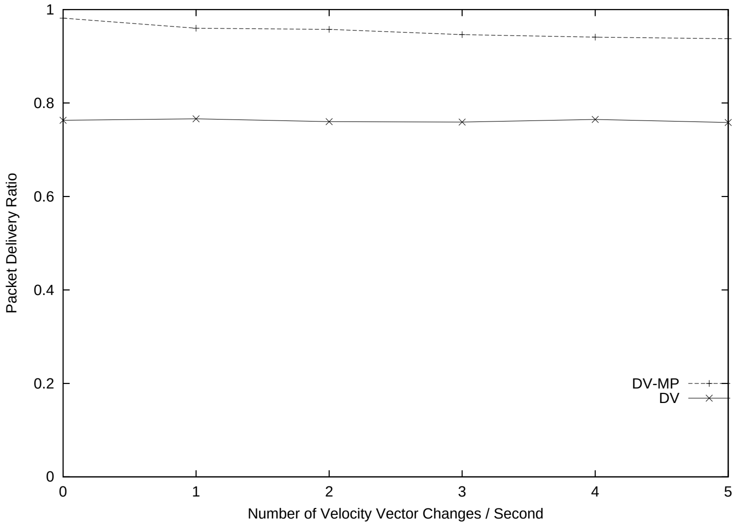


Fig. 25. Packet delivery ratio as a function of direction changing frequency.

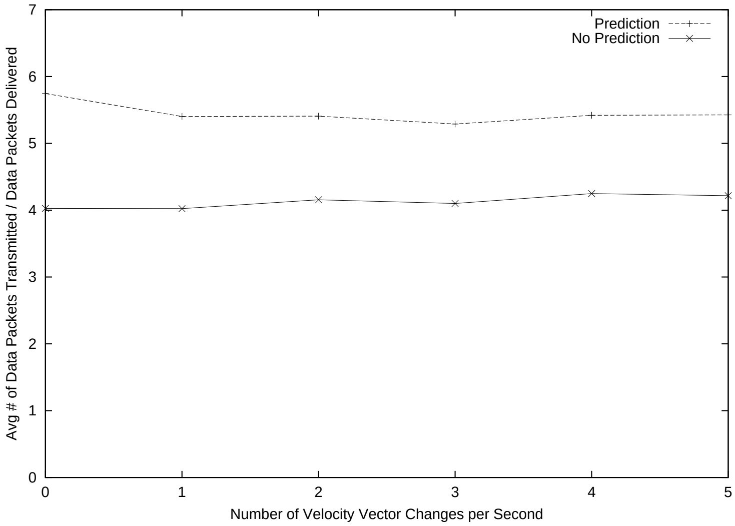


Fig. 26. Average number of data packets transmitted per data packet delivered as a function of direction changing frequency.

In Figure 24, the result for number of control bytes transmitted per data byte delivered as a function of GPS accuracy is shown. At low mobility, the prediction method has higher ratios than the no prediction method and the numbers for prediction method go up as accuracy becomes worse. At high mobility, the prediction method also shows the increase as the accuracy is decreased, but its values are significantly lower than those of no prediction method. This is consistent with the results from Figure 22 since with no prediction, the packet delivery ratio is very poor at high mobility.

C.2 Effect of Changing Direction on Performance

The results as a function of rate of change in direction are shown from Figures 25 to 28. We can see from Figure 25 that the packet delivery ratio for the no prediction case is not affected by the change rate and it maintains a steady ratio at above 0.76. The delivery ratio for prediction mechanism is as high as 0.98 when

