A Performance Comparison Study of the Location Prediction Based Routing Protocol with Position Based Routing Protocols for Mobile Ad Hoc Networks

Natarajan Meghanathan

Department of Computer Science, Jackson State University, 1400 John Lynch St, Jackson, MS 39217, USA
natarajan.meghanathan@jsums.edu

ABSTRACT

In a recent work, we published the Location Prediction Based Routing (LPBR) protocol for mobile ad hoc networks (MANET). LPBR is the first such beaconless MANET protocol to simultaneously minimize the hop count of the paths as well as the routing control overhead measured in terms of the number of control messages received during an on-demand broadcast discovery. LPBR works as follows: If the minimum hop route discovered through a regular broadcast route discovery fails, instead of the source immediately launching another broadcast route discovery, the destination attempts to locally predict the global network topology based on the location and mobility information of the nodes learnt during the most recent broadcast route discovery. If the predicted path does exist in reality, the source learns the path from the destination and continues to send data packets without launching a new broadcast route discovery. The performance of LPBR has been so far studied mainly with the topology-based routing protocols that initiate on-demand route discoveries. In this paper, we compare the performance of LPBR with position-based routing protocols in which the forwarding decisions are taken independently for each data packet at every forwarding node, depending on the estimated location of the destination. Through extensive simulations, we illustrate that LPBR performs significantly better compared to the well-known position-based routing protocols and their variants with respect to several performance metrics under diverse conditions of node mobility, network density and offered traffic load.

KEYWORDS

Location Prediction, Position-based Routing, Simulation, Performance, Mobile Ad hoc Networks

1. INTRODUCTION

A mobile ad hoc network (MANET) is a dynamic distributed system of wireless nodes that move independently of each other. The operating transmission range of the nodes is limited and as a result, MANET routes are often multi-hop in nature. Any node in a MANET can become a source or destination, and each node can function as a router, forwarding data for its peers. MANET routing protocols are either proactive or reactive in nature. Proactive routing protocols determine and maintain routes between any pair of nodes irrespective of their requirement. The reactive on-demand routing protocols determine a route only when required. As the network topology changes dynamically, reactive on-demand routing has been preferred over proactive routing [4][9].

Position-based routing protocols do not conduct on-demand route discovery to learn and maintain routes. Instead, forwarding decisions are taken independently for each data packet at every forwarding node (including the source) depending on the position of the forwarding node,

the intermediate nodes and the destination. Normally, the source includes its estimated location information of the destination in every data packet. The position-based routing protocols are mostly designed to choose the intermediate forwarding nodes that lie on the shortest path or close to the shortest path from the source to the destination. Greedy Perimeter Stateless Routing (GPSR) [11], Ellipsoid Routing [18] and Geographical Routing Protocol based on Prediction (GRPP) [6] are some examples of position-based routing protocols. Each node is assumed to know the locations of its neighbors through periodic beacon exchange. The effectiveness of the position-based routing protocols depends on the accuracy of the destination location information included in the header of the data packets, method adopted for disseminating location information and the method adopted to learn the latest location information of the destination node. In this research work, we assume the source uses the Hierarchical Location Service (HLS) [12], a robust and scalable location service scheme proposed for MANETs. The source node queries the responsible cells (in a region of the network) of the HLS at a certain time period uniformly distributed in the range $[0… MaxT_{update}]$ where $MaxT_{update}$ is the maximum time period between successive destination location update searches. The source node includes the recently learnt location co-ordinates of the destination in the header of the data packets. The shorter the time between consecutive location update searches by the source node, the more accurate will be the location prediction and shorter will be the hop count. But, this advantage comes at the cost of a higher cellcast (broadcast within the responsible cells) control message overhead in frequently querying the HLS about the latest location of the destination. Due to the limited queue size at the nodes, a higher control message overhead to query HLS can also result in dropping of the data packets. More information on HLS is given in Section 2.

In [13], we proposed a new mobile ad hoc network routing protocol called "Location Prediction Based Routing" (LPBR) that is aimed to simultaneously minimize the routing control overhead as well as the number of hops in the paths used for a source-destination session. With the latest advances in development of small inexpensive, low power GPS (Global Positioning System) receivers [8], it is possible for a node to locate where it is in an ad hoc network. LPBR works as follows: Whenever a source node has data packets to send to a destination node and does not have a route to the destination node, the source node initiates a global flooding-based route discovery by broadcasting a Route-Request (RREQ) packet. During this flooding process, each node forwards the RREQ packet exactly once after incorporating its location update vector (LUV) in the RREQ packet. The LUV of a node comprises of the node id, the current X and Y co-ordinates of the node, the current velocity and angle of movement with respect to the X-axis. We assume all the nodes in the network are position-aware (using techniques like GPS [8]) and the clocks across all nodes are synchronized. The destination node collects the LUV information of all the nodes in the network from the RREQ packets received through several paths. The destination node sends a Route-Reply (RREP) packet to the source on the minimum hop path traversed by a RREQ packet. The source node then sends the data packets on the path learnt through the RREP packet. If any more data packet is anticipated to be sent, the source node informs the destination node about the time of next packet dispatch through the header of the data packet currently being sent.

If an intermediate node could not forward a data packet, it sends a Route-Error packet to the source node. The source node then waits a little while for the destination node to find a route based on the LUVs gathered from the latest flooding-based route discovery. If the destination does not receive the data packet within the expected time, it locally constructs the global topology by predicting the locations of the nodes learnt from the LUVs gathered during the latest flooding-based route discovery. The destination predicts the location of a node by assuming that the node will continue to move in the same direction and speed as mentioned in its latest LUV. If at least one path exists in the predicted global topology, the destination node sends a LPBR-RREP packet on the minimum hop path [5] in the predicted topology. If the

predicted path actually exists in reality, the intermediate nodes on the predicted route manage to forward the LPBR-RREP packet to the source. The source uses the route informed through the latest LPBR-RREP packet to send the data packets. Thus, a resource-consuming global flooding-based route discovery is avoided. If an intermediate node could not forward the LPBR-RREP packet (i.e., the predicted path does not exist in reality), the intermediate node sends back a LPBR-RREP-Error packet to the destination node informing it of the failure to forward the LPBR-RREP packet. The destination node discards all the LUVs learnt and waits for the source to initiate another global flooding-based route discovery. The source node initiates the next flooding-based route discovery after timing out waiting for the LPBR-RREP packet from the destination. Figure 1 presents a comprehensive illustration of the working of the LPBR protocol.
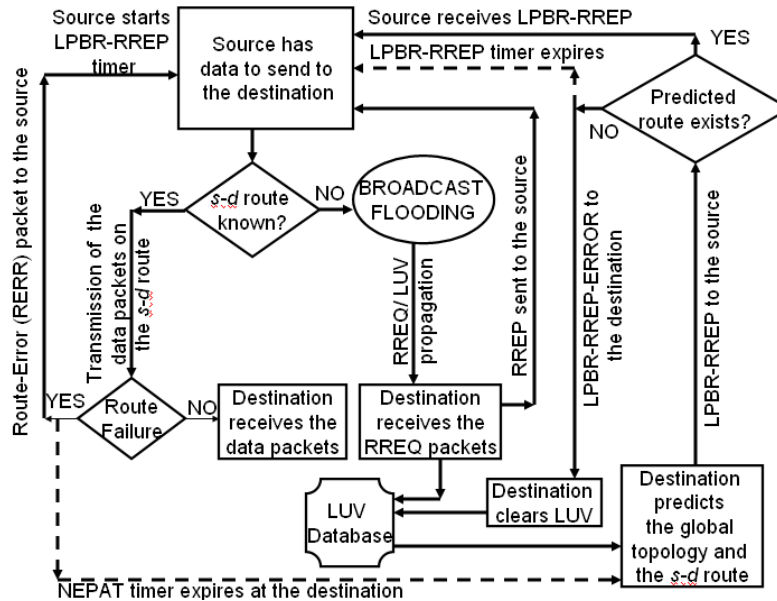

