

GPS-Based Route Discovery Algorithms for On-Demand Routing Protocols in MANETs

Mehran Abolhasan and Tadeusz Wysocki

Telecommunication and Information Research Institute,
University of Wollongong, NSW 2522, Australia
mehran@titr.uow.edu.au, wysocki@uow.edu.au

Abstract. This paper presents new Global Positioning System (GPS)-based route discovery algorithms for on-demand routing in MANETs, called Position-based Selective Flooding (PSF). We applied our route discovery algorithm to our previous routing protocol, which is called Location-based Point-to-point Adaptive routing (LPAR) protocol and investigated its performance by simulation. Simulation results show that our position based flooding algorithm produce fewer routing overheads than the pure flooding, expanding ring search (used in AODV), LAR1 and our existing LPAR strategy, as network traffic and density is increased. Furthermore, we propose a number of improvements and variations which can be used instead of, or to further improve the performance of PSF under different network conditions.

1 Introduction

Mobile Ad Hoc Networks (MANETs) have become one of the most highly researched areas in wireless local area networking. The current research in MANETs involves all layers in the TCP/IP model, as the very nature of these networks demands new design rules for each layer in order to provide efficient end-to-end communications. MANETs are made up of a number of nodes, such as laptops and Personal Digital Assistance (PDA), which are capable of communicating with each other without using a fixed base station. This means that each node performs routing in a distributed manner. The limitation in transmission range and the highly dynamic nature of these networks, makes data transmission between the source and the destination travel over multiple hops, which can vary over time. Therefore, routing in MANETs is a challenging task. In the MANET literature, a number of different routing protocols have been proposed. Designing routing strategies for MANETs began by optimising the routing protocols designed for wired networks[7]. The route discovery in these protocols were proactive in nature, which means that each protocol periodically exchanged routing information with other nodes in the network, in order to build their routing tables. However, this approach to routing lacks scalability as the size of the network grows. On-demand routing protocols were designed to reduce the route discovery overheads by allowing each node to determine routes when

they are required, rather than maintaining a route to every destination. This routing strategy consists of two phases: route discovery and route maintenance.

A node which requires a route to a particular destination, starts a route discovery phase, where a Route Request (RREQ) packet is propagated through the network until the destination or an intermediate node to the destination is found or the packet expires. When a route is found, a Route Reply (RREP) is sent back to the destination using link reversal if the RREQ has travelled over bidirectional links or by flooding if unidirectional links are used. The route maintenance phase is initiated by an intermediate node, which experiences a link failure while a route is still active. In this phase, the route can be repaired locally at the point of failure by using a localised route maintenance strategy [4] or the node which detected the link failure notifies the source via a Route Error (RERR) message and the source will either use another route or initiate another route discovery [6].

Routing in on-demand protocols can be classified into two groups: source routing and point-to-point (also called hop-by-hop) routing. In source routing protocols such as [6], each data packet carries the complete source to destination address, whereas in point-to-point routing [4][3], data packets only carry the destination address and the address of the next hop which leads to the destination. This means that each intermediate node in an active route can make routing decisions, thereby allowing active routes to be adaptable to topology changes, whereas in source routing all the routing decisions are made at the source. This means that link failure in an active route may result in initiation of additional route discoveries at the source or at the point of failure¹. Furthermore, in source routing, an increase in the number of hops in the active route will result in an increase in the amount of overhead carried by each packet. In contrary, in point-to-point routing the size of each packet is not affected by multihopping. Therefore, point-to-point routing has more potential to scale better as the size of the network increases.

In our previous study of point-to-point routing [3], we proposed Location-based Point-to-point Adaptive Routing (LPA). In this study, we introduce a new approach to reduce route discovery overheads, given that each source node possesses location information about the required destination. Furthermore, in LPA, we proposed a number of different strategies to minimise the effects of link failure on the active route and increase the stability of each route. In this paper, we introduce new strategies to reduce route discovery overhead, while maintaining high levels of throughput when the source has no location information about the destination. We implemented our route discovery strategy on top of LPA and compared its performance with AODV and LAR1 using simulation. Our results show that this new approach has fewer overheads than AODV and LAR1, and has higher levels of scalability as the size (i.e. boundary), node density and traffic in the network grows. The rest of this paper is organised as follows. Section 2, describes our route discovery strategy. Section 3, describes the simulation environment and the parameters used. Section 4, presents a dis-

¹ If a localised route repair strategy is used.

cussion on our simulation results. Section 5, presents a number of alternative strategies and improvements for our routing strategy and section 6 presents the concluding remarks.

