AN EVALUATION OF MULTIPLE PATH ROUTING AND ITS IMPACT ON QUALITY OF SERVICE IN MOBILE AD HOC NETWORKS

by

Gregory B. Gerou

REMOVE

A thesis submitted to the Faculty and the Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science (Mathematical and Computer Sciences).

Golden, Colorado
Date ____________________

Signed: ________________________________
Gregory B. Gerou

Approved: ________________________________
Dr. Tracy Camp
Associate Professor

Golden, Colorado
Date ____________________

____
Dr. Graeme Fairweather
Professor and Head
Department of Mathematical and Computer Sciences

ABSTRACT

The topic of multi-path routing protocols has been investigated previously; however, the effectiveness of multi-path routing to improve quality of service by maximizing bandwidth availability, as well as the criteria for determining multi-path routes, have not been clear. For that reason, we have developed a Proactive Disjoint Multipath Routing protocol (PDMR) that we use to compare single-path routing versus multi-path routing. The results of this investigation illustrate the performance of multi-path routes and their impact on quality of service in a variety of mobile ad hoc network scenarios.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I am happy to thank my adviser, Dr. Tracy Camp, for her invaluable guidance and unending patience. I would also like to thank my thesis committee members, Dr. Mike Colagrosso and Dr. Jason Liu. Through the course of my research, their input has always steered me in the right direction. I also appreciate the support I have received from members of the Toilers research group.

Chapter 1

INTRODUCTION

A mobile ad hoc network (MANET) is a network of mobile nodes capable of communicating without the use of any static network infrastructure. Nodes comprising a MANET are not always capable of communicating directly with the entire network. Limited transmission range implies multi-hop routing, the ability for a given node to transmit data to a destination by passing a packet through intermediate nodes. Transmission range limitations also introduce the hidden node problem, the potential for transmission collisions observed by a node with two transmitting neighbors, each unaware of the other's transmission. Multiple-hop routing and the hidden node problem are two examples of difficulties created by the wireless and mobile properties of MANETs. As applications from traditional wired networks are moved into the mobile wireless environment, these properties reveal new problems to be solved.

Quality of Service (QoS) is an example of a traditional wired network desired property that creates new problems when applied to mobile ad hoc networks. QoS refers to the potential performance requirements of network traffic. For example, a particular application may demand that traffic be delivered with a quantified minimum delay, or be allowed a specified amount of network bandwidth. Relative to wired networks, reliable QoS is difficult to achieve in MANETs due to higher network volatility. This volatility leads to difficulty in measuring network state, and applying those measurements to QoS traffic routing.

Stine and Veciana suggest that to include useful quality of service mechanisms, MANET protocols and frameworks should place a greater emphasis on *node state* than on *link state* [1]. In terms of quality of service, node state refers to the available network resources and known characteristics observed at a given node, while link state refers to those resources and characteristics observed over a logical communication *link* between two nodes. This *link* is defined by the ability of neighboring nodes to directly communicate. An example of a link state measurement might be *node A is transmitting 1 kb/s to node B*; an example of a node state measurement might be *node A is observing a local bandwidth usage of 3.2 kb/s (created both by its own transmissions and those of its neighbors).*

The emphasis of node state over link state impacts MANET routing protocols because it includes the impact of neighbor traffic on bandwidth availability. Additionally, it prioritizes efficient use of the RF transmission space surrounding a node rather than that of a link between a pair of nodes. As described in the previous example, link state measurements do not necessarily indicate available network resources. For example, suppose neighbor nodes A and C are reaching their maximum receivable bandwidth, but are not sending or receiving packets to or from each other. Although neither A nor C observe any bandwidth on their shared link, given that neither is capable of receiving further traffic, any new traffic between the two nodes will likely result in a transmission collision. In this example, a QoS framework based on link state measurements would likely yield poorer performance than a QoS framework based on node state measurements.

The change in approach from a link-based to node-based paradigm is also evident in comparing earlier QoS work to current QoS work. For example, the Core-Extraction Distributed Ad hoc Routing Algorithm (CEDAR) [2] and an in-band signaling system, Insignia [3], use link-based metrics while a later approach such as SWAN [4, 5] relies on node-based measurements. SWAN has proved it is possible to obtain good quality of service by using a stateless, reservation-less approach. Specifically, SWAN enables the monitoring and shaping of available bandwidth.

Although robust approaches to QoS in MANETs have been developed and published, most rely on the existence of a single-path that is capable of satisfying the resource requirements for the data flow. A *single path* traffic session incurs a variety of costs such as large bandwidth use due to network traffic along the single path, increased delay due to possible transmission collisions and saturated forwarding queues, reduced battery life due to requisite retransmissions, etc. When a single path is used to transmit a stream of traffic, the costs associated along the single path with that traffic are imposed on the nodes along that path. This problem has motivated the development of protocols that leverage multi-paths to distribute the bandwidth costs associated with a data stream over a broader range of the network. In order for multiple paths to benefit traffic delivery, the following requirements must be met:

1. The source and destination nodes are not bandwidth constrained.
2. Intermediate nodes on the multi-paths are under load such that a) no single path can satisfy the bandwidth requirement, and b) considered together, their bandwidth capacity is sufficient to satisfy the bandwidth requirement.

The goals of this thesis are twofold. The first goal is to present our new multi-path protocol, the *Proactive Disjoint Multi-path Routing protocol.* The second goal is to use PDMR as a tool for evaluating single- versus multi-path routing techniques. Although the above requirements for improving traffic delivery with multi-path routing are relatively straightforward, it is necessary to determine if their frequency is sufficient to observe a significant improvement with multi-path routing relative to single-path routing.

Chapter 2 covers related work, including both existing multi-path routing protocols and established QoS frameworks. Chapter 3 covers the topic of *disjoint routes*, and the criteria for classifying various disjoint path types. This analysis is then applied to the previously covered multi-path routing protocols. Chapter 4 describes the details of our Proactive Disjoint Multi-path Routing protocol (PDMR): a topology dissemination component, a multi-path discovery algorithm, path use policies, and admission and rate control components. Chapter 5 presents our PDMR simulation results and analysis. Chapter 6 presents the conclusions we draw from the results obtained over the course of our research.

Chapter 2

RELATED WORK

Multiple path routing is a concept that has been applied to several applications [16]. However, the fundamental difficulties presented in MANETs caused by node mobility and communication over a volatile medium make new study of multi-path routing protocols within this context interesting. A significant amount of work has been published on the subject of multi-path routing in MANETs. In addition to the work covered in detail later in this chapter, Villela and Duarte present analysis regarding traffic throughput maximization using multi-path routing [9]. Chen, Druschel and Subramania present a multi-path forwarding policy [10]. Cidon, Rom, and Shavitt develop a mathematical model of a multi-path protocol with a QoS reservation mechanism that shows 1) in most networks more than two or three disjoint paths offer no improvement in performance, and 2) real network behavior may deviate from the conditions required to obtain a benefit in performance from multi-path routing [11]. Ogier, Rutenburg, and Shacham present algorithms for computing disjoint paths, without an emphasis on wireless environments [12]. Sidhu, Nair, and Abdallah also present a network-generic algorithm for disjoint path discovery [13]. Taft-Plotkin, Bellur, and Ogier suggest multi-path routing as an implementation for providing QoS services [14]. Vutukury, and Garcia-Luna-Aceves suggest a protocol that calculates multi-paths through the use of distance vector routing. Much of this work includes protocols for discovering and using multi-paths; additionally, Corson and Macher offer an evaluation framework for multi-path protocols [23]. The framework identifies the following fundamental components of a multi-path protocol:

1. *Multiple Route Discovery.* The procedure by which the protocol selects multi-path routes. This procedure generally includes provisions to avoid path looping, and heuristics to generate disjoint multi-path sets.

2. *Filtering Provision.* A protocol component capable of eliminating undesirable multi-path sets based on a metric such as path length, available bandwidth, etc.

3. *Path Usage Policy.* The policy by which the protocol decides what and when to transmit along each of the paths in the multi-path set. Examples of such a policy are:

    - Load Balancing. The transmitting node will send each packet along the least recently used path. For instance, if the multi-path set is comprised of two paths, the transmitting node will alternate packet transmissions between the two paths.

    - Redundancy. The transmitting node will transmit each packet along all paths.

    In the case of redundant path usage, the source node must include a policy to determine the path order for the duplicate packet transmissions. When a node must transmit the same packet to two different destinations individually, a data forwarding policy defines the order of the transmissions to each of these destinations. Examples of a data forwarding mechanism include a round-robin approach, or a heuristic such as path length to determine path priority. The need for path prioritization in a forwarding mechanism is motivated by the difference in predictable behavior between path options.

4. *Multi-path Maintenance Heuristic.* In single-path routing, a set of rules must be built into the routing protocol to enable path recomputation. The only additional complexity, compared to single path routing, is the inclusion of multi-paths. Traditional single-path maintenance heuristics must be updated or replaced to enable efficient use of multi-paths.

5. *Underlying Single Path Routing Protocol.* The use of multi-paths implies the availability of multi-paths, and multi-paths are not always available. If no multi-path is available, but a single path is available, the multi-path protocol must be capable of using and maintaining that single path.

These components comprise a framework for evaluating multi-path routing protocols. In Section 2.1, we discuss the prominent multi-path routing protocols currently available. We summarize these protocols by categorizing them via these fundamental components. Section 2.2 then covers the significant published QoS frameworks for MANETs.