Figure 1: Comprehensive Illustration of the Working of the LPBR Protocol

The effectiveness of LPBR compared to contemporary topology-based routing protocols such as Dynamic Source Routing (DSR) protocol [10] and well known stability-based protocols such as Associativity-based Routing (ABR) protocol [17], Flow-Oriented Routing Protocol (FORP) [16] and the Route-lifetime Assessment based Routing (RABR) protocol [1] has been clearly demonstrated in [13]. In this paper, we demonstrate the effectiveness of LPBR compared to position-based routing protocols. Through extensive simulations, we show that LPBR significantly reduces the number of global route discoveries and incurs a route discovery control overhead significantly lower to that of the position-based protocols. Even though, there is some overhead in including the LUV information in the RREQ packets and the succeeding packet sending time information in the data packets, we find this to be a small and useful overhead compared to the huge route discovery control overhead incurred in the other contemporary topology-based routing protocols and the destination location search overhead incurred in the position-based routing protocols. LPBR minimizes the number of route discoveries as it tries to use the LUV information gathered during the latest route discovery process and locally determines a new route upon a route failure. It does not opt for flooding-based route discovery for every route failure. Also, LPBR always opts for the minimum hop paths in the predicted global topology, and hence the average hop count per path for LPBR routes is significantly smaller. Thus, LPBR simultaneously reduces the number of route discoveries (and hence the route discovery overhead) as well as incurs lower hop count per path. None of the MANET

routing protocols proposed so far in the literature can simultaneously minimize the hop count as well as the route discovery control overhead and also without the use of periodic beacon exchange among the nodes in a neighborhood.

The rest of the paper is organized as follows: In Section 2, we provide a brief overview of the position-based GPSR and GRPP routing protocols and their improved versions. We also discuss the Hierarchical Location Service (HLS) scheme used in the position-based routing protocols. Section 3 presents the simulation conditions used to compare the performance of LPBR with that of the position-based GPSR and GRPP routing protocols and their improved versions. In Section 4, we explain the simulation results observed with the different routing protocols and highlight the reduction in the control overhead, number of route discoveries, hop count per path and end-to-end delay per data packet achieved with LPBR. Section 5 draws the conclusions.

2. POSITION-BASED ROUTING PROTOCOLS AND HIERARCHICAL LOCATION SERVICE – A REVIEW

Position-based routing protocols do not go through a network-wide route discovery process, but attempt to forward the data packets from a source to the destination using the position information of the destination included in the data packet headers and the knowledge of the nodes about the positions of other nodes in their local neighborhood. Examples of position-based routing protocols include the Greedy Perimeter Stateless Routing (GPSR) protocol [11] and the Geographical Routing Protocol based on Prediction (GRPP) [6].

2.1 Greedy Stateless Perimeter Routing (GPSR)

Greedy Perimeter Stateless Routing (GPSR) [11] is a position-based ad hoc routing protocol in which there is no flooding-based route discovery to determine source-destination routes. The source periodically uses a location service scheme (like HLS) to learn about the latest location information of the destination and includes it in the header of every data packet. If the destination is not directly reachable, the source node forwards the data packet to the neighbor node that lies closest to the destination. Such a greedy procedure of forwarding the data packets is also repeated at the intermediate nodes. In case, a forwarding node could not find a neighbor that lies closer to the destination than itself, the node switches to perimeter forwarding. With perimeter forwarding, the data packet is forwarded to the first neighbor node that is come across, when the line connecting the forwarding node and the destination of the data packet is rotated in the anti-clockwise direction. The location of the forwarding node in which greedy forwarding failed (and perimeter forwarding began to be used) is recorded in the data packet. We switch back to greedy forwarding when the data packet reaches a forwarding node which can find a neighbor node that is away from the destination node by a distance smaller than the distance between the destination node and the node at which perimeter forwarding began. GPSR requires each node periodically (for every one second, in this paper) broadcast a beacon containing its latest location information to its neighbors.

2.2 Improvements to Greedy and Perimeter Forwarding

The perimeter forwarding approach of GPSR has been observed to generate wasteful loops when the destination node moves away from the location co-ordinates included in the header of the data packets [15]. To counter the looping problem and to increase the packet delivery ratio when the destination node moves out of its original location, a destination node-location prediction (DNP) approach has been proposed in [15]. According to DNP, each node, before forwarding a data packet based on the location information of the destination in the packet header, searches its neighbor list for the destination node. If the destination node is in the neighbor list, then the data packet is directly forwarded to the destination node. A further

advanced improvement to position-based greedy forwarding called Advanced Greedy Forwarding (AGF) has been proposed in [14]. According to AGF, each node manages to collect the list of nodes in its two-hop neighborhood through the exchange of neighbor lists during periodic beacon broadcast in the one-hop neighborhood. In AGF, each node, before forwarding a data packet based on the location information of the destination in the data packet header, searches for the destination in its one-hop and two-hop neighbor lists. If the destination is in the one-hop neighbor list, the data packet is directly forwarded to the destination (in this case AGF reverts to DNP). If the destination is only in the two-hop neighbor list and not in the one-hop neighbor list, the data packet is forwarded to the neighbor node (of the destination) in the one-hop neighbor list.

