Technical Report

UCAM-CL-TR-674
ISSN 1476-2986

Number 674

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

Landmark Guided Forwarding

Meng How Lim

October 2006

15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom
phone +44 1223 763500

http://www.cl.cam.ac.uk/

This technical report is based on a dissertation submitted September 2006 by the author for the degree of Doctor of Philosophy to the University of Cambridge, St Catharine's College.

Technical reports published by the University of Cambridge Computer Laboratory are freely available via the Internet:

http://www.cl.cam.ac.uk/TechReports/

ISSN 1476-2986

Abstract

Wireless mobile ad hoc network routing presents some extremely challenging research problems. While primarily trying to provide connectivity, algorithms may also be designed to minimise resource consumption such as power, or to trade off global optimisation against the routing protocol overheads. In this thesis, we focus on the problems of maintaining network connectivity in the presence of node mobility whilst providing a balance between global efficiency and robustness. The common design goal among existing wireless ad hoc routing solutions is to search for an optimal topological path between a source and a destination for some shortest path metric. We argue that the goal of establishing an end to end globally optimal path is unsustainable as the network diameter, traffic volume, number of nodes all increase in the presence of moderate node mobility.

Some researchers have proposed using geographic position-based forwarding, rather than a topological-based approach. In position-based forwarding, besides knowing about its own geographic location, every node also acquires the geographic position of its surrounding neighbours. Packet delivery in general is achieved by first learning the destination position from a location service. This is followed by addressing the packet with the destination position before forwarding the packet on to a neighbour that, amongst all other neighbours, is geographically nearest to the destination. It is clear that in the ad hoc scenario, forwarding only by geodesic position could result in situations that prevent the packet from advancing further. To resolve this, some researchers propose improving delivery guarantees by routing the packet along a planar graph constructed from a Gabriel (GG) or a Relative Neighbour Graph (RNG). This approach however has been shown to fail frequently when position information is inherently inaccurate, or neighbourhood state is stale, such as is the case in many plausible deployment scenarios, e.g. due to relative mobility rates being higher than location service update frequency.

We propose *Landmark Guided Forwarding* (LGF), an algorithm that harnesses the strengths of both topological and geographical routing algorithms. LGF is a hybrid scheme that leverages the scaling property of the geographic approach while using local topology knowledge to mitigate location uncertainty. We demonstrate through extensive simulations that LGF is suited both to situations where there are high mobility rates, and deployment when there is inherently less accurate position data. Our results show that Landmark Guided Forwarding converges faster, scales better and is more flexible in a range of plausible mobility scenarios than representative protocols from the leading classes of existing solutions, namely GPSR, AODV and DSDV.

Contents

Chapter 1

Introduction

1.1 Background

IEEE 802.11 access technology is one of the most popular means of wireless communication. Operating at an unregulated 2.4 GHz frequency spectrum, the IEEE 802.11 is capable of providing a useful data transfer rate of up to 54 Mbps, within a radio range of 250 metres. Most importantly, IEEE 802.11 clients support two modes of communication: the infrastructure mode and the ad hoc mode.

When configured in infrastructure mode, a mobile client can communicate with a wireless access point, using it as a relay not only to the wired network, but also to other IEEE 802.11 clients sharing the same access point. Besides extending high speed Internet access to mobile users, wireless infrastructure mode can also be used with Mobile IP [C.Perkins & Mobility (2002)], and has great potential to provide users with seamless connectivity.

In contrast to the infrastructure mode, the IEEE 802.11 ad hoc mode allows mobile clients to communicate directly with one another, without the need for any infrastructure. This not only provides a low cost peer-to-peer inter-networking solution, but also allows mobile clients to connect with one another spontaneously on an ad hoc basis. Many researchers have recognised that this underlying capability for fast setup of infrastuctureless networks has potential to provide self-organising multi-hop services, even in challenging environments.

1.2 Purpose of Research and Methodology

There are many challenges relating to the design of wireless mobile ad hoc networks, amongst which the most prominent is probably the search for appropriate routing algorithms. Of course, there are numerous other problems such as security [Stajano & Anderson (1999)] [Hubaux *et al.* (2001)] and power management [Feeney & Nilsson (2001)] [Ramanathan & Rosales-Hain (2000)] [Xu *et al.* (2001)], some impacting the routing design space too[Sundaresan *et al.* (2004)].

The aim of this research is the design of a new routing algorithm that combines the desirable properties from existing schemes; the evaluation of this algorithm in a number of realistic network scenarios; and the comparison of its performance with existing algorithms. The results will show that the new algorithm can mitigate position uncertainty better than the position-based forwarding algorithm, and that it is more robust to a wide range of realistic mobility patterns

than its main competitors.

In general, many researchers use network simulation to evaluate routing algorithms before deploying them on a testbed for validation. Simulation allows repeatable experiments in controlled environments, which is important for algorithm evaluation, although network simulation can only provide a simplified model of the real environment. In contrast, radio propagation typically requires real-world experimentation to fully evaluate, rather than simulation. In addition, simulation also allows an algorithm to be tested with a large set of network system parameter settings, within a reasonable period of time.

The main goal of our simulations is to compare our algorithm with existing position-based forwarding schemes. Our methodology is to test the algorithms' network performance against parameter settings that may have effects on position uncertainty such as: velocity; network size; inherent position-service inaccuracy; and various realistic mobility patterns. The performance metrics of interest that are reported for each routing protocol, with a variety of values for the above mentioned parameters, are:

- *Delivery ratio* The packet delivery ratio is the ratio between the number of application layer packets received by the receiver, and the number of application layer packets generated by the sender. The packet delivery ratio is a commonly used metric to compare the reliability of routing algorithms, effectively representing their ability to provide connectivity over a longer period.

- *Routing Overhead* The routing overhead is the total number of routing packets sent during the simulation [Broch *et al.* (1998)] [Maltz *et al.* (1999)] [Nain *et al.* (2004)]. The routing overhead is for comparison of scalability of different algorithms, and effectively measures the efficiency of each algorithm.

- *Average Packet Delay* The average packet delay is the average latency for delivered packets during the simulation. The average packet delay reflects the expected latency for the routing mechanism to transfer a packet between the sender and the receiver. This measures the efficiency of the routes delivered by an algorithm.

In order to have a general understanding of the performance for our proposed algorithm, we first benchmark the hybrid algorithm, comparing it both with existing topological-based, and with position-based routing algorithms, without any position inaccuracy, in chapter 2. We then compare the hybrid routing algorithm with a position-based routing algorithm, with a range of position inaccuracy, taken from settings in the literature on location services, in chapter 3. Finally, we compare the hybrid routing algorithm with position-based routing algorithm, with a range of more realistic mobility patterns, taken from recent research results, in chapter 4.

1.3 The Impact of 802.11 Mac Layer

In 802.11, the *Media Access Control* (MAC) sub-layer uses *Carrier Sense Multiple Access* with *Collision Avoidance* (CSMA/CA) protocol. CSMA/CA reduces the probability of collision by the broadcasting of signalling handshakes between the sender and receiver before allowing data to be transferred through the shared wireless medium. Essentially, before putting the data onto

the wireless channel, the sender broadcasts a *Request to Send* signal frame to the network. Having heard the RTS designated for itself, the receiver replies with a *Clear to Send* signal frame to acknowledge the reservation. When the signal is sent, any other node receiving this signal should back off from using the shared wireless medium for a random period of time. This in fact allows the sender to have uninterrupted access to the wireless channel for a brief moment to transfer the data to the receiver. In addition to reducing the contention at the MAC layer, this four-way RTS/CTS/DATA/ACK handshake also resolves the well known hidden node and interfering node issues.

Research on the capacity of ad hoc wireless networks has identified that this handshake protocol is unsuitable for multi-hop packet delivery [Li *et al.* (2001)] [Ray *et al.* (2003)] [S.Xu & T.Saadawi (2001)]. The findings indicate that sharing a single radio channel for sending and receiving is disruptive to multi-hop wireless packet forwarding. For example, a forwarding node needs to compete against the same wireless channel with its last hop as well as its next hop before it can put the packet out onto the shared wireless channel. Although this seems to be an orthogonal issue that might be able to be solved by a cross layer solution using multiple radio channels with a single MAC layer protocol [Sundaresan *et al.* (2004)], it can also be argued that the design of routing protocol does have a direct interaction with the MAC layer CSMA/CA protocol. For example, the more frequent the routing protocol updates its routing table, the less likely it would be to get a link failure caused by node mobility. However, what emerges is that an increase in the number of updates would also induce more overhead in the MAC layer, which is undesirable for the CSMA/CA protocol and disruptive to wireless multi-hop forwarding.

1.4 Ad Hoc Routing Protocols

Many wireless ad hoc wireless routing protocols have been proposed over recent years [Haas & Pearlman (2001)] [Perkins & Royer (1999)] [C.Perkins & P.Bhagwat (1994)] [D.Johnson & J.Broch (2001)] [Park & Corson (1997)] [Vutukury & Garcia-Luna-Aceves (2001)]. Amongst those that have been proposed, many are designed to deliver packets for general ad hoc scenarios, while others are designed for specific applications. For example, protocols such as gossip-based routing [Li *et al.* (2002)] and epidemic routing [Vahdat & Becker (2000)] are designed to disseminate messages for emergency purposes.

An early survey paper [M.Royer & Toh (1999)] categorised ad hoc routing protocols as table driven or source driven. In general, table driven protocols proactively gather topological routing information while source driven protocols reactively discover a route or routes to the destination as requested by the source. Pro-active routing protocols such as Destination Sequenced Distance Vector (DSDV) [C.Perkins & P.Bhagwat (1994)], pro-actively exchange routing information between neighbouring nodes. The associated routing state and the network traffic overhead per node is $O(n)$, where n is the number of nodes in the network, which does not scale well in large networks. Reactive routing protocols such as Dynamic Source Routing (DSR) [D.Johnson & J.Broch (2001)], Ad Hoc on Demand Distance Vector (AODV) [Perkins & Royer (1999)]) use flooding techniques to discover new routes and repair existing routes.

An alternative approach to basing routing on topology information is to use position-based forwarding, taking advantage of potential knowledge of the physical location of nodes in the network [Gao *et al.* (2001)] [Liao *et al.* (2001)] [Niculescu & Nath (2002)] [Ko & Vaidya (2000)] [B.Karp & H.T.Kung (2000)]. The assumption made by protocols that take this approach is

that every node knows its own geographical position. By limiting the exchange of positional information to be only between adjacent nodes, the state and network overheads per node are reduced to $O(u)$, where u is number of adjacent nodes. Position-based forwarding protocols in general use greedy forwarding schemes to propagate packets from a source to a destination. Greedy forwarding provides no guarantees about successful packet delivery even if a path exists between source and destination. A level of message delivery guarantee was first introduced in the FACE routing algorithm [Bose *et al.* (2001)]. The FACE-2 algorithm involves a traversal of the whole face of a planar graph, using the right hand rule until the destination is found, and the packet is routed along this path. GPSR [B.Karp & H.T.Kung (2000)] combines greedy forwarding with the FACE routing.

Greedy Perimeter Stateless Routing (GPSR) [B.Karp & H.T.Kung (2000)] is a wireless routing protocol where each node makes a local decision based on the locally available routes via immediately connected neighbours in order to forward packets towards a destination. *Greedy* forwarding is the process of choosing the next hop route based on the geographic location of the destination. A greedy forwarding decision is made based on the connectivity of neighbouring nodes and the most direct geographic forwarding path available. Upon reaching an area where greedy forwarding is not available, the algorithm switches to *Perimeter* routing, also referred to as *Face* routing, which allows the packet to be forwarded around the perimeter of a region; an example of such a region would be a void. This provides a recovery mechanism to avoid dead-end routing situations and locates an alternative path based upon the same geographic routing principles. Since only local routing state is required at each node, GPSR is able to scale to a larger number of nodes. In addition, as the effects of GPSR state changes is limited to the local area, it can adapt quickly to a fast changing MANET topology.

The robustness of GPSR is derived from its ability to dynamically switch between alternative routing algorithms in order to select a different route and thereby avoid dead-end forwarding situations. The efficiency of the GPSR routing protocol does however depend upon the accurate reporting of geographic coordinates by individual nodes. Location accuracy is a common problem for all geographic ad hoc routing protocols since all the mechanisms that might typically be used for identifying the geographical coordinates of a node have an associated degree of error. Even GPS devices, one of the most accurate and most widely available technologies, cannot guarantee perfect results, and suffers from additional errors when the absolute line of sight is obstructed by buildings or subject to other causes of interference.

An important component of an ad hoc network that uses location based routing is an element called the *location service*. In order to make a routing decision, the source node must initially identify the current location of the destination node. This is accomplished using a location service which stores an ID-to-location mapping for all nodes within the system. When a node moves, it must update its mapping within the location service. There is no clearly defined mechanism for providing such a service, and in all cases the performance of such a service is typically assumed to be an orthogonal issue with respect to the ad hoc routing protocol definition.

In the case of GPSR, it is assumed that such a service exists, and that all nodes have access to the relevant information at all times. In this thesis, we do not address any performance issues relating to the availability of the location service, nor the overhead associated with inserting or reading records from the database: in effect we are making the same assumptions as GPSR.

1.5 Problems with Existing Mobile Ad Hoc Routing Protocols

Many papers report that existing topological based protocols do not scale well at maintaining up-to-date connectivity in the presence of node mobility. For example, the periodic distance vector updates used in *proactive* DSDV [C.Perkins & P.Bhagwat (1994)] cannot handle situations when the topology changes much faster than the periodic update frequency [Broch *et al.* (1998)]. A naive workaround might be to increase the periodic update frequency for scenarios that entail high mobility rates. However, an increase in update frequency imposes more load on the links, and may easily reduce the overall throughput. Moreover, the size of the routing table increases with the number of nodes in the network. It is apparent that an increase in the size of the routing table could also have a negative impact on the forwarding lookup process, as well as causing more, and larger updates on the channel.

In other topological routing research, a solution to the scalability issue in the proactive approach is advocated. Instead, a *reactive* approach is employed, that has no routing overhead costs, when there is no data packet to be forwarded [Perkins & Royer (1999)] [D.Johnson & J.Broch (2001)]. This approach uses flooding to discover and establish paths between a source and a destination, before data packet can commence. Once established, the path is maintained by soft state, and any broken path is repaired by looking up the cached route, or by re-flooding in the local area. Since the path is established on an on-demand basis, any protocol using this approach is more likely to have a more up-to-date state and therefore is more robust to node mobility than proactive schemes.

There are however a few trade-offs when using a reactive routing protocol. For example, the sender needs to wait for a path to be established before proceeding to packet delivery. More importantly, the flooding technique generates huge routing overheads to the network and is very likely to degrade the packet delivery when more data traffic or a larger network diameter is introduced in the presence of moderate mobility condition [Pei *et al.* (2000)] [Li *et al.* (2001)].

Having perceived the scalability limits for topological-based routing, researchers have proposed using geographical location as the basis for the forwarding algorithm [B.Karp & H.T.Kung (2000)] [Bose *et al.* (2001)] [S.Basagni & Woolward (1998)] [Giordano *et al.* (2001)]. This showed promise, although not long after GPSR was published [B.Karp & H.T.Kung (2000)], other research demonstrated that both inherent inaccuracy in reporting of nodes' positions by the location system (e.g. due to location service technology limitations), and inconsistency between the current and last reported destination position (e.g. due to rapid mobility) could both have catastrophic impacts on the effectiveness of position-based forwarding [Kim *et al.* (2004)] [Son *et al.* (2004)]. Research showed that the integrity of the baseline greedy forwarding component, and the correctness of the planar graph constructed for the face routing component, could be compromised by such position inaccuracy or inconsistency [Kim *et al.* (2004)]. In other work, research also showed that the performance of position-based forwarding deteriorates when node mobility increases [Son *et al.* (2004)].

Further research suggested the use of additional messages between neighbours to verify the correctness of the planar graph construction phase [Seada *et al.* (2004)] [Kim *et al.* (2005)]; however, there is no proposed solution to the incorrect greedy forwarding problem. In fact, authors agree that the proposed mutual witness verification scheme could also sometimes incorrectly create extra cross links [Seada *et al.* (2004)]. In order to resolve this subsequent problem, another

researcher proposes the mutual witness scheme to be used together with an even more complex messaging handshake, that is capable of detecting and gracefully removing incorrect/excess cross links from the planar graph, generated during a race condition [Kim *et al.* (2005)]. Such complex messaging, however, does come with a price, requiring additional processing cycles, and incurring more overhead in the network. More importantly, verification of this method has only been carried out for *static* sensor networks, and it has not been shown to be viable for *mobile* scenarios.

In summary, designs for existing topology-based protocols require discovery and maintenance protocols to maintain a global optimal path between a sender and receiver. On the other hand, designs for position-based forwarding protocols rely on accurate, consistent and timely geographic information to be able to provide good connectivity for packet delivery.

It is known that maintenance of global state for optimal routing is at least costly, and probably unsustainable in typical mobile ad hoc network scenarios. Although it is possible to provide reasonably accurate geographic co-ordinates in some mobile environment, such technology is still often too expensive for many deployment situations. Apart from the position accuracy problem, there are still many other problems in maintaining a distributed up-to-date view of nodes' geographic positions in dynamic ad hoc environments.

Above all, the use of periodic neighbourhood updates for topology-based or position-based routing is not an efficient way to maintain useful forwarding state for packet delivery. In order to provide a neighbourhood with useful forwarding information, a node that is more likely to encounter a change in the local topology should update its neighbourhood more frequently than a node that is less likely to encounter a change in the local topology.

We find position-based forwarding comes close to providing a scalable packet delivery solution. Unfortunately, it falls short of considering system tolerance in its design. Yet like many other researchers, we are inspired by the efforts in the position-based forwarding approach. In particular, we find the greedy forwarding component powerful, and it seems to offer the way forward for achieving robust packet delivery with efficient use of network resources.

This leads to our intuition that we need to combine somehow the best features of topology-based and position-based forwarding algorithms in a hybrid, adaptive manner. In the next section, we briefly introduce Landmark Guided Forwarding in a little more detail, which is designed to achieve that. The rest of the thesis is then concerned with detailing the design of this protocol, and exploring its performance in scenarios that reveal the comparative performance to existing protocols in depth, and support the claimed advantages of LGF.

1.6 A Brief Overview of the Design of Landmark Guided Forwarding (LGF)

Learning from those problems found in related work, we then choose to design our protocol with following components.

Restrictive Hybrid Route Advertisement In order to sidestep the overheads of maintaining global optimal stage for packet delivery, at the same time as mitigating position inaccuracy, we propose a radical change from existing MANET protocol design. Rather than maintaining a large and unreliable global topological routing table or only geographical

information of one hop neighbours, each node of LGF protocol updates both topology and position states only to other neighbours within a local topological area. This not only reduces the size of the routing table but also cuts down the overall routing overheads.

Local Optimal Routing As for the forwarding algorithm, we incorporate the greedy component as a driving force for path exploration when an optimal topological path to the destination is not available. In essence, since every node only has localised hybrid states, the routing algorithm applies optimal topological routing only if the destination is within the local topological area. Otherwise, it identifies a neighbour that is the closest to the destination and then applies local optimal routing to this neighbour. Our emphasis of this approach is to use only local hybrid state for packet delivery. This effectively reduces the packet forwarding lookup process and allows the protocol to scale up to large networks.

Path Exploration Unlike position-based forwarding that relies on a stable planar graph, constructed for guaranteed delivery, our protocol uses soft state, together with a source path to achieve a systematic geographic depth first search of the entire connecting graph.

Adaptive Neighbourhood Update Our adaptive update uses the distance to the furthest one hop neighbour to regulate the neighbourhood update frequency. The greater the distance between a node and its furthest one hop neighbour, the more frequently the node updates its surrounding neighbours. This is based on an assumption that the larger the distance between two nodes, the more likely the furthest hop is breaking away and therefore the update frequency should be increased in order to maintain state consistency.

Link Failure Recovery Unlike existing protocols that re-queue the packet when link error is encountered, our protocol drops the packet when link error is encountered. We find the re-queue strategy is purely opportunistic and in general is less likely to gain any advantage when mobility is high or network diameter becomes wider.

1.7 The Contributions

Our contributions are comprised in the next three chapters, as follows.

- Chapter 2 introduces Landmark Guided Forwarding: Our contributions in this chapter are the design of LGF, as well as benchmarking of LGF with DSDV, AODV and GPSR protocols. We evaluate the protocols with 50 nodes as well as 100 nodes with various mobility conditions.

- Chapter 3 is on the impact of position uncertainty on position-based routing: Our contributions in this chapter include a discussion of how the various components of LGF are used to mitigate this position uncertainty. This is followed by simulation results of the LGF and GPSR protocols with a variety of mobility and position uncertainty conditions.

- Chapter 4 is on the impact of mobility patterns on different routing algorithms: Our contributions in this chapter include an early assessment of properties of the Random Way Point, Freeway, Manhattan and Reference Point Group Mobility patterns, and the use of them as a basis for simulation experiments. We use the NS2 simulator to compare LGF with GPSR, with a variety of mobility parameter settings for the different patterns, and a range of different position uncertainty conditions.

Chapter 2

Landmark Guided Forwarding

2.1 Introduction

In this chapter, we describe our new mobile ad hoc network routing protocol, *Landmark Guided Forwarding* (LGF). LGF is designed to have the following properties in comparison with prior MANET protocols:

- Increased global resilience to incorrect device positioning information
- Lower average routing state maintenance
- Lower network routing overhead

Landmark Guided Forwarding only requires that every node maintains a small amount of topological information about its neighbours within a localised area. Routing is achieved by using locally optimised algorithms, requiring lower network overhead. If the packet destination resides within the local area, it is routed using the shortest path algorithm. Otherwise, when the destination resides outside the local scope, it is routed towards a geographically determined optimal *Landmark* node. Unlike position-based forwarding schemes such as GPSR [B.Karp & H.T.Kung (2000)], LGF does not rely upon the establishment of a globally optimal path across landmark nodes, but leverages locally available topological routing information at each stage (a use of the classical *late binding* idea), thereby increasing the resilience to inconsistent device position information, and lowering the overall system vulnerability to position errors [Kim *et al.* (2004)].

In the remainder of this chapter we state the assumptions we have made while developing LGF. We follow on to describe LGF in detail before describing how we simulated LGF in different scenarios. Finally, we summarise the results we obtain, and conclude this chapter by suggesting some possible future extensions to this work.

2.2 Assumptions

We make a number of assumptions in LGF. Firstly, we assume that every node knows its own geographic position. This is not an unreasonable assumption since it is feasible to gather position information from GPS or another positioning system. Since LGF does not require high precision,

short range distance measurements from Bluetooth devices or via IEEE 802.11 based ranging systems such as the Intel Place Lab system [LaMarca *et al.* (2004)] are suitable alternatives to a GPS based system. Secondly we assume a distributed location service like the Grid Location Service [Li *et al.* (2000)] is available for use by LGF. Lastly we assume circular radio propagation area.

2.3 Protocol Description

Ad hoc networks rely on nodes in the network to relay packets between a source and a destination on behalf of their peers. As a packet flows between the source and destination, LGF calculates the locally optimal path to the destination and applies the shortest path to the destination if it is within range of local area. In cases where the destination is not within the local area, it applies local optimal routing to the node that is the geographically closest to the destination. The protocol is iterative. Once the packet is forwarded, it will uncover a new set of neighbours and another local optimal next hop towards the destination. Using this technique not only effectively unlocks the scalability constraints associated with global optimal routing as used by existing MANET protocols but also allows routing to be more adaptive to the ever changing MANET topology. As the approach taken by LGF only requires advertisement of topological and geographical information to a node's neighbours that are within a few hops, it localises state dissemination and reduces the overall load on the network. In essence, these properties allow LGF MANETs to be extended to a wider environment.

In this section, we present the various algorithms that form Landmark Guided Forwarding. The protocol consists of seven components, namely: restrictive hybrid route advertisement, adaptive route advertisement, link failure maintenance, next hop selection algorithm, path exploration, dead-end detection and loop avoidance. We describe each of these in turn in the later sections.

2.3.1 Restrictive Hybrid Route Advertisement

Table 2.1 lists the metrics used by LGF, AODV, DSDV and GPSR. As shown in the table, DSDV uses topological information such as the *Node Identifier*, the *Next Hop* and the *Hop Count*, to establish the shortest path between a sender and receiver, while AODV route discovery also uses similar metrics, in the packet header to achieve the same goal. In the case of GPSR, each node exchanges its position information and identity with immediate neighbours. LGF not only uses node position and topological states, but also includes the velocity and a time-stamp for its adaptive neighbourhood update and forwarding algorithms.

LGF exchanges both topological and location information within a limited radius of only a few hops. For *Restrictive Hybrid Route Advertisement*, we state that each node's neighbours include all nodes within a topological area defined by the perimeter P. For each neighbour node j within P, node i maintains its position, x_j, y_j, and additional information as a routing entry RE_{ij}, in the routing table RT_i. A routing entry RE_{ij} is given below:

$$RE_{ij} = \{j, TimeStamp_j, NextHop_{ij}, HopCount_{ij}, \{x_j, y_j\}, \{\dot{x_j}, \dot{y_j}\}, SeqNo_{ij}\}$$

Where j is a globally unique node identifier of all nodes within P, and the $NextHop_{ij}$ is the identifier of an adjacent node to which a packet should be forwarded in order to reach *destination* which is $HopCount_{ij}$ hops away. The route entry also includes a time stamp $TimeStamp_j$. This

	LGF	**DSDV**	AODV	**GPSR**
Neighbour Identity	√	√	√	√
Time Stamp	√			
Next Hop	√	√	√	
Hop Count	√	√	√	
Node Position	√			√
Node Velocity	√			
Sequence Number	√	√	√	

Table 2.1: Protocol metrics

allows the protocol to determine the current position of the neighbour with velocity and position information. A sequence number $SeqNo_{ij}$ is associated with each entry to ensure timeliness. This is achieved by allowing only advertised updates with higher sequence number to replace the earlier entries. The position and velocity of the destination j, are $\{x_j, y_j\}$,$\{\dot{x}_j, \dot{y}_j\}$ respectively. These attributes are used by the forwarding algorithm to resolve a locally optimal path when the destination address dst_p of a packet p is not in RT_i, $\forall j, dst_p \neq j$.

The current protocol implementation is simple and does not include acceleration into the route entry. It is possible that this parameter can be used together with position and velocity in a more advanced protocol algorithm in the future. This, however, may lead to very complex design issues when position and velocity inaccuracy are considered.

For the purposes of this dissertation, LGF is implemented and evaluated for a constant perimeter, P, with the radius being not less than 2 hops. Future work discussed at the end of the dissertation could be to extend this to give the protocol a further view of topology, albeit with the concomitant increase in protocol overhead. However, the evaluations in this and the subsequent two chapters show the value of the simple scheme, as it stands.

This n-hop radius perimeter limits route updates from one node to be contained within n hops. As shown in Figure 2.1, a 2 hop radius scheme allows a node to receive updates from all neighbours within 2 hops. Essentially, the size of the route table increases when n increases. As for the routing overhead, it depends on the number of neighbours in the hop radius, as well as the size of the route entry. Of course, with the maximum frame size of IEEE 802.11 being roughly 1500 bytes and a route entry size is 30 bytes, the routing overhead in terms of packet count would remain about the same level as long as the number of neighbours within the hop radius is not greater than 50.

In general, a larger hop radius scheme can detect a dead-end or loop much earlier than a smaller hop radius scheme. Therefore it is more likely to avoid sub optimal paths while delivering the packet to the destination. However, a larger hop count is not always useful for the routing protocol and the route state associated with far away neighbours are more likely to be unreliable. Research from [G.Pei *et al.* (2000) and S.Basagni & Woolward (1998)] have observed the route entries from further neighbours in the mobile scenario are less accurate and proposed to reduce routing overhead by sending less frequent updates to neighbours that are further away.

The current design with its fixed n-hops scheme is simple to implement, but not a good solution for mobile scenarios where the density of network could be dynamic. We envisage that the total degree of connectivity of the neighbourhood could be used as feedback in the design of

a more adaptive solution, but this would be future work and outside the scope of this thesis.

In order to explain the restricted hybrid routing advertisement process, we use an example. Figure 2.1 shows a small ad hoc network scenario where node 3 moves from its central position to a new position in the top right of the network, all other nodes remain stationary. We demonstrate the scheme by comparing the routing tables and the topological view of the network from Node 5's point of view.

In this simple example, we consider P is 2 hops and therefore the routing information does not propagate more the 2 hops from the source. If we look at node 5's routing table, Table 2.2, it should reflect its topological view of the network before node 3 moves away from its initial position, as illustrated in Figure 2.2.

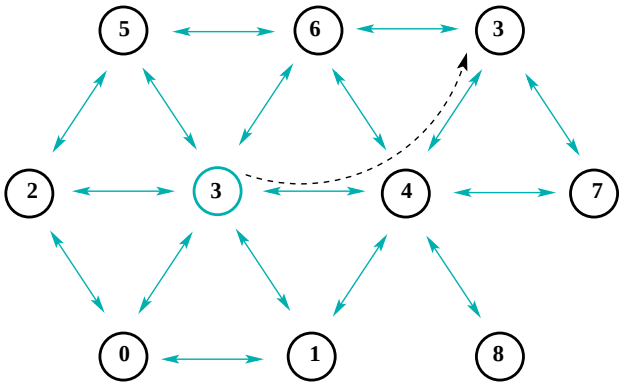


Figure 2.1: Mobility scenario in an ad hoc network

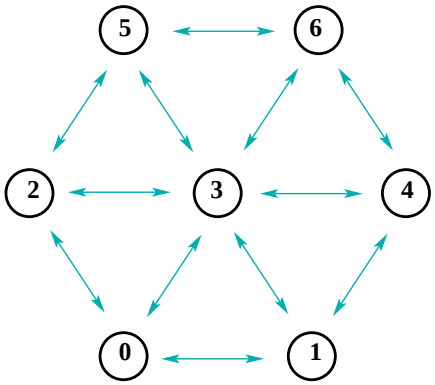


Figure 2.2: Node 5's topological view of the network before node 3 moves

If we re-examine node 5's routing table, Table 2.3, and its topological view of the network, Figure 2.3, after the movement of node 3, Node 1 is now no longer routable using local optimal routing, nodes 3 and 4 are now only routable via node 6 and node 0 is only routable via node 2.

Table 2.2: Node 5's routing table before node 3 moves

Dst	Next Hop	Metric	x	y
0	2	2	300.00	2.00
1	3	2	450.00	2.00
2	2	1	225.00	132.00
3	3	1	375.00	132.00
4	3	2	525.00	132.00
5	5	0	300.00	262.00
6	6	1	450.00	262.00

Table 2.3: Routing table of node 5 after node 3 moves away

Dst	Next Hop	Metric	x	y
0	2	2	300.00	2.00
2	2	1	225.00	132.00
3	6	2	600.00	262.00
4	6	2	525.00	132.00
5	5	0	300.00	262.00
6	6	1	450.00	262.00

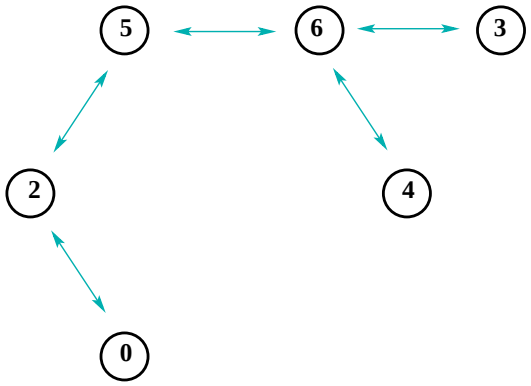


Figure 2.3: Topological view of node 5 after node 3 moves away

In this example, node 3's movement causes the routing algorithm to make the following adjustments to the routing table of node 5, i.e. Table 2.2 is transformed to Table 2.3.

- Remove entries for destinations which have a hop count greater than 2.
- Update location information
- Update next hop and metric information

Specifically, we can see that the entry for node 1 has been removed from routing table of node 5 in Table 2.3. The position of node 3 has been updated and the next hop and metric of nodes 3 and 4 have been updated accordingly.

Routing updates are sent out as part of the route advertisement procedure, which we describe later. These routing updates are processed upon reception by the *RouteUpdate* procedure. The procedure accepts only advertised routes that report link failures ($HopCount = \infty$) or are within a node's topological perimeter ($HopCount \leq P$). This reduces the state propagation and enables this protocol to scale to large networks. The *RouteUpdate* is as follows.

---

procedure *RouteUpdate* ***begin***

if *broadcast route advertisement is received*
for *all neighbours of sender in route advertisement*
if *neighbour's HopCount is within topological parameter or infinity*
invoke *update routing table*
end if
end for
end if

end procedure

2.3.2 Link Failure Maintenance

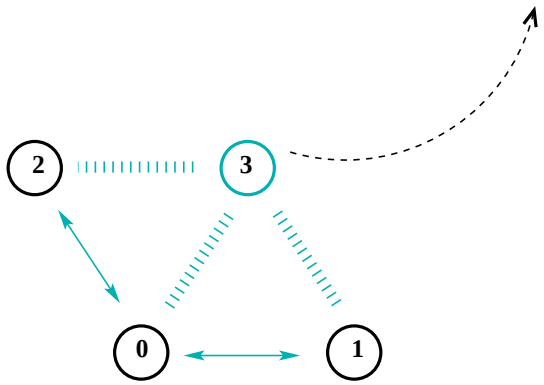


Figure 2.4: State propagation and maintenance

When a node moves out of range of its neighbours, established links are likely to break. Typically a broken link may be detected either by the link layer protocol timing out a connection, or it may be inferred at a higher level through the loss of a periodic broadcast signal which is expected within a predefined time. In our protocol, a node represents a broken link with ∞.

Figure 2.4 illustrates a mobility scenario in which node 3 moves out of range of nodes 0, 1 and 2. Node 1 is initially a neighbour of node 3, and records a route to node 3 with a metric of 1 as shown in Table 2.4. As node 3 moves out of range, the node detects the loss of a link, and updates its table accordingly.

Table 2.5 illustrates the change in routing metrics; the routes to both node 3 and node 2, which originally travelled via 3, are set to ∞. Node 1 subsequently broadcasts these routing entries to all its single hop neighbours. Once the routing state has been synchronised in this manner, the node performs a periodic state maintenance process, removing or replacing the

Table 2.4: Routing table of node 1 before link broken

Dst	Next Hop	Metric	x	y
0	0	1	300.00	2.00
1	1	0	450.00	2.00
2	3	2	225.00	132.00
3	3	1	375.00	132.00

Table 2.5: Routing table of node 1 after link broken

Dst	Next Hop	Metric	x	y
0	0	1	300.00	2.00
1	1	0	450.00	2.00
2	3	∞	225.00	132.00
3	3	∞	375.00	132.00

Table 2.6: Routing table of node 1 after state maintenance

Dst	Next Hop	Metric	x	y
0	0	1	300.00	2.00
1	1	0	450.00	2.00
2	0	2	225.00	132.00

entries with ∞ metrics with cheaper routes, as shown in table 2.6.

2.3.3 Next hop Selection Algorithm

With respect to the forwarding algorithm, LGF takes advantage of the geographical position of those nodes that are within each node's topological scope as its basis. Each node i maintains the topological distance $HopCount_{ij}$, position x_j, y_j and other information for every other node j that is within its topological scope. The next hop is selected using the shortest path algorithm to each packet's destination dst_p, if it is found in the set J with $j \in J$. Otherwise, the next hop is determined by LGF.

Figure 2.5 shows a subgraph that demonstrates our forwarding algorithm where the topological scope is limited to 2 hops. If we consider the packet arrives at node 0, the next hop forwarding option is either node 1, 3 or 5. If the destination for the packet is node 4, the next hop is found to be node 3 by applying the shortest path algorithm to the destination. In the case where a packet's destination is not within the coverage of the topological scope, the next hop is chosen by the shortest path algorithm to a landmark node V, where node V is geographically closer to the destination D and topologically further away from i. In the example in Figure 2.6, the next hop is node 1, with node 2 being a temporary landmark node V, if node 2 is found to be closer to the destination than node 4.