2.1 MANET Multi-path Routing Protocols

In this section we describe the Diffusing Algorithm for Shortest Multi-path (DASM) [16], the Disjoint Pathset Selection protocol (DPSP) [17], the Multi-path Dynamic Source Routing protocol (MDSR) [8], the Graph Multi-path Routing protocol (GMR) [7], and the Split Multi-path Routing protocol (SMR) [6]. Multi-path protocols based on AODV, such as the Ad Hoc On-Demand Multi-path Distance Vector routing protocol (AOMDV) [24], are not covered due to their similarity to Multipath Dynamic Source Routing (MDSR); specifically, the use of route request/route reply schemes in AOMDV is similar to MDSR.

2.1.1 Diffusing Algorithm for Shortest Multi-path (DASM)

The Diffusing Algorithm for Shortest Multi-path (DASM) [16] is a protocol designed to maintain forwarding tables at each node such that for any destination, the information included in the forwarding tables of all nodes implies a directed acyclic graph (DAG) leading to the destination. The primary goal of the protocol is to ensure loop-free routing, which is ensured through the use of the DAG. However, the use of a DAG for multi-path routing does not have any implications in terms of whether or not the multi-paths are disjoint; the DAG may supply an alternate path that includes many of the same nodes as the shortest path, or it may supply an alternate path that is link disjoint from the shortest path. Every entry in a node's routing table includes the following information:

1. the destination node address,
2. the next shortest path hop, and
3. the next shortest multi-path hop.

Both the next shortest path hop and the next shortest multi-path hop are chosen from the DAG. An example network is depicted in Figure 2.1; in this example, the destination is indicated as D. Each of the arrows indicate the routing table contents at the nodes from which each arrow originates. Dotted arrows denote the next shortest path hops and solid arrows denote the next shortest multi-path hops. For instance, the center node in Figure 2.1 has both a dotted arrow and a solid arrow originating from it. The dotted arrow is the next shortest path hop to the destination. The solid arrow is the next *shortest multi-path hop*, or the second next shortest path hop, to the destination.

When a next hop is not defined for a given destination at a given node, that node queries its neighbors for information regarding the destination. If any of the neighbors have the requested destination node in their routing tables, a reply is sent to the querying node indicating the next hop and the total distance to the destination. If no neighbor has the requested destination node in their routing tables, the request is repeated. This process continues until a querying node has been answered with a distance to the destination. At network startup, when all routing tables are empty, this process takes the form of a flooded route request. As replies are sent back to the source, neighboring nodes promiscuously update their routing tables. We note that neighbors of each querying node may not hear route reply messages sent to the querying node. Thus, although the querying node may have successfully discovered the next hop to the destination, the route request will continue to be forwarded. Route replies include a hop distance measurement, enabling querying nodes to maintain a shortest path hop, and additionally maintain the *shortest multi-path hop*. DASM's path usage policy dictates that when the shortest path fails, a shortest multi-path is used if one is available. If no shortest multi-path route is available when the shortest path route fails, a new route request is triggered. If a shortest multi-path is available and if both the shortest path and the shortest multi-paths have failed, a new route discovery is triggered. Table 2.1 describes DASM according to the fundamental components of multi-path protocols, and compares it to the other multi-path protocols described in this chapter.

2.1.2 Disjoint Pathset Selection Protocol (DPSP)

The Disjoint Pathset Selection protocol (DPSP) [17] is essentially an algorithm for path selection. It assumes an underlying routing protocol is capable of generating

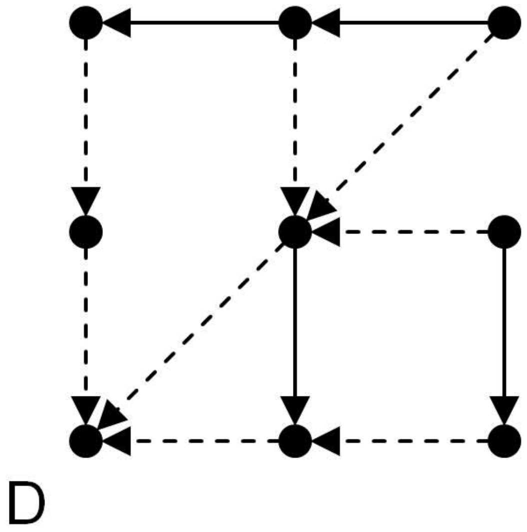

Fig. 2.1. DASM's path classification example

path information sufficient to build (at least) a partial graph representation of the network between the source and destination nodes. DPSP introduces the notion of *path reliability* for use in path selection. The protocol defines reliability as a function of both a path's length and the delivery rate over each of the path's links. This function is essentially DPSP's path filtering provision. A path set is generated by performing a search on the graph for a maximally reliable path (according to DPSP's reliability function). This path is added to a multi-path set; the intermediate nodes in the discovered path are removed from the graph (which we term a *node disjoint* filter), and a new search is performed. The algorithm includes a threshold mechanism to determine when additional paths no longer offer a reliability benefit, according to DPSP's reliability function. Once a multi-path set has been calculated, data is transmitted redundantly on each of the paths, theoretically improving the probability the packetized data will arrive at the destination. The data forwarding mechanism uses a round robin scheme to determine a path order to be used for redundant data packet transmissions. If the reliability of a calculated route falls below a defined threshold, a new route calculation is triggered. It is possible for the protocol to calculate a single path route, given that it meets the requirements to be considered reliable. Table 2.1 describes DPSP according to its fundamental components, and compares it to the other multi-path protocols described in this chapter.

2.1.3 Multi-path Dynamic Source Routing (MDSR)

In Dynamic Source Routing (DSR) [18], the source is responsible for acquiring a route to its destination. The source includes that route in the data packets intended for that destination. DSR builds these routes through a route discovery process, and ideally, a subsequent route reply from the destination to the source. Specifically, the source generates and transmits a route request (RREQ). At each hop toward the destination, an intermediate node adds its address to the list of visited nodes included in the RREQ. When the RREQ reaches the destination, the RREQ will contain a list of nodes that comprise a route from the source to the destination. While the destination may receive information about multi-paths in DSR, only the first discovered route is used; additional paths are discarded by the destination. After receiving the first discovered route, the destination creates a route reply (RREP) and sends it along the discovered path back to the source. Once it is received by the source, the source may begin sending traffic to the destination along the discovered path. Route replies are cached by intermediate nodes for possible use in replying to route requests within a configurable timeout period.

Multi-path Dynamic Source Routing (MDSR) [8] first extends DSR by labeling the initial route request received at a destination node the *primary source route*. Additional route requests received are from *node disjoint* routes, given that route requests are forwarded at most once by any given node involved in any given route discovery. These additional routes are generally longer, as these route requests arrive after the primary source route's request arrives. Route replies for each route are sent to the source. The MDSR route usage policy dictates secondary routes are only used after the primary route has failed. The use of previously discovered secondary routes reduces route discovery overhead. If only one RREQ is received by the destination (i.e., there is only one path from source to destination), MDSR acts essentially the same as the single path DSR protocol.

MDSR offers two mutually exclusive multi-path maintenance heuristics. The authors label these versions of MDSR as "Protocol 1" and "Protocol 2". Protocol 1 prevents intermediate node route caching for alternate routes, which implies that only the source is capable of fixing a broken path. Figure 2.2 depicts an example of Protocol 1 for a given source, S, and destination, D; the dotted lines represent the primary route, and the solid lines represent the alternate paths. Protocol 2, on the other hand, allows intermediate nodes to cache alternate routes; intermediate nodes use this information saved to fix routes when broken links are detected. For instance, if an intermediate node determines a current route has failed, it begins using any available cached alternate route to the destination. If no such route is cached, a route error is sent to the next upstream node where the process is repeated. Figure 2.3 depicts an example of Protocol 2; the dotted lines represent the primary route, and the solid lines represent the alternate routes. Protocol 1 offers the simplicity of maintaining alternate disjoint paths for each traffic session only at the source; Protocol 2 requires more overhead for maintaining alternate disjoint paths at all intermediate nodes, but is able to offer backup path maintenance. Table 2.1 describes MDSR according to its fundamental components, and compares it to the other multi-path protocols described in this chapter.

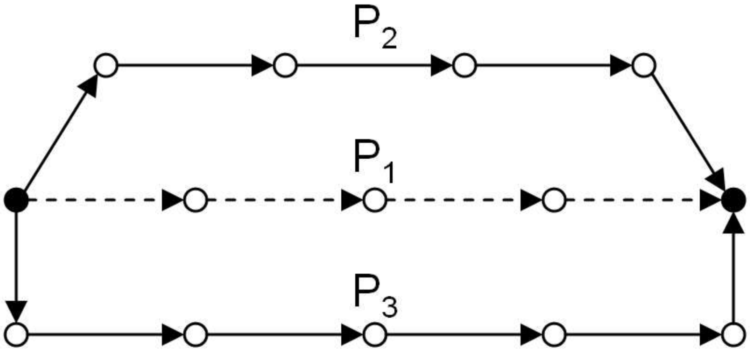

FIG. 2.2. MDSR - Protocol 1 example

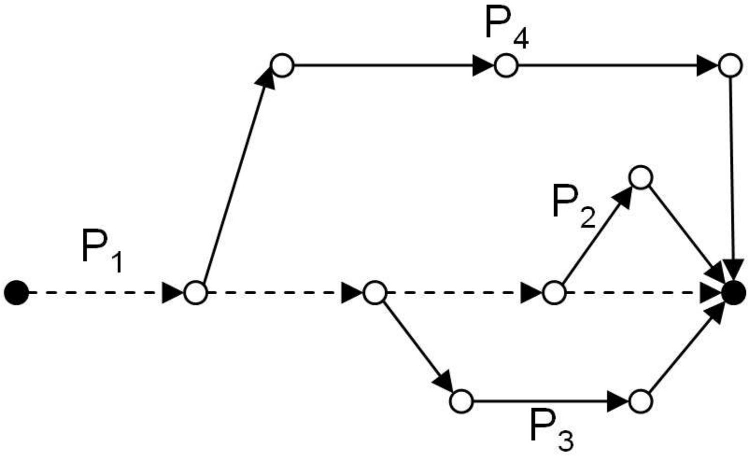

FIG. 2.3. MDSR - Protocol 2 example

2.1.4 The Graph Multi-path Routing Protocol (GMR)

Similar to MDSR, the Graph Multi-path Routing protocol [7], or GMR, floods a route request; however, each node that receives the route request in GMR pauses, waiting for duplicate route requests. Once the pause time has elapsed, information from all received route requests are combined and forwarded as a single route request. This method enables the destination to build a graph representation of the network and to determine multiple link disjoint routes. Once these paths are communicated to the source, data packets begin transmitting. The published work on GMR does not specify what policy is used to determine how the multi-paths are employed, but it is reasonable to assume that once it has been determined that all calculated paths have failed, a new route discovery process is triggered. Additionally, it is possible for GMR to function as a single path routing protocol, assuming only a single path exists between the source and destination nodes.