2.3 Geographical Routing Protocol based on Prediction (GRPP)

The Geographical Routing Protocol based on Prediction (GRPP) [6] is a novel approach of deciding the next hop node at a forwarding node, based on the current and future positions of the forwarding node and its neighboring nodes with respect to the ultimate destination node of the data packet. Based on its own movement and the periodic beacons received from its neighbors, each node learns the location of itself and its neighbors at the current time instant t (say, in seconds) and predicts the location of itself and its neighbors for the next 3 seconds (i.e., at time instants $t+1$, $t+2$ and $t+3$). Let I be the intermediate node from which a data packet needs to be forwarded so that it can reach its ultimate destination D. For every neighbor node N of I, I computes the distances $d_{IN}^t, d_{ND}^t, d_{IN}^{t+1}, d_{ND}^{t+1}, d_{IN}^{t+2}, d_{ND}^{t+2}, d_{IN}^{t+3}, d_{ND}^{t+3}$ between itself and N and between N and D for the current time instant t and for each of the next three seconds. The location of the destination D is assumed to be fixed for the current time instant and for each of the next three seconds. Only the location of the forwarding node I and its neighbors are predicted. The forwarding node I chooses the next hop as the neighbor node N such that the sum $d_{IN}^t + d_{ND}^t + d_{IN}^{t+1} + d_{ND}^{t+1} + d_{IN}^{t+2} + d_{ND}^{t+2} + d_{IN}^{t+3} + d_{ND}^{t+3}$ is minimized. GRPP is an improvement over the Ellipsoid algorithm [18] that considers only minimizing the sum $d_{IN}^t + d_{ND}^t$ while choosing the next hop node at the intermediate forwarding node I. Both GRPP and the Ellipsoid protocols aim to select the next hop node as the node that lies closer to the straight line joining the forwarding node and the destination. By also considering the predicted locations of the neighbor nodes to determine the next hop, GRPP selects stable links (links having positive increase in the signal strength between consecutively transmitted packets) in the presence of node mobility. Like the original version of GPSR, GRPP also relies on the co-ordinates of the destination location information in the data packet header to apply the above described procedure to determine the next hop.

In this paper, we study the original versions of GPSR and GRPP and also apply DNP and AGF to improve the performance of both GPSR and GRPP. The improved versions are referred to as GPSR_DNP, GPSR_AGF, GRPP_DNP and GRPP_AGF.

2.4 Hierarchical Location Service (HLS)

We use the Hierarchical Location Service (HLS) [12] approach to let the source discover the location of the destination. According to HLS, the entire network area is divided into a hierarchy of regions. The topmost region is the whole network. A region at a particular level in the hierarchy is sub-divided into several regions of the next lower level in the hierarchy until the lowest level (called a cell) is reached. The only requirement to characterize a region as cell is that a node should be able to send packets to all other nodes in the same cell via direct transmission or cell-wide broadcast called *cellcast*. A given node I uses a hash function to choose one specific cell (called the responsible cell) in each level of the hierarchy to maintain its latest position and mobility information. A commonly used hash function (also used in this

paper) by node *I* to select the responsible cell at level *n* containing *I* is H(*I*, *n*) = ID(*I*) mod ‖S(*I*, *n*)‖, where ‖S(*I*, *n*)‖ is the number of cells at level *n* containing node *I*. The hash value is used to select the responsible cell from the set S(*I*, *n*). A node *J* requiring the position of node *I* has to use the same hash function to determine the information about the responsible cells that may hold the latest information about *I*. Node *J* then iteratively queries through cell-wide broadcast (cellcast) among the nodes in the chosen responsible cells in the order of the hierarchy (from the smallest level to the entire network) until a reply is received about the current position of node *I*.

As a node moves from one cell to another cell, it either (i) updates all its responsible cells in the network, called the "direct" update strategy or (ii) updates the responsible cells at the higher levels only when the responsible cell on the next lower level changes, called the "indirect" strategy. The worst case message complexity of both the update strategies is O($\sqrt{n}$) where *n* is the number of nodes in the entire network. Nevertheless, in practice, as the majority of location update packets need to travel only a few hops, the indirect location update strategy incurs a significantly lower message transmission overhead compared to the direct update strategy. We hence adopt the indirect location update strategy in this paper.

3. SIMULATION ENVIRONMENT AND MODELS

We use ns-2 (version 2.28) [7] as the simulator for our study. We implemented the LPBR, GPSR, GRPP protocols and their improved versions GPSR_DNP, GPSR_AGF, GRPP_DNP and GRPP_AGF in ns-2. 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 is 25 and 50 nodes representing networks of low and high density respectively. Initially, nodes are uniformly randomly distributed in the network. The maximum time period (*MaxT_{update}*) between successive location update queries sent by a source node to the Hierarchical Location Service for the position-based routing protocols is varied from 20 seconds to 200 seconds. For a given value of *MaxT_{update}*, the time period between two successive queries launched by the source node towards the responsible cells (of the HLS) for the latest information about the destination location is uniformly distributed within the range [0… *MaxT_{update}*]. The simulation conditions are listed in Table 1 and explained below.

Table 1: Simulation Conditions

Network Size	1000m x 1000m	
Number of nodes	25 (low density) and 50 (high density)	
Physical Layer	Signal Propagation Model	Two-ray ground reflection model [4]
MAC Layer	IEEE 802.11 [3]	
	Link Bandwidth	2 Mbps
	Interface Queue	FIFO-based, size 100
Routing Protocols	LPBR [13], GPSR [11], GRPP [6], GPSR_DNP, GPSR_AGF, GRPP_DNP and GRPP_AGF	
Mobility Model	Random Way Point Model [2]	
	Minimum Node Speed, m/s	0 m/s
	Maximum Node Speed, m/s	Low-5, 10; Medium-20, 30; High-40, 50
	Pause Time	0 second
Traffic Model	Constant Bit Rate (CBR), UDP	
	# Source-Destination (*s-d*) Sessions	15 (low traffic) and 30 (high traffic)
	Data Packet Size	512 bytes

	Packet Sending Rate per *s-d* Pair	4 Packets/ second
Maximum Location Update Search Time Interval	$MaxT_{update}$ (seconds)	20, 50, 80, 120, 150 and 200

3.1 Traffic Model

Traffic sources are constant bit rate (CBR). The number of source-destination (*s-d*) sessions used is 15 (indicating low traffic load) and 30 (indicating high traffic load). The starting timings of these *s-d* sessions are uniformly distributed between 1 to 50 seconds. The sessions continue until the end of the simulation time, which is 1000 seconds. Data packets are 512 bytes in size and the packet sending rate is 4 data packets/second. For each node, we made sure that the node does not end up a source for more than two sessions and/ or not as a destination for more than two sessions.

3.2 Node Mobility Model

The node mobility model used in all of our simulations is the Random Waypoint model [2], a widely used mobility model in MANET simulation studies. According to this model, each node starts moving from an arbitrary location to a randomly selected destination location at a speed uniformly distributed in the range [v_{min},…,v_{max}]. Once the destination is reached, the node may stop there for a certain time called the pause time and then continue to move by choosing a different target location and a different velocity. In this paper, we set $v_{min} = 0$, and each node chooses speed uniformly distributed between 0 and v_{max}. The v_{max} values used are 5 and 10 m/s (representing low node mobility), 20 and 30 m/s (representing moderate node mobility), 40 and 50 m/s (representing high node mobility). Pause time is 0 seconds.

