DTIC DEFENSE TECHNICAL INFORMATION CENTER

Information for the Defense Community

DTIC® has determined on 11/30/2009 that this Technical Document has the Distribution Statement checked below. The current distribution for this document can be found in the DTIC® Technical Report Database.

☒ **DISTRIBUTION STATEMENT A.** Approved for public release; distribution is unlimited.

☐ **© COPYRIGHTED**; U.S. Government or Federal Rights License. All other rights and uses except those permitted by copyright law are reserved by the copyright owner.

☐ **DISTRIBUTION STATEMENT B.** Distribution authorized to U.S. Government agencies only (fill in reason) (date of determination). Other requests for this document shall be referred to (insert controlling DoD office)

☐ **DISTRIBUTION STATEMENT C.** Distribution authorized to U.S. Government Agencies and their contractors (fill in reason) (date of determination). Other requests for this document shall be referred to (insert controlling DoD office)

☐ **DISTRIBUTION STATEMENT D.** Distribution authorized to the Department of Defense and U.S. DoD contractors only (fill in reason) (date of determination). Other requests shall be referred to (insert controlling DoD office).

☐ **DISTRIBUTION STATEMENT E.** Distribution authorized to DoD Components only (fill in reason) (date of determination). Other requests shall be referred to (insert controlling DoD office).

☐ **DISTRIBUTION STATEMENT F.** Further dissemination only as directed by (inserting controlling DoD office) (date of determination) or higher DoD authority.

Distribution Statement F is also used when a document does not contain a distribution statement and no distribution statement can be determined.

☐ **DISTRIBUTION STATEMENT X.** Distribution authorized to U.S. Government Agencies and private individuals or enterprises eligible to obtain export-controlled technical data in accordance with DoDD 5230.25; (date of determination). DoD Controlling Office is (insert controlling DoD office).

FINAL PROJECT REPORT

SEPTEMBER 23, 2008 TO SEPTEMBER 22, 2009

Agreement No: W911NF-08-2-0061

Routing Protocols to Minimize the Number of Route Disconnections for Communication in Mobile Ad Hoc Networks

Recipient Program Manager
(Principal Investigator)

Dr. Natarajan Meghanathan
Assistant Professor
Department of Computer Science
Jackson State University
Jackson, MS 39217
Email: natarajan.meghanathan@jsums.edu
Phone: 601-979-3661

ARL Cooperative Agreement Manager

Dr. Raju R. Namburu
US Army Research Laboratory
ATTN: AMSRD-ARL-CI
2800 Powder Mill Road
Adelphi, MD 20783-1197
Email: raju@arl.army.mil
Phone: 301-394-2100

TABLE OF CONTENTS

TECHNICAL STATUS REPORT

I. Project Team and Project Steering Committee

Project Team

Role	Name	Affiliation
Principal Investigator (PI)	Dr. Natarajan Meghanathan	Assistant Professor, Computer Science
Key Research Personnel	Dr. Kamal Ali	Professor, Computer Engineering
	Dr. Abdelnasser A. Eldek	Assistant Professor, Computer Engineering
	Dr. Ali Abu-El Humos	Assistant Professor, Computer Science
Project Consultant	Dr. Loretta A. Moore	Department Chair, Computer Science

Project Steering Committee

Name	Affiliation
Dr. Robert Whalin	Associate Dean, School of Engineering
Dr. Mahmoud Manzoul	Department Chair, Computer Engineering
Dr. Shahrouz Aliabadi	Professor, Computer Engineering
Dr. Tarek El-Bawab	Associate Professor, Computer Engineering
Dr. Gordon W. Skelton	Associate Professor, Computer Engineering

II. Outline of the Constituent Tasks of the Research Activities and their Current Status

Research Activity 1: An Energy-Efficient Density and Mobility Aware Route Discovery Strategy to Minimize the Number of Route Discoveries in Mobile Ad hoc Networks

Research Personnel: Dr. Natarajan Meghanathan, PI

Task No.	Task	Current Status
1	Study the related work on different broadcast route discovery strategies	Completed
2	Build a density and mobility aware model for the broadcast transmission range	Completed
3	Develop an algorithm for automatic dynamic selection of DMEF parameters	Completed
4	Conduct simulations of Dynamic Source Routing (DSR) protocol and the Location Prediction Based Routing (LPBR) protocol using flooding and DMEF	Completed
5	Analyze the simulation results with respect to different performance metrics	Completed

Research Activity 2: A Multicast Version of the Location Prediction Based Routing Protocol (MLPBR)

Research Personnel: Dr. Natarajan Meghanathan, PI

Task No.	Task	Current Status
1	Study the related work on multicast routing protocols for mobile ad hoc networks (MANETs)	Completed
2	Develop the multicast extensions to LPBR (NR-MLPBR and R-MLPBR)	Completed
3	Conduct simulations of MLPBR and compare its performance with some of the currently existing MANET multicast routing protocols	Completed
4	Analyze the simulation results with respect to different performance metrics	Completed

Research Activity 3: A Node-disjoint Multi-path Version of the Location Prediction Based Routing Protocol (LPBR-M)

Research Personnel: Dr. Natarajan Meghanathan, PI

Task No.	Task	Current Status
1	Study the related work on multi-path routing protocols for mobile ad hoc networks (MANETs)	Completed
2	Develop the algorithm for the node-disjoint multi-path version of LPBR (LPBR-M)	Completed
3	Conduct simulations of LPBR-M and compare its performance with some of the currently existing MANET multi-path routing protocols	Completed
4	Analyze the simulation results with respect to different performance metrics	Completed

Research Activity 4: Design of a Highly-Directional Antenna for Wireless Networks

Research Personnel: Dr. Kamal Ali and Dr. Abdelnasser Eldek

Task No.	Task	Current Status
1	Hiring the students to work on the tasks.	Completed
2	Training the students on self-organizing maps and Antenna modeling software	Completed
3	Algorithm development and Antenna geometry suggestion and modification	Completed
4	Simulations	Completed
5	Results' analysis	Completed
6	Final results	Completed

Research Activity 5: Medium Access Control (MAC) Layer Design for a Wireless Sensor Network (WSN) Simulator

Research Personnel: Dr. Ali Abu-El Humos

Task No.	Task	Current Status
1	Literature review and problem definition	Completed
2	Simulate a WSN in NS2 using its current energy model	Completed
3	Simulate a WSN in NS2 using the modified energy model	Completed
4	Results, analysis and final report	Completed

III. Listing of Publications and Articles under Review/Revision

Peer-reviewed Journal Publications

[J1] **N. Meghanathan**, "Multicast Extensions to the Location Prediction Based Routing Protocol for Mobile Ad hoc Networks," *ISAST Transactions on Computers and Intelligent Systems*, Vol. 1, No. 1, pp. 56 – 65, August 2009.

[J2] **N. Meghanathan**, "A Density and Mobility Aware Energy-Efficient Broadcast Route Discovery Strategy for Mobile Ad hoc Networks," accepted for publication in the *International Journal of Computer Science and Network Security*, Vol. 9, No. 11, November 2009.

Peer-reviewed Conference Publications/ Proceedings

[C1] **N. Meghanathan,** "A Density and Mobility Aware Energy-Efficient Broadcast Strategy to Minimize the Number of Route Discoveries in Mobile Ad hoc Networks," *Proceedings of the 2009 International Conference on Wireless Networks, ICWN 09,* pp. 167 – 173, Las Vegas, July 13-16, 2009.

[C2] **N. Meghanathan**, "Multicast Extensions to the Location-Prediction Based Routing Protocol for Mobile Ad hoc Networks," *International Conference on Wireless Algorithms, Systems and Applications*, Boston, USA, August 16-18, 2009, published in the *Springer Verlag Lecture Notes of Computer Science Series*, LNCS 5682, B. Liu et al. (Eds.), pp. 190-199, August 2009.

[C3] **N. Meghanathan**, "A Node-Disjoint Multi-path Extension of the Location Prediction Based Routing Protocol for Mobile Ad hoc Networks," accepted for publication in the *International Conference on Signal Processing and Communication Systems,* Omaha, Nebraska, USA, September 28-30, 2009.

Research Activity – 1

An Energy-Efficient Density and Mobility Aware Broadcast Strategy to Minimize the Number of Route Discoveries in Mobile Ad hoc Networks

Dr. Natarajan Meghanathan
Assistant Professor
Department of Computer Science
Jackson State University
Jackson, MS 39217
Email: natarajan.meghanathan@jsums.edu
Phone: 601-979-3661

I. Breakdown of the Research Activity to Tasks

Task No.	Task	Current Status
1	Study the related work on different broadcast route discovery strategies	Completed
2	Build a density and mobility aware model for the broadcast transmission range	Completed
3	Develop an algorithm for automatic dynamic selection of DMEF parameters	Completed
4	Conduct simulations of Dynamic Source Routing (DSR) protocol and the Location Prediction Based Routing (LPBR) protocol using flooding and DMEF	Completed
5	Analyze the simulation results with respect to different performance metrics	Completed

II. Description of the Tasks

Task 1: Study the Related Work on Different Broadcast Route Discovery Strategies

We surveyed the literature for different broadcast route discovery strategies that have been proposed to reduce the route discovery overhead and we describe below the strategies relevant to the research conducted. In Section 5.3, we qualitatively analyze the advantages of our DMEF broadcast strategy compared to the broadcast strategies described below in Sections 1.1 and 1.2.

1.1 Reliable Route Selection (RRS) Algorithm

In [1], the authors proposed a Reliable Route Selection (referred to as RRS) algorithm based on Global Positioning System (GPS) [2]. The RRS algorithm divides the circular area formed by the transmission range of a node into two zones: stable zone and caution zone. A node is said to maintain stable links with the neighbor nodes lying in its stable zone and maintain unstable links with the neighbor nodes lying in its caution zone. If R is the transmission range of a node, then the radius of the stable zone is defined as $r = R-\delta S$ where S is the speed of the node. The status zone is a circular region (with its own center) inscribed inside the circular region formed by the transmission range of the node. The center of the status zone need not be the center of the circular region forming the transmission range of the node, but always lies in the direction of movement of the node.

RRS works as follows: The Route-Request (RREQ) message of a broadcast route discovery process includes the co-ordinates representing the current position of the transmitter of the RREQ message, the co-ordinates representing the center of the stable zone of the transmitter, the value of parameter δ to be used by an intermediate node and the stable zone radius of the transmitter of the message. The source node of the route discovery process broadcasts the RREQ message in the complete neighborhood formed by the transmission range R. The RRS-related fields are set to initial values corresponding to the source node. An intermediate node receiving the RREQ message broadcasts the message further, only if the node lies in the stable zone of the transmitter. If a route discovery attempt based on a set value of δ is unsuccessful, the source node decrements the value of δ and launches another global broadcast based route discovery. This process is continued (i.e., the value of δ decremented and global broadcast reinitiated) until the source finds a path to the destination. If the source cannot find a route to the destination even while conducting route discovery with δ set to zero, then the source declares that the destination is not connected to it.

1.2 Efficient Broadcast Route Discovery Strategies

In [3], the authors propose several broadcast route discovery strategies that could reduce the number of retransmitting nodes of a broadcast message. These strategies can be grouped into four families: probability-based, counter-based, area-based and neighbor-knowledge based methods:

(i) Probability-based method: When a node receives a broadcast message for the first time, the node rebroadcasts the message with a certain probability. If the message received is already seen, then the node drops the message irrespective of whether or not the node retransmitted the message when it received the first time.

(ii) Counter-based method: When a node receives a broadcast message for the first time, it waits for a certain time before retransmitting the message. During this broadcast-wait-time, the node maintains a counter to keep track of the number of redundant broadcast messages received from some of its other neighbors. If this counter value exceeds a threshold within the broadcast-wait-time, then the node decides to drop the message. Otherwise, the node retransmits the message.

(iii) Area-based method: A broadcasting node includes its location information in the message header. The receiver node calculates the additional coverage area that would be obtained if the message were to be rebroadcast. If the additional coverage area is less than a threshold value, all future receptions of the same message will be dropped. Otherwise, the node starts a broadcast-wait-timer. Redundant broadcast messages received during this broadcast-wait-time are also cached. After the timer expires, the node considers all the cached messages and recalculates the additional coverage area if it were to rebroadcast the particular message. If the additional obtainable coverage area is less than a threshold value, the cached messages are dropped. Otherwise, the message is rebroadcast.

(iv) Neighbor-knowledge based method: This method requires nodes to maintain a list of 1-hop neighbors and 2-hop neighbors, learnt via periodic beacon exchange. Using these lists, a node calculates the set (of the smallest possible size) of 1-hop neighbors required to reach all the 2-hop neighbors. The minimum set of 1-hop neighbors that will cover all of the 2-hop neighbors is called the Multi Point Relays (MPRs).

Task 2: Build a Density and Mobility Aware Model for the Broadcast Transmission Range

We design and develop a novel distance and mobility aware energy-efficient route discovery strategy (DMEF) that attempts to reduce the energy consumed due to broadcast route discoveries by letting a node to broadcast only within a limited neighborhood. The size of the neighborhood to which a node should advertise itself as part of the route discovery process is decided based on the number of neighbors surrounding the node and velocity of the node. The neighborhood size for rebroadcast is reduced in such a way that the RREQ packets still make it to the destination through one or more relatively long-living paths. Note that, throughout this report, the terms 'path' and 'route' are used interchangeably. They mean the same.

2.1 Terminology and Assumptions

Every node (say node u) in the network is configured with a maximum transmission range ($Range_u^{Max}$). If the distance between two nodes is less than or equal to the maximum transmission range, then the two nodes are said to be within the "complete neighborhood" of each other. Each node broadcasts periodically a beacon message in its complete neighborhood. The time between two successive broadcasts is chosen uniformly, randomly, by each node from within the range $[0 \ldots T_{wait}]$. Using this strategy, each node learns about the number of nodes in its complete neighborhood.

2.2 Basic Idea of DMEF

The twin objectives of DMEF are to increase the time between successive global broadcast route discoveries and to reduce the energy consumed during the broadcast route discoveries vis-à-vis flooding. DMEF achieves this by taking into consideration the number of neighbors of a node (a measure of node density) and node velocity. The basic idea behind DMEF is as follows: The transmission range of a RREQ broadcast for route discovery is not fixed for every node. A node that is surrounded by more neighbors in the complete neighborhood should broadcast the RREQ message only within a smaller neighborhood that would be sufficient enough to pick up the message and forward it to the other nodes in the rest of the network. On the other hand, a node that is surrounded by fewer neighbors in the complete neighborhood should broadcast the RREQ message to a larger neighborhood (but still contained within the complete neighborhood) so that a majority of the nodes in the complete neighborhood can pick up the message and rebroadcast it further. A node rebroadcasts a RREQ message at most once. The density aspect of DMEF thus helps to reduce the unnecessary transmission and reception of broadcast RREQ messages and conserves energy.

To discover stable routes that exist for a longer time, DMEF takes the following approach: A node that is highly mobile makes itself available only to a smaller neighborhood around itself, whereas a node that is less mobile makes itself available over a larger neighborhood (but still contained within the complete neighborhood). The reasoning is that links involving a slow moving node will exist for a longer time. Hence, it is better for a slow moving node to advertise itself to a larger neighborhood so that the links (involving this node) that are part of the routes discovered will exist for a longer time. On the other hand, a fast moving node will have links of relatively longer lifetime with neighbors that are closer to it. Hence, it is worth to let a fast moving node advertise only to its nearby neighbors.

2.3 DMEF Mathematical Model

DMEF effectively uses the knowledge of node density and mobility so that they complement each other in discovering stable routes in a more energy-efficient fashion. The transmission range used by a node u, $Range_u^{RREQ}$, to rebroadcast a RREQ message is given by the following model:

$$Range_u^{RREQ} = Range_u^{Max} - \left[\left(\frac{|Neighbors_u|}{\alpha} \right) * v_u^{\beta} \right] \quad \ldots\ldots\ldots\ldots (1)$$

In order to make sure, $Range_u^{RREQ}$ is always greater than or equal to zero, the value of parameter α should be chosen very carefully. For a given value of parameter β, the necessary condition is:

$$\alpha \geq \left[\left(\frac{|Neighbors_u|}{Range_u^{Max}}\right) * v_u^{\beta}\right] \quad \text{.. (2)}$$

In practice, the value of parameter α has to be sufficiently larger than the value obtained from equality (2), so that the RREQ message reaches neighbors who can forward the message further to the rest of the network. Otherwise, certain source-destination nodes may not be reachable from each other, even though there may exist one or more paths between them in the underlying network.

Task 3: Develop an Algorithm for Automatic Dynamic Selection of DMEF Parameters

We now describe the algorithm that allows for each node to dynamically choose at run-time the appropriate values for the critical operating parameters α and β depending on the perceived number of nodes in the complete neighborhood of the node and the node's own velocity. A node has to be simply pre-programmed with the appropriate values of α and β to be chosen for different range of values of the number of nodes in the complete neighborhood and node velocity.

Let *maxNeighbors_lowDensity*, *maxNeighbors_moderateDensity* represent the maximum number of neighbors a node should have in order to conclude that the complete neighborhood density of the node is low and moderate respectively. If a node has more than *maxNeighbors_moderateDensity* number of neighbors, then the node is said to exist in a complete neighborhood of high density. Let *lowDensity_α*, *moderateDensity_α* and *highDensity_α* represent the values of α to be chosen by a node for complete neighborhoods of low, moderate and high density respectively. Let *maxVel_lowMobility*, *maxVel_moderateMobility* represent the maximum velocity values for a node in order to conclude that the mobility of the node is low and moderate respectively. If the velocity of a node is more than *maxVel_moderateMobility*, then the mobility of the node is said to be high. Let *lowMobility_β*, *moderateMobility_β* and *highMobility_β* represent the values of β to be chosen by a node when its mobility is low, moderate and high respectively. Let v_u^t represent velocity of a node *u* at time *t* and let $Neighbors_u^t$ represent the set of neighbors in the complete neighborhood determined by node *u* based on the latest periodic beacon exchange in the complete neighborhood formed by the maximum transmission range, $Range_u^{Max}$. The algorithm to dynamically choose the values of parameters α and β (represented as α_u^t and β_u^t) for a node *u* is illustrated below:

```
Input: $Neighbors_u^t$ and $v_u^t$
Output: $\alpha_u^t$ and $\beta_u^t$
Begin DMEF_Parameter_Selection
    if ($v_u^t \leq maxVel\_lowMobility$) $\beta_u^t \leftarrow lowMobility\_\beta$
    else if ($v_u^t \leq maxVel\_moderateMobility$) $\beta_u^t \leftarrow moderateMobility\_\beta$
    else $\beta_u^t \leftarrow highMobility\_\beta$
    $minimum\_\alpha_u^t \leftarrow \left[\left(\frac{|Neighbors_u^t|}{Range_u^{Max}}\right) * \left(v_u^t\right)^{\beta_u^t}\right]$
    if ($|Neighbors_u^t| \leq maxNeighbors\_lowDensity$) $\alpha_u^t \leftarrow$ Maximum ($minimum\_\alpha_u^t$, $lowDensity\_\alpha$)
```

```
    else if (| $Neighbors_u^t$ | ≤ maxNeighbors_moderateDensity)
        $\alpha_u^t$ ← Maximum (minimum_$\alpha_u^t$, moderateDensity_α)
    else $\alpha_u^t$ ← Maximum (minimum_$\alpha_u^t$, highDensity_α)
    return $\alpha_u^t$ and $\beta_u^t$
End DMEF_Parameter_Selection
```

Figure 1: Algorithm to Dynamically Select the Parameter Values for DMEF

After selecting the appropriate values for parameters α and β at time t, a node can determine the transmission range to be used for the broadcast of the RREQ message using equation (1). Note that the number of neighbors in the complete neighborhood and the node velocity can be different for each node at a given time instant and can be different for even a particular node at different time instants. DMEF adapts itself to these dynamically changing conditions of neighborhood size and node velocity.

Task 4: Conduct Simulations of Dynamic Source Routing (DSR) Protocol and the Location Prediction Based Routing (LPBR) Protocol using Flooding and DMEF

The effectiveness of the DMEF strategy has been studied through simulations. We use the well-known minimum-hop based Dynamic Source Routing (DSR) protocol [4] and the recently proposed Location-Prediction Based Routing (LPBR) protocol [5] to reduce the number of global broadcast route discoveries, as the routing protocols that use DMEF as their route discovery strategy. The benchmark used for DMEF evaluation is the performance of DSR and LPBR with flooding as the route discovery strategy. The simulation models used and the values for the simulation parameters are listed in Table 1. The simulations were conducted using a MANET discrete-event simulation software developed by the PI in Java. The simulations were run in a Laptop (Dell Inspiron 6000, 1.5 GHz processor speed, 1 GB RAM and 70 GB Hard disk space).

Table 1: Simulation Models and Simulation Parameters

Network Dimensions	1000m x 1000m	
Number of Nodes	25 (low density), 50 (moderate density) and 75 (high density)	
Maximum Transmission Range	250m	
Mobility Model	Random Waypoint model [6]	v_{min} = 0 m/s; v_{max} = 10 m/s (low mobility); 30 m/s (moderate mobility) and 50 m/s (high mobility)
Traffic model	Constant Bit Rate (CBR) Traffic	15 source-destination sessions; 4 Data packets per second; 512 bytes per data packet
Energy Consumption Model	Transmission Energy	1.4 W [7]
	Reception Energy	1 W [7]
Network Bandwidth	2 Mbps	
MAC Layer Model	IEEE 802.11 [8]	
Parameter T_{wait} (for DMEF)	10 seconds	
Simulation Time	1000 seconds	

Task 5: Analyze the Simulation Results with respect to Different Performance Metrics

5.1 Performance Metrics

The performance metrics studied are as follows:

- *Total Energy Lost per Route Discovery*: This is the average of the total energy consumed for the global broadcast based route discovery attempts. This includes the sum of the energy consumed to transmit (broadcast) a RREQ packet to all the nodes in the neighborhood and to receive the RREQ packet sent by each node in the neighborhood, summed over all the nodes.
- *Percentage of Total Energy Spent for Route Discovery*: This is the ratio of the total energy spent for route discovery to the sum of the energy spent across all the nodes in the network.
- *Hop Count per Path*: This is the average hop count per path, time-averaged over all the *s-d* sessions. For example, if we have been using two paths P1 of hop count 3 and P2 of hop count 5 for 10 and 20 seconds respectively, then the time-averaged hop count of P1 and P2 is (3*10+5*20)/30 = 4.33.
- *Time between Successive Route Discoveries*: This is the average of the time between two successive global broadcast based route discovery attempts. Larger the time between two successive route discoveries, lower will be the control overhead.
- *Packet Delivery Ratio*: This is the ratio of the data packets delivered to the destination to the data packets originated at the source, computed over all the *s-d* sessions.
- *Energy Throughput*: This is the average of the ratio of the number of data packets reaching the destination to the sum of the energy spent across all the nodes in the network.

5.2 Analysis of Simulation Results

We now analyze the simulation results obtained for each of the above performance metrics under different conditions of network density and node mobility.

5.2.1 *Total Energy Spent Route Discovery*

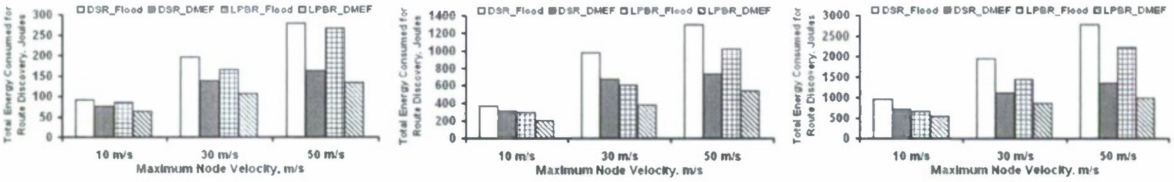


Figure 2.1: 25 Nodes **Figure 2.2:** 50 Nodes **Figure 2.3:** 75 Nodes

Figure 2: Total Energy Consumed for Route Discovery

Performance results in figures 2.1 through 2.3 illustrate that the DMEF strategy achieves its purpose of reducing the energy spent in the network due to global broadcast route discoveries. The reduction in the energy spent for route discoveries is evident in the case of both DSR and LPBR protocols. The reduction in the energy spent for route discoveries is also more evident as we increase the network density and/or node mobility. This illustrates the effectiveness of DMEF because the strategy aims to minimize the unnecessary rebroadcasts in a network especially when the network density is high. In high-density networks, it is enough to rebroadcast through a reduced set of nodes to find a set of paths between a source and destination rather than broadcasting through all the nodes in the network. Compared to DSR, LPBR incurs relatively lower number of global broadcast based route discoveries. In addition, DMEF helps the protocol to reduce the energy spent per broadcast based route discovery. Aided by both these factors, LPBR incurs a significantly lower energy due to route discoveries compared to DSR.

5.2.2 *Percentage of Total Energy Spent for Route Discovery*

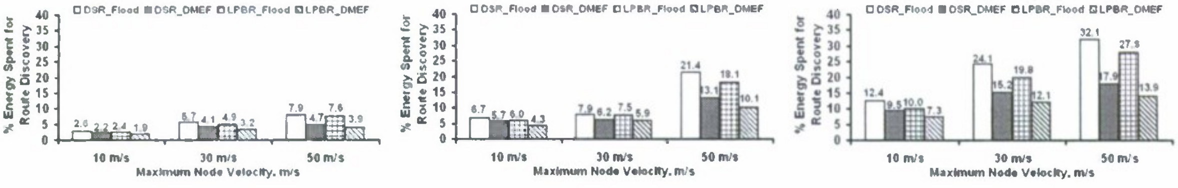


Figure 3.1: 25 Nodes **Figure 3.2:** 50 Nodes **Figure 3.3:** 75 Nodes

Figure 3: Percentage of Total Energy Spent for Route Discovery

As observed in Figures 3.1 through 3.3, for both DSR and LPBR, the difference in the percentage of total energy spent for route discovery using flooding and DMEF increases as we increase the network density and/or node mobility. For a given level of node mobility, the energy savings obtained with DMEF increases with increase in network density. Similarly, for a given network density, the energy savings obtained with DMEF, relative to flooding, increases with increase in the level of node mobility. For a given network density and level of node mobility, the relative reduction in the percentage of total energy spent for route discoveries due to the usage of DMEF vis-à-vis flooding is almost the same for both DSR and LPBR. This illustrates that DMEF can be used for energy-efficient route discovery by any routing protocol for mobile ad hoc networks.

5.2.3 *Average Hop Count per Path*

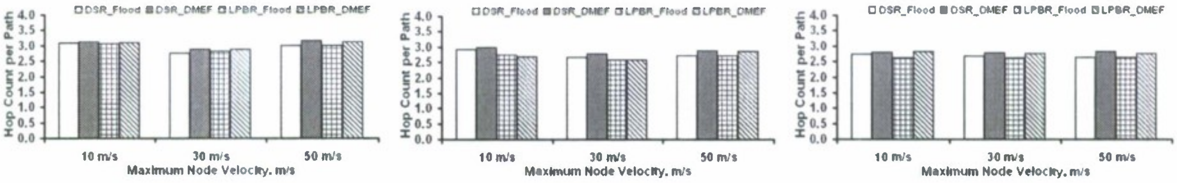


Figure 4.1: 25 Nodes **Figure 4.2:** 50 Nodes **Figure 4.3:** 75 Nodes

Figure 4: Average Hop Count per Path

DMEF prefers to determine long-living routes by primarily broadcasting the RREQ message through nodes that are relatively slow moving in the network. As a result, the routes determined for the DSR and LPBR protocols need not have hop count matching with that of the minimum hop count paths in the network. DMEF determines routes that have at most 8% larger hop count compared to the minimum hop routes, but the routes determined through DMEF exist for a relatively larger lifetime compared to the routes determined using flooding. For both DSR and LPBR, for a given node mobility in the network, as we increase the network density from low to moderate and to high, the average hop count per path decreases (by about 5%-15%).

5.2.4 *Time between Successive Route Discoveries*

The twin objectives of DMEF are to be energy-efficient and to determine routes that exist for a long time. DMEF accomplishes the latter objective by preferring to broadcast the RREQ messages primarily through nodes that have been moving relatively slowly in the network. As a result, the routes determined using DMEF exist for a relatively longer time in the network. The lifetime of routes determined for both DSR and LPBR protocols using DMEF as the route discovery strategy is significantly larger compared to that of the DSR and LPBR routes determined using flooding. This is because DMEF prefers to propagate RREQ packets through relatively slow moving nodes that are also close to each other. In addition, LPBR attempts to increase the time between successive global broadcast discoveries by predicting a source-destination route using the Location Update Vectors (LUVs) collected during the latest broadcast route discovery. As we increase the network density, the chances of correctly predicting at least one source-destination path in the network increases. Hence, in the case of LPBR, for a given node mobility, the time between two successive global broadcast route discoveries increases as the network density increases. For both DSR and LPBR, compared to flooding, the relative increase in the lifetime of the routes discovered using DMEF and the reduction in the frequency of DMEF route discoveries can be significantly observed with increase in network density and/or node mobility.

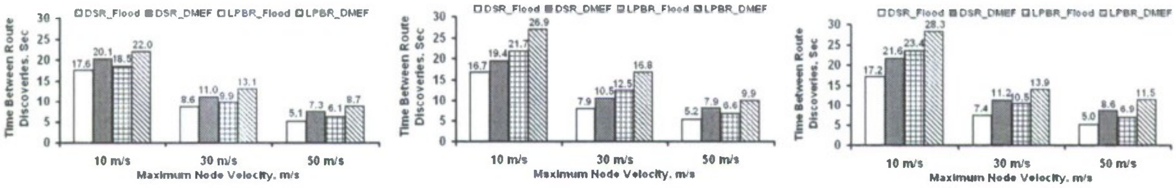


Figure 5.1: 25 Nodes **Figure 5.2:** 50 Nodes **Figure 5.3:** 75 Nodes

Figure 5: Time between Two Successive Route Discoveries

5.2.5 *Packet Delivery Ratio*

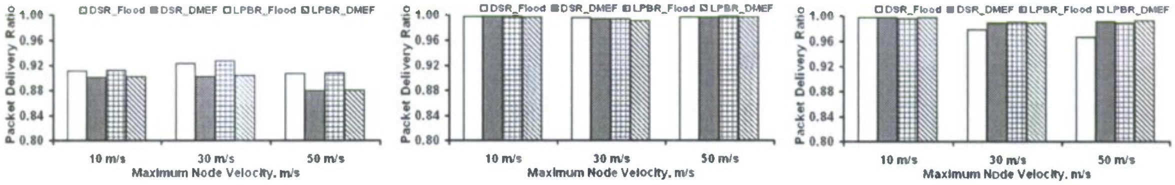


Figure 6.1: 25 Nodes **Figure 6.2:** 50 Nodes **Figure 6.3:** 75 Nodes

Figure 6: Packet Delivery Ratio

Performance results in Figures 6.1 through 6.3 illustrate that the packet delivery ratio of the two routing protocols using DMEF can be lower than that obtained using flooding only by at most 3% in low-density networks. In moderate density networks, both the route discovery strategies yield almost the same packet delivery ratio. In high density networks, the packet delivery ratio of routing protocols using DMEF can be larger than that obtained using flooding by about 3%. In high-density networks, even though flooding helps to propagate the RREQ messages through several routes, the excessive overhead generated by these redundant RREQ messages block the queues of certain heavily used nodes in the network, thus leading to sometimes a relatively lower packet delivery ratio compared to DMEF. In low-density networks, DMEF could very rarely fail to determine source-destination routes, even if one exists, due to its optimization approach of trying to shrink the range of broadcast of the RREQ messages. DMEF broadcasts RREQ messages over a relatively larger transmission range in low-density networks compared to those used for high-density networks. As we increase node density, the packet delivery ratio under both flooding and DMEF approaches unity.

5.2.6 *Energy Throughput*

For a given offered data traffic load, larger the energy throughput, the smaller the amount of energy spent in delivering the data packets to the destination. Notice that in our simulations, the number of source-destination sessions is always fixed at 15, i.e., the offered data traffic load is fixed. Based on Figures 6 and 7, we observe that with increase in the network density, the packet delivery ratio approaches unity, but the energy throughput decreases. This is because more nodes participate and spend their energy in moderate and high-density networks to route a given offered data traffic load. Note that energy consumption is in the form of direct transmissions and receptions of the intermediate nodes on a path and indirect receptions at the neighboring nodes of the intermediate nodes on a path. As we increase the network density as well as the level of node mobility, the energy throughput obtained with both DSR and LPBR using DMEF is larger than that obtained using flooding as the route discovery strategy. In low and moderate density networks and low and moderate levels of node mobility, the energy throughput for both DSR and LPBR are almost the same while using both DMEF and flooding for route discoveries.

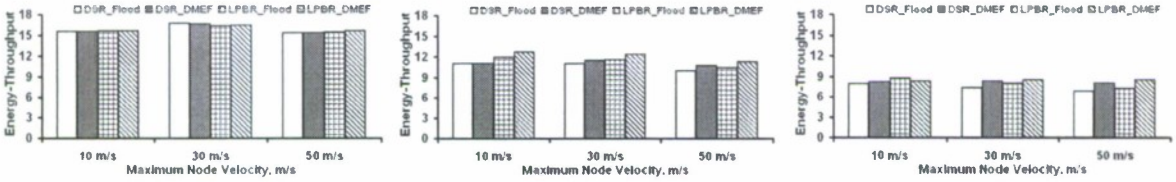


Figure 7.1: 25 Nodes **Figure 7.2:** 50 Nodes **Figure 7.3:** 75 Nodes

Figure 7: Energy-Throughput

5.3 Advantages of DMEF and Differences with Related Work

Our DMEF route discovery strategy is very effective in discovering relatively long-living routes in an energy-efficient manner and differs from the RRS algorithm in the following ways:

- RRS is highly dependent on location-service schemes like GPS, while DMEF is not dependent on any location-service scheme for its normal functionality.
- RRS requires the RREQ message header to be changed while DMEF does not require any change in the structure of the RREQ messages used for broadcasting. DMEF can be thus used with any MANET routing protocol without requiring any change in the routing protocol.
- In the case of RRS, a node lying in the stable zone of the transmitter of the RREQ message rebroadcasts the message in its complete neighborhood determined by the maximum transmission range of the node. It would be energy-efficient if the node could tune down its transmission range to its stable zone radius because it is only the recipient nodes lying in the stable zone of the transmitter that are going to rebroadcast the RREQ message. In DMEF, the transmission range of broadcast is dynamically determined by a node based on the node's own velocity and the perceived number of neighbors for the node. The transmission range for broadcast in DMEF is usually considerably less than the maximum transmission range of a node.
- RRS does not properly handle the scenario where the value of $\delta * S$ exceeds the transmission range of the node R. The value of δ has to be iteratively reduced by trial and error method to determine the connectivity between the source and destination nodes. DMEF is better than RRS because it requires only one broadcast route discovery attempt from the source to determine a route to the destination if the two nodes are indeed connected. The values of the DMEF parameters are dynamically determined at each node by the nodes themselves because a node knows better about its own velocity and neighborhood, compared to the source of the broadcast process.
- The network density does not influence the stable zone radius selected by RRS. As a result, in RRS, the number of nodes retransmitting the RREQ message in a neighborhood increases significantly as the network density is increased. DMEF is quite effective in reducing the number of nodes retransmitting the RREQ message in high-density networks.

The advantages of the DMEF scheme when compared with the broadcast route discovery strategies discussed in Section 1.2 are summarized as follows:

- The probability-based and MPR-based methods do not guarantee that the broadcast message will be routed on a path with the minimum hop count or close to the minimum hop count. Previous research [9] on the impact of these broadcast strategies on the stability and hop count of the DSR routes indicates that the hop count of the paths can be far more than the minimum hop count and the routes have a smaller lifetime than the paths discovered using flooding. The probability-based method cannot always guarantee that the RREQ message gets delivered to the destination. Also, with increase in network density, the number of nodes retransmitting the message increases for both the probability-based and MPR-based methods.

  DMEF determines paths with hop count being close to that of the minimum hop count paths and such paths have a relatively larger lifetime compared to those discovered using flooding. DMEF almost guarantees that a source-destination route is discovered if there is at least one such route in the underlying network. DMEF effectively controls the RREQ message retransmission overhead as the network density increases.
- The counter-based and area-based methods require proper selection of the threshold counter and area of coverage values for their proper functioning. Each node has to wait for a broadcast-wait-time before retransmitting the message. This can introduce significant route acquisition delays. The area-based method also requires the nodes to be location-aware and include the location information in the broadcast messages.

  With DMEF, there is no waiting time at a node to rebroadcast a received RREQ message, if the message has been received for the first time during a particular route discovery process. DMEF does not depend on any location-aware services for its operation and the structure of the RREQ message for a routing protocol need not be changed.

III Summary of Accomplishments in Research Activity 1

We have developed a novel network density and node mobility aware, energy-efficient route discovery strategy called DMEF for mobile ad hoc networks. The twin objectives of DMEF are to increase the time between successive global broadcast route discoveries and reduce the energy consumption during such global broadcast discoveries vis-à-vis flooding. Each node operates with a maximum transmission range and periodically broadcasts beacons to the neighborhood covered (called the complete neighborhood) within this range. DMEF permits each node to dynamically adjust the transmission range to broadcast the Route-Request (RREQ) messages of the route discovery process. A node that is surrounded by more neighbors advertises itself only to a limited set of nearby neighbors and a node that is surrounded by few neighbors will advertise itself to a maximum of those neighbors. Similarly, a node that is slow-moving advertises itself to a majority of its neighbors so that links formed using this node can be more stable. A node that has been fast-moving advertises itself only to the neighbors closer to it. The neighborhood dynamically chosen for a RREQ broadcast is always contained within the complete neighborhood defined by the maximum transmission range of the node. The effectiveness of DMEF has been studied through simulations with the well-known Dynamic Source Routing (DSR) protocol and the recently proposed Location Prediction Based Routing (LPBR) protocol. The benchmark used for the evaluation purposes is the commonly used flooding based global broadcast route discoveries. Simulation results indicate that DMEF is very effective in reducing the total energy spent per route discovery attempt for both DSR and LPBR. In addition, for both DSR and LPBR, DMEF reduces the number of global broadcast route discoveries by determining routes with longer lifetime, reduces the percentage of total energy spent for route discoveries and increases the energy throughput. The increase in the hop count of DSR and LPBR routes compared to that discovered using flooding is at most 8%. We conjecture that DMEF can be similarly very effective with respect to all of the other currently existing on-demand MANET routing protocols, none of which can simultaneously minimize the number of route discoveries as well as the hop count of the paths. DMEF can be used with these MANET routing protocols to discover long-living stable paths with hop count close to that of the minimum hop paths and at the same time incur less control message and energy overhead.

IV. Publication Details

(1) This research work has been published at the *2009 International Conference on Wireless Networks* held as part of the 2009 World Congress in Computer Science, Computer Engineering and Applied Computing at Las Vegas, NV, from July 13-16, 2009. The citation is as follows:

> **N. Meghanathan,** "A Density and Mobility Aware Energy-Efficient Broadcast Strategy to Minimize the Number of Route Discoveries in Mobile Ad hoc Networks," *Proceedings of the 2009 International Conference on Wireless Networks, ICWN 09,* pp. 167 – 173, Las Vegas, July 13-16, 2009.

(2) An extended version of the conference paper, featuring all the results reported in the first quarterly report, has been accepted for publication in the International Journal of Computer Science and Network Security in their Vol. 9, No. 11 Issue to be published at the end of November 2009. The citation is as follows:

> **N. Meghanathan**, "A Density and Mobility Aware Energy-Efficient Broadcast Route Discovery Strategy for Mobile Ad hoc Networks," accepted for publication in the *International Journal of Computer Science and Network Security*, Vol. 9, No. 11, November 2009.

References

[1] Y.-J. Suh, W. Kim and D.-H. Kwon, "GPS-Based Reliable routing Algorithms for Ad Hoc Networks," *The Handbook of Ad Hoc Wireless Networks*, pp. 361 – 374, CRC Press, 2003.

[2] B. Hofmann-Wellenhof, H. Lichtenegger and J. Collins, *Global Positioning System: Theory and Practice*, 5th rev. ed., Springer, Sept. 2004.

[3] S. Ni, Y. Tseng, Y. Chen and J. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," *Proceedings of ACM International Conference on Mobile Computing and Networking,* pp.151-162, 1999.

[4] D. B. Johnson, D. A. Maltz and J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Ad hoc Networks," in *Ad hoc Networking*, Chapter 5, C. E. Perkins, Eds. Addison Wesley, pp. 139 – 172, 2000.

[5] N. Meghanathan, "Location Prediction Based Routing Protocol for Mobile Ad hoc Networks," *Proceedings of the IEEE Global Communications (GLOBECOM) Conference*, New Orleans, Nov-Dec 2008.

[6] C. Bettstetter, H. Hartenstein and X. Perez-Costa, "Stochastic Properties of the Random-Way Point Mobility Model," *Wireless Networks*, pp. 555 – 567, vol. 10, no. 5, September 2004.

[7] L. M. Feeney, "An energy-consumption model for performance analysis of routing protocols for mobile ad hoc networks," *Journal of Mobile Networks and Applications*, vol. 3, no. 6, pp. 239 – 249, June 2001.

[8] IEEE Standards Department, Wireless LAN Medium Access Control and Physical Layer Specifications, IEEE Standard 802.11, 1997.

[9] N. Meghanathan, "Impact of Broadcast Route Discovery Strategies on the Performance of Mobile Ad Hoc Network Routing Protocols," Proceedings of the International Conference on High Performance Computing, Networking and Communication Systems, pp. 144-151, July 2007.

Research Activity – 2

A Multicast Version of the Location Prediction Based Routing Protocol (MLPBR)

Dr. Natarajan Meghanathan
Assistant Professor
Department of Computer Science
Jackson State University
Jackson, MS 39217
Email: natarajan.meghanathan@jsums.edu
Phone: 601-979-3661

I. Breakdown of the Research Activity to Tasks

Task No.	Task	Current Status	Timeline
1	Study the related work on multicast routing protocols for mobile ad hoc networks (MANETs)	Completed	December 2008 to January 2009
2	Develop the Multicast Extensions to LPBR (NR-MLPBR and R-MLPBR)	Completed	February 2009
3	Conduct simulations of MLPBR and compare its performance with some of the currently existing MANET multicast routing protocols	Completed	March 2009 to April 2009
4	Analyze the simulation results with respect to different performance metrics	Completed	March 2009 to April 2009

II. Description of the Tasks

Task 1: Study the Related Work on Multicast Routing Protocols for Mobile Ad hoc Networks

Multicasting is the process of sending a stream of data from one source node to multiple recipients by establishing a routing tree, which is an acyclic connected subgraph containing all the nodes in the tree. The set of receiver nodes form the multicast group. While propagating down the tree, data is duplicated only when necessary. This is better than multiple unicast transmissions. On-demand route discovery (discovering a route only when required) is often preferred over periodic route discovery and maintenance, as the latter strategy will incur significant overhead due to the frequent exchange of control information among the nodes [1]. Multicasting in ad hoc wireless networks has numerous applications [2]: collaborative and distributing computing like civilian operations, emergency search and rescue, law enforcement, warfare situations and etc.

Several MANET multicast routing protocols have been proposed in the literature [3]. They are mainly classified as: tree-based and mesh-based protocols. In tree-based protocols, only one route exists between a source and a destination and hence these protocols are efficient in terms of the number of link transmissions. The tree-based protocols can be further divided into two types: source tree-based and shared tree-based. In source tree-based multicast protocols, the tree is rooted at the source. In shared tree-based multicast protocols, the tree is rooted at a core node and all communication between the multicast source and the receiver nodes is through the core node. Even though shared tree-based multicast protocols are more scalable with respect to the number of sources, these protocols suffer under a single point of failure, the core node. On the other hand, source tree-based protocols are more efficient in terms of traffic distribution. In mesh-based multicast protocols, multiple routes exist between a source and each of the receivers of the multicast group. A receiver node receives several copies of the data packets, one copy through each of the multiple paths. Mesh-based protocols provide robustness at the expense of a larger number of link transmissions leading to inefficient bandwidth usage. A detailed classification tree of the different classes of multicast routing protocols is illustrated in Figure 1. Considering all the pros and cons of these different classes of multicast routing in MANETs, we feel the source tree-based multicast routing protocols are more efficient in terms of traffic distribution and link usage. Hence, all of our work in this research will be in the category of on-demand source tree-based multicast routing.

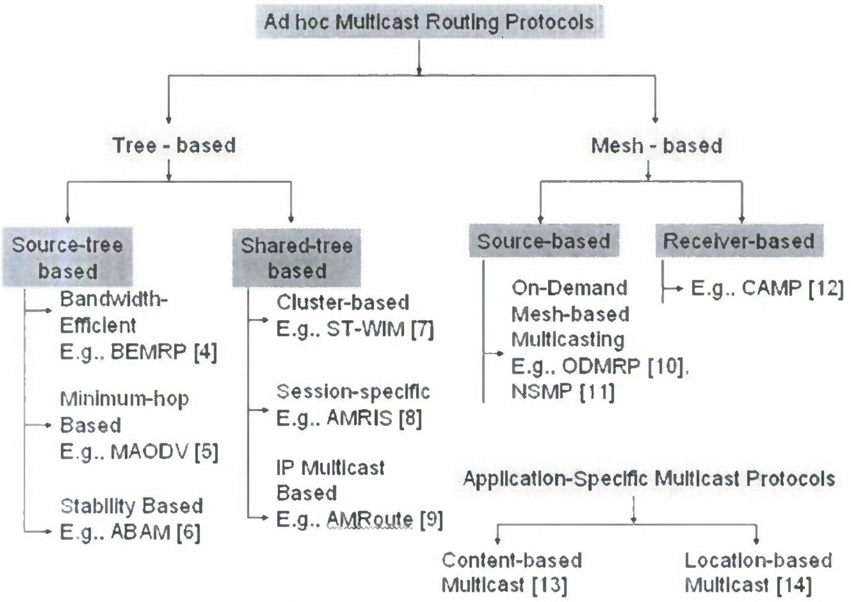


Figure 1: Classification of Ad hoc Multicast Routing Protocols

Within the class of on-demand source tree-based routing protocols, three categories of multicast routing protocols have been identified (i) Bandwidth-efficient protocols that aim to minimize the total number of links in the tree; (ii) Minimum-hop based protocols that aim to minimize the number of hops in the paths from the source to every receiver node and (iii) Stability-based protocols that aim to determine long-living stable trees and reduce the time between successive global tree discoveries. The Bandwidth-Efficient Multicast Routing Protocol (BEMRP) [4], Multicast Extension to the Ad hoc On-demand Distance Vector (MAODV) routing protocol [5] and the Associativity-Based Ad hoc Multicast (ABAM) [6] routing protocols are classical examples of the bandwidth-efficient, minimum-hop based and the stability-based multicast protocol categories. In [15], we conducted a detailed performance study of these three multicast routing protocols. Simulation study results from [15] reveal that MAODV trees are highly unstable, but have an average hop count close to the minimum number of hops between the source and the receivers. BEMRP discovers trees that have a reduced number of links but have a higher average hop count per source-receiver path. ABAM discovers trees that are stable, but have a higher average hop count per source-receiver path as well as higher number of links per tree compared to BEMRP. A significant observation in [15] is that BEMRP trees are as stable as the trees discovered using ABAM. This can be attributed to the reduced number of links in the trees determined by BEMRP, leading to longer lifetime of the trees. Because of these observations, we use only MAODV and BEMRP in our simulation studies conducted in this research work.

Task 2: Develop the Multicast Extensions to LPBR (NR-MLPBR and R-MLPBR)

2.1 Basic Idea of the Multicast Extensions

The multicast extensions of LPBR work as follows: When a source attempts to construct a multicast tree, it floods a Multicast Tree Request Message (MTRM) throughout the network. The location and mobility information of the intermediate forwarding nodes are recorded in the MTRM. Each node, including the receiver nodes of the multicast group, broadcasts the MTRM exactly once in its neighborhood. Each receiver node of the multicast group receives several MTRMs and sends a Multicast Tree Establishment Message (MTEM) on the minimum hop path traversed by the MTRMs. The set of paths traversed by the MTEMs form the multicast tree rooted at the source.

If an intermediate node of the tree notices a downstream node moving away from it, the intermediate node sends a Multicast Path Error Message (MPEM) to the source node. The source node does not immediately initiate another tree discovery procedure. Instead, the source node waits for the appropriate receiver node (whose path to the source has broken) to predict a path to the source. The receiver node predicts a new path based on the location and mobility information of the nodes collected through the MTRMs during the latest global tree discovery procedure. The receiver node attempts to locally construct the global topology by predicting the locations of the nodes in the network using the latest location and mobility information collected about the nodes.

NR-MLPBR and R-MLPBR differ from each other based on the type of path predicted and notified to the source. NR-MLPBR determines the minimum hop path to the source and sends a Multicast Predicted Path Message (MPPM) on the minimum hop path to the source. R-MLPBR assumes that each receiver knows the identity of the other receivers of the multicast group (learnt through the latest broadcast tree discovery process) and hence attempts to choose a path that will minimize the number of newly added intermediate nodes to the multicast tree. In pursuit of this, R-MLPBR determines a set of node-disjoint paths to the source on the predicted topology and sends the MPPM on that path that includes the minimum number of non-receiver nodes. If there is a tie, R-MLPBR chooses the path that has the least hop count. The source waits to receive a MPPM from the affected receiver node. If a MPPM is received within a certain time, the source considers the path traversed by the MPPM as part of the multicast tree and continues to send the data packets down the tree including to the nodes on the new path. Otherwise, the source initiates another global tree discovery procedure by broadcasting the MTRM. R-MLPBR has been thus designed to also reduce the number of links that form the multicast tree, in addition to the source-receiver hop count and the number of global tree discoveries. Nevertheless, as observed in our simulations, R-MLPBR cannot completely nullify the tradeoff between the hop count per source-receiver path and the number of links in the tree.

2.2 Objectives and Assumptions

The objective of the multicast extensions to LPBR (referred to as NR-MLPBR and R-MLPBR) is to simultaneously minimize the number of global broadcast tree discoveries as well as the hop count per source-receiver path. The Non-Receiver aware Multicast extension to LPBR (NR-MLPBR) precisely does this and it does not assume the knowledge of the receiver nodes of the multicast group at every receiver node. The Receiver-aware multicast extension of LPBR (R-MLPBR) assumes that each receiver node knows the identities of the other receiver nodes in the multicast group. This enables R-MLPBR to also reduce the number of links in the multicast tree in addition to reducing the number of global broadcast tree discoveries and the hop count per source-receiver path. Each receiver node running R-MLPBR learns the identity information of peer receiver nodes through the broadcast tree discovery procedure. Both the multicast extensions assume the periodic exchange of beacons in the neighborhood. This is essential for nodes to learn about the moving away of the downstream nodes in the multicast tree. The following sections describe the working of the two multicast extensions in detail. Unless otherwise stated specifically, the description holds good for the both NR-MLPBR and R-LPBR. We also assume that a multicast group comprises basically of receiver nodes that wish to receive data packets from an arbitrary source, which is not part of the multicast group.

2.3 Broadcast of Multicast Tree Request Messages

Whenever a source node has data packets to send to a multicast group and is not aware of a multicast tree to the group, the source initiates a broadcast tree discovery procedure by broadcasting a Multicast Tree Request Message (MTRM) to its neighbors. The source maintains a monotonically increasing sequence number for the broadcast tree discoveries it initiates to form the multicast tree. Each node, including the receiver nodes of the multicast group, on receiving the first MTRM of the current broadcast process (i.e., a MTRM with a sequence number greater than those seen before), includes its Location Update Vector, LUV in the MTRM packet. The LUV of a node comprises the following: node ID, X, Y co-ordinate information, Is Receiver flag, Current velocity and Angle of movement with respect to the X-axis. The *Is Receiver* flag in the LUV, if set, indicates that the node is a receiving node of the multicast group. The node ID is also appended on the "Route record" field of the MTRM packet. The structure of the LUV and the MTRM is shown in Figures 2 and 3 respectively.

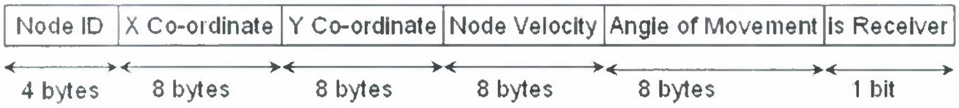


Figure 2: Location Update Vector (LUV) Collected from Each Node

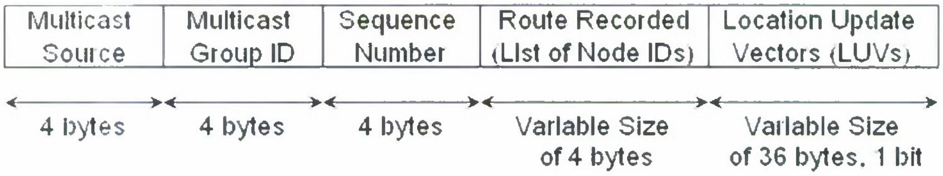


Figure 3: Structure of the Multicast Tree Request Message (MTRM)

2.4 Construction of the Multicast Tree through the Multicast Tree Establishment Message

Paths constituting the multicast tree are independently chosen at each receiver node. A receiver node gathers several MTRMs obtained across different paths and selects the minimum hop path among them by looking at the "Route Record" field in these MTRMs. A Multicast Tree Establishment Message (MTEM) is sent on the discovered minimum hop route to the source. The MTEM originating from a receiver node has the list of node IDs corresponding to the nodes that are on the minimum hop path from the receiver node to the source (which is basically the reverse of the route recorded in the MTRM). The structure of the MTEM packet is shown in Figure 4.

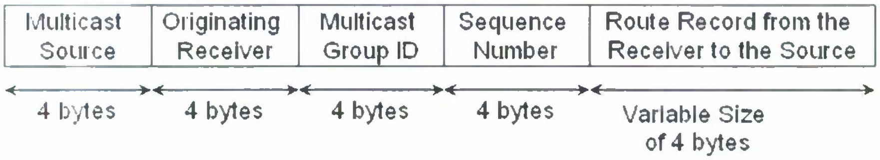
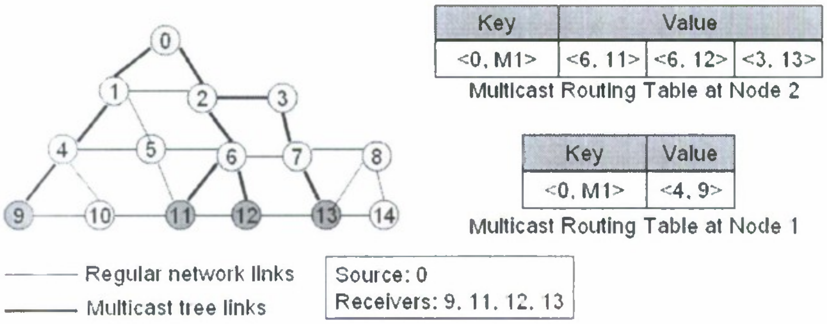


Figure 4: Structure of the Multicast Tree Establishment Message (MTEM)

An intermediate node upon receiving the MTEM packet checks its multicast routing table whether there exist an entry for the <Multicast Source, Multicast Group ID> in the table. The multicast routing table at a node is an ordered entry of <*key*><*value*> pairs, where the key is the tuple <Multicast Source, Multicast Group ID> and the value is the tuple <Downstream node, Receiver node>. The set of downstream nodes are part of the multicast tree rooted at the source node for the multicast group. If an entry exists, the intermediate node merely adds the tuple <One-hop sender of the MTEM, Originating Receiver node of the MTEM> to the list of <Downstream node, Receiver node> tuples for the multicast tree entry and does not forward the MTEM further. If a <Multicast Source, Multicast Group ID> entry does not exist in the multicast routing table, the intermediate node creates an entry and initializes it with the <One-hop sender of the MTEM, Originating Receiver node of the MTEM> tuple. Note that the one-hop sender of the MTEM is learnt through the MAC (Medium Access Control) layer header and verified using the Route Record field in the MTEM. The intermediate node then forwards the MTEM to the next downstream node on the path towards the source. The structure of the multicast routing table at a node is illustrated in Figure 5. Note that the tuples <d_a, r_a>, <d_b, r_b>, <..., ...> indicate the downstream node d_a for receiver node r_a, downstream node d_b for receiver node r_b and so on. A node could be the downstream node for more than one receiver node. Figure 6 shows an example of the multicast routing table established at some of the intermediate nodes for a multicast tree rooted at source node with ID 0 and multicast group with ID M1.