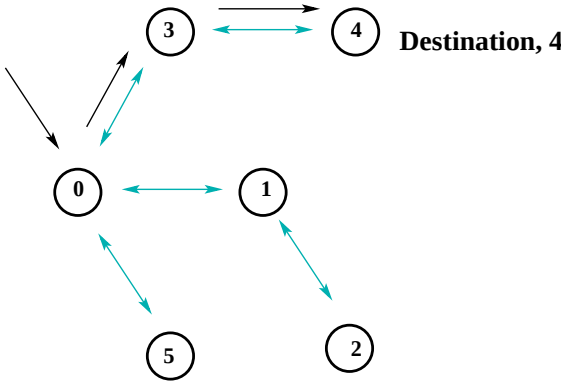


Figure 2.5: Next hop selection - shortest path

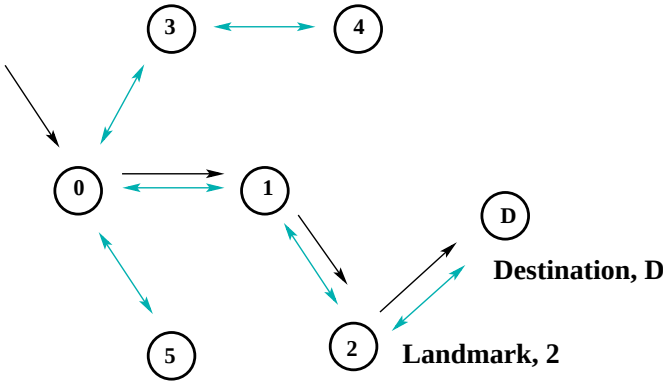


Figure 2.6: Next hop selection - geographic

2.3.4 Path Exploration

Simply forwarding a packet towards its destination position without maintaining any forwarding path history does not provide any facility for preventing the packet from being trapped by a localised loop or dropped due to a routing dead-end, thereby preventing any subsequent back-tracking. The approach adopted in LGF is to include a source path in the packet header, and to maintain soft forwarding state amongst all nodes traversed by a packet. By maintaining a source path in the packet header, LGF provides a trail of forwarding nodes such that in the event that a dead-end is encountered, the packet can be back-tracked until it reaches a node with an alternative path to the destination. In addition, this also enables the algorithm to preserve its loop free property by not selecting a virtual landmark or next hop that is in the source path.

The purpose of maintaining soft-state within the network is to allow LGF to isolate and explore the network systematically. A node temporarily marks a link with the tuple {*Packet_Sequence_Number, Next_Hop, Soft_State_Expiry*}, once it has forwarded a packet along that link. This enables the exploration algorithm to search all available paths and guarantee packet delivery where a path is available between a source and destination.

Figure 2.7 demonstrates how a dead end can be detected while a packet systematically explores a path to the destination. In this scenario, a packet from node S arrived at node 0. Assume the packet's destination is geographically remote and outside the geographical scope of node 0 and the destination is geographically closer to node 2 than node 3, the topological

scope being 2 hops. We denote SP as a sequence of nodes in the packet's source path. At node 0, where $SP = (S)$, we determine the next landmark node, according to our next hop selection algorithm, as node 2. Based on the shortest path algorithm, the next hop node chosen to forward the packet towards node 2 is node 1.

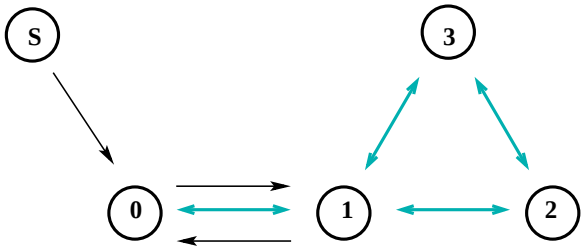


Figure 2.7: Dead end detection and roll back

When the packet arrives at node 1, $SP = (S, 0)$. It becomes apparent that the only node that is 2 hops away from node 1 is S. However, since S is found in the source path SP, the algorithm considers S to be an invalid landmark. With no available landmark and packet's destination not within the topological range, the path exploration detects that the packet is moving towards a dead-end and retracts the packet back to node 0. In this example, node 0 established soft-state when the packet was forwarded from node 0 to node 1 and likewise node S had established soft-state when the packet was forwarded from node S to node 0. Retracting back to node 0, the packet's source path SP is shortened to *(S)*. At this point, the path exploration is aware that the link between node 0 and node 1 has already been visited. Since there is no forwarding path available, the packet is rolled back to node S. With no other link available at node S, the path exploration has exhausted all searches and drops the packet.

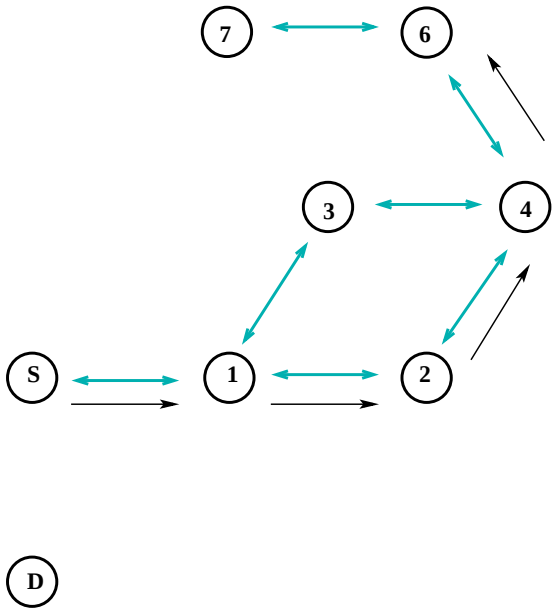


Figure 2.8: Loop avoidance

Figure 2.8 shows a subgraph that demonstrates how a loop is avoided while a packet explores a path to its destination. In this scenario the topological scope is 2 hops and the source node is

S. The destination D is not directly connected to any node in the subgraph. Based on our next hop selection algorithm, the packet at node S identifies node 2 as its Landmark node. Following the shortest path algorithm to node 2, the packet is directed towards node 1. Subsequently, the packet is forwarded to node 2 with Landmark node 4.

The same process is repeated when the packet moves from node 2 to node 4 with node 3 as its respective Landmark node. When the packet arrived at node 4, it found $SP = (S, 1, 2)$ with both node 1 and 7 at its topological range, i.e. within 2 hops of node 4. With node 1 in its source path, the algorithm provides only one option of forwarding towards node 6 with node 7 as the Landmark node. This effectively avoids the creation of a loop between $1 \rightarrow 2 \rightarrow 4 \rightarrow 3$.

2.3.5 Duplicate Loop Exploration

It is possible that the loop avoidance algorithm could cause a packet to revisit the same loop from different directions. In this example, we use two hops restrictive hybrid advertisement as an example to describe this scenario with a subgraph in Figure 2.9. We denote S as the source node and D as the destination node which is not connected to any node in the subgraph. In addition, we define d(a,b) as a function that calculates the distance between node a and b.

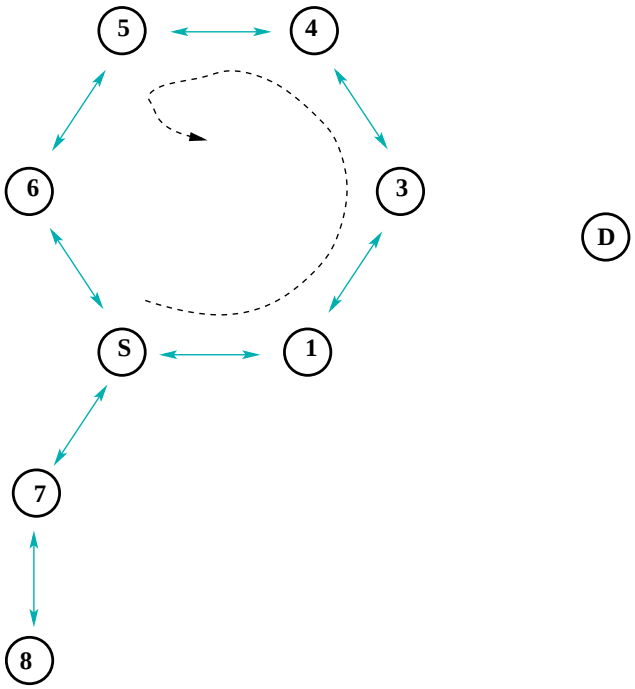


Figure 2.9: Duplicate loop exploration: Forward direction

Our forwarding algorithm first identifies node 3, 5 and 8 as possible landmarks. Since $d(D, 3) < d(D, 5) < d(D, 8)$, the algorithm selects node 3 as the landmark node and forwards the packet to node 1. Upon arriving at node 1, it uncovers node 6, 4 and 7 as a new set of potential landmarks. The algorithm selects node 4 as the landmark node and forwards the packet to node 3 since $d(D, 4) < d(D, 7) < (D, 6)$. Similarly, this depth first search process repeats until the packet arrives at node 5 with a path sequence $SP = (S, 1, 3, 4)$. In addition to the source path, the algorithm maintains soft-state for visited link, namely: $S \rightarrow 1$, $1 \rightarrow 3$, $3 \rightarrow 4$

and $4 \rightarrow 5$.

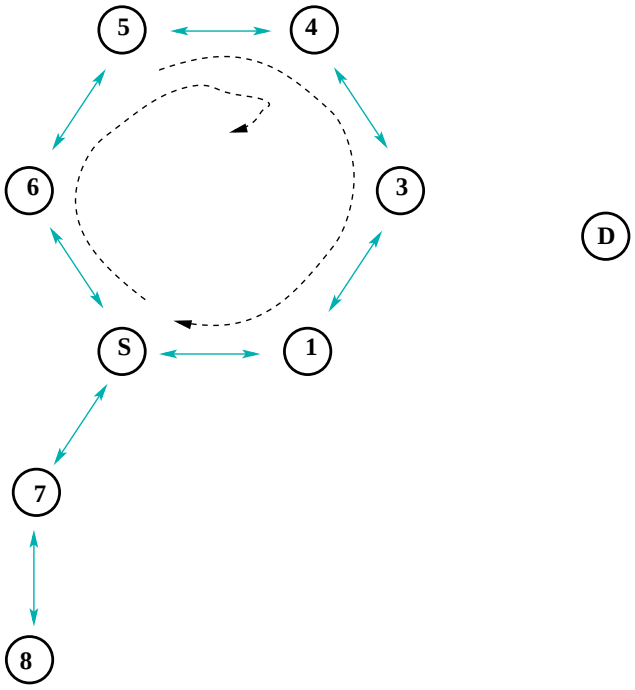


Figure 2.10: Duplicate loop exploration: Reverse direction

Upon arriving at node 5, the algorithm detects only node S is available as a landmark for geographic depth first search. However, since node S is found in the source path, the algorithm detects a cycle between $S \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ and avoids looping by re-tracking back to the previous hop. Figure 2.10 shows the packet re-tracking with a reverse path sequence $5 \rightarrow 4 \rightarrow 3 \rightarrow 1 \rightarrow S$. Likewise, it also maintains soft-states for visited links $5 \rightarrow 4$, $4 \rightarrow 3$, $3 \rightarrow 1$ and $1 \rightarrow S$.

Upon returning back to node S, the source path reduces to null. As the algorithm has not visited link $S \rightarrow 6$ and the source path did not contain any past history of visited loop, the packet continues to explore the loop in a reverse direction with a path sequence $S \rightarrow 6 \rightarrow 5$. At node 5, the algorithm detects link $5 \rightarrow 4$ has been visited by the packet and performs a re-track sequence $5 \rightarrow 6 \rightarrow S$. At node S, the algorithm now finds both link $S \rightarrow 1$ and $S \rightarrow 6$ have been visited. It then continues to search the destination without returning the loop.

This scenario is unique and not likely to occur. First, there has to be a perfect loop where the packet has no other alternate edge and has to backtrack to the entry point of the loop. Then, the surrounding neighbours and the destination have to be in the right condition for the packet to re-enter to the loop from another direction. For the example in Figure 2.9, $d(8, D)$ has to be greater than $d(5, D)$ for the packet to re-enter back to the loop. Below is a proposal for future extension of the protocol to handle such scenario.

In order to resolve this issue without destroying the existing search algorithm implementation, we propose to keep the loop path sequences in the packet header and enable this information for use only after the packet exited from the loop. For the example in figure 2.9, the packet preserves the loop path sequence $S \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ in the packet header when a loop

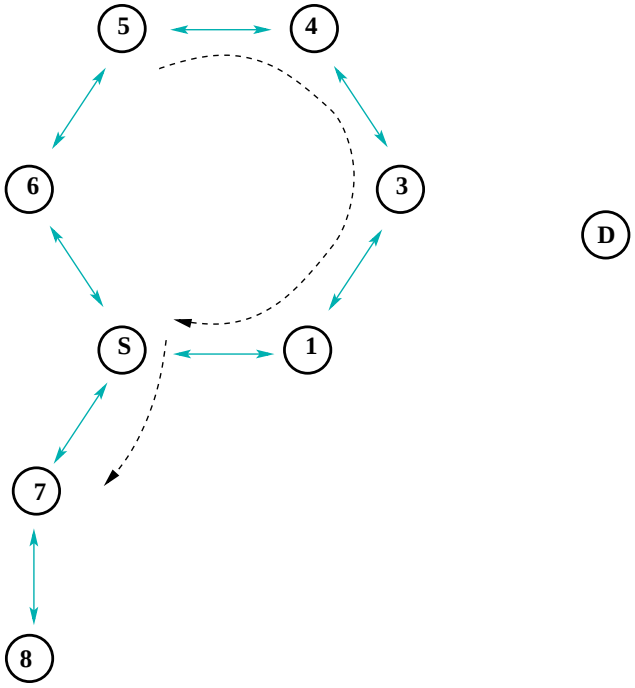


Figure 2.11: Solution for duplicate loop exploration

is detected at node 5. This information is marked as locked by a bit in the packet header. It is then unlocked for the forwarding algorithm when the packet is re-tracked back to node S in Figure 2.11. This prevents the forwarding algorithm from forwarding this packet back to the loop more than once.

Implementation of LGF Exploration

Essentially, we implement a *Forwarding* function that achieves following:

- Apply the shortest path algorithm if the packet's destination is within the routing table
- Switch to Landmark Guided Forwarding if the shortest path algorithm can not be applied
- Backtrack when a dead end is detected
- Avoid packet looping
- Systematic exploration of all possible path with depth first search algorithm

---

function *Forwarding(packet)* ***begin***

$route\_entry \leftarrow NULL$
$route\_entry \leftarrow found\_in\_routetable(packet)$

if$(route\_entry \neq NULL)$
invoke *Mark visited link*
invoke *Push node ID to the source path*

return *entry* \\ *return the shortest path entry to the destination*
end if

route_entry ← *Get_Landmark_Entry*(*packet*)

if(*route_Entry* ≠ *NULL*)
invoke *Mark visited link*
invoke *Push node ID to the source path*
return *entry* \\ *return the shortest path entry to the landmark node*
end if

route_entry ← *Get_Reverse_Entry*(*packet*)
invoke *Mark visited link*
invoke *Pop node ID out from the source path*
return *entry* \\ *return the shortest path entry to the previous hop*

end function

The *Forwarding* function uses *found_in_routetable*(*packet*) to extract an entry from the routing table if the packet's destination can be found in the routing table. If the destination can not be found in the routing table, the algorithm uses *Get_Landmark_Entry*(*packet*) to obtain an entry designated for a suitable landmark node. In the case where no valid landmark node is available, the algorithm uses *Get_Reverse_Entry*(*packet*) to obtain any route entry that allows the packet to re-track back to the previous hop. The returned route entry is then used by another routine to move the packet to the selected next hop. As shown in the pseudo code, the algorithm pushes a node identity into the source path when it can progress closer to the destination. Otherwise, it pops out a node identity from the source path before it retracts back to the previous hop. Besides this, the algorithm also marks the visited link before forwarding the packet to the next hop.

---

function *GetLandmarkEntry(packet)* ***begin***
route_entry ← *NULL*
min_distant ← ∞

\\ *The routing table is a c++ attribute accessible by this method*
for *all routing entries in the routing table*
if *(found_In_Softstate(packet, entry.next hop)*
continue \\ *skip this for systematic exploration*
end if
if *(found_In_Sourcepath(packet.source_path, entry.dst)*
continue \\ *skip this to avoid looping*
end if

if *(entry.metric == topology_range)*
now = get current time
lapsetime = *now* − *entry.timestamp*
$X_{CurrentPos} = entry.x + lapsetime \times entry.vx$

$Y_{CurrentPos} = entry.y + lapsetime \times entry.vy$
$distant = \sqrt{(X_{packet.dst} - X_{CurrentPos})^2 + (Y_{packet.dst} - Y_{CurrentPos})^2}$
if *(min_distant > distant)*
min_distant = distant
route_entry = entry
end if
end if
end for
return *route_entry*
end function

As shown in the *Get_Landmark_Entry* function, verifying that a forwarding path is loop-free is done by ensuring the destination of the route entry *entry.dst* does not match any node identifier of the packet's source path *packet.source_path*. In addition, visited links are isolated by soft-state to allow systematic path exploration. In order to detect dead-ends earlier in the forwarding path, the function only accepts Landmark nodes that are at the *topology_range*. In a situation where multiple Landmark nodes are available, the function chooses the node that is closest to the destination.

---

function *Get_Reverse_Entry(packet)* ***begin***
route_entry $\leftarrow NULL$
previous_hop $\leftarrow$ *get_previous_hop(packet.soft_path)*
\\ The routing table is a c++ attribute accessible by this method
for *all routing entries in the routing table*
if *(entry.metric* $== \infty$*)*
continue
end if

if *(previous_hop == entry.nexthop)*
route_entry = entry
break
end if

end For
return *route_entry*
end function

The *Get_Reverse_Entry* function returns any routing entry which has the next hop equals to the previous hop. First, it uses the *get_previous_hop*(*packet.soft_path*) function to extract the node identifier of the previous hop from the source path. Once the identifier of the previous hop is acquired, it uses this identifier to search through the routing table for an entry that can be used for re-tracking the packet to the previous hop.

2.3.6 Adaptive Route Advertisement

One key feature of LGF is its ability to regulate the restrictive hybrid update frequency according to its interconnectivity with other adjacent nodes within its radio range. The essence of this feature is to associate update frequency with the likelihood that the furthest neighbour moves out of the wireless range. One possible heuristic is to use mobility prediction as an input for adaptive route advertisement. But this method can lead to more complex issues when position and velocity errors are considered. We therefore present it as a future work in section 2.8.4. In this thesis, our heuristic is to associate update frequency with the distance to the furthest node within the radio range. The greater the distance between node and its furthest adjacent node, the more frequently the node must send out its routing updates. This increase in the rate of state propagation enables the network to converge much faster when adapting to changes in the surrounding network connectivity.

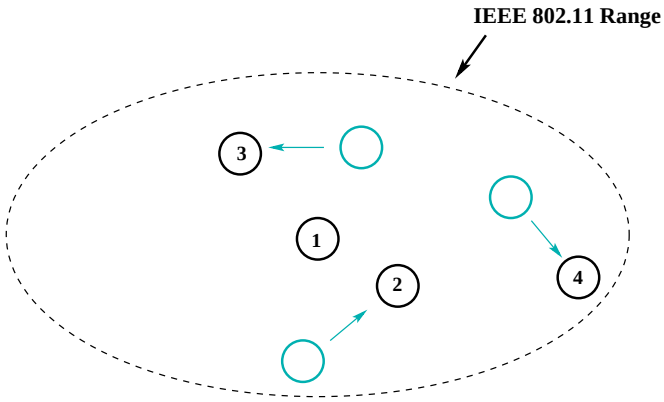


Figure 2.12: Relative displacement of all one hop neighbours

Figure 2.12 shows a mobility scenario of node 1 and its adjacent neighbours. In this example, we show current and last known positions of adjacent node 2, 3 and 4. Essentially, the last known position is the position advertised by adjacent node and the current position is calculated from last advertised position, node velocity, together with the time difference between current time and the time stamp of the route entry from the respective neighbouring node.

The distance between node 1 and its adjacent nodes are such that $d(1,2) < d(1,3) < d(1,4)$, where d(a,b) is the function calculates the distance between node a and node b. For *Adaptive Route Advertisement*, the distance between a node and its furthest adjacent node, $Max\_Distance$, is used to regulate the frequency of route advertisement. In this scenario, the $Max\_Distance$ is d(1,4).

Once the $Max\_Distance$ is determined, the algorithm uses a mapping function to obtain respective expiry for the next route advertisement. The mapping function is graphically represented is figure 2.13.

The *Mapping* algorithm defines the *expiry* time to be inversely proportional to $Max\_Distance$ when $Min\_Threshold < Max\_Distance < Max\_Threshold$. Within this range, the expiry time is determined by following equation.

$$Expiry = ((Radio\_Range * Tuning\_Factor)/Max\_Distance) \tag{2.1}$$

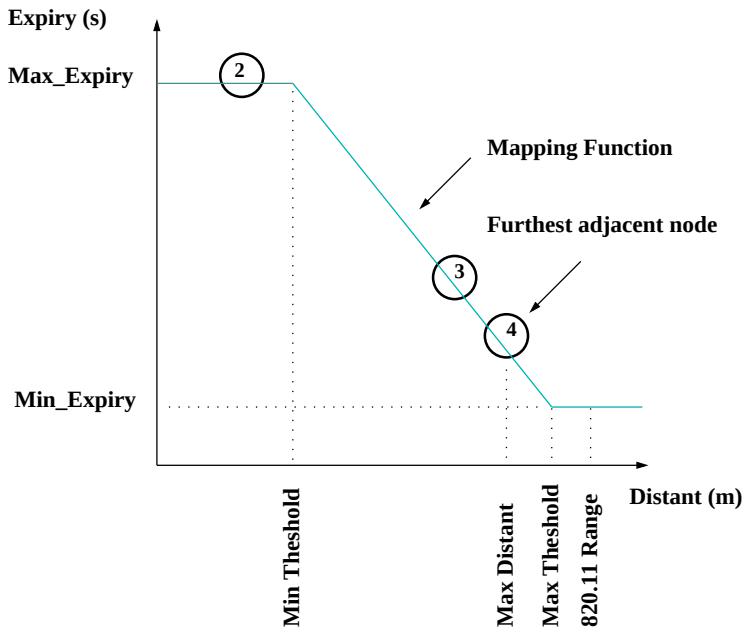


Figure 2.13: Mapping of furthest adjacent node to the expiry time of routing entries.

In order to avoid excessively short lived routing advertisements when link failure is imminent, the minimum expiry time is applied when the maximum distance is greater than $Max\_Threshold$. Conversely, the maximum expiry time is applied to reduce the frequency of updates for near adjacent nodes when $Max\_Distance < Min\_Threshold$. The Tuning_Factor is used as a sensitivity adjustment of this mapping algorithm. An increase in Tuning_Factor increases the expiry. Although this increase in expiry can reduce the total number of broadcast route updates in the network, it can introduce more stale state and reduce the routing performance. Currently the mapping algorithm is simple to implement, we envisage the *Received Signal Strength Indication* (RSSI) of the neighbours can be used together with geographic information for a better algorithm in the future.

Implementation of Adaptive Route Advertisement

The function *GetDistance* is used to calculate an approximate distance between a node and its adjacent neighbour using Pythagoras's theorem. *GetDistance* uses an estimate of the current location of an adjacent node which is derived from a combination of the last advertised position and the node's velocity, together with *lapsetime*, the time difference between current time and the time stamp of the route advertisement received from the respective adjacent node.

---

function *GetDistance(node, neighbour)* ***begin***

$now \leftarrow getcurrenttime$
$lapsetime = now - neighbour.timestamp$

$XPos \leftarrow (neighbour.x + (lapsetime * neighbour.dx))$

$Xdist \leftarrow (node.x - XPos)^2$
$YPos \leftarrow (neighbour.y + (lapsetime * neighbour.dy))$
$Ydist \leftarrow (node.y - YPos)^2$
$dist \leftarrow \sqrt{Xdist + Ydist}$
return *dist*

end function

---

function *GetMaxDistance(node)* ***begin***

$Max\_Distance \leftarrow 0$
$Distance \leftarrow 0$

\\ *The routing table is a c++ attribute accessible by this method*
for *all routing entries in the routing_table*
if *neighbour is one hop away*
$Distance \leftarrow GetDistance(node, routing\_entry)$
$Max\_Distance \leftarrow \max(Max\_Distance, Distance)$
end if
end for
return $Max\_Distance$

end function

The *GetDistance* function is used by *GetMaxDistance* function that iterates through all the one hop neighbours in the routing table to find the node which is furthest away.

---

procedure *Mapping(Max_Distance)* ***begin***

$expiry \leftarrow 0$
if$(Max\_Distance < Min\_Threshold)$
\\ *upper bound of expiry*
$expiry \leftarrow Max\_Expiry$
else if$(Max\_Distance > Max\_Threshold)$
\\ *lower bound of expiry*
$expiry \leftarrow Min\_Expiry$
else
$expiry \leftarrow (radiorange * tuningfactor)/Max\_Distance$
end if
return *expiry*

end procedure

In the NS2 implementation, a *node* object is an object encapsulating the property of a node. This includes current position and prevailing velocity of a node. As shown in below *RouteAdvertisment* procedure, the $Max\_Distance$ is obtained from GetMaxDistance(node) before the

Mapping function works out an expiry for respective *Max_Distance*. The function uses *tuning factor* to adjust the sensitivity of the mapping. In addition, it also controls the upper and lower bounds of the expiry. This expiry is then used as a variable component for scheduling of the subsequent route advertisement.

---

procedure *RouteAdvertisement* ***begin***

$Max\_Distance \leftarrow 0$
$expiry \leftarrow 0$

\\ *node is an c++ object that encapsulated the property of a node*
\\ *The property includes current position and prevailing velocity*
$Max\_Distance \leftarrow GetMaxDistance(node)$
$expiry \leftarrow Mapping(Max\_Distance)$
\\ *Create a route_entry and schedule for a route advertisement*
invoke *CreateRouteAdvertisement*
invoke *scheduleNextRouteAdvertisement(expiry)*
end procedure

2.4 Proof of Loop Freedom Property

Theorem 2.4.1 *LGF ensures data packets never flow in loops.*

Proof LGF establishes a source path in the packet's header while the packet transverses through intermediate nodes to the destination. A packet traverses through a path P = $(X_1, X_2, ..., Xn)$ before it is received by the destination X_n and would encode a source path $SP = (X_1, X_2....X_{n-1})$, where X_i are nodes along the path for $i = 1, 2, ..., n$.

Suppose that there is a loop in P. Then a packet must have been forwarded to one of the nodes twice in order to form a loop. But the algorithm would only select a virtual landmark or use a next-hop that is not in the source path. Therefore, no packet could have transversed through the same node twice. This proof by contradiction shows that LGF is loop free.

2.5 Evaluation of LGF with Existing MANET Protocols

The evaluation has been carried out using the NS2 simulator [ns group at ISI (1989)], with each simulation lasting for 900 seconds. Each node uses the IEEE 802.11 MAC and the physical model models the radius of the radio range as being 250 metres. The simulation uses the random way point model to model node mobility. In all simulation scenarios, each node selects a random destination and moves at a speed uniformly distributed between 0 and maximum node velocity. Upon reaching the destination, the node remains stationary for a configured period (pulse time) before it selects the next random destination and moves on. We use 4 different sets of traffic

patterns. Each traffic pattern uses constant bit rate UDP traffic flows, with 512 byte payloads. The start time for the different flows is uniformly distributed between 0 and 180 seconds with each of the 30 traffic sources sending at the rate of 2 packets per second.

2.5.1 Scenarios with Varying Pause Time

In common with other protocol evaluations, [B.Karp & H.T.Kung (2000)] [C.Perkins & P.Bhagwat (1994)] [Perkins & Royer (1999)] [D.Johnson & J.Broch (2001)], we run several mobility patterns with different pause times at a constant speed. We use 5 different sets of random way point mobility patterns generated with different pause times of 0, 30, 60, 120, 600 and 900 seconds where the maximum node velocity is 15 m/s. These 5 different sets of mobility patterns are then run with 4 sets of traffic patterns to generate 20 sets of simulation results for evaluations. In this set of simulations, the pause time is used to vary the degree of mobility of the simulation. Researchers have shown an increase in pause time reduces the number of connectivity changes [Broch *et al.* (1998)]. This in effect allows us to evaluate the protocols routing performance with different levels of connectivity change. This set of simulations is run in an area of 1500x300 m^2 with 50 nodes randomly placed. The reason for choosing a rectangular simulation region is to ensure longer routes are used between nodes than would occur in a square region with the same node density [Broch *et al.* (1998)].

2.5.2 Scenarios with Varying Node Velocity

In the second set of simulations, we configure the pause time to 0 seconds, i.e. the nodes constantly move, and repeat the simulations with different maximum velocities of 1, 2.5, 5, 7.5, 10, 12.5, 15 m/s. This simulation is run in an area of 1500x500 m^2 with 100 nodes randomly placed. Our key interest in this set of simulations is to evaluate the performance of the protocols with a larger network.

We compare LGF with DSDV, AODV and GPSR using the different simulation scenarios we have just described and we compare the adaptability, performance and overheads of LGF with other MANET routing protocols. Each of the different ad hoc routing protocols has some settings specific to it. We detail these in table 2.7, 2.8, 2.9 and 2.10.

The DSDV and AODV parameters are the default parameters preconfigured in the NS2 package. In fact, these parameters were identical or similar to the parameters used by other research papers [Perkins & Royer (1999)] [Broch *et al.* (1998)]. Likewise, the GPSR parameters are also the default parameters from the GPSR installation package. The GPSR has a feature to allow a broken link to be removed from the forwarding table before enqueuing the packet for an alternate next hop. In our preliminary test with this parameter, we found this feature can not ensure good packet delivery ratio. In fact, redirecting packets can sometimes cause more MAC layer contentions and result in packet drop. In addition, this feature can also cause the average packet delay to increase substantially. It is therefore important to ensure this feature of position-based GPSR algorithm is disabled for the evaluation. As for the case of DSDV and AODV, a timeout mechanism is in place to ensure a packet does not hold in the queue for too long. Our results show the link failure optimisation of DSDV and AODV does not incur a significant amount of packet delay.

In the case of LGF, we use 3 hops hybrid advertisement on 1500x300 m^2 and 1500x500 m^2 simulations. Since the simulation is configured with 250 metres radio range, it is likely to create scenarios that require LGF to use path exploration to deliver the packet to the destination. The *tuning factor*, *max expiry* and *min expiry* are decided by experimentation of different parameters that yield good packet delivery ratio packet without generating excessive routing overheads. These parameters are not optimal but are sufficient to ensure the adaptive neighbourhood update is sensitive enough to handle most situations.

Table 2.7: DSDV specific parameters

Parameter	*Value*
Initial weight settling time	6 s
Periodic update interval	15 s
Number of missed periodic updates before declaring link broken	3
Settling time weight	7/8

Table 2.8: GPSR specific parameters

Parameter	*Value*
Beaconing interval	3 s
Random variation of beaconing interval	0.5 %
Beacon expiration interval	13.5 s
Promiscuous mode	enable
Redirect packet when link failure when link broken	disable
Perimeter mode	enable

Table 2.9: AODV specific parameters

Parameter	*Value*
Lifetime of a route reply message	10 s
Time for which a route is considered active	10 s
Time before route request message is retired	6 s
Time which the broadcast id for a forwarded route request is kept	6 s
Number of route request retries	3
Maximum route request timeout	10 s
Local repair wait time	0.15 s

Table 2.10: LGF specific parameters

Parameter	*Value*
Tuning factor	1.5
Max expiry	15 s
Min expiry	1.2 s
Topological scope	3

2.6 Results

The results are divided into three subsections: performance with varying pause time, performance with varying velocity and path length.

2.6.1 Performance with Varying Pause Time

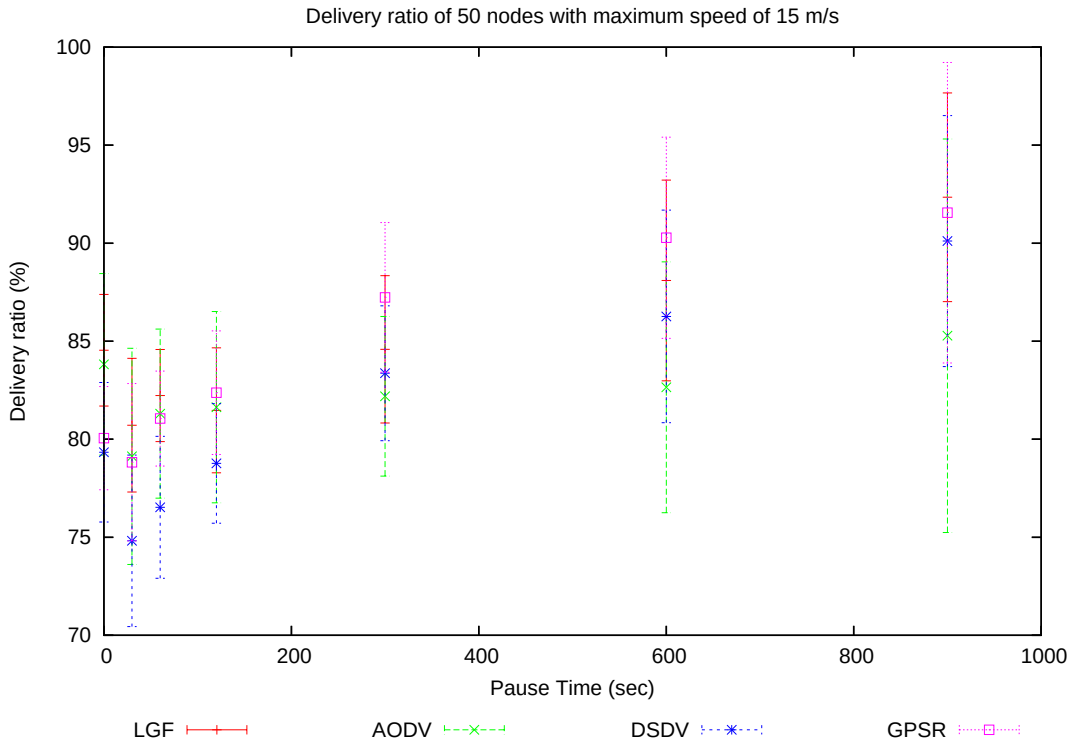


Figure 2.14: Comparison of the packet delivery ratio of the four routing protocols as a function of pause time with the maximum velocity set at 15 m/s

Figure 2.14 evaluates the reliability of packet delivery of the different routing protocols; LGF, GPSR, AODV and DSDV. In general, LGF, DSDV, GPSR and AODV perform better as the pause time used in the random way point model increases. This is because an increase in pause time in effect reduces the mobility of nodes in the network. As a result, the topological states in the network are less likely to become stale and forwarding algorithms are more likely to deliver the packet successfully. However, it is possible that in some cases the random waypoint model can produce a well connected graph with low pause time. Our results show the packet delivery

ratio at 0 pause time is higher than those packet delivery ratio at 30, 60 and 120 second of pause time. Our results show LGF is more robust than other protocols when pause time is configured to 120 seconds and below. This is largely due to the way in which LGF handles the route updates. LGF uses adaptive neighbourhood update to increase the update frequency when link failure is likely to occur. This reduces stale hybrid states from neighbour advertisements and therefore is more robust in handling changes of link connectivities when the pause time is reduced. In contrast, DSDV and GPSR update its neighbour periodically. Therefore they are not able to provide a reliable state for packet forwarding when connectivity change is much faster than the periodic update frequency. Our results show GPSR performs better than DSDV. This is because the GPSR updates its neighbours at an interval of 3 seconds, while DSDV updates its neighbours at an interval of 15 seconds. It is apparent that GPSR is more likely to get a more up-to-date state than DSDV for packet forwarding and therefore is able to perform better than DSDV. As for AODV, it uses a flooding technique or cached route to find a path between a source and destination. Our results show this reactive method is likely to perform better than DSDV when link connectivities are changed at a pause time interval of 120 seconds or less.

Apart from route updates, link failure handling is another factor relating to the reliability of the packet delivery. AODV and DSDV optimise the handling of link failure for stale connectivity, in contrast in LGF and GPSR, it drops packets as soon as it sees the link fail. In the case of LGF, this is a design decision to decrease the average packet delay at the cost of reducing the delivery ratio. We describe this in more detail below.

Both DSDV and AODV, upon notification of link retransmission failure, keep packets in the buffer queue until their route becomes available again. This technique is not described in the published papers about the protocols, but was found to be in the NS2 implementation. In the event of a link retransmission failure, GPSR drops the packet immediately. In LGF, the protocol drops the packet, updates the route entry, and propagates the broken link to other neighbouring nodes.