The motivation for GMR is a limitation of MDSR. Duplicate route requests are ignored in MDSR, just as they are in DSR. Thus, the set of multi-path routes discovered in MDSR is defined by the sequence in which route requests arrive at each node. For instance, in MDSR, a node that receives a route request will forward it and, therefore, belong to the growing potential route included in that request. All other route requests received by that node are ignored, which implies the route requests that reach the destination are comprised of mutually exclusive node sets. GMR, on the other hand, incorporates duplicate route requests, thus allowing potential overlapping route requests to reach the destination. Equipped with these additional, potentially overlapping options, the destination is more capable of calculating disjoint paths. Figure 2.4 illustrates this idea. The solid connecting lines indicate links potentially discovered by MDSR. The combination of the solid connecting lines and the dashed connecting lines are links potentially discovered by GMR. Figure 2.4 illustrates that GMR is capable of providing enough topological data to the destination node to build a graph representation of the pertinent network subset. MDSR is not capable of building such a graph. Table 2.1 describes GMR according to its fundamental components, and compares it to the other multi-path protocols described in this chapter.

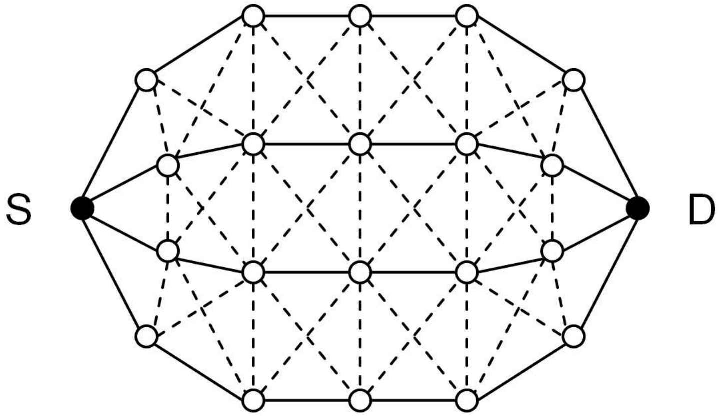

Fig. 2.4. A comparison of GMR and MDSR

2.1.5 The Split Multi-path Routing Protocol (SMR)

The Split Multi-path Routing protocol [6], or SMR, is a protocol intended for use in discovering *maximally disjoint routes.* Similar to MDSR and GMR, SMR's route discovery process relies on a RREQ/RREP scheme; however, unlike MDSR and GMR, SMR repeats route requests. These repeat transmissions increase the likelihood of discovering a multi-path route by allowing multiple path options to share the same intermediate nodes. Once a configurable time has elapsed after the destination receives the first RREQ, it labels the path defined by the first received RREQ as shortest in terms of delay. It then searches the remaining received route requests for a maximally disjoint path, i.e., the discovered path that includes the fewest number of nodes also included in the shortest length path. These two paths are transmitted back to the source, which then injects traffic into the network. The source will use the shortest path until it fails, at which point SMR is configurable to either 1) continue using the single available path or 2) trigger a new route discovery.

Obviously the advantage of not automatically triggering a new route request is a decrease in routing overhead traffic. However, SMR's authors also acknowledge that waiting until all available paths to a destination have failed implies that no data can be sent to the destination during the subsequent route discovery. SMR is configurable such that nodes initiate new route discoveries as soon as the shortest route breaks to ensure that an alternate path is available for data transmission during the route discovery process. If only one path is available, regardless of how SMR is configured, a new route discovery is triggered upon the failure of this single path. Table 2.1 describes SMR according to its fundamental components, and compares it to the other multi-path protocols described in this chapter.

2.2 MANET QoS Frameworks

In the following subsections we discuss CEDAR, Insignia, and SWAN. These three QoS frameworks, which include both MAC and network layer QoS solutions, represent a cross section of the area of QoS frameworks in terms of chronology, approach, and popularity. These three frameworks are presented because they are the prominent approaches to solving QoS provisioning in MANETs.

2.2.1 Core-Extraction Distributed Ad Hoc Routing (CEDAR)

CEDAR, the Core Extraction Distributed Ad hoc Routing algorithm [2] is a proposed QoS routing framework for small- to medium-sized ad hoc networks (tens to hundreds of nodes). The nodes in the network dynamically elect an approximate *minimum dominating set* of core nodes. A set of core nodes is considered a minimum dominating set if they represent the minimum number of nodes required to span the

	DASM	**DPSP**	**MDSR**	**GMR**	**SMR**
Multiple Route Discovery	Route request/route reply scheme.	Relies on underlying topology distribution protocol.	Route request/route reply scheme.	Route request/route reply scheme, combined with graph building.	Route request/route reply scheme.
Filtering Provision	Distance to destination included in route replies.	Filters paths based on reliability.	Dropped duplicate route requests ensure node disjoint routes.	Graph calculations give link disjoint routes.	A node disjoint filter is applied to prospective paths.
Path Usage Policy	Second path is used as a backup.	Redundant packet transmissions.	Second path is used as a backup.	Not specified in literature.	Additional paths are used as backups.
Multi-path Maintenance Heuristic	When the first and second paths have failed, a new route discovery is triggered.	When the route's reliability drops below a threshold, a new route is computed.	When all known paths have failed, a new route discovery is triggered.	When all known paths have failed, a new route discovery is triggered.	When one or both of the paths have failed, a new route discovery is triggered.
Underlying Single Path Protocol	Routing tables enable single path mode.	If a single path meets the reliability requirements, it may be used.	Single path functionality similar to DSR.	Search on the graph can yield single path routes.	Single path functionality similar to DSR.

Table 2.1. A breakdown of the fundamental components of each of the discussed multi-path routing protocols.

network, i.e., every non-core node is a one-hop neighbor of at least one core node. This network of core nodes is used to propagate link bandwidth information in two modes. First, information regarding a significant increase in available bandwidth on a given link is propagated by an *increase wave*, i.e., a packet that is repeated over a broad range of the core network. A significant decrease in available bandwidth on a given link is propagated by a *decrease wave*, i.e., a packet that is repeated over a smaller range of the core network. Essentially, information about usable links is shared over a large amount of the network and information about constrained links is shared locally. Figure 2.5 illustrates an example CEDAR network. The nodes depicted in solid black comprise the network's core or the network's approximated minimum dominating set; all other nodes are depicted as white-centered circles. The solid connecting lines denote links between pairs of nodes, while the dotted lines indicate the core network.

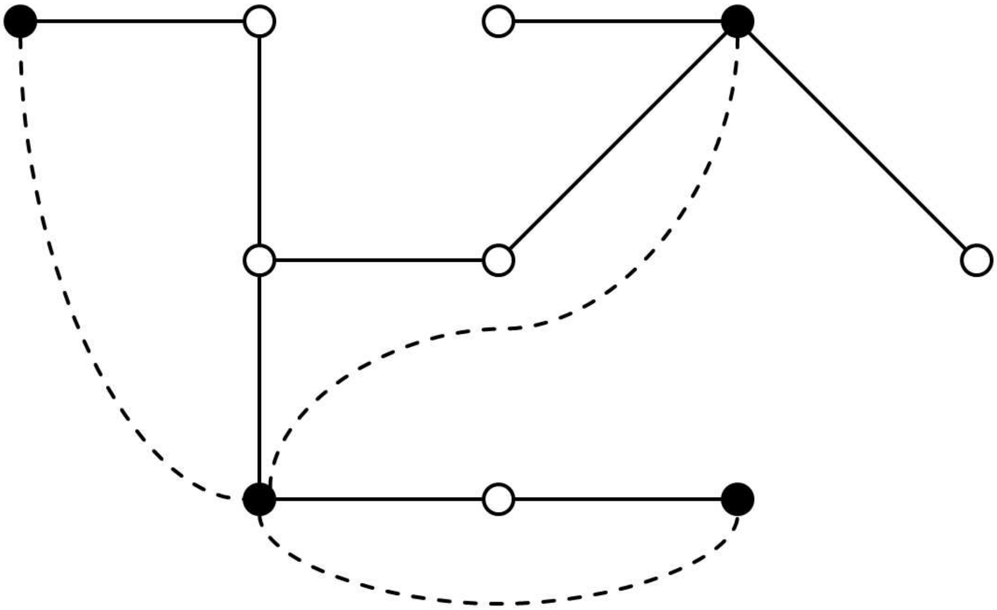

Fig. 2.5. CEDAR example network

2.2.2 Insignia

Insignia [3] is another QoS framework for MANETs. It employs the use of in-bound signaling to indicate bandwidth requirements for each QoS traffic flow. As a stream of QoS packets is transmitted on a certain path, the Insignia packet header informs each node of the amount of bandwidth to reserve for that stream. A reservation is recorded at each intermediate node as traffic is forwarded through that node. If traffic that uses the reservation is not received in a period greater than a pre-defined timeout, the reservation expires. This type of automatically-expiring reservation is termed a *soft reservation*. If a node is no longer capable of sustaining the specified bandwidth reservation, the source node is notified and a new route discovery is triggered. A key feature of Insignia is its independence from routing protocols. Insignia was developed as a means to offer quality of service support to non-QoS-based routing protocols.