3.3 Performance Metrics

We measure the following performance metrics for the routing protocols under each of the simulation conditions listed in Table 1:

- *Control Messages Overhead*: It is the sum of the route discovery control messages (like RREQ, RREP, Route-Error, LPBR-RREP, and LPBR-RREP-Error messages in the case of LPBR) or the destination location update search cellcast messages (in the case of GPSR, GRPP, GPSR_DNP, GPSR_AGF, GRPP_DNP and GRPP_AGF) received by the nodes in the network, computed over all the *s-d* sessions of a simulation run. Note that most of the control messages for route discovery or destination location search are broadcast in nature. The sum of the energy lost at all the receivers of a broadcast message is far greater than the energy lost by the transmitter of the broadcast message. Hence, we measure the control message overhead as the sum of the number of control messages received at all the nodes in the network across all the *s-d* sessions of a simulation run.
- *Hop count per path:* It is the average hop count per path, time-averaged over all the *s-d* sessions. For example, if we have been using two paths P_1 of hop count 3 and P_2 of hop count 5 for time 10 and 20 seconds respectively, then the time-averaged hop count of P_1 and P_2 is (3*10 + 5*20)/30 = 4.33.
- *End-to-end delay per packet*: It 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*: It 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.

-

4. SIMULATION RESULTS

Each data point in Figures 2 through 18 is an average of data collected using 5 mobility trace files and 5 sets of randomly selected 15 or 30 *s-d* sessions, depending on the simulation condition.

4.1 Control Message Overhead

For the topology-based routing protocols including LPBR, we measure the control message overhead as the number of route discovery control messages received by the nodes in the network. Figures 2 and 3 illustrate the routing control message overhead incurred with LPBR and Figure 4 illustrates the corresponding number of global flooding-based route discoveries incurred. For a given network density and offered traffic load, the larger the number of route discoveries, the larger will be the routing control message overhead. For the position-based routing protocols, we measure the control message overhead as the number of destination location update search messages received by the nodes in the network. Note that for both the classes of routing protocols, we measure the control message overhead as the number of control messages received by the nodes in the network, rather than the number of control messages transmitted. This is because the control traffic is broadcast in nature. For example, if a node has 10 neighbours and it broadcasts a packet to its neighbourhood, then there is just one transmission, but there are 10 receptions. All the neighbours of the node lose energy to receive the packet. Broadcast transmissions of control packets are not preceded by MAC layer (Request-To-Send – Clear-To-Send) RTS-CTS mechanisms [3] to reserve the medium. So a receiving node has no option other than to receive the entire control packet and then dump it if it is redundant or not useful.
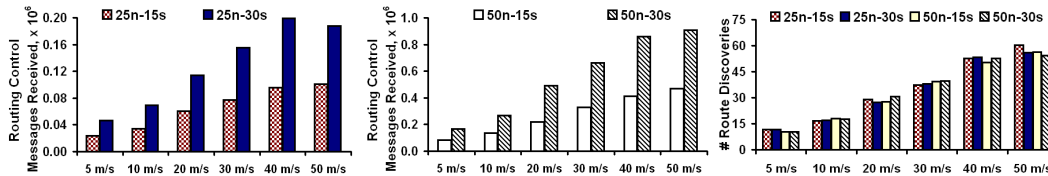
Figure 2: LPBR - Routing Overhead (25 Node Network) **Figure 3:** LPBR - Routing Overhead (50 Node Network) **Figure 4:** LPBR - Broadcast Route Discoveries

The route discovery control messages considered for performance measurement are the RREQ, RREP, Route-Error, LPBR-RREP and LPBR-RREP-Error messages for LPBR and the destination location search cellcast messages in the responsible cells in the case of GPSR, GRPP, GPSR_DNP, GPSR_AGF, GRPP_DNP and GRPP_AGF. The control message overhead incurred by the position-based routing protocols (illustrated in Figures 5 and 6) depends on the value of $MaxT_{update}$ and the number of *s-d* pairs. For a given number of *s-d* pairs, the lower the value of $MaxT_{update}$, the larger the control message overhead and more accurate is the destination location information included by the source in the header of the data packets. Similarly, for a given value of $MaxT_{update}$, the larger the number of *s-d* pairs, more source nodes would initiate destination location searches and larger is the control message overhead.

In a network of 25 nodes, for a given offered traffic load, the routing control message overhead incurred by LPBR at low (v_{max} = 5 m/s), moderate (v_{max} = 30 m/s) and high (v_{max} = 50 m/s) node mobility is respectively about 10%, 34% and 42% of the control message overhead incurred when the position-based routing protocols are operated with a $MaxT_{update}$ value of 20 seconds. On the other hand, the routing control message overhead incurred by LPBR at low, moderate

and high node mobility is respectively about $1.1 - 1.2$, $3.8 - 4.2$, and $4.6 - 5$ times the control message overhead incurred when the position-based routing protocols are operated with a $MaxT_{update}$ value of 200 seconds. We observe a tradeoff between the control message overhead and the percentage of packets delivered. For a given offered data traffic load, the smaller the $MaxT_{update}$ value, the larger will be the percentage of packets delivered with the position-based routing protocols and vice-versa (refer Figures 15 through 18 for packet delivery ratio).
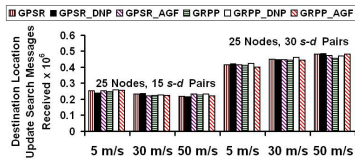
Figure 5.1: $MaxT_{upd} = 20$ sec **Figure 5.2:** $MaxT_{upd} = 50$ sec **Figure 5.3:** $MaxT_{upd} = 80$ sec
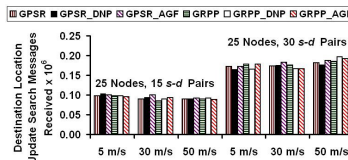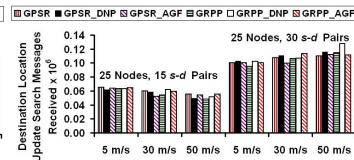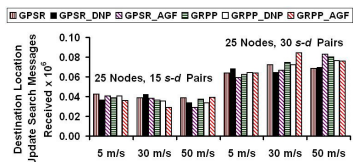
Figure 5.4: $MaxT_{upd} = 120$ sec **Figure 5.5:** $MaxT_{upd} = 150$ sec **Figure 5.6:** $MaxT_{upd} = 200$ sec
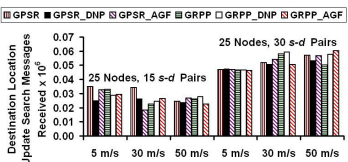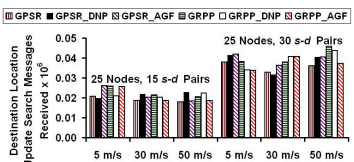
Figure 5: Destination Location Update Search Messages Received at all Nodes (25 Nodes)
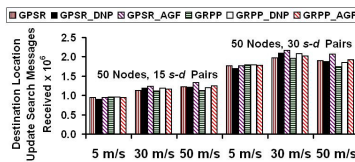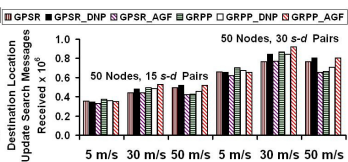