Key	Value			
<Source, Multicast Group ID>	<d_a, r_a>	<d_b, r_b>	<...., ...>	<..., ...>

Figure 5: Structure of the Multicast Routing Table at an Intermediate Node



Key	Value		
<0, M1>	<6, 11>	<6, 12>	<3, 13>

Multicast Routing Table at Node 2

Key	Value
<0, M1>	<4, 9>

Multicast Routing Table at Node 1

Figure 6: Example for Multicast Routing Table Established at Intermediate Nodes

The source node maintains a multicast routing table that has the list of <Downstream node, Receiver node> tuples for each of the multicast groups to which the source is currently communicating through a multicast session. For each MTEM received, the source adds the neighbor node that sent the MTEM and the corresponding Originating Receiver node to the list of <Downstream node, Receiver node> tuples for the multicast group.

2.5 Multicast Tree Acquisition Time and Data Transmission

After receiving the MTEMs from all the receiver nodes within a certain time called the Tree Acquisition Time (*TAT*), the source starts sending the data packets on the multicast tree. The *TAT* is based on the maximum possible diameter of the network (an input parameter in our simulations). The diameter of the network is the maximum of the hop count of the minimum hop paths between any two nodes in the network. The *TAT* is dynamically set at a node depending on the time it took to receive the first MTEM for a broadcast tree discovery procedure. If *perMulticastPeriod* denotes the time between the transmission of successive multicast packets from the source, *delFirstMTEMRecvd* indicates the time lapsed between the initiation of the MTRM broadcast and the receipt of the first MTEM and *hopsFirstMTEMRecvd* denotes the number of hops traversed by the first MTEM received, then,

$$TAT = Minimum\left[perMulticastPeriod, \left(\frac{delFirstMTEM\,\mathrm{Re}cvd * Diameter}{hopsFirstMTEM\,\mathrm{Re}cvd} \right) \right]$$

We assume the source at least knows the multicast group size, if not the identification information for each of the receivers of the multicast group. Hence, if the source fails to receive the required number of MTEMs (equal to the multicast group size), within the *TAT*, the source initiates another global broadcast tree discovery procedure. If the source receives the MTEMs from all the receivers, equaling to the multicast group size, the source starts sending the data packets down the multicast tree.

Multicast Source	Multicast Group ID	Sequence Number	More Packets	Current Dispatch Time	Time Left for Next Dispatch
4 bytes	4 bytes	4 bytes	1 bit	8 bytes	4 bytes

Figure 7: Structure of the Header of the Multicast Data Packet

The structure of the header of the multicast data packet is shown in Figure 7. The source and destination fields in the header include the identification for the source node and the multicast group ID respectively. The sequence number field in the header can be used by the receivers to accumulate and reorder the multicast data packets, incase if they are received out of order. In addition to these regular fields, the header of the multicast data packet includes three specialized fields: the 'More Packets' (*MP*) field, the 'Current Dispatch Time' (*CDT*) field and the 'Time Left for Next Dispatch' (*TNLD*) field. The *CDT* field stores the time as the number of milliseconds lapsed since Jan 1, 1970, 12 AM. These additional overhead (relative to that of the other ad hoc multicast routing protocols) associated with the header of each data packet amounts to only 12 more bytes per data packet.

The source sets the *CDT* field in all the data packets sent. In addition, if the source has any more data to send, it sets the *MP* flag to 1 and sets the appropriate value for the *TLND* field (equal to *perMulticastPeriod)*, which indicates the number of milliseconds since the *CDT*. If the source does not have any more data to send, it will set the *MP* flag to 0 and leaves the *TLND* field blank. As we assume the clocks across all nodes are synchronized, a receiver node will be able to calculate the end-to-end delay for the data packet based on the time the data packet reaches the node and the *CDT* field in the header of the data packet. Several clock synchronization algorithms (example [19][20]) have been proposed for wireless ad hoc networks. The receiver node computes and maintains the average end-to-and delay per data packet for the current path to the source by recording the sum of the end-to-end delays of all the data packets received so far on the path and the number of data packets received on the path. Accordingly, the average end-to-end delay per data packet for the current path is updated every time after receiving a new data packet on the path. If the source node has set the MP flag, the receiver node computes the 'Next Expected Packet Arrival Time' (*NEPAT*), which is *CDT* field + *TLND* field + 2*Average end-to-end delay per data packet. A timer is started for the *NEPAT* value. Since, we are using only the average end-to-end delay per data packet to measure the *NEPAT* value, the variations in the end-to-end delay of particular data packets will not very much affect the *NEPAT* value. So, the source and receiver nodes need not be perfectly synchronized. The clocks across the nodes can have small drifts and this would not very much affect the performance of the multicast extensions of LPBR.

2.6 Multicast Tree Maintenance

We assume that each node periodically exchanges beacon messages with its neighbors, located within its default maximum transmission range. If an intermediate node notices that its link with a downstream node has failed (i.e., the two nodes have moved away and are no longer neighbors), the intermediate node generates and sends a Multicast Path Error Message (MPEM) to the source node of the multicast group entry. The MPEM has information about the receiver nodes affected (obtained from the multicast routing table) because of the link failure with the downstream node. The structure of the MPEM is shown in Figure 8. The intermediate node removes the tuple(s) corresponding to downstream node(s) and the affected receiver node(s). After these deletions, if no more <Downstream node, Receiver node> tuple exists for a <Source node, Multicast group ID> key entry, the intermediate node removes the entire row for this entry from the multicast routing table.

Multicast Source	Originating Intermediate Node	Multicast Group ID	IDs of Affected Receivers
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes

Figure 8: Structure of a MPEM Message

The source node, upon receiving the MPEM, will wait to receive a Multicast Predicted Path Message (MPPM) from each of the affected receivers, within a MPPM-timer maintained for each receiver. The source node estimates a Tree-Repair Time (*TRT*) for each receiver as the time that lapsed between the reception of the MPEM from an intermediate node and the MPPM from the affected receiver. An average value for the TRT per receiver is maintained at the source as it undergoes several path failures and repairs before the next global broadcast based tree discovery. The MPPM-timer (initially set to the time it took for the source to receive the MTEM from the receiver) for a receiver will be then set to 1.5* Average *TRT* value, so that we give sufficient time for the destination to learn about the route failure and generate a new MPPM. Nevertheless, this timer will be still far less than the tree acquisition time that would be incurred if the source were to launch a global broadcast tree discovery. Hence, our approach will only increase the network throughput and does not decrease it.

2.7 Prediction of Node Location using the Location Update Vector

If a receiver node does not receive the data packet within the *NEPAT* time, it will attempt to locally construct the global topology using the location and mobility information of the nodes learnt from the latest broadcast tree discovery. Each node is assumed to be continuing to move in the same direction with the same speed as mentioned in its latest LUV. Based on this assumption and information from the latest LUVs, the location of each node at the *NEPAT* time is predicted. Whenever a node changes its direction, we assume the node is moving in the new direction with a particular velocity and towards a particular targeted destination location. As a result, a node can determine its angle of movement with respect to the X-axis at time *STIME* by computing the slope of the line joining the current location co-ordinates of the node at time *STIME* and the co-ordinates of the targeted location to which the node is moving. After reaching the targeted location, a node can change its velocity and direction to move to a new destination location.

We now explain how to predict the location of a node (say node u) at a time instant *CTIME* based on the LUV gathered from node u at time *STIME*. Let $(X_u^{STIME}, Y_u^{STIME})$ be the X and Y co-ordinates of node u at time *STIME*. Let $Angle_u^{STIME}$ and $Velocity_u^{STIME}$ represent the angle of movement with respect to the X-axis and the velocity at which node u is moving. The distance traveled by node u from time *STIME* to *CTIME* would be: $Distance_u^{STIME\text{-}CTIME} = (CTIME - STIME + 1) * Velocity_u^{STIME}$.

Let $(X_u^{CTIME}, Y_u^{CTIME})$ be the predicted location of node u at time *CTIME*. The value of X_u^{CTIME} is given by $X_u^{STIME} + Offset\text{-}X_u^{CTIME}$ and the value of Y_u^{CTIME} is given by $Y_u^{STIME} + Offset\text{-}Y_u^{CTIME}$. The offsets in the X and Y-axes, depend on the angle of movement and the distance traveled, and are calculated as follows:

$$Offset\text{-}X_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \cos(Angle_u^{STIME})$$

$$Offset\text{-}Y_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \sin(Angle_u^{STIME})$$
$$X_u^{CTIME} = X_u^{STIME} + Offset\text{-}X_u^{CTIME}$$
$$Y_u^{CTIME} = Y_u^{STIME} + Offset\text{-}Y_u^{CTIME}$$

We assume each node is initially configured with information regarding the network boundaries, given by [0, 0], [X_{max}, 0], [X_{max}, Y_{max}] and [0, Y_{max}]. When the predicted X and/or Y co-ordinate is beyond the network boundaries, we set their values to the boundary conditions as stated below.

$$\text{If } (X_u^{CTIME} < 0), \text{ then } X_u^{CTIME} = 0; \quad \text{If } (X_u^{CTIME} > X_{max}), \text{ then } X_u^{CTIME} = X_{max}$$
$$\text{If } (Y_u^{CTIME} < 0), \text{ then } Y_u^{CTIME} = 0; \quad \text{If } (Y_u^{CTIME} > Y_{max}), \text{ then } Y_u^{CTIME} = Y_{max}$$

Based on the predicted locations of each node in the network at time *CTIME*, the receiver node locally constructs the global topology. Note that there exists an edge between two nodes in the locally constructed global topology, if the predicted distance between the two nodes (with the location information obtained from the *LUV*) is less than or equal to the transmission range of the nodes. The two multicast extensions NR-MLPBR and R-MLPBR differ from each other on the nature of the paths predicted at the receiver node.

2.8 NR-MLPBR: Multicast Path Prediction

The receiver node locally runs the Dijkstra's minimum hop path algorithm [17] on the predicted global topology. If at least one path exists from the source node to the receiver node in the generated topology, the algorithm returns the minimum hop path among them. The receiver node then sends a MPPM (structure shown in Figure 9) on the discovered path with the route information included in the message.

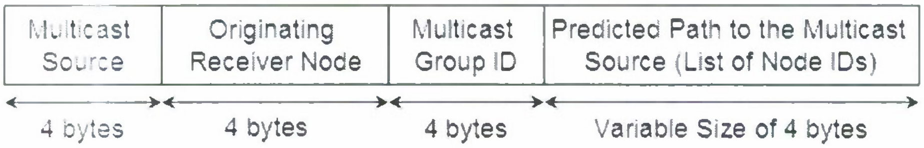


Figure 9: Structure of the Multicast Predicted Path Message (MPPM)

2.9 R-MLPBR: Multicast Path Prediction

The receiver node uses the LUV obtained from each of the intermediate nodes during the latest global tree broadcast discovery process to learn about the identification (IDs) of its peer receiver nodes that are part of the multicast group. If there existed a direct path to the source on the predicted topology, the receiver node chooses that path as the predicted path towards the source. Otherwise, the receiver node determines a set of node-disjoint paths on the predicted global topology. The node-disjoint paths to the source are ranked depending on the number of non-receiver nodes that act as intermediate nodes on the path. The path that has the least number of non-receiver nodes as intermediate nodes is preferred. The reason is a path that has the least number of non-receiver nodes is more likely to be a minimum hop path and if a receiver node lies on that path, the number of newly added links to the tree would also be reduced. R-MLPBR thus aims to discover paths with the minimum hop count and at the same time attempts to conserve bandwidth by reducing the number of links that get newly added to the tree as a result of using the predicted path. The MPPM is hence sent on the predicted path that has minimum number of non-receiver nodes. If two or more paths has the same minimum number of non-receiver nodes, R-MLPBR breaks the tie by choosing the path with the minimum hop count to the source. Figure 10 illustrates the algorithm used by R-MLPBR at a receiver node to select the best predicted path to the source.

```
Input: Graph $G(V, E)$, Set of Multicast receivers $M_R$, source $s$ and receiver $d$
Output: $s$-$d$ path
Auxiliary Variables: Graph $G''(V'', E'')$, Set of Node-disjoint paths $P_N$
Initialization: $G''(V'', E'') \leftarrow G(V, E)$, $P_N \leftarrow \varphi$.

Begin
while ( $\exists$ at least one $s$-$d$ path in $G''$)
    $p \leftarrow$ Minimum hop $s$-$d$ path in $G''$.

    if (hop count of $p = 1$)
        return $p$
    end if

    $P_N \leftarrow P_N \cup \{p\}$
    $\forall_{vertex, v \in p, v \neq s,d;\ edge, e \in Adj-list(v)}$ $G''(V'',E'') \leftarrow G''(V''-\{v\}, E''-\{e\})$

end while

$minNonReceivers \leftarrow \infty$ // the count for the minimum number of non-receivers is initialized to $\infty$.
$bestPath \leftarrow$ NULL // the best path is initialized to NULL
$minHops \leftarrow \infty$ // the minimum hop count of the best path initialized to $\infty$ (a very large value).

for ( $\forall$ path $p \in P_N$)

    $countPathNonReceivers \leftarrow 0$ // keeps track of the number of non-receiver nodes in path $p$

    for ( $\forall$ intermediate node $n \in p$)
        if ($n \notin M_R$)
            $countPathNonReceivers \leftarrow countPathNonReceivers + 1$
        end if
    end for

    if ($minNonReceivers \geq countPathReceivers$)

        if ($minNonReceivers = countPathReceivers$ AND $minHops >$ hop count of $p$)
            $bestPath \leftarrow p$
            $minHops \leftarrow$ hop count of $p$
        end if
        if ($minNonReceivers > countPathReceivers$)
            $minNonReceivers \leftarrow countPathReceivers$
            $bestPath \leftarrow p$
            $minHops \leftarrow$ hop count of $p$
        end if

    end if
```

```
    end for
    return bestPath
End
```

Figure 10: R-MLPBR Predicted Path Selection Algorithm

Note that we designed R-MLPBR to choose the path with the minimum number of non-receiver nodes, rather than the path with the maximum number of receiver nodes, as the latter design has the possibility of yielding paths with significantly larger hop count from the source to the receiver node without any guarantee on the possible reduction in the number of links. Our design of choosing the path with the minimum number of non-receiver nodes helps to maintain the hop count per source-receiver path close to that of the minimum hop count and at the same time does helps to reduce the number of links in the tree to a certain extent.

2.10 Propagation of the Multicast Predicted Path Message towards the Source

An intermediate node on receiving the MPPM, checks its multicast routing table if there already exists a key entry for the source node and the multicast group to which the MPPM belongs to. If an entry exists, the intermediate node merely adds the tuple <One-hop sender of the MPPM, Originating Receiver node of the MPPM> to the list of <Downstream node, Receiver node> tuples for the multicast tree entry. If the <Multicast Source, Multicast Group ID> entry does not exist in the multicast routing table, the intermediate node creates an entry and initializes it with the <One-hop sender of the MPPM, Originating Receiver node of the MPPM> tuple. In either case, the MPPM is then forwarded to the next downstream node on the path towards the source. If the source node receives the MPPM from the appropriate receiver node before the MPPM-timer expires, it indicates that the predicted path does exist in reality. A costly global broadcast tree discovery has been thus avoided. The source continues to send the data packets down the multicast tree. The source node estimates the Tree Repair Time (TRT) as the time lapsed between the reception of the MPEM from an intermediate node and the MPPM from the appropriate receiver node. An average value of the TRT for each receiver node is thus maintained at the source as it undergoes several route failures and repairs before the next global broadcast-based tree discovery.

2.11 Handling Prediction Failure

If an intermediate node attempting to forward the MPPM of a receiver node could not successfully forward the packet to the next node on the path towards the source, the intermediate node informs the absence of the route through a MPPM-Error packet (structure shown in Figure 11) sent back to the receiver node. The receiver node on receiving the MPPM-Error packet discards all the LUVs and does not generate any new MPPM. The receiver will wait for the multicast source to initiate a global broadcast-based tree discovery. After the MPPM-timer expires, the multicast source initiates a new global broadcast-based tree discovery procedure.

Node Sending the MPPM-Error Packet	Multicast Source	Originating Receiver Node	Multicast Group ID	Sequence No. of latest MTRM
4 bytes	4 bytes	4 bytes	4 bytes	4 bytes

Figure 11: Structure of the MPPM-Error Packet

Task 3: Conduct Simulations of MLPBR and Compare its Performance with some of the Currently Existing MANET Multicast Routing Protocols

The network dimension used is a 1000m x 1000m square network. The transmission range of each node is assumed to be 250m. The number of nodes used in the network is 25, 50 and 75 nodes representing networks of low, medium and high density with an average distribution of 5, 10 and 15 neighbors per node respectively. Initially, nodes are uniformly randomly distributed in the network. We compare the performance of NR-MLPBR and R-MLPBR with that of the minimum-hop based MAODV and the link-efficient BEMRP protocols. We implemented all of these four multicast routing protocols in a discrete-event simulator developed in Java. The broadcast tree discovery strategies simulated are the default flooding approach and the density and mobility aware energy-efficient broadcast strategy called DMEF [18]. The simulation parameters are summarized in Table 1.

Table 1: Simulation Conditions

Network Size	1000m x 1000m	
Number of nodes	25 (low density), 50 (moderate density) and 75 (high density)	
Transmission Range	250 m	
Physical Layer	Signal Propagation Model	Two-ray ground reflection model [21]
MAC Layer	IEEE 802.11 [22]	
	Link Bandwidth	2 Mbps
	Interface Queue	FIFO-based, size 100
Routing Protocols	BEMRP [4], MAODV [5], NR-MLPBR and R-MLPBR	
Broadcast Strategy	Flooding and DMEF [18]	
Mobility Model	Random Way Point Model [23]	
	Minimum Node Speed, m/s	0 m/s
	Maximum Node Speed, m/s	Low-10; Medium-30; High-50
	Pause Time	0 second
Traffic Model	Constant Bit Rate (CBR), UDP	
	Multicast Group Size (# Receivers)	Small: 2; Medium: 4, 8; High: 12, 24
	Data Packet Size	512 bytes
	Packet Sending Rate	4 Packets/ second

Simulations are conducted with a multicast group size of 2, 4 (small size), 8, 12 (moderate size) and 24 (larger size) receiver nodes. For each group size, we generated 5 lists of receiver nodes and simulations were conducted with each of them. Traffic sources are constant bit rate (CBR). Data packets are 512 bytes in size and the packet sending rate is 4 data packets/second. The multicast session continues until the end of the simulation time, which is 1000 seconds. The node mobility model used is the Random Waypoint model [23]. The transmission energy and reception energy per hop is set at 1.4 W and 1 W respectively. Initial energy at each node is 1000 Joules. Each node periodically broadcasts a beacon message within its neighborhood to make its presence felt to the other nodes in the neighborhood.

3.1 Multicast Extension of Ad hoc On-demand Distance Vector (MAODV) Routing Protocol

MAODV [5] is the multicast extension of the well-known Ad hoc On-demand Distance Vector (AODV) unicast routing protocol [24]. Here, a receiver node joins the multicast tree through a member node that lies on the minimum-hop path to the source.

A potential receiver wishing to join the multicast group broadcasts a *Route-Request* (RREQ) message. If a node receives the RREQ message and is not part of the multicast tree, the node broadcasts the message in its neighborhood and also establishes the reverse path by storing the state information consisting of the group address, requesting node id and the sender node id in a temporary cache. If a node receiving the RREQ message is a member of the multicast tree and has not seen the RREQ message earlier, the node waits to receive several RREQ messages and sends back a *Route-Reply* (RREP) message on the shortest path to the receiver. The member node also informs in the RREP message, the number of hops from itself to the source. The potential receiver receives several RREP messages and selects the member node which lies on the shortest path to the source. The receiver node sends a *Multicast Activation* (MACT) message to the selected member node along the chosen route. The route from the source to receiver is set up when the member node and all the intermediate nodes in the chosen path update their multicast table with state information from the temporary cache. A similar approach can be used in NR-MLPBR and R-MLPBR when a new receiver node wishes to join the multicast group.

Tree maintenance in MAODV is based on the expanding ring search (ERS) approach, using the RREQ, RREP and MACT messages. The downstream node of a broken link is responsible for initiating ERS to issue a fresh RREQ for the group. This RREQ contains the hop count of the requesting node from the source and the last known sequence number for that group. It can be replied only by the member nodes whose recorded sequence number is greater than that indicated in the RREQ and whose hop distance to the source is smaller than the value indicated in the RREQ.

3.2 Bandwidth-Efficient Multicast Routing Protocol (BEMRP)

According to BEMRP [4], a newly joining node to the multicast group opts for the nearest forwarding node in the existing tree, rather than choosing a minimum-hop count path from the source of the multicast group. As a result, the number of links in the multicast tree is reduced leading to savings in the network bandwidth.

Multicast tree construction is receiver-initiated. When a node wishes to join the multicast group as a receiver, it initiates the flooding of *Join control* packets targeted towards the nodes that are currently members of the multicast tree. On receiving the first *Join control* packet, the member node waits for a certain time before sending a *Reply* packet. The member node sends a *Reply* packet on the path, traversed by the *Join* control packet, with the minimum number of intermediate forwarding nodes. The newly joining receiver node collects the *Reply* packets from different member nodes and would send a *Reserve* packet on that path that has the minimum number of forwarding nodes from the member node to itself.

To provide more bandwidth efficiency, the tree maintenance approach in BEMRP is hard-state based, i.e. a member node transmits control packets only after a link breaks. BEMRP uses two schemes to recover from link failures: *Broadcast-multicast scheme* – the upstream node of the broken link is responsible for finding a new route to the previous downstream node; *Local-rejoin scheme* – the downstream node of the broken link tries to rejoin the multicast group using a limited flooding of the *Join* control packets.

3.3 Broadcast Strategy: Flooding

Flooding is a widely-used approach for disseminating a message from one node to all the nodes in a network. In the case of on-demand ad hoc routing protocols [3][24], flooding has been also used to discover a path between a pair of nodes in the network, whenever required. For a given network density, flooding offers the highest probability for each node in the network to receive one or more copies of the flooded message.

We simulate flooding as follows: The initiating source node sets a monotonically increasing value for the Multicast Tree Request Message (MTRM) and broadcasts the message to its complete neighborhood formed by the default maximum transmission range of the node. Each node that receives the MTRM checks if it has received a MTRM with the same or higher sequence number. If so, the received MTRM is simply discarded. Otherwise, the intermediate node inserts its own ID in the Route Record field of the MTRM and broadcasts the message within its complete neighborhood. Each receiver node of the multicast group upon receiving the first MTRM of a broadcast tree discovery process will include their ID in the route record field and rebroadcast that MTRM further. To select a route to reply back to the source, the receiver node collects the MTRMs received from different paths, selects the minimum hop path and sends a Multicast Tree Establishment Message (MTEM) on the selected minimum hop path to the source.

3.4 Broadcast Strategy: DMEF

In Research Activity – 1 [18], we had proposed a density and mobility aware energy-efficient broadcast strategy (called DMEF) to discover long-living stable routes with a reduced energy spent during route discovery. DMEF takes into consideration the number of neighbors of a node (a measure of network density) and node mobility. The average hop count of the routes discovered using DMEF is only at most about 8% more than that discovered using flooding.

We simulate DMEF as follows for broadcast multicast tree discoveries: The transmission range of a MTRM broadcast is not fixed for every node. A node that is surrounded by more neighbors in the complete neighborhood will broadcast the MTRM only within a smaller neighborhood that would be sufficient enough to pick up the message and forward it to the other nodes in the rest of the network. On the other hand, a node that is surrounded by fewer neighbors in the complete neighborhood will broadcast the MTRM to a larger neighborhood (but still contained within the complete neighborhood) so that a majority of the nodes in the complete neighborhood can pick up the message and rebroadcast it further. A node rebroadcasts a MTRM at most once. The density aspect of DMEF thus helps to reduce the unnecessary transmission and reception of broadcast MTRMs and conserves energy.

To discover stable trees that exist for a longer time, DMEF takes the following approach: A node that is highly mobile makes itself available only to a smaller neighborhood around itself, whereas a node that is less mobile makes itself available over a larger neighborhood (but still contained within the complete neighborhood). The reasoning is that links involving a slow moving node will exist for a long time. Hence, it is better for a slow moving node to advertise itself to a larger neighborhood so that the links (involving this node) that are part of the routes discovered will exist for a longer time. On the other hand, a fast moving node will have links of relatively longer lifetime with neighbors that are closer to it. Hence, it is worth to let a fast moving node advertise only to its nearby neighbors.

The rest of the broadcast process is similar to flooding. The receiver node upon receiving the first MTRM will include its identification field in the MTRM and rebroadcast it further depending on its current perceived neighborhood density and own mobility. To select a route to reply back to the source, the receiver node collects the MTRMs received from different paths, selects the minimum hop path and sends a Multicast Tree Establishment Message (MTEM) on the selected minimum hop path to the source.

3.5 Performance Metrics

The performance metrics studied through this simulation are the following:

- **Number of Links per Tree:** This is the time averaged number of links in the multicast trees discovered and computed over the entire multicast session. The notion of "time-average" is explained as follows: Let there be multicast trees T1, T2, T3 with 5, 8 and 6 links used for time 12, 6 and 15 seconds respectively, then the time averaged number of links in the multicast trees is given by (5*12+8*6+6*15)/ (12+6+15) = 6 and not merely 6.33, which is the average of 5, 8 and 6.
- **Hop Count per Source-Receiver Path**: This is the time averaged hop count of the paths from the source to each receiver of the multicast group and computed over the entire multicast session.
- **Time between Successive Broadcast Tree Discoveries:** This is the time between two successive broadcast tree discoveries, averaged over the entire multicast session. This metric is a measure of the lifetime of the multicast trees discovered and also the effectiveness of the path prediction approach followed in NR-MLPBR and R-MLPBR.
- **Energy Throughput:** This is the average of the ratio of the number of data packets reaching the destination to the sum of the energy spent across all the nodes in the network.

- **Energy Consumed per Node:** This is the sum of the energy consumed at a node due to the transfer of data packets as part of the multicast session, broadcast tree discoveries as well as the periodic broadcast and exchange of beacons in the neighborhood.
- **Energy Consumed per Tree Discovery:** This is the average of the total energy consumed for the global broadcast based tree discovery attempts. This includes the sum of the energy consumed to transmit (broadcast) the MTRM packets to the nodes in the neighborhood and to receive the MTRM packet sent by each node in the neighborhood, summed over all the nodes. It also includes the energy consumed to transmit the MTEM packet from each receiver to the source of the multicast session.

Task 4: Analyze the simulation results with respect to different performance metrics

The performance results for each metric displayed in Figures 12 through 24 are an average of the results obtained from simulations conducted with 5 sets of multicast groups and 5 sets of mobility profiles for each group size, node velocity and network density values. The multicast source in each case was selected randomly among the nodes in the network and the source is not part of the multicast group. The nodes that are part of the multicast group are merely the receivers.

4.1 Number of Links per Multicast Tree

The number of links per multicast tree (refer figures 12 and 13) is a measure of the efficiency of the multicast routing protocol in reducing the number of link transmissions during the transfer of the multicast data from the source to the receivers of the multicast group. The smaller is the number of links in the tree, the larger the link transmission efficiency of the multicast routing protocol. If fewer links are part of the tree, then the chances of multiple transmissions in the network increase and this increases the efficiency of link usage and the network bandwidth. Naturally, the BEMRP protocol, which has been purely designed to yield bandwidth-efficient multicast trees, discovers trees that have a reduced number of links for all the operating scenarios. This leads to larger hop count per source-receiver paths for BEMRP as observed in figures 14 and 15.

R-MLPBR, which has been designed to choose the predicted paths with the minimum number of non-receiver nodes, manages to significantly reduce the number of links vis-à-vis the MAODV and NR-MLPBR protocols. R-MLPBR attempts to minimize the number of links in the multicast tree without yielding to a higher hop count per source-receiver path. But, the tradeoff between the link efficiency and the hop count per source-receiver path continues to exist and it cannot be nullified. In other words, R-MLPBR cannot discover trees that have minimum number of links as well as the minimum hop count per source-receiver path. Nevertheless, R-MLPBR is the first multicast routing protocol that yields trees with the reduced number of links and at the same time, with a reduced hop count (close to the minimum) per source-receiver path.

4.1.1 Number of Links per Tree (Tree Discovery Strategy: Flooding)

- *Impact of Node Mobility*: For a given network density and multicast group size, we do not see any appreciable variation in the number of links per tree for each of the multicast routing protocols studied.
- *Impact of Network Density*: For a given multicast group size, the number of links per tree for MAODV and NR-MLPBR is about 4-15%, 8-28% and 10-38% more than that incurred with BEMRP in networks of low, moderate and high density respectively. This illustrates that as the network density increases, BEMRP attempts to reduce the number of links per tree by incorporating links that can be shared by multiple receivers on the paths towards the source. On the other hand, both MAODV and NR-MLPBR attempt to choose minimum hop paths between the source and any receiver and hence exploit the increase in network density to discover minimum hop paths, but at the cost of the link efficiency. On the other hand, R-MLPBR attempts to reduce the number of links per tree as we

increase the network density. For a given multicast group size, the number of links per tree for R-MLPBR is about 4-15%, 8-18% and 10-21% more than that incurred by BEMRP. This shows that R-MLPBR is relatively more scalable, similar to BEMRP, with increase in the network density.

- *Impact of Multicast Group Size*: For a given level of node mobility, for smaller multicast groups (of size 2), the number of links per tree for MAODV, NR-MLPBR and R-MLPBR is about 3-7%, 8-11% and 9-14% more than that incurred for BEMRP in low, medium and high-density networks respectively. For medium and large-sized multicast groups, the number of links per tree for both MAODV and NR-MLPBR is about 7-15%, 17-28% and 22-38% more than that incurred for BEMRP in low, medium and high-density networks respectively. On the other hand, the number of links per tree for R-MLPBR is about 6-15%, 12-18% and 16-21% more than that incurred for BEMRP in low, medium and high-density networks respectively. This shows that R-MLPBR is relatively more scalable, similar to BEMRP, with increase in the multicast group size.

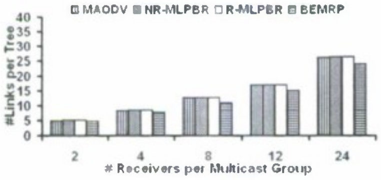


Figure 12.1: 25 nodes, 10 m/s

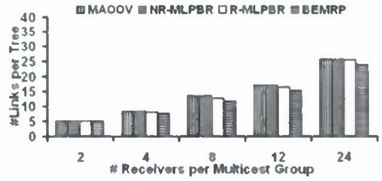


Figure 12.2: 25 nodes, 30 m/s

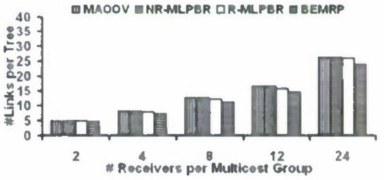


Figure 12.3: 25 nodes, 50 m/s

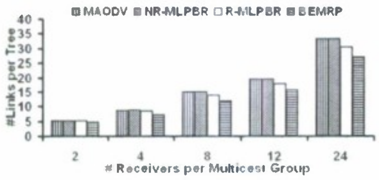


Figure 12.4: 50 nodes, 10 m/s

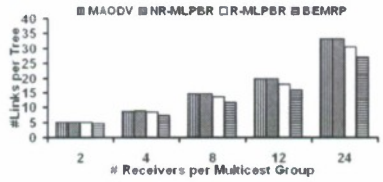


Figure 12.5: 50 nodes, 30 m/s

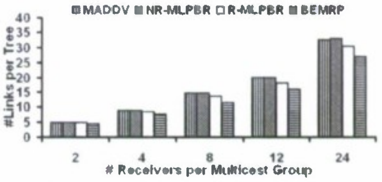


Figure 12.6: 50 nodes, 50 m/s

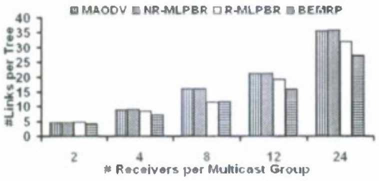


Figure 12.7: 75 nodes, 10 m/s

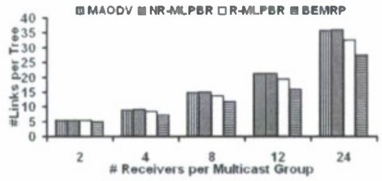


Figure 12.8: 75 nodes, 30 m/s

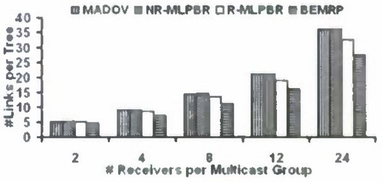


Figure 12.9: 75 nodes, 50 m/s

Figure 12: Average Number of Links per Multicast Tree (Route Discovery Procedure: Flooding)

4.1.2 Number of Links per Tree (Tree Discovery Strategy: DMEF)

- *Impact of Node Mobility*: For each of the multicast routing protocols, as the maximum node velocity is increased from 10 m/s to 30 m/s, the number of links per multicast tree increases as large as up to 24% (for multicast groups of small and moderate sizes) and 3% (for multicast groups of larger size). As the maximum node velocity is increased from 10 m/s to 50 m/s, the number of links per multicast tree increases as large as up to 15% (for multicast groups of small and moderate sizes) and 5% (for multicast groups of larger size). This shows that DMEF can yield multicast trees with reduced number of links in low node mobility, especially for multicast groups of small and moderate sizes.
- *Impact of Network Density*: For a given multicast group size, the number of links per tree for MAODV and NR-MLPBR is about 4-15%, 8-28% and 10-35% more than that incurred with BEMRP in networks of low, moderate and high density respectively. For a given multicast group size, the

number of links per tree for R-MLPBR is about 3-9%, 8-18% and 9-24% more than that incurred by BEMRP. The results are more or less similar to obtained using flooding as the tree discovery strategy.

- *Impact of Multicast Group Size*: For a given level of node mobility, for smaller multicast groups (of size 2), the number of links per tree for MAODV, NR-MLPBR and R-MLPBR is about 4-7%, 8-9% and 9-14% more than that incurred for BEMRP in low, medium and high-density networks respectively. For medium and large-sized multicast groups, the number of links per tree for both MAODV and NR-MLPBR is about 7-15%, 17-28% and 21-35% more than that incurred for BEMRP in low, medium and high-density networks respectively. On the other hand, the number of links per tree for R-MLPBR is about 6-8%, 11-18% and 15-24% more than that incurred for BEMRP in low, medium and high-density networks respectively. These results are almost the same as that obtained when flooding is used as the tree discovery strategy.

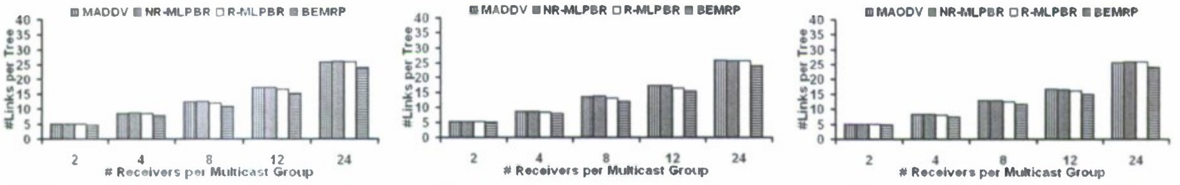


Figure 13.1: 25 nodes, 10 m/s **Figure 13.2:** 25 nodes, 30 m/s **Figure 13.3:** 25 nodes, 50 m/s

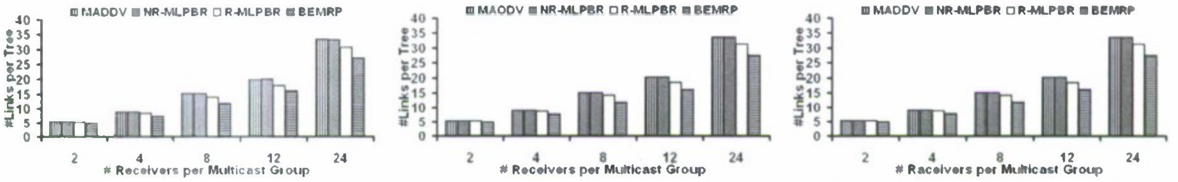


Figure 13.4: 50 nodes, 10 m/s **Figure 13.5:** 50 nodes, 30 m/s **Figure 13.6**: 50 nodes, 50 m/s

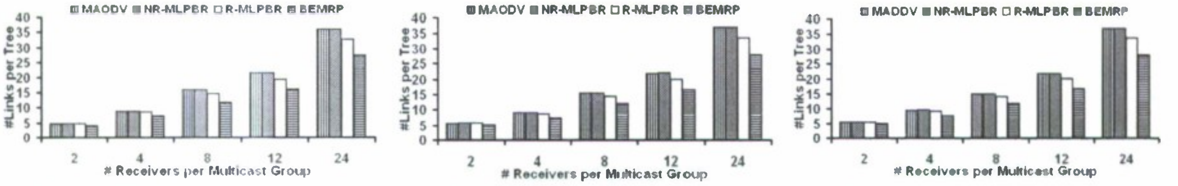


Figure 13.7: 75 nodes, 10 m/s **Figure 13.8:** 75 nodes, 30 m/s **Figure 13.9**: 75 nodes, 50 m/s

Figure 13: Average Number of Links per Multicast Tree (Route Discovery Procedure: DMEF)

4.2 Hop Count per Source-Receiver Path

All the three multicast routing protocols – MAODV, NR-MLPBR and R-MLPBR, incur almost the same average hop count per source-receiver and it is considerably lower than that incurred for BEMRP. The hop count per source-receiver path is an important metric and it is often indicative of the end-to-end delay per multicast packet from the source to a specific receiver. BEMRP incurs a significantly larger hop count per source-receiver path and this can be attributed to the nature of this multicast routing protocol to look for trees with a reduced number of links. When multiple receiver nodes have to be connected to the source through a reduced set of links, the hop count per source-receiver path is bound to increase. In performance figures 14 and 15, we can see a significant increase in the hop count per source-receiver path as we increase the multicast group size. In the case of flooding, the hop count per source-receiver path for BEMRP can be as large as 41%, 57% and 59% more than that of the hop count per source-receiver path incurred for the other three multicast routing protocols. In the case of DMEF, the hop count per source-receiver path for BEMRP can be as large as 36%, 49% and 53% more than that of the hop count per source-receiver path incurred for the other three multicast routing protocols. The increase in the hop count per source-receiver path for BEMRP is slightly less than that obtained under flooding.

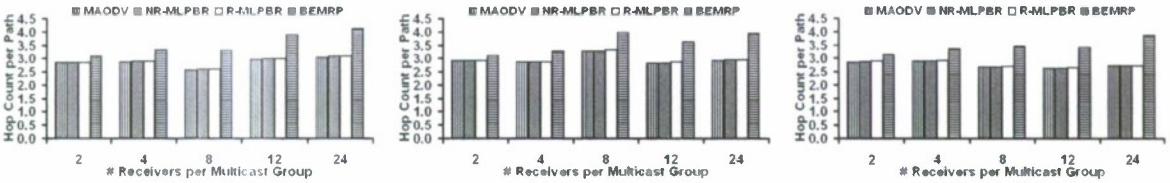


Figure 14.1: 25 nodes, 10 m/s **Figure 14.2:** 25 nodes, 30 m/s **Figure 14.3**: 25 nodes, 50 m/s

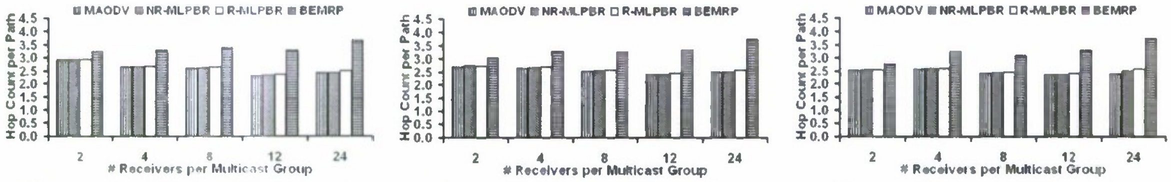


Figure 14.4: 50 nodes, 10 m/s **Figure 14.5:** 50 nodes, 30 m/s **Figure 14.6**: 50 nodes, 50 m/s

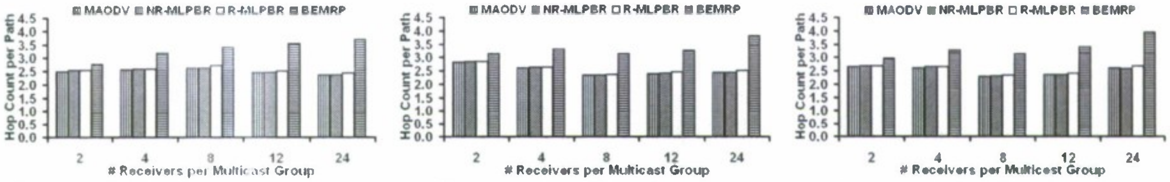


Figure 14.7: 75 nodes, 10 m/s **Figure 14.8:** 75 nodes, 30 m/s **Figure 14.9**: 75 nodes, 50 m/s

Figure 14: Average Hop Count per Source-Receiver Path (Route Discovery Procedure: Flooding)

4.2.1 Hop Count per Source-Receiver Path (Tree Discovery Strategy: Flooding)

- *Impact of Node Mobility*: For a given network density and group size, we do not see any appreciable variation in the hop count per source-receiver path for each of the multicast routing protocols studied.
- *Impact of Network Density*: As we increase the network density, the hop count per source-receiver path decreases. This is mainly observed in the case of the minimum-hop based MAODV, NR-MLPBR and R-MLPBR. In the case of BEMRP, the impact of network density on the decrease in the hop count is relatively less as it is a bandwidth-efficient multicast routing protocol attempting to reduce the number of links in the tree. In networks of moderate density (50 nodes), the hop count per source-receiver path for the three minimum hop based multicast protocols is about 6%, 9-12% and 15-19% less than that incurred in low-density networks for multicast groups of small, medium and larger sizes respectively. In high density networks (75 nodes), the hop count per source-receiver path for the three minimum-hop based multicast protocols is about 7-9%, 11-18% and 15-19% less than that incurred in low-density networks for multicast groups of small, medium and larger sizes respectively. In the case of BEMRP, the maximum reduction in the hop count with increase in network density is within 10%.
- *Impact of Multicast Group Size*: For smaller multicast groups (of size 2), the hop count per source-receiver path for BEMRP can be 6-10%, 8-12% and 10-12% more than that of the other three multicast routing protocols in networks of low, moderate and high density respectively. For medium sized multicast groups, the hop count per source-receiver path for BEMRP can be 14-29%, 21-30% and 23-37% more than that of the other three multicast routing protocols in networks of low,

moderate and high density respectively. For large-sized multicast groups, the hop count per source-receiver path for BEMRP can be 27-41%, 35-57% and 33-59% more than that of the hop count per source-receiver path for the other three multicast routing protocols in networks of low, moderate and high density respectively.

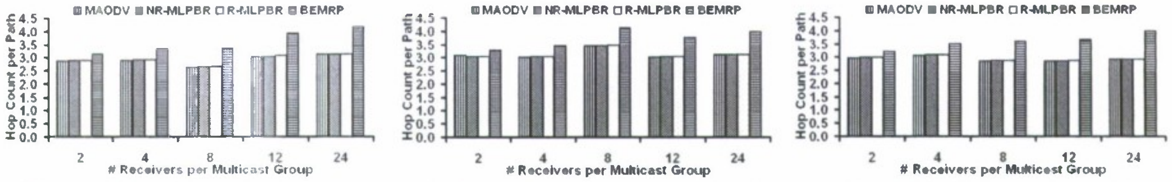


Figure 15.1: 25 nodes, 10 m/s **Figure 15.2:** 25 nodes, 30 m/s **Figure 15.3**: 25 nodes, 50 m/s

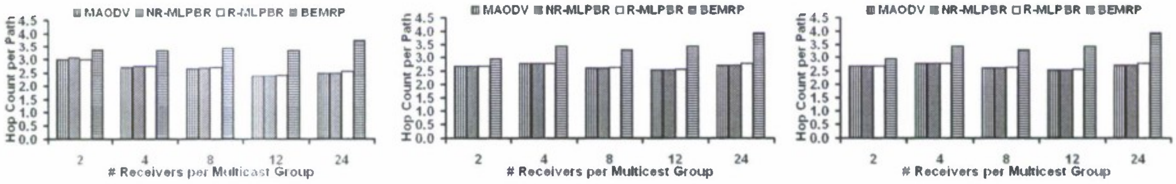


Figure 15.4: 50 nodes, 10 m/s **Figure 15.5:** 50 nodes, 30 m/s **Figure 15.6**: 50 nodes, 50 m/s

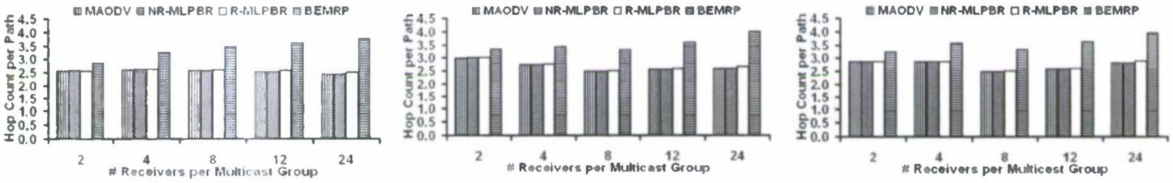


Figure 15.7: 75 nodes, 10 m/s **Figure 15.8:** 75 nodes, 30 m/s **Figure 15.9**: 75 nodes, 50 m/s

Figure 15: Average Hop Count per Source-Receiver Path (Route Discovery Procedure: DMEF)

4.2.2 Hop Count per Source-Receiver Path (Tree Discovery Strategy: DMEF)

- *Impact of Node Mobility*: For each of the multicast routing protocols, as the maximum node velocity is increased from 10 m/s to 30 m/s, we observe that the hop count per source-receiver path increases as large as up to 17% (for multicast groups of small and moderate sizes) and 7% (for multicast groups of larger size). As the maximum node velocity is increased from 10 m/s to 50 m/s, we observe that the number of links per multicast tree increases as large as up to 13% (for multicast groups of small and moderate sizes) and 15% (for multicast groups of larger size). This shows that DMEF can yield multicast trees with reduced hop count per source-receiver path under low node mobility, especially for multicast groups of small and moderate sizes.
- *Impact of Network Density*: The impact is similar to that observed in the case of flooding. For the minimum-hop based multicast protocols, with increase in network density, the hop count per source-receiver path decreases significantly. On the other hand, in the case of BEMRP, the decrease in the hop count per source-receiver path is relatively less, with increase in the network density.
- *Impact of Multicast Group Size*: For smaller multicast groups (of size 2), the hop count per source-receiver path for BEMRP can be 6-9%, 9-12% and 10-12% more than that of the other three multicast routing protocols in networks of low, moderate and high density respectively. For medium sized multicast groups, the hop count per source-receiver path for BEMRP can be 13-28%, 20-29% and 23-34% more than that of the other three multicast routing protocols in networks of low, moderate and high density respectively. For large-sized multicast groups, the hop count per source-receiver path for

BEMRP can be 24-36%, 33-50% and 36-54% more than that of the hop count per source-receiver path for the other three multicast routing protocols in networks of low, moderate and high density respectively.

4.3 Time Between Successive Broadcast Tree Discoveries

The time between successive broadcast tree discoveries is a measure of the stability of the multicast trees and the effectiveness of the location prediction and path prediction approach of the two multicast extensions. For a given condition of node density and node mobility, both NR-MLPBR and R-MLPBR incur relatively larger time between successive broadcast tree discoveries for smaller and medium sized multicast groups. MAODV tends to be more unstable as the multicast group size is increased, owing to the minimum hop nature of the paths discovered and absence of any path prediction approach. For larger multicast groups, BEMRP tends to perform better by virtue of its tendency to strictly minimize only the number of links in the tree. On the other hand, NR-MLPBR attempts to reduce the hop count per source-receiver path and ends up choosing predicted paths that increase the number of links in the tree, quickly leading to the failure of the tree. The time between successive tree discoveries for R-MLPBR is 15-25%, 15-59% and 20-82% more than that obtained for MAODV in networks of low, moderate and high density respectively. For a given level of node mobility and network density, MAODV trees become highly unstable as the multicast group size increases. For multicast groups of size 2 and 4, the time between successive broadcast tree discoveries for NR-MLPBR and R-MLPBR is greater than that obtained for BEMRP, especially in networks of low and moderate network density. For larger multicast group sizes, when we employ flooding, BEMRP tends to incur larger time between successive broadcast tree discoveries compared to NR-MLPBR and R-MLPBR. On the other hand, when we employ DMEF, R-MLPBR tends to incur larger time between successive broadcast tree discoveries compared to BEMRP, even for larger group sizes.

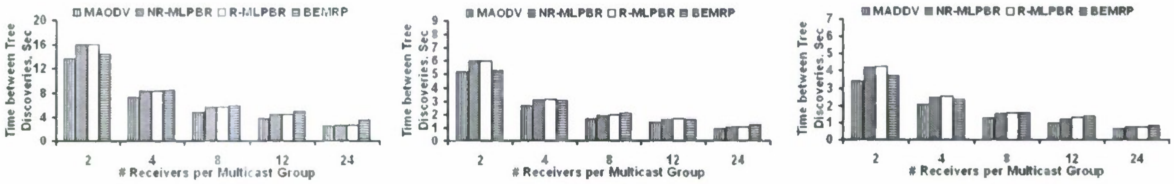


Figure 16.1: 25 nodes, 10 m/s **Figure 16.2:** 25 nodes, 30 m/s **Figure 16.3**: 25 nodes, 50 m/s

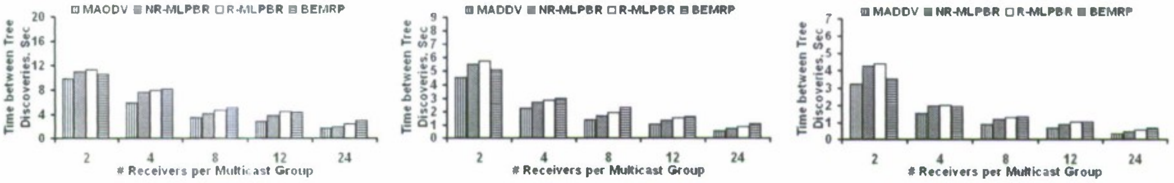


Figure 16.4: 50 nodes, 10 m/s **Figure 16.5:** 50 nodes, 30 m/s **Figure 16.6**: 50 nodes, 50 m/s

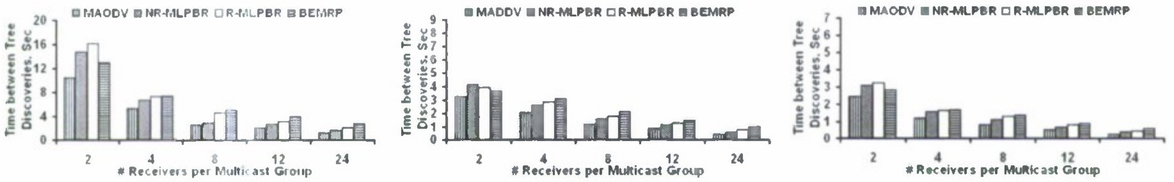


Figure 16.7: 75 nodes, 10 m/s **Figure 16.8:** 75 nodes, 30 m/s **Figure 16.9**: 75 nodes, 50 m/s

Figure 16: Average Time between Successive Tree Discoveries (Route Discovery Procedure: Flooding)

4.3.1 Time Between Successive Broadcast Tree Discoveries (Tree Discovery Strategy: Flooding)

- *Impact of Node Mobility*: For a given multicast group size, network density and multicast routing protocol, the time between successive broadcast tree discoveries at maximal node velocity of 30 m/s is roughly about 28-47% of that obtained at maximal node velocity of 10 m/s. The time between successive broadcast tree discoveries at maximal node velocity of 50 m/s is roughly about 21-36% of that obtained at maximal node velocity of 10 m/s.
- *Impact of Network Density*: For each multicast routing protocol, for a given multicast group size and level of node mobility, as the network density increases, the time between successive broadcast tree discoveries decreases. This is mainly observed for the minimum-hop based multicast protocols (especially MAODV and NR-MLPBR) which incur a reduced hop count per source-receiver path as we increase the network density. But, such minimum hop paths obtained in moderate and high-density networks are relatively less stable than those obtained in low-density networks. For a given multicast group size and low node mobility, the time between successive tree discoveries in networks of moderate density (50 nodes) for MAODV and NR-MLPBR is 67-90% and for R-MLPBR and BEMRP is 73-96% of those obtained in networks of low-density. For a given multicast group size and low node mobility, the time between successive tree discoveries in networks of high density (75 nodes) is 51-80% for MAODV and NR-MLPBR and for R-MLPBR and BEMRP is 70-90% of those obtained in networks of low-density.

  In low-density networks, the time between successive route discoveries for R-MLPBR and NR-MLPBR is about 10-15% more than that obtained for BEMRP for smaller multicast groups and is almost the same as that of BEMRP for moderately sized multicast groups. For larger multicast groups, the time between successive route discoveries for R-MLPBR and NR-MLPBR can be about 10-23% less than that obtained for BEMRP. In moderate and high density networks, the time between successive route discoveries for R-MLPBR is about 7-25% more than that obtained for BEMRP for smaller multicast groups and is about the same of moderately size multicast groups. For larger multicast groups, the time between successive route discoveries for R-MLBPR can be about 15-25% less than that obtained for BEMRP. In both moderate and high-density networks, R-MLPBR incurs larger time between successive route discoveries (as large as 30%) compared to NR-MLPBR.
- *Impact of Multicast Group Size*: For a given network density and node mobility, the time between successive route discoveries decreases as the multicast group size increases. For smaller group sizes, the time between successive broadcast tree discoveries for MAODV and BEMRP is respectively about 80%-90% and 85%-94% of that incurred for NR-MLPBR and R-MLPBR. For larger group sizes, the time between successive broadcast tree discoveries for MAODV is about 70%, 51% and 41% of that incurred for BEMRP in networks of low, moderate and high density respectively. Similarly, for larger group sizes, the time between successive broadcast tree discoveries for NR-MLPBR is about 76%, 64% and 57% of that incurred for BEMRP in networks of low, moderate and high density respectively. On the other hand, R-MLPBR tends to incur relatively larger time between successive tree discoveries even for larger multicast group sizes. For larger multicast groups, the time between successive tree discoveries for R-MLPBR is about 75%-80% of that incurred for BEMRP for all network densities.