Our results show that the link failure techniques used by DSDV and AODV are opportunistic. The idea is to keep or redirect the packet when a link retransmission failure is encountered. Although this could increase the packet delivery ratio in some cases when connectivity is stable, in some scenarios such as where there is node mobility and the opportunity of direct or indirect re-delivery are not available, undelivered packets then linger for too long in the output buffer queue and can contribute to a higher average packet delay. Our results show that the protocols rarely gain enough advantage to perform better than LGF and GPSR. In Figure 2.15, we show that the average packet delay on GPSR and LGF is lower as it does not keep and redirect the undelivered packets. In contrast, AODV and DSDV retain the packet for redirection and has the side effect of a higher average packet delay. Our results show LGF and GPSR consistently have a lower latency than other routing protocols. LGF and GPSR achieve this by not holding the packets in the event of link retransmission failure.

Figure 2.16 highlights the communication overheads of the different routing protocols. In LGF, the node that is most likely to experience link failure within the next hop neighbourhood advertises more frequently than nodes that are less likely to encounter link failure. Although the advertisement is restricted to the local scope, LGF in general is sending out more frequent but restricted updates to its neighbours within its local scope. This explains why the overall communication overheads of LGF in this simulation is equally as high as DSDV. Moreover, DSDV monitors the difference between each update and reduces the update frequency when

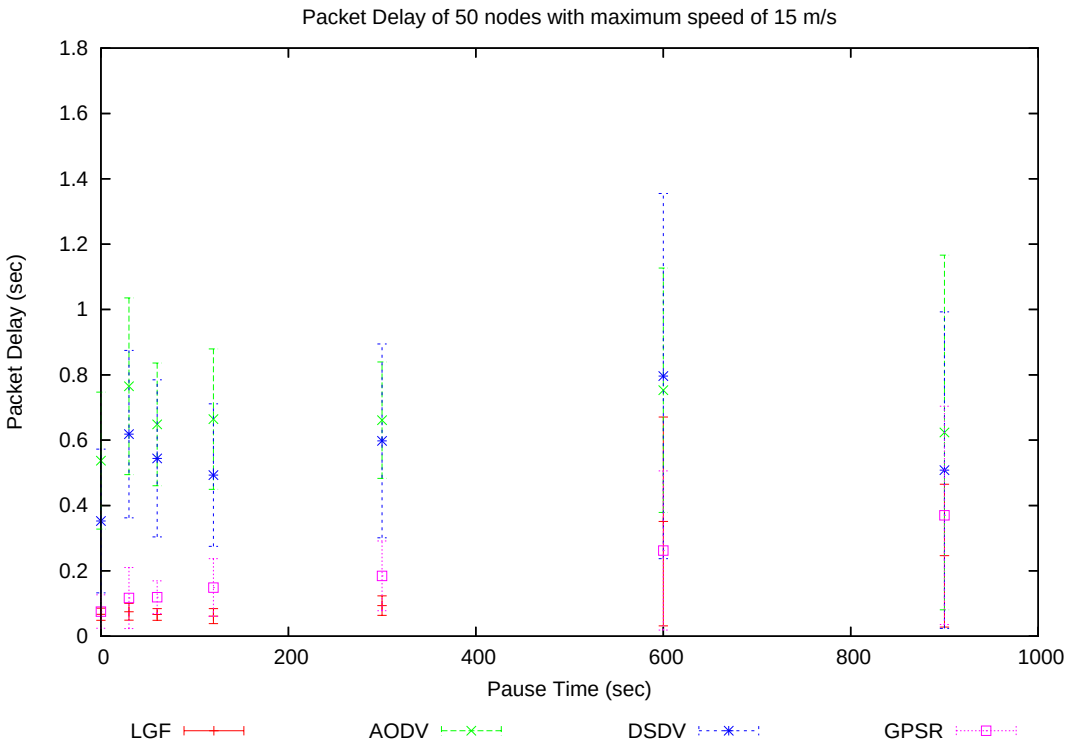


Figure 2.15: Comparison of average packet delay between the four routing protocols as a function of pause time with the maximum velocity set at 15 m/s

the difference between updates is less. Our results show DSDV routing overheads reduce when the pause time increases. When compared with other protocols, LGF has lower communication overheads than reactive AODV or GPSR. The high routing overhead of AODV is mainly contributed by the flooding mechanism used during the path discovery and recovery. In the case of GPSR, it reduces the overheads by piggy backing position information on all out going data packets. Although the existing overhead of GPSR is more than DSDV and LGF, it is possible to reduce the overhead by using an adaptive approach similar to LGF.

2.6.2 Performance with Varying Velocity

In this simulation, we tested performance of a system with 100 nodes over a wider area. Compared to previous simulations, the network diameter is larger, and therefore nodes are expected, on average, to take more hops between the source and destination. Additionally, the density of nodes in this simulation is 133 nodes per km^2 as compared to the previous density of 111 nodes per km^2. With more network overhead introduced as a result of the denser and larger system, it is further anticipated that contention and interference issues experienced in IEEE 802.11 networks could be more critical than previously measured. As a result, the channel capacity of the network is reduced [Li *et al.* (2001)] [Gupta & Kumar (1999)], and consequently the average packet delay in general increases and the ratio of successful delivery decreases compared to previous simulations.

In comparison to other protocols, the results in figure 2.17 indicate that LGF and GPSR are relatively robust and able to achieve a higher measured delivery ratio over a variety of veloci-

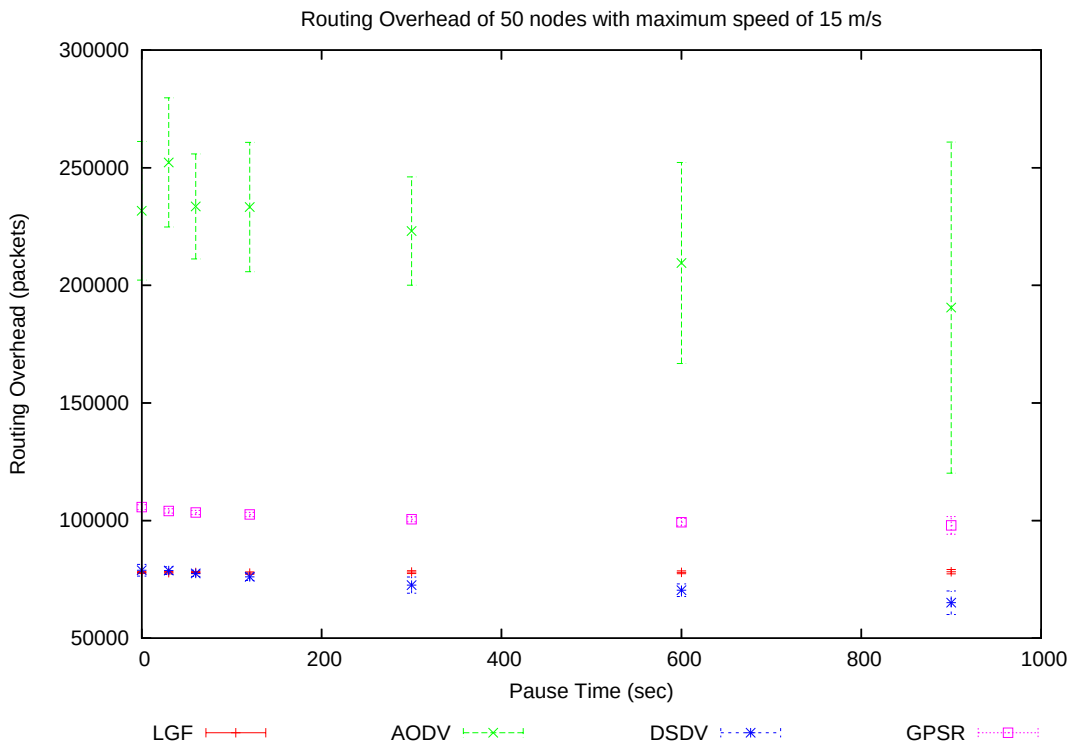


Figure 2.16: Comparison of routing overhead between the four protocols as a function of pause time with the maximum velocity set at 15 m/s

ties. The results show that both LGF and GPSR are more reliable and adaptive to unsettled, dynamic topologies than other protocols. In contrast, the results show DSDV is not able to converge fast enough when tested in the larger network. The results also indicate the flooding method used by AODV is not able to sustain good packet delivery performance in the larger network.

Our results in figure 2.18 show that LGF performs consistently well with respect to routing overhead over a variety of speeds. These results are similar to the previous simulation results. The high communication overheads associated with reactive AODV are the result of the higher number of route discoveries and local repairs AODV is performing.

Compared with earlier results where we used fewer nodes and smaller physical area, the overheads we observed are more onerous than in the previous simulation. Our observations show that the overheads associated with LGF are lower than the other protocol as the number of nodes is doubled from 50 to 100. Because DSDV needs to maintain a global state for all the nodes in the network, its overheads increase in proportion to the number of nodes in the network. In the case of GPSR, every node advertises every 3 seconds which increases its overheads compared to DSDV where nodes advertise every 15 seconds. Although GPSR advertisements are only sent to its one hop neighbours, the results indicate that the higher frequency of GPSR updates can result in higher overheads as the network size increases than would be observed with DSDV in a similar scenario. In contrast, the restricted route update in LGF adapts well to the increased size of the network with the results confirming that LGF's communication overheads scale better than other protocols.

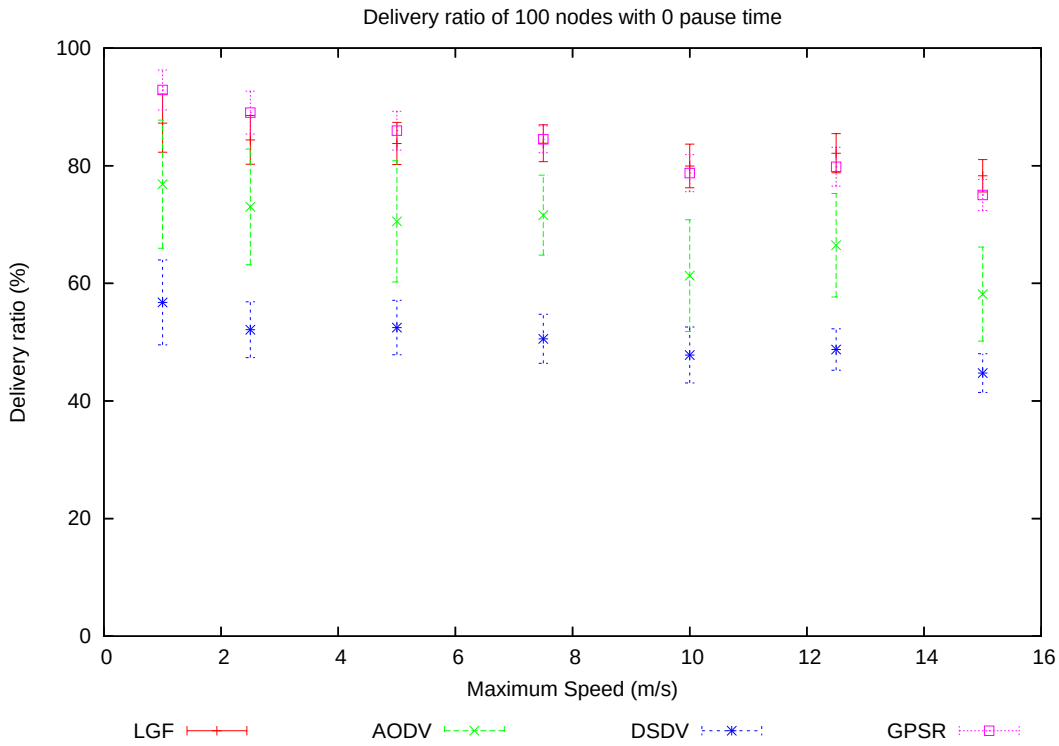


Figure 2.17: Comparison of packet delivery ratio between the four protocols as a function of maximum velocity where the pause time is zero

As shown in figure 2.19, DSDV does not converge fast enough to cope with the changes in connectivity when tested in a larger network. As a result of this, more undelivered packets are held in the queues in the network before the route entry to the destination becomes available.

Our results show the on demand path setup of AODV has a lower average packet delay than DSDV when simulating 100 nodes. This accounts for the performance advantage shown for the AODV local repair scheme in a dense network. If we consider the overall performance of all the protocols on packet delivery ratio, routing overheads and average packet delay. Both LGF and GPSR provide a better overall and balanced performance than other protocols.

2.6.3 Path Length

Figure 2.20 compares the path length for successful delivered packets for each protocol against the ideal shortest path retrieved from the NS2 simulator. The ideal shortest path is the shortest possible path only constrained by the physical radio range. The evaluation was carried out with a random way point mobility model using a 0 second pause time with a maximum velocity of 15 m/s and 50 nodes placed randomly in area of 1500x300 m^2. The results indicate that LGF on average achieves approximately 83 % of optimal path length while GPSR obtains approximately 78 % of optimal path length. Although, theoretically, DSDV is supposed to maintain an optimal path, the slow update interval does not prevent misleading stale state from being used by the packet delivery mechanism, resulting in sub-optimal routing. DSDV only routes approximately 77 % of its packets via the optimal path. Only approximately 55 % percent of AODV's packets are routed by the optimal path. A contributing factor to this is AODV's local repair algorithm,

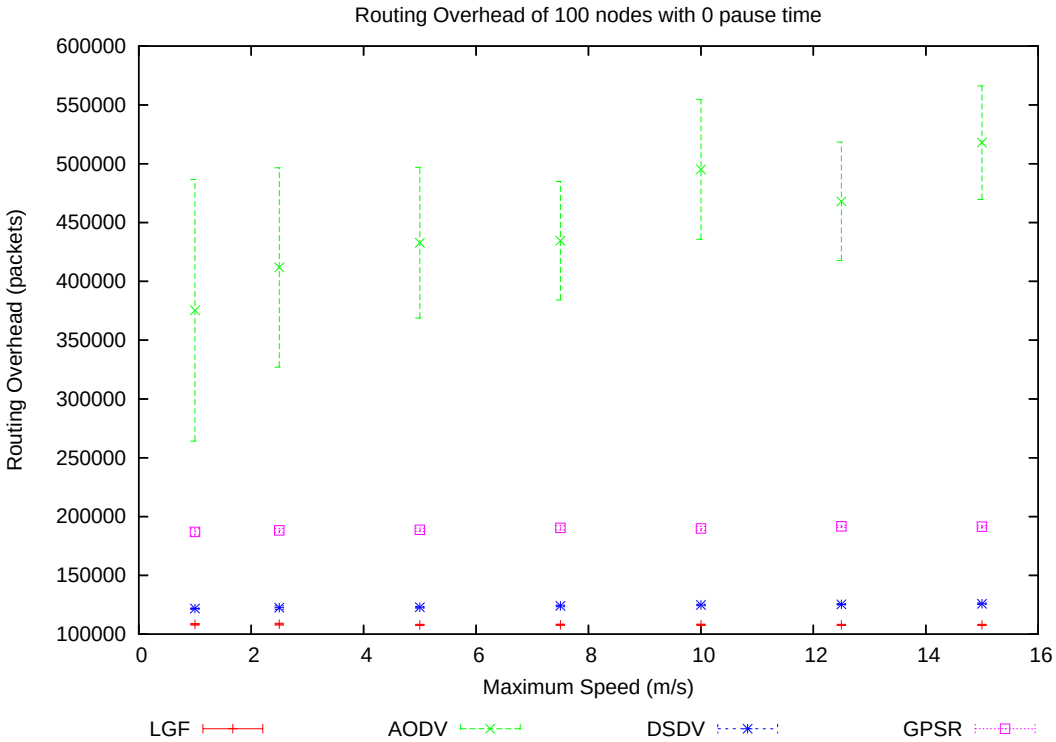


Figure 2.18: Comparison of the routing overheads between the four routing protocols as a function of maximum velocity where the pause time is zero

which fixes broken paths without considering what is the alternative optimal path between the source and destination.

2.7 Conclusion

In summary, we present a hybrid routing protocol, Landmark Guided Forwarding, using restrictive hybrid advertisement at a rate regulated by its connectivity sensitive algorithm, applying optimal routing when the destination is within its topological range, systematically resolves a transient next hop through local optimal resolution when an optimal route is unavailable.

We run simulations with 50 nodes and 100 nodes. The results indicate the overheads of LGF scale better than other protocols when the number of nodes is doubled from 50 to 100. In our performance evaluation with varying pause time, the results show that route optimisations by AODV and DSDV are less likely to gain enough advantage to outperform LGF and GPSR. In fact, the simulation results conclude these optimisation could give side effects and yielding a higher reading in average packet delay. The effect is more pronounced when we simulated it at 100 nodes with slightly a wider network diameter. In the case of GPSR and LGF, it does not need to wait for global topological state to converge and therefore it is able to adapt much faster than topology-based protocols. Our results show LGF and GPSR are able to maintain a steady, swift and reliable delivery even with an increased chance of unstable network connectivity. When comparing the path length with other protocols, LGF has the highest score of optimal routing over other protocols.

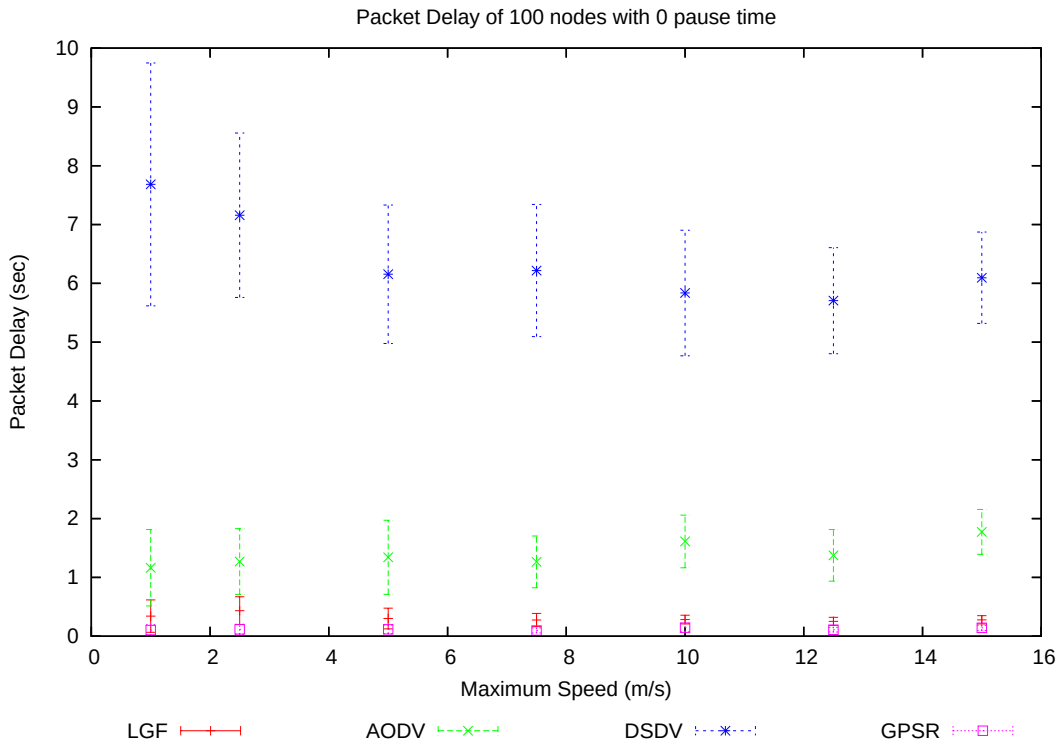


Figure 2.19: Comparison of average packet delay between the four routing protocols as a function of maximum velocity where the pause time is zero

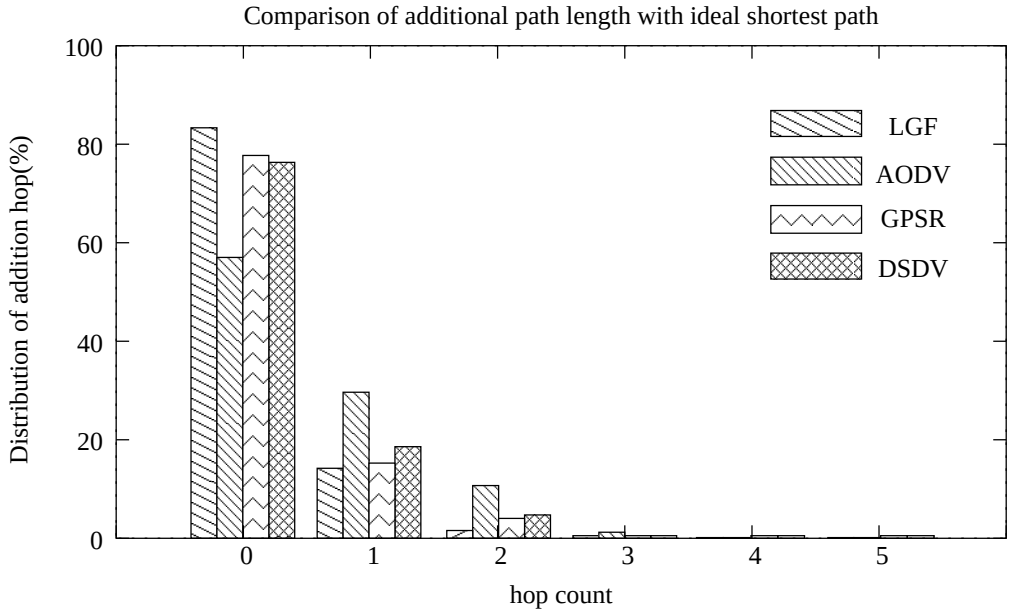


Figure 2.20: Comparison of the average path length for each of four protocols with ideal shortest path

In conclusion, local optimal routing unlocks the constraint of maintaining a globally optimal path, as generally required by existing MANET protocols. LGF is therefore a relatively scalable

and robust protocol with low overheads as compared to other ad hoc routing protocols.

2.8 Future work

2.8.1 Interface with Internet Coordinate Scheme

In future work we wish to investigate using a coordinate system with the Landmark Guided Forwarding protocol to exploit their common goals of reducing routing overheads. The Internet coordinate systems such as Lighthouses [Pias *et al.* (2003)] and Virtual Landmarks [Tang & Crovella (2003)] could be used to supplement the process of selecting the topologically closest node to the destination. LGF could additionally exploit the topological data from the coordinate systems to avoid routing errors when removing edges or nodes that violate the triangle inequality.

2.8.2 Path Optimisation

In the current protocol the landmark node is chosen by its proximity to the destination, this normally results in sub-optimal routing that could be addressed by depositing additional soft-state information in the nodes. This information could include the paths learnt during previous packet deliveries and could be used to enable the path exploration algorithm used for other packets to exploit this knowledge while searching different destinations.

2.8.3 Resilient to Position Errors

We considered the handling of position inconsistencies while we designed the Landmark Guided Forwarding protocol, but the performance of this feature has not yet been discussed. This is the topic for the next chapter.

2.8.4 Advanced Adaptive Route Advertisement

The existing adaptive route advertisement uses the position of neighbours to determine the expiry of the next route update. This method can be improved by considering both the velocity and acceleration of the neighbouring nodes. In fact, mobility prediction has been an interesting research topic [Samaan & Karmouch (2005)][Shen *et al.* (2000)][Akyildiz & Wang (2004)][Bhattacharya & Das (1999)] and handling of position error with mobility prediction is a challenging area that we can explore in the future.

Figure 2.21 shows the position uncertainty model between a node and its neighbour. In this figure, a node is situated at the centre of the big circle with a circular wireless propagation area of R radius. E(v) is the estimated value of velocity, which can be obtained through a mobility prediction mechanism. We assume there is such a system that can provide estimated velocity of a neighbour by incorporating the analysis of real time neighbour mobility data (including position, velocity and acceleration), with other environmental factors such as the map of the local vicinity or statistical log of the nodes mobility in the local area. Given that the distance between a node and its neighbour is r, the time, T_{away}, when the neighbour cross the range of the wireless cell can be determined by equation 2.2 and the lower bound update frequency, $Freq_{est}$, can be determined by equation 2.3.

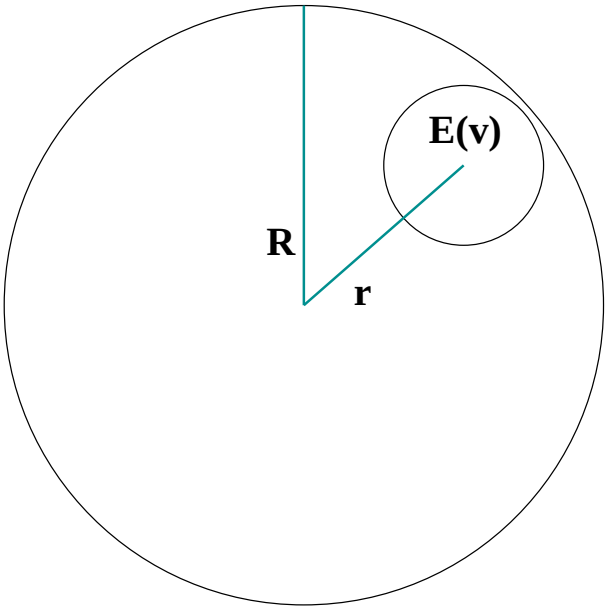


Figure 2.21: position uncertainty model of a neighbour

$$T_{away} \leq t = \frac{R - r}{E(v)} \tag{2.2}$$

$$Freq_{est} \geq \frac{E(v)}{R - r} \tag{2.3}$$

Chapter 3

The Impact of Position Uncertainty

3.1 Introduction

The problems surrounding positional accuracy in GPSR have been uncovered by earlier research [Kim *et al.* (2004)] [Seada *et al.* (2004)] [Son *et al.* (2004)]. It has been shown that there are serious limitations in the robustness of the system when subjected to positional inaccuracy and less frequent updates of location information. Location inaccuracy occurs as a result of the lack of precision of services such as the GPS or base station triangulation in systems such as GSM, IEEE 802.11 or Bluetooth. In this chapter, we first discuss existing location tracking technologies and their accuracy, with a view to discovering appropriate values to use in our later simulation scenarios. We discuss the related work on attempts to solve the problems for GPSR-like algorithms in the presence of inaccurate location data. Then we evaluate the impact of position inaccuracy on position-based forwarding algorithms through our own simulations. This is then followed by an exposition of how LGF is able to mitigate positional inaccuracy, with the results, analysis, conclusions and potential future work.

3.2 Location Tracking Technologies

Providing geographic location information on mobile device has long been an interesting area of research. There have been many technologies built to provide location information for different applications. In this section, we discuss various technologies and their accuracy.

3.2.1 The Active Badge/Bat System

The Active Badge System is a indoor position tracking system built for locating people in the office environment [Want *et al.* (1992)]. In the Active Badge System, a personal tag is designed to emit a unique beacon signal periodically. This signal is then received by sensors installed in a confine area. In addition, a server is used to query and extract the badge sightings from the senors before it is processed to become useful position information for the user.

The Active Badge System uses a Pulse Width Modulated Infrared signal between the badge and the sensor. This signal operates with a range of 6 metres and will not penetrate through office partitions or walls. However, since the signal can only travel through line-of-slight, it requires the user to wear the badge in a position that is suitable for sensors to pick up the signal. Although the Active Badge system is cost effective, robust and sufficient to be used for many

location aware applications, it can only distinguish whether the object is within a confined area.

An ultrasonic location solution, the Active Bat System, has been proposed [Hopper *et al.* (1993)]. It provides a higher level of location and orientation accuracy of an object in the indoor environment. To measure the time-of-flight of the ultrasonic signal, the Active Bat System uses a controller to synchronise the transmitter and receiver at every 200ms. The controller encodes a unique 16 bits address of the selected device on a message before it is sent to the device using radio. Subsequently, the device receives this signal and broadcasts an ultrasonic pulse. The device then switches to sleep mode before it is self-reactivated for the next radio signal.

A matrix of ultrasonic detectors is installed on the ceiling of the office area. Each of these detectors are placed 1.2m apart. To determine the three dimensional position of an object, the system requires triangulation of time-of-flight measurements from at least three detectors. Results shows 95% of measurements are within 3cm of the actual position.

3.2.2 The Place Lab System

The Place Lab System is a location tracking system that provides indoor and outdoor position for mobile devices such as mobile phone, laptop or Personal Digital Assistant(PDA) [LaMarca *et al.* (2004)]. The system leverages the beacon signals from existing widespread wireless technologies such as GSM and IEEE 802.11. In addition to GSM and IEEE 802.11 beacon signals, the system also listens to beacon signal from stationary Bluetooth devices.

Essentially, the system maintains a database that associates beacon signals to geographic positions. After acquiring beacon signals from the vicinity, the Place Lab client looks up the position of the associating beacon location from a local cache map for estimating its current position. The accuracy of the system relies upon the type of beacon signal as well as the density of beacons in the neighbourhood.

The recent results from Intel's PlaceLab project [LaMarca *et al.* (2004)], as shown in Table 3.1, demonstrate that positional accuracy of $13.4m \sim 31.3m$ can be achieved with 100% coverage in urban, residential and suburban by using a combination of IEEE 802.11 and GSM position tracking systems.

Table 3.1: Accuracy and coverage of GSM and IEEE 802.11 position tracking system.

	IEEE 802.11		GSM+ IEEE 802.11	
	Accuracy	**Coverage**	Accuracy	**Coverage**
Urban	107.2 m	**100**%	21.8 m	**100**%
Residential	161.4 m	**100**%	13.4 m	**100**%
Suburban	216.2 m	**99.7**%	31.3 m	**100**%

In fact, the accuracy of the system can be enhanced further. In other related work [C.Komar & Ersoy (2004)], researchers have found the accuracy of position information could be improved to 5 metres by synchronising the measurement of signal strength from three IEEE 802.11 base stations.

3.2.3 The Global Positioning System

The Global Positioning System [Hofmann-Wellenhof *et al.* (1998)] [Kaplan (1996)] is a satellite based navigation system made up by constellation of 24 satellites deployed by the Department of Defence (DoD) of the United States. GPS was originally aimed to provide all weather, round the clock, world wide position services to the military forces and government services, but it has been made available for civilian use without charge by the US government.

Each of these satellites make one complete revolution around the globe every 12 hours. The orbiting satellites are arranged in such a way that at any time, anywhere on Earth, there are at least four satellites available and in the line of sight with the receivers. Essentially, the GPS receiver locates and calculate its distance to these satellites. It then uses this measurement to deduce its location by simple trilateration.

Using trilateration with at least four satellites, the system can pinpoint the location of the receiver in three dimensional space. However, the reception of a GPS receiver normally suffers from some levels of interference. The receiver, in general, can achieve an accuracy of less than 100 metres in the present of interference from noise and bias in the vicinity.

- **Noise**
  - **Ambient Noise**: Interference of radio frequency noise in the surrounding area. Noise from this group could be generated by electronic devices from surrounding area. There are various methods to resolve this. Some use better shielding while others suggest the inclusion of a filter circuitry or signal processor to remove this noise.
  - **Receiver Noise**: Interference of radio frequency noise generated within the receiver. Noise from this group is normally caused by white noise in the electronic component or poor design of printed circuit broad. Most of these problems could be solved by better shielding, grounding and good printed circuit broad routing.
- **Biasing**
  - **Selective Availability (SA)**: Selective Availability is a feature designed to degrade the quality of the positioning system by varying the time parameter in a cycle of few hours. Its intention is to disrupt the measurement so that the civilian user obtains less precise readings from the system.
  - **Skewed Receiver Clock**: Although the system periodically synchronises the receiver quartz clock with satellite atomic clock, there is still a marginal clock skew between the clock synchronisation.
  - **Reflected signal**: Measurements taken from reflected signals could cause the satellite to appear to be further away than its actual location.
  - **Propagation Delay**: Signal propagation delay in troposphere and ionosphere layers could slow down signal propagation. The delay could have a marginal impact on distance measurement.

Biases can normally be corrected by Differential GPS (DGPS) [Halsall (2000)]. The basic idea of DGPS is to reduce the bias errors at one location by taking account of measured bias errors at a known position. It uses a reference receiver or base station to compute offsets for

the signal of each satellite and then uses these offsets to correct the bias errors on other receivers.

There are several surveys evaluate the performance of position accuracy of GPS. In a book by Clarke, the author defines *the rings of accuracy* for GPS position accuracy [Clarke (1994)]. He uses four rings to separate the distribution of GPS position accuracy. In fact, the survey gives a general impression that most positions reported by GPS receiver are more likely to be closer than further away from the actual position of the receiver.

Other research efforts show the GPS position measurements taken from the field can be approximated by normal distribution [Navigation & Position Group at University of New South Wales (1999)] [Wilson (2001)] which is commonly being described as a bell curve as shown in figure 3.1, with mean μ and standard deviation σ defined by the probability function:

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{3.1}$$

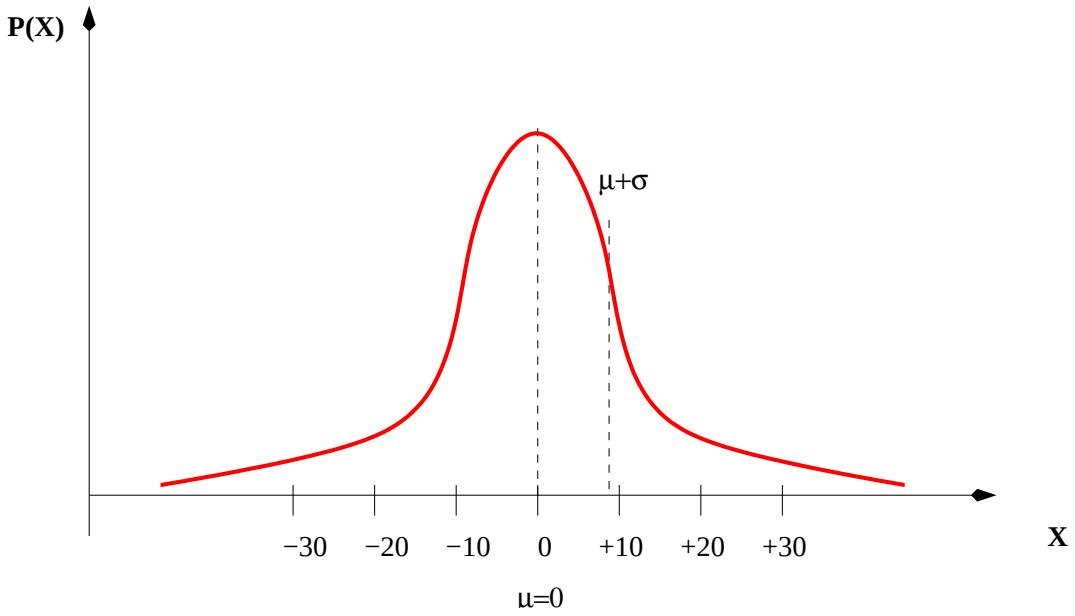


Figure 3.1: The effects of time lapse with position uncertainty

Apart from the above surveys, there are other researchers found the inaccuracy of the GPS readings not only has spatial but also has temporal correlations [El-Rabbany & A.E-S (1994)] [Wanninger.L (1993)] [Glennon & Dempster (2004)]. According to their findings, receivers nearby are more likely to share the same set of satellites as well as expose to the same atmospheric, clock skew, noises and SA conditions. Besides, their results also show the GPS inaccuracy are temporal correlated. This is confirmed by showing that the temporal correlation decreases significantly as the time interval between GPS signal increases.

3.3 Related Work

A number of papers have shown a link between inconsistent positioning and significant negative impact on position-based forwarding protocols. In [Kim *et al.* (2004)], it was shown that the

effects of location inconsistencies on greedy forwarding and perimeter mode routing caused the wrong greedy decision to be made that led to suboptimal routing and looping. In particular it showed that for a significant proportion of the time, the construction of the planar graph was invalid, when the standard deviation of position accuracy was set to 25 metres (with similar simulation scenarios described in the previous chapter). These findings show that the impact of location inconsistencies could compromise the correctness of position-based routing protocols.

In another paper [Seada *et al.* (2004)], researchers identify the problems that cause Face routing to fail and classify them into three main categories: 1) incorrect edge removal, 2) cross-links and 3) inaccuracy in the destination location. The paper only proposes a fix to address the incorrect edge removal problem. It uses additional message exchanges between neighbours to confirm the correctness of the edge removal process. This, however, is not a perfect solution, beside creating more routing overheads, the researchers admit it may add some extra cross-links in some cases.