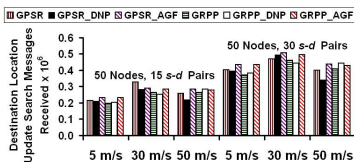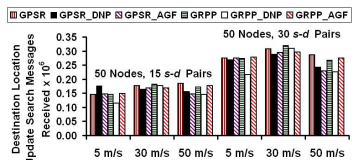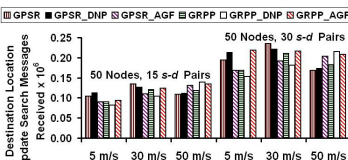**Figure 6.1:** $MaxT_{upd} = 20$ sec **Figure 6.2:** $MaxT_{upd} = 50$ sec **Figure 6.3:** $MaxT_{upd} = 80$ sec

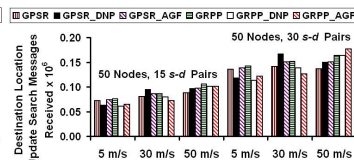**Figure 6.4:** $MaxT_{upd} = 120$ sec **Figure 6.5:** $MaxT_{upd} = 150$ sec **Figure 6.6:** $MaxT_{upd} = 200$ sec

Figure 6: Destination Location Update Search Messages Received at all Nodes (50 Nodes)

In a network of 50 nodes, for a given offered traffic load, the routing control message overhead incurred by LPBR at low ($v_{max} = 5$ m/s), moderate ($v_{max} = 30$ m/s) and high ($v_{max} = 50$ m/s) node mobility is respectively about 9%, 30% and 44% of the control message overhead incurred when the position-based routing protocols are operated with a $MaxT_{update}$ value of 20 seconds. On the other hand, the routing control message overhead incurred by LPBR at low, moderate and high node mobility is respectively about $1.2 - 1.3$, $3.9 - 4.5$ and $4.8 - 5.8$ times the control message overhead incurred when the position-based routing protocols are operated with a $MaxT_{update}$ value of 200 seconds.

Note that we do not take into account the periodic beacon exchange overhead incurred in protocols like GRPP, GPSR, GRPP_DNP, GRPP_AGF, GPSR_DNP and GPSR_AGF while

measuring the routing control message overhead. These routing protocols require periodic beacon exchange among neighbors in order to learn about the positions and mobility of the neighbor nodes. In this paper, we have set the beacon exchange interval to be 1 second. The reason for the omission is that we want to only consider control messages whose scope for transmission and reception is larger, i.e., either within the cells of a HLS region or network-wide. Though we do not measure the number of beacons received at the nodes, we do take into consideration the presence of beacons and other control messages in the queues of the nodes and thus consider their impact on the packet delivery ratio and end-to-end delay per data packet. LPBR does not require periodic beacon exchange among the nodes in a neighborhood.

In the case of GRPP, GPSR and their improved versions, after each destination location update search, each source node, independently, uniformly and randomly selects a waiting time value from the range [0… $MaxT_{upd}$ seconds] and initiates the next destination location search process. By doing so, we avoid the situation of having all nodes simultaneously and periodically broadcasting (cellcasting) their location query message updates, which would trigger congestion at the nodes.

4.2 Hop Count per Path

In Figures 7 through 10, we observe that the hop count incurred by LPBR routes is almost close to that incurred by GPSR routes and is larger than that of the GPSR_DNP and GPSR_AGF routes, for certain simulation conditions only, by at most 10-20%. On the other hand, the hop count of LPBR routes is significantly smaller compared to that incurred by GRPP. This indicates the effectiveness of the route prediction approach adopted in LPBR. The hop count of the routes predicted upon a route failure is close to being the minimum in the network at that instant of time.

As the routing protocols simulated in this paper do not take the queue size into consideration while determining the routes, the hop count of the routes for each protocol is independent of the offered data traffic load. Similarly, the hop count of the routes chosen by LPBR is almost independent of the maximum velocity of the nodes. On the other hand, the hop count of the routes chosen by GPSR and GRPP are somewhat influenced by the dynamics of node mobility.
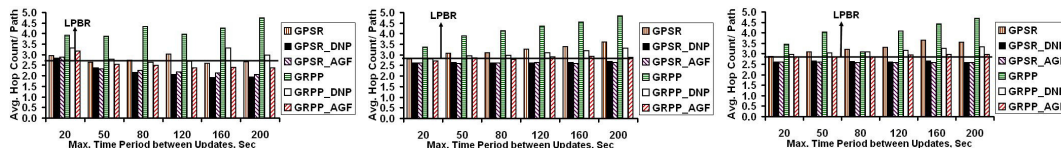

Figure 7.1: v_{max} = 5 m/s   **Figure 7.2:** v_{max} = 30 m/s   **Figure 7.3:** v_{max} = 50 m/s

Figure 7: Hop Count per Path of Position-based Routing Protocols (25 nodes and 15 *s-d* pairs)
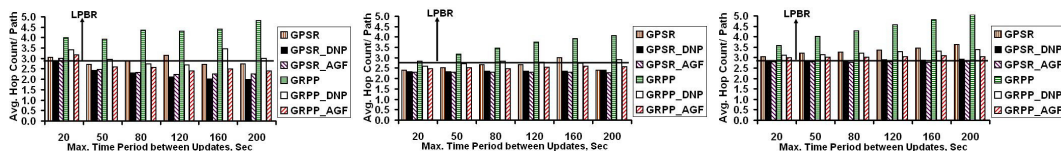

Figure 8.1: v_{max} = 5 m/s   **Figure 8.2:** v_{max} = 30 m/s   **Figure 8.3:** v_{max} = 50 m/s

Figure 8: Hop Count per Path of Position-based Routing Protocols (25 nodes and 30 *s-d* pairs)
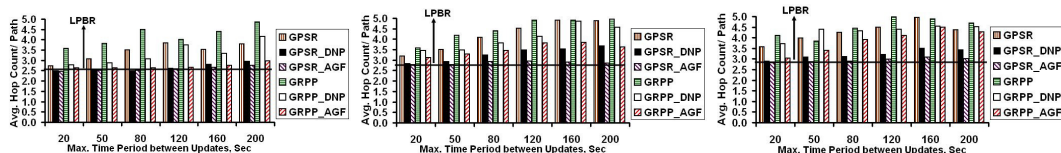
Figure 9.1: v_{max} = 5 m/s   **Figure 9.2:** v_{max} = 30 m/s   **Figure 9.3:** v_{max} = 50 m/s

Figure 9: Hop Count per Path of Position-based Routing Protocols (50 nodes and 15 *s-d* pairs)

For a given value of *MaxT_update*, the percentage of data packets getting forwarded through perimeter routing for GPSR increases as the maximum node velocity increases. With perimeter routing, due to the lack of nodes lying on a straight line path connecting the source and the destination, packets get forwarded on a path around the straight line. As more data packets get forwarded through perimeter routing, the average hop count of the paths for a session increases. The average hop count of GPSR routes in networks of moderate and higher node mobility is 15-20% more than the hop count incurred in networks of low node mobility. For a given offered data traffic load and node mobility, as the value of *MaxT_update* is increased from 20 to 200 seconds, the increase in the hop count of GPSR is by a factor of 20-40%.