2.2.3 Stateless Wireless Ad Hoc Networks (SWAN)

The Stateless Wireless Ad Hoc Networks (SWAN) protocol [4] was also developed by the author's of Insignia [3]. SWAN offers a stateless/reservation-less efficient and effective QoS framework. The two components in SWAN are a rate controller and an admission controller. The goal of SWAN is to adapt bandwidth usage according to the measured one hop delay to ensure that 1) delay does not become excessive and 2) network bandwidth does not become saturated to the point of significant packet loss and reduced throughput. SWAN improves upon INSIGNIA by abandoning the idea of bandwidth reservations in favor of efficient use of the RF medium on a node-by-node basis. This modification improves network efficiency. SWAN handles best effort (non-priority) traffic easily, and includes provisions for priority packets to enable quality of service support. QoS packets are given priority treatment in terms of admission and bandwidth usage. Upon network saturation, best effort packet throughput and delivery rates suffer before high priority throughput and delivery rates are impacted. In other words, by sacrificing best effort traffic delivery, a high network throughput of high priority traffic is maintained. In non-saturated conditions, high priority traffic continues to receive queuing preference, minimizing delay for QoS packets.

Chapter 3

DISJOINT PATH CLASSIFICATION

Before multi-path route discovery can occur, the criteria used to define a multi-path route must be firmly established. This task can be difficult as several of the multi-path protocols discussed in Chapter 2 include different criteria for determining multi-path sets. These criteria are usually defined by the protocol's behavior. For instance, MDSR only discovers paths containing mutually exclusive sets of nodes because MDSR does not forward duplicate route requests. As another example, GMR builds a graph representation of the network, but only computes routes we term *strict link disjoint.*

Despite a variety of multi-path route types, previous work generally agrees that *disjoint routes* are preferable [9,14]. Paths are *disjoint* if they are not mutually dependent upon the same network resources. However, the term *network resources* is interpreted in various ways. For instance, both SMR and GMR require two paths to share only the source and destination nodes, avoiding the use of a common intermediate node – a practice which may prevent the overuse of a common intermediate node's available bandwidth. On the other hand, SMR also attempts to find a maximally disjoint set of paths. This means that multi-paths do not share any intermediate nodes and, therefore, bandwidth due to neighbor traffic is minimized due to each path transmitting in a different part of the network.

Further constraints can be added to define a hierarchy of disjoint path criteria. Specifically, paths can be *node disjoint* or *link disjoint.* Two paths comprise a node disjoint route if the only nodes shared by both paths are the source and destination

(Figure 3.1). Two paths comprise a link disjoint route if the intermediate nodes in one path are not one hop neighbors of the intermediate nodes in the other path, with the exception of the source and destination (Figure 3.2). Additionally, we consider a relaxed version of the link disjoint route which states that the second and second to last hops of each path can be node disjoint (Figure 3.3). For future reference, we define these two versions of link disjoint routes as *strict link disjoint* and *loose link disjoint*, respectively. Lastly, we consider a relaxed version of the *node disjoint* type, *semi-node disjoint* routes, in which both paths can overlap, but cannot be identical. Figures 3.4 and 3.5 illustrate examples of *semi-node disjoint* routes.

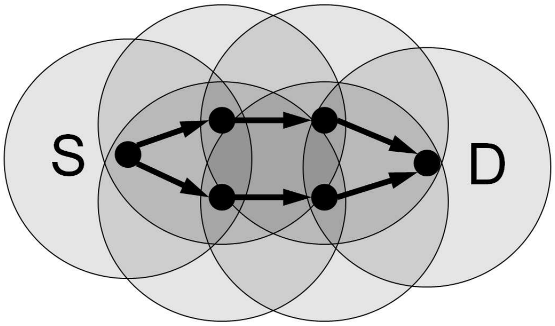

Fig. 3.1. Node disjoint route with two paths

We note that these disjoint route classifications make up a hierarchy, as depicted in Figure 3.6. Strict link disjoint routes are a more strict version of loose link disjoint routes; they require that no intermediate nodes in the first path be one hop neighbors with any intermediate nodes in the second path. Likewise, loose link disjoint routes are a more strict version of node disjoint routes; link disjoint routes are necessarily node disjoint. Lastly, a node disjoint route is a more strict version of a semi-node disjoint route; a node disjoint route includes two paths that are not identical, and do not overlap. Given that a wider distribution of network resources (e.g., bandwidth) is

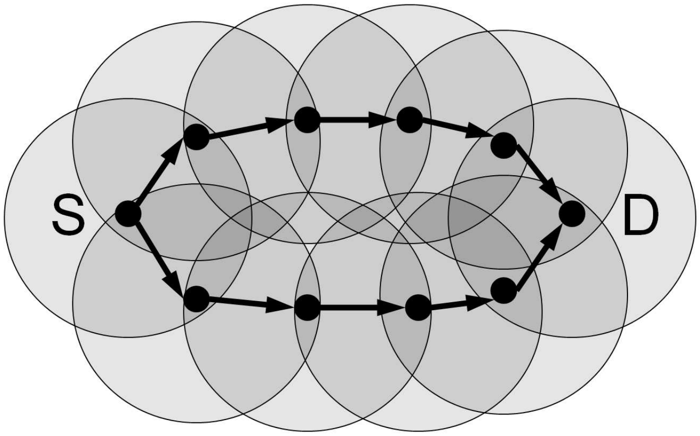

Fig. 3.2. Strict link disjoint route with two paths

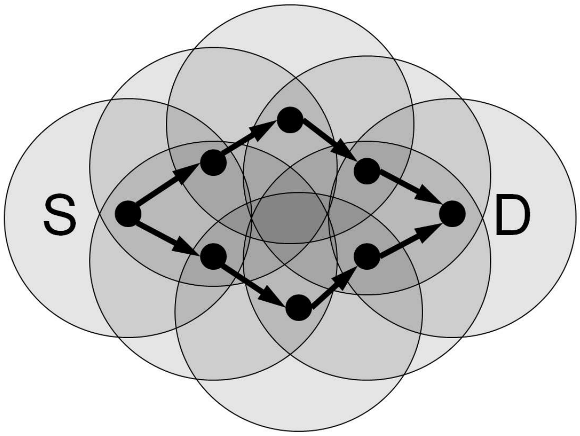

Fig. 3.3. Loose link disjoint route with two paths

generally more desirable than a narrower distribution, the value of the various multi-path route types increases as the paths comprising the routes become more disjoint. Thus, the value of the various disjoint route types increases as you move closer to the center of the diagram depicted in Figure 3.6.

Table 3 describes the multi-path route types discovered by each of the multi-path protocols covered in Chapter 2. The classifications for the disjoint route types discovered by each protocol follow the hierarchy in Figure 3.6; if a protocol is capable

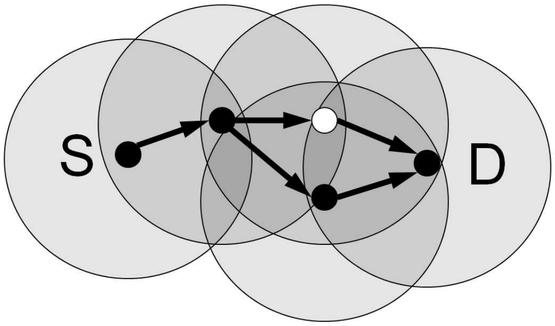

FIG. 3.4. Semi-node disjoint route with two paths of equal length

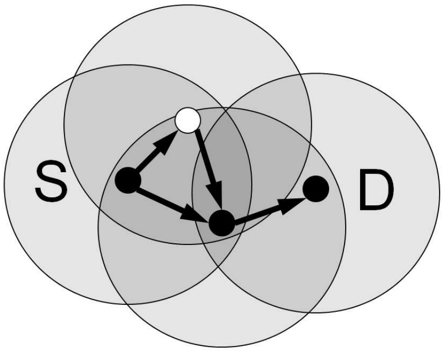

FIG. 3.5. Semi-node disjoint route with two paths of different length

of discovering a disjoint route of a certain type, then that protocol is also capable of discovering all of the other disjoint route types higher in the hierarchy. For instance, because DPSP discovers node disjoint routes, it also discovers link disjoint and strict link disjoint routes. We note the high variance in the definition of disjoint routes among the protocols published in the literature. DASM and SMR are two protocols capable of discovering each of the disjoint types in the hierarchy. Both DASM and SMR have relatively low standards regarding how disjoint their multi-path routes must be; neither protocol includes provisions for specifically prioritizing one disjoint route type over another.