Another recent paper, [Kim *et al.* (2005)], also proposes similar approach, known as mutual witness. The researchers concur this approach indeed would create cross-links that jeopardise the correctness of the Face routing algorithm. To address this issue, they propose a Cross Link Detection Protocol (CLDP) that uses probe messages to find and remove cross-links from the planar graph. The process involves complex messaging to eliminate a possible race condition while probing and removing a cross-link. We observe the results of both papers were gathered from static sensor network scenarios. So far, there are no results showing this solution could effectively resolve the incorrect edge removal issue as well as removing cross-links in a MANET scenario.

An alternative approach [Son *et al.* (2004)] to the location inconsistency problem has been proposed to use mobility prediction to mitigate the effects of location inconsistency. However, this work considers only the effects of inconsistent positioning caused by the beacon frequency of neighbours and node mobility. It recommends the use of spatial and temporal correlation between nodes, together with information about geographic restrictions to reduce the effects of location inconsistency. It propose Neighbour Location Predication, (NLP) and Destination Location Prediction, (DLP) schemes. These schemes, however, assume that the node knows its transmission range and would not attempt to forward a packet outside this range to an estimated node position. This paper did not model the inconsistencies caused by position accuracy or a delayed location reporting service.

3.4 The Impact of Position Inaccuracy

3.4.1 The Impact of Position Inaccuracy on Greedy Packet Forwarding

Figure 3.2 illustrates a scenario in which an incorrect greedy forwarding decision is caused by device position inaccuracy, incorrectly directing the packet towards a sub-optimal neighbouring route. Figure 3.2 shows two nodes, X and Y which are adjacent to the packet source node S. The packet is to be forwarded to the destination node D which is outside the range of S and therefore must be relayed via the ad hoc routing system. As a result of limited precision in the position tracking system, node Y inaccurately advertises its position to its *surrounding*

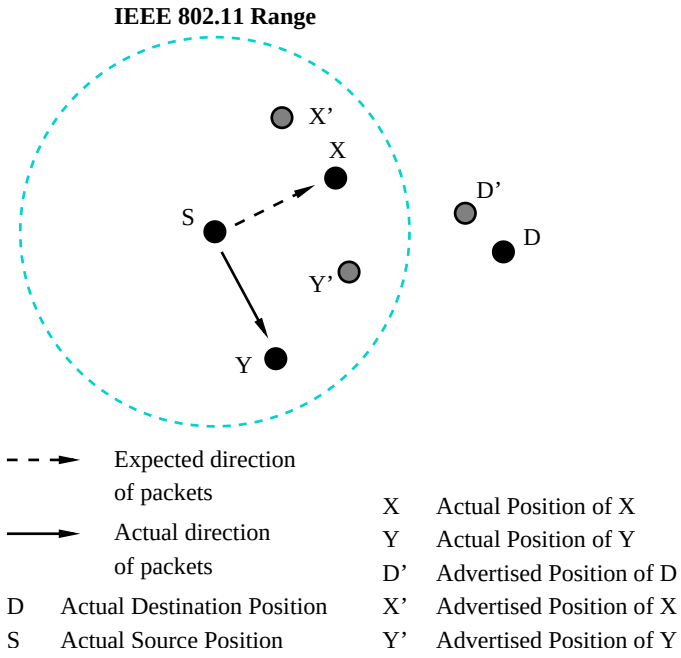


Figure 3.2: False greedy packet forwarding

neighbours as Y' while node X inaccurately advertises its position to its *surrounding neighbours* as X'. Based on the announced location and connectivity of X and Y, S incorrectly assumes that the optimal forwarding route for the packet is via node Y when, in fact, as illustrated in figure 3.2, the optimal path is via node X.

This is a simple example of the impact of position inaccuracy on greedy forwarding; however, it is apparent that when applied throughout a larger system, this inaccuracy may be compounded by incorrect decisions at many locations along a multi-hop path on account of position inaccuracy, thereby lowering the general ability of the system to forward data. We present results to illustrate the impact of position errors on GPSR in section 3.7. An alternative example of the situation where this issue would cause a problem is where the incorrect routing decision caused the packet to be directed incorrectly around a void or into a dead end.

3.4.2 The Impact of Position Inaccuracy on Perimeter Routing

Perimeter routing is a position-based forwarding approach that guarantees delivery of a packet if a path between a source and destination exists. It is used to eliminate the incorrect decision that would otherwise be made through a greedy packet forwarding algorithm causing packets to be forwarded into a dead-end.

In perimeter routing, each node builds a planar graph of all surrounding nodes, where a planar graph infers a table of neighbouring nodes with all *intersecting* intermediate nodes removed. In figure 3.3, a node W is considered to intersect the path between 2 other nodes, U and V if W lies within the shaded circle with $|UV|$ as the diameter of the circle. In this manner, each node constructs a planar graph incorporating a subset of all neighbouring nodes. The purpose of constructing such a graph is to be able to apply a perimeter routing algorithm that will allow forwarding along the graph and has the property that if a route between the source and

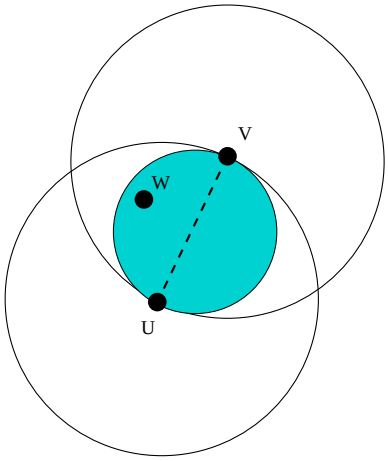


Figure 3.3: Removal of cross edge while planarizing Gabriel Graph

destination exists, even if it is not a direct route, the algorithm will discover the path [Mauve *et al.* (2001)] [Bose *et al.* (2001)].

Two algorithms that can construct a planar graph are the Relative Neighbour Graph (RNG) planarization algorithm and the Gabriel Graph (GG) planarization algorithm. RNG uses the intersection between the two radii of the distance between 2 nodes (in this case U and V). The GG algorithm uses a smaller region encompassing a circle between the 2 nodes U and V as illustrated in figure 3.3.

In this section, we illustrate the vulnerability of the GG planarization algorithm to position inaccuracy causing the algorithm to mis-construct the planar graph and potentially cause incorrect *perimeter* routing decisions as a result of device position inaccuracies. We consider the case of GG planarization algorithm decisions based on inaccurate node location information. The first case illustrates incorrect exclusion of a witness, and the second case illustrates incorrect inclusion of a witness. In both cases we denote W' as the inaccurate position advertised by W.

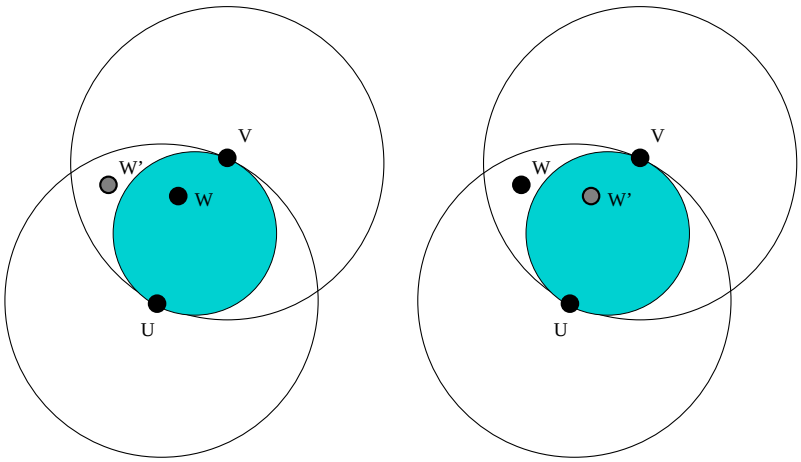
Incorrect Witness Exclusion

In Figure 3.4(a), based on the GG planarization Algorithm, edge UV would not be removed since W' lies outside the inclusion circle between node U and V; however, this decision is in fact incorrect since the real location of W lies within the inclusion zone.

Incorrect Witness Inclusion

In Figure 3.4(b), while W physically resides outside the inclusion zone, W' appears to reside within the zone. The GG algorithm therefore incorrectly includes W and wrongly removes edge UV.

In an article, the researchers have modelled the probability of these issues [Kim *et al.* (2004)]. They studied the problem with different levels of position inaccuracy along with various average distances between U and V. Apart from showing that the probability of false planar graph construction increases with position inaccuracy, the findings also show the probability of incorrect



(a) (b)

Figure 3.4: False Planar Graph Construction

construction of planar increases when the average distance between U and V decreases. The results show the probability becomes 1 with location inaccuracy of 25m and average distance of 50m.

3.4.3 The Impact of Position Inconsistency from Location Service on GPSR

In order for nodes in the network to forward packets to their destinations, position-based forwarding protocols require the source node to encode the destination position into the packet header. Intermediate nodes then extract the destination position from packet header in order to decide on the next hop. To discover the location of the destination node prior to forwarding, the source must first lookup the location using the location service.

One such mechanism might be the Grid Location Service (GLS) [Li *et al.* (2000)], which uses a consistent hash function to distribute node positions across a set of other nodes within a region. In this chapter, we do not consider the mechanism used to retrieve the information, but rather we consider the impact of the slow insertion of location information by the individual nodes into the location service database thereby causing a source node to retrieve incorrect or stale location state relating to the destination.

In Figure 3.5, D" represents the stale and therefore inaccurate state of destination D that was deposited in the location service. As the source node encodes the packet header with an incorrect location for D, it confuses the position-based forwarding algorithm, driving the packet towards an incorrect position. Earlier research [Seada *et al.* (2004)] [Son *et al.* (2004)] has demonstrated that this incorrect state information could cause the packet to loop searching for D at D" without ever actually reaching D.

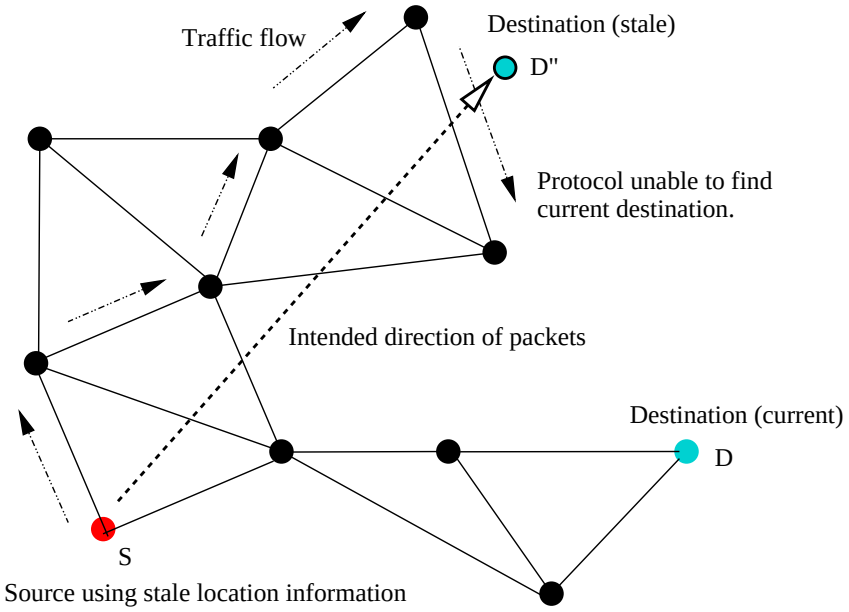


Figure 3.5: Position forwarding with stale destination position state

3.4.4 The Impact of Position Error on GPSR Neighbourhood Update

The GPSR protocol adopts a proactive routing approach. Each GPSR node periodically broadcasts its position to all neighbours within its radio range. In order to reduce the inconsistency between neighbours' current and last updated positions, GPSR use a short time interval to update its neighbours' positions. Although the update is only for neighbours within the radio range, it can introduce considerable network overhead and interference when more load is introduced into the network. To reduce this effect, GPSR reschedules the periodic update and allows location update to be piggybacked onto transiting data packets [B.Karp & H.T.Kung (2000)]. Although this feature can effectively reduce the overheads from periodic updates when load is introduced, it has an undesirable effect when position inaccuracy is considered. Our results in section 3.7 showed the overall routing overhead reduces as the position inaccuracy increases.

In essence, the routing overhead of an ad hoc routing protocol consists of route updates and routing packets [Broch *et al.* (1998)]. The route update is the control packet used by routing protocol to maintain network connectivity. The routing packet is the data packet in transit between a sender and a receiver.

In order to understand the effect of position inaccuracy on GPSR routing overhead, we perform 30 simulation runs with 50 nodes on a region of 1500 x 300 metres square. Our aim is to evaluate the periodic broadcast updates and overall routing overheads with various settings of traffic loads and position inaccuracies. We use 10, 15 and 20 traffic sessions with position inaccuracy of 0, 10, 20, 40, 80 metres. Each session uses 512 bytes payload and sending at a rate of 2 packets per second.

Figure 3.6 shows that broadcast route updates reduce when communication sessions increase. This is because the increase in the number of traffic session generates more data packets onto the network. This, in turn, reschedules the periodic broadcast updates more often and allows

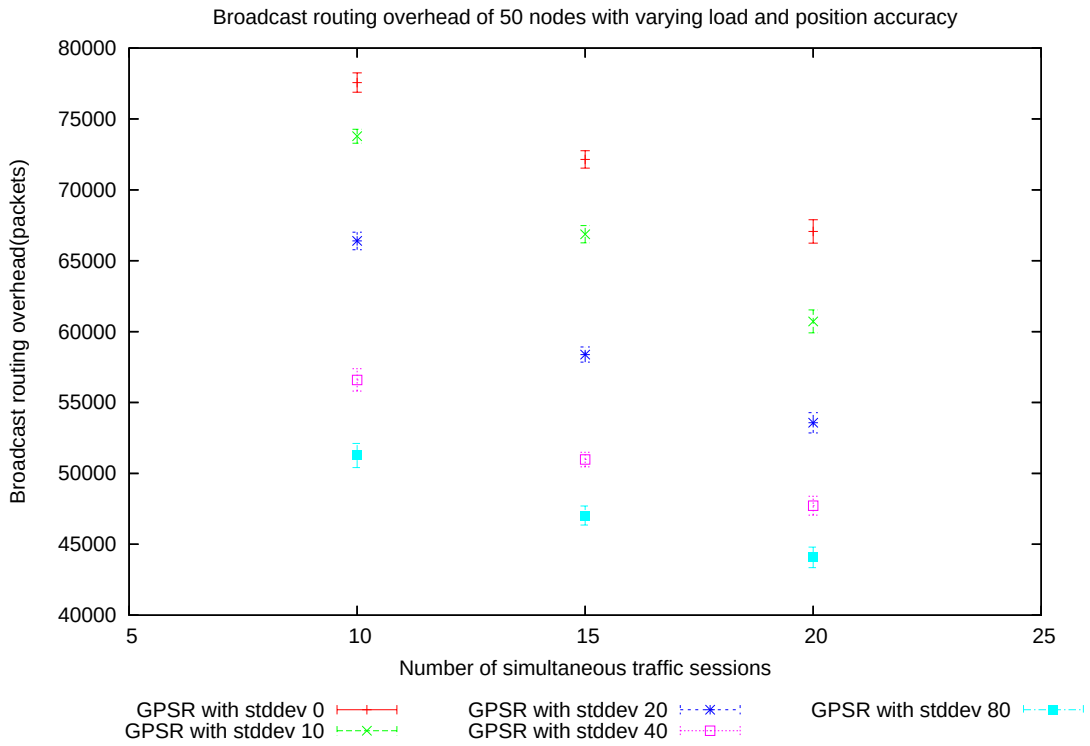


Figure 3.6: Broadcast routing overhead of 50 nodes with varying load and position accuracy

more periodic updates to be delivered through GPSR piggyback.

In addition, the results also showed that periodic broadcast updates reduce when position inaccuracy increases. This is because the increase in position inaccuracy has adverse effects on GPSR forwarding algorithm. Researchers have shown this is likely to cause the routing packets to be caught in a loop or travelled through less optimal route [Kim *et al.* (2004)] [Seada *et al.* (2004)] [Son *et al.* (2004)]. This, causes the periodic neighbour updates to be rescheduled more often since more routing packets are available for piggyback delivery.

Most importantly, an increase in position inaccuracy also increase the number of data packets being dropped at loops, dead ends or broken links. Our results in section 3.7 show that end to end packet delivery reduces as position inaccuracy increases. This infers that some of the data packets are dropped before they can reach the destination. This also means that those piggyback packets that are supposed to replace the periodic updates can be lost. As a results of this, there are not enough routing packets to make up the reduction of rescheduled periodic update. This explains our results in Figure 3.7, showing that overall routing overhead reduces when position inaccuracy increases.

3.4.5 The Impact of Position Error on LGF Updates

In section 2.3.6, we present the adaptive route advertisement for the routing protocol. The adaptive route advertisement uses position information of the neighbours to regulate the frequency of route advertisement. In this section, we discuss the effects of position uncertainty on

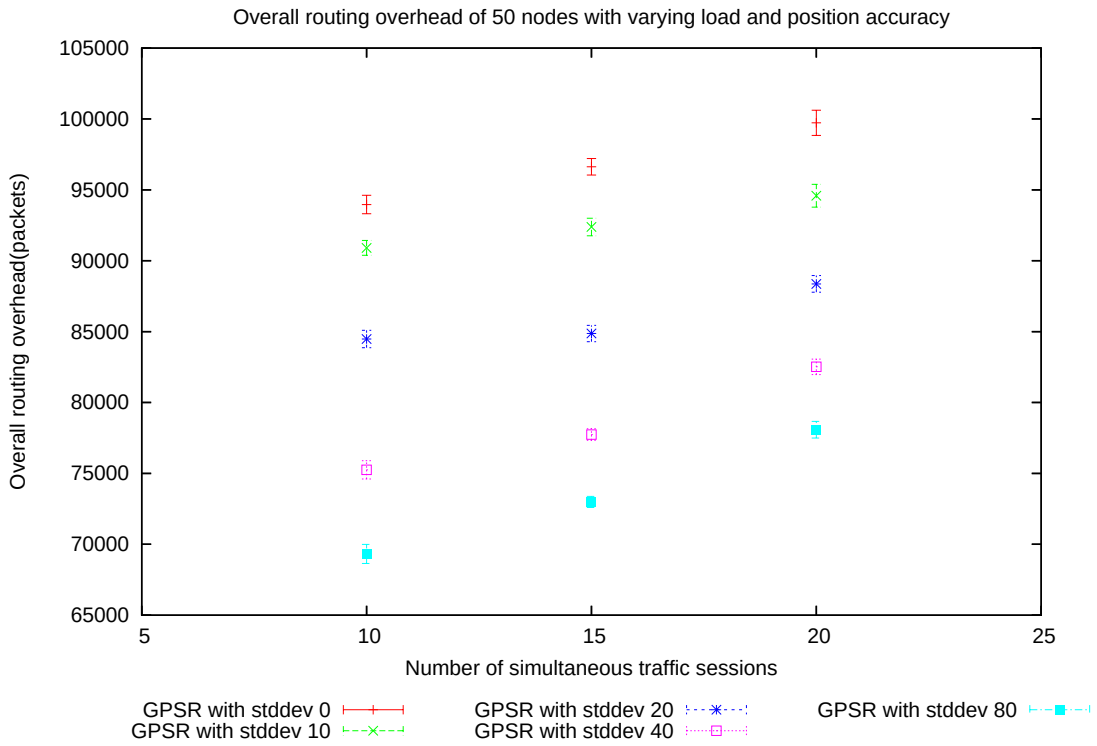


Figure 3.7: Overall routing overhead of 50 nodes with varying load and position accuracy

this algorithm, namely: not equally likely event and compound event.

Not Equally Likely Events

In a scenario as shown in Figure 3.8. A node i has only one adjacent neighbour j. Let R be the actual distance between node i and node j. In addition, we use a circular area to model the possible positions where j's location could be advertised with position inaccuracy, with centre of the circle at the actual location where r is and the radius of the circle equal to the maximum possible inaccurate position.

We define a function $d(x,y)$ that measures the distance between two nodes, x and y. We denote J as a possible inaccurate location of j. As we can clearly see, the area where $d(i,J) > R$ is greater than the area where $d(i,J) < R$.

We have previously determined that GPS inaccuracy can be approximated by a normal distribution. It is therefore possible to represent the distribution of advertised position of node j by using the same circle without compromising the integrity of our observation.

In order to show the area where $d(i,J) > R$, we uses the big centre with node i being the centre of the circle with radius equal to R. Based on our observation in Figure 3.8, the angle θ can be calculated as follows.

$$\theta = \arcsin \frac{r}{2R} \tag{3.2}$$

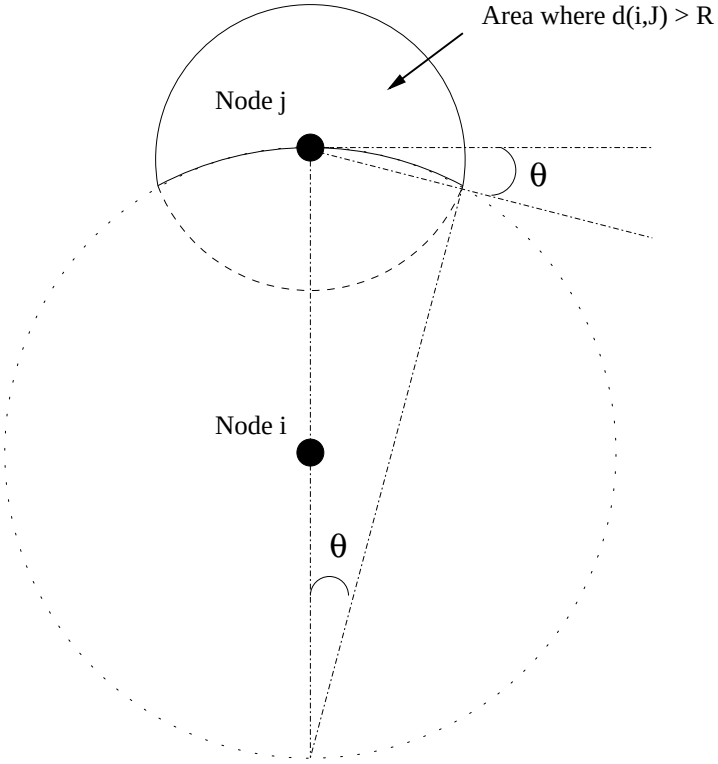


Figure 3.8: Not equal likely events

Where r is the radius of the small circle with j being the centre of the circle and R is the radius of big circle with i being the centre of the circle.

We now denote E as an event when $d(i,J) > R$ and $P(E)$ as the probability of event E.

$$P(E) = \frac{n(E)}{n(S)}$$

$$> 0.5 \tag{3.3}$$

Where $n(E)$ is the number of elements in *'E'* and $n(S)$ is the number of elements in sample space *'S'*. In addition, we can also approximate the probability of the event of j's advertised position being nearer than the actual distance as follows.

$$1 - P(E) < 0.5 \tag{3.4}$$

Compound Event

The adaptive route advertisement uses the maximum distance to the furthest neighbour within the radio range to regulate the update frequency of route advertisement. Below is the analysis of the impact of position error on this adaptive route advertisement.

Let x_i be the actual distance between a node and its neighbours within the wireless range. Then if there are n neighbours, the distance to the neighbours are x_1, x_2, x_3, ... x_n (iid).

Our findings in section 3.4.5 show that there is slightly higher probability of getting a larger value than a lower value from a neighbour advertisement when position inaccuracy is considered. In order to simplify the analysis, we assume equal probability of getting a lower or higher value from the advertisement. In addition, we assume the error is spatially correlated and every neighbour within the wireless range has identical position error, δ.

Then, the measured distance between the node and its neighbours are D_1, D_2, D_3,.....D_n(iid).

$$\begin{aligned} ||x_1 + \delta|| &= D_1 \\ ||x_2 + \delta|| &= D_2 \\ ||x_3 + \delta|| &= D_3 \\ &..... \\ &..... \\ ||x_n + \delta|| &= D_n \end{aligned}$$

Although δ can be a positive or negative number, the resultant distance is always an absolute value.

Let $Max(D_1, D_2, D_3.....D_n)$ be the maximum measured distance between the node and its neighbours. Then,

$$Prob(Max(D_1, D_2, D_3) > Max(D_1, D_2))$$

$$> Prob(Max(D_1, D_2) > Max(D_1, D_2, D_3)) \tag{3.5}$$

This is straightforward and has shown that it is more likely to obtain a larger number from $Max()$ function when more neighbours are available.

Let δ_1 and δ_2 be the random position error of position tracking system, subject to $\delta_2 > \delta_1$.

Let $||x_i + \delta_1|| = D_i{}'$
Let $||x_i + \delta_2|| = D_i{}''$

$$Prob(Max(D_1'', D_2'', D_3'', .., D_n'') > Max(D_1', D_2', D_3, .., D_n'))$$

$$> Prob(Max(D_1'', D_2'', D_3'', .., D_n'') < Max(D_1', D_2', D_3', .., D_n')) \tag{3.6}$$

As the measured distance from n neighbour can be treated as independent identically distributed variables. When n is sufficiently large, the value obtained from Max() is more likely to increase with an increase in positional inaccuracy δ. Based on equation 2.1 of the adaptive route

advertisement in section 2.3.6, this increase in measured distance is transformed to more frequent neighbourhood updates. Our simulation results in section 3.7 show that routing overhead increases when position inaccuracy increases.

3.5 Resilience of LGF to Position Uncertainty

The LGF protocol is designed to overcome some of the limitations of the approaches taken by existing ad hoc routing protocols. In particular it is designed to be resilient to position inaccuracies whether arising from location device error or from the update frequency of the *location service*.

LGF is a hybrid protocol that subsumes some of the design decisions made in position-based forwarding protocols. It limits the propagation of advertisements and uses position information to steer a packet when a node does not have an established path to the destination. This approach not only enables the reactive element of the protocol to generate more frequent advertisements while adapting to fast changing connectivities but also allows the protocol to contain its overheads and scale well to larger networks. In order to resolve the issues encountered as a result of the inaccuracy of the position system and the staleness of a node's location within the *Location Service*, LGF uses *Restrictive Hybrid Route Advertisement* along with stateful *LGF Exploration* combined with its *Next Hop Selection* algorithm. Details description of LGF protocols can be found in chapter 2. In this section, we discuss its resilience to positional inaccuracy.

3.5.1 Resilience of LGF to Inaccurate Location Information

With restricted hybrid state in every node, LGF uses hierarchical routing to preserve an equilibrium between selecting an optimal routing decision and minimising routing overheads. In the following section, we describe scenarios that illustrate how local routing and path exploration operate in the presence of inaccurate location information and stale node location state held by the location service. We consider two further scenarios where LGF performs differently depending upon the proximity of the destination to the source. The first scenario is where the destination is located within the source local forwarding set, and the second scenario is where the destination is further away.

Scenario 1 - Local Destination

Figure 3.9 is a scenario in which the destination lies within the local scope of the hybrid advertisement that has been limited to two hops from the source S. Let R be the set of nodes in this diagram, $r \in R$, R=$\{S,1,2,3,4,5,D\}$. $\forall r$, r' represents the inaccurate location of r. In addition, D" represents the stale position information of D in the location service depository. Since D is within the range of the hybrid advertisement of S, the source S would be able to apply the shortest path algorithm to deliver the packet from $S \rightarrow 1 \rightarrow D$ regardless of the inconsistencies of the surrounding neighbours r' and D". This scenario illustrates that neither the inaccuracy of the positioning system nor stale state reported by the location service have an impact on the protocol within the local area.

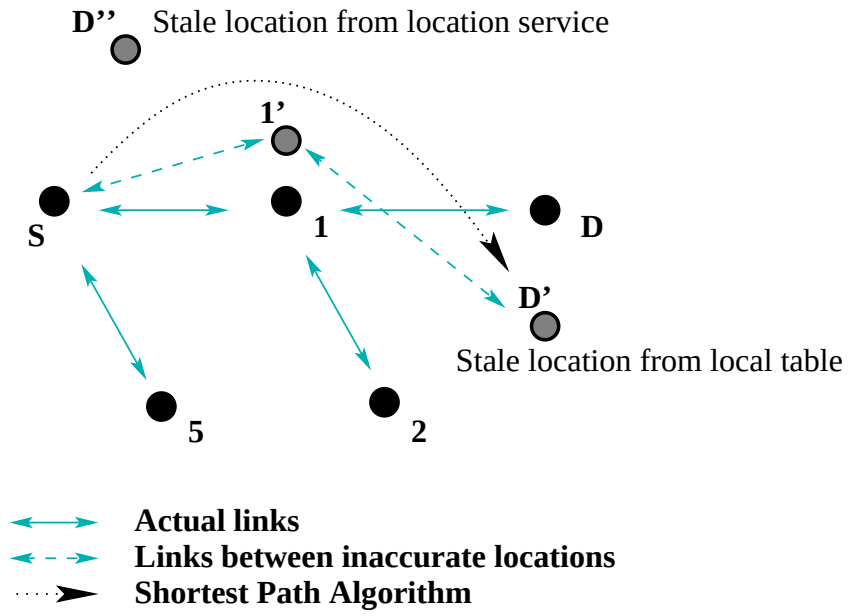


Figure 3.9: When destination is within local area

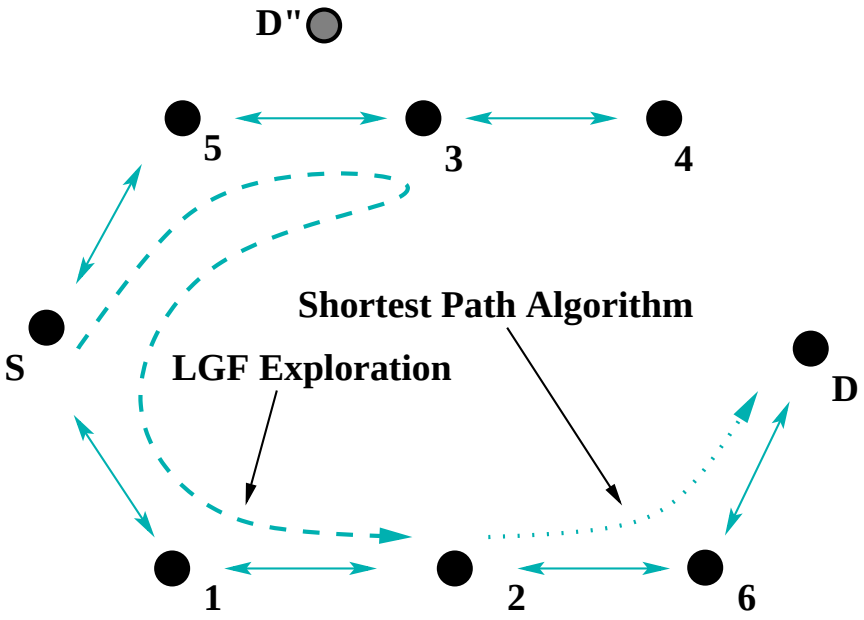


Figure 3.10: When destination is beyond local area

Scenario 2 - Destination Outside Local Scope

Figure 3.10 is a scenario in which the destination is beyond the local scope of the hybrid advertisement that has been limited to two hops from the source S. Let R be the set of nodes in the diagram. $r \in R$, R=$\{S,1,2,3,4,5,D\}$. $\forall r$, r' represents the inaccurate location of r. In addition, D" represents the stale position information of D in the location service repository.

As the source S does not have a shortest path to the destination, it applies the LGF exploration algorithm to locate a path to the destination D. However, since S takes D" as the destination position, the exploration leads the packet to traverse $S \rightarrow 5 \rightarrow 3$, bearing in the

direction of D". At node 3, the exploration algorithm detects a dead-end, and backtracks to continue path exploration from an alternative route. Upon arriving at node 2, the algorithm finds the packet's destination in its routing table, and directs the packet to the destination through the shortest path. This shows LGF exploration is able to find the destination regardless of the influence from inconsistency from surrounding neighbours r' and D".

3.6 Simulation Experiment Design

In this section, we outline the models, parameters and assumptions of our evaluation. The focus of our evaluation is to model the performance of existing position-based forwarding techniques and to establish the resilience of LGF to position error. We discuss the mechanisms and parameters we have used to measure the performance, beginning with a discussion of the positioning system accuracy model, followed by discussion of the update frequency for the location service model, and concluding with a detailed account of how the simulations were conducted.

3.6.1 Accuracy of Positioning System

In modelling the position tracking system, we considered various techniques and their reported accuracy. Some researchers have proposed using the Global Positioning System (GPS) to retrieve geographical information for position-based forwarding protocols [Li *et al.* (2000)]. In general, the GPS receiver needs to have visibility of a number of satellites and, consequently, is not practical indoors, or in cities with many tall buildings where the satellites are obscured. In addition, GPS devices are still quite expensive for ordinary consumers, although it does have a distinct benefit of providing almost world wide coverage, with the accuracy ranging from 10 to 100 metres. In recent trials of various position tracking technologies as reported by Transport of London [Transport of London (2005)], they evaluated the confidence of tracking a target within zones of 60 and 250 metre boundaries in London. The GPS technology did not meet the required 99% confidence level.

In our simulation, a Gaussian Distribution function was used to model position error with mean value of 0. To model the accuracy of a wide range of location services, we use standard deviations of 0, 10, 20, 40, 80, 160 and 200 metres.

3.6.2 Update Frequency for Location Service

The Grid Location Service (GLS) has been proposed as a distributed location service that stores updates and returns responses to queries from a set of nodes in a spatial hierarchical order [Li *et al.* (2000)]. Using a consistent hash function, each node periodically updates its position in a set of nodes in the network. Likewise, a destination location is retrieved by applying the same hash function. Although GLS is a robust system for providing a location repository service for a wireless ad hoc network, there is always a time lapse between the last update and the later retrievals of a destination's position. In our simulation, we use update frequencies of 0, 300 and 600 seconds to model the inconsistency caused by the update frequency time lapse.

3.6.3 Simulation Scenario

The simulations have been carried out using the NS2 simulator [ns group at ISI (1989)], with each simulation lasting for 900 seconds. We use the same physical, MAC, traffic and mobility models as in section 2.5 of the previous chapter.

The Effects of Positional Accuracy with Varying Velocity

To evaluate the effects of position accuracy on constantly changing connectivity, we use 5 different sets of mobility patterns generated with maximum velocity of 1, 2.5, 5, 7.5, 10, 12.5 and 15 m/s at 0 pause time, with 100 nodes in a geographic area of 1500x500 m^2. These 5 sets of mobility patterns are then run with 4 sets of traffic patterns to generate 20 sets of simulation results for evaluation. To isolate the influence of position accuracy from the effects of location update frequency, we use an ideal location update setting for both protocols with various settings of position accuracy discussed in 3.6.1.

In addition to the performance comparison of LGF with GPSR, we also include a simulation to evaluate the performance of LGF when the position inaccuracy is greater than the scope of restriction by the hybrid route advertisement. Since our simulations are conducted with a radio propagation model of 250 metres radius and 3 hops of restrictive hybrid advertisement, an additional set of test scenarios with 800 metre position inaccuracy is adequate for this evaluation.

The Effects of Location Update Frequency with Varying Position Accuracy

In addition, we evaluate the effects of variable update frequency with position accuracy variation on LGF. In the case of GPSR, we have not found a way to reconfigure the GPSR location service from its ideal location service model. Nevertheless, this set of simulations focuses solely on demonstrating the resilience of LGF with GPSR configured with an ideal location service. We use 5 different sets of mobility patterns generated with maximum velocity of 15 m/s at 0 pause time, with 100 nodes in a geographic area of 1500x500 m^2. These 5 sets of mobility patterns are then run with 4 sets of traffic patterns to generate 20 sets of simulation results for evaluation.

We compare LGF and GPSR with the above effects. Each ad hoc routing protocol has some settings specific to it that can be found in Table 2.10 and Table 2.8 of the previous chapter.

3.7 Results and Analysis

In this section, we present our findings of two sets of simulations: namely, performance with varying maximum velocity and position accuracy, and performance with varying position accuracy and location update frequency.