The hop count of GRPP is higher than that of LPBR by a factor of 20 to 50%. The main reason for this increase in the hop count for GRPP is that while forwarding a data packet towards the destination, the forwarding node chooses the next hop node as the neighbor node that exists (i.e., currently exists and predicted to exist for the next three seconds) closer to the straight line joining the forwarding node's location and the location of the ultimate packet destination, as specified in the data packet header. The hop count would have minimized if the forwarding node chooses the next hop node as the neighbor node that is closer to the destination (similar to the GPSR approach). But, GRPP adopts the "stay on the line through a sequence of stable links" approach to maximize the chances of a data packet reaching the ultimate destination, even if the tradeoff is a higher hop count.
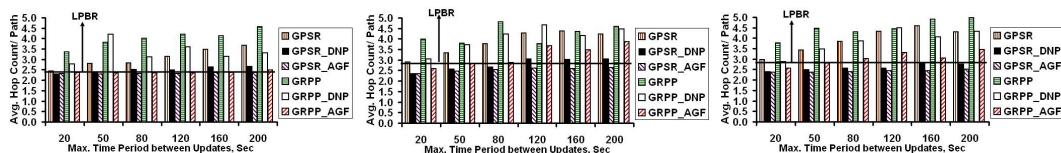

Figure 10.1: v_{max} = 5 m/s   **Figure 10.2:** v_{max} = 30 m/s   **Figure 10.3:** v_{max} = 50 m/s

Figure 10: Hop Count per Path of Position-based Routing Protocols (50 nodes and 30 *s-d* pairs)

We also observe that for a given offered data traffic load and node mobility, the hop count of GRPP paths in networks of high density is 10 to 25% more than that of the GRPP paths in networks of low density. The advantage is a relatively higher packet delivery ratio (by a factor of 10 to 20%) for GRPP in high-density networks when compared to low-density networks. In high density networks, there are greater chances of finding a source-destination path with all the intermediate nodes located on or closer to the straight line joining the source and the destination. GRPP prefers to go through such "stay on the line" paths (which can have more intermediate nodes) rather than paths that connect the source and destination with the minimum required number of intermediate nodes, but the intermediate nodes are located far away from the straight line joining the source and destination locations.

The hop count of the GRPP routes in networks of moderate and higher node mobility is 15 to 35% more than the GRPP hop count incurred in networks of low node mobility. This could be attributed to the fact that at the time of forwarding a data packet using GRPP, the forwarding node attempts to choose a next hop node as the neighbor node that would be connected to it for

the immediate future so that the data packet sent to the neighbor node does not get lost due to the neighbor node suddenly moving away. In order for a forwarding node to choose a neighbor node with which a stable link is predicted to exist at least for the immediate future, the forwarding node and the neighbor node should be either moving towards each other or be moving parallel to each other separated by a smaller distance.

The DNP and AGF versions are observed to be very effective in reducing the hop count per path incurred by the original versions of both GPSR and GRPP. For a given value of $MaxT_{update}$, the reduction in the hop count achieved with GPSR_DNP and GPSR_AGF (in comparison with GPSR) is by factors of 20-30% and 25-45% respectively. Similarly, for a given value of $MaxT_{update}$, the reduction in the hop count achieved with GRPP_DNP and GRPP_AGF (in comparison with GRPP) is by factors of 5-15% and 10-30% respectively. Both DNP and AGF are effective strategies to avoid routing loops encountered with the original versions of GPSR and GRPP. If the destination node is located in the one-hop neighborhood (in the case of DNP and AGF) or is located in the two-hop neighborhood (in the case of AGF), a data packet is forwarded to the destination node (if located in the one-hop neighborhood) or to the neighbor of the destination node (if located in the two-hop neighborhood) and is not forwarded based on the destination location co-ordinates in the data packet header.

4.3 End-to-End Delay per Data Packet

Figures 11 through 14 illustrate the end-to-end delay per data packet for all the routing protocols. LPBR incurs the lowest end-to-end delay per data packet for all of the simulation conditions, except the scenario of low network density and high offered data traffic load, in which LPBR incurs the second lowest end-to-end delay per data packet. In networks of low density and high offered traffic load, almost all the nodes play the role of intermediate nodes for at least one *s-d* session and each node acts as the source or destination for at least one session. The end-to-end delay per data packet incurred by LPBR is larger than that incurred by GPSR_DNP and GPSR_AGF by a factor of 10% – 16%.
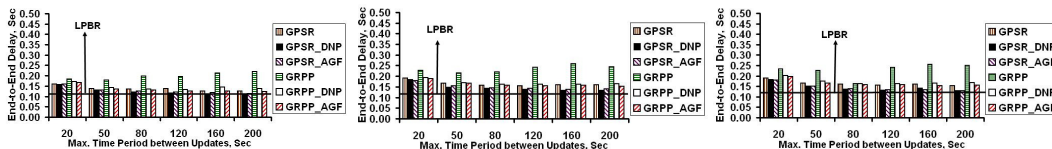

Figure 11.1: v_{max} = 5 m/s   **Figure 11.2:** v_{max} = 30 m/s   **Figure 11.3:** v_{max} = 50 m/s

Figure 11: Delay per Packet for Position-based Routing Protocols (25 nodes, 15 *s-d* pairs)
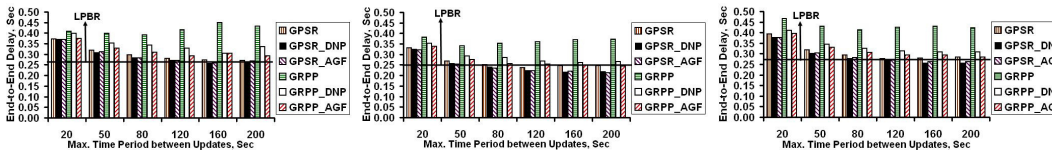

Figure 12.1: v_{max} = 5 m/s   **Figure 12.2:** v_{max} = 30 m/s   **Figure 12.3:** v_{max} = 50 m/s

Figure 12: Delay per Packet for Position-based Routing Protocols (25 nodes, 30 *s-d* pairs)
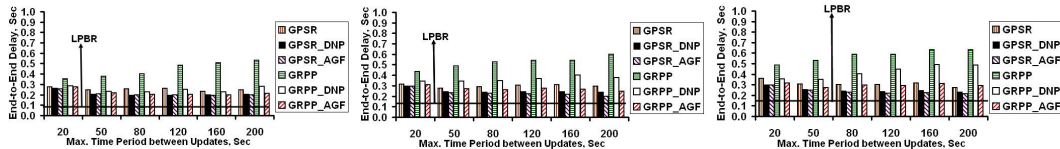
Figure 13.1: v_{max} = 5 m/s   **Figure 13.2:** v_{max} = 30 m/s   **Figure 13.3:** v_{max} = 50 m/s