2 Proposed Strategy

In this section, we propose Position-based Selective Flooding (PSF). In pure flooding or in ERS, all the neighbouring nodes usually rebroadcast the RREQ message, unless the TTL has expired. In a dense network, routing overhead can be significantly reduced by strategically selecting the retransmitting nodes to cover the entire network (or a selected area). In PSF, only a number of different nodes forward the RREQ packet, based on a selection criteria described below. We have also proposed a number of variations and improvements to make PSF more efficient.

2.1 Overview and Definition

This strategy reduces the number of re-broadcasts during route discovery by allowing nodes, which are positioned in a determined region, to re-broadcast the routing packets. To illustrate how this strategy works, suppose node S (see Figure 1), wants to determine a route to node D. Node S will initiate its route discovery, and a RREQ is broadcasted, which stores the source nodes location information. The receiving nodes then determine their relative distance to node S and rebroadcast the RREQ if they map into the Forwarding Region (FR). Note that the idea behind choosing FR comes as a result of the following observations:

1. Nodes that are located near the boundary of the transmission range, R , will create unstable (or short lived) links if they are selected as intermediate nodes in an active route.
2. Selection of intermediate nodes which are close together will increase the number of hops in each route. This means that end-to-end delay will increase during data transmission. Furthermore, probability of route failure may increase. Since the number of intermediate nodes in an active route increases, then the probability of a link failure causing the route failure will increase.
3. In a dense network, flooding over neighbours which are very close to each other may not significantly increase the probability of a successful route discovery or searching the entire network. In this case, routing overhead can be significantly reduced by strategically selecting the rebroadcasting nodes.

Each node, which receives a rebroadcasted RREQ packet, will also calculate their own FR. If their location coordinates map within the FR and they are further away from the source than the previous hop, they will rebroadcast the RREQ packet. This is done by (see Figure 1) multiplying R_{max} by the hop count and setting $R_{min} = R_{max} - K$. Therefore, the RREQ packet will continue to propagate away from the source at each hop. Note that K is a variable, which

determines the width of FR. In our simulation, we used a constant value for K . However, in section 6, we propose a number of different strategies, which can be used to dynamically select values for R_{max} and K at each hop by taking into account the location and the number of neighbouring nodes, for each node. The advantage of PSF is that RREQ packets do not need to carry a forwarding list² to limit the number of rebroadcasts, as compared to the neighbour aware strategies such as MPR. This means that the size of each RREQ packet will be smaller. Furthermore, nodes do not need to maintain 2-hop topology information.

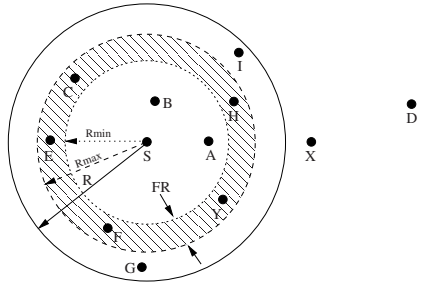


Fig. 1. Illustration of the forwarding region

2.2 Mathematical Model

In PSF, a node rebroadcasts a RREQ message if it satisfies the following three conditions:

1. The rebroadcasting node must be further away from the source than the downlink node it received the RREQ from.
2. The RREQ packet has not been seen before, or it has not expired
3. The forwarding node must lie within the FR.

We calculate FR at each node, such that condition one is also met. To show how this is achieved, assume that the MANET topology can be seen as an undirected graph $G = (V, E)$, where V represents a set of mobile nodes connected by a set of edges (i.e. links), E , if the distance between two nodes is less than R . Each node i , in G , has a set of neighbours n_x such that³ $d(i, n_x) \leq R$. Let h be the hop count. At each hop the nodes must satisfy the following condition in order to be able to rebroadcast.