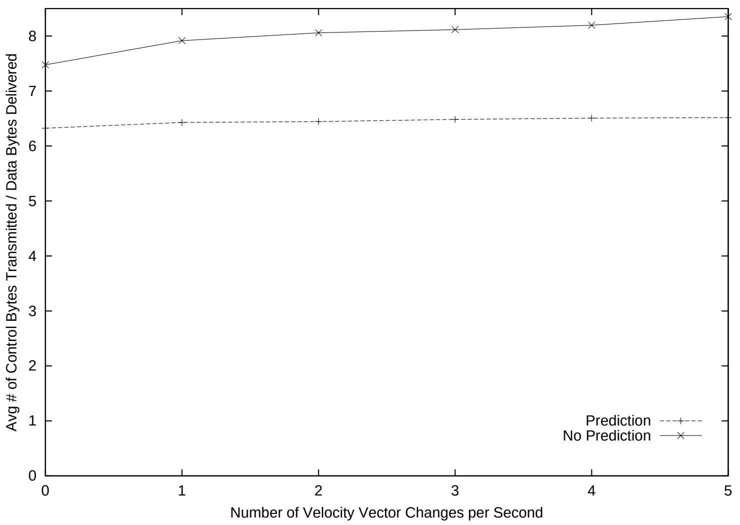


Fig. 27. Average number of control bytes transmitted per data byte delivered as a function of direction changing frequency.

there is no moving direction changes and it drops down to 0.95 when the change rate is increased up to 5 per second. When direction changing frequency increases, predictions become less accurate and cause the delivery ratio to drop.

The plot for number of data packets transmitted per data packet delivered shows a similar trend for the prediction and non-prediction case. The numbers for the prediction case are higher than the no prediction case because with prediction, data packets travel over longer and more stable routes than the no prediction mechanism.

The plot of the average number of control bytes transmitted per data bytes delivered shows that overhead increases as turning frequency increases for the prediction case while the values for the no prediction case maintain relatively stable. This result is consistent with Figure 25 since as turning frequency increases, more packets are misdirected and subsequently dropped. The no prediction case has significantly higher value than the prediction case due to its low packet delivery ratio.

The number of total packets transmitted per data packet delivered for the prediction scheme remains more stable and lower than the no prediction case. When prediction is used, more data are transmitted and most of the data are delivered. Therefore, the prediction scheme is more channel efficient than the no prediction case.

C.3 Effect of Waypoint Distance on Performance

The results for the two schemes (Distance Vector and Distance Vector with Mobility Prediction) as a function of waypoint distance are presented in this section. Remember that as waypoint distance decreases, the mobility pattern of the nodes approaches a Brownian motion. Figure 29 shows the packet delivery ratio as a function of waypoint distance. When no prediction is used, the delivery ratio remains at around 0.85 regardless of the waypoint distance. Delivery ratio is more sensitive to waypoint distance when prediction is used, since more data packets are delivered when waypoint distance is increased. As waypoint distance decreases,

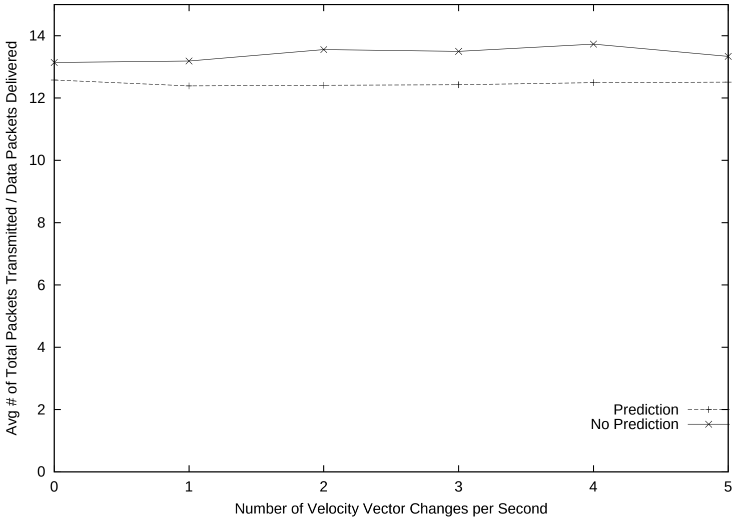


Fig. 28. Average number of total packets transmitted per data packet delivered as a function of direction changing frequency.