Figure 13: Delay per Packet for Position-based Routing Protocols (50 nodes, 15 *s-d* pairs)
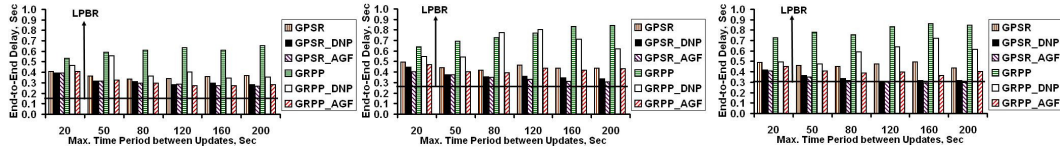

Figure 14.1: v_{max} = 5 m/s   **Figure 14.2:** v_{max} = 30 m/s   **Figure 14.3:** v_{max} = 50 m/s

Figure 14: Delay per Packet for Position-based Routing Protocols (50 nodes, 30 *s-d* pairs)

For the position-based routing protocols, the end-to-end delay per data packet decreases as we use higher values of $MaxT_{update}$. But, the decrease in the end-to-end delay per data packet is not proportional to the increase in $MaxT_{update}$ value. For a given offered data traffic load and node mobility, the decrease in the end-to-end delay per data packet is by factors of 20-30% (for low density networks) and 10-15% (for high density networks) when the $MaxT_{update}$ value is increased from 20 seconds to 200 seconds. In networks of low density (25 nodes) and low offered data traffic load (15 *s-d* pairs), for a given node mobility, the end-to-end delay per data packet incurred by the position-based routing protocols (GPSR, GPSR_DNP, GPSR_AGF, GRPP_DNP and GRPP_AGF) except GRPP is about 1.4 – 1.6 (when $MaxT_{update}$ = 20 seconds) and 1.1 – 1.3 (when $MaxT_{update}$ = 200 seconds) times of that incurred by LPBR. In networks of low density (25 nodes) and high offered data traffic load (30 *s-d* pairs), for a given node mobility, the end-to-end delay per data packet incurred by the position-based routing protocols except GRPP is about 1.3 – 1.6 (when $MaxT_{update}$ = 20 seconds) and 0.85 – 1.2 (when $MaxT_{update}$ = 200 seconds) times of that incurred by LPBR. In networks of high density (50 nodes) and low offered data traffic load (15 *s-d* pairs), for a given node mobility, the end-to-end delay per data packet incurred by the position-based routing protocols except GRPP is about 2.0 – 3.4 (when $MaxT_{update}$ = 20 seconds) and 1.4 – 3.3 (when $MaxT_{update}$ = 200 seconds) times of that incurred by LPBR. In networks of high density (50 nodes) and high offered data traffic load (30 *s-d* pairs), for a given node mobility, the end-to-end delay per data packet incurred by the position-based routing protocols except GRPP is about 1.3 – 3.0 (when $MaxT_{update}$ = 20 seconds) and 1.05 – 2.3 (when $MaxT_{update}$ = 200 seconds) times of that incurred by LPBR. The relative decrease in the magnitude of the difference is attributed to the increase in the route discovery control message overhead (route discoveries and route maintenance/ repair) for LPBR as we increase the offered data traffic load and/or node mobility.

The end-to-end delay per data packet incurred by GRPP could be as large as 1.5 – 2.0 times to that incurred by GPSR. This is mainly attributed to paths of larger hop count chosen by GRPP. The DNP and AGF versions of GRPP attempt to reduce the delay per packet significantly by routing the data packet through the shortest paths as and when possible. The reduction in the end-to-end delay per data packet incurred with GRPP_DNP and GRPP_AGF in comparison with that incurred using GRPP could be as large as by a factor of 1.5 – 2.2.

4.4 Packet Delivery Ratio

Figures 15 through 18 illustrate the packet delivery ratio achieved with the routing protocols simulated in this paper. LPBR achieves the highest packet delivery ratio among all the protocols

in all the simulation conditions tested. This indicates the effectiveness of the location prediction approach and the route prediction technique adopted by LPBR. As LPBR undergoes the minimal number of route discoveries, there is not much route discovery control overhead traffic that blocks the data packets from going through the queues of the nodes. Among the four sets of simulation conditions tested (50 nodes with 30 *s-d* pairs, 50 nodes with 15 *s-d* pairs, 25 nodes with 30 *s-d* pairs, 25 nodes with 30 *s-d* pairs), the 25 nodes with 30 *s-d* pairs scenario generates the maximum amount of data traffic load per node in the network because the routes between the 30 *s-d* pairs have to be handled by only 25 nodes in the network and most of the nodes are also either source and/or destination of at least one *s-d* session. Though the packet delivery ratio of LPBR drops by 10 to 25% for this scenario, LPBR still incurs the highest packet delivery ratio among all the routing protocols. The packet delivery ratio of GRPP, GPSR_AGF and GRPP_AGF are the closest to that of the LPBR and these protocols have the next highest packet delivery ratio in most of the simulation conditions. This is attributed to the "stay on the line through a sequence of stable links" path selection approach of GRPP and the effective one-hop/two-hop neighborhood based destination location identification approach of AGF.

The packet delivery ratio of GPSR is about 5 to 15% less than that of GRPP. Note that the packet delivery ratios for GPSR and GRPP decrease with increase in the value of $MaxT_{update}$. This is attributed to the lack of accurate information of the destination location as the time period between two successive destination location update searches increases. As a result, data packets are subjected to more of perimeter forwarding and routing loops and hence get dropped eventually. On the other hand, the DNP and AGF versions of both GPSR and GRPP are not much affected by the value of $MaxT_{update}$ because they rely on locally finding the destination node based on the neighborhood information collected during the per-second periodic beacon exchange. To be more specific, we observe that the packet delivery ratio incurred by GPSR_DNP, GPSR_AGF, GRPP_DNP and GRPP_AGF is almost close to that of LPBR when $MaxT_{update}$ values are larger. This is attributed to the relative reduction in the HLS cellcasting overhead incurred as part of the destination location update search process and availability of more space in the queue of the nodes for forwarding the data packets.
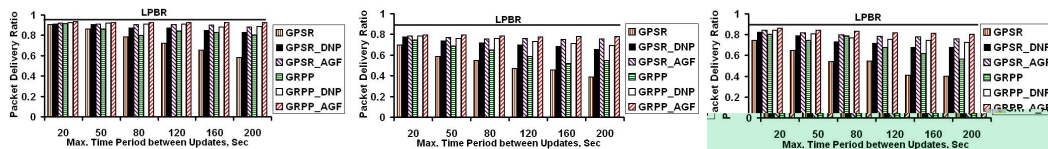

| **Figure 15.1:** v_{max} = 5 m/s | **Figure 15.2:** v_{max} = 30 m/s | **Figure 15.3:** v_{max} = 50 m/s |

Figure 15: Packet Delivery Ratio of Position-based Routing Protocols (25 nodes, 15 *s-d* pairs)

| **Figure 16.1:** v_{max} = 5 m/s | **Figure 16.2:** v_{max} = 30 m/s | **Figure 16.3:** v_{max} = 50 m/s |