$$(r_{min} = r_{max} - K) \leq d(i, n_x) \leq (r_{max} = C * h) \quad (1)$$

For $h = 1 \dots N$

² A list of rebroadcasting nodes

³ Assume all nodes have equal transmission range

2.3 Theoretical Overhead Analysis

In PSF, the number of rebroadcasting nodes are lower than in pure flooding at each hop, when the size of the forwarding region is less than the maximum transmission radius. To show this, let $N_{max} = |N_x|$ be the maximum number of neighbours for a particular node, and let N_{tx} be the number of retransmitting nodes. Now assume that all nodes are equally distributed in the network and all nodes have equal transmission range. In pure flooding the flooding area for a particular node is πR^2 and the number of retransmitting nodes is equal to N_{max} (i.e. $N_{tx} = N_{max}$). In PSF, the flooding area (or the forwarding region) is $A_{FR} = \pi r_{max}^2 - \pi r_{min}^2$. Let $L_{FR} = r_{max} - r_{min}$ be the width of the FR. Then, it can be easily seen that, in PSF, $N_{tx} \leq N_{max}$ for $L_{FR} \leq R$, or:

$$\lim_{L_{FR} \rightarrow R} A_{FR} = \pi R^2 \tag{2}$$

Therefore, only in the worst-case scenario, where $L_{FR} = R$, PSF will converge to pure flooding, and for cases where the required destination could be easily found with $L_{FR} \ll R$, then $N_{tx} \ll N_{max}$. This means that the number of RREQ packets propagating through the network will be far lower. To illustrate this with an example, suppose node S (see figure 2), wants to find a route to node D, and assume node S initiated a route discovery with a Time To Live (TTL) of 2, indicating that the RREQ packet can only travel over 2 hops. Suppose that the FR is calculated as shown in figure 2, which shows that nodes H, Y and M are in FR. Therefore, only these nodes rebroadcast the RREQ. When node I, receives a RREQ from node H, it will send a RREP back to the source, using link reversal⁴. This will then bring the total number of broadcast to 6 (i.e. 4 RREQs and 2 RREPs). Now suppose all nodes within R where able to retransmit, then the total number of broadcasts will be 12 (i.e. 10RREQs and 2RREPs). Therefore, in this simple scenario, PSF produced 50% less control packets than pure flooding.

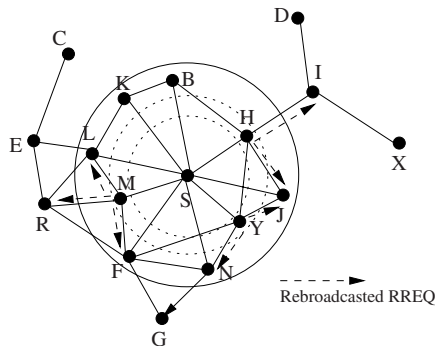


Fig. 2. An example of route discovery using PSF

⁴ assuming only bidirectional links are used

3 Simulation Environment and Performance Metric

We performed our simulations using the GloMoSim[1] simulation package. GloMoSim is an event driven simulation tool designed to carry out large simulations for mobile ad hoc networks. Our simulations were carried out for 100, 200, 300, 400 and 500 node networks, migrating in a 1000m x 1000m area. IEEE 802.11 DSSS (Direct Sequence Spread Spectrum) was used with maximum transmission power of 15dbm at a 2Mb/s data rate. In the MAC layer, IEEE 802.11 was used in DCF mode. The radio capture effects were also taken into account. Two-ray path loss characteristics was considered as the propagation model. The antenna height was set to 1.5m, the radio receiver threshold was set to -81 dbm and the receiver sensitivity was set to -91 dbm according to the Lucent wavelan card[2]. Random way-point mobility model was used with the node mobility ranging from 0 to 20m/s and pause time varied from 0 to 900s. The simulation was run for 900s for 10 different values of pause time, and each simulation was averaged over eight different simulation runs using different seed values. Constant Bit Rate (CBR) traffic was used to establish communication between nodes. Each CBR packet was contained 512 Bytes and each packet were at 0.25s intervals. The simulation was run for 10 and 20 different client/server pairs⁵ and each session was set to last for the duration of the simulation. In our simulation study of PSF, we set constants, which are used to calculate R_{max} and R_{min} (i.e. C and K), to 300m and 150m respectively. Therefore, the length of FR, $L_{FR} = 150m$. This simple model was used in our simulations to show the benefits of PSF in medium to large networks. In section 6, we propose a number of strategies to dynamically select values for C and K to increase the efficiency of the algorithm under different levels of node density.