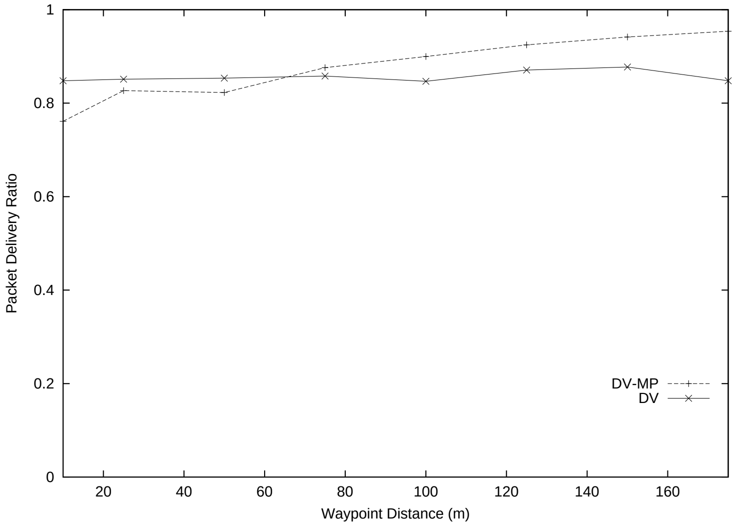


Fig. 29. Packet delivery ratio as a function of waypoint distance.

the chance of data packets being dropped increases because of the incorrect routing table information being disseminated when a node changes its direction. Still, 78% of packets are being delivered even when the waypoint distance is as low as 10 meters. This is only 5% less than the no prediction case. At such low waypoint distance, a mobile node reaches a waypoint every 2 seconds on average. The delivery ratio for prediction case becomes significantly larger than the no prediction case when waypoint distance is farther than 70 meters. Fortunately, in a real life scenario (i.e., battlefield, search and rescue operation, etc.), mobiles will generally maintain trajectory for at least hundreds of meters. Therefore, we can expect the prediction method to improve data delivery performance. Nevertheless, these results indicate the need to explore an adaptive mobility prediction scheme (for the Distance Vector routing strategy) where mobility prediction is disabled if average waypoint distance drops below a given threshold. This decision can be carried out by each node in a distributed fashion. We plan to investigate such hybrid scheme in the future.

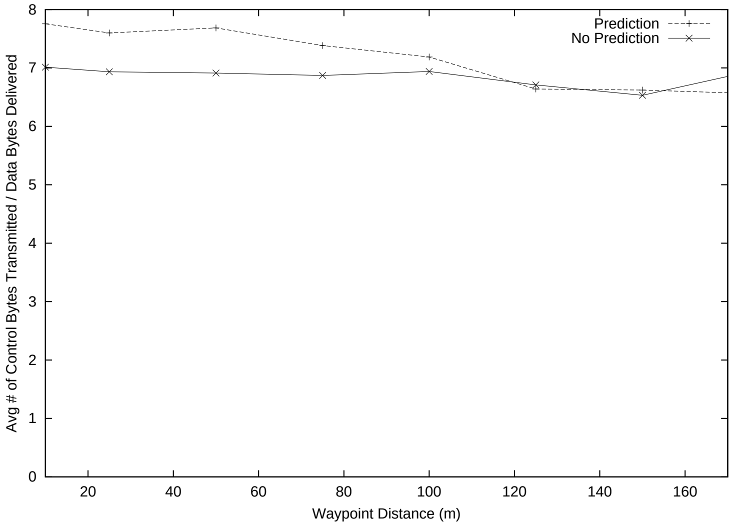


Fig. 30. Average number of control bytes transmitted per data byte delivered as a function of waypoint distance.

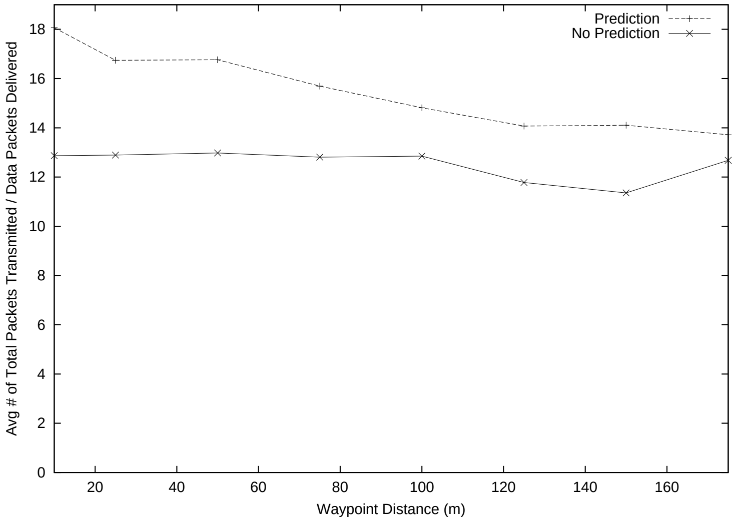


Fig. 31. Average number of total packets transmitted per data packet delivered as a function of waypoint distance.