4.3.2 Time between Successive Broadcast Tree Discoveries (Tree Discovery Strategy: DMEF)

- *Impact of Node Mobility*: For a given multicast group size, network density and multicast routing protocol, the time between successive broadcast tree discoveries at maximal node velocity of 30 m/s is roughly about 38-59% of that obtained at maximal node velocity of 10 m/s in networks of low, moderate and high density respectively. The time between successive broadcast tree discoveries at maximal node velocity of 50 m/s is roughly about 34-50% of that obtained at maximal node velocity of 10 m/s. In each instance, the increase in the time between successive route discoveries while using DMEF is at least 10-15% more than that obtained due to flooding.

- *Impact of Network Density*: As we increase the network density from 25 nodes to 50 nodes, we observe that the time between successive broadcast tree discoveries for MAODV, NR-MLPBR, R-MLPBR and BEMRP decreases by 13%, 9%, 6% and 6% respectively. On the other hand, as we increase from 25 nodes to 75 nodes, we notice that the larger number of nodes in the neighborhood is taken into account by DMEF to discover stable routes and there is no appreciable difference in the time between successive tree discoveries for NR-MLPBR, R-MLPBR and BEMRP. In the case of MAODV, the time between successive tree discoveries decreases by 8%.
- *Impact of Multicast Group Size*: For a given network density and node mobility, the time between successive route discoveries decreases as the multicast group size decreases. For smaller group sizes, the time between successive broadcast tree discoveries for MAODV and BEMRP is respectively about 82% and 87% of that incurred for NR-MLPBR and R-MLPBR. For moderate group sizes, the time between successive broadcast tree discoveries for MAODV, NR-MLPBR and BEMRP is about 77-86%, 96% and 96% of those incurred for R-MLPBR. For larger group sizes, the time between successive broadcast tree discoveries for MAODV and NR-MLPBR is about 80-89% and 92-94% of that obtained for R-MLPBR and BEMRP.

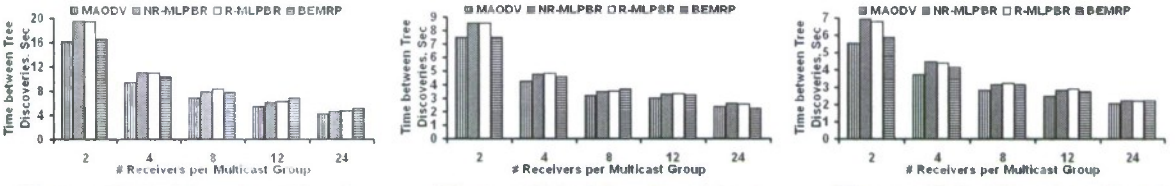


Figure 17.1: 25 nodes, 10 m/s **Figure 17.2:** 25 nodes, 30 m/s **Figure 17.3**: 25 nodes, 50 m/s

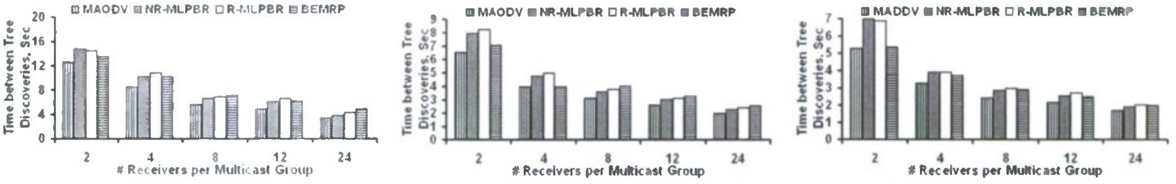


Figure 17.4: 50 nodes, 10 m/s **Figure 17.5:** 50 nodes, 30 m/s **Figure 17.6**: 50 nodes, 50 m/s

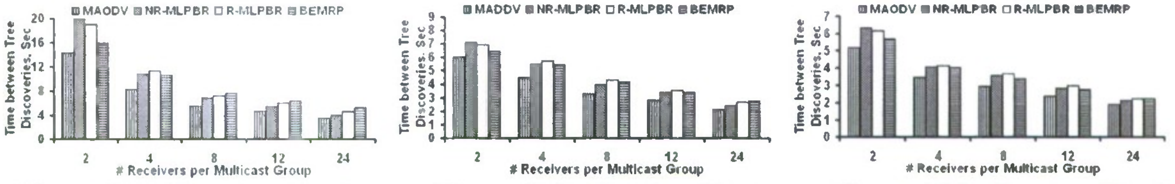


Figure 17.7: 75 nodes, 10 m/s **Figure 17.8:** 75 nodes, 30 m/s **Figure 17.9**: 75 nodes, 50 m/s

Figure 17: Average Time between Successive Tree Discoveries (Route Discovery Procedure: DMEF)

4.4 Energy Consumed per Node

Energy consumption in multicast routing is directly proportional to the number of links in the tree. Larger the number of links, more the transmissions and more will be the energy consumption in the network and vice-versa. The simulation results in Figures 18 and 19 clearly illustrate this. BEMRP incurs the least energy consumption per node and MAODV incurs the largest energy consumption per node. The energy consumed per node for the two multicast extensions is in between these two extremes. The energy consumed per node for R-MLPBR is less than that of NR-MLPBR as the former also attempts to simultaneously reduce the number of links as well as the hop count per source-receiver path. The energy consumption per node increases as the multicast group size increases. For a given multicast group size and multicast routing protocol, the energy consumed per node increases with increase in network density as well as with increase in node mobility.

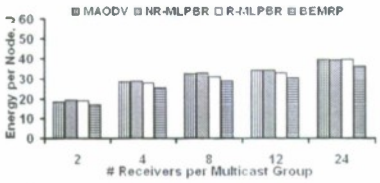


Figure 18.1: 25 nodes, 10 m/s

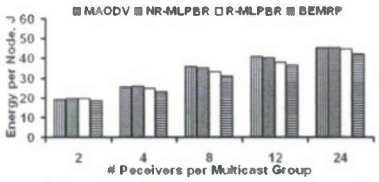


Figure 18.2: 25 nodes, 30 m/s

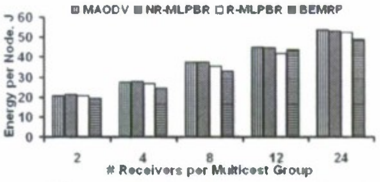


Figure 18.3: 25 nodes, 50 m/s

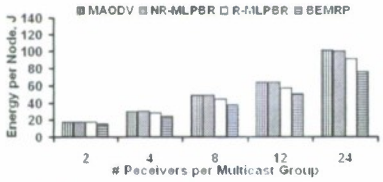


Figure 18.4: 50 nodes, 10 m/s

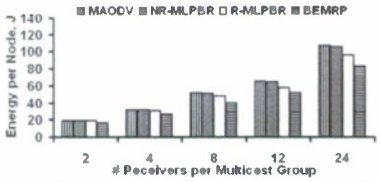


Figure 18.5: 50 nodes, 30 m/s

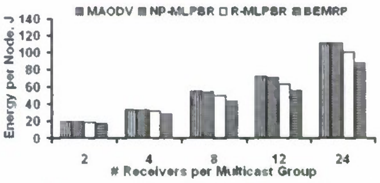


Figure 18.6: 50 nodes, 50 m/s

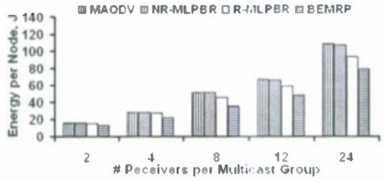


Figure 18.7: 75 nodes, 10 m/s

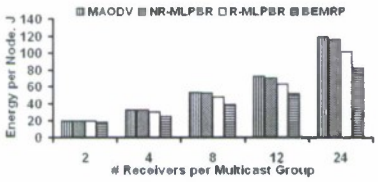


Figure 18.8: 75 nodes, 30 m/s

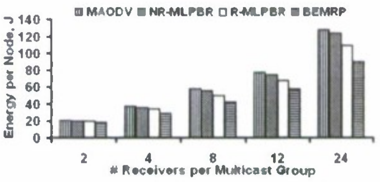


Figure 18.9: 75 nodes, 50 m/s

Figure 18: Average Energy Consumed per Node (Route Discovery Procedure: Flooding)

4.4.1 Energy Consumed per Node (Tree Discovery Strategy: Flooding)

- *Impact of Node Mobility*: For a given multicast group size, network density and multicast routing protocol, the energy consumed per node at maximal node velocity of 30 m/s can grow as large as 10-35% of that obtained at maximal node velocity of 10 m/s. The energy consumed per node at maximal node velocity of 50 m/s can grow as large as 10-40% of that obtained at maximal node velocity of 10 m/s. BEMRP and MAODV incur the largest increase in energy consumed per node with increase in node mobility. NR-MLPBR and R-MLPBR incur a relatively lower increase in the energy consumed per node with increase in node mobility. This can be attributed to the tendency of these multicast routing protocols to reduce the number of broadcast tree discoveries using effective tree prediction.
- *Impact of Network Density*: For multicast groups of size 2 and 4, we observe that with increase in network density from 25 to 50 nodes and from 25 to 75 nodes, the energy consumed per node decreases. This can be attributed to the smaller group size, leading to the effective sharing of the data forwarding load among all the nodes in the network. For larger group sizes, all the nodes in the network end up spending more energy (due to transmission/reception or at least receiving the packets in the neighborhood). As a result, for multicast group sizes of 8, 12 and 24, as we increase the network density from 25 nodes to 50 nodes, the increase in the energy consumed per node for MAODV, NR-MLPBR, R-MLPBR and BEMRP is by factors of 47%-134%, 46%-133%, 42%-122% and 30%-96% respectively. As we increase the network density from 25 nodes to 75 nodes, the increase in the energy consumed per node for MAODV, NR-MLPBR, R-MLPBR and BEMRP is by

factors of 52%-158%, 50%-154%, 42%-125% and 25%-100% respectively. MAODV and NR-MLPBR incur a relatively larger energy consumed per node at high network densities due to the nature of these multicast routing protocols to discover trees with minimum hop count. R-MLPBR and BEMRP discover trees with reduced number of links and hence incur relatively lower energy consumed per node at high network density.

- *Impact of Multicast Group Size*: As we increase the multicast group size from 2 to 24, the energy consumed per node for MAODV and NR-MLPBR increases by a factor of 2.1 to 2.6, 5.7 to 5.9 and 6.0 to 7.0 for low, medium and high density networks respectively. In the case of BEMRP and R-MLPBR, as we increase the multicast group size from 2 to 24, the energy consumed per node increases by a factor of 2.1 to 2.5, 4.9 to 5.2 and 4.6 to 6.2 in networks of low, medium and high density respectively. The increase in the energy consumed per node is below linear. Hence, all the four multicast routing protocols are scalable with respect to the increase in multicast group size.

4.4.2 Energy Consumed per Node (Tree Discovery Strategy: DMEF)

- *Impact of Node Mobility*: For a given multicast group size, network density and multicast routing protocol, the energy consumed per node at maximal node velocity of 30 m/s and 50 m/s can grow as large as 5-20% of that obtained at maximal node velocity of 10 m/s. This indicates the effectiveness of DMEF vis-à-vis flooding in reducing the energy consumed per node. DMEF discovers relatively more stable trees by involving only slow moving nodes in the tree. As a result, the multicast trees exist for a long time and incur less energy for tree discoveries. Similar to that observed for flooding, BEMRP and MAODV incur the largest increase in energy consumed per node with increase in node mobility. NR-MLPBR and R-MLPBR incur a relatively lower increase in the energy consumed per node with increase in node mobility.

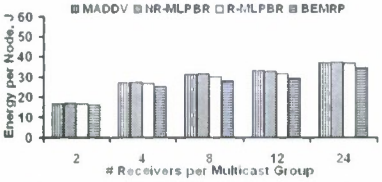


Figure 19.1: 25 nodes, 10 m/s

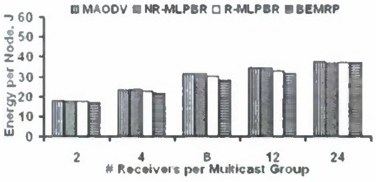


Figure 19.2: 25 nodes, 30 m/s

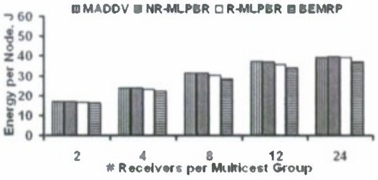


Figure 19.3: 25 nodes, 50 m/s

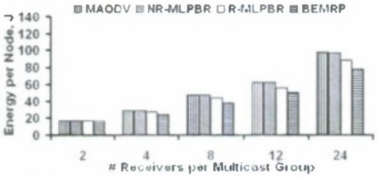


Figure 19.4: 50 nodes, 10 m/s

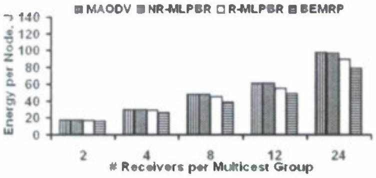


Figure 19.5: 50 nodes, 30 m/s

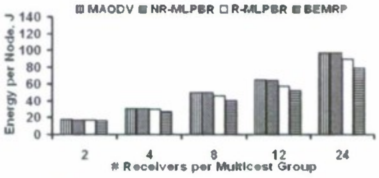


Figure 19.6: 50 nodes, 50 m/s

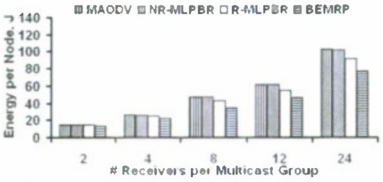


Figure 19.7: 75 nodes, 10 m/s

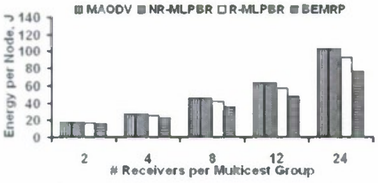


Figure 19.8: 75 nodes, 30 m/s

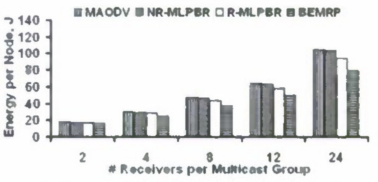


Figure 19.9: 75 nodes, 50 m/s

Figure 19: Average Energy Consumed per Node (Route Discovery Procedure: DMEF)

- *Impact of Network Density*: Similar to the observed for flooding, for multicast groups of size 2 and 4, we observe that with increase in network density from 25 to 50 nodes and from 25 to 75 nodes, the energy consumed per node decreases. For multicast group sizes of 8, 12 and 24, as we increase the network density from 25 nodes to 50 nodes, the increase in the energy consumed per node for MAODV, NR-MLPBR, R-MLPBR and BEMRP is by factors of 54%-157%, 53%-156%, 48%-136% and 38%-118% respectively. As we increase the network density from 25 nodes to 75 nodes, the increase in the energy consumed per node for MAODV, NR-MLPBR, R-MLPBR and BEMRP is by factors of 49%-173%, 47%-172%, 42%-146% and 27%-114% respectively. MAODV and NR-MLPBR incur a relatively larger energy consumed per node at high network densities due to the nature of these multicast routing protocols to discover trees with minimum hop count. R-MLPBR and BEMRP discover trees with reduced number of links and hence incur relatively lower energy consumed per node at high network density. We observe that for a given multicast routing protocol, for a given network density, the energy consumed per node due to flooding can be as large as 5%-16%, 12%-23% and 22%-37% more than that incurred using DMEF in the presence of low, medium and high node mobility respectively.
- *Impact of Multicast Group Size*: As we increase the multicast group size from 2 to 24, the energy consumed per node for MAODV and NR-MLPBR increases by a factor of 2.2 to 2.4, 5.6 to 5.8 and 6.0 to 7.1 for low, medium and high density networks respectively. In the case of BEMRP and R-MLPBR, as we increase the multicast group size from 2 to 24, the energy consumed per node increases by a factor of 2.2 to 2.4, 4.9 to 5.4 and 4.8 to 6.4 in networks of low, medium and high density respectively. The increase in the energy consumed per node is below linear. Hence, all the four multicast routing protocols are scalable with respect to the increase in multicast group size.

4.5 Energy Throughput

For each of the multicast routing protocols and for a given network density and node mobility, the energy throughput decreases with increase in the multicast group size. This can be attributed to the need to spend more energy to deliver a given multicast packet to more receivers vis-à-vis few receivers. For a given network density and multicast group size, the energy throughput of a multicast routing protocol decreases slightly as the node velocity is increased from low to moderate and high. For a given multicast group size and node mobility, the energy throughput of a multicast routing protocol decreases with increase in network density. This can be attributed to the involvement of several nodes (for larger network density) in distributing the offered traffic load to the multicast group. For a given simulation condition, the energy throughput of BEMRP is slightly larger than that of the other multicast routing protocols. This can be attributed to the lower energy consumed per node (and less number of links) for BEMRP.

4.5.1 Energy Throughput (Broadcast Tree Discovery Strategy: Flooding)

- *Impact of Node Mobility*: As we increase the node mobility from low to moderate and high, the energy throughput for a multicast routing protocol reduces as large as by 8%-12%, 12%-17% and 24%-26% in networks of low, moderate and high density respectively. For a given network density, the reduction in the energy throughput with increase in node mobility can be attributed to the relatively larger amount of energy spent for broadcast tree discoveries.
- *Impact of Network Density*: The decrease in energy throughput with increase in network density is more for MAODV and NR-MLPBR, relatively lower for R-MLPBR and is the least for BEMRP. At network density of 50 nodes, the energy throughput of MAODV and NR-MLPBR is 45%-64% and that of R-MLPBR and BEMRP is 50%-65% of that observed at network density of 25 nodes. At network density of 75 nodes, the energy through of MAODV, NR-MLPBR, R-MLPBR and BEMRP is 29%-48%, 30%-50%, 33%-50% and 38%-50% of that observed at network density of 25 nodes.

- *Impact of Multicast Group Size*: As the multicast group size is increased from 2 to 4, the energy throughput of the multicast routing protocols decreased by 30%-40%, 36%-40% and 24%-45% in networks of low, moderate and high density respectively. As the multicast group size is increased from 2 to 24, the energy throughput of the multicast routing protocols decreased by about 78%, 83% and 85% in networks of low, moderate and high density respectively.

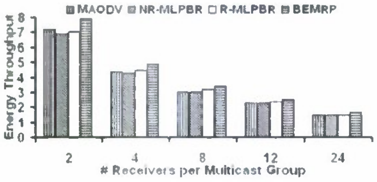


Figure 20.1: 25 nodes, 10 m/s

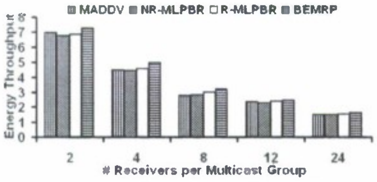


Figure 20.2: 25 nodes, 30 m/s

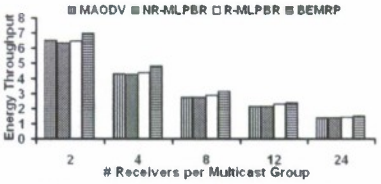


Figure 20.3: 25 nodes, 50 m/s

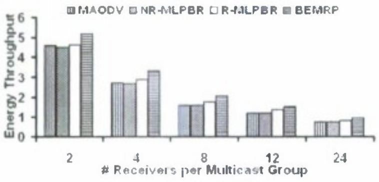


Figure 20.4: 50 nodes, 10 m/s

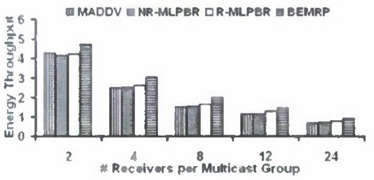


Figure 20.5: 50 nodes, 30 m/s

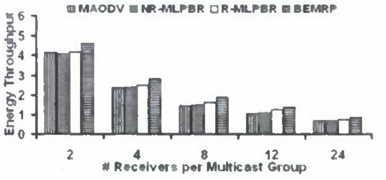


Figure 20.6: 50 nodes, 50 m/s

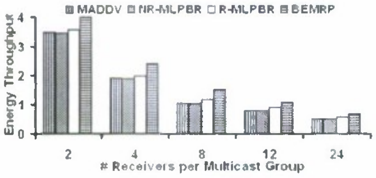


Figure 20.7: 75 nodes, 10 m/s

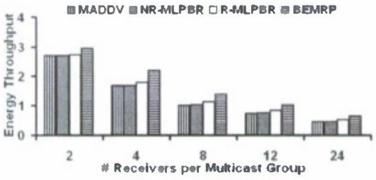


Figure 20.8: 75 nodes, 30 m/s

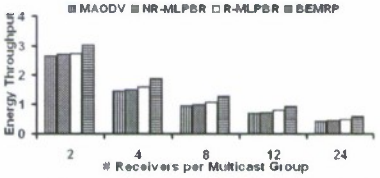


Figure 20.9: 75 nodes, 50 m/s

Figure 20: Energy Throughput: # Packets Delivered per Joule (Route Discovery Procedure: Flooding)

4.5.2 Energy Throughput (Broadcast Tree Discovery Strategy: DMEF)

- *Impact of Node Mobility*: As we increase the node mobility from low to moderate and high, the energy throughput for a multicast routing protocol reduces as large as by 7%-8%, 8%-12% and 16%-17% in networks of low, moderate and high density respectively. The relatively higher energy throughput while using DMEF can be attributed to the tendency of the broadcast strategy to involve only relatively slow moving nodes to be part of the trees. As a result, less energy consumed for broadcast tree discoveries.
- *Impact of Network Density*: The decrease in energy throughput with increase in network density is more for MAODV and NR-MLPBR, relatively lower for R-MLPBR and is the least for BEMRP. At network density of 50 nodes, the energy throughput of MAODV, NR-MLPBR, R-MLPBR and BEMRP is 48%-63%, 47%-63%, 52%-64% and 58%-69% of that observed at network density of 25 nodes. At network density of 75 nodes, the energy through of MAODV, NR-MLPBR, R-MLPBR and BEMRP is 32%-47%, 32%-48%, 36%-48% and 42%-50% of that observed at network density of 25 nodes.
- *Impact of Multicast Group Size*: As the multicast group size is increased from 2 to 4, the energy throughput of the multicast routing protocols decreased by 36%-44%, 35%-45% and 30%-47% in networks of low, moderate and high density respectively. As the multicast group size is increased

from 2 to 24, the energy throughput of the multicast routing protocols decreased by about 80%, 84% and 84% in networks of low, moderate and high density respectively.

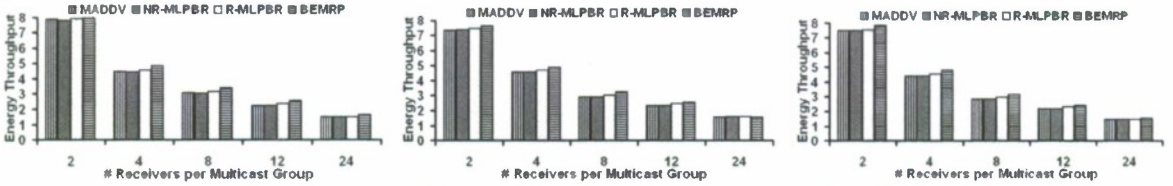


Figure 21.1: 25 nodes, 10 m/s **Figure 21.2:** 25 nodes, 30 m/s **Figure 21.3**: 25 nodes, 50 m/s

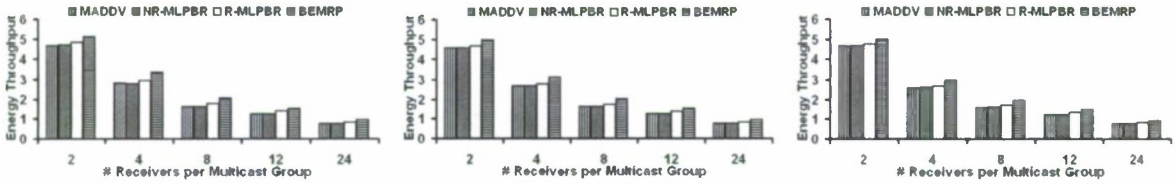


Figure 21.4: 50 nodes, 10 m/s **Figure 21.5:** 50 nodes, 30 m/s **Figure 21.6**: 50 nodes, 50 m/s

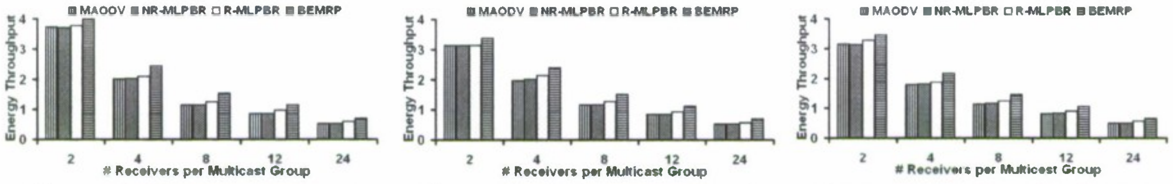


Figure 21.7: 75 nodes, 10 m/s **Figure 21.8:** 75 nodes, 30 m/s **Figure 21.9**: 75 nodes, 50 m/s

Figure 21: Energy Throughput: # Packets Delivered per Joule (Route Discovery Procedure: DMEF)

4.6 Energy Consumed per Tree Discovery

For a given broadcast strategy, the energy consumed per tree discovery is the same for all the four multicast routing protocols. For both flooding and DMEF, the energy consumed increases with increase in network density, attributed to the involvement of multiple nodes in the broadcast of the MTRMs. In low-density networks, the energy consumed per tree discovery using flooding is 10-22%, 19-35% and 14-20% more than that of the energy consumed per tree discovery using DMEF in low, moderate and high node mobility conditions respectively. In moderate density networks, the energy consumed per tree discovery using flooding is about 15%, 23% and 28% more than that of the energy consumed per tree discovery using DMEF in low, moderate and high node mobility conditions respectively. In high-density networks, the energy consumed per tree discovery using flooding is about 18%, 30% and 37% more than the energy consumed per tree discovery using DMEF respectively. As observed, DMEF performs better than flooding with increase in network density and/or node mobility.

For a given multicast group size, the energy consumed while using flooding in moderate (50 nodes) and high density (75 nodes) networks is respectively about 3.8 and 8 times more than that incurred in networks of low density. This indicates that as the number of nodes is increased by *x* times ($x = 2$ for moderate density and $x = 3$ for high density), the energy consumed due to flooding increases by 2^x times. In the case of DMEF, for a given multicast group size, the energy consumed in moderate density networks is about 3.7, 3.5 and 3.2 times more than that observed in low density networks for low, moderate and high node mobility conditions respectively. For a given multicast group size, the energy consumed during DMEF in high-density networks is about 7.8, 7.2 and 6.6 times more than that observed in low-density networks for low, moderate and high node mobility conditions respectively. Thus, the energy consumed while using DMEF does not increase exponentially as observed for flooding. DMEF performs appreciably well in lowering the energy consumed per tree discovery with increase in node mobility and/or increase in network density.

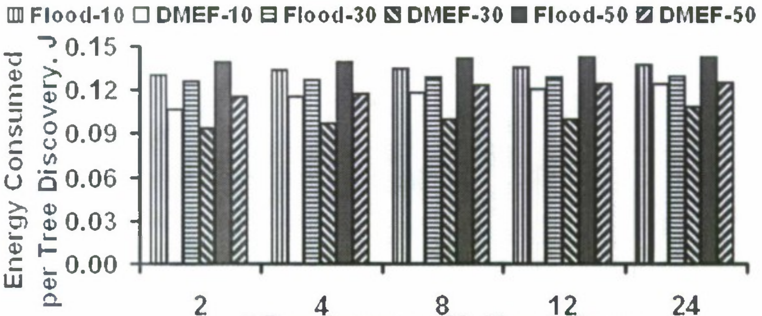


Figure 22: Energy Consumed per Broadcast Tree Discovery: Flooding vs. DMEF (25 Nodes)

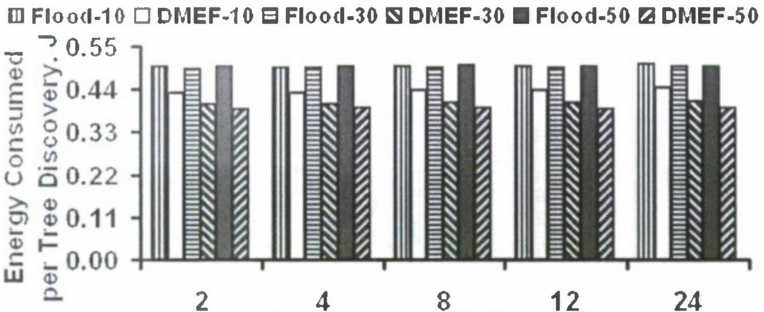


Figure 23: Energy Consumed per Broadcast Tree Discovery: Flooding vs. DMEF (50 Nodes)

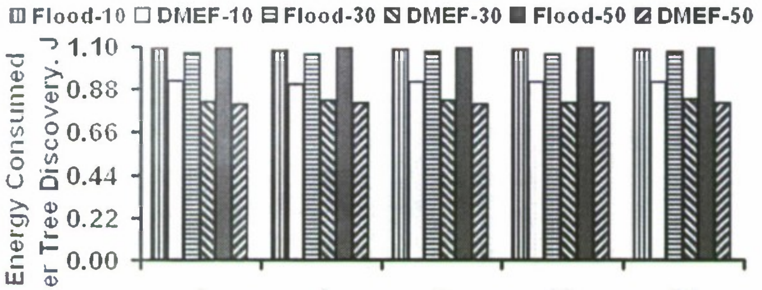


Figure 24: Energy Consumed per Broadcast Tree Discovery: Flooding vs. DMEF (75 Nodes)

III. Summary of Accomplishments in Research Activity 2

This research work contributed to the design and development of the multicast extensions to the location prediction based routing (LPBR) protocol for mobile ad hoc networks (MANETs). LPBR has been proposed to simultaneously minimize the number of route discoveries as well as the hop count of the paths for unicast routing in MANETs. The multicast extensions of LPBR (referred to as NR-MLPBR and R-MLPBR) have been proposed to simultaneously reduce the number of tree discoveries and the hop

References

[1] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu and J. Jetcheva, "A Performance of Comparison of Multi-hop Wireless Ad hoc Network Routing Protocols," in *Proceedings of the 4th Annual ACM/IEEE Conference on Mobile Computing and Networking*, pp. 85 – 97, October 1998.

[2] C. K. Toh, G. Guichal and S. Bunchua, "ABAM: On-demand Associatvity-based Multicast Routing for Ad hoc Mobile Networks," in *Proceedings of the 52nd IEEE VTS Fall Vehicular Technology Conference*, Vol. 3, pp. 987 – 993, September 2000.

[3] C. S. R. Murthy and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols," Prentice Hall, June 3, 2004.

[4] T. Ozaki, J-B. Kim and T. Suda, "Bandwidth-Efficient Multicast Routing for Multihop, Ad hoc Wireless Networks," *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies* (INFOCOM), vol. 2, pp. 1182-1192, Anchorage, USA, April 2001.

[5] E. Royer and C. E. Perkins, "Multicast Operation of the Ad-hoc On-demand Distance Vector Routing Protocol," *Proceedings of the 5th ACM/IEEE Annual Conference on Mobile Computing and Networking*, (MOBICOM), pp. 207-218, Seattle, USA, August 1999.

[6] C-K. Toh, G. Guichal and S. Bunchua, "ABAM: On-Demand Associativity-Based Multicast Routing for Ad hoc Mobile Networks," *Proceedings of the 52nd IEEE Vehicular Technology Conference* (Fall-VTC), vol. 3, pp. 987-993, September 2000.

[7] C. Chiang, M. Gerla and L. Zhang, "Shared Tree Wireless Network Multicast," *Proceedings of the 6th International Conference on Computer Communications*, pp. 28-33, 1997.

[8] C. W. Wu and Y. C. Tay, "AMRIS: A Multicast Protocol for Ad hoc Wireless Networks," *Proceedings of the IEEE Military Communications Conference* (MILCOM), pp. 25-29, 1999.

[9] J. Xie, R. R. Talpade, A. McAuley and M. Liu, "AMRoute: Ad Hoc Multicast Routing Protocol," *Mobile Networks and Applications*, vol. 7, no. 6, pp. 429-439, December 2002.

[10] S. Bae, S. Lee, W. Su and M. Gerla, "The Design, Implementation and Performance Evaluation of the On-Demand Multicast Routing Protocol in Multi-hop Wireless Networks," *IEEE Networks*, vol. 4, no. 1, pp. 70-77, 2000.

[11] S. Lee and C. Kim, "Neighbor Supporting Ad hoc Multicast Routing Protocol," *Proceedings of the 1st Annual ACM Workshop on Mobile and Ad hoc Networking and Computing* (MOBIHOC), pp. 37-44, Boston, August 2000.

[12] J. J. Garcia-Luna-Aceves and E. L. Madruga, "The Core-Assisted Mesh Protocol," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1380-1394, 1999.

[13] H. Zhou and S. Singh, "Content-Based Multicast (CBM) in Ad hoc Networks," *Proceedings of the 1st Annual ACM Workshop on Mobile and Ad hoc Networking and Computing* (MOBIHOC), pp. 51-60, Boston, August 2000.

[14] Y. Ko and N. H. Vaidya, "Location-Based Multicast in Mobile Ad Hoc Networks," Technical Report TR98-018, Texas A&M University, College Station, TX, USA, 1998.

[15] N. Meghanathan, "Stability and Link Efficiency of On-Demand Source Tree-Based Routing Protocols for Mobile Ad Hoc Networks," *Proceedings of the International Conference on High Performance Computing, Networking and Communication Systems*, pp. 159-166, July 2008.

[16] N. Meghanathan, "Location Prediction Based Routing Protocol for Mobile Ad Hoc Networks," *Proceedings of the IEEE Global Communications Conference* (GLOBECOM), New Orleans, Nov-Dec 2008.

[17] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms," 2nd Edition, MIT Press/ McGraw Hill, Sept. 2001.

[18] N. Meghanathan, "A Density and Mobility Aware Energy-Efficient Broadcast Strategy to Determine Stable Routes in Mobile Ad hoc Networks," accepted for publication in the *Proceedings of the International Conference on Wireless Networks*, Las Vegas, NV, July 13-16, 2009.

[19] J.-P. Sheu, C.-M. Chao, W.-K. Hu and C.-W. Sun, "A Clock Synchronization Algorithm for Multi-hop Wireless Ad hoc Networks," Wireless Personal Communications: An International Journal, vol. 43, no. 2, pp. 185-200, 2007.

[20] D. Zhou, "A Compatible and Scalable Clock Synchronization Protocol in IEEE 802.11 Ad hoc Networks," Proceedings of the International Conference on Parallel Processing, pp. 295-302, Washington DC, USA, June 2005.

[21] L. Breslau, et. al., "Advances in Network Simulation," *IEEE Computer*, vol. 33, no. 5, pp. 59-67, 2000.

[22] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal of Selected Areas in Communication*, vol. 18, no. 3, pp. 535-547, 2000.

[23] C. Bettstetter, H. Hartenstein and X. Perez-Costa, "Stochastic Properties of the Random-Waypoint Mobility Model," *Wireless Networks*, vol. 10, no. 5, pp. 555-567, 2004.

[24] C. E. Perkins and E. M. Royer, "The Ad hoc On-demand Distance Vector Protocol," Ad hoc Networking, edited by C. E. Perkins, pp. 173-219, Addison-Wesley, 2000.

Research Activity – 3

A Node-disjoint Multi-path Version of the Location Prediction Based Routing Protocol (LPBR-M)

Dr. Natarajan Meghanathan
Assistant Professor
Department of Computer Science
Jackson State University
Jackson, MS 39217
Email: natarajan.meghanathan@jsums.edu
Phone: 601-979-3661

I. Breakdown of the Research Activity to Tasks

Task No.	Task	Current Status	Timeline
1	Study the related work on multicast routing protocols for mobile ad hoc networks (MANETs)	Completed	December 2008 to March 2009
2	Develop the Multicast Extensions to LPBR (NR-MLPBR and R-MLPBR)	Completed	April 2009
3	Conduct simulations of MLPBR and compare its performance with some of the currently existing MANET multicast routing protocols	Completed	May 2009 to June 2009
4	Analyze the simulation results with respect to different performance metrics	Completed	June 2009 to July 2009

II. Description of the Tasks

Task 1: Study the Related Work on Multi-path Routing Protocols for Mobile Ad hoc Networks

On-demand routing protocols incur high route discovery latency and also incur frequent route discoveries in the presence of a dynamically changing topology. Recent research has started to focus on multi-path routing protocols for fault tolerance and load balancing. Multi-path on-demand routing protocols tend to compute multiple paths, at both the traffic sources as well as at intermediary nodes, in a single route discovery attempt. This reduces both the route discovery latency and the control overheads as a route discovery is needed only when all the discovered paths fail. Spreading the traffic along several routes could alleviate congestion and bottlenecks. Multi-path routing also provides a higher aggregate bandwidth and effective load balancing as the data forwarding load can be distributed over all the paths.

Multi-paths can be of two types: link-disjoint and node-disjoint. For a given source *s* and destination *d*, the set of link-disjoint *s-d* routes comprises of paths that have no link present in more than one constituent *s-d* path. Similarly, the set of node-disjoint *s-d* routes comprises of paths that have no node (other than the source and destination) present in more than one constituent *s-d* path. Multi-path routing protocols proposed for ad hoc networks make use of the propagation of the Route-Request (RREQ) messages along several paths to the destination and let the destination to send Route-Reply (RREP) along more than one path. The routing protocols avoid the RREP storm by selecting only few of the different paths. Since nodes communicate through the shared wireless medium, the selected paths need to be as independent as possible in order to avoid transmissions from a node along one path interfering with transmissions on a different path. The aggregate bandwidth achieved with multi-path routing may not be the sum of the bandwidth of the individual paths. Metrics such as correlation factor and coupling factor are used to calculate the relative degree of independence among the multiple paths [1]. The correlation factor, measured only for node-disjoint paths, indicates the number of links connecting two node-disjoint paths. The coupling factor, measured for both node-disjoint and link-disjoint paths, is defined as the average number of nodes that are blocked from receiving data on one of the paths when a node in the other path is transmitting. Node-disjoint routes offer the highest degree of fault tolerance and aggregate bandwidth.

In [2], the authors advocate the need to consider similarity among the multiple *s-d* paths with that of the shortest *s-d* path and stress the need to use similar paths for multi-path data propagation. Routing using multiple paths similar to the shortest path will reduce the chances of out-of-order packet delivery and also result in lower end-to-end delay per packet. The authors in [3] develop an analytical model for evaluating the effectiveness of multi-path routing. They show that unless we use a very large number of paths, the load distribution with multi-path routing is almost the same as in single path routing.

Most of the multi-path routing protocols proposed in the literature are either extensions of the Dynamic Source Routing (DSR) protocol [4] or the Ad hoc On-demand Distance Vector (AODV) routing protocol [5]. The multi-path routing protocols that are currently being reviewed include: (i) Split multi-path routing (SMR) [6] protocol, an extension of DSR; (ii) Ad hoc On-demand Multi-path Distance Vector (AOMDV) routing protocol [7], an extension of AODV to compute multiple loop-free link-disjoint routes; (iii) AODV-Multi-path (AODVM) routing protocol [8], an extension of the AODV protocol to determine node-disjoint routes; (iv) Geographic Multi-path Routing Protocol (GMRP) [9] proposed to reduce interference due to route coupling and (v) Energy-aware Multi-path Routing Protocol (EMRP) [10] that considers the available energy and the forwarding load at the intermediate nodes of the multiple paths before distributing the load across them.

1.1 Split Multi-path Routing Protocol

In Split multi-path routing (SMR) [6], the intermediate nodes forward RREQs that are received along a different link and with a hop count that is not larger than the first received RREQ. The destination selects the route on which it received the first RREQ packet (which will be a shortest delay path), and then waits to receive more RREQs. The destination node then selects the path which is maximally disjoint from the shortest delay path. If more than one maximally disjoint path exists, the tie is broken by choosing the path with the shortest hop count.

1.2 Ad hoc On-demand Multi-path Distance Vector (AOMDV) Routing Protocol

The Ad hoc On-demand Multi-path Distance Vector (AOMDV) routing protocol [7] is an extension of AODV to compute multiple loop-free link-disjoint routes. The RREQs that arrive via different neighbors of the source node define the maximum number of node-disjoint/link-disjoint paths that are possible. For every destination node *d*, an intermediate node *i* maintains the list of next hop nodes, the hop count for the different paths to the destination node *d* and the "advertised hop count"(the maximum hop count for all paths from *i* to *d*), with respect to the latest known sequence number for *d*. An intermediate node accepts and forwards a route advertisement as an alternate path to the destination only if the route advertisement came from a neighbor node that has not yet sent the route advertisement for the destination sequence number and the hop count in the route advertisement is less than the advertised hop count to the destination. When a node receives a route advertisement for the destination with a greater sequence number, the next hop list and the advertised hop count values are reinitialized. The destination node replies for the RREQs arriving from unique neighbors. A multi-path routing scheme that extends AOMDV by using a traffic-path allocation scheme has been proposed in [11] and it is based on cross-layer measurements of path statistics that reflects the queue size and congestion level of each path. The proposed scheme also utilizes the Fast Forward (FF) MAC forwarding mechanism [12] to reduce the effects of self-contention among frames at the MAC layer.

1.3 AODV-Multi-path (AODVM) Routing Protocol

The AODV-Multi-path (AODVM) routing protocol [8] is an extension of the AODV protocol to determine node-disjoint routes. An intermediate node does not discard duplicate RREQ packets and records them in a RREQ table. The destination responds with an RREP for each RREQ packet received. An intermediate node on receiving the RREP, checks its RREQ table and forwards the packet to the neighbor that lies on the shortest path to the source. The neighbor entry is then removed from the RREQ table. Also, whenever a node hears a neighbor node forwarding the RREP packet, the node removes the entry for the neighbor node in its RREQ table.

1.4 Geographic Multi-path Routing Protocol

The Geographic Multi-path Routing Protocol (GMRP) [9] has been proposed to reduce interference due to route coupling. The RREQ will have information regarding the locations of the first hop and the last hop intermediate nodes on the path. The destination chooses the path through which it first received the RREQ. For a subsequently received RREQ, the destination measures the distance between the first hops of the path traversed by this RREQ and the already selected paths and also the distance between the last hops of the path traversed by this RREQ and the already selected paths. If both these distances are greater than twice the transmission range of the nodes, the path traversed by the received RREQ is selected.

1.5 Energy-aware Multi-path Routing Protocol

EMRP is an energy-aware multi-path routing protocol [10] that considers the available energy and the forwarding load at the intermediate nodes of the multiple paths before distributing the load across them. The destination node replies with a RREP packet for each RREQ packet. An intermediate node receiving the RREP packet updates information regarding the distance between the node and the next hop node, the number of retransmission attempts corresponding to the last successful transmission, the current queue length, the current remaining energy of the node. The source node then computes a weight for each route through which the RREP traversed. Routes with minimum weight are preferred as such routes have more remaining energy, less energy consumption due to transmission and reception, less crowded channel in the neighborhood of the nodes in the path and more bandwidth available.

Task 2: Develop Algorithm for the Node-Disjoint Multi-path Version of LPBR (LPBR-M)

The Location Prediction Based Routing (LPBR) protocol [15] was recently published by the PI to simultaneously minimize the number of route discoveries as well as the hop count of the paths for unicast routing in mobile ad hoc networks (MANETs). In this research activity, we develop the multi-path version of the LPBR protocol (referred here after as LPBR-M) to determine a set of node-disjoint routes between the source and destination nodes in a MANET. When one of the paths in the set of node-disjoint routes fails, LPBR-M would explore the use of the Location Update Vectors (LUVs) to predict the current locations of the nodes and determine a new set of node-disjoint paths. The destination then notifies the source node of the new set of node-disjoint routes through LPBR-M-Route-Reply packets sent along those new routes. We opt for node-disjoint multi-path routing vis-à-vis link-disjoint multi-path routing because of an observation in one of the PI's recent work [13] that for different conditions of network density and node mobility, the number of broadcast route discoveries needed for node-disjoint multi-path routing is not significantly different from the number of route discoveries for link-disjoint multi-path routing. Also, there is no much difference in the average hop count of the node-disjoint paths and the link-disjoint paths.

2.1 Basic Idea of the Multi-path Extension of LPBR (LPBR-M)

The multi-path extension of LPBR works as follows: When a source attempts to send data to the destination and does not know any path to reach the latter, the source broadcasts a Multi-path Route Request (MP-RREQ) message throughout the network. Any broadcast algorithm (for example: flooding or DMEF [14]) can be used for this purpose. The location and mobility information of the intermediate forwarding nodes are recorded in the MP-RREQ messages as a sequence of Location Update Vectors (LUVs) [15]. The destination node receives several MP-RREQs and runs a local node-disjoint path selection algorithm to identify the set of node-disjoint paths, ordered in the increasing order of their hop count. The destination sends out the Multi-path Route Reply (MP-RREP) messages to the source along each of the node-disjoint paths selected. The source receives the MP-RREPs and stores the set of node-disjoint paths (NDP-Set) in its local cache.

For data propagation, the source uses the minimum-hop path in the NDP-Set discovered and continues to use the path until it exists. If an intermediate node could not forward a data packet, it sends a MP-RERR message back to the source. When the source receives the MP-RERR message, it removes the failed path from the NDP-Set and sends the data packet on the next minimum-hop path in the NDP-Set. This procedure is repeated until the source no longer receives a MP-RERR message from an intermediate node or until the NDP-Set is exhausted. In the latter case, the source does not immediately opt for a broadcast discovery procedure. The source waits for the destination to predict a new set of node-disjoint paths based on the LUVs collected in the latest broadcast discovery procedure.

The destination predicts the current location of the nodes and locally constructs a predicted global graph. The node-disjoint path selection heuristic [13] is run on this graph and a set of predicted node-disjoint paths is determined. The destination sends a sequence of MP-LPBR-RREP messages to the source along each of these predicted paths. If a predicted path does not exist, an intermediate node (on the predicted path) cannot forward the MP-LPBR-RREP message further towards the source and instead sends a MP-LPBR-RERR message back to the destination. If the destination receives MP-LPBR-RREP-RERR messages for all the MP-LPBR-RREP messages sent, it discards the LUVs and waits for the source to initiate a new broadcast discovery procedure. If the destination does not receive the MP-LPBR-RREP-RERR message for a particular MP-LPBR-RREP message, it means the corresponding predicted path does actually exist at the current time. If the source receives at least one MP-LPBR-RREP message, it stores them the corresponding path in its NDP-Set. For data propagation, the source follows the same procedure of using the paths in its updated NDP-Set in the increasing order of their hop counts. If the source does not receive even one MP-LPBR-RREP message within a certain timeout period, the source then initiates a new broadcast discovery procedure.

2.2 Objectives and Assumptions

The objective of the multi-path extension to LPBR (LPBR-M) is to simultaneously minimize the number of multi-path broadcast discoveries as well as the hop count of the source-destination path. If the broadcast discovery procedure used is the recently proposed Density and Mobility-aware Energy Efficient (DMEF) strategy, we assume the periodic exchange of beacons in the neighborhood of each node at a frequency determined from a time period uniformly and randomly selected from [0…5 seconds]. We also assume that the clocks across all nodes are synchronized. This is essential to ensure proper timeouts at the nodes for failure to receive a certain control message.

2.3 Broadcast of Multi-path Route Request (MP-RREQ) Messages

Whenever a source node has data packets to send to a destination and is not aware of any path to the latter, the source initiates a broadcast route discovery procedure by broadcasting a Multi-path Route Request (MP-RREQ) message to its neighbors. Any broadcast route discovery procedure (e.g., flooding or DMEF) can be used for this purpose. The source maintains a monotonically increasing sequence number for the broadcast route discoveries it initiates to find the node-disjoint multi-paths. Each node, except the destination, on receiving the first MP-RREQ of the current broadcast process (i.e., a MP-RREQ with a sequence number greater than those seen before), includes its Location Update Vector, LUV, in the MP-RREQ message. The LUV of a node comprises the following: node ID, X, Y co-ordinate information, Current velocity and Angle of movement with respect to the X-axis. The node ID is also appended on the "Route Record" field of the MP-RREQ message. The structure of the LUV and the MP-RREQ message is shown in Figures 1 and 2 respectively. Note that upon receiving a MP-RREQ message, we do not let an intermediate node to immediately generate a MP-RREP message to the source, even though the intermediate node might know of one or more routes to the destination. We intentionally do this so that we could collect the latest LUVs of each node in the network through the MP-RREQ messages and also able to determine the set of valid of node-disjoint paths that really exist at the time of the broadcast multi-path route discovery process.

Node ID	X Co-ordinate	Y Co-ordinate	Node Velocity	Angle of Movement
4 bytes	8 bytes	8 bytes	8 bytes	8 bytes

Figure 1: Location Update Vector (LUV) Collected from Each Node

Source ID	Destination ID	Sequence Number	Route Recorded (List of Node IDs)	Location Update Vectors (LUVs)
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes	Variable Size of 36 bytes

Figure 2: Structure of the Multi-path Route Request (MP-RREQ) Message

2.4 Determination of the Set of Node-Disjoint Paths using the MP-RREQ Messages

When a destination receives a MP-RREQ message, it extracts the path traversed by the message (sequence of Node IDs in the Route Record) and the LUVs of the source and the intermediate nodes that forwarded the message. The destination stores the path information in a set, *RREQ-Path-Set*, maintained for every source with which the destination is in communication. The paths in the *RREQ-Path-Set* are stored in the increasing order of their hop count. Ties between paths with the same hop count are broken in the order of their time of arrival at the destination node. The LUVs are stored in the LUV-Database maintained for the latest broadcast route discovery procedure initiated by the source. The destination runs a local path selection heuristic to extract the set of node-disjoint paths from the *RREQ-Path-Set*. The heuristic makes sure that in the set of node-disjoint paths, except the source and the destination nodes, a node can serve as an intermediate node in at most only one path. A *RREQ-ND-Set* (set of Node-Disjoint paths) is initialized and updated with the paths extracted from the *RREQ-Path-Set* satisfying this criterion.

Input: *RREQ-Path-Set* // set of paths traversed by the MP-RREQ messages received

Output: *RREQ-ND-Set* // set of node-disjoint paths to be extracted from the *RREQ-Path-Set*

Initialization: *RREQ-ND-Set* ← Φ

Auxiliary Variables: *candidatePath* // used to store information whether a path extracted from *RREQ-Path-Set* can be added to *RREQ-ND-Set* or not

```
Begin RREQ-ND-Path-Selection
while (RREQ-Path-Set $\neq \Phi$) do
```

```
        Extract the first path P in RREQ-Path-Set // basically removes the path P from RREQ-Path-Set
        candidatePath ← True
        for (every intermediate node u ∈ P) do
            for (every node-disjoint path ND-P in RREQ-ND-Set) do
                if (u is an intermediate node of ND-P) then
                    candidatePath ← False
                end if
            end for
        end for
        if (candidatePath is set to True) then
            RREQ-ND-Set ← RREQ-ND-Set U {P}
        end if
    end while
    return RREQ-ND-Set
End RREQ-ND-Path-Selection
```

Figure 3: Heuristic to Extract Node-Disjoint Paths from the MP-RREQ Messages Received

The heuristic (illustrated in Figure 3) traverses through the *RREQ-Path-Set* in the order of the paths stored in it (in the increasing order of the hop counts). A path *P* in the *RREQ-Path-Set* is added to the *RREQ-ND-Set* only if none of the intermediate nodes in *P* are already part of any of the paths in the *RREQ-ND-Set*. Once the *RREQ-ND-Set* is formed, the destination sends a Multi-path Route Reply (MP-RREP) message for every path in the *RREQ-ND-Set*. The structure of the MP-RREP message is shown in Figure 4. An intermediate node receiving the MP-RREP message updates its routing table by adding the neighbor that sent the message as the next hop on the path from the source to the destination. The MP-RREP message is then forwarded to the next node towards the source as indicated in the Route Record field of the message.

Originating Source ID of the MP-RREQ	Targeted Destination ID of the MP-RREQ	Sequence Number of the MP-RREQ	Route Recorded in the MP-RREQ (List of Node IDs)
4 bytes	4 bytes	4 bytes	Variable Size Multiples of 4 bytes

Figure 4: Structure of the MP-RREP Message

2.5 Multi-path Acquisition Time and Data Transmission

After receiving the MP-RREP messages from the destination within a certain time called the Multi-path Acquisition Time (*MP-AT*), the source stores the paths learnt in a set of node-disjoint paths, *NDP-Set*. The *MP-AT* is based on the maximum possible diameter of the network (an input parameter in our simulations). The diameter of the network is the maximum of the hop count of the minimum hop paths between any two nodes in the network. The *MP-AT* is dynamically set at a node depending on the time it took to receive the first MP-RREP for a broadcast discovery process. If *pktOriginInterval* denotes the time between the transmission of successive packets from the source, *delFirstRREQRecvd* indicates the time lapsed between the initiation of the MP-RREQ broadcast and the receipt of the first MP-RREP and *hopsFirstRREQRecvd* denotes the number of hops traversed by the first MP-RREP received, then,

$$MP-AT = Minimum\left[pktOriginInterval, \left(\frac{delFirstRREQ\,\mathrm{Re}cvd * Diameter}{hopsFirstRREQ\,\mathrm{Re}cvd} \right) \right]$$

Source ID	Destination ID	Sequence Number	Number of Disjoint Paths	More Packets	Current Dispatch Time	Time Left for Next Dispatch
4 bytes	4 bytes	4 bytes	1 byte	1 bit	8 bytes	4 bytes

Figure 5: Structure of the Data Packet

When the source begins to start propagating the data packets using the newly formed *NDP-Set*, the source uses the path with the minimum hop count among the paths in the *NDP-Set*. The structure of a data packet is illustrated in Figure 5. The sequence number field in the header can be used by the destination to accumulate and reorder the data packets, incase if they are received out of order. In addition to these regular fields, the header of the data packet includes four specialized fields: the 'Number of Disjoint Paths (*NDP-Set Size*)' field that indicates the number of active node-disjoint paths currently being stored in the Node-Disjoint Path Set of the source, the 'More Packets' (*MP*) field, the 'Current Dispatch Time' (*CDT*) field and the 'Time Left for Next Dispatch' (*TNLD*) field. The *CDT* field stores the time as the number of milliseconds lapsed since Jan 1, 1970, 12 AM. These additional overhead (relative to that of the other ad hoc multicast routing protocols) associated with the header of each data packet amounts to only 13 more bytes per data packet.

The source sets the *CDT* field in all the data packets sent. In addition, if the source has any more data to send, it sets the *MP* flag to 1 and sets the appropriate value for the *TLND* field (equal to *pktOriginInterval)*, which indicates the number of milliseconds since the *CDT*. If the source does not have any more data to send, it will set the *MP* flag to 0 and leaves the *TLND* field blank. As we assume the clocks across all nodes are synchronized, the destination node will be able to calculate the end-to-end delay for the data packet based on the time the data packet reaches the node and the *CDT* field in the header of the data packet. Several clock synchronization algorithms (example [16][17]) have been proposed for wireless ad hoc networks. The destination node computes and maintains the average end-to-and delay per data packet for the current path to the source by recording the sum of the end-to-end delays of all the data packets received so far on the path and the number of data packets received on the path. Accordingly, the average end-to-end delay per data packet for the current path is updated every time after receiving a new data packet on the path. If the source node has set the MP flag, the destination node computes the 'Next Expected Packet Arrival Time' (*NEPAT*), which is *CDT* field + *TLND* field + 2**NDP-Set Size***Average end-to-end delay per data packet*. A timer is started for the *NEPAT* value. Since, we are using only the average end-to-end delay per data packet to measure the *NEPAT* value, the variations in the end-to-end delay of particular data packets will not very much affect the *NEPAT* value. So, the source and destination nodes need not be perfectly synchronized. The clocks across the nodes can have small drifts and this would not very much affect the performance of LPBR-M.