The performance of each routing protocol is compared using the following performance metrics.

- Packet Delivery Ratio (PDR)
- Control (O/H)
- End-to-End Delay

PDR is the Ratio of the number of packet sent by the source node to the number of packets received by the destination node. Control (O/H) presents the number of routing packets transmitted through the network for the duration of the simulation. This metric will illustrate the levels of the introduced routing overhead in the network. Finally, the End-to-End Delay metric illustrates the average end to end delay for transmitting one data packet from the source to the destination.

4 Results

In this section, we present a discussion on our simulation results. Note that we implemented the PSF strategy on the top of our existing routing protocol,

⁵ Note that the terms Client/Server, src/dest and Flows are used interchangeably

which is called LPAR, and we refer to this as LPAR-PSF. The performance of LPAR-PSF was compared with LPAR-S⁶, LPAR, LAR1 and AODV.

4.1 Packet Delivery Ratio Results

Figure 3 and 4, show the PDR for a 100 node and 500 node network, with 10 src/dest pairs. These results, illustrate the performance of the protocols in a moderately dense and a highly dense network. In the 100 node scenario all protocols achieve over 95% PDR during the high mobility phase, where the pause time is low, and achieve over 97% PDR for mid-range mobility to zero mobility. However, in the 500 node scenario, the point to point based routing strategies outperform the source routing strategy. This is more evident under high levels of mobility, where LAR1, under-performs the other strategies. This can be due to a number of different reasons. Firstly, when a route failure occurs, in LAR1, a source node scans its route cache to use an alternate route. Under low levels of mobility, the routes in the route cache will stay active longer than under high levels of mobility. This is because the probability of a link failure in an active route (since complete source to destination address is used) increases in high mobility, which means that more route discoveries will be initiated at the source and more packets may be dropped in the process. This may become more evident as the amount of multihopping increases within each route. Furthermore, in the point to point routing strategies (i.e. LPAR and AODV) each intermediate node in an active route can learn and use a better route to the destination, which means that the route between the source to the destination may stay valid for longer time, whereas in LAR1, since each data packet relies on the source to destination address given in the header, a link failure in an active route may immediately cause a route failure, which require a RERR message to be sent back to the source where another route must be calculated or used (if available).

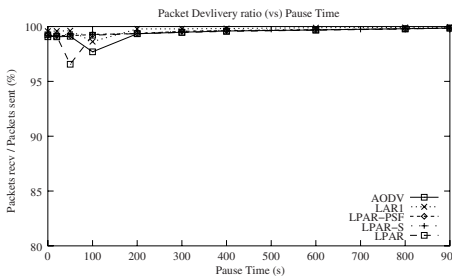


Fig. 3. PDR: 100N, 10 Flows

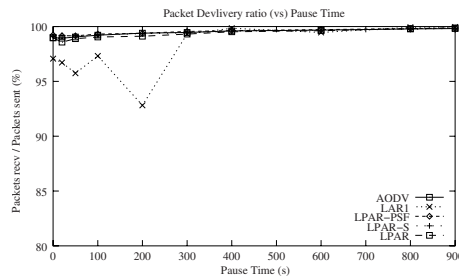


Fig. 4. PDR: 500N, 10 Flows

Figure 5 and 6, show the PDR for a 100 node and 500 node network, with 20 src/dest pairs. In the 100 node network, all protocols produce over 90% PDR.

⁶ This is the LPAR routing strategy, which also selects stable routes, in our previous study we referred to this as LPAR-S[3]

The point-to-point routing protocols perform very consistently over the different levels of mobility, in particular LPAR, LPAR-S and LPAR-PSF maintain over 97% PDR for all levels of mobility. LAR1, slightly under-performs under high mobility, where its performance drops to 90% for the 20 second pause sample. In the 500 node network, the performance of each routing strategy can be clearly distinguished. Here, LPAR-PSF has the best performance, where it maintains over 98% PDR. LPAR and LPAR-S also produce over 95% PDR. However, AODV's performance significantly drops under high mobility when compared to the 100 node scenario. This clearly highlights the advantage of exploiting location information to generate a more strict route discovery procedure and reduce bandwidth consumption in highly dense networks. LAR1 again shows the worst performance under high mobility, where its PDR varies between 80 and 85%.