3.7.1 Performance with Varying Maximum Velocity and Position Accuracy

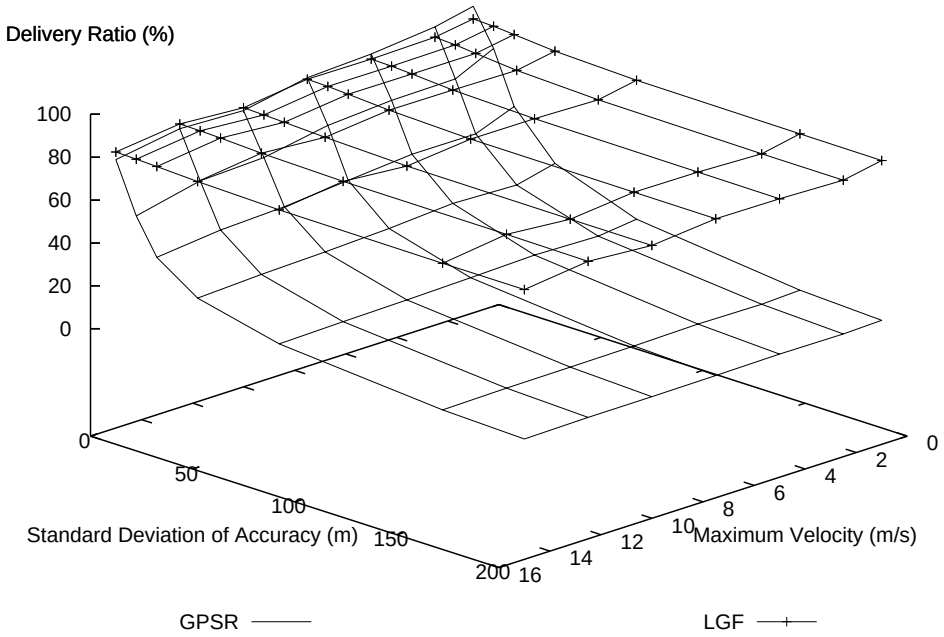


Figure 3.11: Delivery ratio with varying maximum velocity and position accuracy

Figure 3.11 shows the delivery ratio of LGF and GPSR with various settings of maximum velocity and position accuracy. In order to discuss this clearly, we present the details of delivery ratio in Table 3.2. From our observation, there is a declining delivery ratio trend for GPSR as the standard deviation of accuracy increases.

For example, in the case in which the maximum velocity is configured to 1 m/s, our results show a substantial drop of delivery ratio from 92.87% to 7.85 %, as the standard deviation of positional accuracy increases from 0 to 200 metres. In contrast, LGF maintains a relatively stable delivery ratio in the 80% region, with a marginal decline of 4.70%, as the standard deviation of position accuracy increases from 0 to 200 metres.

In general, both protocols present a gradual decline of delivery ratio when the node maximum velocity increases from 1 m/s to 15 m/s. However, in the simulation scenario with the standard deviation of position accuracy reset to 0, using the GPSR routing protocol, the delivery ratio falls by 17.85%, from 92.87% to 75.02%, when the maximum velocity setting increases from 1 m/s to 15 m/s. In contrast, the delivery ratio of LGF falls by 8.42%, from 86.98% to 78.46%, when the maximum velocity setting increases from 1 m/s to 15 m/s.

From these findings, it is apparent that the hybrid stateful approach implemented by LGF is more resilient to handling position inaccuracy as compared to the near stateless position-based forwarding technique. However, in scenarios with 0 standard deviation of position accuracy, GPSR yields a better packet delivery ratio than LGF when maximum velocity is below 7.5 m/s. The performance difference could be accounted for by the efficiency of GPSR perimeter routing on planar graph when position inconsistency induced by node mobility is less critical. In the case of LGF, it use a path exploration with a restrictive scope of three hops to search the 100

Table 3.2: Delivery ratio with varying maximum velocity and position accuracy

Protocol	Stdev of pos. Error (m)	Delivery ratio(stdev) with various maximum velocity(%) 1	**2.5**	5	**7.5**
LGF	0	86.98(4.94)	**84.21(4.01)**	83.57(3.93)	**83.79(3.26)**
LGF	10	86.67(5.12)	**83.74(4.17)**	83.38(3.97)	**83.46(3.23)**
LGF	20	85.86(5.31)	**82.81(4.06)**	82.80(3.99)	**82.94(3.38)**
LGF	40	84.24(5.63)	**81.02(4.36)**	81.40(4.36)	**81.62(3.49)**
LGF	80	82.94(5.92)	**79.53(4.51)**	80.33(4.67)	**80.42(3.53)**
LGF	160	82.43(5.98)	**78.88(4.52)**	79.93(4.78)	**80.12(3.46)**
LGF	200	82.28(5.96)	**78.83(4.53)**	79.76(4.82)	**79.99(3.71)**
LGF	800	78.44(7.16)	**75.08(5.09)**	75.88(5.20)	**76.61(4.26)**
GPSR	0	92.87(3.38)	**89.02(3.64)**	85.95(3.31)	**84.49(2.26)**
GPSR	10	76.30(4.78)	**67.83(4.00)**	67.41(4.92)	**65.34(3.77)**
GPSR	20	52.30(3.67)	**45.33(2.64)**	45.38(3.23)	**42.83(2.94)**
GPSR	40	31.97(2.50)	**27.64(1.56)**	28.59(1.94)	**26.46(2.07)**
GPSR	80	18.21(4.12)	**16.02(1.13)**	16.79(1.35)	**15.81(1.43)**
GPSR	160	9.65(0.94)	**8.76(0.76)**	9.15(0.97)	**8.48(0.83)**
GPSR	200	7.85(0.76)	**7.18(0.66)**	7.41(0.76)	**6.98(0.66)**
Protocol	Stdev of Pos. Error (sec)	Delivery ratio(stdev) with various maximum velocity(%) 10	**12.5**	15	
LGF	0	80.05(3.70)	**82.12(3.35)**	78.46(2.79)	
LGF	10	79.76(3.69)	**81.96(3.36)**	78.24(2.67)	
LGF	20	79.42(3.80)	**81.63(3.37)**	77.98(2.58)	
LGF	40	78.51(3.92)	**80.74(3.41)**	76.96(2.71)	
LGF	80	77.39(4.10)	**79.84(3.56)**	76.20(3.02)	
LGF	160	77.11(4.12)	**79.62(3.56)**	75.91(3.17)	
LGF	200	77.12(4.27)	**79.34(3.78)**	75.67(3.03)	
LGF	800	73.54(5.04)	**76.39(4.44)**	72.26(3.54)	
GPSR	0	78.74(3.14)	**79.82(3.30)**	75.02(2.65)	
GPSR	10	59.91(5.53)	**58.69(5.78)**	51.85(3.61)	
GPSR	20	40.05(3.72)	**38.79(3.98)**	35.61(2.75)	
GPSR	40	25.18(2.34)	**24.26(2.30)**	22.77(1.60)	
GPSR	80	15.03(1.39)	**14.49(1.08)**	13.68(1.10)	
GPSR	160	8.17(0.77)	**8.00(0.57)**	7.40(0.68)	
GPSR	200	6.68(0.71)	**6.57(0.49)**	6.10(0.51)	

node graph in the simulation area of 1500 x 500 metre square and is therefore more likely to use a longer path than GPSR when position uncertainty is less significant. As a result of using a longer path, LGF is likely to encounter more packet losses than GPSR. Current LGF is not optimised; we would expect to improve performance of LGF in this respect.

Figure 3.12 shows the routing overhead of LGF and GPSR with various settings of maximum velocity and position accuracy. In order to discuss this clearly, we present the details of routing overhead in Table 3.3. From our observation, there is a declining routing overhead trend for GPSR as the standard deviation of accuracy increases. Conversely, there is an inclining routing overhead trend for LGF as the standard deviation of accuracy increases.

For example, in the case in which maximum velocity is configured to 1 m/s, our results show a drop of GPSR routing overhead from 186981 to 157755 packets, as the standard deviation of positional accuracy increases from 0 to 200 metres. In contrast, LGF routing overhead increases from 109601 to 128278 packets, as the standard deviation of position accuracy increases from 0 to 200 metres.

Our findings in section 3.4.5 and 3.4.4 show the neighbourhood advertisement mechanism of both LGF and GPSR are affected by position inaccuracy. Section 3.4.4 shows that inaccuracy in position can have adverse effects on the piggyback feature used by GPSR. This feature reschedules the periodic broadcast and piggybacks position updates onto outgoing data packets.

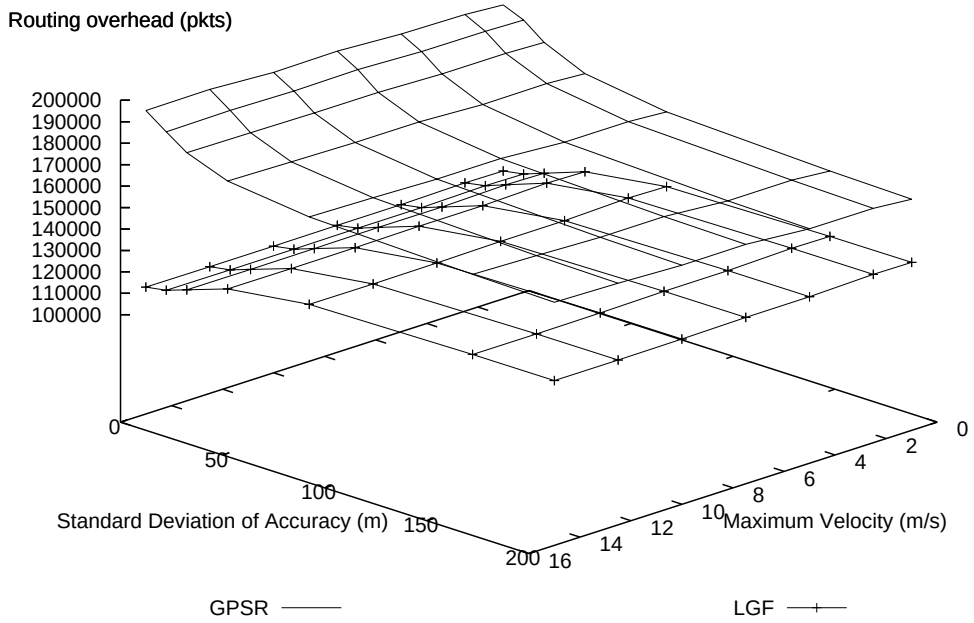


Figure 3.12: Routing overhead with varying maximum velocity and position accuracy

Our findings show more data packets are lost when positional inaccuracy increases. This, in fact, reduces the exchange of states between neighbours in the network. As a result, the protocol is more vulnerable to false construction of the planar graph, which could compromise the correctness of the forwarding technique.

In the case of LGF, our findings in section 3.4.5 show the increase in position inaccuracy has a positive effect on LGF adaptive neighbourhood advertisement. It increases the frequency of neighbourhood updates and may help to reduce the number of stale hybrid states in the network.

Figure 3.13 shows the average packet delay of LGF and GPSR with various settings of maximum velocity and position accuracy. In order to discuss this clearly, we present the details of average packet delay in Table 3.4. From our observation, there is a trend of increases in GPSR average packet delay with increases in standard deviation of position accuracy.

In the case in which maximum velocity is configured to 1 m/s, our results show an increase in average packet delay from 0.11 to 20.48 seconds, as the standard deviation of positional accuracy increases from 0 to 200 metres. In contrast, LGF average packet delay increases from 0.37 to 1.07 seconds, as the standard deviation of position accuracy increases from 0 to 200 metres. This shows that LGF is able to sustain relatively low average packet delay when compared with GPSR, even with a significant amount of position error.

In our earlier discussion of packet delivery ratio in this section, we explained that LGF is more likely to take a longer route than GPSR when position inconsistency is less critical. This means that LGF is more likely to experience much higher average packet delay than GPSR when positional uncertainty is less significant. Our results show that average packet delay of GPSR

Table 3.3: Routing overhead with varying maximum velocity and position accuracy

Protocol	Stdev of pos. Error (m)	Routing overhead(stdev) with various maximum velocity(*packets*) 1	**2.5**	5	**7.5**
LGF	0	109601(648)	**109827(680)**	109295(597)	**109237(497)**
LGF	10	111281(678)	**111576(726)**	110941(549)	**110939(481)**
LGF	20	114717(706)	**115095(781)**	114316(574)	**114358(539)**
LGF	40	121508(814)	**121966(848)**	121014(637)	**121045(523)**
LGF	80	126883(841)	**127272(829)**	126282(673)	**126255(616)**
LGF	160	128124(704)	**128380(790)**	127510(681)	**127463(555)**
LGF	200	128278(757)	**128447(769)**	127600(680)	**127521(544)**
LGF	800	128678(783)	**128887(858)**	128085(724)	**127879(574)**
GPSR	0	186981(2645)	**188262(1552)**	188772(1376)	**190388(1803)**
GPSR	10	183021(1642)	**183479(903)**	183650(1327)	**184794(1169)**
GPSR	20	175644(1332)	**176325(1105)**	176292(1448)	**177289(966)**
GPSR	40	167243(1395)	**168343(1166)**	168121(1568)	**169588(1216)**
GPSR	80	161647(1629)	**162899(1420)**	162856(1751)	**164772(1577)**
GPSR	160	158632(1745)	**160071(1689)**	160028(1915)	**162249(1733)**
GPSR	200	157755(1667)	**159185(1791)**	159276(1881)	**161568(1611)**

Protocol	Stdev of Pos. Error (sec)	Routing overhead(stdev) with various maximum velocity(*packets*) 10	**12.5**	15
LGF	0	109091(533)	**109023(456)**	109054(514)
LGF	10	110731(555)	**110681(489)**	110698(469)
LGF	20	114041(525)	**113926(508)**	113917(461)
LGF	40	120644(609)	**120478(633)**	120513(479)
LGF	80	125760(614)	**125517(579)**	125546(506)
LGF	160	126932(620)	**126715(526)**	126737(492)
LGF	200	127024(642)	**126831(516)**	126806(523)
LGF	800	127361(629)	**127142(483)**	127153(518)
GPSR	0	189966(1705)	**191669(980)**	191395(1066)
GPSR	10	184423(1091)	**184918(796)**	184554(796)
GPSR	20	177237(1009)	**177569(824)**	177854(910)
GPSR	40	169583(1182)	**170182(1018)**	170755(1039)
GPSR	80	164815(1513)	**165693(1254)**	166352(1131)
GPSR	160	1621721(1645)	**163241(1290)**	164052(1068)
GPSR	200	161443(1582)	**162605(1295)**	163327(1080)

is less than that of LGF when the standard deviation of position accuracy is configured to 0.

In this series of simulations, when we removed the effect of positional inaccuracy by setting its standard deviation to 0, our results show that the average packet delay for GPSR is between 0.09 and 0.15 seconds when we increase the maximum velocity from 1 m/s to 15 m/s. Likewise, the figure shows that LGF maintains stable but slightly higher average packet delay between 0.24 to 0.46 seconds than GPSR in similar scenarios. This shows the maximum node velocity in use creates less impact than the positional inaccuracy in the simulations.

Considering these results collectively, it shows there is a correlation between the reduction of routing overhead found in GPSR with a reduction in its performance in terms of delivery ratio, as well as its undesirable average packet delay.

The results also show that LGF is resilient to the effects of positional inaccuracy, even when all nodes in the network are constantly moving, i.e. 0 pause time, with the maximum velocity set to 15 m/s.

In the following set of results, a test scenario for evaluation of the performance of LGF when position inaccuracy is greater than the scope of restrictive hybrid advertisement is presented. Our results show that LGF performance degrades gradually when position inaccuracy increases from 200 to 800 metres. In our simulation results with maximum node velocity configured to 1

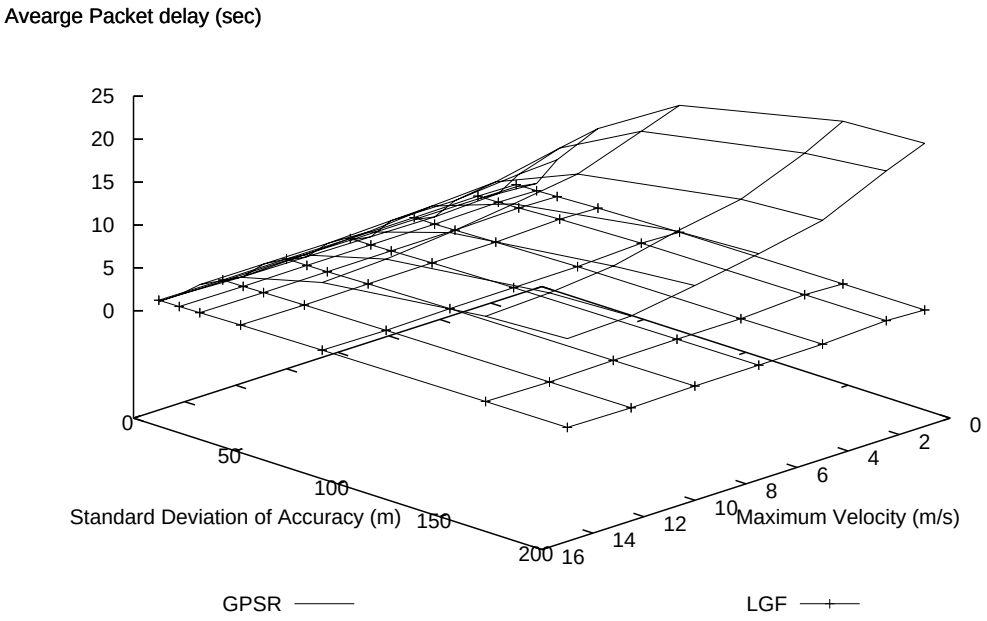


Figure 3.13: Packet delay with varying maximum velocity and position accuracy

m/s, the delivery ratio decreases from 82.28% to 78.44% when positional inaccuracy increases from 200 to 800 metres. The routing overhead increases from 128268 to 128678 packets when positional inaccuracy increases from 200 to 800 metres. The average package delay increases from 1.07 to 1.73 second when the positional inaccuracy increases from 200 to 800 metres.

LGF relies on the geographic information of remote neighbours for path exploration when the destination is not within the hybrid topological area. The results show that a significant increase in positional inaccuracy is likely to degrade the general ability of the path exploration. However, the results also establish the fact that the hybrid algorithm is more resilient to positional uncertainty than the generic position-based forwarding algorithm.

3.7.2 Performance with Varying Position Accuracy and Location Update Frequency

Figure 3.14 shows the delivery ratio of LGF with a combination of various positional accuracy and location update frequency settings, alongside the GPSR delivery ratio obtained from different settings of position accuracy with maximum velocity of 15 m/s and 0 pause time. In order to discuss this clearly, we present the details of delivery ratio in Table 3.5.

The graph shows that LGF packet delivery ratio reduces when location update frequency increases. For example, in the set of simulations where the standard deviation of positional inaccuracy is 0, LGF packet delivery ratio reduces from 78.46% to 74.45% when location update frequency changes from idealised (0 second) to 300 seconds. The packet delivery ratio is further reduced to 72.59% when the location update frequency increases to 600 seconds.

Table 3.4: Average packet delay with varying maximum velocity and position accuracy

Protocol	Stdev of pos. Error (m)	Average packet delay(stdev) with various maximum velocity(*packets*) 1	**2.5**	5	**7.5**
LGF	0	0.37(0.30)	**0.46(0.24)**	0.34(0.21)	**0.29(0.13)**
LGF	10	0.41(0.36)	**0.50(0.26)**	0.35(0.22)	**0.31(0.15)**
LGF	20	0.48(0.42)	**0.60(0.30)**	0.41(0.26)	**0.38(0.20)**
LGF	40	0.69(0.54)	**0.83(0.38)**	0.55(0.32)	**0.52(0.24)**
LGF	80	0.93(0.69)	**1.08(0.52)**	0.74(0.44)	**0.70(0.29)**
LGF	160	1.04(0.75)	**1.22(0.53)**	0.82(0.46)	**0.80(0.30)**
LGF	200	1.07(0.75)	**1.25(0.52)**	0.90(0.51)	**0.84(0.36)**
LGF	800	1.73(1.13)	**1.80(0.73)**	1.36(0.70)	**1.30(0,52)**
GPSR	0	0.11(0.09)	**0.12(0.08)**	0.12(0.07)	**0.09(0.04)**
GPSR	10	1.26(0.50)	**1.51(0.32)**	1.19(0.45)	**1.26(0.29)**
GPSR	20	4.78(0.69)	**4.52(0.51)**	3.85(0.52)	**3.91(0.41)**
GPSR	40	9.94(0.98)	**9.09(1.14)**	7.57(0.83)	**7.14(0.67)**
GPSR	80	15.72(1.61)	**14.13(1.39)**	11.54(1.28)	**10.45(0.97)**
GPSR	160	19.99(2.03)	**17.69(1.66)**	14.71(1.77)	**13.38(1.14)**
GPSR	200	20.48(1.97)	**18.68(1.84)**	15.33(1.80)	**13.83(1.09)**

Protocol	Stdev of Pos. Error (sec)	Average packet delay(stdev) with various maximum velocity(*packets*) 10	**12.5**	15
LGF	0	0.31(0.09)	**0.24(0.06)**	0.29(0.07)
LGF	10	0.32(0.11)	**0.26(0.07)**	0.30(0.07)
LGF	20	0.36(0.11)	**0.30(0.09)**	0.32(0.07)
LGF	40	0.46(0.17)	**0.39(0.11)**	0.43(0.10)
LGF	80	0.63(0.22)	**0.51(0.16)**	0.60(0.17)
LGF	160	0.73(0.23)	**0.61(0.20)**	0.73(0.19)
LGF	200	0.81(0.28)	**0.67(0.20)**	0.79(0.22)
LGF	800	1.20(0.45)	**0.97(0.45)**	1.15(0.30)
GPSR	0	0.15(0.07)	**0.11(0.03)**	0.14(0.04)
GPSR	10	1.33(0.55)	**1.40(0.42)**	1.68(0.38)
GPSR	20	3.63(0.67)	**3.63(0.48)**	3.63(0.50)
GPSR	40	6.51(0.86)	**6.23(0.60)**	6.02(0.55)
GPSR	80	9.49(1.09)	**8.80(0.51)**	8.48(0.81)
GPSR	160	11.74(1.33)	**10.98(0.68)**	10.66(0.76)
GPSR	200	12.51(1.36)	**11.36(0.73)**	11.11(0.83)

The results show that the packet delivery ratio of LGF decreases as the time interval between two updates increases. This is because the longer the time interval between two updates, the more likelihood there is that a node will move further away from its last known location. As a result, packets are more likely to be delivered through a longer path, which can be subjected to more link failures, media contentions or other MANET conditions.

Our results show that LGF is not significantly degraded even by substantial position inaccuracies or by high levels of staleness in the location update state in this 100 node scenario. The packet delivery ratio reduced by approximately 9% when 200 metres of position inaccuracy is introduced with 600 seconds of location update frequency. Conversely, consistent with the set of results in section 3.7.1, there are signs of deterioration with the increase of the standard deviation of positional accuracy induced in GPSR. The graph shows that the packet delivery ratio of GPSR decreases significantly as the standard deviation of position accuracy increases from 0 to 200 metres with idealised location.

Figure 3.15 shows the routing overhead of LGF with a range of position accuracies and location update frequencies. In order to discuss this clearly, we present the details of routing overhead in Table 3.6.

It is our observation that the routing overhead of LGF increases as the time interval between location updates increases. As discussed earlier with the results of packet delivery ratio, the path

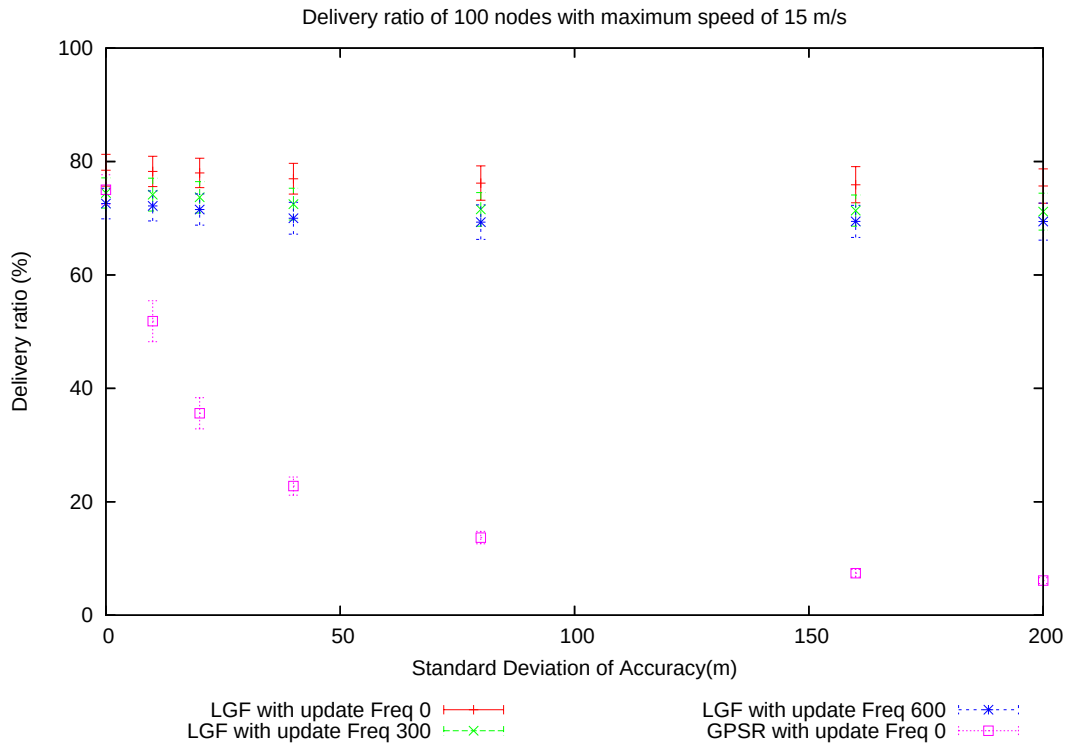


Figure 3.14: Delivery ratio with varying update frequency

length is likely to increase as the time interval between location updates increases. This increase in path length increases the number of routing packets in the network, which, in turn, increases the overall routing overhead that comprises route updates and routing packets.

Figure 3.16 shows the average packet delay with a range of positional accuracy and location update frequency settings. In order to discuss this clearly, we present the details of average packet delay in Table 3.7.

Despite the constraints of position accuracy and location update frequency, all results shows that LGF average packet delay is between 0.29 and 1.40 seconds. In contrast, the average packet delay of GPSR increases from 0.29 to 11.11 seconds when the positional inaccuracy increases from 0 to 200 metres. The results show that LGF is more robust than GPSR despite significant constraints imposed by a combination of positional accuracy and location update frequency. However, our results showed staleness of location update service has an impact on LGF performance.

For example, in the set of simulations where the standard deviation of position inaccuracy is 0, LGF average packet delay increases from 0.29 to 0.67 seconds when location update frequency increases from idealised (0 second) to 300 seconds. The average packet delay is further increased to 0.83 seconds when the location update frequency increases to 600 seconds. This can be accounted for by additional queueing or media contention delays as a result of packets needing to transit through more hops in order to reach the destination.

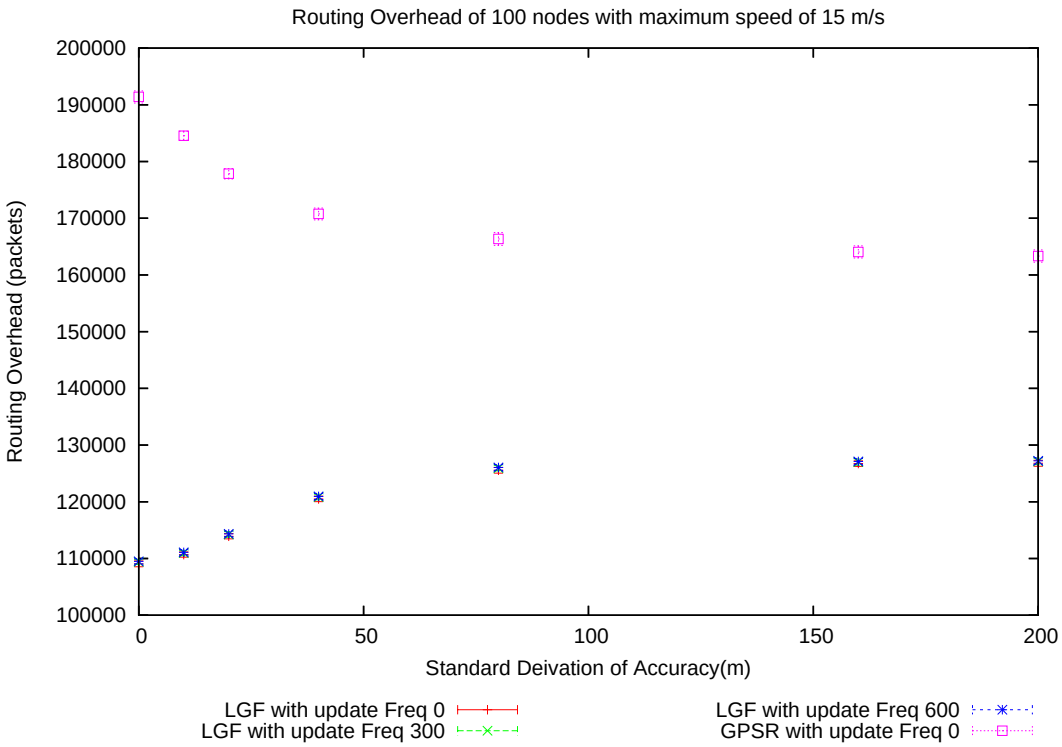


Figure 3.15: Routing overhead with varying position accuracy and update frequency

Table 3.5: Delivery ratio with various standard deviation of position inaccuracy.

Protocol	Location Update (sec)	Delivery ratio(stdev) with various stdev. of inaccuracy(%) 0	**10**	20	**40**
LGF	0	78.46(2.79)	**78.24(2.67)**	77.98(2.58)	**76.96(2.71)**
LGF	300	74.45(2.66)	**74.15(2.89)**	73.69(2.75)	**72.48(2.80)**
LGF	600	72.59(2.70)	**72.16(2.64)**	71.53(2.57)	**69.99(2.79)**
GPSR	0	75.02(2.65)	**51.85(3.61)**	35.61(2.75)	**22.77(1.60)**
Protocol	Location Update (sec)	Delivery ratio(stdev) with various stdev. of inaccuracy(%) 80	**160**	200	
LGF	0	76.20(3.02)	**75.91(3.17)**	75.67(3.03)	
LGF	300	71.55(2.80)	**71.35(2.75)**	71.16(3.25)	
LGF	600	69.32(3.03)	**69.44(2.82)**	69.40(3.26)	
GPSR	0	13.68(1.10)	**7.40(0.68)**	6.10(0.51)	

Table 3.6: Routing overhead with various standard deviation of position inaccuracy.

Protocol	Location Update (sec)	Routing Overhead(stdev) with various stdev. of inaccuracy(%) 0	**10**	20	**40**
LGF	0	109054(514)	**110698(469)**	113917(461)	**120513(479)**
LGF	300	109420(449)	**110980(477)**	114244(481)	**120840(502)**
LGF	600	109497(451)	**111085(474)**	114341(508)	**120946(538)**
GPSR	0	191395(1066)	**184554(796)**	177854(796)	**170755(1039)**
Protocol	Location Update (sec)	Routing Overhead(stdev) with various stdev. of inaccuracy(%) 80	**160**	200	
LGF	0	125546(506)	**126737(492)**	126806(523)	
LGF	300	125931(491)	**127018(553)**	127135(562)	
LGF	600	126049(499)	**127119(517)**	127210(570)	
GPSR	0	166352(1131)	**164052(1068)**	163327(1080)	

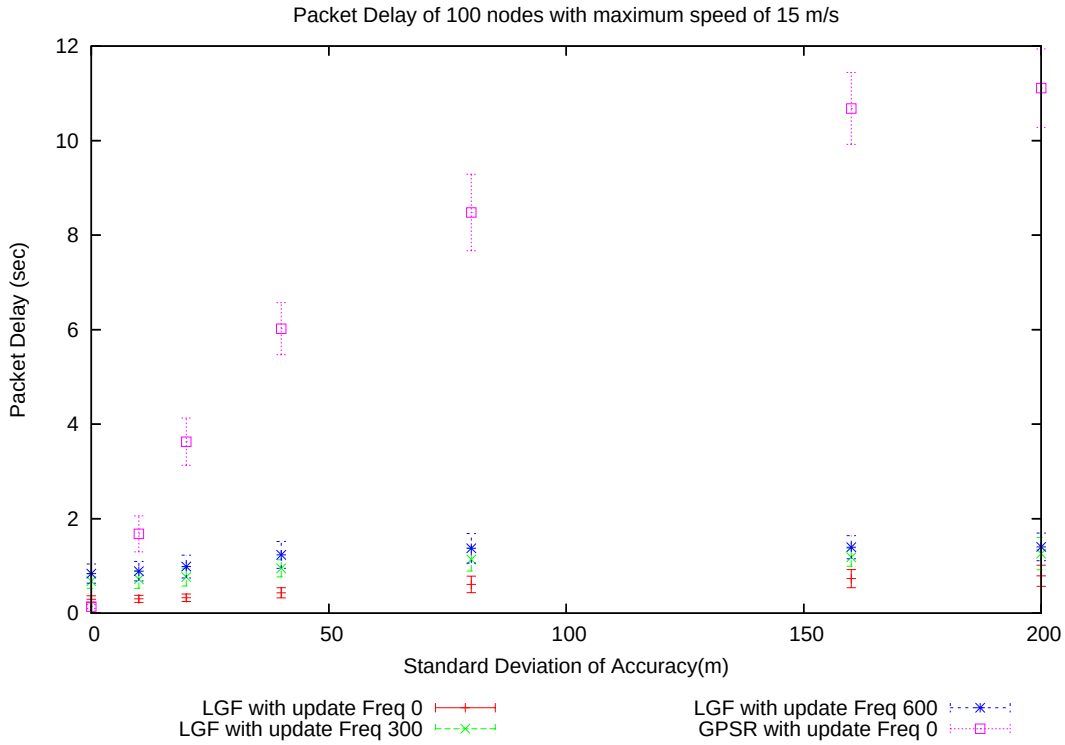


Figure 3.16: Packet delay with varying position accuracy and update frequency

Table 3.7: Packet delay with various standard deviation of position inaccuracy.

Protocol	Location Update (sec)	Packet Delay(stdev) with various stdev. of inaccuracy(%) 0	**10**	20	**40**
LGF	0	0.29(0.073)	**0.30(0.07)**	0.32(0.07)	**0.43(0.10)**
LGF	300	0.67(0.15)	**0.71(0.18)**	0.76(0.18)	**0.94(0.18)**
LGF	600	0.83(0.20)	**0.89(0.20)**	0.98(0.24)	**1.23(0.28)**
GPSR	0	0.14(0.04)	**1.68(0.38)**	3.63(0.50)	**6.02(0.55)**
Protocol	Location Update (sec)	Packet Delay(stdev) with various stdev. of inaccuracy(%) 80	**160**	200	
LGF	0	0.60(0.17)	**0.73(0.19)**	0.79(0.22)	
LGF	300	1.13(0.24)	**1.18(0.18)**	1.26(0.33)	
LGF	600	1.37(0.31)	**1.39(0.24)**	1.40(0.29)	
GPSR	0	8.48(0.81)	**10.66(0.76)**	11.11(0.83)	

3.8 Conclusions and Future work

A variety of ad hoc wireless networking solutions have been proposed to overcome the challenge of infrastructure-less systems. The most recent developments have moved towards position-based forwarding solutions which leverage geographical information in order to build routing tables and forward data amongst nodes. The most well known position based forwarding protocol is GPSR, which adopts a hybrid approach utilising greedy packet forwarding and perimeter routing to ensure a highly successful completion rate for system-wide packet forwarding.

Current solutions, however, are not resilient to location errors in the system. Location errors occur as a result of device location detection inaccuracies such as GPS tracking, which can range

inaccuracy from 10 to 100m depending upon the environment. Other approaches include GSM and/or 802.11 base station triangulation to detect the location of nodes, but all introduce a significant element of inaccuracy. In addition to location coordinate errors, we find that systems also typically experience error as a result of stale state within the Location Server that stores mappings of node ID to current location, and is actively updated by the nodes themselves. For nodes that are mobile and potentially moving at a fast rate, we find that a slow update frequency of the location server by a node can significantly impact the performance of the overall system in routing packets to a destination.