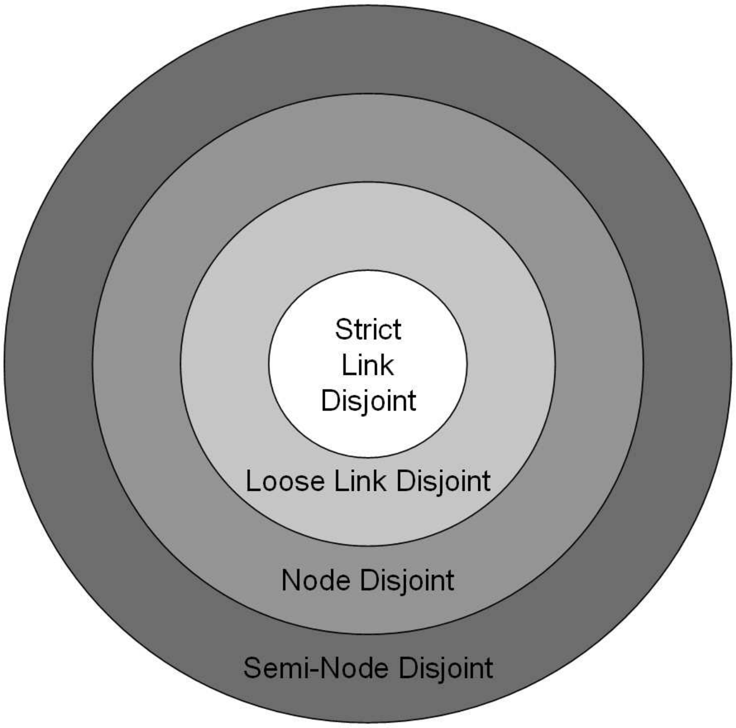

Fig. 3.6. Disjoint route classification hierarchy

	Semi-node Disjoint	Node Disjoint	Loose Link Disjoint	Strict Link Disjoint
DASM	X	X	X	X
DPSP		X	X	X
MDSR:				
Protocol 1		X	X	X
Protocol 2	X	X	X	X
GMR				X
SMR	X	X	X	X

Table 3.1. Relationships between disjoint route types and existing multi-path protocols

Chapter 4

PDMR

The development of the Proactive Disjoint Multi-path Routing (PDMR) protocol was motivated by the lack of a universally agreed upon definition of *disjoint multi-path routing protocol*. Although the underlying concept of leveraging multi-paths to improve network performance is shared among several protocols, each protocol presents a different solution and, therefore, solves a slightly different problem. Unlike the previously developed protocols, we incorporate the disjoint route hierarchy (illustrated in Figure 3.6) into PDMR. To evaluate the benefits of multi-path routing, we chose to leverage the Legend Exchange and Augmentation Protocol's [20] information dissemination technique to disperse global network information to every node. This has two benefits. First, it is an efficient means of information dissemination and, conceptually, separates information gathering and route computation in the route discovery process. Second, it provides enough information to allow each node to perform single path routing or multi-path routing with identical network overhead. Thus, although PDMR is not designed for optimal performance, it is ideal for a single path versus multi-path comparison.

PDMR can be divided into four conceptual components, which together describe PDMR's high-level routing methodology. The first component is a topology information dissemination mechanism. Once topology information has been distributed throughout the network, a route computation component uses the topology information to discover multi-path routes. A set of policies regarding the efficient use of these multi-path routes forms the third component. Finally, SWAN is employed for admission and rate control to enable efficient use of the RF medium on a per-hop basis. The following four subsections cover each of these four components in further detail.

4.1 Network Topology Dissemination

The necessity for accurate and up-to-date routing information provides an argument for using a proactive approach. LEAP, or the Legend Exchange and Augmentation Protocol [20], is an ideal candidate for proactively distributing topology information to all nodes within the network. LEAP relies on the concept of a *legend*, or a global entity containing a table of relevant node state information that travels through the network, choosing the next destination according to a *least recently visited* (LRV) heuristic. LEAP is a simple, efficient all-to-all broadcast which ensures that every node connected to the network is updated at a reasonable interval with whatever information is contained in the legend. Upon receiving the legend, a node will update the legend table with any information it may have that is more recent than the legend's current contents, and make a local copy of the legend. The legend is then forwarded to the next hop. LEAP enables promiscuous updates; a node that hears the legend will update its local information with the more up-to-date information contained in the legend, regardless of the legend's intended next hop destination. Only the addressed destination, however, will modify and retransmit the legend.

If the legend continuously hops through the network without pausing, a significant amount of traffic is incurred with, potentially, little improvement on the accuracy of the legend contents. To address this issue, LEAP has three parameters that can be tuned for different mobility scenarios and traffic situations:

- Maximum hops before pause: The number of hops before the legend is paused at its current location. When the pause time expires and the legend is forwarded, the counter for this value is reset.

- Pause time: The length of time the legend waits to be retransmitted when the maximum hops before pause has been reached.
- Wait time: The delay between each retransmission of the legend.

For the purposes of our PDMR protocol, the legend has been configured to contain the following fields:

- Neighbor table: A table listing each node's neighbors at a given time.
- Last visited time: A time stamp indicating the last time a given node was visited by the legend.
- Bandwidth usage: A table indicating the last perceived bandwidth usage at each node. PDMR calculates a node's bandwidth usage by observing packets transmitted by the node and its neighbors.

When the information within the legend is disseminated throughout the network, every node connected to the network has the information necessary to build multi-path QoS routes for any source/destination pair.

4.2 Multi-path Discovery Algorithm

Intuition suggests that link disjoint routes are more valuable than node disjoint routes due to their more distributed use of the RF medium by a single data stream. However, while strict link disjoint routes are the preferred multi-path route type, they have the most stringent restrictions and are, therefore, less common in a given network. Likewise, loose link disjoint routes are more valuable than node disjoint routes, because loose link disjoint routes only have slightly fewer requirements than strict link disjoint routes. Of course, node disjoint routes are more common than either of the link disjoint route types. Finally, the least valuable multi-path type is a subset of node disjoint routes, i.e., semi-node disjoint, and can include paths that differ in as little as a single node. We note that, technically, strict link and loose link disjoint routes are also node disjoint. Furthermore, node disjoint routes are also semi-node disjoint. This hierarchy of multi-path route criteria provides the basic framework for PDMR's multi-path discovery algorithm.

The first step in PDMR's multi-path discovery algorithm removes all nodes other than the source and destination that exceed a perceived bandwidth usage threshold. This step removes nodes which have limited bandwidth available. Dijkstra's shortest path algorithm is then employed to find the shortest path from the source to the destination with the remaining nodes and links. If no path results from this search, the packet is dropped and an error is reported. In the case where the route between the source and destination is only a single hop, a multi-path approach is unnecessary and the single hop route is returned. Additionally, PDMR is configured by default to return only single path routes for best effort traffic.

For high priority/QoS traffic, PDMR will attempt to discover a second path between the source and destination; specifically, if the first path found is a multi-hop path, a strict link disjoint filter based on the first path is applied to the network topology data and Dijkstra's algorithm is run again. If no second path is found, a loose link disjoint filter is applied. If still no path is found, a node disjoint filter is applied. If no node disjoint path is discovered, a semi-node disjoint filter is applied. If no second path can be found, a single path is returned. Note that this method of multi-path discovery makes finding the shortest path from source to destination a priority. Only after this shortest path has been found does PDMR make an effort to find an additional path. Table 4.1 summarizes our multi-path discovery algorithm. The input to the pseudocode is a graph representation of the network topology, which is created by using the information received during the network topology dissemination component of PDMR.

```
Remove bandwidth constrained nodes
Search for a shortest path
if no shortest path exists
 return Route Not Found
else if  a one hop path is found
 return One Hop Path
else if  best effort traffic
 return single path route
Search for a strict link disjoint path
if no path exists
 Search for a loose link disjoint path
 if no path exists
  search for a node disjoint path
  if no path exists
   search for a semi-node disjoint path
   if no path exists
    return No Multi-path Found
   else
    return Semi-Node Disjoint Path
  else
   return Node Disjoint Path
 else
  return Loose Link Disjoint Path
else
 return Strict Link Disjoint Path
```

Table 4.1. Pseudocode describing the PDMR route discovery algorithm

4.3 Multi-path Use Policies

PDMR is implemented with a route cache which contains previously calculated routes. Upon the legend's arrival containing new topology information, the routes in the cache are invalidated, and the node triggers a new multi-path discovery as soon as the node has data to send. Routes are computed only for packets originating locally on a node; in other words, *source routing* is used. Source routing is an important component of PDMR, as it enables the application of the disjoint route type hierarchy. If source routing were not used, intermediate nodes would not have the capability to maintain disjoint routes, unless additional routing overhead was added. Also, due to source routing, the number of path computations is equal to the number of destinations a node will send to within the legend inter arrival time.

In multi-path mode, if PDMR has discovered multi-paths for a route, it will alternate sending packets between these two paths. This path usage policy was chosen to study the potential load balancing effect of multi-path routing. As the packet travels along a given path, PDMR incorporates a packet acknowledgment policy. Specifically, each forwarding node listens promiscuously for the next forwarding node's retransmission of the packet. Upon the packet's arrival at its destination, a PDMR acknowledgment packet is transmitted from the destination on the reverse path. Should a forwarding node not receive either type of acknowledgment, it retransmits a maximum of two more times before dropping the packet. A maximum of three transmissions of a data packet at a forwarding node was chosen to minimize traffic storms caused by traffic collisions in congested networks, while still maintaining a method for recovering from transmission collisions. A route error packet is not returned to the source, because the source will discover a failed route via the next topology update provided by the legend.