2.6 Multi-path Maintenance

If a link failure occurs due to the two nodes constituting the link drifting away, the upstream node of the broken link (learnt through the failure to successfully transmit the data packet at the link layer) informs about the broken route to the source node through a Multi-path-Route-Error (MP-RERR) message, structure shown in Figure 6. The source node on learning the route failure will remove the failed path from its *NDP-Set* and attempt to send data packet on the next minimum-hop path in the *NDP-Set*. If this path is actually available in the network at that time instant, the data packet will successfully propagate its way to the destination. Otherwise, the source receives a MP-RERR message on the broken path, removes the failed path from the *NDP-Set* and attempts to route the data packet on the next minimum hop path in the *NDP-Set*. This procedure is repeated until the source does not receive a MP-RERR message or runs out of an available path in the *NDP-Set*. In the former case, the data packet successfully reaches the destination and the source continues to transmit the next data packet at the next scheduled time. In the latter case, the source is not able to successfully transmit the data packet to the destination.

Node originating the MP-RERR message	Source ID of the Data packet dropped	Destination ID of the Data packet dropped	Sequence Number of the Data packet dropped	Downstream Node with which the link failed
4 bytes	4 bytes	4 bytes	4 bytes	4 bytes

Figure 6: Structure of the MP-RERR Message

Before initiating another broadcast route discovery procedure, the source will wait for the destination node to inform it of a new set of node-disjoint routes through a sequence of MP-LPBR-RREP messages. The source will run a MP-LPBR-RREP-timer and wait to receive at least one MP-LPBR-RREP message from the destination. For the failure of the first set of node-disjoint paths, the value of this timer would be a variable parameter within the simulations. In this research work, we will be simulating with constant-bit rate (CBR) traffic and so the MP-LPBR-RREP-timer will be set to the route acquisition time (the time it took to get the first MP-RREP message from the destination since the inception of the route discovery), so that we give sufficient time for the destination to learn about the route failure and generate a new sequence of MP-LPBR-RREP messages. For subsequent route-repairs, the MP-LPBR-RREP-timer will be set based on the time it takes to get the first MP-LPBR-RREP message from the destination.

2.7 Prediction of Node Location using the Location Update Vector

If the destination node does not receive the data packet within the *NEPAT* time, it will attempt to locally construct the global topology using the location and mobility information of the nodes learnt from the latest broadcast route discovery. Each node is assumed to be continuing to move in the same direction with the same speed as mentioned in its latest LUV. Based on this assumption and information from the latest LUVs, the location of each node at the *NEPAT* time is predicted. Whenever a node changes its direction, we assume the node is moving in the new direction with a particular velocity and towards a particular targeted destination location. As a result, a node can determine its angle of movement with respect to the X-axis at time *STIME* by computing the slope of the line joining the current location co-ordinates of the node at time *STIME* and the co-ordinates of the targeted location to which the node is moving. After reaching the targeted location, a node can change its velocity and direction to move to a new destination location.

We now explain how to predict the location of a node (say node u) at a time instant *CTIME* based on the LUV gathered from node u at time *STIME*. Let $(X_u^{STIME}, Y_u^{STIME})$ be the X and Y co-ordinates of node u at time *STIME*. Let $Angle_u^{STIME}$ and $Velocity_u^{STIME}$ represent the angle of movement with respect to the X-axis and the velocity at which node u is moving. The distance traveled by node u from time *STIME* to *CTIME* would be: $Distance_u^{STIME\text{-}CTIME} = (CTIME - STIME + 1) * Velocity_u^{STIME}$.

Let $(X_u^{CTIME}, Y_u^{CTIME})$ be the predicted location of node u at time *CTIME*. The value of X_u^{CTIME} is given by $X_u^{STIME} + Offset\text{-}X_u^{CTIME}$ and the value of Y_u^{CTIME} is given by $Y_u^{STIME} + Offset\text{-}Y_u^{CTIME}$. The offsets in the X and Y-axes, depend on the angle of movement and the distance traveled, and are calculated as follows:

$$Offset\text{-}X_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \cos(Angle_u^{STIME})$$
$$Offset\text{-}Y_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \sin(Angle_u^{STIME})$$
$$X_u^{CTIME} = X_u^{STIME} + Offset\text{-}X_u^{CTIME}$$

$$Y_u^{CTIME} = Y_u^{STIME} + Offset\text{-}Y_u^{CTIME}$$

We assume each node is initially configured with information regarding the network boundaries, given by [0, 0], [X_{max}, 0], [X_{max}, Y_{max}] and [0, Y_{max}]. When the predicted X and/or Y co-ordinate is beyond the network boundaries, we set their values to the boundary conditions as stated below.

$$\text{If } (X_u^{CTIME} < 0)\text{, then } X_u^{CTIME} = 0; \quad \text{If } (X_u^{CTIME} > X_{max})\text{, then } X_u^{CTIME} = X_{max}$$
$$\text{If } (Y_u^{CTIME} < 0)\text{, then } Y_u^{CTIME} = 0; \quad \text{If } (Y_u^{CTIME} > Y_{max})\text{, then } Y_u^{CTIME} = Y_{max}$$

Based on the predicted locations of each node in the network at time *CTIME*, the destination node locally constructs the global topology. Note that there exists an edge between two nodes in the locally constructed global topology, if the predicted distance between the two nodes (with the location information obtained from the *LUV*) is less than or equal to the transmission range of the nodes.

2.8 LPBR-M: Multi-path Prediction

The destination node locally runs the algorithm for determining the set of node-disjoint paths [13] on the predicted global topology. The algorithm is explained as follows and is illustrated in Figure 7: Let *G* (*V*, *E*) be the graph representing the predicted global topology. Note that *V* is the set of vertices and *E* is the set of edges in the predicted network graph. Let the source be identified by *s* and destination by *d* and P_N denote the set of node-disjoint *s-d* paths. To start with, we run the O(n^2) Dijkstra minimum-hop path algorithm [18] on *G* to determine the minimum hop *s-d* path in a graph of *n* nodes. If there is at least one *s-d* path in *G*, we include the minimum hop *s-d* path *p* in the set P_N. We then remove all the intermediate nodes (nodes other than source *s* and destination *d*) that were part of the minimum-hop *s-d* path *p* in the original graph *G* to obtain the modified graph *G*' (*V*', *E*'). We determine the minimum-hop *s-d* path in the modified graph *G*' (*V*', *E*'), add it to the set P_N and remove the intermediate nodes that were part of this *s-d* path to get a new updated *G*' (*V*', *E*'). We repeat this procedure until there exists no more *s-d* paths in the network. The set P_N contains the node-disjoint *s-d* paths in the original network graph *G*. Note that when we remove a node from a network graph, we also remove all the links associated with the node.

```
Input: Graph G (V, E), source s and destination d
Output: Set of node-disjoint paths $P_N$
Auxiliary Variables: Graph G'' (V'', E'')
Initialization: G'' (V'', E'') ← G (V, E), $P_N$ ← $\varphi$.
Begin
While ( ∃ at least one s-d path in G'')
    p ← Minimum hop s-d path in G''.
    $P_N \leftarrow P_N \cup \{p\}$
    $\forall_{\substack{vertex, v \in p \\ v \neq s,d \\ edge, e \in Adj\text{-}list(v)}}$   G'' (V'', E'') ← G'' (V''-{v}, E''-{e})
end While
return $P_N$
End
```

Figure 7: Algorithm to Determine the Set of Node-Disjoint Paths (taken from [13])

2.9 Propagation of the MP-LPBR-RREP Messages

The destination *d* sends a MP-LPBR-RREP message to the source *s* on each of the predicted node-disjoint paths. The intermediate nodes on the discovered path attempt to forward the MP-LPBR-RREP message to the next node on the path to the source node *s*. Each intermediate node receiving the MP-LPBR-RREP message updates its routing table to record the incoming interface of the message as the outgoing interface for any new data packets received from the source *s* to the destination *d*. The MP-LPBR-RREP message has a "Number of Disjoint Paths' field to indicate the total number of paths predicted and a 'Is Last Path' Boolean field that indicates whether or not the reported path is the last among the set of node-disjoint paths predicted. If the source node *s* receives at least one MP-LPBR-RREP message before the MP-LPBR-RREP-timer expires, it indicates that the corresponding predicted *s-d* path on which the message propagated through, does exists in reality. The source node creates a new instance of the *NDP-Set* and stores all the newly learnt predicted node-disjoint *s-d* routes and starts sending data on the minimum hop path among them.

Source Node of the Session	Destination Node of the Session	Sequence Number of the Latest MP-RREQ	Number of Disjoint Paths	Is Last Path	Predicted Source – Destination Path (List of Node IDs)
4 bytes	4 bytes	4 bytes	1 byte	1 bit	Variable Size: Multiples of 4 bytes

Figure 8: Structure of the MP-LPBR-RREP Message

The source node estimates the Route-Repair Time (*RRT*) as the time that lapsed between the reception of the last MP-RERR message from an intermediate node and the first MP-LPBR-RREP message from the destination. An average value of the *RRT* is maintained at the source as it undergoes several route failures and repairs before the next broadcast route discovery. The *MP-LPBR-RREP-timer* (initially set to the route acquisition time) will be then set to 1.25*Average *RRT* value, so that we give sufficient time for the destination to learn about the route failure and generate a sequence of MP-LPBR-RREP messages. Nevertheless, this timer value will be still far less than the route acquisition time that would be incurred if the source were to launch a broadcast route discovery. Hence, our approach will only increase the throughput and not decrease it.

2.10 Handling Prediction Failure

If an intermediate node attempting to forward the MP-LPBR-RREP message of the destination could not successfully forward the message to the next node on the path towards the source, the intermediate node informs the absence of the route through a MP-LPBR-RREP-RERR message (structure shown in Figure 9) sent back to the destination. If the destination node receives MP-LPBR-RREP-RERR messages for all the MP-LPBR-RREP messages initiated or the *NEPAT* time has expired, then the node discards all the LUVs and does not generate any new MP-LPBR-RREP message. The destination node will wait for the source node to initiate a broadcast route discovery. After the MP-LPBR-RREP-timer expires, the source node initiates a new broadcast route discovery.

Node originating the MP-LPBR-RREP-RERR message	Sequence Number of MP-LPBR-RREP packet dropped	Source ID of the Data session	Destination ID of the Data session	Downstream Node with which the link failed
4 bytes	4 bytes	4 bytes	4 bytes	4 bytes

Figure 9: Structure of the MP-LPBR-RREP-RERR Message

Task 3: Conduct Simulations of LPBR-M and Compare its Performance with Some of the Currently Existing MANET Multi-path Routing Protocols

The network dimension used is a 1000m x 1000m square network. The transmission range of each node is assumed to be 250m. The number of nodes used in the network is 25, 50 and 75 nodes representing networks of low, medium and high density with an average distribution of 5, 10 and 15 neighbors per node respectively. Initially, nodes are uniformly randomly distributed in the network. We compare the performance of LPBR-M with that of the link-disjoint routing based Ad hoc On-demand Multi-path Distance Vector (AOMDV) routing protocol [7] and the node-disjoint routing based AODV-Multi-path routing protocol [8]. We implemented all of these three multicast routing protocols in a discrete-event simulator developed in Java. The broadcast route discovery strategies simulated are the default flooding approach and the density and mobility aware energy-efficient broadcast strategy called DMEF [14]. The simulation parameters are summarized in Table 1.

Table 1: Simulation Conditions

Network Size	1000m x 1000m	
Number of nodes	25 (low density), 50 (moderate density) and 75 (high density)	
Transmission Range	250 m	
Physical Layer	Signal Propagation Model	Two-ray ground reflection model [19]
MAC Layer	IEEE 802.11 [20]	
	Link Bandwidth	2 Mbps
	Interface Queue	FIFO-based, size 200
Routing Protocols	LPBR-M, AOMDV [7] and AODVM [8]	
Broadcast Strategy	Flooding and DMEF [14]	
Mobility Model	Random Way Point Model [21]	
	Minimum Node Speed, m/s	0 m/s
	Maximum Node Speed, m/s	Low-10; Medium-30; High-50
	Pause Time	0 second
Traffic Model	Constant Bit Rate (CBR), UDP	
	Number of Source-Destination Pairs	15
	Data Packet Size	512 bytes
	Packet Sending Rate	4 Packets/ second
Energy Consumption Model	Transmission Energy	1.4 W [22]
	Reception Energy	1 W [22]

For each combination of network density and node mobility, simulations are conducted with 15 Source-Destination (*s-d*) pairs. Traffic sources are constant bit rate (CBR). Data packets are 512 bytes in size and the packet sending rate is 4 data packets/second. Simulation time is 1000 seconds. The node mobility model used is the Random Waypoint model [21]. The transmission energy and reception energy per hop is set at 1.4 W and 1 W respectively. Initial energy at each node is 1000 Joules.

3.1 Broadcast Strategy: Flooding

Flooding is a widely-used approach for disseminating a message from one node to all the other nodes in a network. In the case of on-demand ad hoc routing protocols [4][5], flooding has been also used to discover a path between a pair of nodes in the network, whenever required. For a given network density, flooding offers the highest probability for each node in the network to receive one or more copies of the flooded message.

We simulate flooding as follows: The initiating source node sets a monotonically increasing value for the Multi-path Route Request (MP-RREQ) message and broadcasts the message to its complete neighborhood formed by the default maximum transmission range of the node. Each node that receives the MP-RREQ checks if it has received a MP-RREQ with the same or higher sequence number. If so, the received MP-RREQ is simply discarded. Otherwise, the intermediate node inserts its own ID in the Route Record field of the MP-RREQ and broadcasts the message within its complete neighborhood. The destination collects all the MP-RREQ messages and selects the set of node-disjoint paths as explained in the heuristic outlined in Figure 3. A sequence of Multi-path Route Reply (MP-RREP) messages, one on each of the node-disjoint paths, is sent back to the source.

3.2 Broadcast Strategy: DMEF

In Research Activity – 1 [14], we had proposed a density and mobility aware energy-efficient broadcast strategy (called DMEF) to discover long-living stable routes with a reduced energy spent during route discovery. DMEF takes into consideration the number of neighbors of a node (a measure of network density) and node mobility. The average hop count of the routes discovered using DMEF is only at most about 8% more than that discovered using flooding.

We simulate DMEF as follows for multi-path broadcast route discoveries: The transmission range of a MP-RREQ broadcast is not fixed for every node. A node that is surrounded by more neighbors in the complete neighborhood will broadcast the MP-RREQ only within a smaller neighborhood that would be sufficient enough to pick up the message and forward it to the other nodes in the rest of the network. On the other hand, a node that is surrounded by fewer neighbors in the complete neighborhood will broadcast the MP-RREQ to a larger neighborhood (but still contained within the complete neighborhood) so that a majority of the nodes in the complete neighborhood can pick up the message and rebroadcast it further. A node rebroadcasts a MP-RREQ at most once. The density aspect of DMEF thus helps to reduce the unnecessary transmission and reception of broadcast MP-RREQ messages and conserves energy.

To discover stable paths that exist for a longer time, DMEF takes the following approach: A node that is highly mobile makes itself available only to a smaller neighborhood around itself, whereas a node that is less mobile makes itself available over a larger neighborhood (but still contained within the complete neighborhood). The reasoning is that links involving a slow moving node will exist for a long time. Hence, it is better for a slow moving node to advertise itself to a larger neighborhood so that the links (involving this node) that are part of the routes discovered will exist for a longer time. On the other hand, a fast moving node will have links of relatively longer lifetime with neighbors that are closer to it. Hence, it is worth to let a fast moving node advertise only to its nearby neighbors.

The rest of the broadcast process is similar to flooding. The destination node collects all the MP-RREQ messages and selects the set of node-disjoint paths using the heuristic outlined in Figure 3. A sequence of Multi-path Route Reply (MP-RREP) messages, one on each of the node-disjoint paths, is sent back to the source.

3.3 Performance Metrics

The performance metrics studied through this simulation are the following:

- **Time between Successive Broadcast Multi-path Route Discoveries:** This is the time between two successive broadcast multi-path route discoveries, averaged over all the *s-d* sessions over the simulation time. We use a set of multi-paths as long as at least one path in the set exists. We opt for a broadcast route discovery when all the paths in a multi-path set fails. Hence, this metric is a measure of the lifetime of the set of multi-paths and the larger the value of this metric, the better the protocol in terms of multi-path route stability and route discovery control overhead.
- **Average Energy Lost per Data Packet Delivered:** This is the sum of the energy consumed for transmission and reception at every hop, the energy consumed at the neighbors for coordination during channel access, the energy lost due to route discoveries and the energy lost due to periodic

beaconing, if any, averaged over all the data packets delivered successfully from the source to the destination.

- **Packet Delivery Ratio:** This is the ratio of the total number of data packets delivered to the destination to that of the total number of data packets originating from the source, averaged over all the *s-d* sessions. With a larger queue size of 200 at each node, the packet delivery ratio is more a representative of the connectivity of the network.
- **Energy Lost per Broadcast Multi-path Route Discovery:** This is the energy consumed per global broadcast based route discovery attempt, averaged over all the *s-d* sessions over the entire simulation time. The energy consumed per global broadcast route discovery attempt includes the energy consumed to transmit (broadcast) a MP-RREQ message to all the nodes in the neighborhood and the energy consumed to receive the MP-RREQ message sent by each node in the neighborhood, summed over all the nodes.
- **Control Message Overhead:** This is the ratio of the total number of control messages (MP-RREQ, MP-RREP, MP-LPBR-RREP and MP-LPBR-RREP-RERR) received at every node to that of the total number of data packets delivered at a destination, averaged over all the *s-d* sessions across the entire simulation time. Note that we prefer to consider the number of control messages received rather than transmitted because, in a typical broadcast operation, the total amount of energy spent to receive a control message at all the nodes in a neighborhood is greater than the amount of energy spent to transmit the message.
- **Average Energy Lost per Node:** The is the energy lost at a node due to transmission and reception of data packets, control packets and beacons, if any, averaged over all the nodes in the network for the entire simulation time.
- **Average Number of Disjoint Paths Found per Multi-path:** This is the number of disjoint-paths (link-disjoint or node-disjoint, depending on the routing protocol) determined during a multi-path broadcast route discovery, averaged over all *s-d* sessions and over the entire simulation time.
- **Average Number of Disjoint Paths used per Multi-path:** This is the number of disjoint-paths (link-disjoint or node-disjoint, depending on the routing protocol) actually used by the routing protocol, averaged over all the *s-d* sessions across the entire simulation time. All the disjoint-paths determined during a broadcast route discovery may not be actually used by a routing protocol. Some of the disjoint paths might have failed before the routing protocol considers using them. Note that we use the disjoint paths in the order of their hop count.
- **Average Hop Count of all Disjoint-paths used:** This is the time-averaged hop count of the disjoint paths determined and used by each of the multi-path routing protocols studied. For example, if a protocol determines the multi-path set MP_1 and MP_2: MP_1 has three disjoint paths $P_{1\text{-}1}$, $P_{1\text{-}2}$ and $P_{1\text{-}3}$ with hop count 3, 4 and 2 and are used for 2, 8 and 3 seconds respectively; MP_2 has two disjoint paths $P_{2\text{-}1}$ and $P_{2\text{-}2}$ with hop count 5 and 3 and are used for 7 and 4 seconds respectively. The time-averaged hop count of the disjoint paths used is 3.79 and is calculated as follows:

$$hopCount = \frac{\sum_{i=1}^{\#Multi-Paths} \sum_{j=1}^{\#Paths[i]} \left[hops(P_{i-j}) * time(P_{i-j}) \right]}{\sum_{i=1}^{\#Multi-Paths} \sum_{j=1}^{\#Paths[i]} time(P_{i-j})}$$

$$hopCount = \frac{[3*2+4*8+2*3]+[5*7+3*4]}{[2+8+3+7+4]} = \frac{91}{24} = 3.79$$

Task 4: Analyze the Simulation Results with respect to Different Performance Metrics

The performance results for each metric displayed in Figures 10 through 18 are an average of the results obtained from simulations conducted with 5 sets of mobility profiles and 15 randomly picked source-destination (*s-d*) pairs for each combination of node velocity and network density values.

4.1 Time between Successive Broadcast Multi-path Route Discoveries

The LPBR-M protocol yields the longest time between successive broadcast multi-path route discoveries (refer Figure 10). This implies that the set of node-disjoint paths discovered and predicted by LPBR-M are relatively more stable than the set of link-disjoint and node-disjoint paths discovered by the AOMDV and AODVM routing protocols respectively. Also, when DMEF is used as the route discovery strategy, each of the three multi-path routing protocols yielded a longer time between route discoveries, compared to the use of flooding as the route discovery strategy.

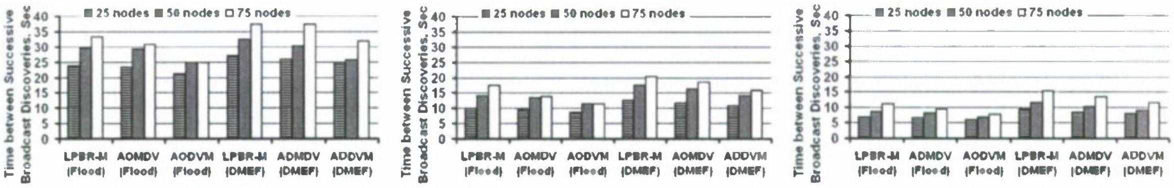


Figure 10.1: v_{max} = 10 m/s **Figure 10.2:** v_{max} = 30 m/s **Figure 10.3:** v_{max} = 50 m/s

Figure 10: Time between Successive Broadcast Multi-path Route Discoveries

As we increase the level of node mobility from low to moderate and high, the difference in the time between successive route discoveries incurred for AOMDV and AODVM vis-à-vis LPBR-M increases. Also, for a given level of node mobility, as we increase the network density from low to moderate and high, the time between successive route discoveries for LPBR-M increases relatively faster compared to those incurred for AOMDV and AODV-M. LPBR-M yields 3%-17% and 15%-44% more time between successive route discoveries compared to AOMDV and AODVM respectively. For each of the three multi-path routing protocols, the increase in the time between route discoveries when DMEF is used as the route discovery strategy is 4%-28%, 16%-38% and 28%-50% more than that incurred with flooding at low, moderate and high node mobility levels respectively.

4.2 Average Energy Lost per Data Packet Delivered

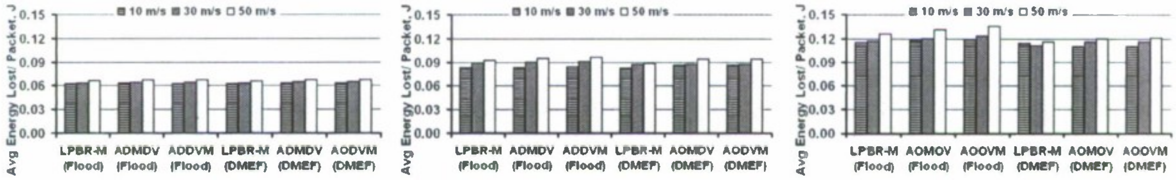


Figure 11.1: 25 Nodes **Figure 11.2:** 50 Nodes **Figure 11.3:** 75 Nodes

Figure 11: Average Energy Lost per Data Packet Delivered

For a given level of node mobility and network density, the energy consumed per data packet (refer Figure 11) for each of three multi-path routing protocols is not very different from each other (the difference is within 3%). However, the energy consumed per data packet at a moderate network density of 50 nodes and a high network density of 75 nodes is respectively about 31%-44% and 75%-100% more than the energy consumed per data packet incurred in a low network density of 25 nodes. This can be attributed to the increase in the number of nodes receiving a broadcast message and transmitting the message in the network. Also, more neighbors are involved in the Request-to-Send and Clear-to-Send message reception during co-ordination for channel access in every hop of a path taken by every data packet. In networks with high level of node mobility, we observe that the energy consumed per data packet with flooding as the route discovery strategy can be 2% (low density)-11% (high density) more than that obtained with DMEF as the route discovery strategy.

4.3 Packet Delivery Ratio

For a given level of node mobility and network density, the packet delivery ratio (refer Figure 12) of each of the multi-path routing protocols almost remained the same. In networks of low density, we observe 86% - 93% packet delivery ratio. Also, in low density networks, we observe that as the level of node mobility increases from low to moderate and high, the packet delivery ratio decreases by about 4%-5%. With a FIFO-based queue of size 200 at each node, the lower packet delivery ratio in low-density networks is mainly attributed to poor network connectivity. In moderate and high density networks, each of the three routing protocols yield a packet delivery ratio of at least 98% and 99% respectively.

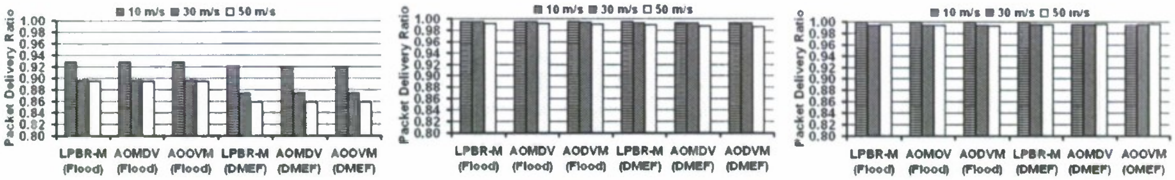


Figure 12.1: 25 Nodes **Figure 12.2:** 50 Nodes **Figure 12.3:** 75 Nodes

Figure 12: Packet Delivery Ratio of LPBR-M, AOMDV and AODVM under both Flooding and DMEF

4.4 Energy Lost per Broadcast Multi-path Route Discovery

For a given level of node mobility and network density, the energy consumed per broadcast multi-path route discovery (refer Figure 13) for each of the three multi-path routing protocols is almost the same as this metric depends only on the route discovery strategy and not on the routing protocol. The energy consumed per route discovery in a moderate network density of 50 nodes and a high network density of 75 nodes is respectively about 3.4 to 4.1 times and 8.0 to 8.5 times more than the energy consumed per route discovery in a low network density of 25 nodes. This can be attributed to the increase in the number of nodes receiving a broadcast message and transmitting the message in the network. With the DMEF strategy, we observe a decrease in the magnitude of energy consumed per route discovery at high network density and high node mobility. This can be attributed to the clever adaptation of the broadcast range by the DMEF strategy. In networks of low and moderate density, flooding consumes 19%-23% more energy per route discovery when compared to DMEF; whereas in high density networks, flooding consumes 32-38% more energy per route discovery compared to DMEF.

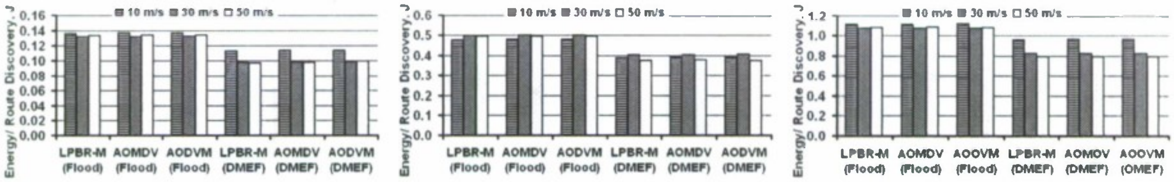


Figure 13.1: 25 Nodes **Figure 13.2:** 50 Nodes **Figure 13.3:** 75 Nodes

Figure 13: Energy Lost per Broadcast Route Discovery under both Flooding and DMEF

4.5 Control Message Overhead

For a given level of node mobility and network density, LPBR-M incurs the lowest control message overhead (refer Figure 14). For a given level of node mobility, AOMDV and AODVM respectively incur 4%-16% and 14%-34% more control message overhead than LPBR-M when flooding is used as the route discovery strategy. On the other hand, when DMEF is used as the route discovery strategy, AOMDV and AODVM respectively incur 10%-14% and 11%-23% more control message overhead than LPBR-M. For a given level of network density, the control message overhead incurred by each of the three routing protocols using flooding as the route discovery strategy in networks of low, moderate and high node mobility is respectively 7%-39%, 32%-58% and 49%-110% more than that incurred with DMEF as the route discovery strategy.

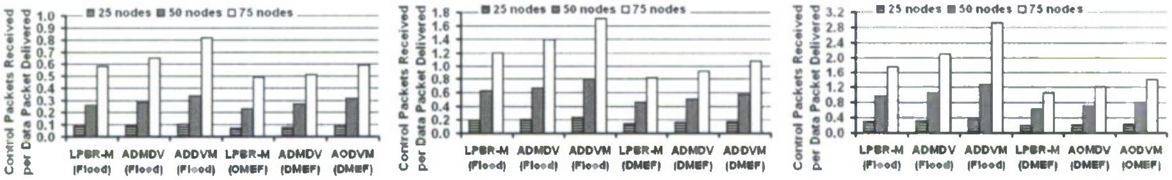


Figure 14.1: 25 Nodes **Figure 14.2:** 50 Nodes **Figure 14.3:** 75 Nodes

Figure 14: Control Message Overhead for LPBR-M, AMDV and AODVM under Flooding and DMEF

In networks of moderate node mobility, the control message overhead incurred by each of the three multi-path routing protocols while using flooding and DMEF is respectively 2.1 (high density) to 3.4 (low density) times and 1.7 to 2.0 times more than that incurred in networks of low node mobility. In networks of high node mobility, the control message incurred by each of the three multi-path routing protocols while using flooding and DMEF is respectively 3.0 (high density) to 3.7 (low density) times and 2.2 (high density) to 2.8 (low density) times more than that incurred in high density networks, the control message overhead incurred in networks of low node mobility. Thus, DMEF substantially reduces the control message overhead as we increase the network density and/or the level of node mobility.

4.6 Average Energy Lost per Node

We conduct all of our simulations with a fixed offered traffic load comprising of 15 *s-d* pairs. Hence, as we increase the network density, the net energy consumed per node decreases as more nodes are available in the network for data transfer. For both flooding and DMEF, the energy lost per node in networks of moderate and high density is respectively about 65%-75% and 70%-84% of the energy lost per node in networks of low mobility. For a given network density, the energy lost per node at high node mobility is greater than the energy lost per node at low node mobility by at most 16% and 10% when operated with flooding and DMEF respectively.

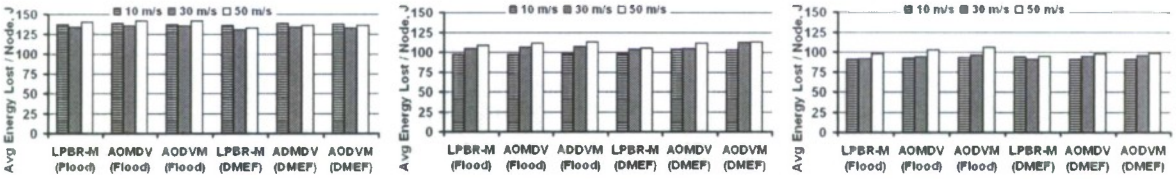


Figure 15.1: 25 Nodes **Figure 15.2:** 50 Nodes **Figure 15.3:** 75 Nodes

Figure 15: Average Energy Lost per Node

4.7 Average Number of Disjoint Paths Found per Multi-path

For a given routing protocol and network density, the average number of disjoint paths discovered per multi-path (refer Figure 16) almost remains the same, irrespective of the level of node mobility. With increase in network density, the number of link-disjoint and node-disjoint paths between a source and destination increases. For a given network density and broadcast route discovery strategy, the link-disjoint path routing based AOMDV determines a larger number of disjoint paths (32%-62% more) than LPBR-M and AODVM; the node-disjoint path routing based LPBR-M determines relatively larger number of disjoint paths (12%-22% more) than the other node-disjoint path routing based AODVM. For each of the three routing protocols, the average number of disjoint paths determined in a moderate density network and high-density network is respectively about 55%-95% and 120%-200% more than that determined in a low-density network. As DMEF reduces the control overhead and the number of nodes forwarding the MP-RREQ messages, the average number of disjoint paths determined for the three routing protocols is about 5% (low density) to 20% (high density) lower than that discovered using flooding.

4.8 Average Number of Disjoint Paths used per Multi-path

For a given level of node mobility and network density, the link-disjoint path based AOMDV had the largest number of disjoint paths actually used. But, the magnitude of the number of AOMDV link-disjoint paths actually used (refer Figure 17) is only at most 25% more than the number of LPBR-M node-disjoint paths or the AODVM node-disjoint paths. Even though AOMDV had a relatively larger number of link-disjoint paths (as explained in Section 4.8), the percentage of such paths successfully used is the lowest among the three multi-path routing protocols. The node-disjoint path based AODVM routing protocol has the largest percentage of the discovered disjoint paths actually being used. The percentage of node-disjoint paths successfully used in the case of LPBR-M is in between to those of AODVM and AOMDV. As the network density increases, the number of disjoint paths actually used by each of the three multi-path routing protocols increases, nevertheless at a significantly reduced rate. As a result, the percentage of the discovered disjoint paths successfully used decreases with increase in network density. This can be attributed to the failure of the disjoint paths over time and the disjoint-paths discovered are not actually available when the routing protocol wants to use them.

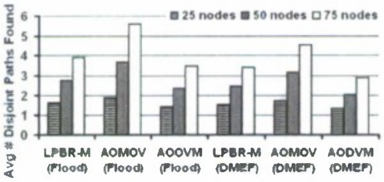


Figure 16: Average Number of Disjoint Paths Found per Multi-path

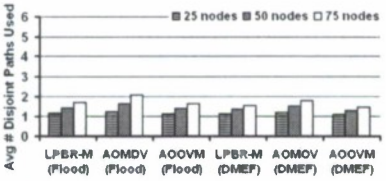


Figure 17: Average Number of Disjoint Paths Used per Multi-path

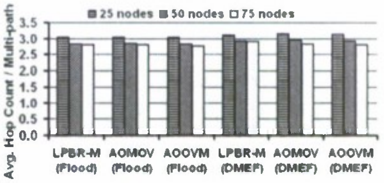


Figure 18: Average Hop Count of All Disjoint Paths Used

4.9 Average Hop Count of All Disjoint-Paths Used

For a given routing protocol and network density, the average hop count (refer Figure 18) of the disjoint-paths used is almost the same, irrespective of the level of node mobility. As we add more nodes in the network, the hop count of the paths tends to decrease as the source manages to reach the destination through a relatively lesser number of intermediate nodes. With increase in network density, there are several candidates to act as intermediate nodes on a path. The average hop count of the paths in high and moderate density networks is 6%-10% less than the average hop count of the paths in networks of low density. For each of the routing protocols, for all network densities, the average hop count of the paths discovered using DMEF is at most 4% more than the hop count of the paths determined using flooding.

III. Summary of Accomplishments in Research Activity 3

This research work contributed to the design and development of a multi-path extension to the location prediction based routing (LPBR) protocol for mobile ad hoc networks (MANETs). LPBR has been proposed to simultaneously minimize the number of route discoveries as well as the hop count of the paths for unicast routing in MANETs. We have developed a node-disjoint multi-path version of LPBR, referred to as LPBR-M, to simultaneously minimize the number of broadcast route discoveries as well as the hop count of the paths for multi-path routing. LPBR-M is designed as follows: When the source has data to send to the destination, but is not aware of any route to the latter, the source broadcasts MP-RREQ messages throughout the network. Each intermediate node includes its location and mobility information in the MP-RREQ message. The destination receives several MP-RREQ messages and extracts a set of node-disjoint paths that were traversed by the MP-RREQ messages. The destination then sends a sequence of MP-RREP messages, one on each of the node-disjoint paths learnt. The source learns the set of node-disjoint paths and uses them to send data, in the increasing order of their hop count. A node-disjoint path is used as long as it exists. If all the node-disjoint paths known to the source cease to exist, the source does not immediately initiate a new broadcast route discovery, but waits to receive a sequence of MP-LPBR-RREP messages from the destination. The destination predicts the global topology based on the latest location and mobility information collected from the MP-RREQ messages, runs the node-disjoint path algorithm based on the Dijkstra's algorithm and sends a sequence of MP-LPBR-RREP messages, one on each of the predicted node-disjoint paths. If the source does not receive any MP-LPBR-RREP message within a certain time, the source initiates a global broadcast multi-path route discovery. If the source receives at least MP-LPBR-RREP message, it continues to send data using the learnt path(s) and does not initiate any broadcast multi-path route discovery.

Simulations have been conducted with both flooding and DMEF as the broadcast multi-path route discovery strategies. We compared the performance of LPBR-M with that of the link-disjoint path based AOMDV and the node-disjoint path based AODVM multi-path routing protocols. LPBR-M achieves the longest time between successive route discoveries, lowest energy consumed per data packet and the lowest control message overhead. LPBR-M achieves hop count that is almost equal to that obtained with the minimum-hop based AOMDV and AODVM. Moreover, DMEF helps each of the multi-path routing protocols to determine a set of node or link disjoint paths that exist for a long time and at the same time does not increase the source-destination hop count appreciably. Each of the multi-path routing protocols incurred a lower energy spent per route discovery, compared to flooding.

IV. Publication Details

A conference paper primarily featuring the design of the node-disjoint multi-path protocol and the simulation results for all the performance metrics presented in this report has been accepted for publication in the 3rd *International Conference on Signal Processing and Communication Systems*, Omaha, Nebraska, September 28-30, 2009.

References

[1] S. Mueller, R. P. Tsang and D. Ghosal, "Multi-path Routing in Mobile Ad Hoc Networks: Issues and Challenges," *Lecture Notes in Computer Science*, Vol. 2965, pp. 209 – 234, April 2004.

[2] W. Xu, P. Yan and D. Xia, "Similar Node-Disjoint Multi-paths Routing in Wireless Ad Hoc Networks," *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, Vol. 2, pp. 731 – 734, Sept. 2005.

[3] H. Wang, K. Ma and N. Yu, "Performance Analysis of Multi-path Routing in Wireless Ad Hoc Networks," *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, Vol. 2, pp. 723 – 726, Sept. 2005.

[4] D. B. Johnson, D. A. Maltz and J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Ad hoc Networks," in *Ad hoc Networking*, Chapter 5, C. E. Perkins, Eds. Addison Wesley, pp. 139 – 172, 2000.

[5] C. E. Perkins and E. M. Royer, "Ad hoc On-demand Distance Vector Routing," *Proceedings of the 2nd Annual IEEE International Workshop on Mobile Computing Systems and Applications*, pp. 90 – 100, February 1999.

[6] S. Lee and M. Gerla, "Split Multi-path Routing with Maximally Disjoint Paths in Ad Hoc Networks," *Proceedings of IEEE International Conference on Communications*, Vol. 10, pp. 3201-3205, 2001.

[7] M. K. Marina and S. R. Das, "On-demand Multi-path Distance Vector Routing in Ad Hoc Networks," *Proceedings of the IEEE International Conference on Network Protocols*, pp. 14 – 23, Nov. 2001.

[8] Z. Ye, S. V. Krishnamurthy and S. K. Tripathi, "A Framework for Reliable Routing in Mobile Ad Hoc Networks," *Proceedings of the IEEE International Conference on Computer Communications*, 2003.

[9] V. Loscri and S. Marano, "A New Geographic Multi-path Protocol for Ad Hoc Networks to Reduce the Route Coupling Phenomenon," *Proceedings of the 63rd IEEE Vehicular Technology Conference*, VTC 2006-Spring, Vol. 3, pp. 1102-1106, 2006.

[10] M. Li, L. Zhang, V. O. K. Li, X. Shan and Y. Ren, "An Energy-Aware Multi-path Routing Protocol for Mobile Ad Hoc Networks," *Proceedings of the ACM SIGCOMM Asia*, April 2005.

[11] L. Bononi and M. D. Felice, "Performance Analysis of Cross-layered Multi-path Routing and MAC Layer Solutions for Multi-hop Ad Hoc Networks," *Proceedings of the ACM International Workshop on Mobility Management and Wireless Access*, pp. 190 – 197, 2006.

[12] A. Acharya, A. Misra and S. Bansal, "A Label-Switching Packet Forwarding Architecture for Multi-hop Wireless LANs," *Proceedings of the 5th ACM International Workshop on Wireless Mobile Multimedia*, Sept. 2002.

[13] N. Meghanathan, "Stability and Hop Count of Node-Disjoint and Link-Disjoint Multi-path Routes in Ad Hoc Networks," *Proceedings of the 3rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, New York, October 2007.

[14] N. Meghanathan, "A Density and Mobility Aware Energy-Efficient Broadcast Strategy to Determine Stable Routes in Mobile Ad hoc Networks," accepted for publication in the *Proceedings of the International Conference on Wireless Networks*, Las Vegas, NV, July 13-16, 2009.

[15] N. Meghanathan, "A Location Prediction-Based Reactive Routing Protocol to Minimize the Number of Route Discoveries and Hop Count per Path in Mobile Ad Hoc Networks," *The Computer Journal*, vol. 52, no. 4, pp. 461-482, July 2009.

[16] V. Naumov, R. Baumann and T. Gross, "An Evaluation of Inter-Vehicle Ad hoc Networks based on Realistic Vehicular Traces," *Proceedings of the 7th ACM International Symposium on Mobile Ad hoc Networking and Computing*, Florence, Italy, pp. 108-119, May 22-25, 2006.

[17] J-P. Sheu, C-M. Chao, W-K. Hu and C-W. Sun, "A Clock Synchronization Algorithm for Multi-Hop Wireless Ad Hoc Networks," *Wireless Personal Communications: An International Journal*, vol. 43, no. 2, pp. 185-200, October 2007.

[18] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms," 2nd Edition, MIT Press/ McGraw Hill, Sept. 2001.

[19] L. Breslau, et. al., "Advances in Network Simulation," *IEEE Computer*, vol. 33, no. 5, pp. 59-67, 2000.

[20] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal of Selected Areas in Communication*, vol. 18, no. 3, pp. 535-547, 2000.

[21] C. Bettstetter, H. Hartenstein and X. Perez-Costa, "Stochastic Properties of the Random-Waypoint Mobility Model," *Wireless Networks*, vol. 10, no. 5, pp. 555-567, 2004.

[22] L. M. Feeney, "An Energy Consumption Model for Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks," *Journal of Mobile Networks and Applications*, vol. 3, no. 6, pp. 239 – 249, June 2001.

Research Activity – 4

Design of a Highly-Directional Antenna for Wireless Networks

Dr. Kamal S. Ali
Professor
Department of Computer Engineering
Jackson State University
Jackson, MS 39217
Email: kamal.ali@jsums.edu
Phone: 601-979-1183

Dr. Abdelnasser Eldek
Assistant Professor
Department of Computer Engineering
Jackson State University
Jackson, MS 39217
Email: abdelnasser.eldek@jsums.edu
Phone : 601-979-1188

I Description of the Task

Recent comparative study on the performance of multi-path routing using omni-directional and directional antenna shows that directional antenna improves the performance of multi-path routing significantly as compared to that with omni-directional antenna. Within this effort, we propose to design a highly directive antenna. The proposed antenna will be a cavity backed slot antenna through multiple layer superstrate. It is known that a dielectric overlay can enhance the directivity of slot antennas. The multilayer effect on cavity backed slot antennas will be studied here to produce more directive patterns.

In a future effort, the directional antenna proposed will be used to build a system for multi-path routing protocols (like LPBR-M) that will allow communication between nodes at a lower energy cost, while enhancing the routing tables with directional information. The system will comprise multiple directional Microstrip antennas that will initially all radiate to communicate omni-directionally. Upon establishing the direction of the target node, through a comparison of received power on each of the directional antennas, communication is then limited to the antenna best suited for that directional communication. This system will allow the augmentation of routing tables with node directional information increasing the network's topological awareness while improving power conservation. This research effort will investigate the necessary algorithms needed to extract topological information from the augmented routing tables.

II Task Activities

1. Student will be hired to work on the task.
2. Training the student on self organizing maps and Antenna modeling software
3. Algorithm development and Antenna geometry suggestion and modification
4. Simulation
5. Results' analysis
6. Final results

III Time Line

Activity No.	Sept. 23, 08								June 22, 09
1	▨	▨							
2		▨	▨						
3			▨	▨	▨	▨			
4				▨	▨	▨	▨	▨	▨
5							▨	▨	▨
6							▨	▨	▨

IV Status of the Activities

Task	*Status*
1. Hiring the students to work on the tasks.	Completed
2. Training the students on self-organizing maps and Antenna modeling software	Completed
3. Algorithm development/Antenna geometry suggestion and modification	Completed
4. Simulations	Completed
5. Results' analysis	Completed
6. Final results	Comppleted

V Description of the Completed and Current Activities

Task 1: Hiring the Students to Work on the Tasks

The students William Munn, Christopher Munn were charged with the algorithm development for the sensor node topological identification, while the students Chantain Greer and Mohamed Idris were charged with the antenna development software installation and operation.

Task 2: Training the Students on Self-Organizing Maps and Antenna Modeling Software

The students were trained on self organizing maps. However, there were not enough facilities to train them on antenna modeling and simulation. The available commercial software which is Ansoft HFSS has extreme memory and high speed requirements. These include RAM space of at least 4GB, preferably 8GB, and speed not less than 3.7 GHz. Also, the software is working only on Windows XP operating system, and does not work with Vista. The only available PC during the academic year was Dr. Eldek's personal computer at home, where he installed the software and does most of the work related to this project. The students have spent the summer in Oak Ridge National Laboratory. Another student, Felmon Berho, has also joined the group. There, Dr. Eldek gives them a workshop about antennas and the Ansoft HFSS software.

Task 3: Algorithm Development and Antenna Geometry Suggestion and Modification

One of our goals is to develop an algorithm with which to take information from individual nodes containing the relative directions of other nodes within a finite range based on that node's orientation and generate a spatial map of those nodes. To accomplish this goal, we first compile a "Visible Vector Table" consisting of the relative direction to every visible node for each node relative to the node's orientation.

Next, we arbitrarily chose a node as our "Origin Node" and use the Visible Vector Table to determine the orientation of every other node, relative to the Origin Node's orientation. All angles in the Visible Vector Table are then adjusted to be relative to the Origin Node, and redundant vectors purged from the table. At this point, a "Visible Triangle Table" is generated. Using the Visible Triangle Table, vectors are sorted into regions, and relative vector length is calculated within each region. The relative scale of each region is then approximated, and the distance and direction to every node determined relative to the Origin Node. Coordinates of each node are then calculated and plotted in a spatial map.

During the development of this algorithm we made a number of assumptions. First, all nodes have the same viewable range. A path can be made from any node to any other node by a series of visible links. Finally, all nodes can be uniquely identified, which we will call for the purposes of this paper, a node's ID. Several scenarios were explored using this algorithm. So far, the algorithm was found to hold well. Currently we are developing a visualization tool to allow for full visualization and testing of the algorithm.

To start the algorithm development we made the assumption that all nodes are perfectly oriented, in other words, the orientation of each node is known at the time of data collection. This information is essential if we are to triangulate the geolocation of all nodes. Although a simple magnetic sensor can resolve this issue and determine the orientation of a node, an attempt was made to extract nodal orientation from available data. The software will therefore use the routing information to initially determine the orientation of all nodes. Only when orientation is determined would the program attempt the determination of nodal geolocation. Clearly, the accuracy of node orientation will depend on the number of antennae per node or the directionality of each of these antennae. In other words, the error in node orientation will be at least as large as the directionality of the antenna's radiation beam.

When highly directional antennae are obtained, this system may be employed. With high directionality the error in nodal orientation will be small allowing the geolocation system to converge. However, it is advisable to use a local sensor (magnetic or GPS) to determine the orientation of the individual nodes, as this will make for a simpler and more accurate solution. The orientation obtained from these sensors is then transmitted to other nodes for inclusion in the routing tables. In doing so, the accuracy of node orientation is no longer dependant of the antennae directionality.

The algorithm for determining the geolocation of nodes is based on the routing tables that contain nodal orientation. Omitting signal strength the algorithm will operate as follows:

1. Select a central node
2. Find two nodes with routing tables that contain the central node as well as each other.
3. Find an initial placement that can resolve the three routing tables.
4. Introduce nodes in the vicinity, (Nodes that have routing tables with entries for most of the above nodes)
5. Modify the solution to incorporate the new node.
6. Continue till all nodes are incorporated.

This procedure was found to work for most cases. There are, however, cases where this system may not produce a complete solution. In this case the process will have to be re-started with a different set of nodes. In doing so, we try to avoid the local minima and converge at a global minima allowing for valid solution. For a small number of nodes, this system may be used since convergence time is short and a final solution may be arrived at in a timely manner. For large number of nodes a more sophisticated algorithm may need to be used.

To begin extracting geolocation information from routing tables, we started with a perfect system, were perfect nodes occupying grid points. For a system of three nodes, see Fig. 1, with each node having 8 directional antennas, the routing table of node (a) will have b@2 and c@1. The routing table of node (b) will have a@6 and c@0, whereas the table of node (c) will have a@5 and b@4. It is clear, from this example that an infinite number of solutions exist. An exact solution may be obtained only if two of these nodes have fixed, or known locations.

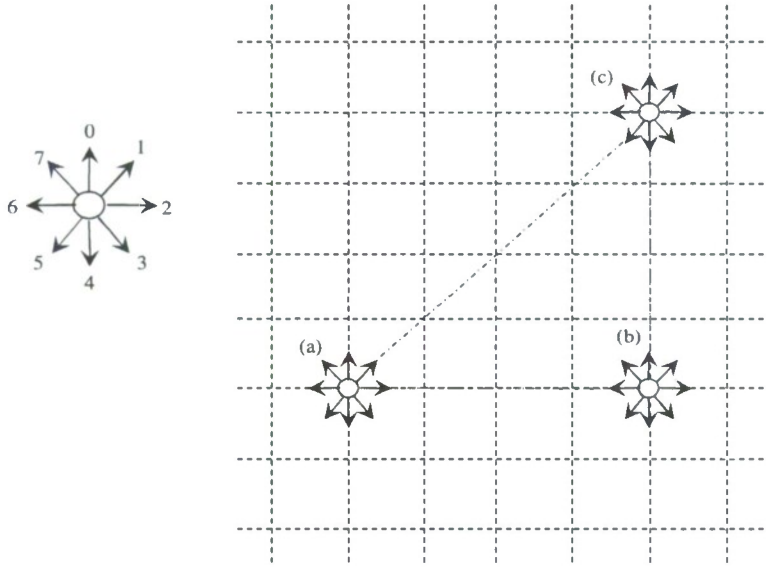


Figure 1: Perfect Nodes on Perfect Grid.

In the example above, if signal strength could be used and translated into distance, a unique relative geolocation may be obtained and with knowing the exact location of a single node an exact geolocation solution for all nodes may be obtained.

Below is a pseudo code outline of the algorithm that will be used to find nodal orientation and relative geolocation.

Note: All array indexes are from 1 to L where L is the size of array.

For all nodes in routing table, assign unique ID 1 to N.

```
N = Number of Nodes
M = Number of Antennae per Node
VVT[N,N] = {0}
VTT[N,N,N] = {0}
VLT[N,N,N] = {0}
ROT[N] = {0}

// Build Visible Vector Table (VVT) and Visible Triangle Table (VTT)
For i from 1 to N {
    For j from i+1 to N {
        If j in node i's routing table
            then {
                VVT[i, j] = angle of antenna on which i sees j
```

```
                VVT[j, i] = angle of antenna on which j sees i
                For all nodes (k) in from j+1 to N {
                    If node i is in node k's routing table
                        then VTT[i,j,k] = 1
                }
            }
    }
}

// Build Relative Orientation Table (ROT)
for all nodes (i) from 2 to N {
    Add ROT[i] = ((VVT[1,i] - VVT[i,1] + 180) modulus 360)
}

// Orient VVT using ROT
For i from 1 to N {
    For j from 1 to N {
    if i != 1 and VVT[i,j] != 0
        then VVT[i,j] -= ROT[i]
}

// Build Vector Length Table (VLT)
VLT[1,2] = 1
For i from 1 to N {
    For j from i+1 to N {
        for k from j+1 to N {
            if VTT[i,j,k] != 0 {
                if VLT[i,j] != 0
                    then {
                        a = i;
                        b = j;
                        c = k;
                } else if VLT[j,k] != 0
                    then {
                        a = j;
                        b = k;
                        c = i;
                } else then {
                    a = k;
                    b = i;
                    c = j;
                }
                if VLT[a,c] == 0
                    then VLT[a,c] = VLT[a,b] * (sine(((VVT[b,a] - VVT[b,c] + 360) modulus 360)) / sine(((VVT[c,a] - VVT[c,b] + 360) modulus 360)))
                if VLT[b,c] == 0
                    then VLT[b,c] = VLT[a,b] * (sine(((VVT[a,b] - VVT[a,c] + 360) modulus 360)) / sine(((VVT[c,a] - VVT[c,b] + 360) modulus 360)))
                }
            }
        }
```

```
        }
}
```

Clearly the more directive the antennas used, the more accurate the geolocation of these nodes.

Our goal is therefore to create a highly directive antenna system that will allow for communication between nodes at a lower energy cost, while enhancing the routing tables with directional information. The system will comprise multiple directional Microstrip antennas that will initially radiate to communicate omni-directionally. Upon establishing the direction of the target node, by measuring received power on each of the directional antennas, communication is then limited to the antenna best suited for that directional communication. This will result in a network communicating at a fraction of the power needed with omni-directional antennae. It is anticipated that temporary communication loss may occur when nodes move, however, this can be remedied by reverting to omni-directional transmission to establish the new antennae set and updating the routing tables accordingly.

By studying existing kinds of antennas and searching literature on high directive antennas, two antennas are chosen for further study: the microstrip-fed double rhombus antenna and the Cavity backed slot antenna. The first antenna is small in size and provides around 7 dB Gain. The suggested modification to improve this gain is to increase the vertical length of the substrate so that it can act as a narrow horn because of its high dielectric constant. It is expected that this modification will increase the antenna gain to around 12 dB. The backed slot antenna has a large ground plane, which increase the overall size of the antenna. However, it provides 16 dB gain. We suggest modifying the geometry of the antenna and its ground plane to decrease its overall size, and then increase the gain.

Task 4: Simulations

Initial geometry of the first antenna is modeled using the commercial software package Ansoft HFSS [1]. The proposed antenna is the double rhombus antenna presented in [2, 3] for UWB applications. By studying this antenna it is noticed that the size of dielectric substrate is the main factor which affects the gain. Therefore we decided to study different configurations of the substrate shown in Fig. 2 in order to see the effect of the length (L) and width (W) of the substrate, the photonic band gap (PBG) structures in the upper layer, and the stair case substrate shape by changing the parameter (t).