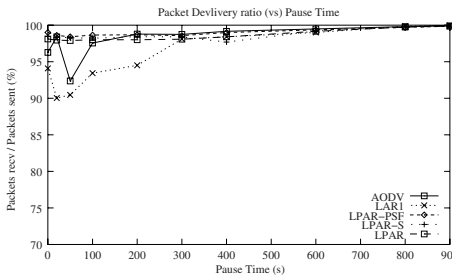


Fig. 5. PDR: 100N, 20 Flows

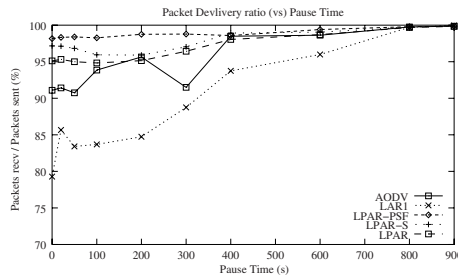


Fig. 6. PDR: 500N, 20 Flows

4.2 Control Overhead Results

Figure 7 and 8 show the number of control packets introduced into the network by each routing protocol, for 10 CBR sources, in a 100 node and a 500 node network respectively. AODV produces more control packets than all other routing strategies. This is more evident under high mobility, where AODV produces up to 10000 more control packets than its nearest competitor (i.e. LAR1). Two factors contribute to reducing routing overhead in LAR1 when compared to AODV. Firstly, nodes can have multiple routes to destinations stored in a route cache (as discussed earlier), which may reduce the number of route discoveries initiated for each src/dest pair, whereas in AODV, each node only stores a single route. Secondly, in LAR1, if source nodes have location information about the required destination, they can use RZS (as described earlier), which minimises (or localises) the search area to a particular region. The advantage of this is that the number of nodes involved in broadcasting RREQ packets is reduced, which means that fewer control packets are transmitted. This also allows more bandwidth to be available for the nodes that are not in the search area and reduce channel contention. LPAR and LPAR-S, which use the 3-state

route discovery algorithm, produce less overhead than LAR1, despite only storing single routes. This is because in our 3-state route discovery algorithm, if unexpired location information is available, the source node will first attempt to discover a route by unicasting rather than broadcasting (as previously described in section 2.4). Hence, fewer control packets are transmitted through the network. LPAR-S further reduces this overhead by flooding over links which have certain level of stability. The advantage of this is that the route may last longer, which means fewer route recalculations will be required and fewer data packets will be dropped. LPAR-PSF produces fewer control packets than all other routing strategies. In the 100 node network LPAR-PSF produces upto 2500 less control packets than LPAR-S and up to 17000 packets less than AODV in high mobility. Under higher node density (i.e. the 500 node scenario), LPAR-PSF produces up to 20000 less packets than LPAR-S and up to 110000 less packets than AODV. In the 20 src/dest scenario (see Figure 9 and 10), the gap between LPAR-PSF and the other strategies increases, particularly during high levels of mobility. It can be seen here that LPAR-PSF, produces 40000 less control packets than LPAR-S and up to 500000 less control packets than AODV. This clearly, shows the benefits of using PSF in highly dense networks.