The results for average number of control bytes transmitted per data bytes delivered as a function of the waypoint distance is presented in Figure 30. The values for the no prediction scheme are not affected by waypoint distance. The values for the prediction case are greater than the no prediction case when the waypoint distance is small. However, the difference becomes less as the waypoint distance is increased. This result is expected since increasing waypoint distance also increases the prediction accuracy, thus control bytes are used more efficiently.

Figure 31 shows the number of total packets transmitted per data packets delivered as a function of waypoint distance. The values for no prediction case are relatively constant, while those of the prediction case increase as waypoint distance decreases. As waypoint distance decreases, mobility pattern becomes more random and more data packets are being dropped. Moreover, since the prediction scheme sends data over longer and more stable routes, the number of data transmitted for the prediction scheme is larger than the no prediction case.

V. Conclusions

Effective delivery of data packets while minimizing connection disruption is crucial in ad hoc networks. In this paper we examined the use of mobility prediction to anticipate topology changes and perform rerouting prior to route breaks. Mobility prediction mechanism was applied to some of the most popular representatives of the wireless ad hoc routing family, namely an on-demand unicast routing protocol, a distance vector routing protocol, and a multicast routing protocol. Simulation results indicate that with mobility prediction enhancements, more data packets were delivered to destinations while the control packets were utilized more efficiently. Routes that are the most stable (i.e., do not become invalid due to node movements) and stay connected longest are chosen by utilizing the mobility prediction. These results are very encouraging and open the way to further research in several directions, as described in the paper.

References

[1] P. Agrawal, D.K. Anvekar, and B. Narendran, “Optimal Prioritization of Handovers in Mobile Cellular Networks,” In *Proceedings of IEEE PIMRC’94*, The Hague, Netherlands, Sep. 1994, pp. 1393-1398.

[2] R. Bagrodia, R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, and H.Y. Song, “PARSEC: A Parallel Simulation Environment for Complex Systems,” *IEEE Computer*, vol. 31, no. 10, Oct. 1998, pp.77-85.

[3] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward, “A Distance Routing Effect Algorithm for Mobility (DREAM),” In *Proceedings of ACM/IEEE MOBICOM’98*, Dallas, TX, Oct. 1998, pp. 76-84.

[4] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade, “AMRoute: Adhoc Multicast Routing Protocol,” *Internet-Draft*, `draft-talpade-manet-amroute-00.txt`, Aug. 1998, Work in progress.

[5] C.-C. Chiang, M. Gerla, and L. Zhang, “Forwarding Group Multicast Protocol (FGMP) for Multihop, Mobile Wireless Networks,” *Baltzer Cluster Computing*, vol. 1, no. 2, 1998, pp. 187-196.

[6] M.S. Corson and A. Ephremides, “A distributed routing algorithm for mobile wireless networks.” *ACM-Baltzer Journal of Wireless Networks*, 1:61-81, January 1995.

[7] M.S. Corson and S.G. Batsell, “A Reservation-Based Multicast (RBM) Routing Protocol for Mobile Networks: Initial Route Construction Phase,” *ACM/Baltzer Wireless Networks*, vol. 1, no. 4, Dec. 1995, pp. 427-450.

[8] “Differential GPS”, Web site at `http://www.ursatec.com/newpage2.htm`

[9] J.J. Garcia-Luna-Aceves and E.L. Madruga, “A Multicast Routing Protocol for Ad-Hoc Networks,” In *Proceedings of IEEE INFOCOM’99*, New York, NY, Mar. 1999, pp. 784-792.