We have designed and simulated Landmark Guided Forwarding, a hybrid routing protocol that leverages position-based forwarding techniques and introduces some new elements in order to build a more robust routing algorithm in the face of position inaccuracy. *Restrictive Hybrid Route Advertisement* is used to increase the local knowledge area of each node, limiting the scope to a small number of hops. *Landmark* forwarding is used to guide packets on a higher level to simplify routing decisions at each node. *Path Exploration* is used to allow recovery of packets from dead-end routes via maintaining soft forwarding state in the nodes as well routing history in the packet header. The final feature of LGF that separates it from existing positional approaches is the *adaptive neighbourhood update* algorithm that allows a node to increase the rate at which it informs its neighbours of its current position when it is likely to encounter a link failure, thereby increasing the local resilience to movement of nodes.

Through extensive simulation of LGF and GPSR, we have demonstrated that LGF is much more resilient to location errors. The resilience can be primarily accounted for by the *Restrictive Hybrid Advertisement* features of the algorithm which provide the ability to search the local area for a node and therefore increase the success rate of the overall system. We have demonstrated that in the basic case where location accuracy is perfect, LGF performs relatively similarly to GPSR, only introducing minor overhead, but as the precision is decreased we show that LGF can outperform GPSR in terms of the packet delivery ratio and average packet delay.

Our results show LGF is able to tolerate low positional accuracy with a standard deviation of up to 200 metres, even with a maximum node velocity of 15 m/s, and no pause time. In addition, our simulation results demonstrate LGF is resilient to a combination of 200 metres standard deviation of positional inaccuracy and an update frequency of 600 seconds, again with a maximum velocity of 15 m/s and 0 pause time. With respect to the distribution of path length, the results show LGF is able to sustain a stable distribution even under significant positional inaccuracy conditions and with slow update frequency of node locations. In conclusion, therefore, the hybrid approach used by LGF is able to sustain a high delivery ratio and low average packet delay even with substantial degrees of positional uncertainty.

With regard to future developments, we intend to extend LGF for use in cases where only partial sets of nodes have access to geographic location information. Most importantly, we would like to implement LGF in a test-bed and test its performance under real environmental conditions. This will allow us to investigate the impact on LGF of a variety of other factors, including: mobility models drawn from real world data; further measured values for position uncertainty taken from testbeds and the complexity in terms of memory and computation of LGF for real world deployment. The NS2 code discussed in this chapter will be made available to other researchers in the near future from the author's website for general use by the wider research community.

Chapter 4

The Impact of Different Mobility Patterns

4.1 Introduction

Research on MANET routing protocols in the last decade has often been evaluated by network simulation. The random waypoint model [D.Johnson & J.Broch (2001)] is the main benchmark mobility pattern used to compare different routing algorithms in the bulk of the existing research.

In the random waypoint model, nodes are first randomly placed within a geographic area defined by simulation during initialisation. After that, each node independently chooses its next destination before it moves with a constant velocity towards the destination. The velocity chosen by the random waypoint model is uniformly distributed between being static and a maximum velocity, denoted by $[0,V_{max}]$. As soon as a node arrives at the destination, it remains static for a predefined period of time("pause time") before it proceeds to the next random destination, and this process repeats until the end of the simulation.

The random waypoint model retains no history. The selection of its next destination and prevailing velocity are independent from past events. It is therefore commonly understood that this model could generate unrealistic mobility patterns, with sudden changes of speed and direction. It also does not account for any spatial and temporal correlation between each node and its neighbours, in which the direction and the speed of some nodes could be dependent on the positions and velocities of other neighbours in the surrounding area. With regard to restrictions on movement, the random waypoint model only considers a square or rectangular geographic boundary and assumes only a free space scenario. It falls short of modelling a more realistic scenario, with node movement restricted to some areas within a boundary.

More realistic mobility models have been discussed in several recent research papers. In one project called *IMPORTANT* [Bai *et al.* (2003)], researchers systematically analysed the impact of different mobility patterns on the performance of routing protocols for ad hoc networks. The researchers presented several mobility models: first, a group mobility model to simulate multiple groups of mobile nodes moving in a conference area; second, it presents a freeway model with the interesting property that vehicles moving in the same direction maintain quite stable connectivities with each other, whilst vehicles moving in the opposite direction can only have relatively brief contact with one another; third, a model that simulates moving vehicles in the Manhattan's grid. In addition to modelling vehicles moving in opposite directions, the model

also considers other moving traffic across lanes.

Besides of the varity in choice of mobility model, there are many other factors that could affect the routing performance evaluation. For example, the network traffic pattern could be one of the contributing factors for routing performance. The wireless propagation model could also be a very important factor in contributing to the overall routing performance. However, in order to isolate the problem and focus only on the mobility model issue, we have chosen to use the same set of random network traffic patterns throughout all simulations. In addition, akin to other related work, we have simplified the radio propagation model by limiting it to a circular area with a radius of 250 metres.

In chapter three, we bench-marked Landmark Guided Forwarding (LGF) against other protocols using the random waypoint model. In this chapter, we use the mobility patterns generated by the software tool from *IMPORTANT*, using their mobility models to evaluate the performance of the LGF and GPSR protocols.

In the following section, we discuss related work. We then present the cumulative frequency distributions for the degree of connectivity and also for the life time of the wireless link. These distributions are used to profile the Random Way-Point (RWP), the Reference Point Group Mobility (RPGM), the Freeway (FWY) and the Manhattan Mobility (MAN) models. We analyse how different mobility patterns affect position-based forwarding protocols such as GPSR [B.Karp & H.T.Kung (2000)] as well as the hybrid solution used in Landmark Guided Forwarding. This is then followed by a section describing the simulation before we make our conclusion of this chapter.

4.2 Related Work

Until recently there has been little research effort concerning realism in mobility patterns for evaluating wireless ad Hoc networks. One exception is the *IMPORTANT* project [Bai *et al.* (2003)], in which researchers propose several mobility models that are inspired by real-world mobility scenarios. These researchers also developed a software tool to generate traces to drive simulations for their proposed RPGM, MAN and FWY mobility models. There are several published articles related to this research project [Bai *et al.* (2003)] [Son *et al.* (2004)] [Sadagopan *et al.* (2003)], which we will summarise and analyse briefly, next

In one of the papers [Bai *et al.* (2003)], the authors outline a series of mobility metrics and analyse the impact of various mobility models with DSR, DSDV and AODV. The findings shows the RPGM model in general is more well connected and has a longer average link duration, where the FWY mobility model has fewer degrees of freedom and has the poorest average link duration across different maximum velocities [Bai *et al.* (2003)].

In another paper [Son *et al.* (2004)], the researchers analyse the performance of the position-based forwarding protocol, GPSR, with FWY, MAN and RPGM mobility models. The paper identifies two main issues of position based forwarding: first, the loss of wireless link (LLNK) that results from the next hop neighbour moving out of radio range; secondly, packet looping (LOOP) results from the destination moving away from its last known position.

Researchers then proposed [Son *et al.* (2004)] Neighbourhood Location Prediction (NLP) and Destination Location Prediction (DLP) to address the LLNK and LOOP problems, respectively.

The researchers showed that location prediction could be used to solve the LLNK and LOOP in predictable mobility scenarios, such as MAN, FWY and RPGM mobility models. The NLP technique is to resolve the LLNK issue by blacklisting neighbours that are predicted to be out of the wireless range. It suggests every node should advertise its beacon advertisement time, along with x co-ordinate, y-co-ordinate, speed and direction to all its one hop neighbours in the surrounding area. The forwarding algorithm leverages this information to avoid using a neighbour as a forwarding node if the calculation at the moment suggests it is already out of the radio range.

A similar approach is also used by DLP to resolve the LOOP problem. DLP predicts the location of the destination when selecting the next hop for packet delivery. The forwarding algorithm uses its single hop forwarding table together with time stamped geographic information encoded in the packet header to decide to which neighbour the packet should be forwarded. It forwards the packet to the destination if the calculation shows that the destination is still within its radio range. Otherwise, it forwards the packet to a neighbour that has the closest Euclidean distance to the last estimated location of the destination.

The proposed NLP and DLP protocols were evaluated with the FWY, MAN and RPGM mobility models. Although the results show location prediction could be used to address both LLNK and LOOP problems found in position-based forwarding, the proposed methods require high precision clock synchronisation across the entire wireless network, which seems to be a stringent constraint. In addition, it was assumed that all nodes maintain constant velocity, and the system has precise knowledge of the radio wave propagation at the physical layer.

Another paper from the *IMPORTANT* project presents the frequency distribution of link and path connectivity with respect to the FWY, MAN, RPGM mobility models [Sadagopan *et al.* (2003)]. Its conclusion matches our intuition on how the link and path should respond to increasing velocity. The simulations were conducted with maximum velocity of 0, 10, 20, 30, 40, 50 and 60 m/s. They conclude that an increase in the node maximum velocity could reduce the average link and path duration. It also reinforces their previous findings from other papers that the RPGM model indeed has longer established link and path duration than the FWY models, across the full range of maximum velocities used in the simulation.

Besides the *IMPORTANT* project, there are a few other relevant papers in this area. In fact, the RPGM model was proposed a few years earlier by Hong et al [Hong *et al.* (1999)] [Hong *et al.* (2001)]. These researchers studied the rate of change of wireless link quality on RWP and RPGM models. Their results show that the RPGM model has smaller link capacity change than the RWP model when both models are benchmarked with Destination Sequenced Distance Vector (DSDV) routing protocol [C.Perkins & P.Bhagwat (1994)]. Their results are consistent with the findings from other related work we have discussed.

4.3 Mobility Model

Based on the related work that has been discussed, it is unrealistic simply to evaluate the performance of a routing protocol based solely on the RWP model. It is clear that the different mobility models are more realistic, and, crucially, will present very different temporal and spatial correlation between nodes. This undoubtedly will have a profound impact on the maintenance of established routes for topological-based routing. On the other hand, the position-based routing protocol advocates local routing where the forwarding decision is made by comparing the proximity of all its one hop neighbours to the destination, and does not need to establish or

maintain a path between a source and a destination. This advantage, however, can be overshadowed by the inability to handle positional inaccuracy, or inconsistent position information due to mobility conditions. In this section, we give a brief introduction to the FWY, MAN and RPGM models, which then we then use to evaluate the performance of Landmark Guided Forwarding and GPSR.

4.3.1 Freeway (FWY) Model

In the *IMPORTANT* project, researchers proposed a model to simulate moving vehicles on the freeway. Typically, the freeway model consists of two sets of multiple tracks running in parallel, with each set of tracks serving the travelling motor vehicles in opposite directions to one another. The key distinctive feature of this model is that the wireless link established by vehicles travelling in the same orientation sustains longer link connectivity than the wireless link established by vehicles moving in the opposite directions. This is because vehicles travel in the same direction at a comparatively slow speed relative to one another. For example, if A and B are vehicles travelling on the freeway in the same direction, vehicle A is travelling at 40 km/hr while B is travelling at 45 km/hr. The velocity between A and B in this case is the difference in velocity between two vehicles. Based on simple mathematical subtraction, the relative velocity between node A and B is 5 km/hr. In contrast, the relative velocity between A and B can be obtained by summing the velocity of two vehicles when A is travelling in the opposite direction to B, which should be 85 km/hr. It is apparent that the greater the resultant velocity from two opposing vehicles, the more likely that the wireless link will break.

Another interesting property of FWY is the temporal correlation between two vehicles travelling along the same track. The model not only incorporates a safety distance in between two travelling vehicles on the same track but also ensure the velocity of vehicle behind be sufficient to overrun the leading vehicle.

Beside this, the restriction of vehicle movement along the freeway also implicitly provides the spatial correlation property of travelling vehicles in the surrounding area, both in the same and opposite directions.

4.3.2 Manhattan(MAN) Model

Besides the FWY model, modelling the mobility pattern of vehicles in an urban area seems to be a natural piece of work for the entire ad hoc mobility study. This is normally represented by the Manhattan (MAN) model, typically consisting of vertical and horizontal collections of two way streets crossing each other. This model has some features similar to the FWY models. For example, the two way street is made up by two sets of multiple tracks, in parallel but in opposing directions to each other. In addition, it also incorporates a safety distance in between two vehicles and prevents the following vehicle from over running the preceding vehicle. Apart from this similarity, the MAN models have an unique property that can not be found in FWY. Distinctively, it has cross junctions at each intersection of horizontal and vertical streets which allow more freedom of movement than the FWY model. For each vehicle arriving at the junction, a vehicle can turn left, turn right or continue going straight. The model assumes the probability of a vehicle making a left turn is 0.25. Likewise, the probability of vehicle turning right is also 0.25. The remaining fraction of 0.5 is assigned to the probability of vehicle staying on the same course at the junction.

The MAN model inherited some of the mobility primitives from FWY model and therefore it retains the temporal correlation between two vehicles, when one vehicle follows another moving on the same track toward the same direction. Similar to the FWY model, the MAN model also imposes restrictions on vehicle mobility. However, the MAN model has slightly more degrees of freedom and therefore could be less spatially correlated than the FWY model

4.3.3 Reference Point Group Mobility (RPGM) Model

As often happens in a conference, guided tour or combat operation, mobile nodes move in small groups, with members of each group following a leader. The Reference Point Group Mobility (RPGM), is a model that emulates user mobility with this kind of collective behaviour. Essentially, in the RPGM model, members of each group uniformly distribute around the group leader at the beginning of the simulation. Each member then constantly associates its next movement with the position of their group leader, normally with slight random deviation away from where the leader is. Apart from this, every group has its own trajectory and is independent of each other.

In general, there is a strong spatial correlation amongst wireless nodes within the same group. In contrast, the spatial property of nodes from different groups are less likely to have a strong correlation. This may explain why the overall spatial correlation decreases when the number of RPGM groups increases from one to four. This also explains why using lesser group in the RPGM model yields longer wireless link duration on average [Sadagopan *et al.* (2003)].

4.4 Simulation Scenarios

To test the robustness of Landmark Guided Forwarding (LGF) with different mobility models, we generated several sets of RWP, FWY, MAN and RPGM mobility models for subsequent NS2 simulation. We ran two sets of simulations with 60 nodes in an area of 1000 x 1000 metres square with 20 sets of traffic scenarios. Each simulation lasted 900 seconds. Each node used the IEEE 802.11 MAC and physical models with the radius of the radio range being 250 metres. In all simulation scenarios, the traffic model used constant bit rate UDP traffic flows, with 512 byte payloads. The start time for the different flows is uniformly distributed between 0 and 180 seconds with each of the 30 traffic sources at the rate of 2 packets per second. We compare LGF with GPSR using the above mentioned mobility models. For the first set, we ran a series of simulations with different maximum velocities of 1, 5, 10 and 20 m/s at 0 pause time. We then continued with the second series of simulation with different standard deviations of position accuracy of 0, 10, 20, 40, 80, 160 and 200 metres, with maximum velocity set to 10 m/s and pause time being configured to 0.

Finally, we have created a set of mobility pattern profiles from the RWP, FWY, MAN and RPGM models, for use in our simulations. We use these profiles to study the degree of connectivity and wireless link duration with each of these mobility models. This study not only provides us with a general understanding of how well these mobility models affect connectivity, but also serves as a basis for our results and discussion.

In order to generate the RPGM, MAN and FWY mobility models in NS2 format for our simulations, we require to input a set of configurations to the software tool provided by IMPORTANT. A check file or map file is used to define the maximum node velocity and other

mobility properties of the mobility models. The RPGM model uses a checkpoint file to configure the trajectory of a group. In our simulation, we use a set of checkpoints files provided by IMPORTANT. As seen in Table 4.1, each group has 4 checkpoint files. Each of these files is used to generate the respective NS2 mobility scenario format with specific maximum node velocity. For example, the check point file mg-5a is used to generate the RPGM mobility file for NS2 simulation with a maximum node velocity of 5 m/s.

Likewise, the FWY and MAN mobility models use a map file to define the trajectory of the node movement. As seen in Tables 4.2 and 4.3, map files are used for generating the NS2 mobility format with specific maximum node velocity. For example, the map file freewayb20.txt is used to generate the FWY mobility scenario file for NS2 simulation with a maximum node velocity of 20 m/s, the map file manhanttan-3-20.txt is used to generate the MAN mobility scenario file for NS2 simulation with a maximum node velocity of 20 m/s.

Each ad hoc routing protocol has some settings specific to it, which can be found in Table 2.10 and Table 2.8 of previous chapter. In addition, each mobility model has configurations which are detailed in the Tables below:

Table 4.1: RPGM specific parameters

Parameter	*Value*
Number of group	4
Number of nodes in each group	15
Speed of deviation	5 m/s
Angle of deviation	5 degree
Checkpoint file for group 1	./rpgm/checkpoint2/mg-1a
	./rpgm/checkpoint2/mg-5a
	./rpgm/checkpoint2/mg-10a
	./rpgm/checkpoint2/mg-20a
Checkpoint file for group 2	./rpgm/checkpoint2/mg-1b
	./rpgm/checkpoint2/mg-5b
	./rpgm/checkpoint2/mg-10b
	./rpgm/checkpoint2/mg-20b
Checkpoint file for group 3	./rpgm/checkpoint2/mg-1c
	./rpgm/checkpoint2/mg-5c
	./rpgm/checkpoint2/mg-10c
	./rpgm/checkpoint2/mg-20c
Checkpoint file for group 4	./rpgm/checkpoint2/mg-1d
	./rpgm/checkpoint2/mg-5d
	./rpgm/checkpoint2/mg-10d
	./rpgm/checkpoint2/mg-20d
Simulation time	900
Network area	1000 x 1000 m

Table 4.2: FWY specific parameters

Parameter	*Value*
Number of Node	60
Acceleration	3 m/s^2
Simulation time	900
Map file	./fwy/mapset2/freewayb1.txt
	./fwy/mapset2/freewayb5.txt
	./fwy/mapset2/freewayb10.txt
	./fwy/mapset2/freewayb20.txt
Network area	1000 x 1000 m

Table 4.3: MAN specific parameters

Parameter	*Value*
Number of Node	60
Maximum allowed velocity	10 m/s
Minimum allowed velocity	1 m/s
Acceleration	3 m/s^2.
Simulation time	900
Map file	./man/mapset2/manhattan3-1.txt
	./man/mapset2/manhattan3-5.txt
	./man/mapset2/manhattan3-10.txt
	./man/mapset2/manhattan3-20.txt
Network area	1000 x 1000 m

4.5 Measuring the Impact of Different Mobility Patterns

In this section, we study how each of the MAN, FWY and RPGM mobility models affect connectivity, both in terms of the degree of connectivity, and duration. We sample these attributes with various mobility models at the rate of 1 sample per second over 900 seconds of simulation time.

4.5.1 Degree of Connectivity

As with other NS2 simulations, the wireless propagation model is assumed to be a circular area with a radius of 250 metres. So, as for the degree of connectivity, only neighbours that are within the 250 metres of radius are considered to be connected with a wireless link. Table 4.5 shows the maximum, minimum and average degree of connectivity for RWP, MAN, FWY and RPGM models. Looking at the results, there seems to be a strong correlation between the maximum and average readings. The readings show that the RPGM model has the highest degree of

Table 4.4: RWP specific parameters

Parameter	*Value*
Version	1
Number of Nodes	60
Pause Time	0 sec.
Simulation time	900
Maximum Velocity	10 m/s
Network area	1000 x 1000 m

connectivity of the models, both in terms of average as well as maximum counts. This, in fact, suggests that the RPGM model has better spatial correlation than other mobility models. This could be due to the fact that members of individual RPGM group follow quite closely to the group leader and therefore there is a good chance that members are not only geographically near the group leader but also likely to be close to other fellow group members. Besides, it is also possible that at some instant of time some members of one group could have been overlapping and sharing their social space with members from another group, which may be another reason why the degree of connectivity for this model is so much superior than others.

Table 4.5: Degree of connectivity for various mobility.

Model	Maximum	Minimum	Average
RWP	32	0	14.55
MAN	29	0	11.40
FWY	40	0	27.08
RPGM	59	0	37.30

As for the FWY model, apart from its very restrictive space of movement, the model also maintains a safety distance in between vehicles and ensures no vehicle from the same track could have run into its preceding vehicle. Our results show the FWY model is fairly well connected and therefore has better spatial correlation than the MAN model.

Unlike the FWY model, apart from moving straight, the MAN model has a 0.5 chance of moving to the left or right at the cross junctions. This additional freedom of movement allows vehicles in the model to spread out geographically and may explain why the MAN model has lower degree of connectivity compared with the FWY model.

The results presented in Table 4.5 also suggest that the RWP model could statistically create a more sparse network than FWY and RPGM models but a slightly denser network than the MAN model. It emerged that the RWP model behaves very differently from the RPGM, FWY and MAN models. The RWP model generally has no space restriction and nodes are free to roam in any direction. Connectivity of the RWP model has been discussed in a research article [Li *et al.* (2000)]; the theoretical analysis shows that when the number of nodes randomly distributed through a unit kilometre square is greater than $(6/r^2)ln(6/r^2)$, where r is the radius of a circular propagation model, then partitions in the network are less likely to occur. For an area of 1000 x 1000 metres square, the theory suggests randomly scattering 438 nodes in a unit

square kilometre might be statistically sufficient to guarantee network connectivity for the RWP model. However, physical connectivity does not guarantee packet delivery: for example, the well known RTS/CTS hidden node interference issue must be considered. A result from the same paper shows that as the number of nodes increases from 100 to 600, the fraction of successful Grid Location Service queries reduces from 0.98 to 0.90.

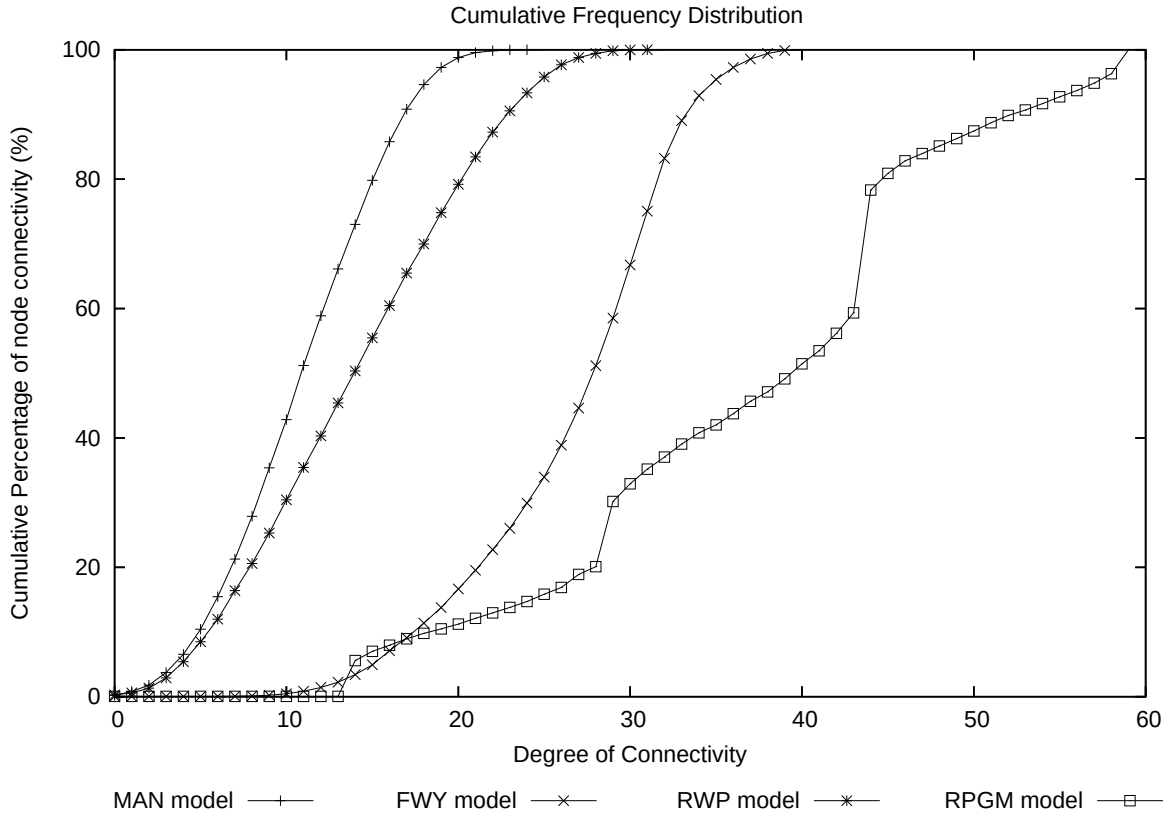


Figure 4.1: Cumulative frequency distribution of degree of connectivity for various mobility models

Figure 4.1 shows the cumulative frequency distribution of degree of connectivity for the RWP, FWY, MAN and RPGM models. The RPGM model has the highest average degree of connectivity of the mobility models in use.

The figure shows the FWY model to have a better distribution of degree of connectivity compared to both MAN and RWP models, with the RWP model just slightly more connected than the MAN model. These distributions are consistent with the average and maximum readings found in Table 4.5.

There are a few other noteworthy features of the RPGM model. The graph in Figure 4.1 shows there are several step changes in the RPGM model. In order to discuss this clearly, we present the details of cumulation frequency distribution for the RPGM model in Table 4.6. At the first step change found between 13 to 14 degrees of connectivity, the distribution changes from 0.00% to 5.58%. The step change also shows that 5.58% of the distribution consists of 14 degree of connectivity. The RPGM model in use was configured with 4 groups. Since each group was assigned 15 members, it seems likely that this 5.58% could be the time where a group

is isolated from all other groups. A snapshot of this scenario is shown in Figure 4.2.

Table 4.6: Cumulative frequency distribution of degree of connectivity for RPGM model

Cumulative frequency Distribution of Degree of Connectivity for RPGM (%)									
Degree	**Dist.**	Degree	**Dist.**	Degree	**Dist.**	Degree	**Dist.**	Degree	**Dist.**
0	**0.00**	1	**0.00**	2	**0.00**	3	**0.00**	4	**0.00**
5	**0.00**	6	**0.00**	7	**0.00**	8	**0.00**	9	**0.00**
10	**0.00**	11	**0.00**	12	**0.00**	13	**0.00**	14	**5.58**
15	**7.02**	16	**8.00**	17	**8.94**	18	**9.79**	19	**10.50**
20	**11.23**	21	**12.11**	22	**12.96**	23	**13.79**	24	**14.72**
25	**15.83**	26	**16.88**	27	**18.90**	28	**20.10**	29	**30.18**
30	**32.90**	31	**35.17**	32	**37.04**	33	**39.05**	34	**40.02**
35	**42.03**	36	**43.74**	37	**45.66**	38	**47.13**	39	**49.13**
40	**51.46**	41	**53.46**	42	**56.18**	43	**59.33**	44	**78.30**
45	**80.90**	46	**82.84**	47	**83.94**	48	**85.13**	49	**86.27**
50	**87.44**	51	**88.72**	52	**89.84**	53	**90.69**	54	**91.68**
55	**92.74**	56	**93.70**	57	**94.85**	58	**96.31**	59	**100.00**

Based on knowledge of group size, we deduce other group connectivity information from a small section of the distribution. For example, we could deduce from the distribution in between 46 to 59 degrees of connectivity and find the amount of time when members of 4 groups overlapping each other. We found 4 groups were overlap each other for about 17.16% of the simulation time. A snapshot of this scenario is shown in Figure 4.3. This could be one of the contributing factors for the superb connectivity of the RPGM model.

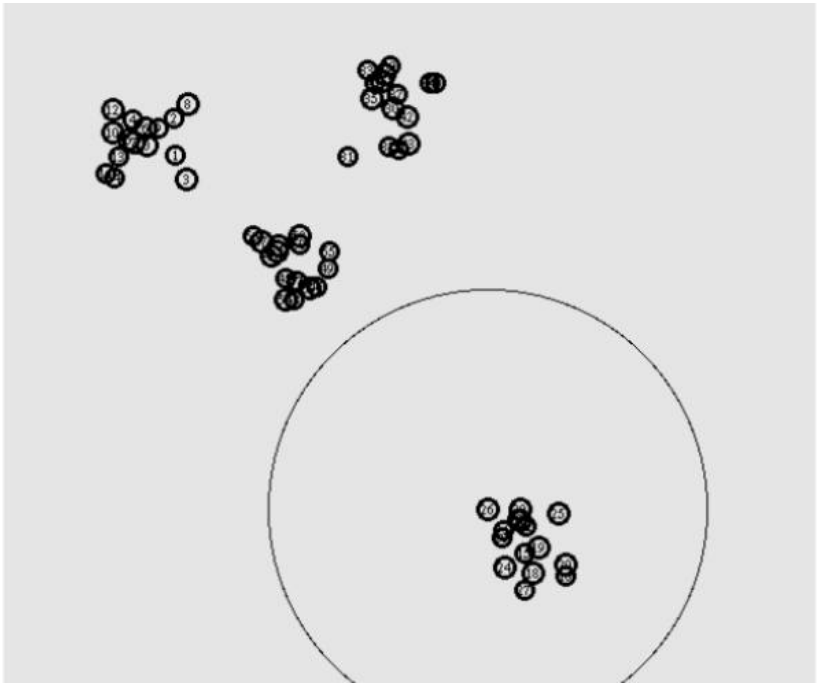


Figure 4.2: Isolated from other groups

4.5.2 Life Time of Wireless Link

As before, a circular area with a radius of 250 metres is used as a basis for modelling wireless propagation. For the link up time, a link between a node and its neighbour is considered to be

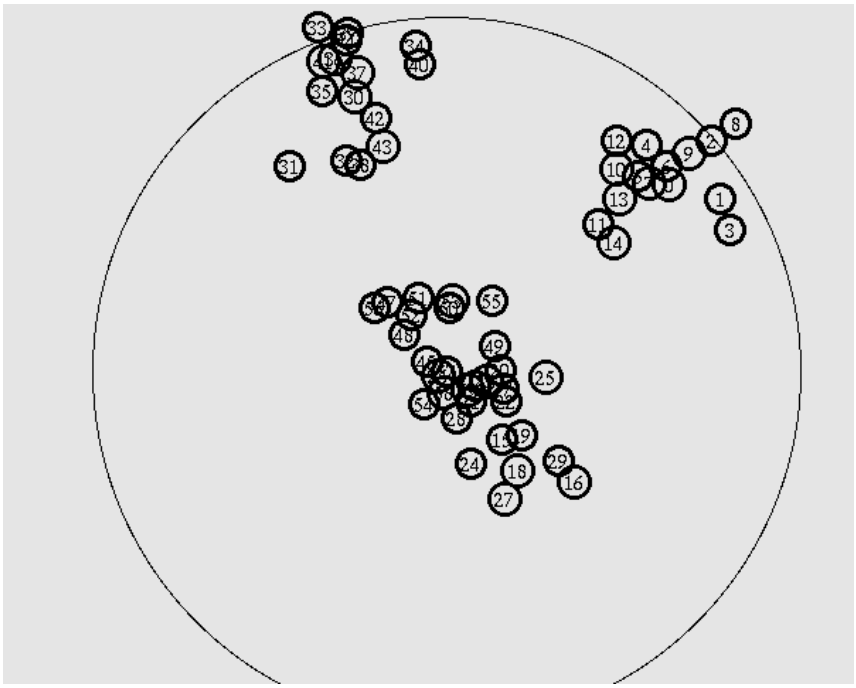


Figure 4.3: Overlapping with other groups

active as long as the distance between these two nodes is within a radio range of 250 metres. Otherwise, the link is considered to be disconnected. Table 4.7 shows the maximum, minimum and average life time for wireless link with RWP, MAN, FWY and RPGM mobility models over a simulation time of 900 seconds.

Table 4.7: Link duration for various mobility models

Model	Maximum	Minimum	Average
RWP	900	1	114
MAN	900	1	68
FWY	691	1	72
RPGM	900	1	126

The results show that, on average, the RPGM model has the longest link duration amongst all mobility models. The RWP model comes next, this is followed by the FWY model and then the MAN model. In theory, members of the RPGM follow where their leader goes, and move along together as a group. Although this potentially encourages longer duration of the wireless link, there are cases where groups overlap each other, increasing the likelihood that members from different groups will move away from each other. A similar study by other researchers [Sadagopan *et al.* (2003)] shows that an increase in the configuration of maximum velocity could decrease the life time of wireless links. Their results show that this effect can be consistently reproduced in the RPGM, MAN, FWY models .

Both the MAN and FWY models have quite similar temporal correlation since both models maintain safety distance between vehicles and ensure that no vehicle overruns a preceding vehicle by regulating the speed of the following vehicle. This may just explain the similarity of

their readings on average link duration.

The MAN model has a different model of mobility restriction as compared to the FWY model. Unlike the FWY model, where nodes only move along a straight path, the MAN model has more degree of freedom at each junction of the Manhattan grid model. Most important, the 1000 x 1000 metres square area used in our FWY, MAN and RWP simulations would wrap around when a node exceeds the predefined borders. This is more critical for the FWY model since it moves only in a straight path and therefore is more likely to exceed the borders than the RWP and MAN models. Our results show maximum link duration for FWY is 691 seconds which is the lowest amongst all models. Our results show other models sustained at least one wireless link throughout the 900 seconds of simulation time. As for the RPGM model, we notice the checkpoint files provided by the project *IMPORTANT* avoid moving the groups out of the boundary and therefore would never be in the wrapped around situation.

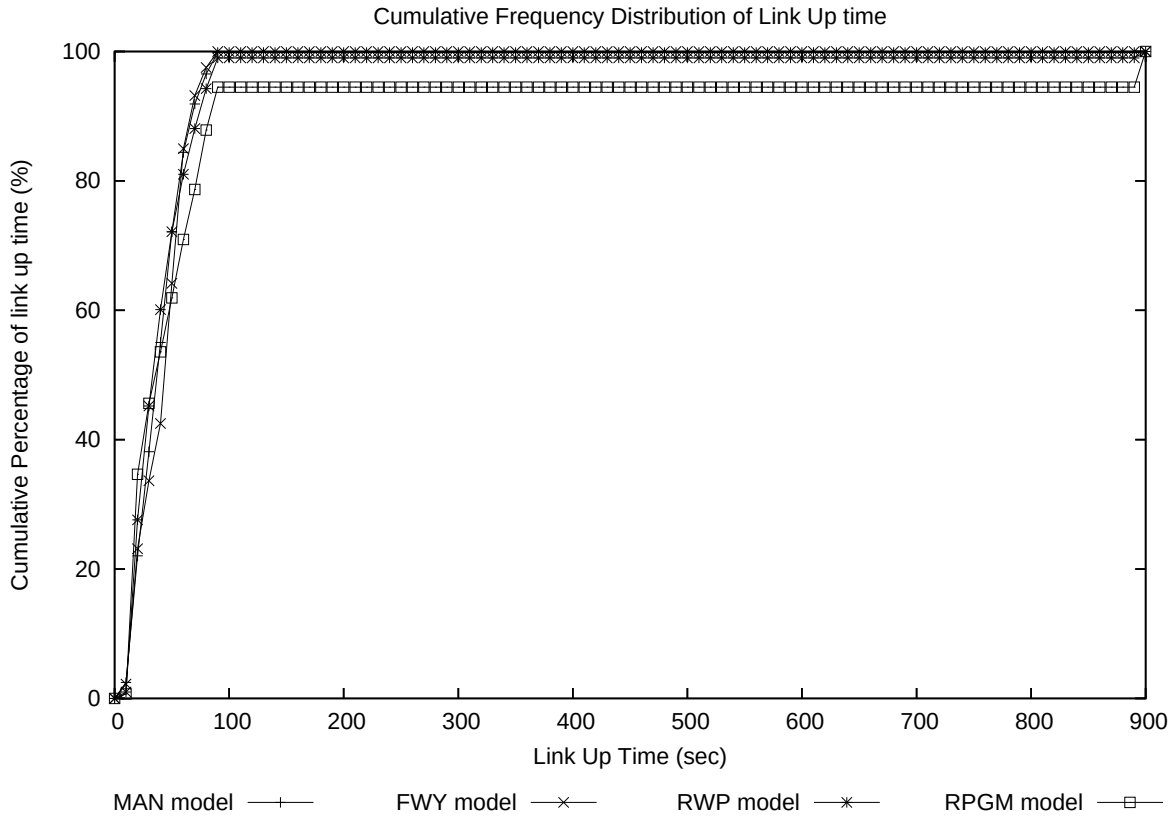


Figure 4.4: Full frequency distribution of link up time with various mobility models

Figure 4.4 is a cumulative frequency distribution of life time of wireless link for RWP, FWY, MAN and RPGM models with maximum velocity of 10 m/s. The results show the distribution of link duration for all models are similar when link duration is below 80 seconds. Above 80 seconds and beyond, every model appears to experience different amount of sustaining wireless links. Results from other related work show that the RPGM model has a longer average link duration than the MAN, FWY and RWP models. This is consistent with our results in Table 4.7 as well as the cumulative frequency distribution in Figure 4.4.