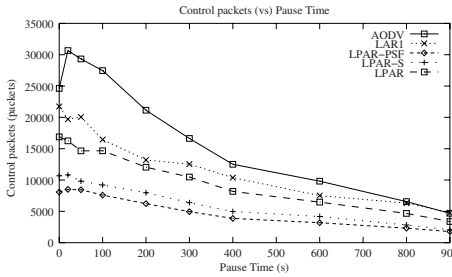


Fig. 7. CTRL: 100N, 10 Flows

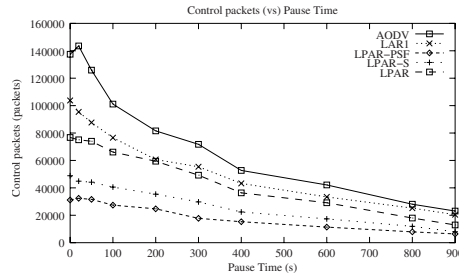


Fig. 8. CTRL: 500N, 10 Flows

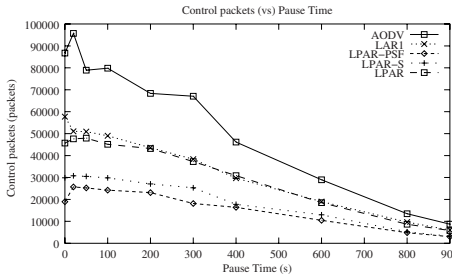


Fig. 9. CTRL: 100N, 20 Flows

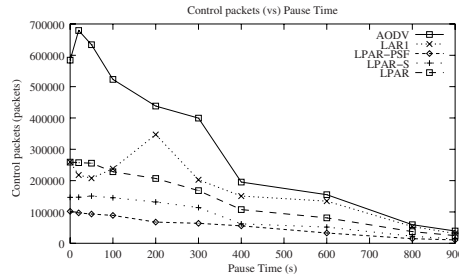


Fig. 10. CTRL: 500N, 20 Flows

4.3 Delay Results

Figures 11 and 12 show the average end-to-end delay experienced by each data packet for a 100 and a 200 node network with 10 src/dest pairs. As expected, all protocols experienced larger delays during high mobility, since more frequent link failure may cause route recalculation. This means that packets may experience longer delays before they reach their destination. In the 100 node network scenario, AODV has lowest end-to-end delay when compared to the other protocols. This is because, AODV always uses the shortest route to the destination and it only maintains a single route, whereas LAR1 can store multiple routes. This means that if the optimal route fails (the one with the shortest src/dest path), an alternate route from the route cache may be used. Therefore, some packet may travel over longer routes to reach the destination. Similarly in LPAR and LPAR-S and LPAR-PSF a secondary route may be available for each active route [3]. Therefore, if the primary route fails, some packets may travel over the secondary route, which may be longer in length. Hence, they may experience slightly longer delays. From the figure 11, we can see that LPAR and LPAR-S have on average about 5ms more delay across all ranges of mobility. However, by using a secondary route, LPAR and LPAR-S are able to successfully transmit more data packets, and reduce the number of route recalculations, which means fewer control packets. In the 200 node scenario, the gap between AODV and the other routing strategies becomes smaller. This is because with a higher density more nodes are contending for the medium. Therefore, since AODV produces significantly more overhead than the other strategies, it will introduce more channel contention and consume more bandwidth than the other strategies. Hence, longer delays may be experienced by intermediate nodes in active route before they can gain access to the medium. In LPAR-PSF, however, significantly fewer control packets are transmitted than in AODV. Furthermore, only a number of selected nodes rebroadcast, which means that there will be less channel contention than in AODV. Therefore, in LPAR-PSF, even though the packets may travel over more hops, they still experience similar levels of delay when compared to AODV.

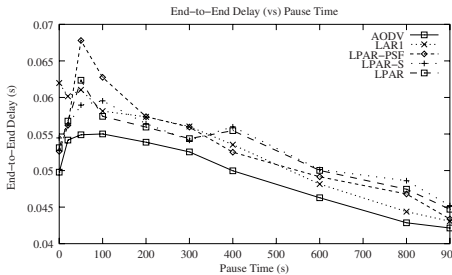


Fig. 11. Delay for 100 nodes

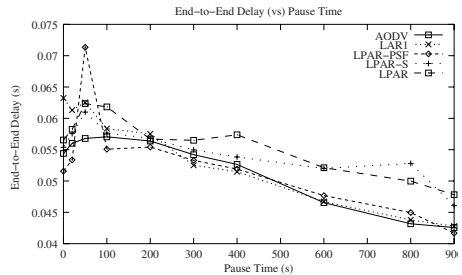


Fig. 12. Delay for 200 nodes

5 Alternative Strategies and Improvements

In our simulation study of PSF, we chose two constant values for C and K for every node to calculate the R_{max} and R_{min} in the PSF algorithm. This was done to illustrate the benefits which PSF could have in a medium to large network. However, selecting constant values may not be beneficial for every network size or topology. For example, chosen values for C and K for a large network may not produce good results in a small network and vice versa. Furthermore, we want every node to be able to calculate different FR's according to the node density of their neighbouring topology. Thus being able to successfully forward the RREQ message to different parts of the network, while minimising the number of re-transmitting nodes. In this section, we present a number of different strategies to dynamically select FR's at each node. Furthermore, we present an alternative strategy, which may also reduce route discovery overheads.