[10] Internet Engineering Task Force Mobile Ad hoc Network (MANET) Working Group. `http://www.ietf.org/html.charters/manet-charter.html`

[11] D.B. Johnson and D.A. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks”, *In Mobile Computing*, edited by T. Imielinski and H. Korth, Chapter 5, Kluwer Publishing Company, 1996, pp. 153-181.

[12] L. Ji and M.S. Corson, “A Lightweight Adaptive Multicast Algorithm,” In *Proceedings of IEEE GLOBECOM’98*, Sydney, Australia, Nov. 1998, pp. 1036-1042.

[13] J. Jubin and J.D. Tornow, “The DARPA Packet Radio Network Protocols,” *Proceedings of the IEEE*, vol. 75, no. 1, Jan. 1987, pp. 21-32.

[14] E.D. Kaplan (Editor), *Understanding the GPS: Principles and Applications*, Artech House, Boston, MA, Feb. 1996.

[15] Y.-B. Ko and N.H. Vaidya, “Location-Aided Routing (LAR) in Mobile Ad Hoc Networks,” In *Proceedings of ACM/IEEE MOBICOM’98*, Dallas, TX, Oct. 1998, pp. 66-75.

[16] S.-J. Lee, M. Gerla, and C.-C. Chiang, “On-Demand Multicast Routing Protocol,” In *Proceedings of IEEE WCNC’99*, New Orleans, LA, Sep. 1999, pp. 1298-1304.

[17] S.-J. Lee, W. Su, and M. Gerla, “On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks,” *Internet-Draft*, draft-ietf-manet-odmrp-01.txt, Jun. 1999, Work in progress.

[18] IEEE Computer Society LAN MAN Standards Committee, *Wireless LAN Medium Access Protocol (MAC) and Physical Layer (PHY) Specification*, IEEE Std 802.11-1997. The Institute of Electrical and Electronics Engineers, New York, NY, 1997.

[19] S. Murthy and J.J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks," *ACM/Baltzer Mobile Networks and Applications*, vol. 1, no. 2, Oct. 1996, pp. 183-197.

[20] D.L. Mills, "Internet Time Synchronization: the Network Time Protocol," *IEEE Transactions on Communications*, vol. 39, no. 10, Oct. 1991, pp. 1482-1493.

[21] B. Narendran, P. Agrawal, and D.K. Anvekar, " Minimizing Cellular Handover Failures Without Channel Utilization Loss," In *Proceedings of IEEE GLOBECOM'94*, San Francisco, CA, Dec. 1994, pp. 1679-1685.

[22] V.D. Park and M.S. Corson, "A Highly Adaptive distributed routing algorithm for mobile wireless networks." In *Proceedings of IEEE INFOCOM'97*, Kobe, Japan, Apr. 1997, pp. 1405-1413.

[23] M.R. Pearlman and Z.J. Haas, "Determining the Optimal Configuration for the Zone Routing Protocol," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, Aug. 1999, pp. 1395-1414.

[24] C.E. Perkins and Pravin Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", In *Proceedings of ACM SIGCOMM'94*, London, UK, Sep. 1994, pp. 234-244.

[25] C.E. Perkins and E.M. Royer, "Ad Hoc On-Demand Distance Vector Routing", In *Proceedings of IEEE WMCSA'99*, New Orleans, LA, Feb. 1999, pp. 90-100.

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

[27] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: A Core-Extraction Distributed Ad Hoc Routing Algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, Aug. 1999, pp. 1454-1465.

[28] W. Su and M. Gerla, "IPv6 Flow Handoff in Ad-Hoc Wireless Networks Using Mobility Prediction," *To appear in Proceedings of IEEE GLOBECOM'99*, Rio de Janeiro, Brazil, Dec. 1999.

[29] C.-K. Toh, "Associativity-Based Routing for Ad-Hoc Mobile Networks," *Wireless Personal Communications Journal*, Special Issue on Mobile Networking and Computing Systems, Kluwer Academic Publishers, vol. 4, no. 2, Mar. 1997, pp. 103-139.

[30] UCLA Parallel Computing Laboratory and Wireless Adaptive Mobility Laboratory, *GloMoSim: A Scalable Simulation Environment for Wireless and Wired Network Systems*, `http://pcl.cs.ucla.edu/projects/domains/glomosim.html`