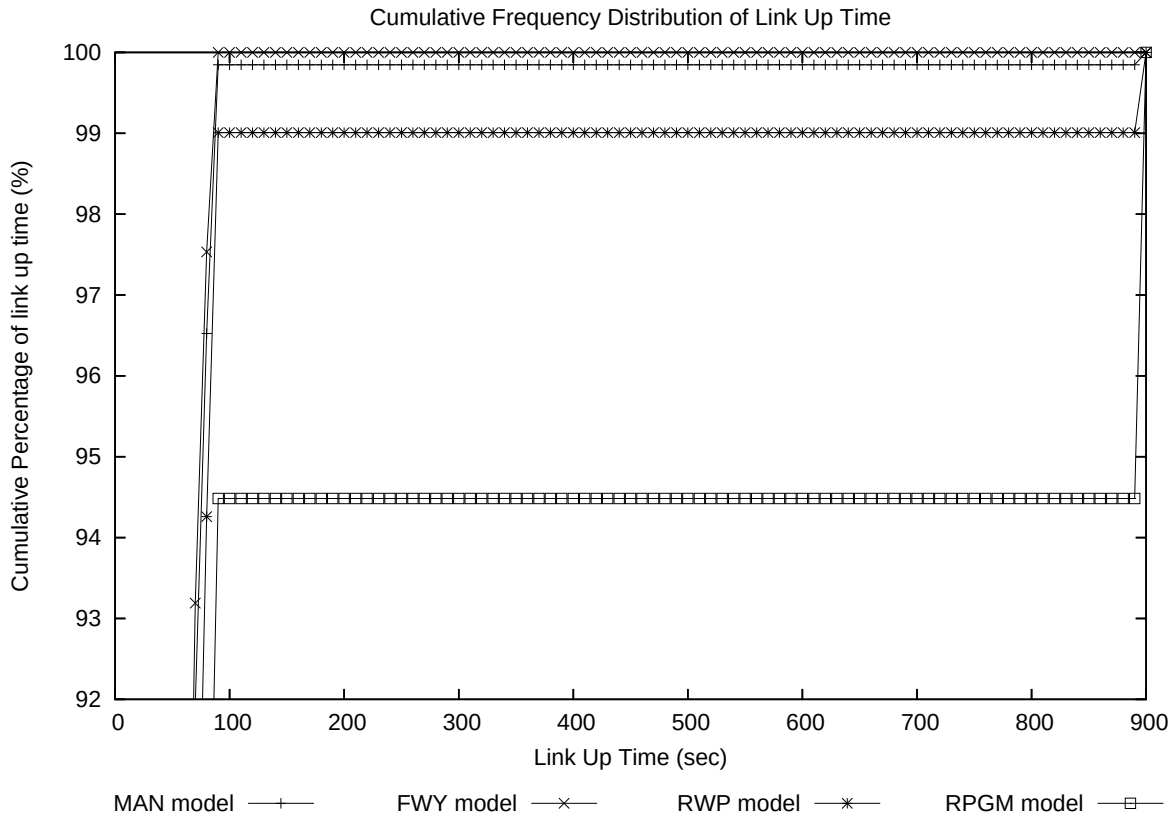


Figure 4.5: Partial frequency distribution of link up time with various mobility models

In order to have a clearer picture of the cumulative frequency distribution when link duration is more than 80 seconds, we zoom into the top potion of Figure 4.4, as shown in Figure 4.5. The figure clearly shows that the RPGM model has 5.5% of its simulation time sustaining links as long as the entire simulation duration of 900 seconds. This is followed by the RWP model that has 1% of its simulation time sustaining links for the duration of 900 seconds, whilst the MAN model has only 0.2% of its simulation time sustaining links with the same duration. This figure shows none of link from the FWY model has sustained similar duration. In fact, the longest link duration ever sustained by the FWY model is 691 seconds, as previously presented in Table 4.7.

4.6 Results and Discussion

Knowing the connectivity at the physical layer is helpful for our assessment of the routing protocol. We run two sets of simulations with these mobility models: first with various maximum velocity; this is then followed with different levels of position inaccuracy with a maximum velocity of 10 m/s.

4.6.1 Performance with Varying Maximum Velocity

Figures 4.6 and 4.7 show the packet delivery ratio of LGF and GPSR, with the RWP, FWY, MAN and RPGM models, alongside various maximum velocity settings over a simulation time of 900 seconds. The results show both LGF and GPSR perform similarly when simulated with RWP, MAN and RPGM models. However, the results show LGF performs better than GPSR

when simulated with the FWY model.

For example, with the MAN mobility model, the LGF successfully delivers approximately 50% of the packets when the maximum velocity increases from 1 m/s to 20 m/s. Similarly, the GPSR also delivers approximately 50% of the packets when the maximum velocity increases from 1 m/s to 20 m/s. With the RWP model, the LGF successfully delivers above 80% of the packets when the maximum velocity increases from 1 m/s to 20 m/s, and the GPSR also delivers above 80% of the packets when the maximum velocity increases from 1 m/s to 20 m/s.

In the case of the RPGM model, the LGF packet delivery ratio increases from approximately 95% to 99% when the maximum velocity increases from 1 m/s to 5 m/s, and the GPSR packet delivery ratio also increases from approximately 95% to 99% when the maximum velocity increases from 1 m/s to 5 m/s. In fact, the LGF and GPSR performance are similar when the maximum node velocity is 15 m/s or below. Our results showed the LGF performs slightly better than GPSR in the scenario of RPGM model when the maximum node velocity is configured to 20 m/s. The LGF average packet delivery ratio is approximately at 99%, while the GPSR average packet delivery ratio is 93%. This is likely to be caused by the effects of position inconsistency on GPSR when maximum node velocity is increased to 20 m/s.

As we discussed in section 4.5, the RPGM model has a strong spatial correlation. It has a high degree of connectivity as well as longer established wireless connectivity. The RPGM model in use has 4 groups with 15 members in each group and members of these groups could be overlapping one another at different instances of simulation time. Besides this, the RPGM group members follow where the group leader goes: the faster the group moves amongst other overlapping groups, the more likely the last known positions of the neighbours become outdated. In fact, this compromises the integrity of the greedy forwarding principle as well as the construction of the planar graph upon which the guaranteed delivery of geographic based forwarding is relying. For example, providing an outdated geographic position to the neighbour could create a situation in which the greedy or the perimeter forwarding picks the wrong neighbour, which could result in packet routing through a less optimal path, thereby increasing the likelihood of packets being trapped in dead ends or loops.

The results in general show the delivery ratio of both protocols are very competitive when simulated with RWP, MAN and RPGM models with different settings of maximum node velocity.

The results with the FWY model shows that both LGF and GPSR packet delivery ratio decreases when maximum node velocity increases. The LGF packet delivery ratio decreases from approximately 95% to 82% when the maximum velocity increases from 1 m/s to 20 m/s, while the GPSR packet delivery ratio decreases from approximately 95% to 48% when the maximum velocity increases from 1 m/s to 20 m/s.

Although all simulations are performed with 60 nodes, the results show the protocols can respond differently on different mobility models. In the FWY model, the sheer restriction of the road space and the opposing movement of neighbour vehicles could be the contributing factor for position inconsistency. For example, the FWY model has a relatively high degree of connectivity count, but the restrictive road space could only allow large numbers of neighbours to be positioned in front or behind it. Besides, there are also vehicles travelling in an opposing direction that make the accuracy of a node position and the correctness of a planar graph construction difficult to sustain. Furthermore, rather than forwarding the packet to a neighbour

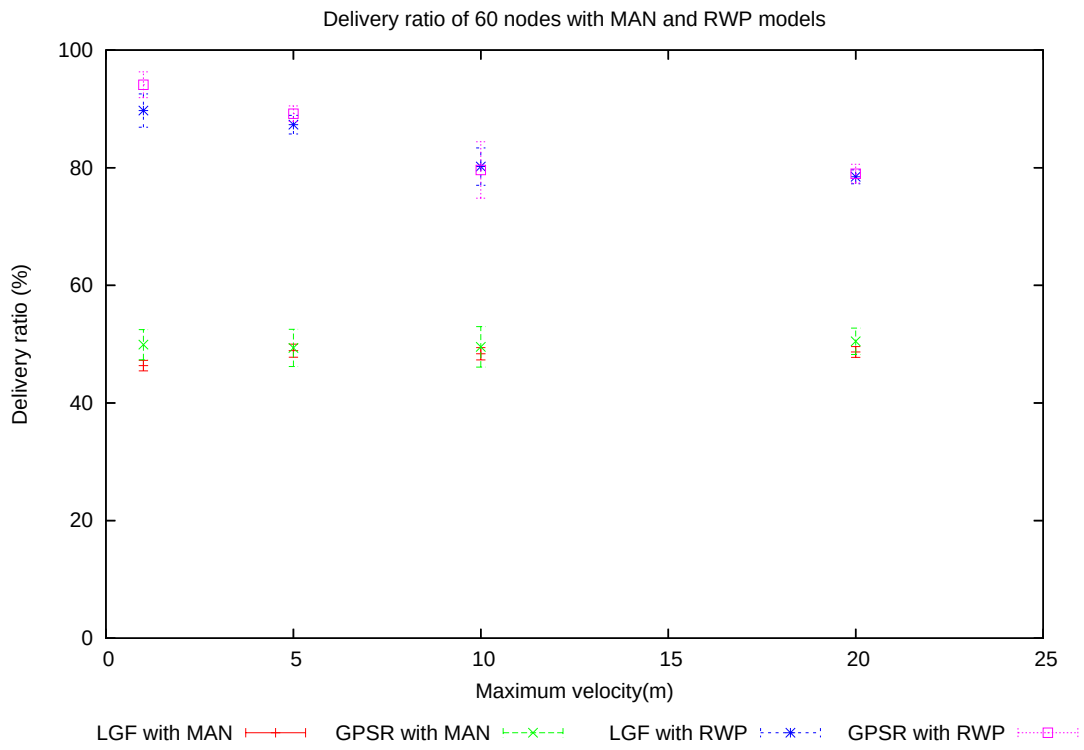


Figure 4.6: Packet delivery with varying position accuracy for MAN and RWP models

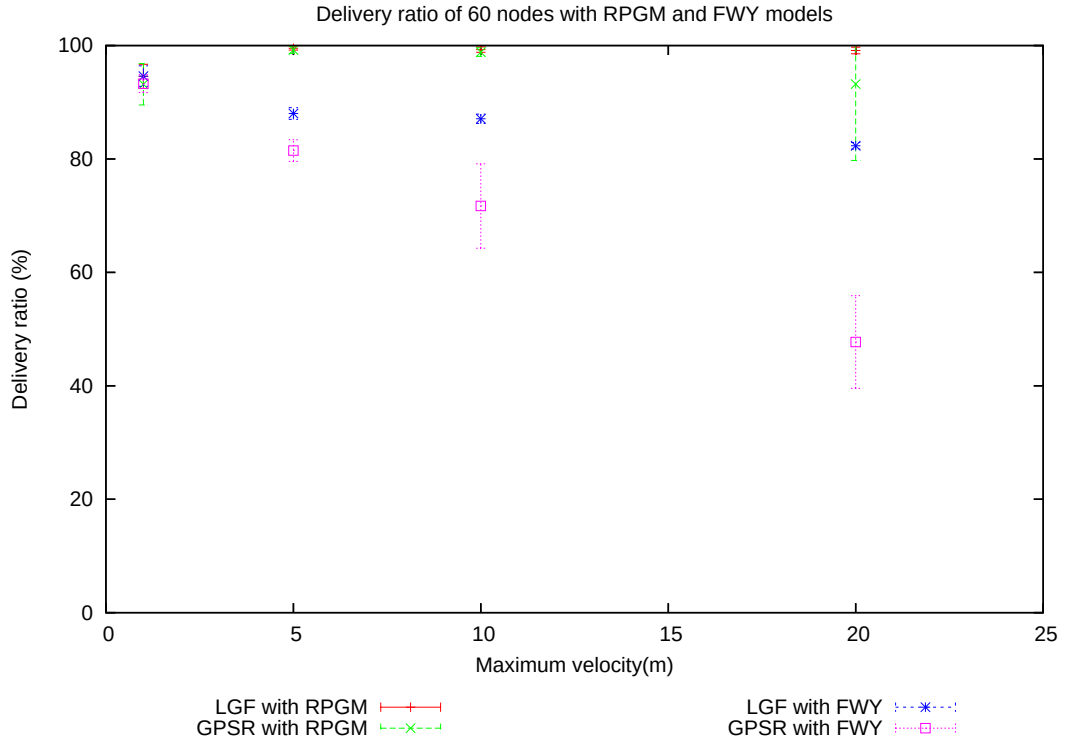


Figure 4.7: Packet delivery with varying position accuracy for FWY and RPGM models

that is travelling towards the destination, there is also a likelihood that the packet could be forwarded to a neighbour that is travelling in the opposing direction to the destination. The results show that GPSR routing performance degrades when the maximum node velocity of the FWY increases, while the LGF is able to operate with little performance loss.

Interestingly, when we compare the successful packet delivery ratios amongst various mobility models, it emerges that the LGF performs much better under the RPGM model than all other models. This is followed by the RWP model, followed by the FWY model, with the MAN model coming in last.

This appears to be in exactly the same order as the average link duration in Table 4.7. Theoretically, it seems that there is a strong correlation between the link stability and the packet delivery ratio. In general, a longer average link duration is less likely to disrupt the forwarding algorithm and thereby yields better results in the packet delivery ratio. However, more needs to be done before drawing this conclusion. This is left for future work.

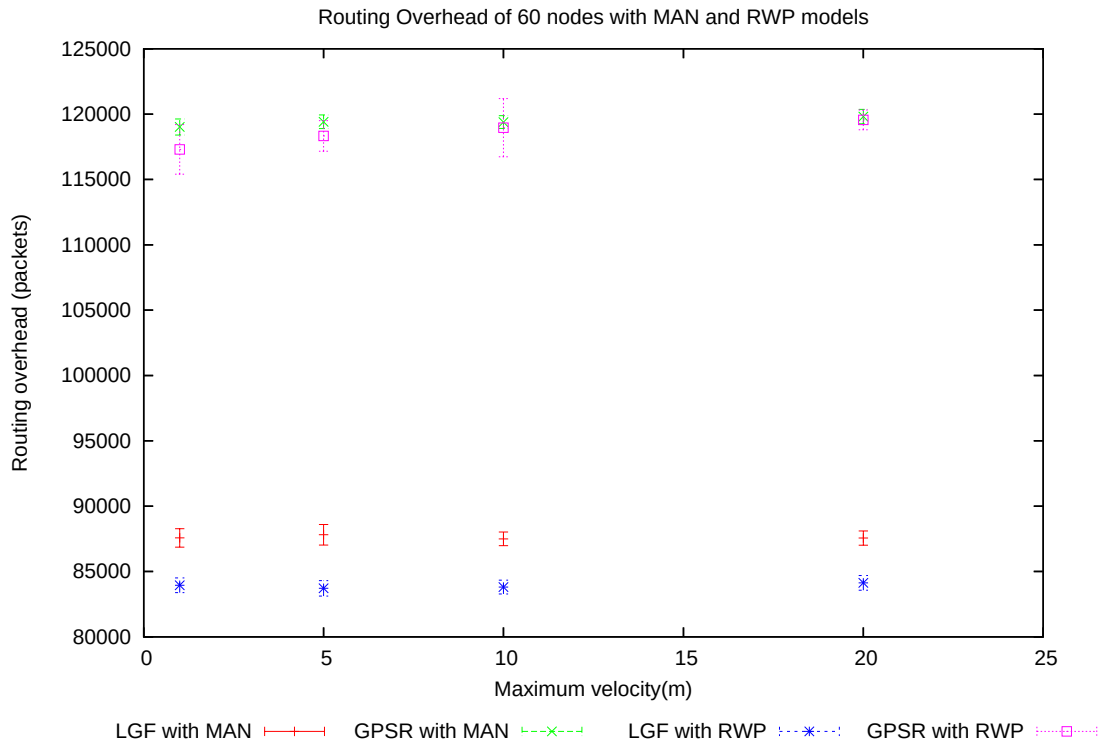


Figure 4.8: Routing overhead with varying position accuracy for MAN and RWP models

Figures 4.8 and 4.9 show the routing overhead of the LGF and GPSR, with RWP, FWY, MAN and RPGM models, alongside various maximum velocity over a simulation time of 900 seconds. The results show GPSR in general incurs more routing overhead than LGF.

For example, with the MAN mobility model, the LGF routing overhead is maintained at approximately 87900 packets when the maximum velocity increases from 1 m/s to 20 m/s, while the GPSR routing overhead is maintained at approximately 119000 packets when the maximum velocity increases from 1 m/s to 20 m/s. With the RWP model, the LGF routing overhead is maintained at approximately 84000 packets when the maximum velocity increases from 1 m/s to

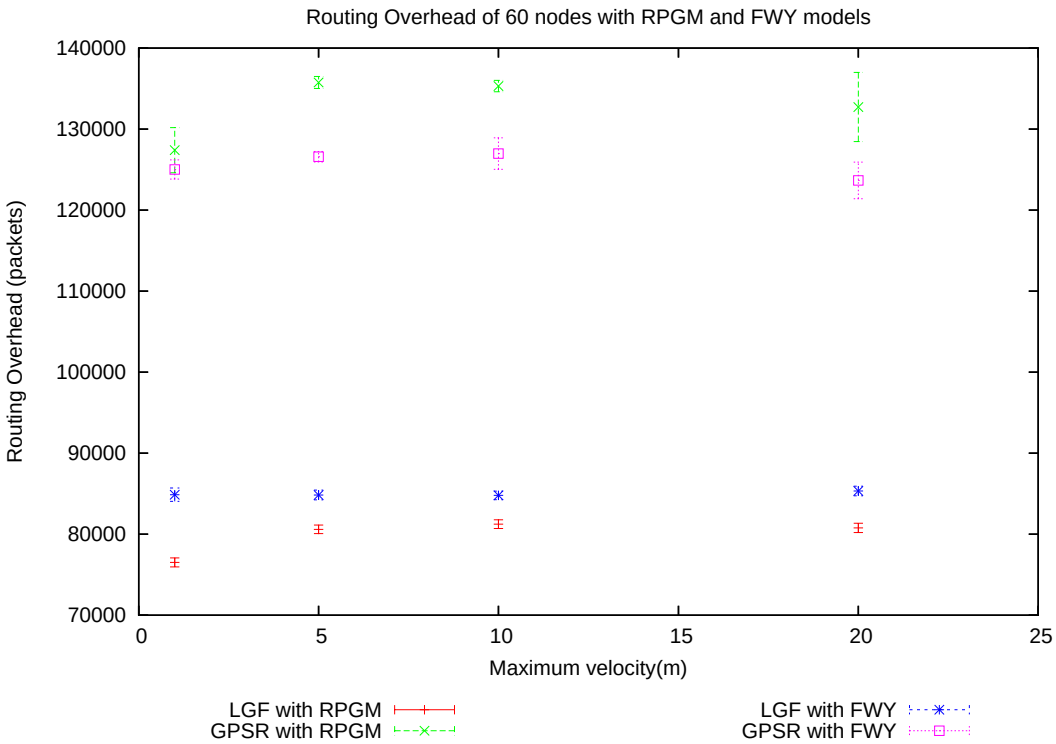


Figure 4.9: Routing overhead with varying position accuracy for FWY and RPGM models

20 m/s, while the GPSR routing overhead is maintained at approximately 118000 packets when the maximum velocity increases from 1 m/s to 20 m/s. With the FWY model, the LGF routing overhead is maintained at approximately 84000 packets with various settings of maximum velocities, while the GPSR routing overhead is maintained at approximately 125000 packets when the maximum velocity increases from 1 m/s to 20 m/s.

In the case of the RPGM model, the LGF routing overhead is maintained at about 80000 packets when the maximum velocity increases from 5 m/s to 20 m/s, while the GPSR routing overhead is maintained at about 135000 packets when the maximum velocity increases from 5 m/s to 20 m/s. Both GPSR and LGF show lower routing overheads when simulated with RPGM model at 1 m/s. It is possible that the topology at 1 m/s scenario is not so well connected than other scenarios from the same set of simulation. Our results show that the packet delivery ratio for GPSR and LGF are less when 1 m/s is used. In fact, the routing overhead consists of route updates and routing packets. The reduction of packet delivery ratio means more routing packets are dropped in transit. This explains the reduction of routing overhead at 1 m/s.

It can be observed that the routing overhead of LGF varies with mobility models. The results show the MAN model has the highest routing overheads and the RPGM model has the lowest routing overheads; the RWP and FWY models are in the middle. In fact, the readings for the RWP and FWY models are quite close to one another. This seems to be correlated to the average link duration in Table 4.7. The LGF adaptive neighbourhood update algorithm is designed to increase the frequency of route advertisement when wireless links are more likely to break away. The results show this adaptive method is able to regulate the update frequency according to the link condition.

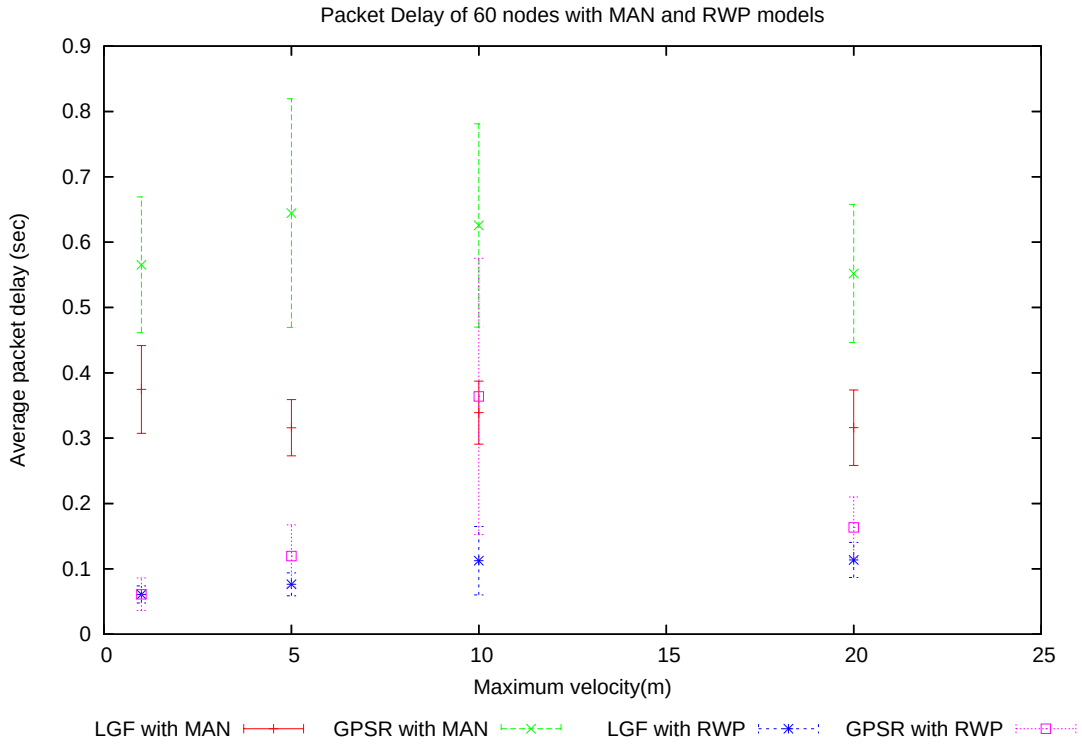


Figure 4.10: Average packet delay with varying position accuracy for MAN and RWP models

Figures 4.10 and 4.11 show the packet delay for LGF and GPSR, with the RWP, FWY, MAN and RPGM models, alongside various maximum velocities over a simulation time of 900 seconds. The results show that the LGF in general achieves not only low but also stable average packet delay across various settings of maximum velocities. We found the readings for the average packet delay for LGF, across different settings of maximum velocities with various mobility models, are between 0.05 and 0.5 seconds. In contrast, GPSR presents a slightly higher delay. Other than the delay of 1 second when FWY is configured with 20 m/s maximum node velocity, the results are between 0.05 and 0.65 seconds.

As a whole, we notice that different mobility models have different properties of link connectivity and can yield very different results of the packet delivery ratio, routing overhead, as well as average packet delay. The results show GPSR and LGF are competitive with one another when comparing the MAN, RWP and RPGM models. However, when comparing the protocols using the FWY model, GPSR experiences packets loss as the maximum node velocity increases. GPSR relies on accurate location of the immediate neighbours to make forwarding decisions and is therefore more likely to encounter the positional uncertainty problem in the FWY model, where neighbours can travel on a path in the opposite direction.

4.6.2 Performance with Varying Standard Deviation of Position Inaccuracy

Figures 4.12 and 4.13 show the successful packet delivery ratio for LGF and GPSR, with a maximum velocity for the RWP, FWY, MAN and RPGM models configured to 10 m/s, alongside

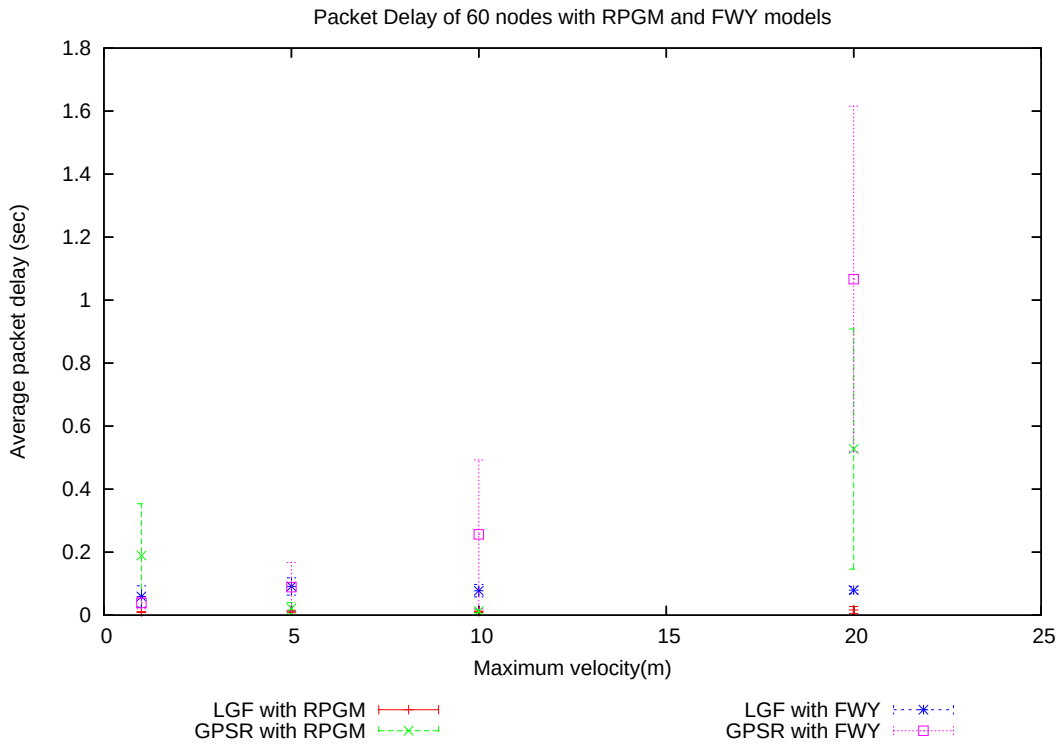


Figure 4.11: Average packet delay with varying position accuracy for FWY and RPGM models

0, 10, 20, 40, 80, 160 and 200 metres of positional inaccuracy, over a simulation time of 900 seconds. The results show that LGF performs much better than GPSR across the full range of standard deviations for position inaccuracy settings using the RWP, FWY, MAN and RPGM models.

For example, with the RPGM model in use, LGF steadily achieves approximately 99% successful packet delivery ratio even when the positional inaccuracy progressively increases from 0 to 200 metres, while the GPSR's successful packet delivery ratio decreases from approximately 99% to about 7% when the positional inaccuracy is progressively increased from 0 to 200 metres.

For the FWY model, the LGF presents a stable 87% of successful packet delivery ratio as the positional inaccuracy is progressively increased from 0 to 200 metres, while GPSR's successful packet delivery decreases from approximately 72% to about 6% when the positional inaccuracy is progressively increased from 0 to 200 metres.

From our NS2 trace with the RWP model, LGF presents a stable 80% of successful packet delivery ratio as the positional inaccuracy progressively increases from 0 to 200 metres, while GPSR successful packet delivery ratio decreases from approximately 80% to about 9% as the positional inaccuracy is progressively increased from 0 to 200 metres.

In our simulation with the MAN model, LGF presents stable results between 48% and 49% successful packet delivery when the positional inaccuracy is progressively increased from 0 to 200 metres, while GPSR's successful packet delivery decreases from approximately 49% to about 7% as the positional inaccuracy is progressively increased from 0 to 200 metres.

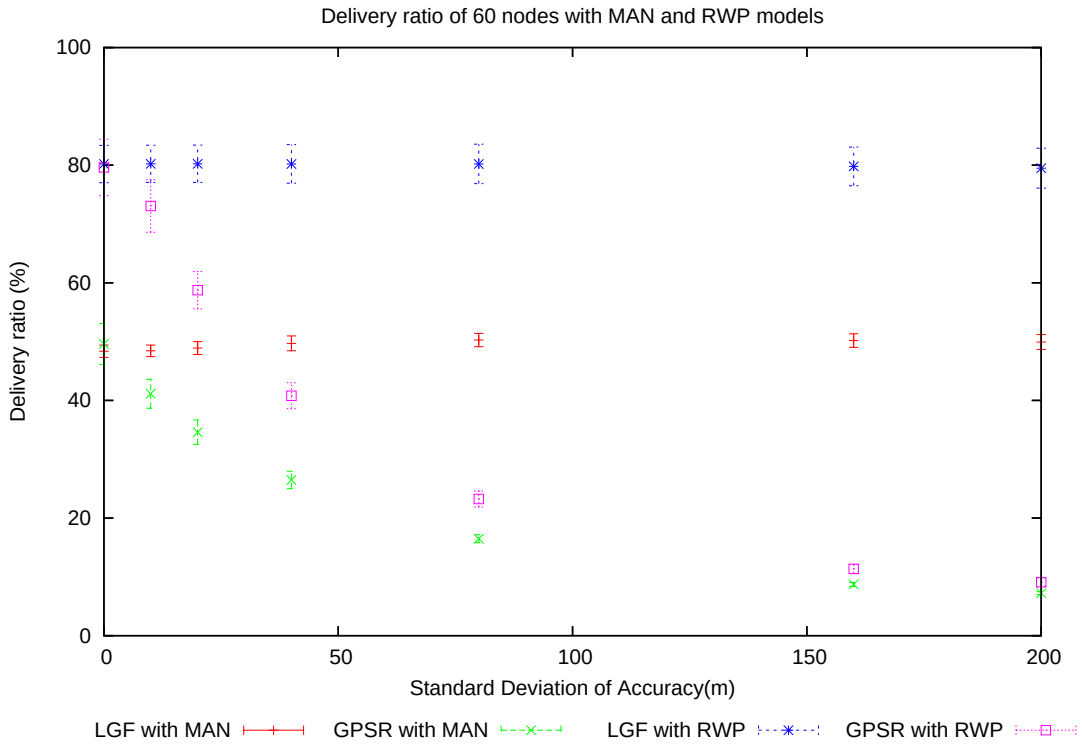


Figure 4.12: Packet delivery with varying position accuracy and update frequency for MAN and RWP models

The results show that the successful packet delivery ratio of GPSR is competitive to LGF, when the positional inaccuracy is reduced to 0. However, the successful packet delivery ratio of GPSR decreases when the positional inaccuracy is increased.

From our observation, the packet delivery ratio of GPSR with the RPGM mobility model significantly reduces from 98% to 41% when the positional inaccuracy increases from 0 to 10 metres. Similarly, the packet delivery ratio of GPSR with the FWY mobility model also significantly reduces from 71% to 28% when the positional inaccuracy increases from 0 to 10 metres. In contrast, the packet delivery ratio of GPSR with the MAN mobility model reduces by 8% when the positional inaccuracy increases from 0 to 10 metres and the packet delivery ratio of the packet delivery ratio of the RWP mobility model is reduced by 6% when the positional inaccuracy is increased from 0 to 10 metres.

When comparing these results with Table 4.5 in section 4.5.1, there seems to be some correlation between the network density and the measured packet delivery ratio. In general, the average distance between two adjacent neighbours is closer in a dense network than a sparse network. This means that the position-based forwarding algorithm is more likely to select a wrong neighbour for greedy forwarding or planar graph construction in a dense network than a sparse network. Apart from the network density, the sharp fall in performance for the RPGM and FWY models is caused by two other contributing factors: first, the position inaccuracy that we have instilled into the system; second, the spatial correlation of the RPGM and FWY model.

In our earlier discussion about degree of connectivity in section 4.5, we found the RPGM

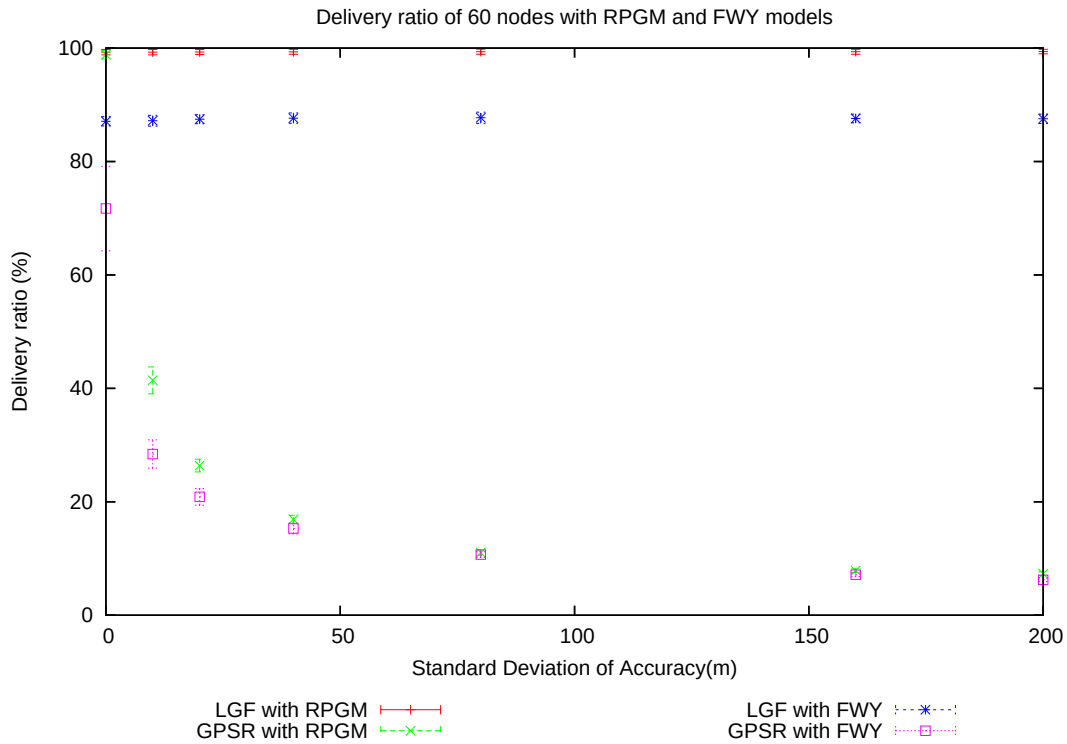


Figure 4.13: Packet delivery with varying position accuracy and update frequency for FWY and RPGM models

model has a high degree of connectivity. This is probably caused by a partition of the space into disjoint groups, each with its own mobility pattern, most likely contributed to by the overlapping of space amongst groups, where each of them has its individual group mobility pattern. This means members from different groups could also be close to each other. One possibility is that this transient proximity between members from independent groups could potentially disrupt the planar graph construction in high network density situations. For example, it is difficult to maintain a stable and up to date planar graph when many neighbours in the same area are moving faster than the rate of neighbourhood update. Apart from that, it is also difficult to ensure the correctness of greedy forwarding in such a situation. Likewise, the opposing traffic in the FWY model also presents difficult conditions for planar graph maintenance or correct greedy forwarding. In the previous chapter, we have shown that incorrect greedy forwarding as well as construction of planar graph could result in packet looping, sub-optimal routing and packet drop [Son *et al.* (2004)].