5.1 Source-Driven FR Selection

One way to introduce variable FR in the PSF algorithm is to determine its size at the source. We call this strategy Source-Driven FR selection or FR-SD. In this strategy, the source node specifies the values for C and K to be used at each hop. During the route discovery phase, the source node determines values for K and C and includes these values in the RREQ packet. Each time the route discovery fails (or the source node times out and no route is found), the distance between K and C, or R_{max} and R_{min} is increased and another route discovery is initiated. This process continues until a route is found or the distance between K and C becomes equal to R (i.e. $D_{KC} = R$). The *FR-SD* algorithm is outlined below.

Algorithm *FR-SD*

(* The FR-SD algorithm *)

1. $A \leftarrow$ Average Source/ neighbour distance
2. $D_{AR} \leftarrow R - A$ (* distance between R and A *)
3. $P \leftarrow \{0.25, 0.5, 0.75, 1.0\}$ (* % inc for K and C *)
4. **for** $i \leftarrow 0, i \neq 5, i++$
5. $C \leftarrow A + D_{AR}P_i$
6. $K \leftarrow A - AP_i$
7. initiate Route Discovery
8. wait for reply
9. **if** *Route = found*
10. break loop
11. initiate data transmission

In the FR-SD algorithm, the value of A can be determined by employing any of the following methods⁷

1. Assume a maximum number of nodes for a network with known area, then A can be approximated for each node[5]

⁷ Methods 1 and 2 can assume an equal node distribution

2. If the source has location information about every neighbour, then $A = \frac{N_T}{D_T}$, where N_T is the total number of neighbours and D_T is the total distance.
3. Assuming each node calculates A, and exchanges it with its neighbours using hello beacon messages, from the collected values of A from every neighbour, each node can calculate an average value of A. This will give an average neighbour distance per node for a 2-hop region.

5.2 Distributed Node-Density Based FR

In the previous strategy (i.e. FR-SD), the size of C and K were determined and enforced by the source. The disadvantage of this is that the calculated values for C and K may not result in optimal retransmission and coverage at every forwarding node. In this strategy, we attempt to make the selection of C and K distributed (hence the name FR-DN), thus allowing each forwarding node to determine an optimal FR for itself. To describe how this strategy works, suppose that node S initiates a route discovery, and broadcasts a RREQ packet, which includes its location information and FR. The receiving neighbours check to see if they lie in the FR according to the information in the received RREQ packet. If this is true, they calculate their own FR, which replaces the existing FR in the received RREQ packet. Similarly, the nodes which receive the rebroadcasted RREQ packet check to see if they lie in the previous hop's FR. If they do and they have not seen the packet before, then they calculate their own FR and rebroadcast the RREQ as long as they are further away from the source than the previous hop. To determine C and K for R_{max} and R_{min} at each hop, we calculate A as before (in FR-SD), then calculate $D_{AR} = R - A$, $C = A + D_{AR}P$ and $K = A - AP$. However, this time we want P to be inversely proportional to the node density N_d , where P is varied from 0% to 100%. To do this, we assign a minimum FR density threshold (D_{FR}), where we increase the size of P until we get a minimum number of nodes falling within the FR. Note that we can set D_{FR} to be a certain percentage of N_d , so that for large values of N_d , a small FR will be selected, and for a small N_d , a large FR will be selected. We also want $L_{FR} \propto \frac{1}{N_d}$. One way to determine this is to use a hyperbolic function such as, $f(x) = \frac{c}{x+1}$ or a sigmoid function, such as $f(x) = \frac{1}{1+e^{ax}}$, to scale the required D_{FR} according to the node density. To illustrate how a sigmoid function can be used to calculate the percentage difference, P, between C and K, let N_{max} be the maximum possible number of neighbours at each node and P be equal to the sigmoid. Then we want our sigmoid function to vary between 0 and 1, such that for $N_d \rightarrow N_{max}$ our sigmoid, $f(N_d) \rightarrow 0$ and for $N_d \rightarrow 0$, $f(N_d) \rightarrow 1$. Therefore, we let $f(N_d) = \frac{1}{1+e^{0.05(N_d - \frac{N_{max}}{2})}}$. Figure 13 illustrates the sigmoid function for $N_{max} = 100$. From this figure, it can be seen that as N_d approaches 100 the sigmoid starts to approach 0, and as N_d approaches to 0 the sigmoid starts to approach 1. This characteristic will allow us to scale P according to the node density, which varies between 0 and 1 (or 0% to 100%).