4.4 The Role of SWAN

SWAN offers rate and admission control to enforce low one hop delays and prevent excessive bandwidth usage. This rate and admission control prevents saturation of the network, maintains a high throughput, and includes preferential treatment for QoS traffic. SWAN does not depend on any network layer implementation; in other words, SWAN offers identical QoS provisioning for any routing protocol, making it ideal for comparing the route discovery methods of different protocols. In addition to motivations based on performance comparisons, SWAN's stateless approach to handling quality of service traffic is arguably ideal for mobile ad hoc networks.

Chapter 5

SIMULATION RESULTS

In this chapter, we first present simulation results indicating the statistical characteristics of PDMR routing; specifically, we examine route type frequencies, average route lengths, and the average length differences between the two paths included in multi-path routes generated by PDMR. We then evaluate the effectiveness of multi-path routing by comparing PDMR in single path and multi-path modes under various traffic conditions. As these two modes require similar network traffic overhead, their relative performance is a reasonable indicator for the general effectiveness of multi-path routing compared to single path routing. We then examine the effectiveness of PDMR single and multi-path routing with various fixed route lengths. Finally, we present the results of a contrived traffic scenario, specifically designed to emphasize the benefits of multi-path routing.

5.1 Environment

All simulation results presented in this investigation were generated using ns-2 version 2.26 [25] with the parameters summarized in Table 5.1. For most of the results presented in this chapter, the network is comprised of 100 nodes. A few exceptions include any results that explicitly present data with a varying node count. In those cases, the number of nodes used in the network is specified in the figure. In Table 5.1, node density refers to the ratio of the number of nodes to the network area. Coverage area is the area of the circle with radius equal to one node's transmission range. A node's transmission footprint is the percentage of the network area covered by a node's coverage area. The maximum path length is the length of a diagonal of the simulation area. The network diameter is the maximum path length divided by a node's transmission range. The network connectivity is based on the average number of neighbors.

Input Parameters	
Number of Nodes	Max. 100
Simulation Duration	1,250 seconds
Simulation Area Size	300 m x 600 m
Transmission Range	100 m
Derived Parameters	
Node Density	1 node per 1,800 m^2
Coverage Area	$31,416\ m^2$
Transmission Footprint	17.46%
Maximum Path Length	671 m
Network Diameter (max. hops)	6.71 hops
Network Connectivity (node degree)	17.4
Simulator	
Simulator	NS-2 (version 2.26)
Medium Access Protocol	IEEE 802.11 (SWAN)
Link Bandwidth	11 Mbps
Number of trials	10
Confidence Interval	95%

Table 5.1. Simulation parameters.

Very low or zero mobility is ideal for comparing route criteria, as it eliminates mobility as a factor in delivery rates. However, mobility simulations require significant routing traffic overhead, because they require up-to-date routing information. Simulations with zero mobility also require accurate routing information, but because the network topology is static, there are no requirements regarding the age of the topology data used in routing. For these reasons, we consider both static network models (networks without mobility) and mobile network models. This ensures that the results presented give an accurate account of the effectiveness of the protocol's path decisions and policies. Table 5.2 lists the parameters used for the generation of static network models. Table 5.3 lists the parameters used for the generation of mobile network models. Although these models are mobile, the average node speed is only 1 m/s. With average node speeds higher than 1 m/s, the protocol's drawbacks impact delivery rates far more than whether the protocol is running in multi-path mode or single-path mode. In other words, because PDMR is designed for the evaluation of various disjoint route types, and for the comparison of multi-path routing versus single-path routing, it does not include features that might improve delivery rates in a dynamic network. When delivery rates are drastically affected by the lack of these features, single- versus multi-path performance comparisons become impossible. Examples of these missing features include:

1. *No mid-route improvements.* Although intermediate nodes may have information to improve upon the route the source has chosen, these nodes do not redirect traffic. Routing decisions made at the source regarding what paths are chosen can not be modified by intermediate nodes. Otherwise, the chosen disjoint route type may not be used.

2. *No route failure messages.* Route failure messages are not implemented in PDMR. PDMR includes link-by-link provisions to help ensure delivery of a packet. However, if a packet cannot be delivered and the forwarding node does not receive confirmation of the packet's delivery at the next hop, the packet is simply dropped; no message is sent upstream to the source. This implies that the source continues to send packets to the destination on a route that has failed. For a multi-path versus single-path comparison, it is sufficient to allow the source to be notified of route failures via the legend.

Static Network Model Parameters	
Initialization	Steady State Random Waypoint [23]
Number of Nodes	25, 50, 75, 100
Average Speed	0 m/s
Pause Time	N/A

Table 5.2. Network model parameters for static (0 m/s) simulations.

Mobile Network Model Parameters	
Mobilty	Steady State Random Waypoint [23]
Number of Nodes	25, 50, 75, 100
Average Speed	1 m/s ± 1 m/s
Pause Time	10 s ± 1 s

Table 5.3. Network model parameters for mobile (1 m/s) simulations.

In both mobile and static simulations, we use the same traffic models. We considered two types of traffic: QoS traffic and best effort traffic. Both types of traffic included packets with 20 bytes of non-PDMR headers, 18 bytes of PDMR header, and 160 bytes of data payload. Both types of traffic were generated using the Constant Bit Rate traffic generator included in ns-2.26 [25]. A best effort traffic session used 0.2 second packet send intervals for a total maximum payload throughput of 800 bytes per second. A QoS session used 0.1 second packet send intervals for a total maximum payload throughput of 1600 bytes per second. For every simulation, two parameters were given to describe a particular traffic environment: the number of concurrent best effort traffic sessions and the number of concurrent QoS traffic sessions. In general, source/destination pairs were chosen such that no node would be the source and/or destination for more than one session at a time. This was enforced by choosing source/destination pairs by their addresses. In situations when the number of sessions required more nodes than those available, some pairs were forced to support more than one session at a time. Every traffic session had a lifetime of 20 seconds, and then a different source/destination pair started a new session. This pattern continuously changed the network traffic environment, exercising various routes throughout the network in every simulation. These traffic parameters have been chosen to both simulate a potential real world situation, and to observe the effect of various levels of traffic severity.

5.2 Static Network Models

To compare single path performance and multi-path performance, we considered two versions of PDMR. The first version of PDMR included multi-path routing, as previously described. The second version of PDMR included a route discovery process that only returned a single path, regardless of the availability of multi-paths. The topology dissemination mechanism and SWAN medium access components remained unchanged. Because the two versions differed only in the routing mechanism, we consider the evaluation of the two versions a reliable means to compare single- and multi-path routing.

The motivation for using multi-path routes to improve throughput includes the assumption that the intermediate nodes along a single path have the following constraints:

1. The source and destination nodes are not bandwidth constrained.
2. Intermediate nodes on the multi-paths are under load such that a) no single path can satisfy the bandwidth requirement, and b) considered together, their bandwidth capacity is sufficient to satisfy the bandwidth requirement.

With these constraints in mind, we chose to simulate networks containing both quality of service (multi-path) traffic sessions as well as single path best effort traffic at various levels of traffic severity such that we may test for the above constraints in a variety of situations.

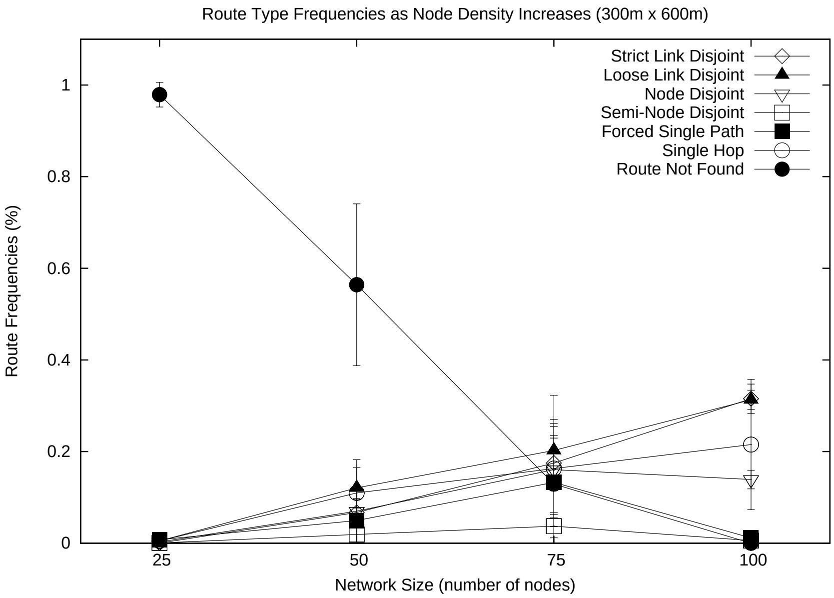

FIG. 5.1. Route frequencies for various network types (including *Route Not Found* error frequencies) as the number of nodes in the network increases

Figures 5.1 and 5.2 illustrate the frequency at which each route type is discovered. We simulated within a 300m x 600m network region, and we varied the number of nodes from 25 to 100. The key in figures 5.1 and 5.2 indicates the following route types: *Strict Link Disjoint*, *Loose Link Disjoint*, *Node Disjoint*, *Semi Disjoint*, *Forced Single Path*, *Single Hop*, and *Route Not Found*. The disjoint route types are the same as those discussed in Chapters 3 and 4. *Forced Single Path* implies the route discovery process was unable to find a disjoint path route, but was able to find a single path route. *Single Hop* indicates the source and destination nodes are one-hop neighbors. In this case, a multi-path route is not necessary as the destination node is capable of receiving all transmissions from the source node directly. *Route*