Figures 4.14 and 4.15 show the routing overheads of LGF and GPSR, with maximum velocity of the RWP, FWY, MAN and RPGM models configured to 10 m/s , alongside 0, 10, 20, 40, 80, 160 and 200 metres of positional inaccuracy over a simulation time of 900 seconds. The results show LGF in general incurs more routing overhead than GPSR.

From our observation, the routing overhead of LGF increases when the positional inaccuracy increases. In contrast, the GPSR routing overhead decreases when the positional inaccuracy increases.

For example, with the MAN mobility model in use, LGF routing overhead increases from

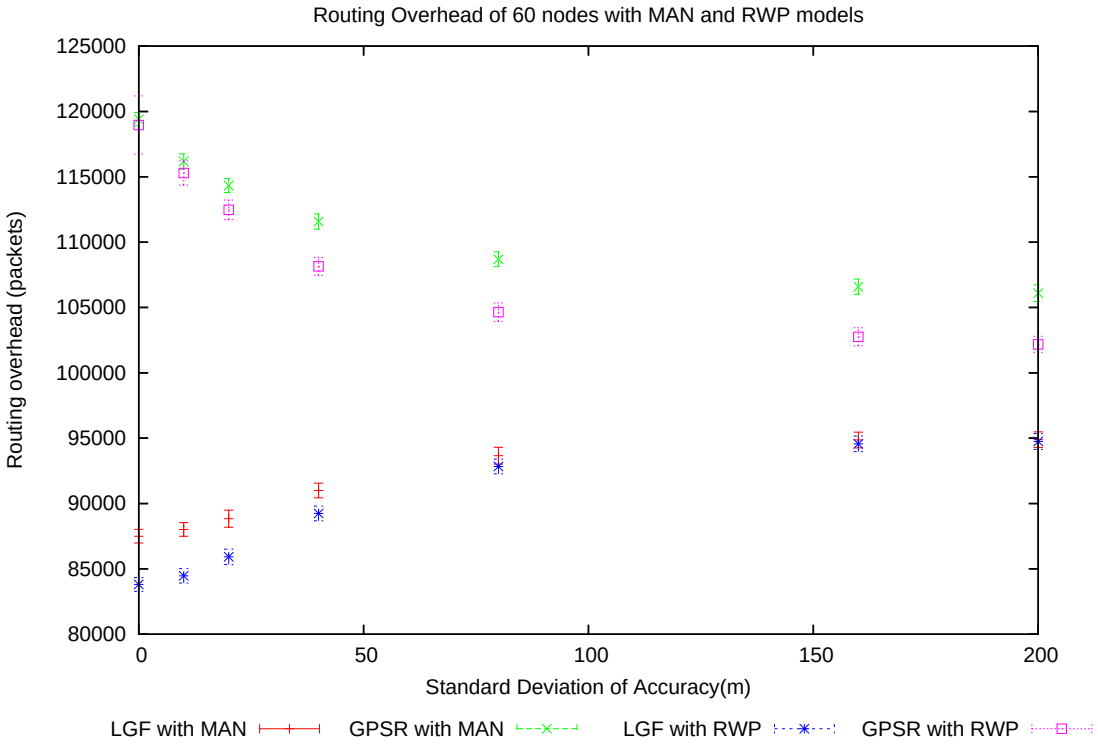


Figure 4.14: Routing overhead with varying position accuracy and update frequency for MAN and RWP models

82000 packets to 95000 packets when the positional inaccuracy varies from 0 to 200 metres, while GPSR routing overhead decreases from 118000 packets to 106000 packets when the positional inaccuracy varies from 0 to 200 metres.

As for the RWP model, LGF routing overhead increases from 83000 packets to 95000 packets when the positional inaccuracy varies from 0 to 200 metres, while GPSR routing overhead decreases from 118000 packets to 101000 packets when the positional inaccuracy varies from 0 to 200 metres.

In the case of the FWY model, LGF routing overhead increases from 81000 packets to 94000 packets when the positional inaccuracy varies from 0 to 200 metres, while GPSR routing overhead decreases from 135000 packets to 105000 packets when the positional inaccuracy varies from 0 to 200 metres.

With the RPGM model, LGF routing overhead increases from 85000 packets to 94000 packets when the positional inaccuracy varies from 0 to 200 metres, while GPSR routing overhead decreases from 128000 packets to 105000 packets when the positional inaccuracy varies from 0 to 200 metres.

The LGF uses adaptive neighbourhood update to regulate the frequency of update. Our findings in section 3.4.5 show that this method has positive effects on routing performance when positional inaccuracy is considered. It steadily increases the rate of neighbourhood update when the positional inaccuracy increases. This reduces the probability of a packet being forwarded to a broken link and increases the successful packet delivery ratio. In contrast, the results from

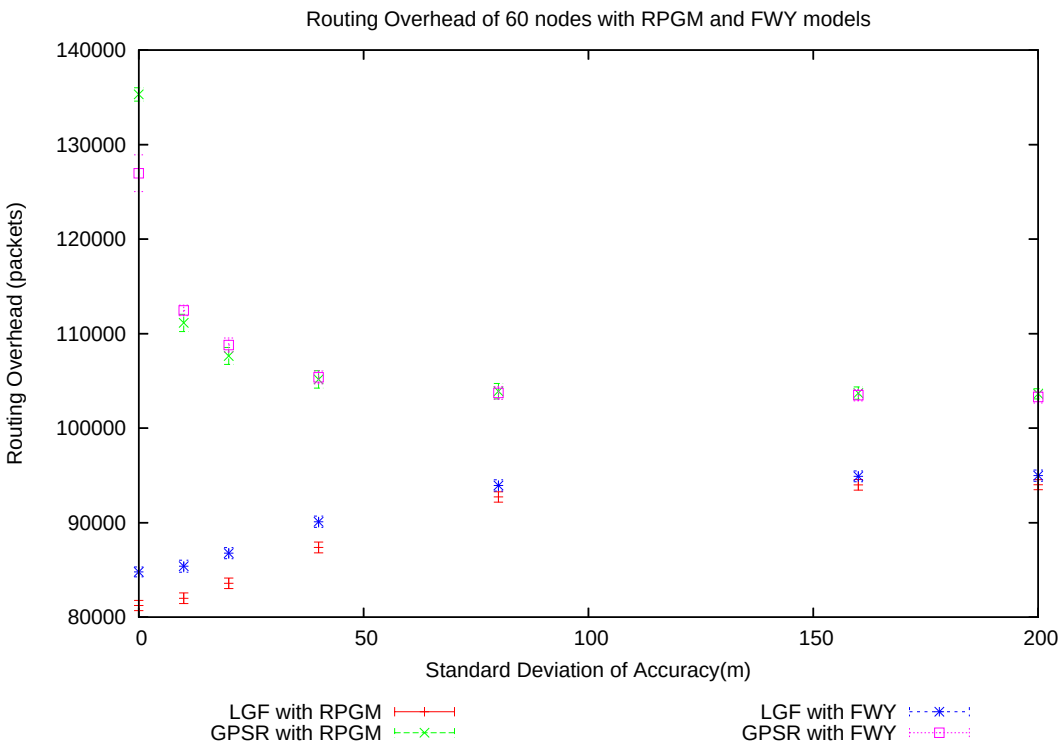


Figure 4.15: Routing overhead with varying position accuracy and update frequency for FWY and RPGM models

GPSR show routing overhead decreases with the increase of positional inaccuracy. Our findings in section 3.4.4 show this to be related to the piggyback feature used in the GPSR protocol. GPSR piggybacks position updates on outgoing data traffic. However, our results show these packets could be lost when positional inaccuracy is considered. As a result, nodes in the network get less up-to-date geographic states from the neighbours and potentially compromises the integrity of the position-based forwarding algorithm. The sharp drop of routing overhead in the RPGM model from Figure 4.15 and the sharp drop in successful packet delivery ratio in the RPGM model from Figure 4.13 has demonstrated this issue.

Figures 4.16 and 4.17 show the average packet delay of LGF and GPSR, with maximum velocity of the RWP, FWY, MAN and RPGM models configured to 10 m/s, alongside 0, 10, 20, 40, 80, 160 and 200 metres of standard deviation for position inaccuracy over a simulation time of 900 seconds. The results show LGF in general generates much lower average delay than GPSR.

For example, with the MAN mobility model, LGF maintains steady average packet delay between 0.34 seconds to 0.36 seconds when the positional inaccuracy varies from 0 to 200 metres, while the GPSR average packet delay increases from 0.62 seconds to 3.53 seconds when the positional inaccuracy varies from 0 to 200 metres.

With the RWP model, LGF maintains steady average packet delay between 0.10 and 0.19 seconds when the positional inaccuracy varies from 0 to 200 metres, while the GPSR average packet delay increases from 0.36 seconds to 11.46 seconds when the positional inaccuracy varies from 0 to 200 metres.

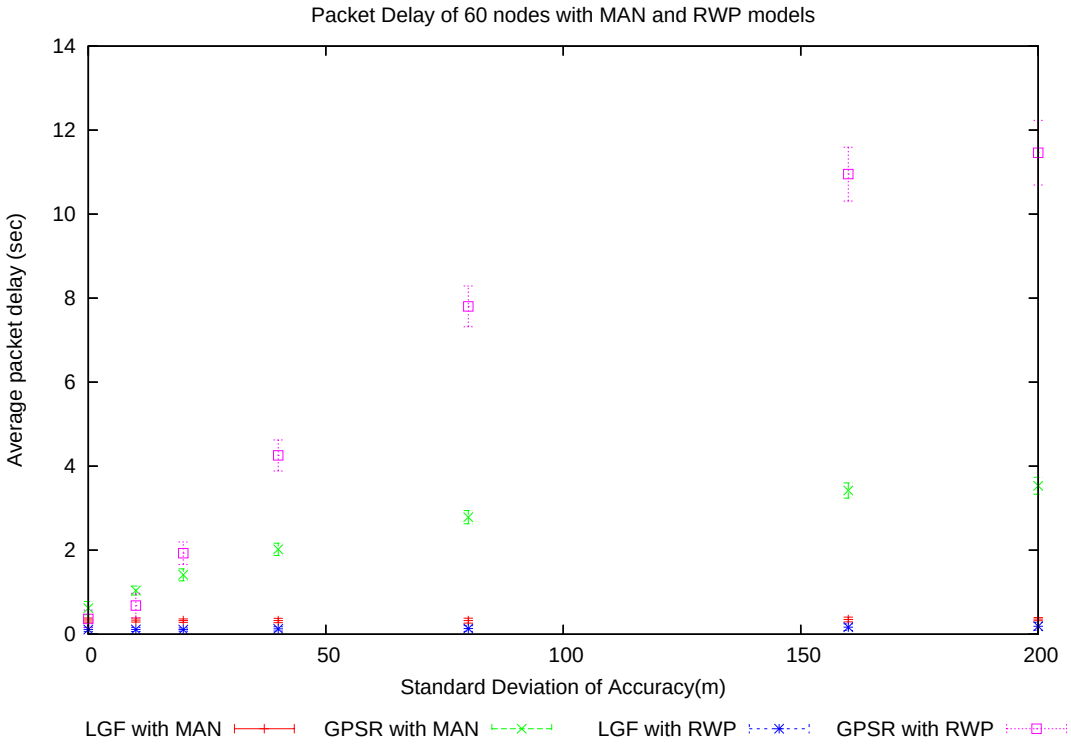


Figure 4.16: Average packet delay with varying position accuracy and update frequency for MAN and RWP models

As for the FWY model, LGF maintains steady average packet delay between 0.07 and 0.09 seconds when the positional inaccuracy varies from 0 to 200 metres, while the GPSR average packet delay increases from 0.25 seconds to 12.49 seconds when the positional inaccuracy varies from 0 to 200 metres.

In the RPGM model, LGF maintains steady average packet delay at about 0.01 seconds when the position inaccuracy varies from 0 to 200 metres, while the GPSR average packet delay increases from 0.01 and 20.86 seconds when the positional inaccuracy varies from 0 to 200 metres.

The results show that LGF is able to deliver packets swiftly when tested with RWP, FWY, MAN and RPGM models at different levels of positional inaccuracy. In contrast, the average packet delay of GPSR increases when positional inaccuracy increases. This increase is more likely to be caused by temporary looping or suboptimal routing of GPSR protocol when positional inaccuracy is introduced.

4.7 Future Work

Despite the fact that our LGF protocol has been thoroughly tested with the RWP, FWY, MAN and RPGM models, we feel more evaluations could be carried out with different traffic models. This is particularly important for models like RPGM and FWY models. For the RPGM model, more study needs to be done on inter-group communications, especially with different numbers of group members overlapping each other. As for the FWY model, it would be interesting to investigate the possibility of taking advantage vehicle mobility to bring the packet physically

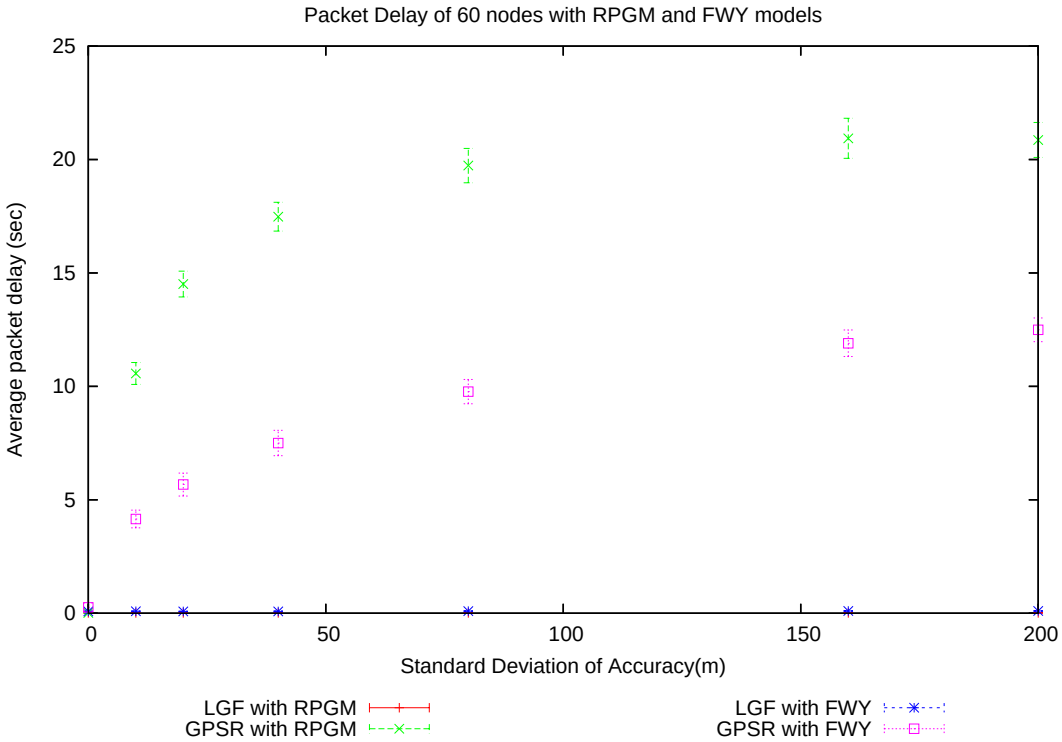


Figure 4.17: Average packet delay with varying position accuracy and update frequency for FWY and RPGM models

closer to the destination, rather than away from the destination.

In the last chapter, we showed that the performance of LGF can be degraded when substantial positional errors or staleness of location is introduced. In this chapter, other than the RWP model which has shown slight performance loss of LGF packet delivery ratio when positional inaccuracy is configured at 200 metres, the MAN, FWY and RPGM models did not show any sign of performance loss. This is because the MAN, FWY and RPGM models have much more mobility and space restrictions than the RWP model. Therefore, the actual network diameter could be smaller than the less restrictive model used by the RWP model. Nevertheless, our aim in this chapter is to show that LGF is more resilient than GPSR when positional inaccuracy is introduced with various mobility models. For future work, we would like to study the performance of LGF with different MAN, FWY and RPGM mobility model settings in a much larger network.

Like many other researchers' evaluation of MANET protocols, we have been using a circular area in our radio propagation. Modelling the radio propagation is a non trivial matter. However, we would like to evaluate with LGF with more complex propagation models, if a suitable tool becomes available in simulation systems in the future.

Another area of possible future work is to evaluate our protocol with radio propagation and mobility data traces obtained from the real world.

4.8 Conclusion

The random waypoint mobility model has never been considered a realistic model for routing protocol evaluation. Many mobility models have been proposed in recent years. We found the software tool from the *IMPORTANT* project particularly useful. We used this software to generate Freeway, Manhattan and Reference Point Group Mobility profiles for our simulation experiments. In addition, we also generated a set of random waypoint model profiles from the NS2 utility.

We first gather some mobility profiles of each of these mobility models. More specifically, the degree of connectivity and the duration of wireless link. Our intention is to understand the constraint at the physical layer and later use that as a basis for discussion of the results we obtained from simulation.

Each model has a very different impact on physical connectivity over time, and it is apparent that the effect on performance varies greatly from one model to another. We find the RPGM model has a good degree of connectivity, and longer average link duration. However, this does not seem to serve well for GPSR when position inaccuracy is considered.

We compare our proposed hybrid LGF with position-based GPSR. Our results show that LGF proves not only to provide reliable packet delivery, but is also able to deliver packets with low routing overhead, when tested with the RWP, FWY, MAN and RPGM models.

In the first set of simulations, we compare LGF and GPSR with RWP, FWY, MAN and RPGM models. Our results show both protocols are competitive when tested with RWP, MAN and RPGM models. However, LGF performs better than GPSR when the maximum node velocity of the FWY is 5 m/s or more.

In the second set of simulations, we further evaluate the ability of LGF and GPSR to handle position inaccuracy with RWP, FWY, MAN and RPGM models. LGF proved robust across a wide range of values for the standard deviation of position inaccuracy. Our results again show that both successful packet delivery by LGF, and average packet delay, are stable and outperform the GPSR. In contrast, the performance of GPSR decreases with increases of positional inaccuracy. In particular, in the results with the RPGM and FWY mobility model, the performance drops sharply for a setting of the positional inaccuracy of 10 metres.

From our observations, we consistently found a correlation between the routing overhead and the failure to provide successful packet delivery and average packet delay. Our findings show that the positional inaccuracy has negative effect on the GPSR piggyback feature. As a result, the overall neighbour update rate for GPSR falls as the positional inaccuracy increases. This is likely to lead to a further reduction in the accuracy of position reports, and therefore further disrupt the position-based forwarding algorithm. In contrast, the positional inaccuracy has a positive effect on LGF, in the sense that the number of hybrid advertisement increases as positional inaccuracy increases. This reduces the staleness of positional state and can improve the routing performance.

Recent research papers cited in our related work section show that mobility and positional inaccuracy have profound effects on the fundamental building blocks of the position-based forwarding protocol. The research shows the construction of planar graph can be compromised

by mobility and positional inaccuracy. Furthermore, they also show it is not easy to maintain correct greedy forwarding when mobility and positional inaccuracy are considered. The results obtained from our simulations are consistent with this.

Based on the results, related work and our extensive observation, we conclude that the hybrid LGF protocol is robust in handling positional inaccuracy and a range of mobility scenarios. In contrast, the current position-based GPSR is not able to handle similar level of positional inaccuracy and mobility.

Chapter 5

Conclusion and Future Work

If the wireless research community is serious about providing wireless multiple hop routing for mobile users in the near future, the notion of establishing an optimal path has to be reconsidered. We find that existing topology-based protocols have scalability problems when trying to sustain global optimal routing in a mobile environments. Although position-based forwarding does directly address the scalability problem, the technique relies on precision of geographic state and is not able to tolerate with positional uncertainty. Although recent published work claims success in solving the problem with constructing a correct planar graph in the presence of this inaccuracy in positional information, that method has only been verified in static sensor networks. At the time of writing, no-one has published a viable solution for mobile scenarios.

Beside this, there are other constraints relating to wireless multi-hop routing. Researchers have demonstrated that sharing of a single 802.11 wireless channel for both sender and receiver is disruptive to wireless multi-hop routing. Though this seems to be an issue to be resolved by cross layer design of data link and physical layer, the constraints of underlying protocol were not ignored when we designed the routing protocol. In this case, our recommendation is to disengage from the goal of global optimal routing, and concentrate on using mainly knowledge from the local area to achieve robust packet delivery with low routing overheads.

We have proposed *Landmark Guided Forwarding*, a hybrid solution that harnesses the strengths of both topology-based and position-based routing algorithms. The protocol takes advantage of the local topology knowledge to mitigate the effects of positional uncertainty, while leveraging geographic awareness to find the destination without global topology information. We claim that we have demonstrated the following properties of LGF through simulation experiments:

Better Performance in Scaling Protocol Overheads We evaluate LGF with DSDV, AODV and GPSR, along with 50 nodes and 100 nodes scenarios, using various settings of mobility as shown in chapter 2. Our results indicate clearly that LGF scales better than DSDV, AODV and GPSR when the size of network increases from 50 nodes to 100 nodes. Compared to other protocols, both LGF and GPSR yield relatively low average packet delay. This shows that the link failure strategy shared by existing MANET protocols might have not been a sensible solution. Although the strategy might increases the packet delivery ratio in some cases when re-queue packet can find an alternate path to the destination, our results show the recovering of packet delivery by re-queuing the packet is less likely to work well when network diameter is larger or node mobility increases. This method in

fact could have kept the packet in the network for too long and therefore caused longer average packet delay.

Better Performance in Location Service Accuracy In chapter 3, we assess the resilience of LGF in mitigating of position uncertainty, specifically, inaccurate reporting of geographic position and inconsistent state from the location service. We evaluate the performance of LGF and GPSR with various settings of positional inaccuracy and inconsistency, along with various settings of maximum node velocity. The results clearly show that the performance of GPSR decreases when the positional inaccuracy increases. Similarly, GPSR performance also decreases when maximum velocity increases. In contrast, the hybrid approach used by LGF is resilient to positional inaccuracy, inconsistent position or various settings of maximum velocity.

Better Performance in Mobility Patterns The protocol assessment continues in chapter 4, with more realistic mobility models generated by a software tool from the *IMPORTANT* project. We evaluate the performance of LGF and GPSR with Random Way Point (RWP), Freeway (FWY), Manhatten (MAN) and Reference Point Group Mobility (RPGM) models. We also profiled the physical properties of RWP, FWY, MAN and RPGM models and use it as a basis for discussion. Our results show the RPGM model has better connectivity as well as longer sustaining link than others mobility models. However that does not seem to give GPSR better performance when node mobility or position inaccuracy increase. In contrast, LGF performance is much more robust than GPSR across all mobility models when simulated with different settings of positional inaccuracy and node velocity.

In general, results from all three chapters are consistent, and clearly support our hypothesis that LGF is robust, flexible, and scalable, compared to existing schemes, in delivering packets in a variety of mobility scenarios. We conclude that LGF meets the key challenge we have set out for mobile multi-hop routing.

However, one aspect of the evaluation of LGF in realistic settings has not been achieved, and that is to consider more accurate radio propagation scenarios. There is a rough consensus in the wireless networking research community that radio propagation requires practical experiments rather than simulation, as radio models in simulators are rarely able to capture levels of detail necessary to reflect the real world, unlike the scaling, location service accuracy, and mobility patterns that we have explored. Hence, for future work, we would like to move on to prototyping LGF and explore the algorithm's operations in a testbed environment.

The simulation results we have presented were from experiments carried out for parameter settings that allow side-by-side fair comparison LGF with existing protocols, with similar parameter value ranges used by researchers who invented those other protocols (GPSR, AODV, DSDV etc). The MANET design space is potentially very large though, and to explore it all through simulation would be difficult. Instead, an analytical model of the system as a whole, covering density distribution (including, for example non-uniform density) of nodes, radio range and shape of radio (e.g. non circular), and the mobility model (RWP, FWY, MAN, RPGM) could be developed, although it would be complex, and might involve techniques such as temporal graphs or stochastic geometric processes [Chaintreau *et al.* (2006)], which are outside of the scope of this research.

Recently, car manufacturers have shown great interest in providing vehicle safety information via wireless communication. In fact, the very recent IEEE 802.11p standard defines a wireless communication specification for vehicle to vehicle (Ad Hoc) as well as vehicle to infrastructure (Infrastructure) modes. Operating at 1000 feet range with data rate of 6 Mbps, a low cost 802.11p client could be installed in every motor vehicle, while wireless access points could be placed strategically along the road, such that it has to be economically feasible without compromising the usability of the system.

A robust and scalable multi-hop ad hoc routing protocol could have a potential in reducing the infrastructure costs of deploying any system similar to a vehicle safety messaging system. Yet the problems in sharing a single channel for multi-hop routing must also be resolved before we can exploit the full potential of hybrid routing.

References

Akyildiz, I.F. & Wang, W. (2004). The predictive user mobility profile framework for wireless multimedia networks. *IEEE/ACM Trans. Netw.*, **12**, 1021–1035.

Bai, F., Sadagopan, N. & Helmy, A. (2003). Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In *Proceeding of the IEEE INFOCOM 2003, San Diego, USA*.

Bhattacharya, A. & Das, S.K. (1999). Lezi-update: an information-theoretic approach to track mobile users in pcs networks. In *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, 1–12, ACM Press, New York, NY, USA.

B.Karp & H.T.Kung (2000). GPSR: Greedy perimeter stateless routing for wireless networks. *In Proceedings of 6th Annual Internation Conference on Mobile computing and Networking(MobiCom 2000), Boston, MA, USA*, 243–254.

Bose, P., Morin, P., Stojmenovic, I. & Urrutia, J. (2001). Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks*, **7**, 609–616.

Broch, J., Maltz, D.A., Johnson, D.B., Hu, Y.C. & Jetcheva, J. (1998). A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Mobile Computing and Networking*, 85–97.

Chaintreau, A., Hui, P., Crowcroft, J., Diot, C., Gass, R. & Scott, J. (2006). Impact on human mobility on the design of opportunistic forwarding algorithms. In *in Proceeding of the IEEE INFOCOM 2006*.

C.Komar & Ersoy, C. (2004). Location tracking and location based service using IEEE 802.11 WLAN infrastructure. *European Wireless 2004, Barcelona Spain*.

Clarke, B. (1994). *Aviator's guide to GPS*. McGraw-Hill Inc.

C.Perkins & Mobility, I. (2002). IP Mobility Support for IPv4, Request for Comments (Proposed Standard) 3344, Internet Engineering Task Force.

C.Perkins & P.Bhagwat (1994). Highly dynamic destinatin-sequenced distance-vector routing(dsdv) for mobile computing. *Computer Communication Review*, **24**, 234–244.

D.Johnson, D. & J.Broch (2001). *DSR: The Dynamic Source Routing Protocol For Multihop Wireless Ad Hoc Networks, Ad Hoc Networking*. Addison-Wesley Longman Publishing Co. Inc., Boston, MA.

El-Rabbany & A.E-S (1994). The effect of physical correlations on the ambiguity resolution and accuracy estimation in gps differential positioning. Tech. rep., Department of Geodesy and Geomatic Engineering. University of New Brunswick.Canada.

Feeney, L. & Nilsson, M. (2001). Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proceedings of IEEE INFOCOM (Anchorage, AK, 2001)*.

Gao, J., Guibas, L.J., Hershberger, J., Zhang, L. & Zhu, A. (2001). Geometric spanner for routing in mobile networks. In *MobiHoc '01: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, 45–55, ACM Press, New York, NY, USA.

Giordano, S., Stojmenovic, I. & Blazevie, L. (2001). Position based routing algorithms for ad hoc networks: a taxonomy. World Wide Web,`http://www.site.uottawa.ca/-ivan/routing-survey.pdf`.

Glennon, E. & Dempster, A.G. (2004). A review of gps cross correlation mitigration techniques. *Proc GNSS 2004*.

G.Pei, M.Gela & Chen, T.W. (2000). Fisheye state routing: A routing scheme for ad hoc wireless networks. *IEEE ICC*, **1**, 70–74.

Gupta, P. & Kumar, P. (1999). Capacity of wireless networks. In *Technical report, University of Illinois, Urbana-Champaign, 1999*.

Haas, Z.J. & Pearlman, M.R. (2001). *ZRP: a hybrid framework for routing in Ad Hoc networks*. Addison-Wesley Longman Publishing Co., Inc.

Halsall, C. (2000). An introduction to differential gps. World Wide Web,`http://www.oreillynet.com/pub/a/wireless/2000/12/29/two_gps.html`.

Hofmann-Wellenhof, B., H, L. & J, C. (1998). *GPS Theory and Practice*. Springer-Verlag.

Hong, X., Gerla, M., Pei, G. & Chiang, C. (1999). A group mobility model for ad hoc wireless networks. In *Proceedings of ACM/IEEE MSWiM'99, Seattle, WA, Aug. 1999, pp.53-60*.

Hong, X., Kwon, T.J., Gerla, M., Gu, D.L. & Pei, G. (2001). A mobility framework for ad hoc wireless networks. *Lecture Notes in Computer Science*, **1987**, 185–196.

Hopper, A., Harter, A. & Blackie, T. (1993). The active badge system. In *CHI '93: Proceedings of the SIGCHI conference on Human factors in computing systems*, 533–534, ACM Press, New York, NY, USA.

Hubaux, J., Buttyan, L. & Capkun, S. (2001). The quest for security in mobile ad hoc networks. In *Proceedings of ACM Symposium on Mobile Ad Hoc Network and Computing (MobiHoc)*.

Kaplan, E. (1996). *Understanding GPS Principles*. Artech House.

Kim, Y., Lee, J.J. & Helmy, A. (2004). Modeling and analyzing the impact of location inconsistencies on geographic routing in wireless networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, **8**, 48–60.

Kim, Y.J., Govindan, R., Karp, B. & Shenker, S. (2005). Geographic routing made practical. *Proceedings of the Second USENIX/ACM Symposium on Networked System Design and Implementation(NSDI 2005), Boston, MA*.

Ko, Y.B. & Vaidya, N.H. (2000). Location-aided routing (LAR) in mobile ad hoc networks. *Wirel. Netw.*, **6**, 307–321.

LaMarca, A., Chawathe, Y., Conolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, P., Borriello, G. & Schilt, B. (2004). Place lab: Device positioning using radio beacons in the wild. Tech. rep., Intel Research.

Li, J., Jannotti, J., De Couto, D., Karger, D. & Morris, R. (2000). A scalable location service for geographic ad-hoc routing. In *Proceedings of the 6th ACM International Conference on Mobile Computing and Networking (MobiCom '00)*, 120–130.

Li, J., Blake, C., Couto, D.S.J.D., Lee, H.I. & Morris, R. (2001). Capacity of ad hoc wireless networks. In *Mobile Computing and Networking*, 61–69.

Li, L., Halpern, J. & Haas, Z. (2002). Gossip-based ad hoc routing. In *Proceeding of the IEEE INFOCOM 2002*.

Liao, W., Tseng, Y. & Sheu, J. (2001). Grid: A fully location-aware routing protocol for mobile ad hoc networks. *Telecommunication Systems*, **18**, 307–321.

Maltz, D., Broch, J., Jetcheva, J. & Johnson, D. (1999). The effects of on-demand behavior in routing protocols for multi-hop wireless ad hoc networks. *IEEE JSAC, August 1999*.

Mauve, M., Widmer, J. & Hartenstein, H. (2001). A survey on position-based routing in mobile ad hoc networks. *IEEE Network Magazine*, **15**, 30–39.

M.Royer, E. & Toh, C.K. (1999). A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications Magazine*, 46–55.

Nain, D., Petigara, N. & Balakrishnan, H. (2004). Integrated routing and storage for messaging applications in mobile ad hoc networks. *Mob. Netw. Appl.*, **9**, 595–604.

Navigation, S. & Position Group at University of New South Wales, A., Sydney (1999). Principles and practice of gps surveying. World Wide Web,`http://www.gmat.unsw.edu.au/snap/snap.htm`.

Niculescu, D. & Nath, B. (2002). Trajectory based forwarding and its applications. Tech. Rep. DCS-TR-488, Department of Computer Science, Rutgers University.

ns group at ISI (1989). NS 2 home page. World Wide Web, `http://www.isi.edu/nsnam/ns/`.

Park, V. & Corson, M. (1997). A highly adaptive distributed routing algorithm for mobile wireless networks. In *IEEE Infocom, 1997*.

Pei, G., Gerla, M. & Hong, X. (2000). LANMAR: Landmark routing for large scale wireless ad hoc networks with group mobility. In *Proceedings of IEEE/ACM MobiHOC 2000, Boston, MA, Aug. 2000, pp. 11-18*.

Perkins, C. & Royer, E.M. (1999). Ad hoc on demand distance vector routing. *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA.*, 90–100.

Pias, M., Crowcroft, J., Wilbur, S., Bhatti, S. & Harris, T. (2003). Lighthouses for scalable distributed location. In *Second International Workshop on Peer-to-Peer Systems (IPTPS '03), Feb 2003*.

Ramanathan, R. & Rosales-Hain, R. (2000). Topology control of multihop wireless networks using transmit power adjustment. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM), pages 404-413, Tel Aviv, Israel, March 2000*.

Ray, S., Carruthers, J. & Starobinski, D. (2003). RTS/CTS-induced congestion in adhoc wireless lans. In *in IEEE Wireless Communication and Networking Conference (WCNC), March 2003, pp. 1516–1521*.

Sadagopan, N., Bai, F., Krishnamachari, B. & Helmy, A. (2003). Paths: analysis of path duration statistics and their impact on reactive manet routing protocols. In *Proceedings of MOBIHOC 2003, Maryland, USA*.

Samaan, N. & Karmouch, A. (2005). A mobility prediction architecture based on contextual knowledge and spatial conceptual maps. *IEEE Transactions on Mobile Computing*, **4**, 537–551.

S.Basagni, V.S., I Chlamtac & Woolward (1998). A distance routing effect algorithm for mobility(DREAM). *in Proceedings of ACM/IEEE Internation Conference on Mobile Computing and Networking(Mobicom 98)*, 76–84.

Seada, K., Helmy, A. & Govindan, R. (2004). On the effect of localization errors on geographic face routing in sensor networks. In *IPSN'04: Proceedings of the third international symposium on Information processing in sensor networks*, 71–80, ACM Press.

Shen, X., Mark, J.W. & Ye, J. (2000). User mobility profile prediction: an adaptive fuzzy inference approach. *Wirel. Netw.*, **6**, 363–374.

Son, D., Helmy, A. & Krishnamachari, B. (2004). The effect of mobility-induced location errors on geographic routing in ad hoc networks: Analysis and improvement using mobility prediction. In *IEEE Wireless Communications and Networking Conference (WCNC), March 2004*.

Stajano, F. & Anderson, R. (1999). The resurrecting duckling: Security issues for ad-hoc wireless networks. In *Security Protocols, 7th International Workshop Proceedings, Lecture Notes in Computer Science, 1999*.

Sundaresan, K., Hsieh, H.Y. & Sivakumar, R. (2004). IEEE 802.11 over multi-hop wireless networks: problems and new perspectives. *Ad Hoc Networks*, **2**, 109–132.

S.Xu & T.Saadawi (2001). Does the IEEE MAC protocol work well in multihop wireless ad hoc networks. *IEEE Communication Magazine*, **39**, 130–137.

Tang, L. & Crovella, M. (2003). Virtual landmarks for the Internet. *In Proceedings of Internet Measurement Conference 143–152, Miami Beach, FL.*.

Transport of London (2005). London congestion charging technology trials report. World Wide Web, http://www.tfl.gov.uk/tfl/downloads/pdf/congestion-charging /technology-trials.pdf.

Vahdat, A. & Becker, D. (2000). Epidemic routing for partially connected ad hoc networks. In *Technical Report CS-200006, Duke University, April 2000*.

Vutukury, S. & Garcia-Luna-Aceves, J. (2001). MDVA: A distance-vector multipath routing protocol. In *Proceedings of the IEEE INFOCOM, pages 557–564, 2001.*.

Wanninger.L (1993). Effects of the equatorial ionosphere on gps. *GPS World*, 18–54.

Want, R., Hopper, A., Falc, V. & Gibbons, J. (1992). The active badge location system. *ACM Trans. Inf. Syst.*, **10**, 91–102.

Wilson, D.L. (2001). David l. wilson's gps accuracy web page. World Wide Web,http://users.erols.com/dlwilson/gps.htm.

Xu, Y., Heidemann, J. & Estrin, D. (2001). Geography-informed energy conservation for ad hoc routing. In *Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking, Rome, Italy, July 2001. ACM.*.