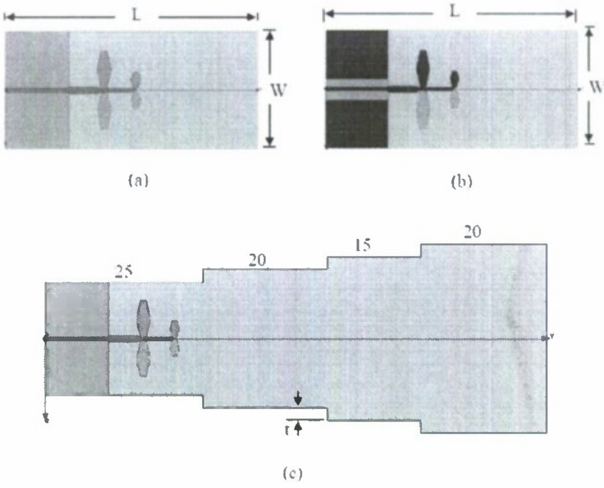
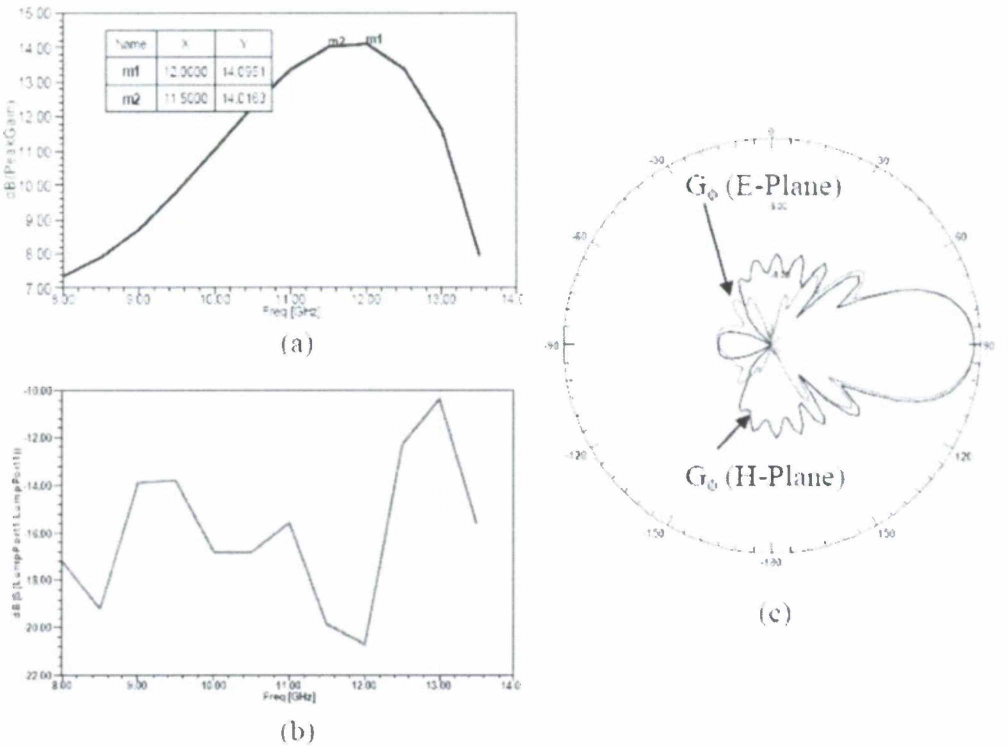


Figure 2: Different Antenna Configurations

Task 5: Results' Analysis

The three antenna configurations in Fig. 2 are studied using Ansoft HFSS. Particularly, we studied the effect of L while fixing W as we need the antenna as small in width as possible to fit into an array which will be used in the nodes. In addition, we studied the effect of PBG structures in order to decrease the back radiation and decrease the total length of the antenna. Tables 1 to 3 summarize the simulation results. As L increases, the maximum gain increases, and deceases the frequency of the maximum gain. A quite high gain of 14.1 dB gain is achieved at 12 GHz when L =120 mm compared to7.2 dB for the original antenna in [2, 3], which is 96% improvement in the antenna gain. Fig. 3 shows the radiation properties of the antenna results with W = 18 mm and L = 120 mm: (a) peak gain in dB vs. frequency in GHz, (b) Return loss in dB, and (c) radiation patterns in the E- and H-Planes at 12 GHz. The antenna is operating in a wide bandwidth that extends from less than 8 GHz to more than 14 GHz. Table 2 shows that the PBG structure helps decreasing the frequency of maximum gain without significant increase in the maximum gain. Table 3 summarizes the effect of t in the third configuration. As t increases from -2 to 4mm, the gain increases from 12.21 to 14.39 dB, and the frequency of the maximum gain decreases from 13 to 11.5 GHz. This result is helpful, especially when the node consists of circular antenna array, which allows for increasing the antenna width from the far end.



(a)

(b)

(c)

Figure 3: Antenna results with W = 18 mm and L = 120 mm: (a) Peak gain in dB vs. frequency in GHz, (b) Return loss in dB, and (c) Radiation patterns at 12 GHz.

Table 1: Effect of L for Antenna in Fig. 2(a) with W = 18 mm.

L (mm)	40	60	80	120
G_{max} (dB)	9.58	12.26	13.45	14.1
G_{max} frequency (GHz)	13	13	12.5	12

Table 2: Effect of PBG for Antenna in Fig. 2(b) with L = 40 mm and W = 18 mm.

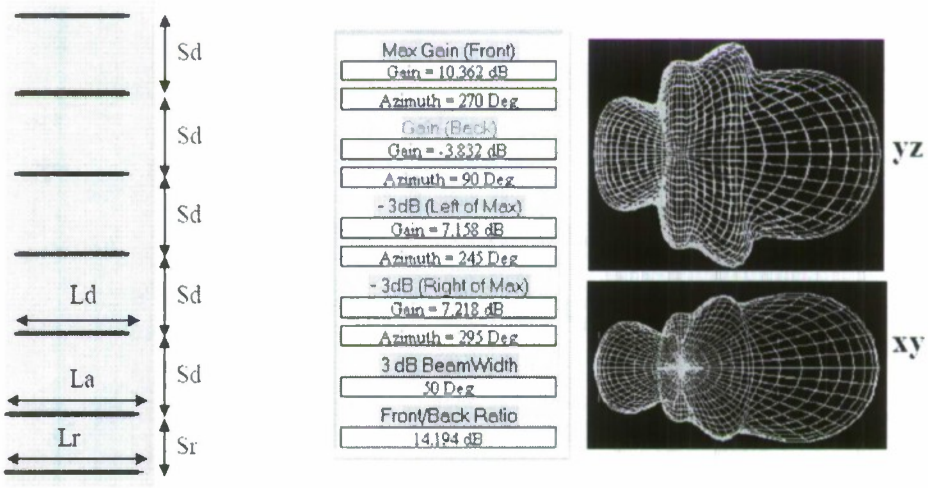
	Without PBG	With PBG
G_{max} (dB)	9.58	9.60
G_{max} frequency (GHz)	13	12

Table 3: Effect of t for Antenna in Fig. 2(c) with W = 18 mm.

t (mm)	-2	0	2	4
G_{max} (dB)	12.21	9.58	14.24	14.39
G_{max} frequency (GHz)	13	13	11.5	11.5

Another antenna is designed for a high gain radiation patterns. A yagi-Uda antenna is designed using NEC-WIN software. The standard design and its radiation patterns and properties are shown in Fig. 4. Its dimensions can be calculated as follow:

Antenna length (La) = 0.45□ – 0.49□
Director spacing (Sd) = 0.3□□□
Reflector spacing (Sr) = 0.25□
Director length (Ld)□□□□□□
Reflector length (Lr) = 0.5□or greater



Max Gain (Front)
Gain = 10.362 dB
Azimuth = 270 Deg
Gain (Back)
Gain = -3.832 dB
Azimuth = 90 Deg
- 3dB (Left of Max)
Gain = 7.158 dB
Azimuth = 245 Deg
- 3dB (Right of Max)
Gain = 7.218 dB
Azimuth = 295 Deg
3 dB BeamWidth
50 Deg
Front/Back Ratio
14.194 dB

Figure 4: Standard Yagi-Uda Antenna Geometry and Results

Two optimized versions of this antenna are modeled using NEC-WIN: one (3-reflector antenna) with 2 more reflectors to improve the front-to-back ratio and the gain, and another one (5-reflector antenna) with 2 extra reflectors placed in the direction of the side lobes to reduce them. These two designs, dimensions, and resulting radiation patterns and properties are shown in Fig. 5. The positions of the reflectors in the 5-reflector antenna are studied to find the optimum values for highest gain. The 5-reflector yagi-uda antenna have a higher gain of 12.75dB than the standard (10.36 dB), better front-to-back ratio of 18.81 dB compared to 14.2 dB, and it is 32% smaller in size small with a 7% bandwidth.

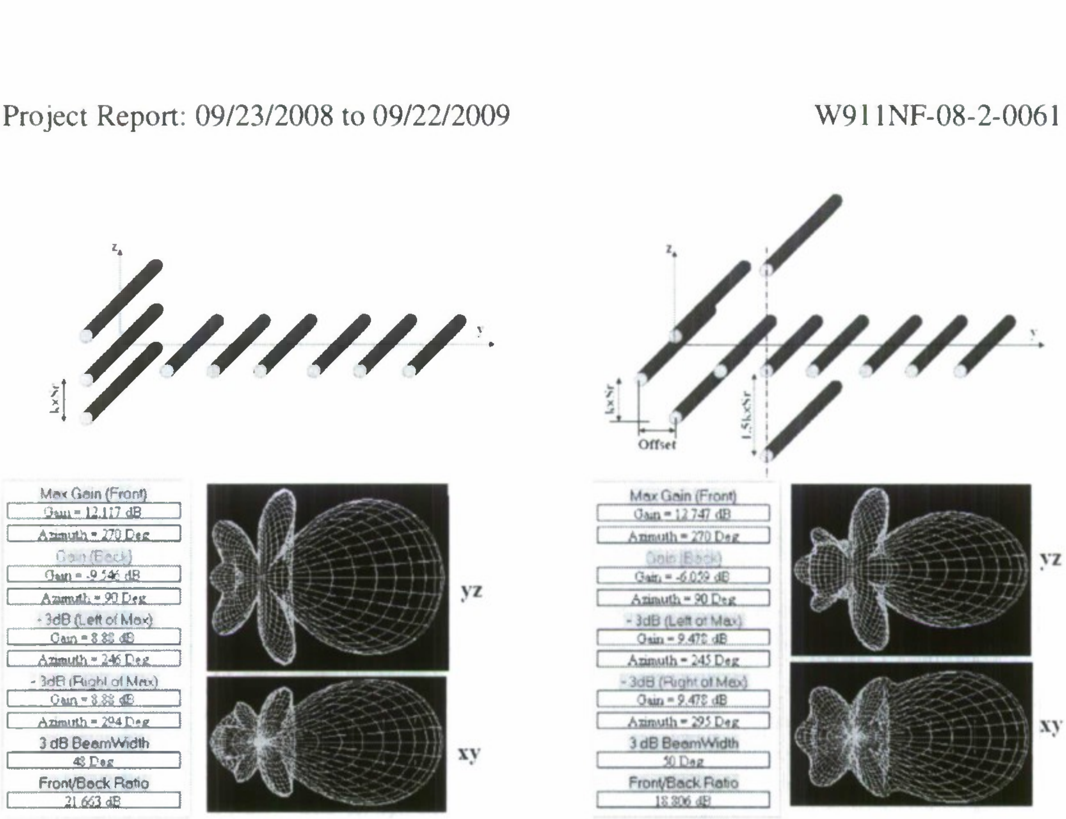


Figure 5: Modified Yagi-Uda Antennas and their Results. La= 135mm, Lr = 155 mm, Sr = 77.5mm, k×Sr = 156mm, Ld = 112mm, Sd = 72mm.

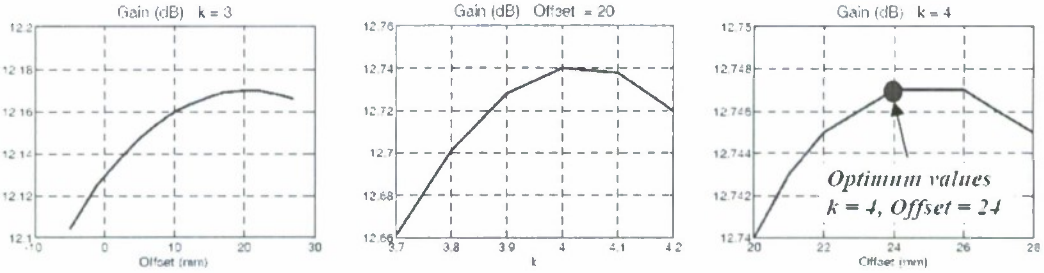
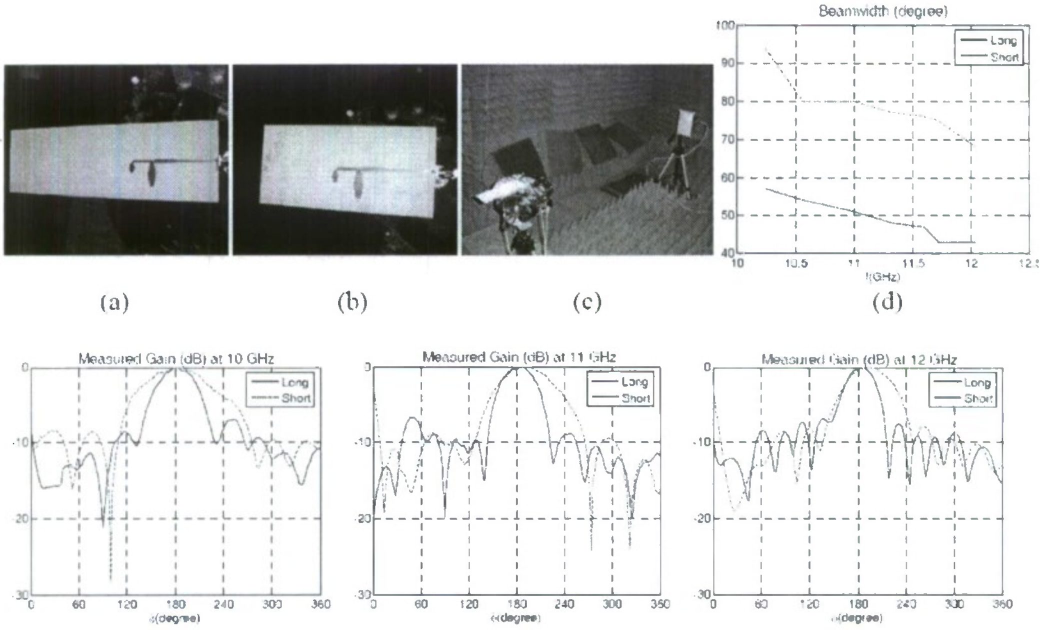


Figure 6: Optimization of the Gain of the 5-Refloctor Antenna using k and Offset.

Task 6: Final Results

In this project, two kinds of antennas were presented with small width, and high gain. The Double Rhombus antenna is wideband and produces high gain that can reach 15 dB. Also, it is printed type of antenna that can be easily integrated with Monolithic Microwave Integrated Circuits (MMIC). The only drawback is the relatively high cost of its substrate. On the other hand, the Yagi Uda with a comparable size can produce about 12.74 dB. It is not expensive since it consists of wires but it can not be fabricated using MMIC and it has narrow bandwidth. Since the proposed wireless network can be used in different frequency range, the Double Rhombus antenna is a better candidate because of it wide bandwidth.

To validate the computed results of the Double Rhombus, the antenna is fabricated using milling machine and its radiation patterns are measured for two prototypes: short one and long one. The measured beamwidth for both antennas along with pictures of the prototypes, and measurement setup and results are depicted in Fig. 7. The long antenna provides around 30^{o} less beamwidth than the short one. The measured radiation patterns at 10, 11 and 12 GHz also shows the improvement in the directivity in the long the antenna.



(a) (b) (c) (d)

(e)

Figure 7: (a) Long Antenna Prototype, (b) Short Antenna Prototype, (c) Measurement Setup, (d) Measured Beamwidth Comparison, and (e) Measured Radiation patterns at 10, 11 and 12 GHz.

References

[1] HFSS: High Frequency Structure Simulator Based on the Finite Element Method, 2008. ver. 11, Ansoft Corp.

[2] A. A. Eldek, "Ultra Wideband Double Rhombus Antenna with Stable Radiation Patterns for Phased Array Applications," *IEEE Trans. Antennas Propagat.*, vol. 55, no. 1, pp. 84-91, Jan. 2007.

[3] A. A. Eldek, "A 100% Bandwidth Microstrip Antenna with Stable Endfire Radiation Patterns for Phased Array Applications," IEEE International Symposium on Antennas and Propagation and URSI National Radio Science Meeting, Albuquerque, NM, pp. 3751-3754, Jul. 2006.

Research Activity 5

MAC Layer Design for a WSN Simulator

Dr. Ali Abu-El Humos
Assistant Professor
Department of Computer Science
Jackson State University
Jackson, MS 39217
Email: ali.a.humos@jsums.edu
Phone: 601-979-3319

I. Breakdown of the Research Activity to Tasks

Task No.	Task	Current Status	Timeline
1	Literature review and problem definition	Completed	11/08- 12/08
2	Simulate a WSN using NS2 current energy model	Completed	12/08-1/09
3	Simulate a WSN using NS2 modified energy model	Completed	2/09-6/09
4	Results, analysis and final report	Completed	4/09-6/09

II. Tasks Description

Task 1: Literature Review and Problem Definition

We discuss the deficiencies of the current implementation of the energy model in NS2 network simulator. We will show how it can be modified; we will then provide some experimental tests to validate our modification to the NS2 energy model.

Wireless Networks Simulation tools are very important to evaluate any protocol or algorithm designed for WSNs in terms of energy, latency, scalability and computability. There are many simulation tools that can be used to evaluate the performance of WSNs protocols. NS2 [1], OMNet++ [2, 3], Glomosim [4], QualNet [1], Jist [5] and TOSSIM [6] are some of many simulators used to test WSNs. All of these simulators (but TOSSIM) existed before the introduction of WSNs. They were used to simulate Mobile Ad hoc Wireless Networks (MANETs) and were later modified to simulate WSNs. Since sleep mode is not an important issue in MANETs, it is not implemented in the energy model design for some of these simulators [7] (e.g. NS2, OMNet++ and Qualnet).

NS2 was designed based on the five-layer Internet Model shown in Figure 1 [8, 9]. Originally, it was used to simulate wired networks and later modified to simulate wireless networks by the CMU Monarch group [10]. The energy model added to NS2 to handle wireless networks has three operation states as shown in Figure 2:

- Transmit
- Receive
- Idle

When a node transmits a packet, the physical layer calls the energy model function *DecrTxEnergy* to decrement the wireless node energy by the energy required to transmit this packet. Similarly, when a node receives a packet, the physical layer calls the energy model function *DecRcvEnergy* to decrement the wireless node energy by the energy required to receive a packet. The node is assumed to be in idle listening mode otherwise, and the energy model function *DecrIdleEnergy* is used to decrement the wireless node energy while it is in idle listening mode.

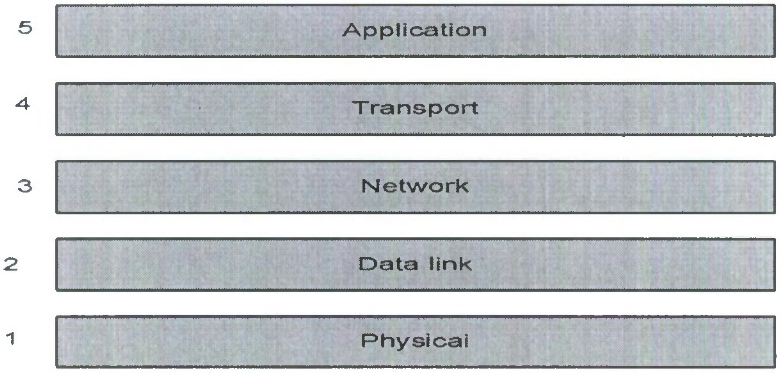


Figure 1: Internet Layer Model

The energy model of NS2 works well when simulating the traditional MANETs. However, when NS2 is used to simulate WSNs it will produce incorrect node energy results. This is because WSNs have a sleep mode which is used to turn the node transceiver off while it is in idle listening mode.

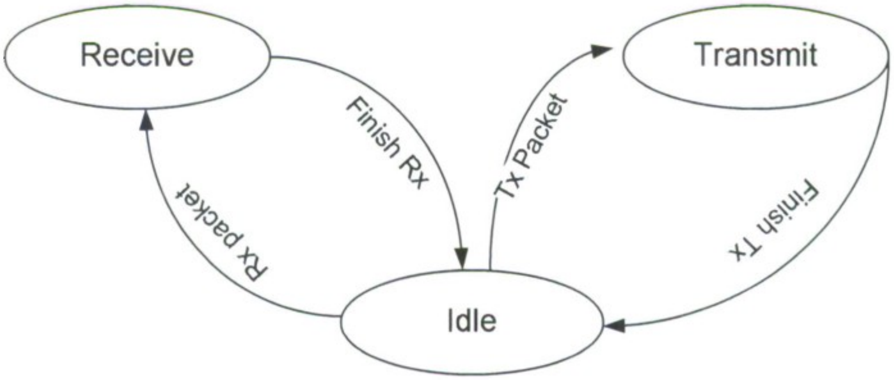


Figure 2: Wireless Node Transceiver States

Task 2: Simulate a WSN in NS2 using its Current Energy Model

We simulated a simple two node network (shown in figure 3) in NS2 using NS2 energy model. The MAC protocol used in the simulation resembles the IEEE 802.11 protocol with on/off switching periods (SMAC protocol). A trace file is generated of all packets generated during the simulation and the idle listening periods. The total consumed energy was calculated from the trace file. A summary of these results are shown in Table 1. More details and analysis of these results will be provided in task 4.

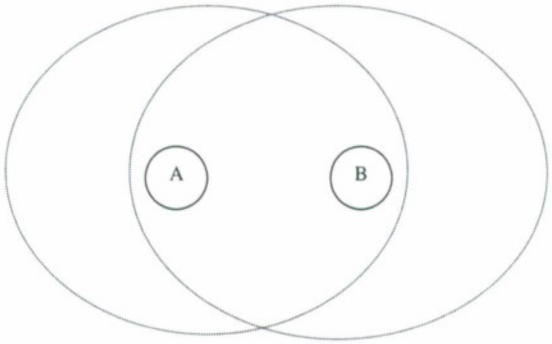


Figure 3: Two Node Setup to Test NS2 Energy Model

Table 1: Simulation Results of Task 2

Event	Event Duration(S)	Number of times event occurred	Total event time(S)	Event Power(W)	Total Event Energy(J)
SYNCPktTX	0.0102	4	0.0408	2	0.0816
SYNCPktRX	0.0102	3	0.0306	1	0.0306
RTSPktTX	0.011	1	0.011	2	0.022
RTSPktRX	0.011	1	0.011	1	0.011
CTSPktTX	0.011	1	0.011	2	0.022
CTSPktRX	0.011	1	0.011	1	0.011
DATAPktTX	0.043	2	0.086	2	0.172
DATAPktRX	0.043	1	0.043	1	0.043
ACKPktTX	0.011	1	0.011	2	0.022
ACKPktRX	0.011	1	0.011	1	0.011
Sleep	0.1432	9	0	0	0
Idle	NA	NA	9.733599	1	9.733599
			9.999999		**10.159799**

Task 3: Simulate a WSN in NS2 using the Modified Energy Model

We will use the same setup in Task 2 for simulation, except we will modify the energy model of NS2. However, this time we expect the total consumed energy calculated from the trace file to match the theoretical calculations.

The problem identified in Task 2 can be fixed as follows: The MAC layer is responsible to turn the node transceiver on and off. The implementation of the MAC layer for any WSN MAC protocol should have the following two functions:

- *Sleep*
- *Wakeup*

When the MAC layer function *Sleep* is called, the node will switch from idle listening mode to sleep mode and start a timer to count how long it will spend in sleep mode. When the MAC layer function *Wakeup* is called, the energy model function *DecrSleepEnergy* will be first called to decrement the wireless node energy during its sleep mode and then the node will switch to idle listening mode again as shown in Figure 3. It will be possible in NS2 to set the sleep power the same way the transmit, receive and idle powers are set using the following set of commands in the *Tcl* script file:

```
$ns_ node-config            -txPower 2.00\
                             -rxPower 1.00\
                             -idlePower 1.00\
                             -sleepPower 0.00
```

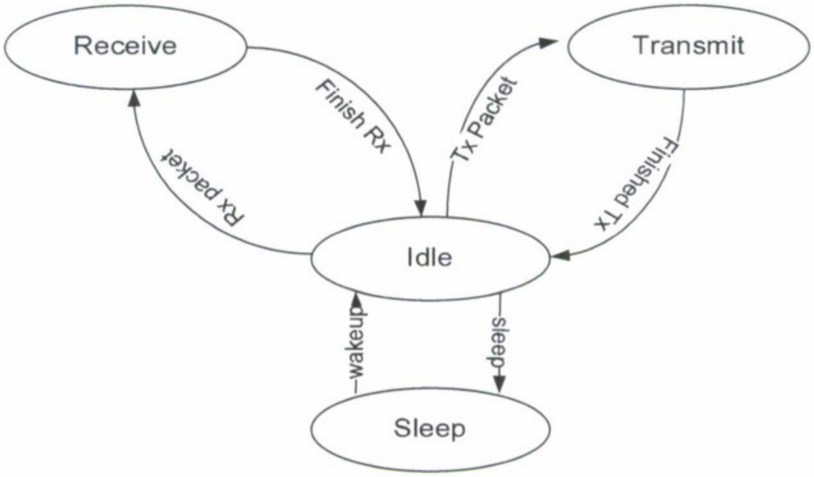


Figure 4: Wireless Sensor Node Transceiver States

In addition, when the transceiver switches from sleep mode into idle mode or vise versa, it consumes some power that needs to be added to the model of Figure 4. Usually, the amount of energy dissipated by the transceiver to switch from sleep to idle is not equal to the energy dissipated to switch from idle to sleep. That is because the switching times from sleep to idle and vise versa are not equal. However, in this simulation, they are assumed to be equal as shown in Figure 5 to simplify the analysis. The simulation results are shown in Table 2.

Usually there is some energy dissipated during sleep mode due to leakage current [11]. Table 3 shows the simulation results when *sleepPower = 0 .05 Watts.*

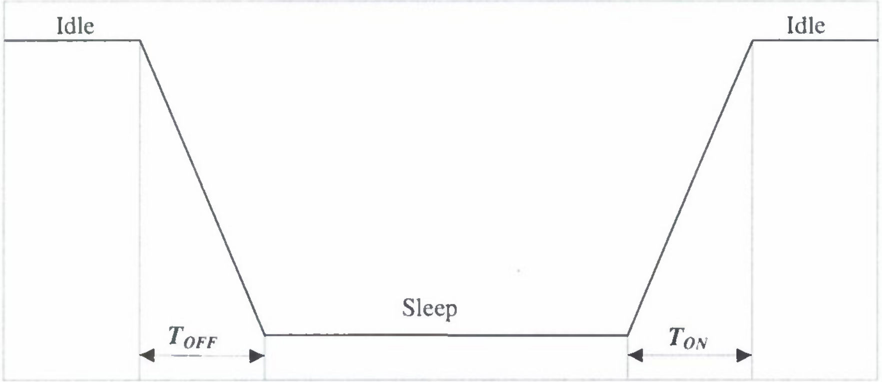


Figure 5: T_{OFF} and T_{ON} Transceiver Switching Times are assumed to be Equal

Table 2: Simulation results of Task 3, *sleepPower* = 0 Watts

Event	**Event Duration(S)**	**Number of times event occurred**	**Total event time(S)**	**Event Power(W)**	**Total Event Energy(J)**
SYNCPktTX	0.0102	4	0.0408	2	0.0816
SYNCPktRX	0.0102	3	0.0306	1	0.0306
RTSPktTX	0.011	1	0.011	2	0.022
RTSPktRX	0.011	1	0.011	1	0.011
CTSPktTX	0.011	1	0.011	2	0.022
CTSPktRX	0.011	1	0.011	1	0.011
DATAPktTX	0.043	2	0.086	2	0.172
DATAPktRX	0.043	1	0.043	1	0.043
ACKPktTX	0.011	1	0.011	2	0.022
ACKPktRX	0.011	1	0.011	1	0.011
Sleep	0.1432	9	1.2888	0	0.000
Idle	NA	NA	8.529939	1	8.529939
SwitchOnOff	0.000001	18	0.000018	0.06	0.00000108
Total			**10.085157**		**8.95614008**

Table 3: Simulation results of Task 3, *sleepPower* = 0 .05 Watts

Event	**Event Duration(S)**	**Number of times event occurred**	**Total event time(S)**	**Event Power(W)**	**Total Event Energy(J)**
SYNCPktTX	0.0102	4	0.0408	2	0.0816
SYNCPktRX	0.0102	3	0.0306	1	0.0306
RTSPktTX	0.011	1	0.011	2	0.022
RTSPktRX	0.011	1	0.011	1	0.011
CTSPktTX	0.011	1	0.011	2	0.022
CTSPktRX	0.011	1	0.011	1	0.011
DATAPktTX	0.043	2	0.086	2	0.172
DATAPktRX	0.043	1	0.043	1	0.043
ACKPktTX	0.011	1	0.011	2	0.022
ACKPktRX	0.011	1	0.011	1	0.011
Sleep	0.1432	9	1.2888	0.05	0.06444
Idle	NA	NA	8.529939	1	8.529939
SwitchOnOff	0.000001	18	0.000018	0.06	0.00000108
Total			**10.085157**		**9.02058008**

Task 4: Results, Analysis and Final report

The results obtained in table 1 were for a simulation time of 10 s. The table shows that the node goes to sleep mode 9 times during the simulation. However, it is clear from the table that the time the node spends in sleep mode is not counted for as indicated by the shaded cell. This can be easily verified if the entries under the *Total event time* column are summed up. The summation will be equal to 10 s (**9.999999 s**). This is an inherited error from the original implementation of the energy model of the 802.11 MAC protocol in NS2. When the 802.11 MAC protocol was first implemented in NS2, power consumption during sleep mode was ignored when compared to Idle, Receive and Transmit powers. That was justified for the early 802.11 applications where battery life time was not a concern. In contrast, battery life time is a major concern in WSNs. Consequently, using the original energy model of NS2 to simulate WSNs will produce incorrect energy dissipation results.

The results of table 2 takes into consideration the time a node may spend in sleep mode as indicated by the shaded table cell. Note that the simulation time in this case is **10.085157 s**. It is clear from this table that the new energy model produces the correct results when used to simulate WSNs.

Table 3 shows the simulation results when *sleepPower* is not equal to zero. It should be noted that under the new NS2 energy model, the user can directly change the value of the *sleepPower* in the TCl file. In addition to that, the new energy model can be used with any MAC layer protocol such SMAC and 802.11 MAC protocols.

As future enhancement to the NS2 energy model, CPU packet processing power should be also considered for a more comprehensive energy model for analyzing WSNs.

References

[1] http://www.scalable-networks.com/

[2] S. Dulman, P. Havinga, "A Simulation Template for Wireless Sensor Networks," *Supplement of the Sixth International Symposium on Autonomous Decentralized Systems*, Pisa, Italy, April 2003.

[3] A. Varga, "The OMNeT++ Discrete Event Simulation System," *In European Simulation Multiconference (ESM'01)*, June 2001.

[4] http://pcl.cs.ucla.edu/projects/glomosim/

[5] http://jist.ece.cornell.edu/

[6] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications," *In Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003).*

[7] C. Margi and K. Obraczka, "Instrumenting Network Simulators for Evaluating Energy Consumption in Power-Aware Ad-Hoc Network Protocols," *In Proceedings of the IEEE/ACM MASCOTS*, October 5-7, 2004.

[8] W. Stallings, *Data and Computer Communications*, 7th ed., Prentice Hall, 2004.

[9] A. Tanenbaum, *Computer Networks*, 4th ed., Prentice Hall, 2003.

[10] The Network Simulator ns-2, http://www.isi.edu/nsnam/ns

[11] T. Floyd, *Electronics Fundamentals*, 7th ed., Prentice Hall, 2007.

BUSINESS STATUS REPORT

An account for the Co-operative Agreement Award (W911NF-08-2-0061) has been setup by the Office of Sponsored Programs at Jackson State University. All financial and human resource usage of this project was done as scheduled and as proposed in the original agreement.

I Budget Overview

Item	Total Amount for Use in the Budget	Amount Used	Total Amount not Used
Faculty Salary	$47,941	$46,441	$1,500
Conference Travel	$2,500	$2,490.05 [$2,118.08 for travel and $371.97 for Office Supplies]	$9.95
Publication	$1,500	$1,500 [$1,165.78 for Publication and $334.22 for Office Supplies]	
Office Supplies	$1,000	$968.96	$31.04
Total Direct Costs	**$52,941**	**$51,400.01**	**$1540.99**
Fringe Benefits	$15,341	$15,341	
Total Indirect Costs	$31,409	$31,409	
TOTAL	**$99,691**	**$98,150.01**	**$1540.99**

II Details of Total Amount Used in the Budget

Item	Details	Amount
Faculty Salary	Dr. Natarajan Meghanathan	$20,441
	Dr. Ali Abu-El Humos	$6,000
	Dr. Kamal Ali	$6,000
	Dr. Abdelnasser Eldek	$6,000
	Dr. Loretta Moore	$5,000
	Dr. Gordon Skelton	$1,500
	Dr. Tarek El-Bawab	$1,500
	Total Amount Used for Faculty Salary	**$46,441**
Conference Travel	Travel and Registration to the International Conference on Wireless Networks (ICWN 2009) – Las Vegas, Dr. Meghanathan	$1618.08
	Registration Fee for International Conference on Signal Processing and Communication Systems – Omaha Nebraska, Dr. Meghanathan	$500
	Total Amount Used for Conference Travel	**$2,118.08**

Publication	WASA 09 Registration/ Publication Fee	$600
	ISAST Transactions Journal Publication Fee	$565.78
	Total Amount Used for Publication	**$1,165.78**
Office Supplies	Black Toner Cartridge	$85
	Color Toner Cartridge	$211.97
	Binders, Pad Folios	$328.62
	White Papers	$108.20
	USB Flash Drives	$311.54
	Toner	$219.98
	Foam Poly Spiral Expanding File	$43.78
	File Cabinets	$196.00
	CD R/W Discs	$38.05
	Ethernet Cables	$25.86
	Eraser Marker Sets	$61.04
	Portable File Box	$45.21
	Total Amount Used for Office Supplies	**$1676.05**

III Explanation for Total Amount not Used

- Dr. Shahrouz Aliabadi served in the Project Steering Committee. During Spring 2009, he indicated that his salary of $1,500 would not be required and it could be reprogrammed for any high priority requirement the project might experience. The funds were not reprogrammed. So, they remain unexpended.

- The remaining amount of $40.99 (from the Conference Travel, Publication and Office Supplies budget lines) was left as a backup amount in the Co-operative Agreement, for any unexpected change in the prices of the items under requisition and for any possible shipping charges that may be incurred. Neither situation did occur; this amount was left unexpended.

APPENDIX – Publications

I Peer-reviewed Journal Publications

J1. ISAST Transactions on Computers and Intelligent Systems

Multicast Extensions to the Location-Prediction Based Routing Protocol for Mobile Ad hoc Networks

Natarajan Meghanathan

Abstract— **We propose multicast extensions to the location prediction-based routing protocol (referred to as NR-MLPBR and R-MLPBR) for mobile ad hoc networks to simultaneously reduce the number of tree discoveries and the hop count per path from the source to each of the receivers of the multicast group. Nodes running NR-MLPBR are not aware of the receivers of the multicast group. R-MLPBR assumes that each receiver node also knows the identity of the other receiver nodes of the multicast group. The multicast extensions work as follows: Upon failure of a path to the source, a receiver node attempts to locally construct a global topology using the location and mobility information collected during the latest global broadcast tree discovery. NR-MLPBR attempts to predict a path that has the minimum number of hops to the source and R-MLPBR attempts to predict a path to the source that has the minimum number of non-receiver nodes. If the predicted path exists in reality, the source accommodates the path as part of the multicast tree and continues to send the multicast packets in the modified tree. Otherwise, the source initiates another global broadcast tree discovery. Simulation studies illustrate that NR-MLPBR and R-MLPBR simultaneously minimize the number of global broadcast tree discoveries as well as the hop count per source-receiver path in the multicast trees. In addition, R-MLPBR determines multicast trees with relatively reduced number of links.**

Index Terms— **Multicast Routing, Mobile Ad hoc Networks, Link Efficiency, Hop Count, Simulation**

I. INTRODUCTION

A mobile ad hoc network (MANET) is a dynamic distributed system of wireless nodes that move independent of each other in an autonomous fashion. The network bandwidth is limited and the medium is shared. As a result, transmissions are prone to interference and collisions. The battery power of the nodes is constrained and hence nodes operate with a limited transmission range, often leading to multi-hop routes between any pair of nodes in the network. Due to node mobility, routes between any pair of nodes frequently change and need to be reconfigured. As a result, on-demand route discovery (discovering a route only when required) is often preferred over periodic route discovery and maintenance, as the latter strategy will incur significant overhead due to the frequent exchange of control information among the nodes [1]. We hence deal with on-demand routing protocols for the rest of this paper.

In an earlier work [2], we developed a location prediction based routing (LPBR) protocol for unicast routing in MANETs. The specialty of LPBR is that it attempts to simultaneously reduce the number of global broadcast route discoveries as well as the hop count of the paths for a source-destination session. LPBR works as follows: During a regular flooding-based route discovery, LPBR collects the location and mobility information of the nodes in the network and stores the collected information at the destination node of the route search process. When the minimum-hop route discovered through the flooding-based route discovery fails, the destination node attempts to predict the current location of each node using the location and mobility information collected during the latest flooding-based route discovery. A minimum hop path Dijkstra algorithm [3] is run on the locally predicted global topology. If the predicted minimum hop route exists in reality, no expensive flooding-based route discovery is needed and the source continues to send data packets on the discovered route; otherwise, the source initiates another flooding-based route discovery.

Multicasting is the process of sending a stream of data from one source node to multiple recipients by establishing a routing tree, which is an acyclic connected subgraph containing all the nodes in the tree. The set of receiver nodes form the multicast group. While propagating down the tree, data is duplicated only when necessary. This is better than multiple unicast transmissions. Multicasting in ad hoc wireless networks has numerous applications [4]: collaborative and distributing computing like civilian operations, emergency search and rescue, law enforcement, warfare situations and etc.

Several MANET multicast routing protocols have been proposed in the literature [4]. They are mainly classified as: tree-based and mesh-based protocols. In tree-based protocols, only one route exists between a source and a destination and hence these protocols are efficient in terms of the number of link transmissions. The tree-based protocols can be further divided into two types: source tree-based and shared tree-based. In source tree-based multicast protocols, the tree is rooted at the source. In shared tree-based multicast protocols,

---

This research has been funded by the Army Research Lab through grant number W911NF-08-2-0061.

Natarajan Meghanathan is with the Department of Computer Science, Jackson State University, Jackson, MS 39056 USA (phone: 601-979-3661; fax: 601-979-2478; e-mail: nmeghanathan@jsums.edu).

the tree is rooted at a core node and all communication between the multicast source and the receiver nodes is through the core node. Even though shared tree-based multicast protocols are more scalable with respect to the number of sources, these protocols suffer under a single point of failure, the core node. On the other hand, source tree-based protocols are more efficient in terms of traffic distribution. In mesh-based multicast protocols, multiple routes exist between a source and each of the receivers of the multicast group. A receiver node receives several copies of the data packets, one copy through each of the multiple paths. Mesh-based protocols provide robustness at the expense of a larger number of link transmissions leading to inefficient bandwidth usage. Considering all the pros and cons of these different classes of multicast routing in MANETs, we feel the source tree-based multicast routing protocols are more efficient in terms of traffic distribution and link usage. Hence, all of our work in this research will be in the category of on-demand source tree-based multicast routing.

In this paper, we propose two multicast extensions to LPBR, referred to as NR-MLPBR and R-MLPBR. Both the multicast extensions are aimed at minimizing the number of global broadcast tree discoveries as well as the hop count per source-receiver path of the multicast tree. They use a similar idea of letting the receiver nodes to predict a new path based on the locally constructed global topology obtained from the location and mobility information of the nodes learnt through the latest broadcast tree discovery. Receiver nodes running NR-MLPBR (Non-Receiver aware Multicast extensions of LPBR) are not aware of the receivers of the multicast group, whereas each receiver node running R-MLPBR (Receiver-aware Multicast Extension of LPBR) is aware of the identity of the other receivers of the multicast group. NR-MLPBR attempts to predict a minimum hop path to the source, whereas R-MLPBR attempts to predict a path to the source that has the minimum number of non-receiver nodes. If more than one path has the same minimum number of non-receiver nodes, then R-MLPBR breaks the tie among such paths by choosing the path with the minimum number of hops to the source. Thus, R-MLPBR is also designed to reduce the number of links in the multicast tree, in addition to the average hop count per source-receiver path and the number of global broadcast tree discoveries.

The rest of the paper is organized as follows: Section II provides the detailed design of the two multicast extensions. Section III explains the simulation environment and reviews the MAODV and BEMRP protocols that are studied along with NR-MLPBR and R-MLPBR as part of our simulation studies. In Section IV, we illustrate and explain simulation results for the four multicast routing protocols (MAODV, NR-MLPBR, R-MLPBR and BEMRP) with respect to different performance metrics. Section V concludes the paper.

II. MULTICAST EXTENSIONS TO LPBR

The objective of the multicast extensions to LPBR (referred to as NR-MLPBR and R-MLPBR) is to simultaneously minimize the number of global broadcast tree discoveries as well as the hop count per source-receiver path. In addition, R-MLPBR aims to also reduce the number of links that are part of the multicast tree. The Non-Receiver aware Multicast extension to LPBR (NR-MLPBR) does not assume the knowledge of the receiver nodes of the multicast group at every receiver node. Each receiver node running R-MLPBR learns the identity information of peer receiver nodes through the broadcast tree discovery procedure. Both the multicast extensions assume the periodic exchange of beacons in the neighborhood. This is essential for nodes to learn about the moving away of the downstream nodes in the multicast tree. We assume that a multicast group comprises basically of receiver nodes that wish to receive data packets from an arbitrary source, which is not part of the multicast group.

A. *Broadcast of Multicast Tree Request Messages*

Whenever a source node has data packets to send to a multicast group and is not aware of a multicast tree to the group, the source initiates a broadcast tree discovery procedure by broadcasting a Multicast Tree Request Message (MTRM) to its neighbors. The source maintains a monotonically increasing sequence number for the broadcast tree discoveries it initiates to form the multicast tree. Each node, including the receiver nodes of the multicast group, on receiving the first MTRM of the current broadcast process (i.e., a MTRM with a sequence number greater than those seen before), includes its Location Update Vector, LUV in the MTRM packet. The LUV of a node comprises the following: node ID, X, Y co-ordinate information, Is Receiver flag, Current velocity and Angle of movement with respect to the X-axis. The *Is Receiver* flag in the LUV, if set, indicates that the node is a receiving node of the multicast group. The node ID is also appended on the "Route record" field of the MTRM packet. The structure of the LUV and the MTRM is shown in Figures 1 and 2 respectively.

Node ID	X Co-ordinate	Y Co-ordinate	Node Velocity	Angle of Movement	Is Receiver
4 bytes	8 bytes	8 bytes	8 bytes	8 bytes	1 bit

Figure 1: Location Update Vector (LUV) per Node

Multicast Source	Multicast Group ID	Sequence Number	Route Recorded (List of Node IDs)	Location Update Vectors (LUVs)
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes	Variable Size of 36 bytes, 1 bit

Figure 2: Structure of the Multicast Tree Request Message

Multicast Source	Originating Receiver	Multicast Group ID	Sequence Number	Route Record from the Receiver to the Source
4 bytes	4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes

Figure 3: Structure of Multicast Tree Establishment Message

Key	Value			
<Source, Multicast Group ID>	$<d_a, r_a>$	$<d_b, r_b>$	<.... ...>	<.... ...>

Figure 4: Structure of the Multicast Routing Table at an Intermediate Node

B. Construction of the Multicast Tree through the Multicast Tree Establishment Message

Paths constituting the multicast tree are independently chosen at each receiver node. A receiver node gathers several MTRMs obtained across different paths and selects the minimum hop path among them by looking at the "Route Record" field in these MTRMs. A Multicast Tree Establishment Message (MTEM) is sent on the discovered minimum hop route to the source. The MTEM originating from a receiver node has the list of node IDs corresponding to the nodes that are on the minimum hop path from the receiver node to the source (which is basically the reverse of the route recorded in the MTRM). The structure of the MTEM packet is shown in Figure 3.

An intermediate node upon receiving the MTEM packet checks its multicast routing table whether there exist an entry for the <Multicast Source, Multicast Group ID> in the table. If an entry exists, the intermediate node merely adds the tuple <One-hop sender of the MTEM, Originating Receiver node of the MTEM> to the list of <Downstream node, Receiver node> tuples for the multicast tree entry and does not forward the MTEM further. The set of downstream nodes are part of the multicast tree rooted at the source node for the multicast group. If a <Multicast Source, Multicast Group ID> entry does not exist in the multicast routing table, the intermediate node creates an entry and initializes it with the <One-hop sender of the MTEM, Originating Receiver node of the MTEM> tuple. Note that the one-hop sender of the MTEM is learnt through the MAC (Medium Access Control) layer header and verified using the Route Record field in the MTEM. The intermediate node then forwards the MTEM to the next downstream node on the path towards the source. The structure of the multicast routing table at a node is illustrated in Figure 4. Note that the tuples $<d_a, r_a>$, $<d_b, r_b>$, <…, …> indicate the downstream node d_a for receiver node r_a, downstream node d_b for receiver node r_b and so on. A node could be the downstream node for more than one receiver node. The source node maintains a multicast routing table that has the list of <Downstream node, Receiver node> tuples for each of the multicast groups to which the source is currently communicating through a multicast session. For each MTEM received, the source adds the neighbor node that sent the MTEM and the corresponding Originating Receiver node to the list of <Downstream node, Receiver node> tuples for the multicast group.

C. Multicast Tree Acquisition and Data Transmission

After receiving the MTEMs from all receiver nodes within a certain time called Tree Acquisition Time (*TAT*), the source starts sending the data packets on the multicast tree. The *TAT* is based on the maximum possible diameter of the network (an input parameter in our simulations). The diameter of a network is the maximum of the hop count of the minimum hop paths between any two nodes in the network. The *TAT* is dynamically set at a node based on the time it took to receive the first MTEM for a broadcast tree discovery procedure.

The structure of the header of the multicast data packet is shown in Figure 5. The source and destination fields in the header include the identification for the source node and the multicast group ID respectively. The sequence number field in the header can be used by the receivers to accumulate and reorder the multicast data packets, incase if they are received out of order. In addition to these regular fields, the header of the multicast data packet includes three specialized fields: the 'More Packets' (*MP*) field, the 'Current Dispatch Time' (*CDT*) field and the 'Time Left for Next Dispatch' (*TNLD*) field. The *CDT* field stores the time as the number of milliseconds lapsed since Jan 1, 1970, 12 AM. These additional overhead (relative to that of the other ad hoc multicast routing protocols) associated with the header of each data packet amounts to only 12 more bytes per data packet.

Multicast Source	Multicast Group ID	Sequence Number	More Packets	Current Dispatch Time	Time Left for Next Dispatch
4 bytes	4 bytes	4 bytes	1 bit	8 bytes	4 bytes

Figure 5: Structure of the Header of the Multicast Data Packet

The source sets the *CDT* field in all the data packets sent. In addition, if the source has any more data to send, it sets the *MP* flag to 1 and sets the appropriate value for the *TLND* field, which indicates the number of milliseconds since the *CDT*. If the source does not have any more data to send, it will set the *MP* flag to 0 and leaves the *TLND* field blank. As we assume the clocks across all nodes are synchronized, a receiver node will be able to calculate the end-to-end delay for the data packet based on the time the data packet reaches the node and the *CDT* field in the header of the data packet. Several clock synchronization algorithms (example [5][6]) have been proposed for wireless ad hoc networks. The receiver node computes and maintains the average end-to-and delay per data packet for the current path to the source by recording the sum of the end-to-end delays of all the data packets received so far on the path and the number of data packets received on the path. Accordingly, the average end-to-end delay per data packet for the current path is updated every time after receiving a new data packet on the path. If the source node has set the MP flag, the receiver node computes the 'Next Expected Packet Arrival Time' (*NEPAT*), which is *CDT* field + *TLND* field + 2*Average end-to-end delay per data packet. A timer is started for the *NEPAT* value. Since, we are using only the average end-to-end delay per data packet to measure the *NEPAT* value, the variations in the end-to-end delay of particular data packets will not very much affect the *NEPAT* value. So, the source and receiver nodes need not be perfectly synchronized. The clocks across the nodes can have small drifts and this would not very much affect the performance of the multicast extensions of LPBR.

D. Multicast Tree Maintenance

We assume that each node periodically exchanges beacon messages with its neighbors, located within its default maximum transmission range. If an intermediate node notices that its link with a downstream node has failed (i.e., the two nodes have moved away and are no longer neighbors), the intermediate node generates and sends a Multicast Path Error Message (MPEM) to the source node of the multicast group entry. The MPEM has information about the receiver nodes affected (obtained from the multicast routing table) because of the link failure with the downstream node. Figure 6 shows the structure of an MPEM. The intermediate node removes the tuple(s) corresponding to downstream node(s) and the affected receiver node(s). After these deletions, if no more <Downstream node, Receiver node> tuple exists for a <Source node, Multicast group ID> entry, the intermediate node removes the entire row for this entry from the routing table.

Multicast Source	Originating Intermediate Node	Multicast Group ID	IDs of Affected Receivers
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes

Figure 6: Structure of a MPEM Message

The source, upon receiving the MPEM, will wait to receive a Multicast Predicted Path Message (MPPM) from each of the affected receivers, within a MPPM-timer maintained for each receiver. The source estimates a Tree-Repair Time (*TRT*) for each receiver as the time that lapsed between the reception of the MPEM from an intermediate node and the MPPM from the affected receiver. An average value for the TRT per receiver is maintained at the source as it undergoes several path failures and repairs before the next global broadcast based tree discovery. The MPPM-timer (initially set to the time it took for the source to receive the MTEM from the receiver) for a receiver will be then set to 1.5* Average *TRT* value, so that we give sufficient time for the destination to learn about the route failure and generate a new MPPM. Nevertheless, this timer will be still far less than the tree acquisition time that would be incurred if the source were to launch a global broadcast tree discovery. Hence, our approach will only increase the network throughput and does not decrease it.

E. Prediction of Node Location using the LUVs

If a multicast receiver does not receive the data packet within the *NEPAT* time, it will attempt to locally construct the global topology using the location and mobility information of the nodes learnt from the latest broadcast tree discovery. Each node is assumed to be moving in the same direction with the same speed as mentioned in its latest LUV. Based on this assumption and information from the latest LUVs, the location of each node at the *NEPAT* time is predicted.

We now explain how to predict the location of a node (say node u) at a time instant *CTIME* based on the LUV gathered from node u at time *STIME*. Let $(X_u^{STIME}, Y_u^{STIME})$ be the X and Y co-ordinates of u at time *STIME*. Let $Angle_u^{STIME}$ and $Velocity_u^{STIME}$ represent the angle of movement with respect to the X-axis and the velocity at which u is moving. The distance traveled by node u from time *STIME* to *CTIME* would be: $Distance_u^{STIME\text{-}CTIME} = (CTIME - STIME + 1) * Velocity_u^{STIME}$.

Let $(X_u^{CTIME}, Y_u^{CTIME})$ be the predicted location of node u at time *CTIME*. The value of X_u^{CTIME} and Y_u^{CTIME} are given by $X_u^{STIME} + Offset\text{-}X_u^{CTIME}$ and $Y_u^{STIME} + Offset\text{-}Y_u^{CTIME}$ respectively. The offsets in the X and Y-axes, depend on angle of movement and the distance traveled, and are calculated as follows:

$$Offset\text{-}X_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \cos(Angle_u^{STIME})$$
$$Offset\text{-}Y_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \sin(Angle_u^{STIME})$$

$$X_u^{CTIME} = X_u^{STIME} + Offset\text{-}X_u^{CTIME}$$
$$Y_u^{CTIME} = Y_u^{STIME} + Offset\text{-}Y_u^{CTIME}$$

We assume each node is initially configured with information regarding the network boundaries, given by [0, 0], $[X_{max}, 0]$, $[X_{max}, Y_{max}]$ and $[0, Y_{max}]$. When the predicted X and/or Y co-ordinate is beyond the network boundaries, we set their values to the boundary conditions as stated below.

If $(X_u^{CTIME} < 0)$, then $X_u^{CTIME} = 0$;
If $(X_u^{CTIME} > X_{max})$, then $X_u^{CTIME} = X_{max}$
If $(Y_u^{CTIME} < 0)$, then $Y_u^{CTIME} = 0$;
If $(Y_u^{CTIME} > Y_{max})$, then $Y_u^{CTIME} = Y_{max}$

Based on the predicted locations of each node in the network at time *CTIME*, the receiver node locally constructs the global topology. Note that there exists an edge between two nodes in the locally constructed global topology, if the predicted distance between the two nodes (with the location information obtained from the *LUV*) is less than or equal to the transmission range of the nodes. The two multicast extensions NR-MLPBR and R-MLPBR differ from each other on the nature of the paths predicted at the receiver node.

Multicast Source	Originating Receiver Node	Multicast Group ID	Predicted Path to the Multicast Source (List of Node IDs)
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes

Figure 7: Structure of the Multicast Predicted Path Message

F. NR-MLPBR: Multicast Path Prediction

The receiver node locally runs the Dijkstra's minimum hop path algorithm [3] on the predicted global topology. If at least one path exists from the source node to the receiver node in the generated topology, the algorithm returns the minimum hop path among them. The receiver node then sends a MPPM (structure shown in Figure 7) on the discovered path with the route information included in the message.