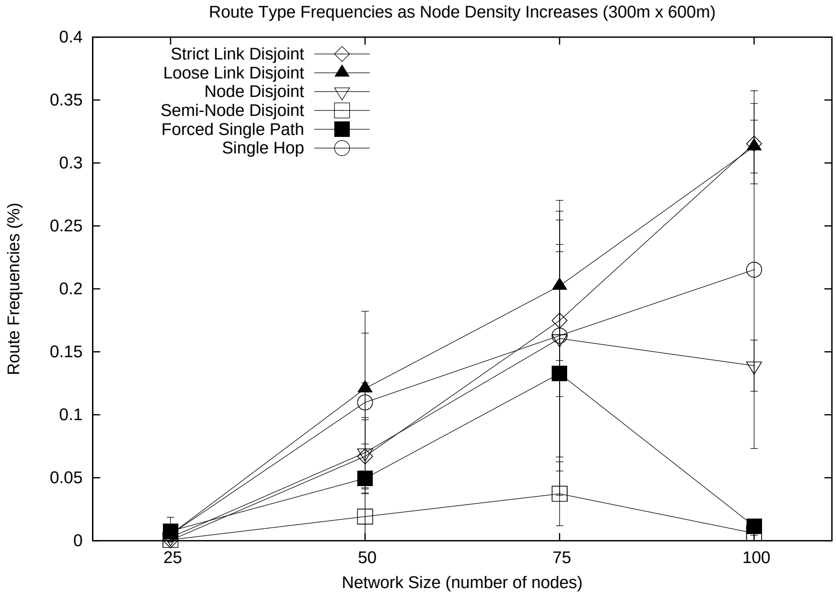

FIG. 5.2. Route frequencies for various network types (without *Route Not Found* error frequencies) as the number of nodes in the network increases

Not Found errors imply that no known usable path exists in the network between the source and destination. A *Route Not Found* error can be caused by one of the following situations:

1. The source and destination nodes are in different partitions.
2. The local copy of the legend at the source node does not include enough topology information to find a route to the destination.

The difference between Figure 5.1 and Figure 5.2 concerns the frequency of *Route Not Found* errors. We include the frequency of *Route Not Found* errors in Figure 5.1; we omit this frequency in Figure 5.2 for clarity. Figure 5.1 illustrates that as the node count increases, the *Route Not Found* percentage decreased significantly. In fact, an increase in node density eliminated *Route Not Found* errors. Figure 5.2 illustrates that the percentage of all route types increased as the number of nodes increased from 25 to 75. As the number of nodes increased from 75-100, the stringent link disjoint route types continued to increase in frequency; in other words, an increase in network density yields more opportunity to discover multi-path link disjoint routes. The large confidence intervals on some of the data points in Figure 5.1 and Figure 5.2 indicate that route type frequencies are sensitive to network topology; given two networks comprised of the same number nodes and constrained to the same network region size, but with different node distributions, route frequency types may be significantly different.

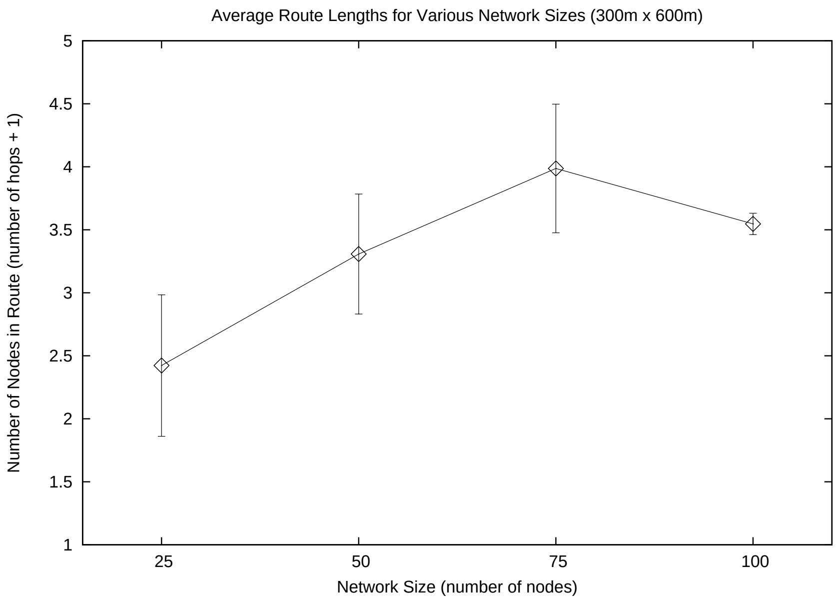

FIG. 5.3. Average route lengths for networks simulated within a region of 300m x 600m, as the number of nodes in the network increases

Figure 5.3 shows the average route lengths for the networks examined in figures 5.1 and 5.2 as the number of nodes increases. Similar to the previously presented data, these results were highly sensitive to the network region's size and shape and the network's node density. The results in Figure 5.3 show the average shortest path lengths. For networks consisting of 25 nodes, routes contained an average of approximately 2.5 nodes. However, a simulation area of 300m x 600m with 25 nodes tends to include a high amount of partitioning (illustrated by the high frequency of *Route Not Found* errors shown in Figure 5.1). In partitioned networks, the legend can only be transmitted to those nodes within the partition containing the legend. All nodes outside of this region receive no network topology data, and thus all of their route discoveries will result in *Route Not Found* errors. Given that a high number of *Route Not Found* errors occur with PDMR operating in low-density networks, the low average node count per route, indicated in Figure 5.3, is not surprising.

Figure 5.3 also indicates that the average route length increased as the network density increased. Again, comparing Figure 5.3 with Figure 5.1, as the network density increased to the point where *Route Not Found* errors were eliminated (i.e., no partitioning in the network), the average route length settled at approximately 3.5 nodes per route. Finally, once network partitioning is eliminated (at 100 nodes), Figure 5.3 indicates a smaller confidence interval. In networks with partitioning, only those nodes near to each other may communicate, yielding low route lengths. However, although one network may be partitioned, another generated with the same parameters may not be, implying longer average route lengths for the second network. These types of discrepancies in average route lengths are what account for the large confidence intervals in Figure 5.3. Furthermore, the gradual increase in route length exists because of the decrease in network partitioning; as network partitioning decreases, longer routes become available. Once network partitioning is eliminated at around 100 nodes, additional network density enables straighter routes (and therefore shorter routes, in terms of hops). This fact explains the apparent slight drop in average route lengths from 75 to 100 nodes.

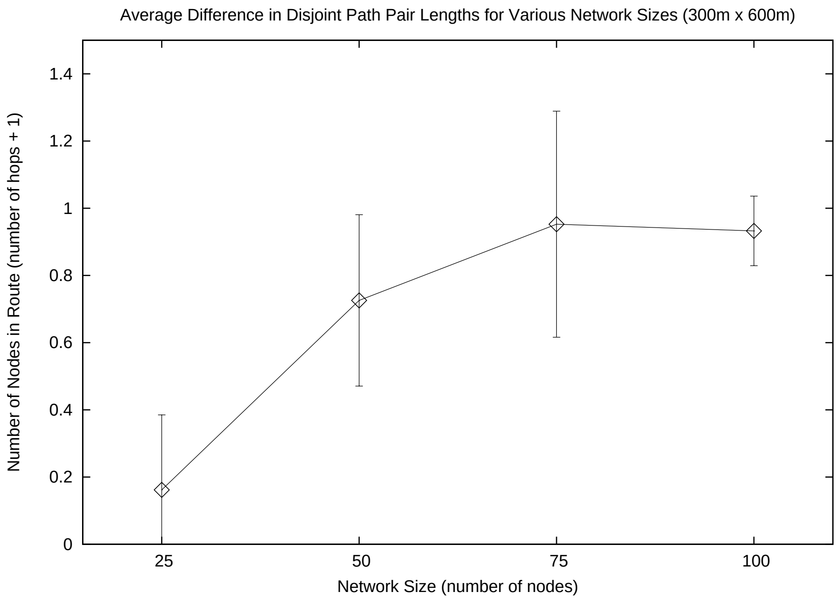

FIG. 5.4. The average difference between the two paths included in each multi-path route, as the number of nodes in the network increases

Figure 5.4 shows the average difference in path lengths between the shortest and second paths discovered by PDMR. Similar to Figure 5.3, the confidence interval in Figure 5.4 decreased as the network's density became sufficient to avoid partitioning. We note that the average difference between the shortest path and a path disjoint to the shortest path was approximately one hop.