Figure 16: Packet Delivery Ratio of Position-based Routing Protocols (25 nodes, 30 *s-d* pairs)
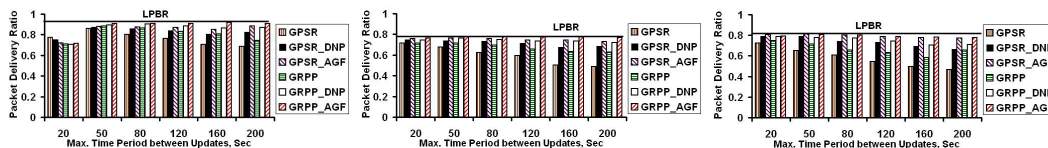
| **Figure 17.1:** v_{max} = 5 m/s | **Figure 17.2:** v_{max} = 30 m/s | **Figure 17.3:** v_{max} = 50 m/s |

Figure 17: Packet Delivery Ratio of Position-based Routing Protocols (50 nodes, 15 *s-d* pairs)

| **Figure 18.1:** v_{max} = 5 m/s | **Figure 18.2:** v_{max} = 30 m/s | **Figure 18.3:** v_{max} = 50 m/s |

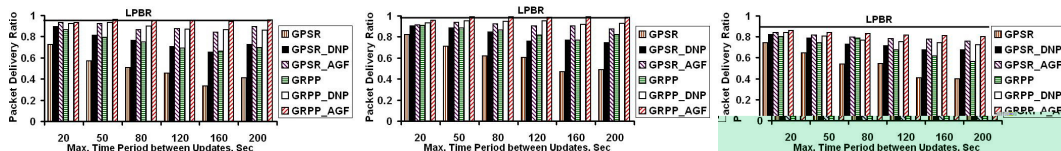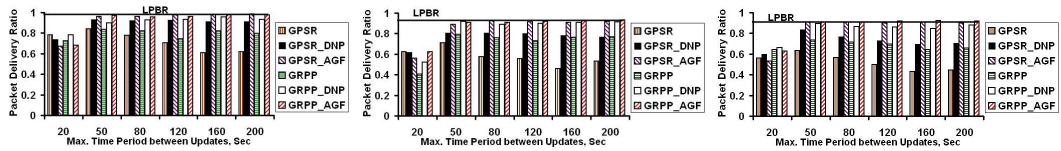**Figure 18:** Packet Delivery Ratio of Position-based Routing Protocols (50 nodes, 30 *s-d* pairs)

5. CONCLUSIONS

The high-level contribution of this paper is the performance comparison of our recently proposed mobile ad hoc network routing protocol called Location Prediction Based Routing (LPBR) protocol with that of the position-based routing protocols. LPBR is the first such beaconless MANET routing protocol to simultaneously minimize the route discovery control overhead as well as the hop count per path. We ran extensive simulations (in ns-2) by varying the network density, node mobility and offered data traffic load. We compared the performance of LPBR with the position-based GPSR, GRPP protocols and their improved versions GPSR_DNP, GPSR_AGF, GRPP_DNP and GRPP_AGF. LPBR incurs the least route discovery control message overhead and the highest packet delivery ratio among all the routing protocols simulated. We observe a tradeoff between the packet delivery ratio and the destination location search overhead in the case of the position-based routing protocols. As the maximum time period between two successive destination location searches is reduced, the accuracy of destination location information included in the data packet header is increased leading to an improvement in the packet delivery ratio. But, this is achieved at the cost of a higher destination location search control message overhead. Nevertheless, the DNP and AGF versions of GPSR and GRPP could yield packet delivery ratios that are off by only at most 10% (compared to LPBR) and still incur a lower control message overhead.

LPBR does not require periodic beacon exchange within the one-hop neighborhood. We observe that LPBR incurs the least number of route discoveries and the hop count per path is only at most 10% more than the minimum hop count and is larger than that of the GPSR_DNP and GPSR_AGF routes, for certain simulation conditions only, by at most 10-20%. LPBR incurs the lowest end-to-end delay per data packet for all of the simulation conditions, except the scenario of the low network density and high offered data traffic load, in which LPBR incurs the second lowest end-to-end delay per data packet and it is larger than that incurred by GPSR_DNP and GPSR_AGF by about 10% – 16%.

All of the above performance results indicate the effectiveness of the location prediction approach and the route prediction approach of LPBR. Even though, there is some overhead in including the location and mobility information in the RREQ packets and the succeeding packet sending time information in the data packets, we find this to be a small and useful overhead compared to the huge, frequently occurring destination location search overhead incurred with the position-based routing protocols. As there is currently no single beaconless routing protocol,

other than LPBR, that simultaneously minimizes the number of route discoveries as well as the hop count per path, LPBR is a valuable addition to the MANET literature.

REFERENCES

[1] S. Agarwal, A. Ahuja, J. P. Singh and R. Shorey, "Route-Life Time Assessment Based Routing Protocol for Mobile Ad Hoc Networks," *Proceedings of IEEE International Conference on Communications,* pp. 1697 – 1701, June 2000.

[2] 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.

[3] 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, March 2000.

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

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

[6] W. Creixell and K. Sezaki, "Routing Protocol for Ad hoc Mobile Networks using Mobility Prediction," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 2, no. 3, pp. 149-156, 2007.

[7] K. Fall and K. Varadhan, "ns notes and documentation," The VINT Project at LBL, Xerox PARC, UCB, and USC/ISI, http://www.isi.edu/nsnam/ns, August 2001.

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

[9] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek and M. Degermark, "Scenario-based Performance Analysis of Routing Protocols for Mobile Ad hoc Networks," *Proceedings of the 5th ACM Annual International Conference on Mobile Computing and Networking*, pp. 195-206, Seattle, WA, USA, August 1999.

[10] 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.

[11] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, pp. 243-254, Boston, MA, USA, August 2000.

[12] W. Keiss, H. Fuessler and J. Widmer, "Hierarchical Location Service for Mobile Ad hoc Networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 4, pp. 47-58, 2004.

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

[14] 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*, pp. 108-119, 2006.

[15] D. Son, A. Helmy and B. Krishnamachari, "The Effect of Mobility-Induced Location Errors on Geographic Routing in Mobile Ad hoc Sensor Networks: Analysis and Improvement using Mobility Prediction," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 233-245, July-August 2004.

[16] W. Su, S-J Lee and M. Gerla, "Mobility Prediction and Routing in Ad Hoc Wireless Networks," *International Journal of Network Management*, vol. 11, no. 1, pp. 3-30, January-February 2001.

[17] C-K. Toh, "Associativity-Based Routing for Ad hoc Mobile Networks," *IEEE Personal Communications*, vol. 4, no. 2, pp. 103 – 139, March 1997.

[18] K. Yamazaki and K. Sezaki, "A Proposal of Geographic Routing Protocols for Location-aware Services," *Electronics and Communications in Japan (Part I: Communications)*, vol. 87, no. 4, pp. 26-34, December 2003.