G. R-MLPBR: Multicast Path Prediction

The receiver node uses the LUV obtained from each of the intermediate nodes during the latest global tree broadcast discovery to learn about the identification of its peer receiver nodes that are part of the multicast group. If there existed a direct path to the source on the predicted topology, the receiver chooses that path as the predicted path towards the source. Otherwise, the receiver determines a set of node-disjoint paths on the predicted global topology. The node-disjoint paths to the source are ranked depending on the number of non-receiver nodes that act as intermediate nodes on the path. The path that has the least number of non-receiver nodes as intermediate nodes is preferred. The reason is a path that has the least number of non-receiver nodes is more likely to be a minimum hop path and if a receiver node lies on that path, the number of newly added links to the tree would also be reduced. R-MLPBR thus aims to discover paths with the minimum hop count and at the same time attempts to conserve bandwidth by reducing the number of links that get newly added to the tree as a result of using the predicted path. The

MPPM is hence sent on the predicted path that has minimum number of non-receiver nodes. If two or more paths has the same minimum number of non-receiver nodes, R-MLPBR breaks the tie by choosing the path with the minimum hop count to the source. Figure 8 illustrates the algorithm used by R-MLPBR at a receiver node to select the best predicted path to the source.

```
Input: Graph $G(V, E)$, Set of Multicast receivers $M_R$, source $s$ and receiver $d$
Output: $s$-$d$ path
Auxiliary Variables: Graph $G''(V'', E'')$, Set of Node-disjoint paths $P_N$
Initialization: $G''(V'', E'') \leftarrow G(V, E)$, $P_N \leftarrow \varphi$.

Begin
while ( $\exists$ at least one $s$-$d$ path in $G''$)
    $p \leftarrow$ Minimum hop $s$-$d$ path in $G''$.
    if (hop count of $p = 1$)
        return $p$
    end if
    $P_N \leftarrow P_N \cup \{p\}$
    $\forall$ $vertex, v \in p, v \neq s,d$   $G''(V'',E'') \leftarrow G''(V''-\{v\},E''-\{e\})$
    $edge, e \in Adj-list(v)$
end while
minNonReceivers $\leftarrow \infty$ // the count for the minimum number of non-receivers is initialized to $\infty$.
bestPath $\leftarrow$ NULL // the best path is initialized to NULL
minHops $\leftarrow \infty$ // the minimum hop count of the best path initialized to $\infty$ (a very large value).
for ( $\forall$ path $p \in P_N$)
    countPathNonReceivers $\leftarrow$ 0 // keeps track of the number of non-receiver nodes in path $p$
    for ( $\forall$ intermediate node $n \in p$)
        if ($n \notin M_R$)
            countPathNonReceivers $\leftarrow$ countPathNonReceivers + 1
        end if
    end for
    if (minNonReceivers $\geq$ countPathReceivers)
        if (minNonReceivers = countPathReceivers AND minHops > hop count of $p$)
            bestPath $\leftarrow p$
            minHops $\leftarrow$ hop count of $p$
        end if
        if (minNonReceivers > countPathReceivers)
            minNonReceivers $\leftarrow$ countPathReceivers
            bestPath $\leftarrow p$
            minHops $\leftarrow$ hop count of $p$
        end if
    end if
end for
return bestPath
End
```

Figure 8: R-MLPBR Predicted Path Selection Algorithm

Note that we designed R-MLPBR to choose the path with the minimum number of non-receiver nodes, rather than the path with the maximum number of receiver nodes, as the latter design has the possibility of yielding paths with significantly larger hop count from the source to the receiver node without any guarantee on the possible reduction in the number of links. Our design of choosing the path with the minimum number of non-receiver nodes helps to maintain the hop count per source-receiver path close to that of the minimum hop count and at the same time does helps to reduce the number of links in the tree to a certain extent.

H. Propagation of the Multicast Predicted Path Message towards the Source

An intermediate node on receiving the MPPM, checks its multicast routing table if there already exists an entry for the source node and the multicast group to which the MPPM belongs to. If an entry exists, the intermediate node merely adds the tuple <One-hop sender of the MPPM, Originating Receiver node of the MPPM> to the list of <Downstream node, Receiver node> tuples for the multicast tree entry. If the <Multicast Source, Multicast Group ID> entry does not exist in the multicast routing table, the intermediate node creates an entry and initializes it with the <One-hop sender of the MPPM, Originating Receiver node of the MPPM> tuple. In either case, the MPPM is then forwarded to the next downstream node on the path towards the source. If the source node receives the MPPM from the appropriate receiver node before the MPPM-timer expires, it indicates that the predicted path does exist in reality. A costly global broadcast tree discovery has been thus avoided. The source continues to send the data packets down the multicast tree. The source node estimates the Tree Repair Time (TRT) as the time lapsed between the reception of the MPEM from an intermediate node and the MPPM from the appropriate receiver node. An average value of the TRT for each receiver node is thus maintained at the source as it undergoes several route failures and repairs before the next global broadcast-based tree discovery.

I. Handling Prediction Failure

If an intermediate node attempting to forward the MPPM of a receiver node could not successfully forward the packet to the next node on the path towards the source, the intermediate node informs the absence of the route through a MPPM-Error packet (structure shown in Figure 9) sent back to the receiver node. The receiver node on receiving the MPPM-Error packet discards all the LUVs and does not generate any new MPPM. The receiver will wait for the multicast source to initiate a global broadcast-based tree discovery. After the MPPM-timer expires, the multicast source initiates a new global broadcast-based tree discovery procedure.

Node Sending the MPPM-Error Packet	Multicast Source	Originating Receiver Node	Multicast Group ID	Sequence No. of latest MTRM
4 bytes	4 bytes	4 bytes	4 bytes	4 bytes

Figure 9: Structure of the MPPM-Error Packet

III. SIMULATION ENVIRONMENT AND PROTOCOL REVIEW

The network dimension used is a 1000m x 1000m square network. The transmission range of each node is assumed to be 250m. The number of nodes used in the network is 25 and 75 nodes representing networks of low and high density with an average distribution of 5 and 15 neighbors per node respectively. Initially, nodes are uniformly randomly distributed in the network. We compare the performance of NR-MLPBR and R-MLPBR with that of the minimum-hop based MAODV and the link-efficient BEMRP protocols. We implemented all of these four multicast routing protocols in a discrete-event simulator developed in Java. The broadcast tree discovery strategy employed is the default flooding approach. The simulation parameters are summarized in Table 1.

Table 1: Simulation Conditions

Network Size	1000m x 1000m	
Number of nodes	25 (low density) and 75 (high density)	
Transmission Range	250 m	
Physical Layer	Signal Propagation Model	Two-ray ground reflection model [7]
MAC Layer	IEEE 802.11 [8]	
	Link Bandwidth	2 Mbps
	Interface Queue	FIFO-based, size 100
Routing Protocols	BEMRP [9], MAODV [10], NR-MLPBR and R-MLPBR	
Broadcast Strategy	Flooding	
Mobility Model	Random Way Point Model [11]	
	Minimum Node Speed, m/s	0 m/s
	Maximum Node Speed, m/s	Low-10; Medium-30; High-50
	Pause Time	0 second
Traffic Model	Constant Bit Rate (CBR), UDP	
	Multicast Group Size (# Receivers)	Small: 2; Medium: 4, 8; High: 12, 24
	Data Packet Size	512 bytes
	Packet Sending Rate	4 Packets/ second

Simulations are conducted with a multicast group size of 2, 4 (small size), 8, 12 (moderate size) and 24 (larger size) receiver nodes. For each group size, we generated 5 lists of receiver nodes and simulations were conducted with each of them. Traffic sources are constant bit rate (CBR). Data packets are 512 bytes in size and the packet sending rate is 4 data packets/second. The multicast session continues until the end of the simulation time, which is 1000 seconds. The node mobility model used is the Random Waypoint model [11]. The transmission energy and reception energy per hop is set at 1.4 W and 1 W respectively. Initial energy at each node is 1000 Joules. Each node periodically broadcasts a beacon message within its neighborhood to make its presence felt to the other nodes in the neighborhood.

A. Multicast Extension of Ad hoc On-demand Distance Vector (MAODV) Routing Protocol

MAODV [10] is the multicast extension of the well-known Ad hoc On-demand Distance Vector (AODV) unicast routing protocol [12]. Here, a receiver node joins the multicast tree through a member node that lies on the minimum-hop path to the source. A potential receiver wishing to join the multicast group broadcasts a *Route-Request* (RREQ) message. If a node receives the RREQ message and is not part of the multicast tree, the node broadcasts the message in its neighborhood and also establishes the reverse path by storing the state information consisting of the group address, requesting node id and the sender node id in a temporary cache. If a node receiving the RREQ message is a member of the multicast tree and has not seen the RREQ message earlier, the node waits to receive several RREQ messages and sends back a *Route-Reply* (RREP) message on the shortest path to the receiver. The member node also informs in the RREP message, the number of hops from itself to the source. The potential receiver receives several RREP messages and selects the member node which lies on the shortest path to the source. The receiver node sends a *Multicast Activation* (MACT) message to the selected member node along the chosen route. The route from the source to receiver is set up when the member node and all the intermediate nodes in the chosen path update their multicast table with state information from the temporary cache. A similar approach can be used in NR-MLPBR and R-MLPBR when a new receiver node wishes to join the multicast group.

B. Bandwidth-Efficient Multicast Routing Protocol (BEMRP)

According to BEMRP [9], a newly joining node to the multicast group opts for the nearest forwarding node in the existing tree, rather than choosing a minimum-hop path from the source of the multicast group. As a result, the number of links in the multicast tree is reduced leading to savings in the network bandwidth. Multicast tree construction is receiver-initiated. When a node wishes to join the multicast group as a receiver, it initiates the flooding of *Join control* packets targeted towards the nodes that are currently members of the multicast tree. On receiving the first *Join control* packet, the member node waits for a certain time before sending a *Reply* packet. The member node sends a *Reply* packet on the path, traversed by the *Join* control packet, with the minimum number of intermediate forwarding nodes. The newly joining receiver node collects the *Reply* packets from different member nodes and would send a *Reserve* packet on that path that has the least number of forwarding nodes from the member node to itself.

C. Performance Metrics

The performance metrics studied through this simulation are the following:

- **Number of Links per Tree:** This is the time averaged number of links in the multicast trees discovered and computed over the entire multicast session. The notion of "time-average" is explained as follows: Let there be multicast trees T1, T2, T3 with 4, 8 and 6 links used for time 12, 6 and 15 seconds respectively, then the time averaged number of links in the multicast trees is given by (4*12+8*6+6*15)/ (12+6+15) = 5.6 and not merely 6.0, which is the average of 4, 8 and 6.
- **Hop Count per Source-Receiver Path**: This is the time averaged hop count of the paths from the source to each receiver of the multicast group and computed over the entire multicast session.
- **Time between Successive Broadcast Tree Discoveries:** This is the time between two successive broadcast tree discoveries, averaged over the entire multicast session. This metric is a measure of the lifetime of the multicast trees discovered and also the effectiveness of the path prediction approach followed in NR-MLPBR and R-MLPBR.
- **Energy Consumed per Node:** This is the sum of the energy consumed at a node due to the transfer of data packets as part of the multicast session, broadcast tree discoveries as well as the periodic broadcast and exchange of beacons in the neighborhood.

IV. SIMULATION RESULTS

The performance results for each metric displayed in Figures 10 through 13 are an average of the results obtained from simulations conducted with 5 sets of multicast groups and 5 sets of mobility profiles for each group size, node velocity and network density values. The multicast source in each case was selected randomly among the nodes in the network and the source is not part of the multicast group. The nodes that are part of the multicast group are merely the receivers.

A. Number of Links per Multicast Tree

The number of links per multicast tree (refer figure 10) is a measure of the efficiency of the multicast routing protocol in reducing the number of link transmissions during the transfer of the multicast data from the source to the receivers of the multicast group. The smaller is the number of links in the tree, the larger the link transmission efficiency of the multicast routing protocol. If fewer links are part of the tree, then the chances of multiple transmissions in the network increase and this increases the efficiency of link usage and the network bandwidth. Naturally, the BEMRP protocol, which has been purely designed to yield bandwidth-efficient multicast trees, discovers trees that have a reduced number of links for all the operating scenarios. This leads to larger hop count per source-receiver paths for BEMRP as observed in figures 11.

R-MLPBR, which has been designed to choose the predicted paths with the minimum number of non-receiver nodes, manages to significantly reduce the number of links vis-à-vis the MAODV and NR-MLPBR protocols. R-MLPBR attempts to minimize the number of links in the multicast tree without yielding to a higher hop count per source-receiver path. But, the tradeoff between the link efficiency and the hop count per source-receiver path continues to exist and it cannot be nullified. In other words, R-MLPBR cannot discover trees that have minimum number of links as well as the minimum hop count per source-receiver path. Nevertheless, R-MLPBR is the first multicast routing protocol that yields trees with the reduced number of links and at the same time, with a reduced hop count (close to the minimum) per source-receiver path.

For a given network density and multicast group size, we do not see any appreciable variation in the number of links per tree for each of the multicast routing protocols studied. As the network density increases, BEMRP attempts to reduce the number of links per tree by incorporating links that can be shared by multiple receivers on the paths towards the source. On the other hand, both MAODV and NR-MLPBR attempt to choose minimum hop paths between the source and any receiver and hence exploit the increase in network density to discover minimum hop paths, but at the cost of the link efficiency. On the other hand, R-MLPBR attempts to reduce the number of links per tree as we increase the network density. For a given multicast group size, the number of links per tree for R-MLPBR is about 4-15%, 8-18% and 10-21% more than that incurred by BEMRP. This shows that R-MLPBR is relatively more scalable, similar to BEMRP, with increase in the network density. For medium and large-sized multicast groups, the number of links per tree for both MAODV and NR-MLPBR is about 7-15%, 17-28% and 22-38% more than that incurred for BEMRP in low, medium and high-density networks respectively. On the other hand, the number of links per tree for R-MLPBR is about 6-15%, 12-18% and 16-21% more than that incurred for BEMRP in low, medium and high-density networks respectively. This shows that R-MLPBR is relatively more scalable, similar to BEMRP, with increase in the network density.

B. Hop Count per Source-Receiver Path

All the three multicast routing protocols – MAODV, NR-MLPBR and R-MLPBR, incur almost the same average hop count per source-receiver and it is considerably lower than that incurred for BEMRP. The hop count per source-receiver path is an important metric and it is often indicative of the end-to-end delay per multicast packet from the source to a specific receiver. BEMRP incurs a significantly larger hop count per source-receiver path and this can be attributed to the nature of this multicast routing protocol to look for trees with a reduced number of links. When multiple receiver nodes have to be connected to the source through a reduced set of links, the hop count per source-receiver path is bound to increase. The hop count per source-receiver path increases significantly as we increase the multicast group size. The hop count per source-receiver path for BEMRP can be as large as 41%, 57% and 59% more than that of the hop count per source-receiver path incurred for the other three multicast routing protocols.

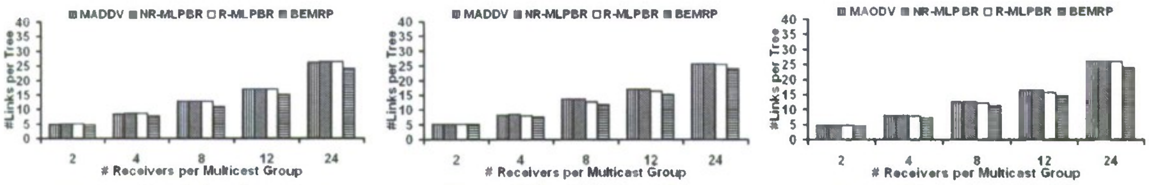


Figure 10.1: 25 nodes, 10 m/s **Figure 10.2:** 25 nodes, 30 m/s **Figure 10.3**: 25 nodes, 50 m/s

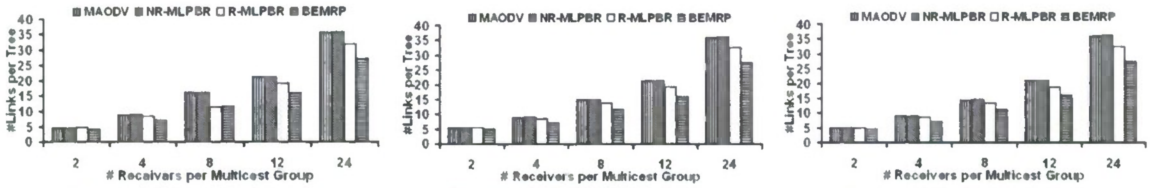


Figure 10.4: 75 nodes, 10 m/s **Figure 10.5:** 75 nodes, 30 m/s **Figure 10.6**: 75 nodes, 50 m/s

Figure 10: Average Number of Links per Multicast Tree (Route Discovery Procedure: Flooding)

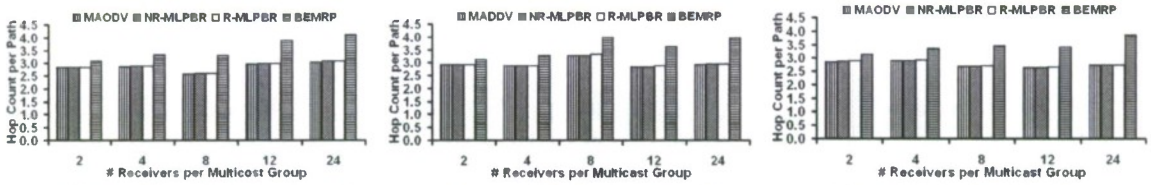


Figure 11.1: 25 nodes, 10 m/s **Figure 11.2:** 25 nodes, 30 m/s **Figure 11.3**: 25 nodes, 50 m/s

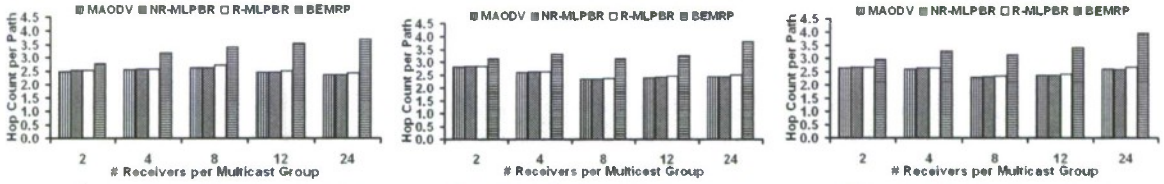


Figure 11.4: 75 nodes, 10 m/s **Figure 11.5:** 75 nodes, 30 m/s **Figure 11.6**: 75 nodes, 50 m/s

Figure 11: Average Hop Count per Source-Receiver Path (Route Discovery Procedure: Flooding)

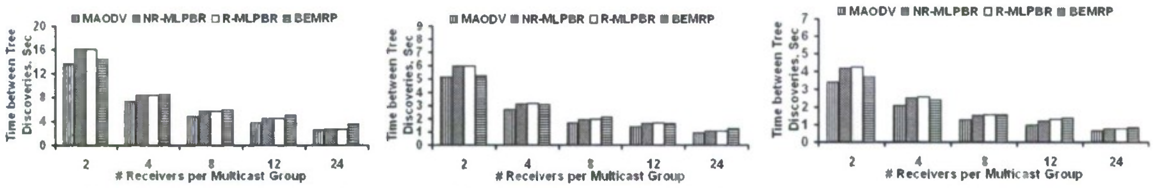


Figure 12.1: 25 nodes, 10 m/s **Figure 12.2:** 25 nodes, 30 m/s **Figure 12.3**: 25 nodes, 50 m/s

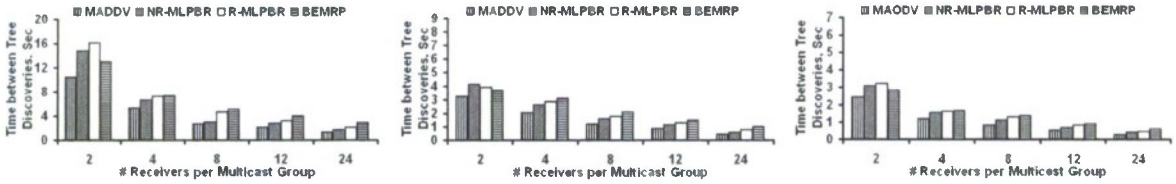


Figure 12.4: 75 nodes, 10 m/s **Figure 12.5:** 75 nodes, 30 m/s **Figure 12.6**: 75 nodes, 50 m/s

Figure 12: Average Time between Successive Tree Discoveries (Route Discovery Procedure: Flooding)

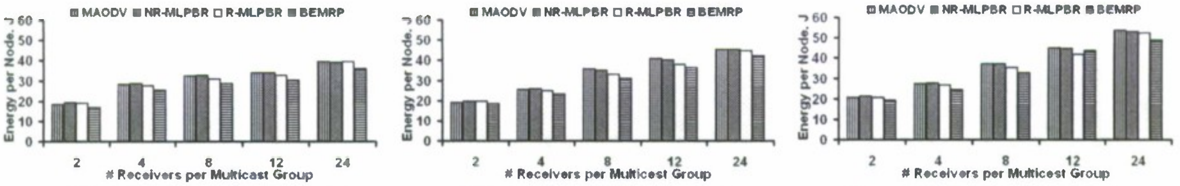


Figure 13.1: 25 nodes, 10 m/s **Figure 13.2:** 25 nodes, 30 m/s **Figure 13.3**: 25 nodes, 50 m/s

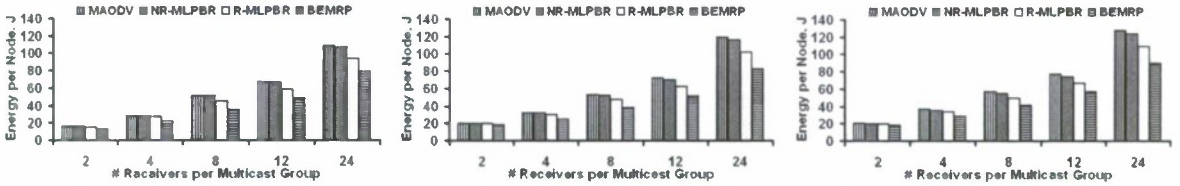


Figure 13.4: 75 nodes, 10 m/s **Figure 13.5:** 75 nodes, 30 m/s **Figure 13.6**: 75 nodes, 50 m/s

Figure 13: Average Energy Consumed per Node (Route Discovery Procedure: Flooding)

For a given network density and group size, there is no appreciable variation in the hop count per source-receiver path for each of the multicast routing protocols studied. As we increase the network density, the hop count per source-receiver path decreases. This is mainly observed in the case of the minimum-hop based MAODV, NR-MLPBR and R-MLPBR. With BEMRP, the impact of network density on the decrease in the hop count is relatively less as it is a bandwidth-efficient multicast routing protocol attempting to reduce the number of links in the tree. The hop count per source-receiver path for BEMRP increased with increase in the multicast group size, while the hop count per source-receiver path for the other multicast routing protocols almost remained the same with increase in multicast group size.

C. *Time between Successive Broadcast Tree Discoveries*

The time between successive broadcast tree discoveries is a measure of the stability of the multicast trees and the effectiveness of the location prediction and path prediction approach of the two multicast extensions. For a given condition of node density and node mobility, both NR-MLPBR and R-MLPBR incur relatively larger time between successive broadcast tree discoveries for smaller and medium sized multicast groups. MAODV tends to be more unstable as the multicast group size is increased, owing to the minimum hop nature of the paths discovered and absence of any path prediction approach. For larger multicast groups, BEMRP tends to perform better by virtue of its tendency to strictly minimize only the number of links in the tree. On the other hand, NR-MLPBR attempts to reduce the hop count per source-receiver path and ends up choosing predicted paths that increase the number of links in the tree, quickly leading to the failure of the tree. The time between successive tree discoveries for R-MLPBR is 15-25%, 15-59% and 20-82% more than that obtained for MAODV in networks of low, moderate and high density respectively. For a given level of node mobility and network density, MAODV trees become highly unstable as the multicast group size increases. For multicast groups of size 2 and 4, the time between successive broadcast tree discoveries for NR-MLPBR and R-MLPBR is greater than that obtained for BEMRP, especially in networks of low and moderate network density. For larger multicast group sizes, BEMRP tends to incur larger time between successive broadcast tree discoveries compared to NR-MLPBR and R-MLPBR. While using a broadcast strategy that will lead to the discovery of inherently stable trees, we conjecture that R-MLPBR will tend to incur larger time between successive broadcast tree discoveries compared to BEMRP, even for larger group sizes.

For each multicast routing protocol, for a given multicast group size and level of node mobility, as the network density increases, the time between successive broadcast tree discoveries decreases. This is mainly observed for the minimum-hop based multicast protocols (especially MAODV and NR-MLPBR) which incur a reduced hop count per source-receiver path as we increase the network density. But, such minimum hop paths obtained in moderate and high-density networks are relatively less stable than those obtained in low-density networks. For a given multicast group size and low node mobility, the time between successive tree discoveries in networks of high density (75 nodes) is 51-80% for MAODV and NR-MLPBR and for R-MLPBR and BEMRP is 70-90% of those obtained in networks of low-density. For a given network density and node mobility, the time between successive route discoveries decreases as the multicast group size increases. For smaller group sizes, the time between successive broadcast tree discoveries for MAODV and BEMRP is respectively about 80%-90% and 85%-94% of that incurred for NR-MLPBR and R-MLPBR. For larger groups, the time between successive tree discoveries for NR-MLPBR and R-MLPBR is respectively about 57%-76% and 75%-80% of that incurred for BEMRP for all network densities.

D. *Energy Consumed per Node*

Energy consumption in multicast routing is directly proportional to the number of links in the tree. Larger the number of links, more the transmissions and more will be the energy consumption in the network and vice-versa. Simulation results in Figure 13 clearly illustrate this. BEMRP incurs the least energy consumption per node and MAODV incurs the largest energy consumption per node. The energy consumed per node for the two multicast extensions is in between these two extremes. The energy consumed per node for R-MLPBR is less than that of NR-MLPBR as the former also attempts to simultaneously reduce the number of links as well as the hop count per source-receiver path. The energy consumption per node increases as the multicast group size increases. For a given multicast group size and multicast routing protocol, the energy consumed per node increases with increase in network density as well as with increase in node mobility.

For a given multicast group size, network density and multicast routing protocol, the energy consumed per node at higher node velocities of 30 m/s and 50 m/s can grow as large as 10-40% of that obtained at maximal node velocity of 10 m/s. BEMRP and MAODV incur the largest increase in energy consumed per node with increase in node mobility. NR-MLPBR and R-MLPBR incur a relatively lower increase in the energy consumed per node with increase in node mobility. This can be attributed to the tendency of these multicast routing protocols to reduce the number of broadcast tree discoveries using effective tree prediction.

For multicast groups of size 2 and 4, as we increase the network density from 25 to 75 nodes, the energy consumed per node decreases. This is due to the smaller group size, leading to the effective sharing of the data forwarding load among all the nodes in the network. For larger group sizes, all the nodes in the network end up spending more energy (due to transmission/reception or at least receiving the packets in the neighborhood). MAODV and NR-MLPBR incur a relatively larger energy consumed per node at high network densities due to the nature of these routing protocols to discover trees with minimum hop count. R-MLPBR and BEMRP discover trees with reduced number of links and hence incur relatively lower energy consumed per node at high network density.

V. CONCLUSIONS

In this paper, we propose multicast extensions to the location prediction based routing (LPBR) protocol for mobile ad hoc networks (MANETs). The multicast extensions of LPBR (referred to as NR-MLPBR and R-MLPBR) have been proposed to simultaneously reduce the number of tree discoveries and the hop count per path from the source to each of the receivers of the multicast group. NR-MLPBR and R-MLPBR differ from each other based on the type of path predicted and notified to the source. NR-MLPBR determines the minimum hop path to the source and sends a Multicast Predicted Path Message on the minimum hop path to the source. R-MLPBR assumes that each receiver knows the identity of the other receivers of the multicast group and hence attempts to choose a path that will minimize the number of newly added intermediate nodes to the multicast tree. R-MLPBR has been thus designed to also reduce the number of links that form the multicast tree, in addition to the source-receiver hop count and the number of global tree discoveries. Nevertheless, the number of links per tree discovered using R-MLPBR is still about 15-20% more than that discovered using BEMRP, but the hop count per source-receiver path is significantly smaller (by about 40%-60%) than those observed in trees discovered using BEMRP and is the same as that discovered using MAODV (aims to minimize the hop count between source-receiver paths). We conjecture that with the deployment of broadcast tree discovery strategies (such as DMEF [13]) that can discover inherently stable trees, the performance of NR-MLPBR and R-MLPBR with respect to the time between successive tree discoveries and energy consumed per node actually improved relatively more than that observed for BEMRP and MAODV. This can be attributed to the effective path prediction of the two multicast extensions, an idea inherited from LPBR.

REFERENCES

[1] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu and J. Jetcheva, "A Performance of Comparison of Multi-hop Wireless Ad hoc Network Routing Protocols," in *Proceedings of the 4th Annual ACM/IEEE Conference on Mobile Computing and Networking*, pp. 85 – 97, October 1998.

[2] N. Meghanathan, "Location Prediction Based Routing Protocol for Mobile Ad Hoc Networks," *Proceedings of the IEEE Global Communications Conference* (GLOBECOM), New Orleans, Nov-Dec 2008.

[3] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms," 2nd Edition, MIT Press/ McGraw Hill, Sept. 2001.

[4] C. S. R. Murthy and B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols," Prentice Hall, June 3, 2004.

[5] J.-P. Sheu, C.-M. Chao, W.-K. Hu and C.-W. Sun, "A Clock Synchronization Algorithm for Multi-hop Wireless Ad hoc Networks," Wireless Personal Communications: An International Journal, vol. 43, no. 2, pp. 185-200, 2007.

[6] D. Zhou, "A Compatible and Scalable Clock Synchronization Protocol in IEEE 802.11 Ad hoc Networks," Proceedings of the International Conference on Parallel Processing, pp. 295-302, Washington DC, USA, June 2005.

[7] L. Breslau, et. al., "Advances in Network Simulation," *IEEE Computer*, vol. 33, no. 5, pp. 59-67, 2000.

[8] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal of Selected Areas in Communication*, vol. 18, no. 3, pp. 535-547, 2000.

[9] T. Ozaki, J-B. Kim and T. Suda, "Bandwidth-Efficient Multicast Routing for Multihop, Ad hoc Wireless Networks," *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies* (INFOCOM), vol. 2, pp. 1182-1192, Anchorage, USA, April 2001.

[10] E. Royer and C. E. Perkins, "Multicast Operation of the Ad-hoc On-demand Distance Vector Routing Protocol," *Proceedings of the 5th ACM/IEEE Annual Conference on Mobile Computing and Networking*, (MOBICOM), pp. 207-218, Seattle, USA, August 1999.

[11] C. Bettstetter, H. Hartenstein and X. Perez-Costa, "Stochastic Properties of the Random-Waypoint Mobility Model," *Wireless Networks*, vol. 10, no. 5, pp. 555-567, 2004.

[12] C. E. Perkins and E. M. Royer, "The Ad hoc On-demand Distance Vector Protocol," Ad hoc Networking, edited by C. E. Perkins, pp. 173-219, Addison-Wesley, 2000.

[13] N. Meghanathan, "A Density and Mobility Aware Energy-Efficient Broadcast Strategy to Determine Stable Routes in Mobile Ad hoc Networks," submitted for journal publication, March 2009.

Dr. Natarajan Meghanathan is currently working as Assistant Professor of Computer Science at Jackson State University, Mississippi, USA, since August 2005. Dr. Meghanathan received his MS and PhD in Computer Science from Auburn University, AL and The University of Texas at Dallas in August 2002 and May 2005 respectively.

Dr. Meghanathan's main area of research is ad hoc networks. He has more than 30 peer-reviewed publications in leading international journals and conferences in this area. Recently, Dr. Meghanathan has received grants from the Army Research Laboratory (ARL) to conduct research on ad hoc routing protocols and from the National Science Foundation (NSF) to conduct a three-year Summer Research Experience for Undergraduates (REU) program at Jackson State University for the years 2009-11. Dr. Meghanathan currently serves as the editor of a number of international journals and also in the program committee and organization committees of several leading international conferences in the area of networks.

I Peer-reviewed Journal Publications

J2. International Journal of Computer Science and Network Security

IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.11, November 2009

A Density and Mobility Aware Energy-Efficient Broadcast Route Discovery Strategy for Mobile Ad hoc Networks

Dr. Natarajan Meghanathan
natarajan.meghanathan@jsums.edu
Jackson State University, Jackson, MS 39217

Summary

We propose a novel network density and mobility aware energy-efficient broadcast route discovery strategy (called DMEF) to determine stable routes in mobile ad hoc networks (MANETs). During the on-demand route discovery process, each node dynamically chooses its own broadcast transmission range for the Route-Request message depending on the perceived number of neighbor nodes in its default maximum transmission range and the node's own mobility values during the time of broadcast. A node surrounded by more neighbors advertises itself only to a limited set of nearby neighbors and a node surrounded by few neighbors advertises itself to a maximum of its neighbors. Similarly, a slow-moving node advertises itself to a majority of its neighbors so that links formed using this node can be more stable. A fast-moving node advertises itself only to the neighbors closer to it. Simulation results indicate that DMEF is very effective, vis-à-vis flooding, in reducing the number of broadcast route discoveries by determining routes with a longer lifetime and as well as in reducing the energy consumed per route discovery. DMEF does not require any changes in the packet headers and can be used with any MANET routing protocol that has been proposed in the literature.

Key words:
Route discovery, Flooding, Energy efficiency, Stable routes, Mobile ad hoc networks

1. Introduction

A mobile ad hoc network (MANET) is a dynamic distributed system of mobile, autonomous wireless nodes. The network has limited bandwidth and the nodes have limited battery charge. In order to conserve battery charge, each node has a limited transmission range (i.e., transmits the data signals only to a limited distance). As a result, MANET routes are typically multi-hop in nature. As nodes move independent of each other, routes between a source and destination node often break and new routes have to be discovered. MANET routing protocols are of two types: proactive and reactive. Proactive routing protocols require the nodes to periodically exchange the table updates to pre-determine routes between any pair of source-destination nodes. Reactive (on-demand) routing protocols determine routes only when a route is required from a source to a destination. In dynamically changing environments, typical of MANETs, reactive routing protocols incur lower control overhead to discover routes compared to the proactive routing protocols [1]. In this paper, we work only with the on-demand reactive routing protocols.

Flooding is the default route discovery approach for on-demand MANET routing protocols. The flooding algorithm to discover routes can be briefly explained as follows: Whenever a source node needs a route to a destination node, it broadcasts a Route Request (RREQ) message to its neighbors. Neighbor nodes of the source node broadcast the received RREQ further, if they have not already done so. A RREQ message for a particular route discovery process is forwarded by a node exactly once. The destination node receives the RREQs along several routes, selects the best route according to the route selection principles of the particular routing protocol and notifies the selected route to the source through a Route-Reply (RREP) packet. The source starts sending data packets on the discovered route.

Flooding is inefficient and consumes significantly high energy and bandwidth. When a node receives a message for the first time in its neighborhood, at least 39% of the neighborhood would have seen it already and on the average only 41% of the additional area could be covered with a rebroadcast [2]. In this paper, we propose a novel density and mobility aware energy-efficient broadcast strategy called DMEF that attempts to reduce the energy consumed due to broadcast route discoveries by letting a node to broadcast only within a limited neighborhood. The neighborhood size to which a node advertises itself as part of the route discovery process is decided by the number of neighbors surrounding the node and the mobility of the node. The neighborhood size for rebroadcast is reduced in such a way that the RREQ packets still make it to the destination through one or more paths with a reduced energy spent per route discovery and such paths are also more stable compared to those incurred using flooding.

The rest of the paper is organized as follows: Section 2 describes the proposed DMEF strategy in detail. Section 3 discusses related work and the advantages of DMEF.

Section 4 discusses the simulation environment and presents simulation results illustrating the effectiveness of DMEF. Section 5 concludes the paper. Note that, throughout this paper, the terms 'path' and 'route', 'message' and 'packet' are used interchangeably. They mean the same.

2. DMEF Strategy

2.1 Terminology and Assumptions

Every node (say node u) in the network is configured with a maximum transmission range ($Range_u^{Max}$). If the distance between two nodes is less than or equal to the maximum transmission range, the two nodes are said to be within the "complete neighborhood" of each other. Each node broadcasts periodically a beacon message in its complete neighborhood. The time between two successive broadcasts is chosen uniform-randomly, by each node from the range $[0 \ldots T_{wait}]$. Using this strategy, each node learns about the number of nodes in its complete neighborhood.

2.2 Basic Idea

The twin objectives of DMEF are to discover stable routes with a reduced energy consumption compared to that incurred using flooding. DMEF achieves this by considering the number of neighbors of a node (a measure of node density) and node mobility. The basic idea behind DMEF is as follows: The transmission range of a RREQ broadcast for route discovery is not fixed for every node. A node that is surrounded by more neighbors in the complete neighborhood should broadcast the RREQ message only within a smaller neighborhood that would be sufficient enough to pick up the message and forward it to the other nodes in the rest of the network. On the other hand, a node that is surrounded by fewer neighbors in the complete neighborhood should broadcast the RREQ message to a larger neighborhood (but still contained within the complete neighborhood) so that a majority of the nodes in the complete neighborhood can pick up the message and rebroadcast it further. A node rebroadcasts a RREQ message at most once. The density aspect of DMEF thus helps to reduce the unnecessary transmission and reception of broadcast RREQ messages and conserves energy.

To discover stable routes that exist for a longer time, DMEF takes the following approach: A node that is highly mobile makes itself available only to a smaller neighborhood around itself, whereas a node that is less mobile makes itself available over a larger neighborhood (but still contained within the complete neighborhood). The reasoning is that links involving a slow moving node will exist for a long time. Hence, it is better for a slow moving node to advertise itself to a larger neighborhood so that the links (involving this node) that are part of the routes discovered will exist for a longer time. On the other hand, a fast moving node will have links of relatively longer lifetime with neighbors that are closer to it. Hence, it is worth to let a fast moving node advertise only to its nearby neighbors.

2.3 DMEF Mathematical Model

DMEF effectively uses the knowledge of neighborhood node density and mobility so that they complement each other in discovering stable routes in a more energy-efficient fashion. The transmission range used by a node u, $Range_u^{RREQ}$, to rebroadcast a RREQ message is given by the following model:

$$Range_u^{RREQ} = Range_u^{Max} - \left[\left(\frac{|Neighbors_u|}{\alpha} \right) * v_u^{\beta} \right] \cdot \quad (1)$$

In order to make sure, $Range_u^{RREQ}$ is always greater than or equal to zero, the value of parameter α should be chosen very carefully. For a given value of parameter β, the necessary condition is:

$$\alpha \geq \left[\left(\frac{|Neighbors_u|}{Range_u^{Max}} \right) * v_u^{\beta} \right] \ldots\ldots\ldots\ldots\ldots\ldots \quad (2)$$

In practice, the value of parameter α has to be sufficiently larger than the value obtained from (2), so that the RREQ message reaches neighbors who can forward the message further to the rest of the network. Otherwise, certain source-destination nodes may not be reachable from one another even though there may exist one or more paths between them in the underlying network.

2.4 Dynamic Selection of DMEF Parameter Values

The specialty of DMEF is that it allows for each node to dynamically choose at run-time the appropriate values for the critical operating parameters α and β depending on the perceived number of nodes in the complete neighborhood of the node and the node's own velocity. A node has to be simply pre-programmed with the appropriate values of α and β to be chosen for different values of the number of nodes in the complete neighborhood and node velocity.

Let *maxNeighb_lowDensity*, *maxNeighb_modDensity* represent the maximum number of neighbors a node should have in order to conclude that the complete neighborhood density of the node is low and moderate respectively. If a node has more than *maxNeighb_modDensity* number of neighbors, then the node is said to exist in a complete neighborhood of high density. Let *lowDensity_α*, *modDensity_α* and *highDensity_α* represent the values of α to be chosen by a node for complete neighborhoods of low, moderate and high density respectively. Let *maxVel_lowMobility*, *maxVel_modMobility* represent the maximum velocity values for a node in order to conclude that the mobility of the node is low and moderate respectively. If the velocity of a node is more than *maxVel_modMobility*, then the mobility of the node is said to be high. Let *lowMobility_β*, *modMobility_β* and *highMobility_β* represent the values of β to be chosen by a node when its mobility is low, moderate and high respectively.

Let $Neighbors_u^t$ and v_u^t represent the set of neighbors in the complete neighborhood and velocity of a node u at time t. Note that the set $Neighbors_u^t$ is determined by node u based on the latest periodic beacon exchange in the complete neighborhood formed by the maximum transmission range, $Range_u^{Max}$. The algorithm, *DMEF_Parameter_Selection*, to dynamically choose the values of parameters α and β (represented as α_u^t and β_u^t) is illustrated below in Figure 1:

```
Input: $Neighbors_u^t$ and $v_u^t$
Auxiliary Variables:
$minimum\_\alpha_u^t$ // minimum value of $\alpha$ to be chosen to avoid the transmission range of a node from becoming negative
$Range_u^{Max}$ // the maximum transmission range of a node for complete neighborhood

Density related variables: maxNeighb_lowDensity, maxNeighb_modDensity, lowDensity_α, modDensity_α, highDensity_α

Node Velocity related variables: maxVel_lowMobility, maxVel_modMobility, lowMobility_β, modMobility_β, highMobility_β
Output: $\alpha_u^t$ and $\beta_u^t$

Begin DMEF_Parameter_Selection
    if ($v_u^t \le maxVel\_lowMobility$)
        $\beta_u^t \leftarrow lowMobility\_\beta$
    else if ($v_u^t \le maxVel\_moderateMobility$)
        $\beta_u^t \leftarrow moderateMobility\_\beta$
    else
        $\beta_u^t \leftarrow highMobility\_\beta$
    $minimum\_\alpha_u^t \leftarrow \left[ \left( \frac{|Neighbors_u^t|}{Range_u^{Max}} \right) * (v_u^t)^{\beta_u^t} \right]$
    if ($|Neighbors_u^t| \le maxNeighb\_lowDensity$)
        $\alpha_u^t \leftarrow$ Maximum ($minimum\_\alpha_u^t$, $lowDensity\_\alpha$)
    else if ($|Neighbors_u^t| \le maxNeighb\_modDensity$)
        $\alpha_u^t \leftarrow$ Maximum ($minimum\_\alpha_u^t$, $modDensity\_\alpha$)
    else
        $\alpha_u^t \leftarrow$ Maximum ($minimum\_\alpha_u^t$, $highDensity\_\alpha$)
    return $\alpha_u^t$ and $\beta_u^t$
End DMEF_Parameter_Selection
```

Figure 1: Algorithm to Dynamically Select the Parameter Values for DMEF

3 Related Work

We surveyed the literature for different broadcast route discovery strategies that have been proposed to reduce the route discovery overhead and we describe below the strategies relevant to the research conducted in this paper. In Section 3.3, we qualitatively analyze the advantages of our DMEF broadcast strategy compared to the broadcast strategies described below in Sections 3.1 and 3.2.

3.1 Reliable Route Selection (RRS) Algorithm

In [3], the authors proposed a Reliable Route Selection (referred to as RRS) algorithm based on Global Positioning System (GPS) [4]. The RRS algorithm divides the circular area formed by the transmission range of a node into two zones: stable zone and caution zone. A node is said to maintain stable links with the neighbor nodes lying in its stable zone and maintain unstable links with the neighbor nodes lying in its caution zone. If R is the transmission range of a node, then the radius of the stable zone is defined as $r = R\text{-}\delta S$ where S is the speed of the node. The status zone is a circular region (with its own center) inscribed inside the circular region formed by the transmission range of the node. The center of the status zone need not be the center of the circular region forming the transmission range of the node, but always lies in the direction of movement of the node.

RRS works as follows: The Route-Request (RREQ) message of a broadcast route discovery process includes the co-ordinates representing the current position of the transmitter of the RREQ message, the co-ordinates representing the center of the stable zone of the transmitter, the value of parameter δ to be used by an intermediate node and the stable zone radius of the transmitter of the message. The source node of the route discovery process broadcasts the RREQ message in the complete neighborhood formed by the transmission range R. The RRS-related fields are set to initial values corresponding to the source node. An intermediate node receiving the RREQ message broadcasts the message further, only if the node lies in the stable zone of the transmitter. If a route discovery attempt based on a set value of δ is unsuccessful, the source node decrements the value of δ and launches another global broadcast based route discovery. This process is continued (i.e., the value of δ decremented and global broadcast reinitiated) until the source finds a path to the destination. If the source cannot find a route to the destination even while conducting route discovery with δ set to zero, then the source declares that the destination is not connected to it.

3.2 Efficient Broadcast Route Discovery Strategies

In [2], the authors propose several broadcast route discovery strategies that could reduce the number of retransmitting nodes of a broadcast message. These strategies can be grouped into four families: probability-based, counter-based, area-based and neighbor-knowledge based methods:

(i) **Probability-based method:** When a node receives a broadcast message for the first time, the node rebroadcasts the message with a certain probability. If the message received is already seen, then the node drops the message irrespective of whether or not the node retransmitted the message when it received the first time.

(ii) **Counter-based method:** When a node receives a broadcast message for the first time, it waits for a certain time before retransmitting the message. During this broadcast-wait-time, the node maintains a counter to keep track of the number of redundant broadcast messages received from some of its other neighbors. If this counter value exceeds a threshold within the broadcast-wait-time, then the node decides to drop the message. Otherwise, the node retransmits the message.

(iii) **Area-based method:** A broadcasting node includes its location information in the message header. The receiver node calculates the additional coverage area that would be obtained if the message were to be rebroadcast. If the additional coverage area is less than a threshold value, all future receptions of the same message will be dropped. Otherwise, the node starts a broadcast-wait-timer. Redundant broadcast messages received during this broadcast-wait-time are also cached. After the timer expires, the node considers all the cached messages and recalculates the additional coverage area if it were to rebroadcast the particular message. If the additional obtainable coverage area is less than a threshold value, the cached messages are dropped. Otherwise, the message is rebroadcast.

(iv) **Neighbor-knowledge based method:** This method requires nodes to maintain a list of 1-hop neighbors and 2-hop neighbors, learnt via periodic beacon exchange. Using these lists, a node calculates the set (of the smallest possible size) of 1-hop neighbors required to reach all the 2-hop neighbors. The minimum set of 1-hop neighbors that will cover all of the 2-hop neighbors is called the Multi Point Relays (MPRs).

3.3 Advantages of DMEF and Differences with Related Work

Our DMEF route discovery strategy is very effective in discovering relatively long-living routes in an energy-efficient manner and differs from the RRS algorithm in the following ways:

- RRS is highly dependent on location-service schemes like GPS, while DMEF is not dependent on any location-service scheme for its normal functionality.
- RRS requires the RREQ message header to be changed while DMEF does not require any change in the structure of the RREQ messages used for broadcasting. DMEF can be thus used with any MANET routing protocol without requiring any change in the routing protocol.
- In RRS, a node lying in the stable zone of the transmitter of the RREQ rebroadcasts the message in its complete neighborhood. However, it is only the recipient nodes lying in the stable zone of the transmitter that rebroadcast the RREQ. Hence, RRS is not energy-efficient. On the other hand, in DMEF, the transmission range for broadcast at a node is dynamically and locally determined using the node's velocity and neighborhood density values and is usually considerably less than the maximum transmission range.
- RRS does not properly handle the scenario where the value of $\delta * S$ exceeds the transmission range of the node R. The value of δ has to be iteratively reduced by trial and error method to determine the connectivity

between the source and destination nodes. DMEF is better than RRS because it requires only one broadcast route discovery attempt from the source to determine a route to the destination if the two nodes are indeed connected. The values of the DMEF parameters are dynamically determined at each node by the nodes themselves because a node knows better about its own velocity and neighborhood, compared to the source of the broadcast process.
- The network density does not influence the stable zone radius selected by RRS. As a result, in RRS, the number of nodes retransmitting the RREQ message in a neighborhood increases significantly as the network density is increased. DMEF is quite effective in reducing the number of nodes retransmitting the RREQ message in high-density networks.

The advantages of the DMEF scheme when compared with the broadcast route discovery strategies discussed in Section 3.2 are summarized as follows:

- The probability-based and MPR-based methods do not guarantee that the broadcast message will be routed on a path with the minimum hop count or close to the minimum hop count. Previous research [5] on the impact of these broadcast strategies on the stability and hop count of the DSR routes indicates that the hop count of the paths can be far more than the minimum hop count and the routes have a smaller lifetime than the paths discovered using flooding. The probability-based method cannot always guarantee that the RREQ message gets delivered to the destination. Also, with increase in network density, the number of nodes retransmitting the message increases for both the probability-based and MPR-based methods.

  DMEF determines paths with hop count being close to that of the minimum hop count paths and such paths have a relatively larger lifetime compared to those discovered using flooding. DMEF almost always guarantees that a source-destination route is discovered if there is at least one such route in the underlying network. DMEF effectively controls the RREQ message retransmission overhead as the network density increases.
- The counter-based and area-based methods require careful selection of the threshold counter and area of coverage values for their proper functioning. Each node has to wait for a broadcast-wait-time before retransmitting the message. This can introduce significant route acquisition delays. The area-based method also requires the nodes to be location-aware and include the location information in the broadcast messages.

  With DMEF, there is no waiting time at a node to rebroadcast a received RREQ message, if the message has been received for the first time during a particular route discovery process. DMEF does not depend on any location-aware services for its operation and the structure of the RREQ message for a routing protocol need not be changed.

4 Simulations

The effectiveness of the DMEF strategy has been studied through simulations conducted using a MANET discrete-event simulation software developed by us in Java. We use the well-known minimum-hop based Dynamic Source Routing (DSR) protocol [6] and the recently proposed Location-Prediction Based Routing (LPBR) protocol [7] to reduce the number of global broadcast route discoveries, as the routing protocols that use DMEF as their route discovery strategy. The benchmark used for DMEF evaluation is the performance of DSR and LPBR with flooding as the route discovery strategy. The network dimensions are: 1000m x 1000m. The maximum transmission range of a node is 250m. Network density is varied by conducting simulations with 25 (low density), 50 (moderate density) and 75 (high density) nodes. The mobility model used is the Random Waypoint model [8] according to which the velocity of each node is uniformly randomly distributed in the range [v_{min}... v_{max}]. The value of v_{min} is 0 m/s and the value of v_{max} is 10, 30 and 50 m/s representing average node velocities of 5 (low mobility), 15 (moderate mobility) and 25 m/s (high mobility) respectively. The traffic model used is the constant bit rate (CBR) model with a data packet of size 512 bytes sent every 0.25 seconds. There are 15 source-destination (*s-d*) pairs. The transmission energy is 1.4 W and the reception energy is 1 W [9]. Network bandwidth is 2 Mbps. The Medium Access Control (MAC) layer model followed is the IEEE 802.11 Distributed Coordinated Function (DCF) model [10]. The DMEF parameter values are given in Table 1. Total simulation time is 1000 seconds.

Table 1: DMEF Parameter Values

DMEF Parameter	**Value**
maxNeighb_lowDensity	5
maxNeighb_modDensity	10
lowDensity_α	5
modDensity_α	10
highDensity_α	20
maxVel_lowMobility	5
maxVel_modMobility	15
lowMobility_β	1.6
modMobility_β	1.3
highMobility_β	1.1
T_{wait}	10 seconds

4.1 Dynamic Source Routing (DSR) Protocol

The unique feature of DSR [6] is source routing: data packets carry information about the route from the source to the destination in the packet header. As a result, intermediate nodes do not need to store up-to-date routing information in their forwarding tables. Route discovery is by means of the broadcast query-reply cycle. A source node *s* wishing to send a data packet to a destination *d*, broadcasts a RREQ packet throughout the network. The RREQ packet reaching a node contains the list of intermediate nodes through which it has propagated from the source node. After receiving the first RREQ packet, the destination node waits for a short time period for any more RREQ packets, then chooses a path with the minimum hop count and sends a RREP along the selected path. If any RREQ is received along a path whose hop count is lower than the one on which the RREP was sent, another RREP would be sent on the latest minimum hop path discovered. To minimize the route acquisition delay, DSR lets intermediate nodes to promiscuously listen to the channel, store the learnt routes (from the RREQ and data packets) in a route cache and use these cached route information to send the RREP back to the source. We do not use this feature as promiscuous listening dominates the energy consumed at each node and DSR could still effectively function without promiscuous listening and route caching. Also, in networks of high node mobility, cached routes are more likely to become stale, by the time they are used.

4.2 Location Prediction Based Routing (LPBR) Protocol

LPBR [7] simultaneously minimizes the number of flooding based route discoveries and the hop count of the paths for a source-destination session. During a regular flooding-based route discovery, LPBR collects the location and mobility information of the nodes in the network and stores the collected information at the destination node of the route search process. When the minimum-hop route discovered through the flooding fails, the destination node attempts to predict the current location of each node using the location and mobility information collected during the latest flooding-based route discovery. A minimum hop path Dijkstra algorithm [11] is run on the locally predicted global topology. If the predicted minimum hop route exists in reality, no expensive flooding-based route discovery is needed and the source continues to send data packets on the discovered route; otherwise, the source initiates another flooding-based route discovery.

4.3 Performance Metrics

The performance metrics studied are as follows:

- *Total Energy Lost per Route Discovery*: This is the average of the total energy consumed for the global broadcast based route discovery attempts. This includes the sum of the energy consumed to transmit (broadcast) a RREQ packet to all the nodes in the neighborhood and to receive the RREQ packet sent by each node in the neighborhood, summed over all the nodes.
- *Percentage of Total Energy Spent for Route Discovery*: This is the ratio of the total energy spent for route discovery to the sum of the energy spent across all the nodes in the network.
- *Hop Count per Path*: This is the average hop count per path, time-averaged over all the *s-d* sessions. For example, if we have been using two paths P1 of hop count 3 and P2 of hop count 5 for 10 and 20 seconds respectively, then the time-averaged hop count of P1 and P2 is (3*10+5*20)/30 = 4.33 and not 4.
- *Time between Route Discoveries*: This is the average of the time between two successive global broadcast based route discovery attempts. Larger the time between two successive route discoveries, lower will be the control overhead.
- *End-to-End Delay per Data Packet*: This is the average of the delay incurred by the data packets that originate at the source and delivered at the destination. The delay incurred by a data packet includes all the possible delays: the buffering delay due to the route acquisition latency, the queuing delay at the interface queue to access the medium, the transmission delay, propagation delay, and the retransmission delays due to the MAC layer collisions.
- *Packet Delivery Ratio*: This is the ratio of the data packets delivered to the destination to the data packets originated at the source, computed over all the *s-d* sessions.
- *Energy Throughput*: This is the average of the ratio of the number of data packets reaching the destination to the sum of the energy spent across all the nodes in the network.

The performance results illustrated in Figures 2 through 8 are an average of simulations conducted with 5 mobility profiles for each operating condition.

4.4 Total Energy Spent Route Discovery

Performance results in figures 2.1 through 2.3 illustrate that DMEF achieves its purpose of reducing the energy spent in the network due to global broadcast route discoveries. The reduction in the energy spent for route discoveries is evident in both DSR and LPBR protocols.