Another possible method, would be to use the standard deviation of A (the average neighbour distance), to vary the FR.

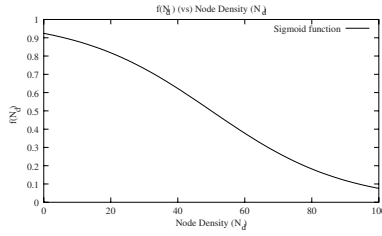


Fig. 13. Determining the percentage difference for FR using a sigmoid

5.3 Directional Node-Density

In FR-DN, the size of the FR is varied according to node density, by ensuring that a minimum number of nodes is present in the FR. However, FR-DN does not take into account the location of the nodes within FR. Hence, we introduce directional node density. The idea behind this strategy is to select the smallest L_{FR} that will contain a set of nodes which are located in a number of different parts of the FR. That is, we start with a small FR and increase its area until a number of strategically located nodes can be found within the FR. ⁸ To do this, we select a number of sample point (this can be any number of points, e.g. 4 point to represent North, East, South and West), which are A meters away from the source (like before A is the average node to neighbour distance), as shown in Figure 14. We then check to see if there is a set of nodes within the current FR, which are closely located (by a threshold distance that can be optimised by using it as a simulation parameter) to each of these points. If there are at least one node close to every point, the current FR will be accepted as large enough to be able to rebroadcast the RREQ message away from the source and to different parts of the network. If not, the size of the FR is increased, and the new FR is scanned to see if the new boundary will host nodes, which are close to the required points.

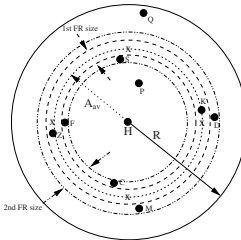


Fig. 14. Illustration of FR selection using Directional Density

⁸ Note that this strategy is an optimisation for FR-DN. Hence, it must be built on top of FR-DN.

To illustrate this, suppose node H (see Figure 14), calculates the 1st FR as shown. From the figure, it can be seen that within this FR only two nodes can be found near the required point (i.e. nodes K and Z). However, in the 2nd FR increase, at least one node is found near each required point (i.e nodes S, K, D, F, Z, C and M). Therefore, node H will use the second FR. The advantage of this strategy is that the FR ensures that the RREQ packet is propagated to different part of the network, and guarantees a certain level of coverage at each hop.

6 Conclusion

This paper presented a new routing discovery strategy for mobile ad hoc networks. We presented Position-based Selective Flooding (or PSF). In this strategy, only a number of selected nodes take part in route discovery. We implemented PSF on the top of LPAR and we referred to as LPAR-PSF. We compared its performance with LPAR, LPAR-S, LAR1 and AODV using simulations. Our results show that LPAR-PSF produces fewer overhead packets than, LPAR, LPAR-S, LAR1 and AODV, and achieves the highest levels of throughput in medium to large networks.

References

1. Glomosim scalable simulation environment for wireless and wired network systems. In <http://pcl.cs.ucla.edu/projects/glomosim/>.
2. Orinoco pc card. In <http://www.lucent.com/orinoco>.
3. Mehran Abolhasan, Tadeusz Wysocki, and Eryk Dutkiewicz. LPAR: An Adaptive Routing Strategy for MANETs. In *journal of Telecommunication and Information Technology*, pages 28–37, 2/2003.
4. S. Das, C. Perkins, and E. Royer. Ad Hoc On Demand Distance Vector (AODV) Routing. In *Internet Draft, draft-ietf-manet-aodv-11.txt*, work in progress, 2002.
5. Horst Hellbrück and Stefan Fischer. Towards Analysis and Simulation of Ad-Hoc Networks. In *ICWN02, pages 69–75, Las Vegas, Nevada, USA, June 2002*.
6. D. Johnson, D. Maltz, and J. Jetcheva. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks. In *Internet Draft, draft-ietf-manet-dsr-07.txt*, work in progress, 2002.
7. C.E. Perkins and T.J. Watson. Highly Dynamic Destination Sequenced Distance Vector Routing (DSDV) for Mobile Computers. In *ACM SIGCOMM'94 Conference on Communications Architectures*, London, UK, 1994.