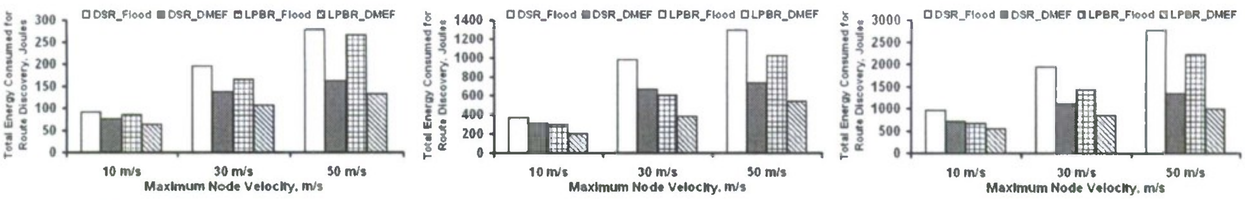


Figure 2.1: 25 Nodes **Figure 2.2:** 50 Nodes **Figure 2.3:** 75 Nodes

Figure 2: Total Energy Consumed for Route Discovery

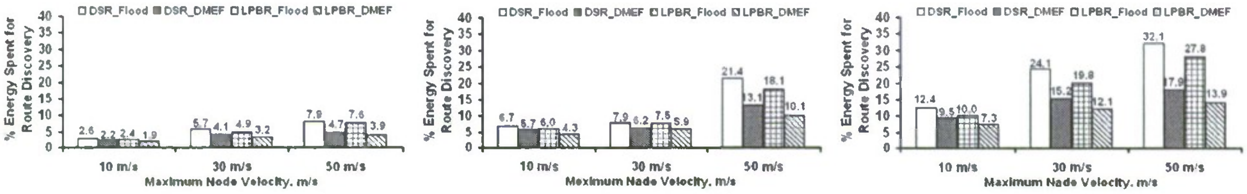


Figure 3.1: 25 Nodes **Figure 3.2:** 50 Nodes **Figure 3.3:** 75 Nodes

Figure 3: Percentage of Total Energy Spent for Route Discovery

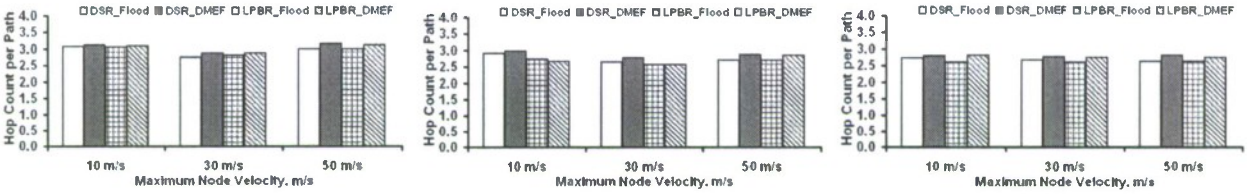


Figure 4.1: 25 Nodes **Figure 4.2:** 50 Nodes **Figure 4.3:** 75 Nodes

Figure 4: Average Hop Count per Path

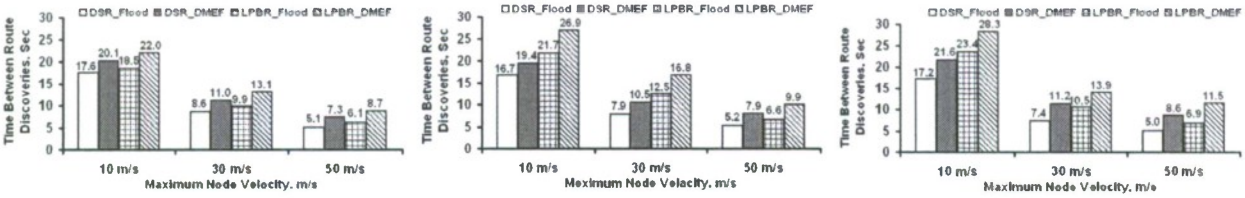


Figure 5.1: 25 Nodes **Figure 5.2:** 50 Nodes **Figure 5.3:** 75 Nodes

Figure 5: Time between Two Successive Route Discoveries

The reduction in the energy spent for route discoveries is also more evident as we increase the network density and/or node mobility. This illustrates the effectiveness of DMEF because the strategy aims to minimize the unnecessary rebroadcasts in a network especially when the network density is high. In high-density networks, it is enough to rebroadcast through a reduced set of nodes to find a set of paths between a source and destination rather than broadcasting through all the nodes in the network. Compared to DSR, LPBR incurs relatively lower number of global broadcast based route discoveries. In addition, DMEF helps the protocol to reduce the energy spent per broadcast based route discovery. Aided by both these factors, LPBR incurs a significantly lower energy due to route discoveries compared to DSR.

4.5 Percentage of Total Energy Spent for Route Discovery

As observed in Figures 3.1 through 3.3, for both DSR and LPBR, the difference in the percentage of total energy spent for route discovery using flooding and DMEF increases as we increase the network density and/or node mobility. For a given node mobility, the energy savings obtained with DMEF increases with increase in network density. Similarly, for a given network density, the energy savings obtained with DMEF, relative to flooding, increases with increase in the level of node mobility. For a given network density and node mobility, the relative reduction in the percentage of total energy spent for route discoveries due to DMEF vis-à-vis flooding is almost the same for both DSR and LPBR. This illustrates that DMEF can be used for energy-efficient route discovery by any routing protocol for mobile ad hoc networks.

4.6 Average Hop Count per Path

DMEF prefers to determine long-living routes by primarily broadcasting the RREQ message through nodes that are relatively slow moving in the network. As a result, the routes determined for the DSR and LPBR protocols need not have hop count matching with that of the minimum hop count paths in the network. DMEF determines routes that have at most 8% larger hop count compared to the minimum hop routes, but the routes determined through DMEF exist for a relatively larger lifetime compared to the routes determined using flooding. For both DSR and LPBR, for a given node mobility in the network, as we increase the network density from low to moderate and to high, the average hop count per path decreases (by about 5%-15%).

4.7 Time between Successive Route Discoveries

The twin objectives of DMEF are to be energy-efficient and to determine routes that exist for a long time. DMEF accomplishes the latter objective by preferring to broadcast the RREQ messages primarily through nodes that have been moving relatively slowly in the network. As a result, the routes determined using DMEF exist for a relatively longer time in the network. The lifetime of routes determined for both DSR and LPBR protocols using DMEF as the route discovery strategy is significantly larger compared to that of the DSR and LPBR routes determined using flooding. This is because DMEF prefers to propagate RREQ packets through relatively slow moving nodes that are also close to each other. In addition, LPBR attempts to increase the time between successive global broadcast discoveries by predicting a source-destination route using the Location Update Vectors (LUVs) collected during the latest broadcast route discovery. As we increase the network density, the chances of correctly predicting at least one source-destination path in the network increases. Hence, in the case of LPBR, for a given node mobility, the time between two successive global broadcast route discoveries increases as the network density increases. For both DSR and LPBR, compared to flooding, the relative increase in the lifetime of the routes discovered using DMEF and the reduction in the frequency of DMEF route discoveries can be significantly observed with increase in network density and/or node mobility.

4.8 End-to-End Delay per Data Packet

DMEF exerts a relatively lower control overhead to determine routes compared to flooding. This is evident as DSR incurs a relatively lower end-to-end delay per data packet (refer Figure 6) when routes are determined using DMEF compared to flooding. The relative difference between the delays per data packet for DSR routes discovered using flooding and DMEF increases as we increase the node mobility and/or network density. With DSR, the route discovery overhead incurred due to relatively unstable routes discovered using flooding weighs far more than the slightly larger hop count of routes discovered using DMEF. In LPBR, there is a relatively slight reduction in the delays per data packet with DMEF in networks of high density/ high mobility. This is due to the relatively less congestion in the nodes attributed to the reduced number of route discovery attempts. In networks of low node mobility, the delay per data packet for LPBR using DMEF is sometimes observed to be slightly larger than the delays per packet obtained with flooding. This is due to the slightly larger hop count of the paths discovered in such networks and lower route discovery overhead.

4.9 Packet Delivery Ratio

Performance results in Figures 7.1 through 7.3 illustrate that the packet delivery ratio of the two routing protocols using DMEF can be lower than that obtained using flooding only by at most 3% in low-density networks. In moderate density networks, both the route discovery strategies yield almost the same packet delivery ratio. In high density networks, the packet delivery ratio of routing protocols using DMEF can be larger than that obtained using flooding by about 3%. In high-density networks, even though flooding helps to propagate the RREQ messages through several routes, the excessive overhead generated by these redundant RREQ messages block the queues of certain heavily used nodes in the network, thus leading to sometimes a relatively lower packet delivery ratio compared to DMEF. In low-density networks, DMEF could very rarely fail to determine source-destination routes, even if one exists, due to its optimization approach of trying to shrink the range of broadcast of the RREQ messages. DMEF broadcasts RREQ messages over a relatively larger transmission range in low-density networks compared to those used for high-density networks. As we increase node density, the packet delivery ratio under both flooding and DMEF approaches unity.

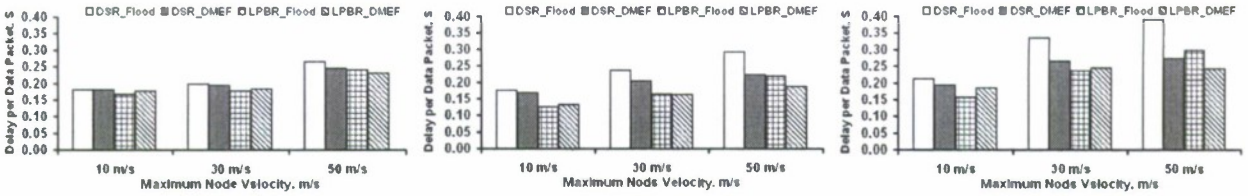


Figure 6.1: 25 Nodes **Figure 6.2:** 50 Nodes **Figure 6.3:** 75 Nodes

Figure 6: Average End-to-End Delay per Data Packet Delivered

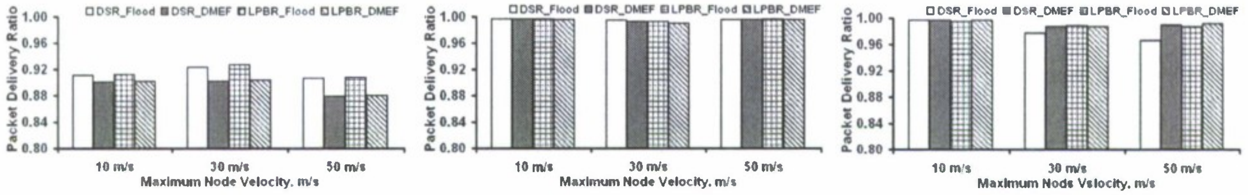


Figure 7.1: 25 Nodes **Figure 7.2:** 50 Nodes **Figure 7.3:** 75 Nodes

Figure 7: Packet Delivery Ratio

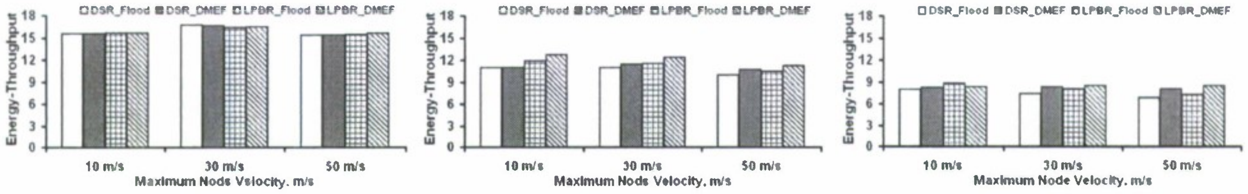


Figure 8.1: 25 Nodes **Figure 8.2:** 50 Nodes **Figure 8.3:** 75 Nodes

Figure 8: Energy-Throughput

4.10 Energy Throughput

For a given offered data traffic load, larger the energy throughput, the smaller the amount of energy spent in delivering the data packets to the destination. Notice that in our simulations, the number of source-destination sessions is always fixed at 15, i.e., the offered data traffic load is fixed. Based on Figures 7 and 8, we observe that with increase in the network density, the packet delivery ratio approaches unity, but the energy throughput decreases. This is because more nodes participate and spend their energy in moderate and high-density networks to route a given offered data traffic load. Note that energy consumption is in the form of direct transmissions and receptions of the intermediate nodes on a path and indirect receptions at the neighboring nodes of the intermediate nodes on a path. As we increase the network density as well as the level of node mobility, the energy throughput obtained with both DSR and LPBR using DMEF is larger than that obtained using flooding as the route discovery strategy. In low and moderate density networks and low and moderate levels of node mobility, the energy throughput for both DSR and LPBR are almost the same while using both DMEF and flooding for route discoveries.

5 Conclusions

The high level contribution of this paper is the design and development of a novel network density and node mobility aware, energy-efficient route discovery strategy called DMEF for mobile ad hoc networks. The twin objectives of DMEF are to increase the time between successive global broadcast route discoveries and reduce the energy consumption during such global broadcast discoveries vis-à-vis flooding. Each node operates with a maximum transmission range and periodically broadcasts beacons to the neighborhood covered (called the complete neighborhood) within this range. DMEF permits each node to dynamically adjust the transmission range to broadcast the Route-Request (RREQ) messages of the route discovery process. A node that is surrounded by more neighbors advertises itself only to a limited set of nearby neighbors and a node that is surrounded by few neighbors will advertise itself to a maximum of those neighbors. Similarly, a node that is slow-moving advertises itself to a majority of its neighbors so that links formed using this node can be more stable. A node that has been fast-moving advertises itself only to the neighbors closer to it. The neighborhood dynamically chosen for a RREQ broadcast is always contained within the complete neighborhood defined by the maximum transmission range of the node.

The effectiveness of DMEF has been studied through simulations with the well-known Dynamic Source Routing (DSR) protocol and the recently proposed Location Prediction Based Routing (LPBR) protocol. The benchmark used for the evaluation purposes is the commonly used flooding based global broadcast route discoveries. Simulation results indicate that DMEF is very effective in reducing the total energy spent per route discovery attempt for both DSR and LPBR. In addition, for both DSR and LPBR, DMEF reduces the number of global broadcast route discoveries by determining routes with longer lifetime, reduces the percentage of total energy spent for route discoveries, reduces the end-to-end delay per data packet and increases the energy throughput. The increase in the hop count of DSR and LPBR routes compared to that discovered using flooding is at most 8%. We conjecture that DMEF can be similarly very effective with respect to all of the other currently existing on-demand MANET routing protocols, none of which can simultaneously minimize the number of route discoveries as well as the hop count of the paths. DMEF can be used with these MANET routing protocols to discover long-living stable paths with hop count close to that of the minimum hop paths and at the same time incur less control message and energy overhead.

Acknowledgments

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-08-2-0061. The views and conclusions in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

[1] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu and J. Jetcheva, "A Performance Comparison of Multi-hop Wireless Ad Hoc Network Routing Protocols," Proceedings of the 4th ACM Mobile Computing and Networking Conference, pp. 85-97, Oct. 1998.

[2] S. Ni, Y. Tseng, Y. Chen and J. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," *Proceedings of ACM International Conference on Mobile Computing and Networking,* pp.151-162, 1999.

[3] Y.-J. Suh, W. Kim and D.-H. Kwon, "GPS-Based Reliable routing Algorithms for Ad Hoc Networks," *The Handbook of Ad Hoc Wireless Networks*, pp. 361 – 374, CRC Press, 2003.

[4] B. Hofmann-Wellenhof, H. Lichtenegger and J. Collins, *Global Positioning System: Theory and Practice*, 5th rev. ed., Springer, Sept. 2004.

[5] N. Meghanathan, "Impact of Broadcast Route Discovery Strategies on the Performance of Mobile Ad Hoc Network Routing Protocols," Proceedings of the International Conference on High Performance Computing, Networking and Communication Systems, pp. 144-151, July 2007.

[6] D. B. Johnson, D. A. Maltz and J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Ad hoc Networks," in *Ad hoc Networking*, Chapter 5, C. E. Perkins, Eds. Addison Wesley, pp. 139 – 172, 2000.

[7] N. Meghanathan, "Location Prediction Based Routing Protocol for Mobile Ad hoc Networks," *Proceedings of the IEEE Global Communications (GLOBECOM) Conference*, New Orleans, Nov-Dec 2008.

[8] C. Bettstetter, H. Hartenstein and X. Perez-Costa, "Stochastic Properties of the Random-Way Point Mobility Model," *Wireless Networks*, pp. 555 – 567, vol. 10, no. 5, September 2004.

[9] L. M. Feeney, "An energy-consumption model for performance analysis of routing protocols for mobile ad hoc networks," *Journal of Mobile Networks and Applications*, vol. 3, no. 6, pp. 239 – 249, June 2001.

[10] IEEE Standards Department, Wireless LAN Medium Access Control and Physical Layer Specifications, IEEE Standard 802.11, 1997.

[11] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms," 2nd Edition, MIT Press/ McGraw-Hill, Sept. 2001.

Dr. Natarajan Meghanathan is an Assistant Professor of Computer Science at Jackson State University, USA. He received his MS and PhD in Computer Science form Auburn University and The University of Texas at Dallas respectively. Dr. Meghanathan's main area of research is ad hoc networks. He has more than 45 peer-reviewed publications in leading international journals and conferences in this area. He serves as the editor of a number of international journals and also in the program committee of several leading international conferences in the area of networks.

II Peer-reviewed Conference Publications/ Proceedings

C1. International Conference on Wireless Networks, ICWN 09, Las Vegas, NV, USA

A Density and Mobility Aware Energy-Efficient Broadcast Strategy to Minimize the Number of Route Discoveries in Mobile Ad hoc Networks

Natarajan Meghanathan
Jackson State University
P. O. Box 18839, Jackson, MS 39217
nmeghanathan@jsums.edu

Abstract

We propose a novel network density and node mobility aware, energy-efficient on-demand route discovery strategy called DMEF for mobile ad hoc networks. The twin objectives of DMEF are to increase the time between successive global broadcast route discoveries and reduce the energy consumption during such global broadcast discoveries vis-à-vis flooding. DMEF permits each node to dynamically adjust the transmission range to broadcast the Route-Request (RREQ) messages of the route discovery process. The neighborhood dynamically chosen for a RREQ broadcast is always contained within the complete neighborhood defined by the maximum transmission range of the node. A node surrounded by more neighbors advertises itself only to a limited set of nearby neighbors and a node surrounded by few neighbors will advertise itself to a maximum of its neighbors. Similarly, a slow-moving node advertises itself to a majority of its neighbors so that links formed using this node can be more stable. A node that has been fast-moving advertises itself only to the neighbors closer to it. Simulation results indicate that DMEF is very effective, vis-à-vis flooding, in reducing the total energy spent per route discovery attempt as well as in reducing the number of global broadcast route discoveries by determining routes with longer lifetime.

Keywords: Route discovery, Flooding, Energy efficiency, Stable routes, Mobile ad hoc networks

1. Introduction

A mobile ad hoc network (MANET) is a dynamic distributed system of mobile, autonomous wireless nodes. The network has limited bandwidth and the nodes have limited battery charge. In order to conserve battery charge, each node has a limited transmission range (i.e., transmits the data signals only to a limited distance). As a result, MANET routes are typically multi-hop in nature. As nodes move independent of each other, routes between a source and destination node often break and new routes have to be discovered. MANET routing protocols are of two types: proactive and reactive. Proactive routing protocols periodically exchange table updates to pre-determine routes between any pair of source-destination nodes. Reactive (on-demand) routing protocols determine routes only when a route is required from a source to a destination. In dynamically changing environments, typical of MANETs, reactive routing protocols incur lower control overhead to discover routes compared to the proactive routing protocols [3]. In this paper, we work only with the on-demand reactive routing protocols.

Flooding is the default route discovery approach for on-demand MANET routing protocols. The flooding algorithm to discover routes can be briefly explained as follows: Whenever a source node needs a route to a destination node, it broadcasts a Route Request (RREQ) message to its neighbors. Neighbor nodes of the source node broadcast the received RREQ further, if they have not already done so. A RREQ message for a particular route discovery process is forwarded by a node exactly once. The destination node receives the RREQs along several routes, selects the best route according to the route selection principles of the particular routing protocol and notifies the selected route to the source through a Route-Reply (RREP) packet. The source starts sending data packets on the discovered route.

Flooding is inefficient and consumes significantly high energy and bandwidth. When a node receives a message for the first time in its neighborhood, at least 39% of the neighborhood would have seen it already and on the average only 41% of the additional area could be covered with a rebroadcast [8]. We propose a novel density and mobility aware energy-efficient broadcast strategy called DMEF that attempts to reduce the energy consumed due to broadcast route discoveries by letting a node to broadcast only within a limited neighborhood. The neighborhood size to which a node advertises itself as part of the route discovery process is decided by the number of neighbors surrounding the node and the mobility of the node. The neighborhood size for rebroadcast is reduced in such a way that the RREQ packets still make it to the destination through one or more paths with a reduced energy spent per route discovery and that such paths are also relatively more stable compared to those incurred using flooding.

The rest of the paper is organized as follows: Section 2 describes the proposed DMEF strategy in detail. Section 3 discusses related work and the advantages of DMEF. Section 4 discusses the simulation environment and presents simulation results illustrating the effectiveness of DMEF. Section 5 concludes the paper. Throughout this paper, the terms 'path' and 'route', 'message' and 'packet' are used interchangeably. They mean the same.

2. DMEF Strategy

2.1 Terminology and Assumptions

Every node (say node u) in the network is configured with a maximum transmission range, $Range_u^{Max}$. If the distance between two nodes is less than or equal to $Range_u^{Max}$, then the two nodes are said to be within the "complete neighborhood" of each other. Each node broadcasts periodically a beacon message to learn about the number of nodes in its complete neighborhood. The time between successive broadcasts is chosen uniformly, randomly, by each node, from within the range $[0 \ldots T_{wait}]$.

2.2 Basic Idea of DMEF

The basic idea behind DMEF is as follows: The transmission range of a RREQ broadcast is not fixed for every node. A node surrounded by more neighbors in the complete neighborhood should broadcast the RREQ message only within a smaller neighborhood that would be sufficient enough to pick up the message and forward it to the other nodes in the rest of the network. On the other hand, a node that is surrounded by fewer neighbors in the complete neighborhood should broadcast the RREQ message to a maximum of those neighbors (but still contained within the complete neighborhood) so that a majority of the nodes in the complete neighborhood can pick up the message and rebroadcast it further. A node rebroadcasts a RREQ message at most once. The density aspect of DMEF thus helps to reduce the unnecessary transmission and reception of the RREQ messages and conserves energy.

To discover stable routes that exist for a longer time, DMEF takes the following approach: A node that is highly mobile makes itself available only to a smaller neighborhood around itself, whereas a node that is less mobile makes itself available over a larger neighborhood (but still contained within the complete neighborhood). DMEF lets a slow moving node to advertise itself to a larger neighborhood so that the links (involving this node) that are part of the routes discovered will exist for a longer time. Whereas, a fast moving node will have links of relatively longer lifetime with neighbors that are closer to it. Hence, DMEF lets a fast moving node advertise only to its nearby neighbors.

2.3 DMEF Mathematical Model

DMEF effectively uses the knowledge of node density and mobility so that they complement each other in discovering stable routes in a more energy-efficient fashion. The transmission range used by a node u, $Range_u^{RREQ}$, to rebroadcast a RREQ message is given by the following model:

$$Range_u^{RREQ} = Range_u^{Max} - \left[\left(\frac{|Neighbors_u|}{\alpha} \right) * v_u^{\beta} \right] \quad (1)$$

For a given value of parameter β, in order to make sure that the value of $Range_u^{RREQ}$ is always positive, the necessary condition is:

$$\alpha \geq \left[\left(\frac{|Neighbors_u|}{Range_u^{Max}} \right) * v_u^{\beta} \right] \ldots\ldots\ldots\ldots\ldots (2)$$

In practice, the value of parameter α has to be sufficiently larger than that obtained from equality (2) for the RREQ to reach neighbors who can forward the message further to the rest of the network. Otherwise, certain source-destination nodes may not be reachable from each other, even though there may exist one or more paths between them in the underlying network.

2.4 Algorithm for Dynamic Selection of DMEF Parameters

We now describe an algorithm (refer Figure 1) that allows for each node to dynamically choose at run-time the appropriate values for the critical operating parameters α and β depending on the perceived number of nodes in the complete neighborhood of the node and the node's own velocity. Let the maximum number of neighbors a node should have in order to conclude that the complete neighborhood density of the node is low and moderate be denoted by *maxNeighb_lowDensity* and *maxNeighb_modDensity* respectively. If a node has more than *maxNeighb_modDensity* number of neighbors, then the node is said to exist in a complete neighborhood of high density. Let *lowDensity_α*, *modDensity_α* and *highDensity_α* represent the values of α to be chosen by a node for complete neighborhoods of low, moderate and high density respectively. Let *maxVel_lowMobility*, *maxVel_modMobility* represent the maximum velocity values for a node in order to conclude that the mobility of the node is low and moderate respectively. If the velocity of a node is more than *maxVel_modMobility*, then the mobility of the node is said to be high. Let *lowMobility_β*, *modMobility_β* and *highMobility_β* represent the values of β to be chosen by a node when its mobility is low, moderate and high respectively. Let v_u^t represent velocity of a node u at time t and let $Neighbors_u^t$ represent the set of neighbors in the complete neighborhood determined by node u based on the latest periodic beacon exchange in the complete neighborhood formed by $Range_u^{Max}$. The algorithm to dynamically choose the values of parameters α and β (denoted as α_u^t and β_u^t) for a node u is shown below:

```
Input: $Neighbors_u^t$ and $v_u^t$
Output: $\alpha_u^t$ and $\beta_u^t$
Begin DMEF_Parameter_Selection
    if ($v_u^t \le maxVel\_lowMobility$) $\beta_u^t \leftarrow lowMobility\_\beta$
    else if ($v_u^t \le maxVel\_modMobility$)
        $\beta_u^t \leftarrow modMobility\_\beta$
    else $\beta_u^t \leftarrow highMobility\_\beta$
    $minimum\_\alpha_u^t \leftarrow \left\lceil \left( \frac{|Neighbors_u^t|}{Range_u^{Max}} \right) * \left( v_u^t \right)^{\beta_u^t} \right\rceil$
    if ($|Neighbors_u^t| \le maxNeighbors\_lowDensity$)
        $\alpha_u^t \leftarrow$ Maximum ($minimum\_\alpha_u^t$, $lowDensity\_\alpha$)
    else if ($|Neighbors_u^t| \le maxNeighb\_modDensity$)
        $\alpha_u^t \leftarrow$ Maximum ($minimum\_\alpha_u^t$, $modDensity\_\alpha$)
    else
        $\alpha_u^t \leftarrow$ Maximum ($minimum\_\alpha_u^t$, $highDensity\_\alpha$)
    return $\alpha_u^t$ and $\beta_u^t$
End DMEF_Parameter_Selection
```

Figure 1: Algorithm to Dynamically Select the Parameter Values for DMEF

The number of neighbors in the complete neighborhood and the node velocity can be different for each node at a given time instant and can be different for even a particular node at different time instants. After selecting the appropriate values for parameters α and β at time t, a node can determine the transmission range to be used for the broadcast of the RREQ message using equation (1).

3. Related Work

In [9], the authors proposed a Reliable Route Selection (referred to as RRS) algorithm based on Global Positioning System (GPS) [4]. The RRS algorithm divides the circular area formed by the transmission range of a node into two zones: stable zone and caution zone. A node is said to maintain stable links with the neighbor nodes lying in its stable zone and maintain unstable links with the neighbor nodes lying in its caution zone (outside the stable zone). If R is the transmission range of a node, then the radius of the stable zone is defined as $r = R\text{-}\delta S$ where S is the speed of the node. The stable zone is a circular region (with its own center) inscribed inside the circular region formed by the transmission range of the node. The center of the stable zone always lies in the direction of movement of the node.

RRS works as follows: The RREQ message of a broadcast route discovery process includes the co-ordinates representing the current position of the transmitter of the RREQ, the co-ordinates representing the center of the stable zone of the transmitter, the value of parameter δ to be used by an intermediate node and the stable zone radius of the transmitter of the message. The source node of the route discovery process broadcasts the RREQ in the complete neighborhood formed by the transmission range R. The RRS-related fields are set to initial values corresponding to the source node. An intermediate node receiving the RREQ broadcasts the message further, only if the node lies in the stable zone of the transmitter. If a route discovery attempt based on a set value of δ is unsuccessful, the source node decrements the value of δ and launches another global broadcast based route discovery. This process is continued (i.e., the value of δ decremented and global broadcast reinitiated) until the source finds a path to the destination. If the source cannot find a route to the destination even while conducting route discovery with δ set to zero, then the source declares that the destination is not connected to it.

DMEF is very effective in discovering relatively long-living routes in an energy-efficient manner and differs from the RRS algorithm in the following ways:

- RRS is highly dependent on location-service schemes like GPS. DMEF is not dependent on any location-service scheme for its normal functionality.
- RRS requires the RREQ message header to be changed while DMEF does not require any change in the structure of the RREQ messages. DMEF can be thus used with any MANET routing protocol without requiring any change in the protocol.
- In RRS, a node lying in the stable zone of the transmitter of the RREQ rebroadcasts the message in

its complete neighborhood. However, it is only the recipient nodes lying in the stable zone of the transmitter that rebroadcast the RREQ. Hence, RRS is not energy-efficient. On the other hand, in DMEF, the transmission range for broadcast at a node is dynamically and locally determined using the node's velocity and neighborhood density values and is usually considerably less than the maximum transmission range.

- RRS does not properly handle the scenario when the value of $\delta * S$ exceeds the maximum transmission range, R. The value of δ has to be iteratively reduced by trial and error method to determine the connectivity between the source and destination nodes. On the other hand, DMEF requires only one broadcast route discovery attempt from the source to determine a route to the destination if the two nodes are indeed connected. The values of the DMEF parameters are dynamically determined locally at each node because a node knows better than any other node about its own velocity and neighborhood.
- In RRS, the number of nodes retransmitting the RREQ is the same as that observed with the default route discovery approach of flooding. The network density does not influence the stable zone radius selected by RRS. But, DMEF is quite effective in reducing the number of nodes retransmitting the RREQ message in high-density networks.

4. Simulations

The effectiveness of the DMEF strategy has been studied through simulations conducted using a MANET discrete-event simulation software developed by us in Java. We use the well-known minimum-hop based Dynamic Source Routing (DSR) protocol [6] and the recently proposed Location-Prediction Based Routing (LPBR) protocol [7] to reduce the number of global broadcast route discoveries, as the routing protocols that use DMEF as their route discovery strategy. The benchmark used for DMEF evaluation is the performance of DSR and LPBR with flooding as the route discovery strategy. The network dimensions are: 1000m x 1000m. The maximum transmission range of a node is 250m. Network density is varied by conducting simulations with 25 (low density), 50 (moderate density) and 75 (high density) nodes. The mobility model used is the Random Waypoint model [1] according to which the velocity of each node is uniformly randomly distributed in the range $[v_{min} \ldots v_{max}]$. The value of v_{min} is 0 m/s and the value of v_{max} is 10, 30 and 50 m/s representing average node velocities of 5 (low mobility), 15 (moderate mobility) and 25 m/s (high mobility) respectively. The traffic model used is the constant bit rate (CBR) model with a data packet of size 512 bytes sent every 0.25 seconds. There are 15 source-destination (s-d) pairs. The transmission energy is 1.4 W and the reception energy is 1 W [3]. Network bandwidth is 2 Mbps. The Medium Access Control (MAC) layer model followed is the IEEE 802.11 Distributed Coordinated Function (DCF) model [5]. The DMEF parameter values are given in Table 1. Total simulation time is 1000 seconds.

Table 1: DMEF Parameter Values

DMEF Parameter	Value
maxNeighb_lowDensity	5
maxNeighb_modDensity	10
lowDensity_α	5
modDensity_α	10
highDensity_α	20
maxVel_lowMobility	5
maxVel_modMobility	15
lowMobility_β	1.6
modMobility_β	1.3
highMobility_β	1.1
T_{wait}	10 seconds

4.1 Overview of DSR and LPBR Protocols

In DSR [6], data packets carry information about the route from the source to the destination in the packet header. As a result, intermediate nodes do not need to store up-to-date routing information in their forwarding tables. Route discovery is by means of the broadcast query-reply cycle. The RREQ packet reaching a node contains the list of intermediate nodes through which it has propagated from the source node. After receiving the first RREQ packet, the destination waits for a short time period for any more RREQs, then chooses a path with the minimum hop count and sends a Route-Reply Packet (RREP) along the selected path. If any RREQ is received along a path whose hop count is lower than the one on which the RREP was sent, another RREP would be sent on the latest minimum hop path discovered.

LPBR [7] simultaneously minimizes the number of broadcast route discoveries and the hop count of the paths for a source-destination session. During a regular broadcast route discovery, LPBR collects the location and mobility information of the nodes in the network and stores the collected information at the destination node of the route search process. When the minimum-hop route discovered through the broadcast route discovery fails, the destination attempts to predict the current location of each node using the location and mobility information collected during the latest broadcast route discovery. A minimum hop path Dijkstra algorithm [2] is run on the locally predicted global topology. If the predicted minimum hop route exists in reality, no expensive broadcast route discovery is needed and the source continues to send data packets on the discovered route; otherwise, the source initiates another broadcast route discovery.

4.2 Performance Metrics

The performance metrics studied are as follows:

- *Total Energy Lost per Route Discovery*: This is the average of the total energy consumed for the global broadcast route discovery attempts. This includes the sum of the energy consumed to broadcast a RREQ packet to all the nodes in the neighborhood and to receive the RREQ packet sent by each node in the neighborhood, summed over all the nodes.
- *Percentage of Total Energy Spent for Route Discovery*: This is the ratio of the total energy spent for route discovery to the sum of the energy spent across all the nodes in the network.
- *Time between Successive Route Discoveries*: This is the average of the time between two successive global broadcast based route discovery attempts. Larger the time between two successive route discoveries, lower will be the control overhead.
- *End-to-End Delay per Data Packet*: This is the average of the delay incurred by the data packets that originate at the source and delivered at the destination. The delay incurred by a data packet includes: the buffering delay due to the route acquisition latency, the queuing delay at the interface queue to access the medium, transmission delay, propagation delay, and the retransmission delay due to the MAC layer collisions.
- *Packet Delivery Ratio*: This is the ratio of the data packets delivered to the destination to the data packets originated at the source, averaged over all the source-destination sessions.

The performance results illustrated in Figures 2 through 6 are an average of simulations conducted with 5 mobility profiles for each operating condition.

4.3 Total Energy Spent for Route Discovery

For both DSR and LPBR, DMEF reduces the energy spent in the network due to global broadcast route discoveries (refer Figure 2). The reduction in the energy spent for route discoveries is more evident as we increase the network density and/or node mobility. In high-density networks, DMEF reduces the unnecessary rebroadcasts by broadcasting only through a reduced set of nodes to find a set of paths between a source and destination rather than broadcasting through all the nodes in the network. Compared to DSR, LPBR incurs relatively lower number of global broadcast based route discoveries. In addition, DMEF helps LPBR to reduce the energy spent per broadcast route discovery. Aided by both these factors, LPBR incurs a significantly lower energy due to route discoveries compared to DSR.

4.4 Percentage of Total Energy Spent for Route Discovery

As observed in Figures 3.1 through 3.3, for both DSR and LPBR, the difference in the percentage of total energy spent for route discovery using flooding and DMEF increases as we increase the network density and/or node mobility. For a given level of node mobility, the energy savings obtained with DMEF increases with increase in network density. Similarly, for a given network density, the energy savings obtained with DMEF, relative to flooding, increases with increase in the level of node mobility. For a given network density and level of node mobility, the relative reduction in the percentage of total energy spent for route discoveries due to the usage of DMEF vis-à-vis flooding is almost the same for both DSR and LPBR. Thus, DMEF can be used for energy-efficient route discovery for any MANET routing protocol.

4.5 Time between Successive Route Discoveries

DMEF prefers to broadcast the RREQ messages primarily through nodes that have been moving relatively slowly in the network. As a result, the routes determined using DMEF exist for a relatively longer time (refer Figure 4) in the network. For a given node density and mobility, the lifetime of routes determined for both DSR and LPBR protocols using DMEF as the route discovery strategy is significantly larger compared to that of the DSR and LPBR routes determined using flooding. In addition, LPBR attempts to increase the time between successive global broadcast discoveries by predicting a source-destination route using the Location Update Vectors (LUVs) collected during the latest broadcast route discovery. As we increase the network density, the chances of correctly predicting at least one source-destination path in the network increases. Hence, in the case of LPBR, for a given node mobility, the time between two successive global broadcast route discoveries increases as the network density increases. For both DSR and LPBR, compared to flooding, the relative increase in the lifetime of the routes discovered using DMEF and the reduction in the frequency of DMEF route discoveries can be significantly observed with increase in network density and/or node mobility.

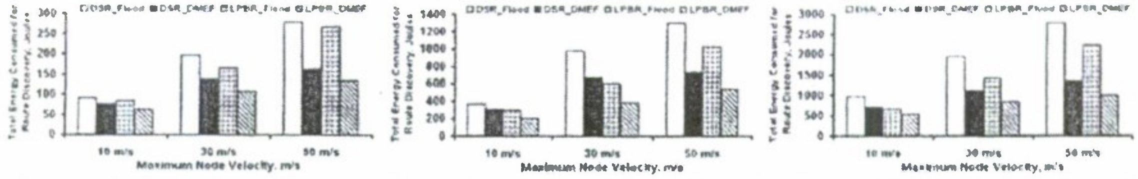


Figure 2.1: Low Density Network **Figure 2.2:** Moderate Density Network **Figure 2.3:** High Density Network

Figure 2: Total Energy Consumed for Route Discovery

Figure 3.1: Low Density Network **Figure 3.2:** Moderate Density Network **Figure 3.3:** High Density Network

Figure 3: Percentage of Total Energy Spent for Route Discovery

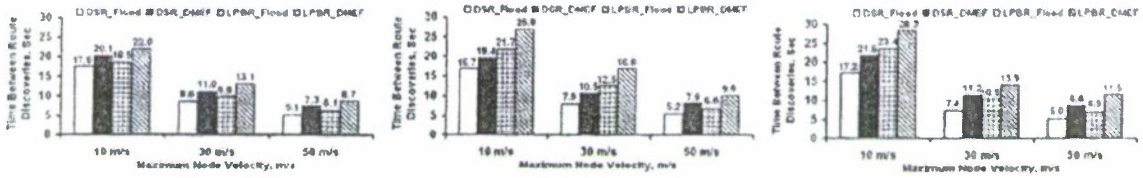


Figure 4.1: Low Density Network **Figure 4.2:** Moderate Density Network **Figure 4.3:** High Density Network

Figure 4: Time between Two Successive Route Discoveries

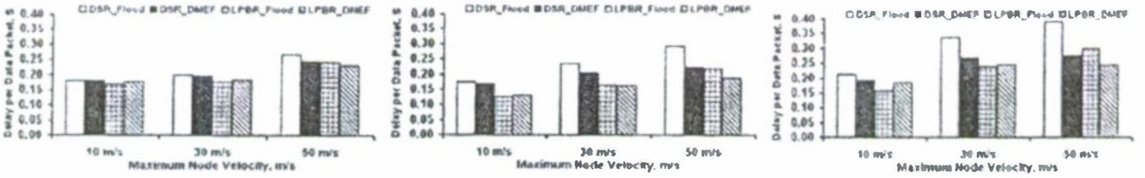


Figure 5.1: Low Density Network **Figure 5.2:** Moderate Density Network **Figure 5.3:** High Density Network

Figure 5: Average End-to-End Delay per Data Packet Delivered

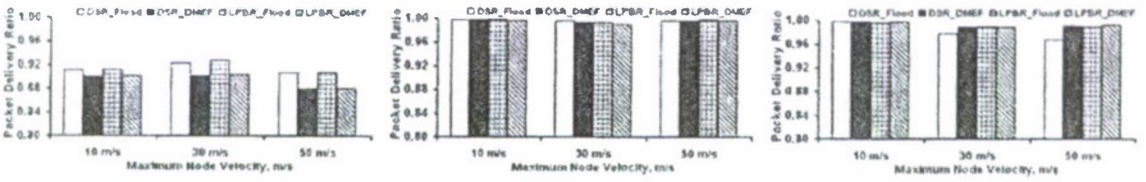


Figure 6.1: Low Density Network **Figure 6.2:** Moderate Density Network **Figure 6.3:** High Density Network

Figure 6: Packet Delivery Ratio

4.6 End-to-End Delay per Data Packet

DMEF exerts a relatively lower control overhead to determine routes compared to flooding. This is evident in the relatively lower end-to-end delay per data packet (refer Figure 5) incurred for DSR when routes are determined using DMEF compared to flooding. The relative difference between the end-to-end delays per data packet for DSR routes discovered using flooding and DMEF increases as we increase the node mobility and/or network density. With DSR, the route discovery overhead incurred due to relatively unstable routes discovered using flooding weighs far more than the slightly larger hop count of routes discovered using DMEF. In LPBR, there is a relatively slight reduction in the delays per data packet with DMEF in networks of high density/ high mobility. This is due to the relatively less congestion in the nodes attributed to the reduced number of route discovery attempts. In networks of low node mobility, the delay per data packet for LPBR with DMEF as the route discovery strategy is sometimes observed to be slightly larger than the delays per packet obtained with flooding. This is due to the slightly larger hop count of the paths discovered in such networks and a relatively lower route discovery overhead.

4.7 Packet Delivery Ratio

The packet delivery ratio (refer Figure 6) of both DSR and LPBR using DMEF is lower than that obtained using flooding only by at most 3% in low-density networks. In moderate density networks, both the route discovery strategies yield almost the same packet delivery ratio. In high density networks, the packet delivery ratio of routing protocols using DMEF can be larger than that obtained using flooding by about 3%. In high-density networks, even though flooding helps to propagate the RREQs through several routes, the excessive overhead generated by these redundant RREQs block the queues of certain heavily used nodes in the network, thus leading to sometimes a relatively lower packet delivery ratio compared to DMEF. In low-density networks, DMEF could very rarely fail to determine source-destination routes, even if one exists, due to its optimization approach of trying to shrink the range of broadcast of the RREQ messages. DMEF broadcasts RREQ messages over a relatively larger transmission range in low-density networks compared to those used for high-density networks. As we increase node density, the packet delivery ratio under both flooding and DMEF approaches unity.

5. Conclusions

The high-level contribution of this paper is the design and development of a novel density and mobility-aware, energy-efficient broadcast route discovery strategy (DMEF) that can simultaneously minimize the energy spent per route discovery and increase the lifetime of the routes discovered vis-à-vis flooding. Simulation results for both DSR and LPBR illustrate DMEF to be very effective in reducing the percentage of energy consumed due to route discoveries as well as in increasing the time between successive route discoveries. We conjecture that DMEF can be used with any MANET on-demand routing protocol to discover long-living routes with a reduced route discovery control overhead. Future work will involve studying the effectiveness of DMEF with multicast and multi-path MANET routing protocols.

6. Acknowledgments

This research has been funded by the Army Research Lab through grant number W911NF-08-2-0061. The author thanks Dr. Raju Namburu (Army Research Lab), Dr. Loretta A. Moore and Dr. Robert Whalin (both Jackson State University) for their useful reviews of this research activity.

7. References

[1] C. Bettstetter, H. Hartenstein and X. Perez-Costa, "Stochastic Properties of the Random-Way Point Mobility Model," *Wireless Networks*, vol. 10, no. 5, pp. 555 – 567, September 2004.

[2] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms," 2nd Edition, MIT Press/ McGraw-Hill, Sept. 2001.

[3] L. M. Feeney, "An energy-consumption model for performance analysis of routing protocols for mobile ad hoc networks," *Journal of Mobile Networks and Applications*, vol. 3, no. 6, pp. 239 – 249, June 2001.

[4] B. Hofmann-Wellenhof, H. Lichtenegger and J. Collins, *Global Positioning System: Theory and Practice*, 5th rev. ed., Springer, Sept. 2004.

[5] IEEE Standards Department, Wireless LAN Medium Access Control and Physical Layer Specifications, IEEE Standard 802.11, 1997.

[6] D. B. Johnson, D. A. Maltz and J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Ad hoc Networks," in *Ad hoc Networking*, Chapter 5, C. E. Perkins, Eds. Addison Wesley, pp. 139 – 172, 2000.

[7] N. Meghanathan, "Location Prediction Based Routing Protocol for Mobile Ad hoc Networks," *Proceedings of the IEEE Global Communications Conference*, New Orleans, Nov-Dec 2008.

[8] S. Ni, Y. Tseng, Y. Chen and J. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," *Proceedings of ACM International Conference on Mobile Computing and Networking*, pp.151-162, 1999.

[9] Y.-J. Suh, W. Kim and D.-H. Kwon, "GPS-Based Reliable routing Algorithms for Ad Hoc Networks," *The Handbook of Ad Hoc Wireless Networks*, pp. 361 – 374, CRC Press, 2003.

II Peer-reviewed Conference Publications/ Proceedings

C2. International Conference on Wireless Algorithms, Systems and Applications, Boston, USA

Multicast Extensions to the Location-Prediction Based Routing Protocol for Mobile Ad Hoc Networks

Natarajan Meghanathan

Jackson State University,
Jackson, MS 39217, USA
nmeghanathan@jsums.edu

Abstract. We propose multicast extensions to the location prediction-based routing protocol (NR-MLPBR and R-MLPBR) for mobile ad hoc networks to simultaneously reduce the number of tree discoveries, number of links and the hop count per path from the source to the multicast group. The multicast extensions work as follows: Upon failure of a path to the source, a receiver node attempts to locally construct a global topology using the location and mobility information collected during the latest global broadcast tree discovery. NR-MLPBR predicts a path that has the minimum number of hops to the source and R-MLPBR predicts a path to the source that has the minimum number of non-receiver nodes. If the predicted path exists in reality, the source accommodates the path as part of the multicast tree and continues to send the multicast packets in the modified tree. Otherwise, the source initiates another global broadcast tree discovery.

Keywords: Multicast Routing, Mobile Ad hoc Networks, Link Efficiency, Hop Count, Simulation.

1 Introduction

A mobile ad hoc network (MANET) is a dynamic distributed system of wireless nodes that move independent of each other in an autonomous fashion. Due to node mobility, routes between any pair of nodes frequently change and need to be reconfigured. As a result, on-demand route discovery is often preferred over periodic route discovery and maintenance, as the latter strategy will incur significant overhead due to the frequent exchange of control information among the nodes [1]. Multicasting is the process of sending a stream of data from one source node to multiple recipients by establishing a routing tree, which is an acyclic connected subgraph containing all the nodes in the network. The set of receiver nodes form the multicast group. The data gets duplicated, only when necessary, as it propagates down the tree. This is better than multiple unicast transmissions. Multicasting in ad hoc wireless networks has numerous applications, e.g., distributed computing applications like civilian operations, emergency search and rescue, warfare situations and etc.

In an earlier work [2], we developed a location prediction based routing (LPBR) protocol for unicast routing in MANETs. The specialty of LPBR is that it attempts to simultaneously reduce the number of global broadcast route discoveries as well as the

B. Liu et al. (Eds.): WASA 2009, LNCS 5682, pp. 190–199, 2009.

hop count of the paths for a source-destination session. LPBR works as follows: During a regular flooding-based route discovery, LPBR collects the location and mobility information of the nodes in the network and stores the collected information at the destination node of the route search process. When the minimum-hop route discovered through the flooding-based route discovery fails, the destination node attempts to predict the current location of each node using the location and mobility information collected during the latest flooding-based route discovery. A minimum hop path Dijkstra algorithm [3] is run on the locally predicted global topology. If the predicted minimum hop route exists in reality, no expensive flooding-based route discovery is needed and the source continues to send data packets on the discovered route; otherwise, the source initiates another flooding-based route discovery.

In this paper, we propose two multicast extensions to LPBR, referred to as NR-MLPBR and R-MLPBR. Both the multicast extensions are aimed at minimizing the number of global broadcast tree discoveries as well as the hop count per source-receiver path of the multicast tree. They use a similar idea of letting the receiver nodes to predict a new path based on the locally constructed global topology obtained from the location and mobility information of the nodes learnt through the latest broadcast tree discovery. Receiver nodes running NR-MLPBR (Non-Receiver aware Multicast extensions of LPBR) are not aware of the receivers of the multicast group, whereas each receiver node running R-MLPBR (Receiver-aware Multicast Extension of LPBR) is aware of the identity of the other receivers of the multicast group. NR-MLPBR attempts to predict a minimum hop path to the source, whereas R-MLPBR attempts to predict a path to the source that has the minimum number of non-receiver nodes. If more than one path has the same minimum number of non-receiver nodes, then R-MLPBR breaks the tie among such paths by choosing the path with the minimum number of hops to the source. Thus, R-MLPBR is also designed to reduce the number of links in the multicast tree, in addition to the average hop count per source-receiver path and the number of global broadcast tree discoveries.

The rest of the paper is organized as follows: Section 2 provides the detailed design of the two multicast extensions. Section 3 explains the simulation environment and illustrates the simulation results with respect to different performance metrics. Section 4 concludes the paper.

2 Multicast Extensions to LPBR

We assume periodic exchange of beacons in the neighborhood. We also assume that a multicast group comprises basically of receiver nodes that wish to receive data packets from an arbitrary source, which is not part of the multicast group.

2.1 Broadcast of Multicast Tree Request Messages

Whenever a source node has data packets to send to a multicast group and is not aware of a multicast tree to the group, the source initiates a broadcast tree discovery procedure by broadcasting a Multicast Tree Request Message (MTRM) to its neighbors. Each node, including the receiver nodes of the multicast group, on receiving the first MTRM of the current broadcast process (i.e., a MTRM with a sequence number greater than those seen before), includes its Location Update Vector, LUV in the MTRM packet. The LUV of a node comprises the following: node ID, X, Y co-ordinate information, Is Receiver flag, Current velocity and Angle of movement with respect to the X-axis. The *Is Receiver* flag in the LUV, if set, indicates that the node is a receiving node of the multicast group. The node ID is also appended on the "Route record" field of the MTRM packet. The structure of the LUV and the MTRM is shown in Figures 1 and 2 respectively.

Node ID	X Co-ordinate	Y Co-ordinate	Node Velocity	Angle of Movement	Is Receiver
4 bytes	8 bytes	8 bytes	8 bytes	8 bytes	1 bit

Fig. 1. Location Update Vector (LUV) per Node

Multicast Source	Multicast Group ID	Sequence Number	Route Recorded (List of Node IDs)	Location Update Vectors (LUVs)
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes	Variable Size of 36 bytes, 1 bit

Fig. 2. Structure of the Multicast Tree Request Message

Multicast Source	Originating Receiver	Multicast Group ID	Sequence Number	Route Record from the Receiver to the Source
4 bytes	4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes

Fig. 3. Structure of Multicast Tree Establishment Message

2.2 Construction of the Multicast Tree

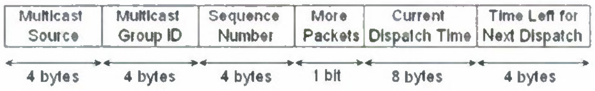
Paths constituting the multicast tree are independently chosen at each receiver node. A receiver node gathers several MTRMs obtained across different paths and selects the minimum hop path among them by looking at the "Route Record" field in these MTRMs. A Multicast Tree Establishment Message (MTEM) is sent on the discovered minimum hop route to the source. The MTEM originating from a receiver node has the list of node IDs corresponding to the nodes that are on the minimum hop path from the receiver node to the source (which is basically the reverse of the route recorded in the MTRM). The structure of the MTEM packet is shown in Figure 3.

An intermediate node upon receiving the MTEM packet checks its multicast routing table whether there exist an entry for the <Multicast Source, Multicast Group ID> in the table. If an entry exists, the intermediate node merely adds the tuple <One-hop sender of the MTEM, Originating Receiver node of the MTEM> to the list of <Downstream node, Receiver node> tuples for the multicast tree entry and does not forward the MTEM further. The set of downstream nodes are part of the multicast tree rooted at the source node for the multicast group. If a <Multicast Source, Multicast Group ID> entry does not exist in the multicast routing table, the intermediate node creates an entry and initializes it with the <One-hop sender of the MTEM, Originating

Receiver node of the MTEM> tuple. For each MTEM received, the source adds the neighbor node that sent the MTEM and the corresponding Originating Receiver node to the list of <Downstream node, Receiver node> tuples for the multicast group.

2.3 Multicast Tree Acquisition and Data Transmission

After receiving the MTEMs from all the receivers within the Tree Acquisition Time (*TAT*), the source starts sending the data packets on the multicast tree. The *TAT* is based on the maximum possible diameter of the network (an input parameter in our simulations). The diameter of a network is the maximum of the hop count of the minimum hop paths between any two nodes in the network. The *TAT* is dynamically set at a node based on the time it took to receive the first MTEM for a broadcast tree discovery procedure. The structure of the header of the multicast data packet is shown in Figure 4. In addition to the regular fields like Multicast Source, Multicast Group ID and Sequence Number, the header of the multicast data packet includes three specialized fields: the 'More Packets' (*MP*) field, the 'Current Dispatch Time' (*CDT*) field and the 'Time Left for Next Dispatch' (*TNLD*) field. The *CDT* field stores the time as the number of milliseconds lapsed since Jan 1, 1970, 12 AM. These additional overhead (relative to that of the other ad hoc multicast routing protocols) associated with the header of each data packet amounts to only 12 more bytes per data packet.



Multicast Source	Multicast Group ID	Sequence Number	More Packets	Current Dispatch Time	Time Left for Next Dispatch
4 bytes	4 bytes	4 bytes	1 bit	8 bytes	4 bytes

Fig. 4. Structure of the Header of the Multicast Data Packet

The source sets the *CDT* field in all the data packets sent. If the source has any more data to send, it sets the *MP* flag to 1 and sets the appropriate value for the *TLND* field, which indicates the number of milliseconds since the *CDT*. If the source does not have any more data to send, it will set the *MP* flag to 0 and leaves the *TLND* field blank. As we assume the clocks across all nodes are synchronized, a receiver will be able to calculate the end-to-end delay for the data packet based on the time the data packet reaches the node and the *CDT* field in the header of the data packet. An average end-to-end delay per data packet is maintained at the receiver for the current path to the source. If the source node has set the MP flag, the receiver computes the 'Next Expected Packet Arrival Time' (*NEPAT*), as the *CDT* field + *TLND* field + 2*Average end-to-end delay per data packet. A timer is started for the *NEPAT* value.

2.4 Multicast Tree Maintenance

If an intermediate node notices that its link with a downstream node has failed (i.e., the two nodes have moved away and are no longer neighbors), the intermediate node generates and sends a Multicast Path Error Message (MPEM) to the source of the multicast group entry. The MPEM has information about the receiver nodes affected (obtained from the multicast routing table) because of the link failure with the downstream node. Figure 5 shows the structure of an MPEM. The intermediate node removes the tuple(s) corresponding to the downstream node(s) and the affected receiver node(s). After these deletions, if no more <Downstream node, Receiver node> tuple exists for a <Source node, Multicast group ID> entry, the intermediate node removes the entire row for this entry from the routing table.

Multicast Source	Originating Intermediate Node	Multicast Group ID	IDs of Affected Receivers
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes

Fig. 5. Structure of a MPEM Message

The source, upon receiving the MPEM, will wait to receive a Multicast Predicted Path Message (MPPM) from each of the affected receivers, within a MPPM-timer maintained for each receiver. The source estimates a Tree-Repair Time (*TRT*) for each receiver as the time that lapsed between the reception of the MPEM from an intermediate node and the MPPM from the affected receiver. An average value for the TRT per receiver is maintained at the source as it undergoes several path failures and repairs before the next global broadcast based tree discovery. The MPPM-timer (initially set to the time it took for the source to receive the MTEM from the receiver) for a receiver will be then set to 1.5* Average *TRT* value, so that we give sufficient time for the destination to learn about the route failure and generate a new MPPM. Nevertheless, this timer will be still far less than the tree acquisition time that would be incurred if the source were to launch a global broadcast tree discovery. Hence, our approach will only increase the network throughput and does not decrease it.

2.5 Prediction of Node Location Using the LUVs

If a multicast receiver does not receive the data packet within the *NEPAT* time, it will attempt to locally construct the global topology using the location and mobility information of the nodes learnt from the latest broadcast tree discovery. Each node is assumed to be moving in the same direction with the same speed as mentioned in its latest LUV. Based on this assumption and information from the latest LUVs, the location of each node at the *NEPAT* time is predicted.

We now explain how to predict the location of a node (say node u) at a time instant *CTIME* based on the LUV gathered from node u at time *STIME*. Let $(X_u^{STIME}, Y_u^{STIME})$ be the X and Y co-ordinates of u at time *STIME*. Let $Angle_u^{STIME}$ and $Velocity_u^{STIME}$ represent the angle of movement with respect to the X-axis and the velocity at which node u is moving. The distance traveled by node u from time *STIME* to *CTIME* would be: $Distance_u^{STIME\text{-}CTIME} = (CTIME - STIME + 1) * Velocity_u^{STIME}$. We assume each node is initially configured with information regarding the network boundaries, given by [0, 0], $[X_{max}, 0]$, $[X_{max}, Y_{max}]$ and $[0, Y_{max}]$. Let $(X_u^{CTIME}, Y_u^{CTIME})$ be the predicted location of node u at time *CTIME*.

$$X_u^{CTIME} = X_u^{STIME} + Offset\text{-}X_u^{CTIME}\ ; \quad Y_u^{CTIME} = Y_u^{STIME} + Offset\text{-}Y_u^{CTIME}$$
$$Offset\text{-}X_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \cos(Angle_u^{STIME})$$
$$Offset\text{-}Y_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \sin(Angle_u^{STIME})$$

If $(X_u^{CTIME} < 0)$, then $X_u^{CTIME} = 0$; If $(X_u^{CTIME} > X_{max})$, then $X_u^{CTIME} = X_{max}$
If $(Y_u^{CTIME} < 0)$, then $Y_u^{CTIME} = 0$; If $(Y_u^{CTIME} > Y_{max})$, then $Y_u^{CTIME} = Y_{max}$

2.6 Multicast Path Prediction

NR-MLPBR: The receiver node locally runs the Dijkstra's minimum hop path algorithm [3] on the predicted global topology. If at least one path exists from the source to the receiver in the generated topology, the algorithm returns the minimum hop path among them. The receiver node then sends a Multicast Predicted Path Message, MPPM (structure shown in Figure 6), on the discovered path with the route information included in the message.

R-MLPBR: The receiver node uses the LUV obtained from each of the intermediate nodes during the latest global tree broadcast discovery to learn about the identification of its peer receiver nodes that are part of the multicast group. If there existed a direct path to the source on the predicted topology, the receiver chooses that path as the predicted path towards the source. Otherwise, the receiver determines a set of node-disjoint paths on the predicted global topology. The node-disjoint paths to the source are ranked depending on the number of non-receiver nodes that act as intermediate nodes on the path. The path that has the least number of non-receiver nodes as intermediate nodes is preferred. The reason is a path that has the least number of non-receiver nodes is more likely to be a minimum hop path and if a receiver node lies on that path, the number of newly added links to the tree would also be reduced. R-MLPBR thus aims to discover paths with the minimum hop count and at the same time attempts to conserve bandwidth by reducing the number of links that get newly added to the tree as a result of using the predicted path. The MPPM is hence sent on the predicted path that has minimum number of non-receiver nodes. If two or more paths has the same minimum number of non-receiver nodes, R-MLPBR breaks the tie by choosing the path with the minimum hop count to the source.

Multicast Source	Originating Receiver Node	Multicast Group ID	Predicted Path to the Multicast Source (List of Node IDs)
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes

Fig. 6. Structure of the Multicast Predicted Path Message

2.7 Propagation of the Multicast Predicted Path Message towards the Source

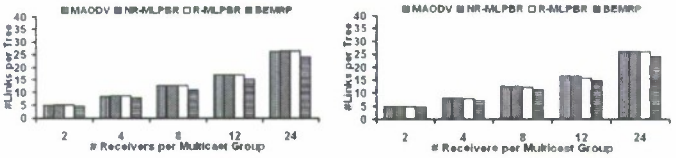
An intermediate node on receiving the MPPM adds the tuple <One-hop sender of the MPPM, Originating Receiver node of the MPPM> to the list of <Downstream node, Receiver node> tuples for the multicast tree entry corresponding to the source node and the multicast group to which the MPPM belongs to. The MPPM is then forwarded to the next downstream node on the path towards the source. If the source node receives the MPPM from the appropriate receiver node before the MPPM-timer expires, it indicates that the predicted path does exist in reality. A costly global broadcast tree discovery has been thus avoided. If an intermediate node could not successfully forward the MPPM to the next node on the path towards the source, it informs the receiver node of the absence of the route through a MPPM-Error packet. The receiver node on receiving the MPPM-Error packet discards all the LUVs and does not generate any new MPPM. After the MPPM-timer expires, the multicast source initiates a new global broadcast-based tree discovery procedure.

3 Simulations

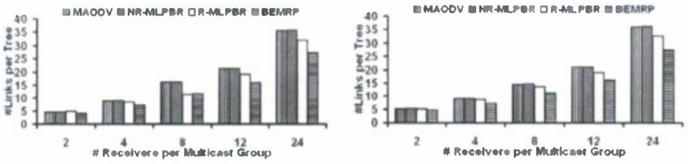
We use a 1000m x 1000m square network. The transmission range per node is 250m. The number of nodes used in the network is 25 and 75 nodes representing networks of low and high density respectively. We compare the performance of NR-MLPBR and R-MLPBR with that of the Multicast Extension [4] of the Ad hoc On-demand Distance Vector [5] (MAODV) routing protocol that minimizes the hop count per source-receiver path and the Bandwidth-Efficient Multicast Routing Protocol (BEMRP) [6] that minimizes the number of links in the multicast tree. We implemented all of these four multicast routing protocols in a discrete-event simulator developed in Java. The simulation parameters are summarized in Table 1.

Table 1. Simulation Conditions

Physical Layer	Propagation Model: Two-ray ground reflection model [1]
MAC Layer	IEEE 802.11 [7], Bandwidth: 2 Mbps, Queue Size: 100
Routing Protocols	BEMRP [6], MAODV [4], NR-MLPBR and R-MLPBR
Mobility Model	Random Way Point Model [8]: Min. Node Speed = 0 m/s,
	Pause Time: 0 s, Max. Node Speed = 10 m/s and 50 m/s
Traffic Model	Constant Bit Rate (CBR), UDP
	# Receivers: 2 (small), 4 and 8 (medium), 12 and 24 (high)
	Data Packet Size:512 bytes, Packet Sending Rate: 4/second



(a) 25 nodes, 10 m/s (b) 25 nodes, 50 m/s

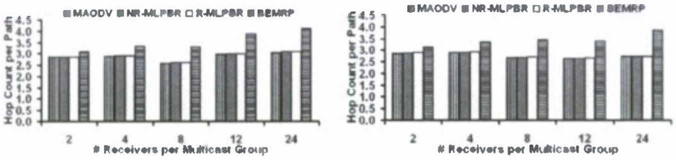


(c) 75 nodes, 10 m/s (d) 75 nodes, 50 m/s

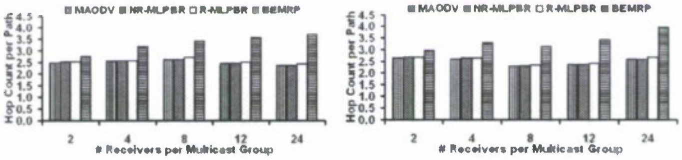
Fig. 7. Average Number of Links per Multicast Tree

The performance metrics studied through the simulations are the following computed over the duration of the entire multicast session. Each of the performance results in Figures 7 through 9 are an average of the results obtained from simulations conducted with 5 sets of multicast groups and 5 sets of mobility profiles.

- **Number of Links per Tree:** This is the time averaged number of links in the multicast trees discovered and computed over the entire multicast session.
- **Hop Count per Source-Receiver Path:** This is the time averaged hop count of the paths from the source to each receiver of the multicast group.
- **Time between Successive Broadcast Tree Discoveries:** This is the average of the time between two successive broadcast tree discoveries.



(a) 25 nodes, 10 m/s (b) 25 nodes, 50 m/s



(c) 75 nodes, 10 m/s (d) 75 nodes, 50 m/s

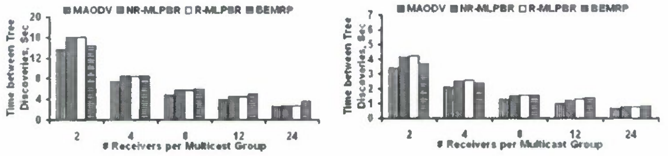
Fig. 8. Average Hop Count per Source-Receiver Path

3.1 Number of Links per Multicast Tree

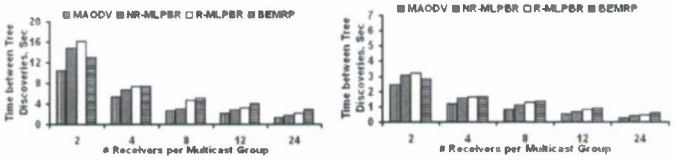
R-MLPBR manages to significantly reduce the number of links vis-à-vis the MAODV and NR-MLPBR protocols without yielding to a higher hop count per source-receiver path. R-MLPBR is the first multicast routing protocol that yields trees with the reduced number of links and at the same time, with a reduced hop count (close to the minimum) per source-receiver path. However, R-MLPBR cannot discover trees that have minimum number of links as well as the minimum hop count per source-receiver path. The BEMRP protocol discovers trees that have a reduced number of links for all the operating scenarios. However, this leads to larger hop count per source-receiver paths for BEMRP as observed in figure 8.

3.2 Average Hop Count per Source-Receiver Path

All the three multicast routing protocols – MAODV, NR-MLPBR and R-MLPBR, incur almost the same average hop count per source-receiver path and it is considerably lower than that incurred for BEMRP. The hop count per source-receiver path is an important metric and it is often indicative of the end-to-end delay per multicast packet from the source to a specific receiver. BEMRP incurs a significantly larger hop count per source-receiver path and this can be attributed to the nature of this multicast routing protocol to look for trees with a reduced number of links. When multiple receiver nodes have to be connected to the source through a reduced set of links, the hop count per source-receiver path is bound to increase. The hop count per source-receiver path increases significantly as we increase the multicast group size.



(a) 25 nodes, 10 m/s (b) 25 nodes, 50 m/s



(c) 75 nodes, 10 m/s (d) 75 nodes, 50 m/s

Fig. 9. Average Time between Successive Tree Discoveries

3.3 Time between Successive Broadcast Tree Discoveries

The time between successive broadcast tree discoveries is a measure of the stability of the multicast trees and the effectiveness of the location prediction and path prediction approach of the two multicast extensions. For a given condition of node density and node mobility, both NR-MLPBR and R-MLPBR incur relatively larger time between successive broadcast tree discoveries for smaller and medium sized multicast groups. MAODV tends to be more unstable as the multicast group size is increased, owing to the minimum hop nature of the paths discovered and absence of any path prediction approach. For larger multicast groups, the multicast trees discovered using BEMRP are relatively more stable by virtue of the protocol's tendency to strictly minimize only the number of links in the tree.

4 Conclusions and Future Work

The number of links per tree discovered using R-MLPBR is only about 15-20% more than that discovered using BEMRP, but the hop count per source-receiver path is significantly smaller (by about 40%-60%) than those observed in trees discovered using BEMRP and is the same as that discovered using MAODV. NR-MLPBR and R-MLPBR incur larger time between successive tree discoveries for smaller and medium sized multicast groups, where as BEMRP discovers stable trees for larger multicast groups. We conjecture that with the deployment of broadcast tree discovery strategies (such as DMEF [9]) that can discover inherently stable trees, the performance of NR-MLPBR and R-MLPBR with respect to the time between successive tree discoveries can be further improved vis-à-vis BEMRP and MAODV.

Acknowledgments. Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-08-2-0061. The views and conclusions in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

1. Broch, J., Maltz, D.A., Johnson, D.B., Hu, Y.C., Jetcheva, J.: A Performance of Comparison of Multi-hop Wireless Ad hoc Network Routing Protocols. In: 4th International Conference on Mobile Computing and Networking, Dallas, pp. 85–97. ACM, New York (1998)
2. Meghanathan, N.: Location Prediction Based Routing Protocol for Mobile Ad Hoc Networks. In: Global Communications Conference, New Orleans. IEEE, Los Alamitos (2008)
3. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn. MIT Press/ McGraw Hill, New York (2001)
4. Royer, E., Perkins, C.E.: Multicast Operation of the Ad-hoc On-demand Distance Vector Routing Protocol. In: 5th Conference on Mobile Computing and Networking, Seattle, USA, pp. 207–218. ACM, New York (1999)
5. Perkins, C.E., Royer, E.M.: The Ad hoc On-demand Distance Vector Protocol. In: Perkins, C.E. (ed.) Ad hoc Networking, pp. 173–219. Addison-Wesley, New York (2000)
6. Ozaki, T., Kim, J.-B., Suda, T.: Bandwidth-Efficient Multicast Routing for Multihop, Ad hoc Wireless Networks. In: 20th International Conference of Computer and Communications Societies, Anchorage, USA, vol. 2, pp. 1182–1192. IEEE, Los Alamitos (2001)
7. Bianchi, G.: Performance Analysis of the IEEE 802.11 Distributed Coordination Function. IEEE Journal of Selected Areas in Communication 18(3), 535–547 (2000)
8. Bettstetter, C., Hartenstein, H., Perez-Costa, X.: Stochastic Properties of the Random-Waypoint Mobility Model. Wireless Networks 10(5), 555–567 (2004)
9. Meghanathan, N.: A Density and Mobility Aware Energy-Efficient Broadcast Strategy to Determine Stable Routes in Mobile Ad hoc Networks. submitted for journal publication (2009)

II Peer-reviewed Conference Publications/ Proceedings

C3. International Conference on Signal Processing and Communication Systems, Omaha, USA

A Node-Disjoint Multi-path Extension of the Location Prediction Based Routing Protocol for Mobile Ad hoc Networks

Natarajan Meghanathan
Jackson State University
1400 John R. Lynch Street
Jackson, MS 39217
E-mail: natarajan.meghanathan@jsums.edu

Abstract – **We propose a node-disjoint multi-path extension to the location prediction-based routing protocol (LPBR-M) to reduce the number of broadcast multi-path route discoveries for mobile ad hoc networks. During a broadcast route discovery, the intermediate forwarding nodes include their location and mobility information in the Route-Request messages. Upon failure of all the node-disjoint paths learnt from the latest route discovery, the destination runs the algorithm to determine the set of node-disjoint paths on a predicted global topology, constructed from the location and mobility information collected during the latest broadcast route discovery, and sends a sequence of Route-Reply messages on each of the predicted paths. If the source receives at least one Route-Reply message within certain time, it continues to send the data packets along the newly learnt node-disjoint paths. Otherwise, the source initiates another broadcast route discovery. Simulation results of LPBR-M along with the link-disjoint path based AOMDV and node-disjoint path based AODVM routing protocols indicate that LPBR-M incurs the longest time between successive broadcast route discoveries and a hop count close to that incurred by the minimum hop count based multi-path protocols.**

I. INTRODUCTION

On-demand routing protocols for mobile ad hoc networks (MANETs) incur high route discovery latency and frequent route discoveries in the presence of a dynamically changing topology. Recent research has started to focus on multi-path routing protocols that tend to compute multiple paths, at both the traffic sources as well as at intermediary nodes, in a single route discovery attempt. This reduces both the route discovery latency and the control overhead as a route discovery is needed only when all the discovered paths fail. Spreading the traffic along several routes could alleviate congestion and bottlenecks. Multi-path routing also provides a higher aggregate bandwidth and effective load balancing as the data forwarding load is effectively distributed over all the paths.

Multi-paths can be of two types: link-disjoint and node-disjoint. For a given source *s* and destination *d*, the set of link-disjoint *s-d* routes comprises of paths that have no link present in more than one constituent *s-d* path. Similarly, the set of node-disjoint *s-d* routes comprises of paths that have no node (other than the source and destination) present in more than one constituent *s-d* path. MANET multi-path routing protocols make use of the propagation of the Route-Request (RREQ) messages along several paths to the destination and let the destination to send Route-Reply (RREP) along more than one path. The routing protocols avoid the RREP storm by selecting only few of the different paths. Since nodes communicate through the shared wireless medium, the selected paths need to be as independent as possible in order to avoid transmissions from a node along one path interfering with transmissions on a different path. Thus, the aggregate bandwidth achieved with multi-path routing may not be the sum of the bandwidth of the individual paths. Node-disjoint routes offer the highest degree of fault tolerance and aggregate bandwidth [1]. Throughout the paper, the terms path and route are used interchangeably. They mean the same.

Most of the MANET multi-path routing protocols are either extensions of the Dynamic Source Routing (DSR) protocol [2] or the Ad hoc On-demand Distance Vector (AODV) routing protocol [3]. Examples are: (i) Split multi-path routing (SMR) [4] protocol, an extension of DSR; (ii) Ad hoc On-demand Multi-path Distance Vector (AOMDV) routing protocol [5], an extension of AODV to compute multiple loop-free link-disjoint routes; (iii) AODV-Multi-path (AODVM) routing protocol [6], an extension of the AODV protocol to determine node-disjoint routes; (iv) Geographic Multi-path Routing Protocol (GMRP) [7] to reduce interference due to route coupling and (v) Energy-aware Multi-path Routing Protocol (EMRP) [8] that considers the available energy and forwarding load at intermediate nodes of the multiple paths before distributing the load across them.

In [9], we developed a location prediction based routing (LPBR) protocol for single-path unicast routing in MANETs. LPBR attempts to simultaneously reduce the number of global broadcast route discoveries as well as the hop count of the paths for a source-destination session. LPBR works as follows: During a regular broadcast route discovery, LPBR collects the location and mobility information of the nodes in the network in the form of Location Update Vectors (LUVs) and stores the LUVs at the destination node of the route search process. When the minimum-hop route discovered through the broadcast route discovery fails, the destination node attempts to predict the current location of each node using the location and mobility information collected during the latest broadcast route discovery. A minimum hop path Dijkstra algorithm [10] is run on the locally predicted global topology. If the predicted

978-1-4244-4474-8/09/$25.00 ©2009 IEEE

minimum hop route exists in reality, no expensive broadcast route discovery is needed and the source continues to send data packets on the discovered route; otherwise, the source initiates another broadcast route discovery.

In this paper, we develop a node-disjoint multi-path extension to the LPBR protocol, referred to as LPBR-M. In [11], we observed that the number of broadcast route discoveries needed for node-disjoint multi-path routing is not significantly different from the number of route discoveries for link-disjoint multi-path routing. Also, there is no much difference in the average hop count of the node-disjoint paths and the link-disjoint paths. On the other hand, node-disjoint paths are preferred for fault tolerance, load balancing and extending the lifetime of the nodes. LPBR-M minimizes the control overhead by reducing the number of broadcast route discoveries as much as possible using multi-path routing. Also, LPBR-M yields an average hop count per multi-path that is almost equal to that of the minimum-hop based multi-path routing protocols. The rest of the paper is organized as follows: Section II provides a detailed design of the LPBR-M protocol. Section III describes the simulation environment, defines the performance metrics, presents the simulation results and interprets them. Section IV concludes the paper.

II. DESIGN OF THE LPBR-M PROTOCOL

We assume that the clocks across all nodes are synchronized. This is essential to ensure proper timeouts at the nodes for failure to receive a certain control message.

A. Broadcast of Route-Request Messages

When a source has data and is not aware of a path to send data packets to a destination, it initiates a route discovery procedure by broadcasting a Multi-path Route Request (MP-RREQ) message to its neighbors. Each node, except the destination, on receiving the first MP-RREQ of the current broadcast process (i.e., a MP-RREQ with a sequence number greater than those seen before), includes its Location Update Vector, LUV, in the MP-RREQ message. The LUV of a node (ref. Fig. 1) comprises the following: Node ID, X, Y co-ordinates, Current velocity and Angle of movement with respect to the X-axis. The Node ID is also appended in the "Route Record" field of the MP-RREQ (ref. Fig. 2).

Node ID	X Co-ordinate	Y Co-ordinate	Node Velocity	Angle of Movement
4 bytes	8 bytes	8 bytes	8 bytes	8 bytes

Fig. 1. Location Update Vector (LUV)

Source ID	Destination ID	Sequence Number	Route Recorded (List of Node IDs)	Location Update Vectors (LUVs)
4 bytes	4 bytes	4 bytes	Variable Size of 4 bytes	Variable Size of 36 bytes

Fig. 2. Multi-path Route Request (MP-RREQ)

B. Generation of the Route-Reply Messages

When the destination receives a MP-RREQ message, it extracts the path traversed by the message (sequence of Node IDs in the Route Record) and the LUVs of the nodes (including the source) that forwarded the message. The destination stores the paths learnt in a set, *RREQ-Path-Set*, maintained in the increasing order of their hop count. Ties between paths with the same hop count are broken in the order of the time of arrival of their corresponding MP-RREQ messages at the destination. The LUVs are stored in a LUV-Database maintained for the latest broadcast route discovery procedure initiated by the source. The destination runs a local path selection heuristic to extract the set of node-disjoint paths, *RREQ-ND-Set*, from the *RREQ-Path-Set*. The heuristic makes sure that except the source and the destination nodes, a node can serve as an intermediate node in at most only one path in the *RREQ-ND-Set*. The *RREQ-ND-Set* is initialized and updated with the paths extracted from the *RREQ-Path-Set* satisfying this criterion.

Once the *RREQ-ND-Set* is built, the destination sends a Multi-path Route Reply (MP-RREP) message for every path in the *RREQ-ND-Set*. An intermediate node receiving the MP-RREP message (ref. Fig. 3) updates its routing table by adding the neighbor that sent the message as the next hop on the path from the source to the destination. The MP-RREP message is then forwarded to the next node towards the source as indicated in the Route Record field of the message.

Originating Source ID of the MP-RREQ	Targeted Destination ID of the MP-RREQ	Sequence Number of the MP-RREQ	Route Recorded in the MP-RREQ (List of Node IDs)
4 bytes	4 bytes	4 bytes	Variable Size Multiples of 4 bytes

Fig. 3. Multi-path Route Reply (MP-RREP)

C. Multi-path Acquisition Time and Data Transmission

After receiving the MP-RREP messages from the destination within a certain time called the Multi-path Acquisition Time (*MP-AT*), the source stores the paths learnt in a set of node-disjoint paths, *NDP-Set*. The *MP-AT* is based on the maximum possible diameter of the network (an input parameter in our simulations). The diameter of the network is the maximum of the hop count of the minimum hop paths between any two nodes in the network. The *MP-AT* is dynamically set at a node depending on the time it took to receive the first MP-RREP for a broadcast discovery process.

Source ID	Destination ID	Sequence Number	Number of Disjoint Paths	More Packets	Current Dispatch Time	Time Left for Next Dispatch
4 bytes	4 bytes	4 bytes	1 byte	1 bit	8 bytes	4 bytes

Fig. 4. Structure of the Data Packet

For data transmission, the source uses the path with the minimum hop count among the paths in the *NDP-Set*. The structure of a data packet is illustrated in Fig. 4. In addition to the regular fields of source and destination IDs and the sequence number, the header of the data packet includes four specialized fields: the 'Number of Disjoint Paths' field that indicates the number of active node-disjoint paths currently being stored in the *NDP-Set* of the source, the 'More Packets'

(*MP*) field, the 'Current Dispatch Time' (*CDT*) field and the 'Time Left for Next Dispatch' (*TNLD*) field. The *CDT* field stores the time as the number of milliseconds lapsed since Jan 1, 1970, 12 AM. These additional overhead (relative to the other routing protocols) associated with the header amounts to only 13 more bytes per data packet.

The source sets the *CDT* field in all the data packets sent. In addition, if the source has any more data to send, it sets the *MP* flag to 1 and sets the appropriate value for the *TLND* field, which indicates the number of milliseconds since the *CDT*. If the source does not have any more data to send, it will set the *MP* flag to 0 and leaves the *TLND* field blank. As we assume the clocks across all nodes are synchronized, the destination node uses the *CDT* field in the header of the data packet and the time of arrival of the packet to update the average end-to-end delay per data packet for the set of multi-paths every time after receiving a new data packet on one of these paths. If the MP flag is set, the destination node computes the 'Next Expected Packet Arrival Time' (*NEPAT*), which is *CDT* field + *TLND* field + 2**NDP-Set Size**Average end-to-end delay per data packet. A timer is started for the *NEPAT* value. In order for the destination to wait until the source manages to successfully route a packet along a path in the *NDP-Set*, the *NEPAT* time takes the *NDP-Set Size* into account.

D. Multi-path Maintenance

If an intermediate node could not forward the data packet due to a broken link, the upstream node of the broken link informs about the broken route to the source node through a Multi-path-Route-Error (MP-RERR) message, structure shown in Fig. 5. The source node on learning the route failure will remove the failed path from its *NDP-Set* and attempt to send data packet on the next minimum-hop path in the *NDP-Set*. If this path is actually available in the network at that time instant, the data packet will successfully propagate its way to the destination. Otherwise, the source receives a MP-RERR message on the broken path, removes the failed path from the *NDP-Set* and attempts to route the data packet on the next minimum hop path in the *NDP-Set*. This procedure is repeated until the source does not receive a MP-RERR message or runs out of an available path in the *NDP-Set*. In the former case, the data packet successfully reaches the destination and the source continues to transmit data packets as scheduled. In the latter case, the source is not able to successfully transmit the data packet to the destination.

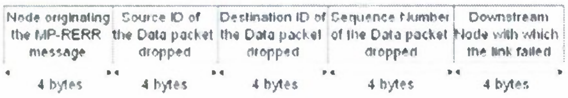


Fig. 5. Multi-path Route Error (MP-RERR) Message

Before initiating another broadcast route discovery procedure, the source will wait for the destination node to inform it of a new set of node-disjoint routes through a sequence of MP-LPBR-RREP messages. The source will run a *MP-LPBR-RREP-timer* and wait to receive at least one MP-LPBR-RREP message from the destination. For the failure of the first set of node-disjoint paths, the value of this timer would be set to the multi-path acquisition time (the time it took to get the first MP-RREP message from the destination since the inception of route discovery), so that we give sufficient time for the destination to learn about the route failure and generate a new sequence of MP-LPBR-RREP messages. For subsequent route-repairs, the *MP-LPBR-RREP-timer* will be set based on the time it takes to get the first MP-LPBR-RREP message from the destination.

E. Prediction of Node Location using LUVs

If the destination does not receive the data packet within the *NEPAT* time, it will attempt to locally construct the global topology using the location and mobility information of the nodes learnt from the latest broadcast route discovery. Each node is assumed to be continuing to move in the same direction with the same velocity as mentioned in its latest LUV. Based on this assumption and information from the latest LUVs, the location of each node at the *NEPAT* time is predicted. Note that there exists an edge between two nodes in the locally constructed global topology, if the predicted distance between the two nodes is less than or equal to the transmission range of the nodes.

We now explain how to predict the location of a node (say node u) at a time instant *CTIME* based on the LUV gathered from u at time *STIME*. Let $(X_u^{STIME}, Y_u^{STIME})$ be the X and Y co-ordinates of node u at time *STIME*. Let $Angle_u^{STIME}$ and $Velocity_u^{STIME}$ represent the angle of movement with respect to the X-axis and the velocity at which u is moving. The distance traveled by node u from time *STIME* to *CTIME* would be: $Distance_u^{STIME\text{-}CTIME} = (CTIME - STIME + 1)^* \ Velocity_u^{STIME}$. We assume each node is initially configured with information regarding the network boundaries: [0, 0], $[X_{max}, 0]$, $[X_{max}, Y_{max}]$ and $[0, Y_{max}]$.

Let $(X_u^{CTIME}, Y_u^{CTIME})$ be the predicted location of node u at time *CTIME*. The value of X_u^{CTIME} and Y_u^{CTIME} are given by $X_u^{STIME}+Offset\text{-}X_u^{CTIME}$ and $Y_u^{STIME}+Offset\text{-}Y_u^{CTIME}$ respectively. The offsets in the X and Y-axes depend on angle of movement and distance traveled. They are calculated as follows:

$$Offset\text{-}X_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \cos(Angle_u^{STIME})$$
$$Offset\text{-}Y_u^{CTIME} = Distance_u^{STIME\text{-}CTIME} * \sin(Angle_u^{STIME})$$

If $(X_u^{CTIME} < 0)$, then $X_u^{CTIME} = 0$;
If $(X_u^{CTIME} > X_{max})$, then $X_u^{CTIME} = X_{max}$
If $(Y_u^{CTIME} < 0)$, then $Y_u^{CTIME} = 0$;
If $(Y_u^{CTIME} > Y_{max})$, then $Y_u^{CTIME} = Y_{max}$

F. LPBR-M: Multi-path Prediction

The destination locally runs the algorithm for determining the set of node-disjoint paths [11] on the predicted global topology. The algorithm is explained as follows: Let G (V, E) be the graph representing the predicted global topology, where V is the set of vertices and E is the set of edges in the predicted network graph. Let P_N denote the set of node-disjoint s-d paths between source s and destination d. To start with, we run the $O(|V|^2)$ Dijkstra algorithm [10] on G to determine the minimum hop s-d path. If there is at least one s-d path in G, we include the minimum hop *s-d* path *p* in the set P_N. We then remove all the intermediate nodes (nodes other than source *s* and destination *d*) that were part of the minimum-hop *s-d* path *p* in the original graph *G* to obtain the modified graph *G*' (*V*', *E*'). We then determine the minimum-hop *s-d* path in *G*' (*V*', *E*'), add it to the set P_N and remove the intermediate nodes that were part of this *s-d* path to get a new updated *G*' (*V*', *E*'). We repeat this procedure until there exists no more *s-d* paths in the network. The set P_N contains the node-disjoint *s-d* paths in the original network graph *G*. When we remove a node from a graph, we also remove all the links associated with the node.

G. MP-LPBR-RREP Message Propagation

The destination *d* sends a MP-LPBR-RREP message (ref. Fig. 6) to the source *s* on each of the predicted node-disjoint paths. Each intermediate node receiving the MP-LPBR-RREP message updates its routing table to record the incoming interface of the message as the outgoing interface for any new data packets received from the source *s* to the destination *d*. The MP-LPBR-RREP message has a "Number of Disjoint Paths' field to indicate the total number of paths predicted and a 'Is Last Path' Boolean field that indicates whether or not the reported path is the last among the set of node-disjoint paths predicted. If the source *s* receives at least one MP-LPBR-RREP message before the *MP-LPBR-RREP-timer* expires, it indicates that the corresponding predicted *s-d* path on which the message propagated through does exists in reality. The source node creates a new instance of the *NDP-Set* and stores all the newly learnt node-disjoint *s-d* routes and starts sending data on the minimum hop path among them.

Source Node of the Session	Destination Node of the Session	Sequence Number of the Latest MP-RREQ	Number of Disjoint Paths	Is Last Path	Predicted Source – Destination Path (List of Node IDs)
4 bytes	4 bytes	4 bytes	1 byte	1 bit	Variable Size: Multiples of 4 bytes

Fig. 6. Structure of the MP-LPBR-RREP Message

The source estimates the Route-Repair Time (*RRT*) as the time that lapsed between the reception of the last MP-RERR message from an intermediate node and the first MP-LPBR-RREP message from the destination. An average value of the *RRT* is maintained at the source as it undergoes several route failures and repairs before the next broadcast route discovery. The *MP-LPBR-RREP-timer* (initially set to the multi-path acquisition time) will be then set to 1.25*Average *RRT* value, so that the destination gets sufficient time to learn about the route failure and generate the MP-LPBR-RREP messages.

H. Handling Prediction Failure

If an intermediate node attempting to forward a MP-LPBR-RREP message of the destination could not successfully forward the message to the next node on the path towards the source, the intermediate node informs the absence of the route through a MP-LPBR-RREP-RERR message (ref. Fig. 7) sent back to the destination. If the destination receives MP-LPBR-RREP-RERR messages for all the MP-LPBR-RREP messages initiated or the *NEPAT* time has expired, then the node discards all the LUVs and does not generate any new MP-LPBR-RREP message. The destination will wait for the source node to initiate a broadcast route discovery. After the *MP-LPBR-RREP-timer* expires, the source node initiates a new broadcast route discovery.

Node originating the MP-LPBR-RREP-RERR message	Sequence Number of MP-LPBR-RREP packet dropped	Source ID of the Data session	Destination ID of the Data session	Downstream Node with which the link failed
4 bytes	4 bytes	4 bytes	4 bytes	4 bytes

Fig. 7. MP-LPBR-RREP-RERR Message

III. SIMULATIONS

We study the performance of LPBR-M through extensive simulations and also compare its performance with that of the link-disjoint path based AOMDV [5] and the node-disjoint path based AODVM [6] routing protocols. We implemented all these three multi-path routing protocols in a discrete-event simulator developed in Java. Simulation results obtained from this simulator have also been successfully reported in our recent work [12][13] on MANET routing protocols.

We use a 1000m x 1000m square network. The transmission range per node is 250m. The number of nodes used in the network is 25, 50 and 75 nodes representing networks of low, medium and high density with an average distribution of 5, 10 and 15 neighbors per node respectively. For each combination of network density and node mobility, simulations are conducted with 15 source-destination (*s-d*) pairs. Traffic sources are constant bit rate (CBR). Data packets are 512 bytes in size and the packet sending rate is 4 data packets/second. Simulation time is 1000 seconds. The node mobility model used is the Random Waypoint model [14]. During every direction change, the velocity of a node is uniformly and randomly chosen from the range $[0,\ldots,v_{max}]$ and the values of v_{max} used are 10, 30 and 50 m/s, representing node mobility levels of low, moderate and high respectively. The Medium-Access Control (MAC) layer model used is the IEEE 802.11 model [15] involving Request-to-Send (RTS) and Clear-to-Send (CTS) message exchange for coordinating channel access. The transmission energy and reception energy per hop is set at 1.4 W and 1 W respectively [16]. Initial energy at each node is 1000 Joules.

The broadcast route discovery strategies simulated are the default flooding approach and the density and mobility aware energy-efficient broadcast strategy called DMEF [13]. We simulate DMEF as follows: During the on-demand route discovery process, each node dynamically chooses its own broadcast transmission range for the MP-RREQ message depending on the perceived number of neighbor nodes and the node's own mobility values during the time of broadcast. The broadcast transmission range at every node is however contained within the complete neighborhood defined by the default maximum transmission range of the node. A node surrounded by more neighbors advertises itself only to a limited set of nearby neighbors and a node surrounded by few neighbors will advertise itself to a maximum of its neighbors. Similarly, a slow-moving node advertises itself to a majority of its neighbors so that links formed using this node can be more stable. A fast-moving node advertises itself only to the neighbors closer to it. DMEF does not require any changes in the headers of the routing protocols and can be used with any MANET routing protocol. When we use DMEF, the periodic exchange of beacons in the neighborhood of each node occurs at a frequency determined from a time period uniformly and randomly selected from [0…5 seconds].

A. Performance Metrics

The performance metrics studied are the following:

- *Time between Successive Broadcast Multi-path Route Discoveries*: This is the time between two successive broadcast multi-path route discoveries, averaged over all the *s-d* sessions over the simulation time. We use a set of multi-paths as long as at least one path in the set exists, in the increasing order of their hop count. We opt for a broadcast route discovery when all the paths in a multi-path set fails. Hence, this metric is a measure of the lifetime of the set of multi-paths and a larger value is preferred for a routing protocol.
- *Average Energy Lost per Data Packet Delivered*: This is the sum of the energy consumed for transmission and reception at every hop, the energy consumed at the neighbors for coordination during channel access, the energy lost due to route discoveries and the energy lost due to periodic beaconing, if any, averaged over all the data packets delivered successfully at the destination.
- *Packet Delivery Ratio*: This is the ratio of the total number of data packets delivered to the destination to that of the total number of data packets originating from the source, averaged over all the *s-d* sessions. With a larger queue size (FIFO-based) of 200 at each node, the packet delivery ratio is a measure of network connectivity.
- *Energy Lost per Broadcast Multi-path Route Discovery*: This is the energy consumed per global broadcast based route discovery attempt, averaged over all the *s-d* sessions. This includes the energy consumed to transmit (broadcast) a MP-RREQ message to all the nodes in the neighborhood and the energy consumed to receive the MP-RREQ message sent by each node in the neighborhood, summed over all the nodes.
- *Control Message Overhead*: This is the ratio of the total number of control messages (MP-RREQ, MP-RREP, MP-LPBR-RREP and MP-LPBR-RREP-RERR) received at every node to that of the total number of data packets delivered at a destination, averaged over all the *s-d* sessions for the entire simulation time. In a typical broadcast operation, the total amount of energy spent to receive a control message at all the nodes in a neighborhood is greater than the amount of energy spent to transmit the message.
- *Average Energy Lost per Node*: The is the energy lost at a node due to transmission and reception of data packets, control packets and beacons, if any, averaged over all the nodes in the network for the entire simulation time.
- *Average Number of Disjoint Paths Found per Multi-path*: This is the number of disjoint-paths (link-disjoint or node-disjoint, depending on the protocol) determined during a multi-path broadcast route discovery, averaged over all the *s-d* sessions.
- *Average Number of Disjoint Paths used per Multi-path*: This is the number of disjoint-paths (link-disjoint or node-disjoint, depending on the protocol) actually used by the routing protocol, averaged over all the *s-d* sessions.
- *Average Hop Count of all Disjoint-paths used*: This is the time-averaged hop count of the disjoint paths determined and used by each of the multi-path routing protocols.

B. Time between Successive Broadcast Route Discoveries

The LPBR-M protocol yields the longest time between successive broadcast multi-path route discoveries (ref. Fig. 8). Thus, the set of node-disjoint paths discovered and predicted by LPBR-M are relatively more stable than the set of link-disjoint and node-disjoint paths discovered by the AOMDV and AODVM routing protocols respectively. Also, for each of the three multi-path routing protocols, the time between route discoveries when DMEF is used as the route discovery strategy is 4%-28%, 16%-38% and 28%-50% more than that incurred with flooding at low, moderate and high mobility levels respectively.

As we increase node mobility, the difference in the time between successive route discoveries incurred for AOMDV and AODVM vis-à-vis LPBR-M increases. Also, for a given level of node mobility, as we increase the network density, the time between successive route discoveries for LPBR-M increases relatively faster compared to those incurred for AOMDV and AODV-M. LPBR-M yields 3%-17% and 15%-44% more time between successive route discoveries compared to AOMDV and AODVM respectively.

C. Energy Lost per Data Packet Delivered

For a given level of node mobility and network density, the energy consumed per data packet (ref. Fig. 9) for each of three multi-path routing protocols is not very different from each other (the difference is within 3%). However, the energy consumed per data packet at a moderate network density of 50 nodes and a high network density of 75 nodes is respectively about 31%-44% and 75%-100% more than the energy consumed per data packet incurred in a low network density of 25 nodes. This can be attributed to the increase in the number of nodes receiving a broadcast message and transmitting the message in the network. Also, more neighbors are involved in the RTS and CTS message reception during co-ordination for channel access in every hop traversed by a data packet.

D. Packet Delivery Ratio

For a given level of node mobility and network density, the packet delivery ratio (ref. Fig. 10) of each of the multi-path routing protocols almost remained the same. In low-density networks, we observe 86% - 93% packet delivery ratio. Also, in low density networks, as the level of node mobility increases from low to moderate and high, the packet delivery

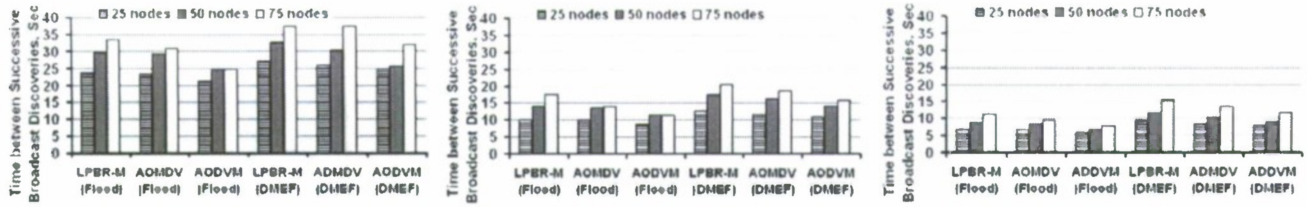


Fig. 8.1. $v_{max} = 10$ m/s Fig. 8.2. $v_{max} = 30$ m/s Fig. 8.3. $v_{max} = 50$ m/s

Fig. 8. Time between Successive Broadcast Multi-path Route Discoveries

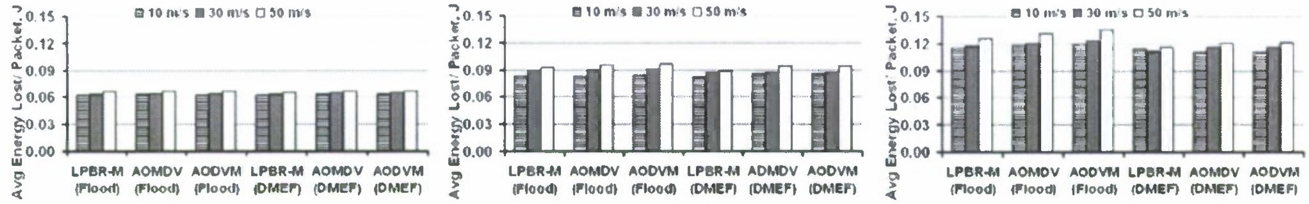


Fig. 9.1. 25 Nodes Fig. 9.2. 50 Nodes Fig. 9.3. 75 Nodes

Fig. 9. Average Energy Lost per Data Packet Delivered

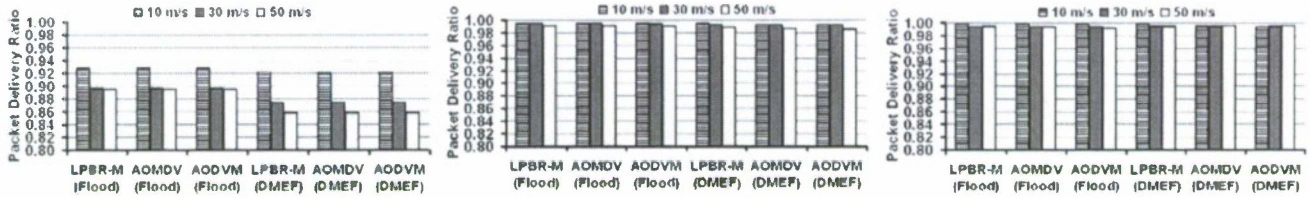


Fig. 10.1. 25 Nodes Fig. 10.2. 50 Nodes Fig. 10.3. 75 Nodes

Fig. 10. Packet Delivery Ratio of LPBR-M, AOMDV and AODVM under both Flooding and DMEF

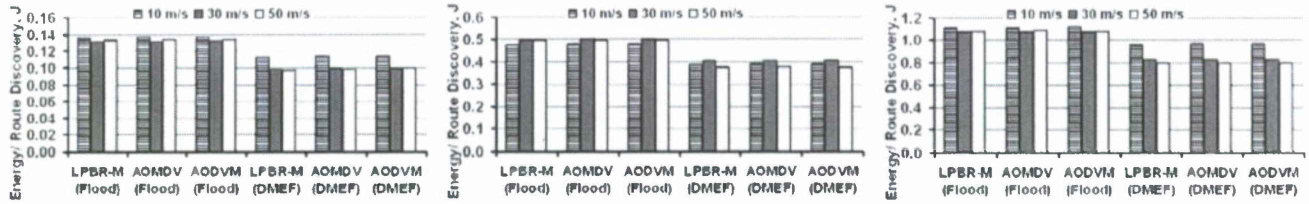


Fig. 11.1. 25 Nodes Fig. 11.2. 50 Nodes Fig. 11.3. 75 Nodes

Fig. 11. Average Energy Lost per Broadcast Route Discovery under both Flooding and DMEF

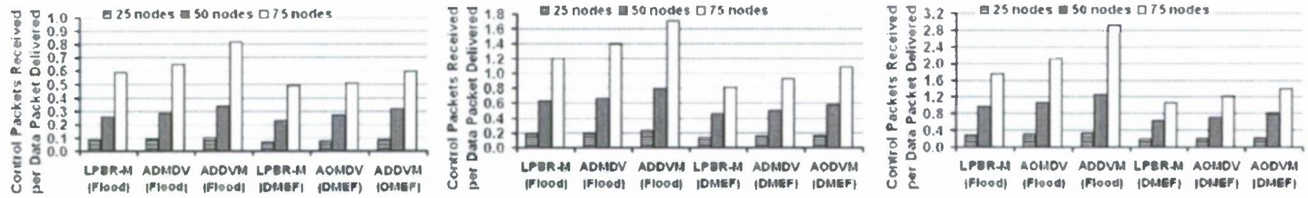


Fig. 12.1. 25 Nodes Fig. 12.2. 50 Nodes Fig. 12.3. 75 Nodes

Fig. 12. Control Message Overhead for LPBR-M, AMDV and AODVM under Flooding and DMEF

ratio decreases by about 4%-5%.

E. Energy Lost per Broadcast Multi-path Route Discovery

For a given level of node mobility and network density, the energy consumed per broadcast multi-path route discovery (ref. Fig. 11) for each of the three multi-path routing protocols is almost the same as this metric depends only on the route discovery strategy and not on the routing protocol. The energy consumed per route discovery in a moderate network density of 50 nodes and a high network density of 75 nodes is respectively about 3.4 to 4.1 times and 8.0 to 8.5 times more than the energy consumed per route discovery in a low density

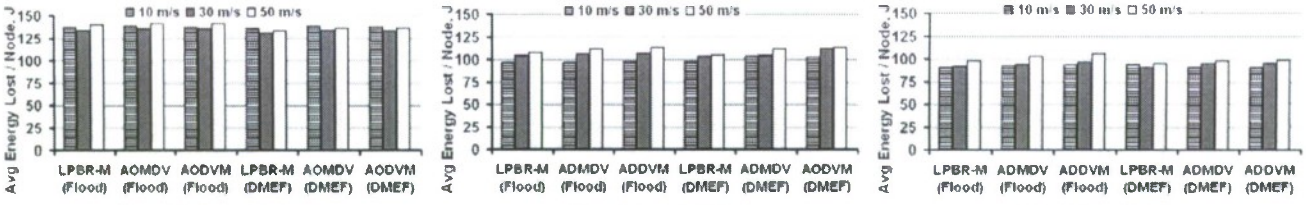


Fig. 13.1. 25 Nodes

Fig. 13.2. 50 Nodes

Fig. 13.3. 75 Nodes

Fig. 13. Average Energy Lost per Node

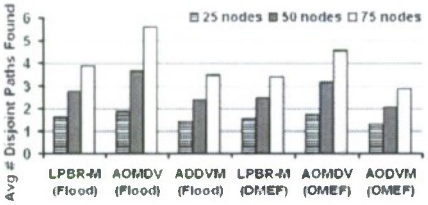


Fig. 14. Average Number of Disjoint Paths Found per Multi-path

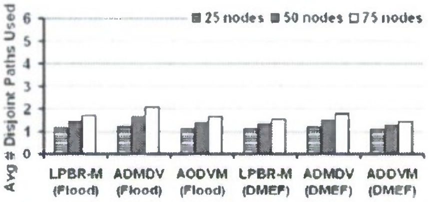


Fig. 15. Average Number of Disjoint Paths Used per Multi-path

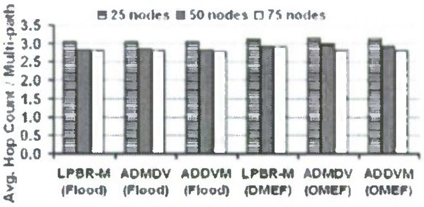


Fig. 16. Average Hop Count of All Disjoint Paths Used

network of 25 nodes. This can be attributed to the increase in the number of nodes receiving a broadcast message and transmitting the message in the network. With the DMEF strategy, we observe a decrease in the magnitude of energy consumed per route discovery at high network density and high node mobility. This can be attributed to the clever adaptation of the broadcast range by the DMEF strategy in such scenarios. In networks of low and moderate density, flooding consumes 19%-23% more energy per route discovery when compared to DMEF; whereas in high density networks, flooding consumes 32%-38% more energy per route discovery compared to DMEF.

F. Control Message Overhead

For a given node mobility and network density, LPBR-M incurs the lowest control message overhead (ref. Fig. 12). For a given node mobility, AOMDV and AODVM respectively incur 4%-16% and 14%-34% more control message overhead than LPBR-M when flooding is used. When DMEF is used as the route discovery strategy, AOMDV and AODVM respectively incur 10%-14% and 11%-23% more control message overhead than LPBR-M. In networks of moderate node mobility, the control message overhead incurred by the three multi-path routing protocols while using flooding and DMEF is respectively 2.1 (high density) to 3.4 (low density) times and 1.7 to 2.0 times more than that incurred in networks of low node mobility. In networks of high node mobility, the control message incurred by the three multi-path routing protocols while using flooding and DMEF is respectively 3.0 to 3.7 times and 2.2 to 2.8 times more than that incurred in networks of low node mobility. Thus, DMEF substantially reduces the control message overhead as we increase the network density and/or the level of node mobility.

G. Average Energy Lost per Node

We conduct all of our simulations with a fixed offered traffic load comprising of 15 *s-d* pairs. Hence, as we increase the network density, the net energy consumed per node (ref. Fig. 13) decreases as more nodes are available in the network for data transfer. For both flooding and DMEF, the energy lost per node in networks of moderate and high density is respectively about 65%-75% and 70%-84% of the energy lost per node in networks of low density. For a given network density, the energy lost per node at high node mobility is greater than the energy lost per node at low node mobility by at most 16% and 10% when operated with flooding and DMEF respectively.

H. Average Number of Node-Disjoint Paths Found and Used per Multi-path

For a given routing protocol and network density, the average number of disjoint paths discovered per multi-path (ref. Fig. 14) almost remains the same, irrespective of the level of node mobility. With increase in network density, the number of link-disjoint and node-disjoint paths between a source and destination increases. For a given network density and broadcast route discovery strategy, the link-disjoint path routing based AOMDV determines a larger number of disjoint paths (32%-62% more) than LPBR-M and AODVM; LPBR-M determines relatively larger number of disjoint paths (12%-22% more) than AODVM. For each of the three routing protocols, the average number of disjoint paths determined in a moderate density network and high-density network is respectively about 55%-95% and 120%-200% more than that determined in a low-density network. As DMEF reduces the control overhead and the number of nodes forwarding the MP-RREQ messages, the average number of disjoint paths determined for the three routing protocols is about 5% to 20% lower than that discovered using flooding.

Even though AOMDV had a relatively larger number of link-disjoint paths, the percentage of such paths successfully used is the lowest among the three multi-path routing protocols. The node-disjoint path based AODVM routing protocol has the largest percentage of the discovered disjoint paths actually being used. As the network density increases, the number of disjoint paths actually used by each of the three multi-path routing protocols (ref. Fig. 15) increases, nevertheless at a significantly reduced rate. As a result, the percentage of the discovered disjoint paths successfully used decreases with increase in network density. This can be attributed to the failure of the disjoint paths over time and the disjoint-paths discovered are not actually available when the routing protocol wants to use them.

I. Average Hop Count per Multi-path

For a given routing protocol and network density, the average hop count (ref. Fig. 16) of the disjoint-paths used is almost the same, irrespective of the level of node mobility. As we add more nodes in the network, the hop count of the paths tends to decrease as the source manages to reach the destination through relatively lesser number of intermediate nodes. With increase in network density, there are several candidates to act as intermediate nodes on a path. The average hop count of the paths in high and moderate density networks is 6%-10% less than the average hop count of the paths in networks of low density. For each of the routing protocols, for all network densities, the average hop count of the paths discovered using DMEF is at most 4% more than the hop count of the paths determined using flooding.

IV. CONCLUSIONS

The high-level contribution of this paper is the design and development of a node-disjoint multi-path extension for the Location Prediction Based Routing protocol (referred to as LPBR-M). LPBR-M reduces the number of global broadcast multi-path route discoveries. Simulations have been conducted with both flooding and DMEF as the broadcast route discovery strategies. We compared the performance of LPBR-M with that of the link-disjoint path based AOMDV and the node-disjoint path based AODVM multi-path routing protocols. LPBR-M achieves the longest time between successive route discoveries and the lowest control message overhead. Also, the LPBR-M multi-paths incur hop count that is very much equal to those obtained with the minimum-hop based AODVM and AODVM routing protocols. Moreover, DMEF helps each of the multi-path routing protocols to determine a set of node or link disjoint paths that exist for a long time and at the same time does not increase the source-destination hop count appreciably. When used with DMEF, each of the multi-path routing protocols incurred a lower energy spent per route discovery, compared to flooding.

ACKNOWLEDGMENT

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-08-2-0061. The views and conclusions in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

REFERENCES

[1] S. Mueller, R. P. Tsang and D. Ghosal, "Multi-path Routing in Mobile Ad Hoc Networks: Issues and Challenges," *Lecture Notes in Computer Science*, Vol. 2965, pp. 209 – 234, April 2004.

[2] D. B. Johnson, D. A. Maltz and J. Broch, "DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Ad hoc Networks," in *Ad hoc Networking*, Chapter 5, C. E. Perkins, Eds. Addison Wesley, pp. 139 – 172, 2000.

[3] C. E. Perkins and E. M. Royer, "Ad hoc On-demand Distance Vector Routing," *Proceedings of the 2nd Annual IEEE International Workshop on Mobile Computing Systems and Applications*, pp. 90 – 100, February 1999.

[4] S. Lee and M. Gerla, "Split Multi-path Routing with Maximally Disjoint Paths in Ad Hoc Networks," *Proceedings of IEEE International Conference on Communications*, Vol. 10, pp. 3201-3205, 2001.

[5] M. K. Marina and S. R. Das, "On-demand Multi-path Distance Vector Routing in Ad Hoc Networks," *Proceedings of the IEEE International Conference on Network Protocols*, pp. 14 – 23, Nov. 2001.

[6] Z. Ye, S. V. Krishnamurthy and S. K. Tripathi, "A Framework for Reliable Routing in Mobile Ad Hoc Networks," *Proceedings of the IEEE Internotionol Conference on Computer Communicotions*, 2003.

[7] V. Loscri and S. Marano, "A New Geographic Multi-path Protocol for Ad Hoc Networks to Reduce the Route Coupling Phenomenon," *Proceedings of the 63rd IEEE Vehicular Technology Conference*, VTC 2006-Spring, Vol. 3, pp. 1102-1106, 2006.

[8] M. Li, L. Zhang, V. O. K. Li, X. Shan and Y. Ren, "An Energy-Aware Multi-path Routing Protocol for Mobile Ad Hoc Networks," *Proceedings of the ACM SIGCOMM Asia*, April 2005.

[9] N. Meghanathan, "A Location Prediction-Based Reactive Routing Protocol to Minimize the Number of Route Discoveries and Hop Count per Path in Mobile Ad Hoc Networks," *The Computer Journal*, vol. 52, no. 4, pp. 461-482, July 2009.

[10] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms," 2nd Edition, MIT Press/ McGraw Hill, Sept. 2001.

[11] N. Meghanathan, "Stability and Hop Count of Node-Disjoint and Link-Disjoint Multi-path Routes in Ad Hoc Networks," *Proceedings of the 3rd IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, New York, October 2007.

[12] N.Meghanathan, "Multicast Extensions to the Location-Prediction Based Routing Protocol for Mobile Ad hoc Networks," accepted for publication in the *Internotionol Conference on Wireless Algorithms, Systems and Applications*, Boston, USA, August 16-18, 2009.

[13] N. Meghanathan, "A Density and Mobility Aware Energy-Efficient Broadcast Strategy to Minimize the Number of Route Discoveries in Mobile Ad hoc Networks," accepted for publication in the *Proceedings of the 2009 Internotionol Conference on Wireless Networks, ICWN 09, Las Vegos*, July 13-16, 2009.

[14] C. Bettstetter, H. Hartenstein and X. Perez-Costa, "Stochastic Properties of the Random-Waypoint Mobility Model," *Wireless Networks*, vol. 10, no. 5, pp. 555-567, 2004.

[15] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal of Selected Areas in Communication*, vol. 18, no. 3, pp. 535-547, 2000.

[16] L. M. Feeney, "An Energy Consumption Model for Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks," *Journal of Mobile Networks ond Applications*, vol. 3, no. 6, pp. 239 – 249, June 2001.