Figure 5.5 illustrates the total delivery rates of seven simultaneous traffic sessions with varying numbers of best effort traffic in a 100 node network. The figure, similar to several of the following figures, indicates no significant difference between multi-path and single path routing methods. Given that multi-path routes are only used for QoS traffic, and that best effort traffic uses single path routes regardless of the presented routing modes, it is not surprising that no significant difference was found between

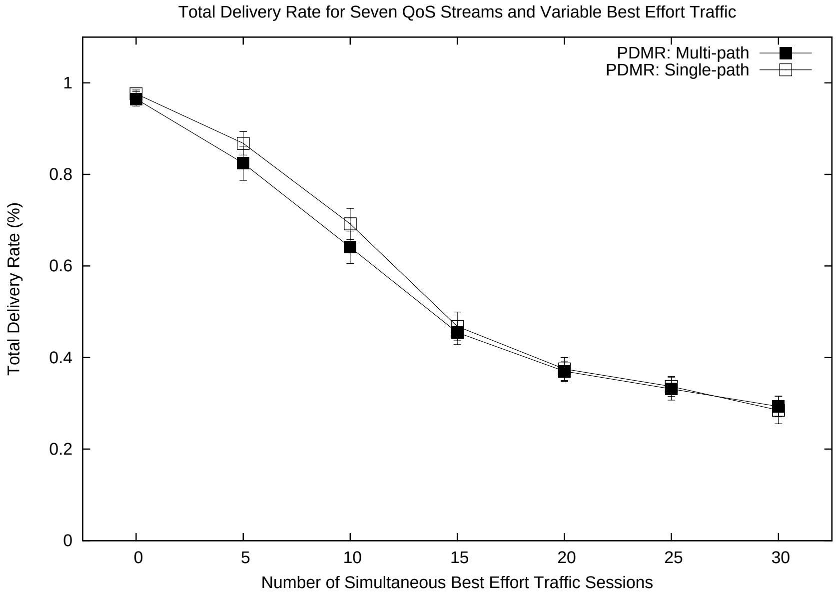

FIG. 5.5. Total delivery rate for a static network with seven concurrent QoS traffic sessions as the number of simultaneous best effort traffic sessions varies

single and multi-path routes for best effort traffic (shown in Figure 5.6). Multi-path routing has either a neutral or negative effect on best effort delivery rates, because best effort traffic only uses single paths; as QoS traffic increases in the network, the SWAN access controllers will ensure that QoS traffic receives priority over best effort traffic. Figure 5.6 indicates that when there were ten active best effort traffic sessions, and seven active QoS traffic sessions, single path PDMR actually performs better than multi-path PDMR. Although PDMR best effort traffic uses single path routes regardless of what type of routing is used for QoS traffic, multi-path QoS traffic can have a detrimental effect on best effort delivery rates. This is caused by the additional paths requiring more hops in many instances and, thus, additional congestion is added to the network.

We present QoS delivery rates for varying best effort traffic levels in figures 5.7-5.9. Each of these figures indicates results from simulations, including a fixed number

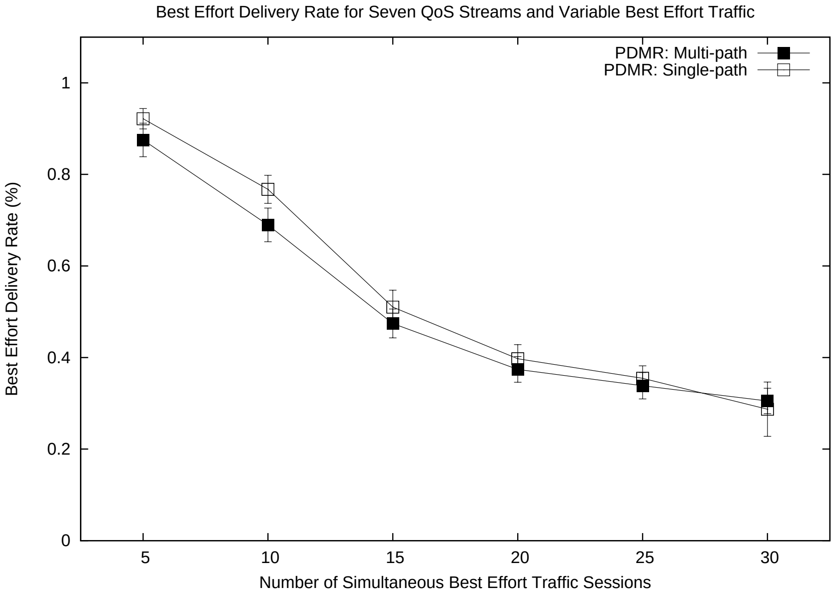

FIG. 5.6. Best effort delivery rate for a static network with seven concurrent QoS traffic sessions as the number of simultaneous best effort traffic sessions varies

of best effort traffic sessions and a variable number of QoS traffic sessions. Figure 5.7, which includes seven QoS traffic sessions, indicates a relatively large decrease in delivery rates between zero and fifteen best effort traffic sessions. Figure 5.8 includes four QoS traffic sessions rather than seven, and indicates a large decrease in delivery rates between ten and fifteen best effort traffic sessions. Finally, Figure 5.9 includes only one QoS traffic session. The delivery rate in Figure 5.9 decreases substantially between fifteen and twenty best effort traffic sessions. This progression reveals a delivery rate cliff where network traffic has reached such an intense degree that QoS delivery rates are significantly negatively impacted. For instance, Figure 5.9 maintained high delivery rates with one QoS traffic session and up to ten best effort traffic sessions. Comparing Figure 5.9 to Figure 5.8, which has three additional QoS traffic sessions, reveals that QoS traffic delivery rates were more sensitive to additional QoS traffic than to additional best effort traffic. This was caused in part by SWAN's access

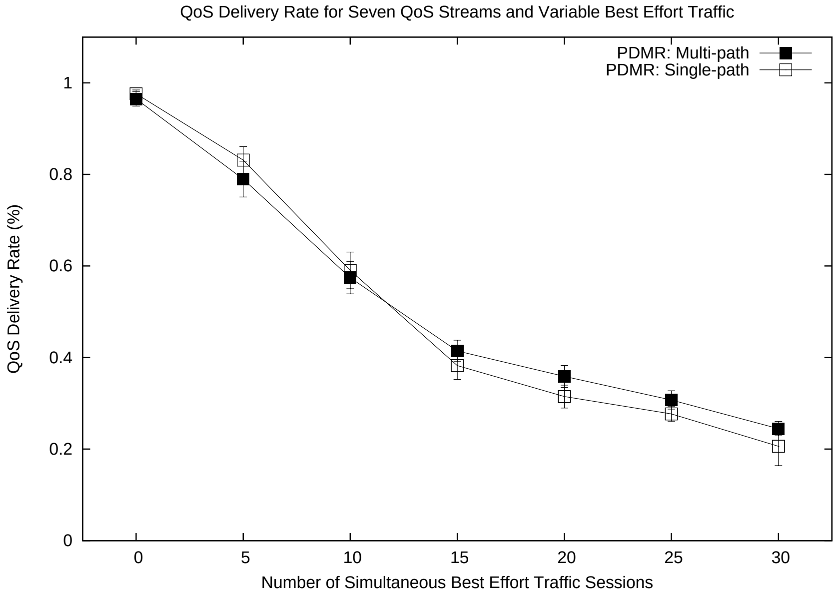

FIG. 5.7. QoS delivery rate for a static network with seven concurrent QoS traffic sessions as the number of simultaneous best effort traffic sessions varies

controllers giving priority to QoS traffic over best effort traffic. The cliff also indicates that traffic severity continues to be the largest contributing factor to protocol performance. Figure 5.8 shows a statistically significant improvement in delivery rate for multi-path PDMR and QoS traffic when the number of best effort traffic sessions is more than 15. Although the improvement is statistically significant, it is not large and is even more questionable in other traffic conditions. This implies that the use of multi-path routing offers only a small improvement, and is highly dependent on network traffic conditions.

Figures 5.10 and 5.11 indicate QoS delivery rates for 25 and 30 concurrent best effort traffic sessions, respectively. Because best effort traffic represented the majority of the traffic in each of the simulated networks, changing the amount of QoS traffic had little impact on delivery rates. However, we note that there is a statistically significant difference between the two routing modes in several cases, indicating a

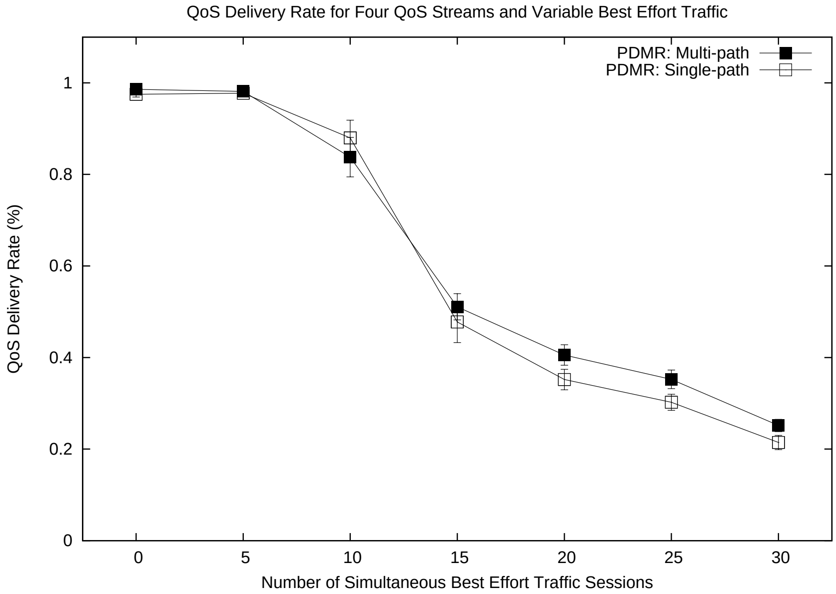

FIG. 5.8. QoS delivery rate for a static network with four concurrent QoS traffic sessions as the number of simultaneous best effort traffic sessions varies

small improvement in delivery rates provided by multi-path routing. However, the improvement is relatively small. This is due to few occurrences of the requirements needed to obtain benefit from multi-path routing (see Section 5.2).

5.3 Mobile Network Models

In this section, we add mobility to our simulation environment. The simulation parameters are identical to the parameters used for static network models except for the addition of mobility (see Table 5.3). Figures 5.12 and 5.13 illustrate results from simulations including mobility and indicates PDMR performance in both single and multi-path modes, with seven concurrent QoS traffic sessions and varying numbers of concurrent best effort traffic sessions. Delivery rates in figures 5.12 and 5.13 are less than those in figures 5.5 and 5.7, which are the same figures using a static network

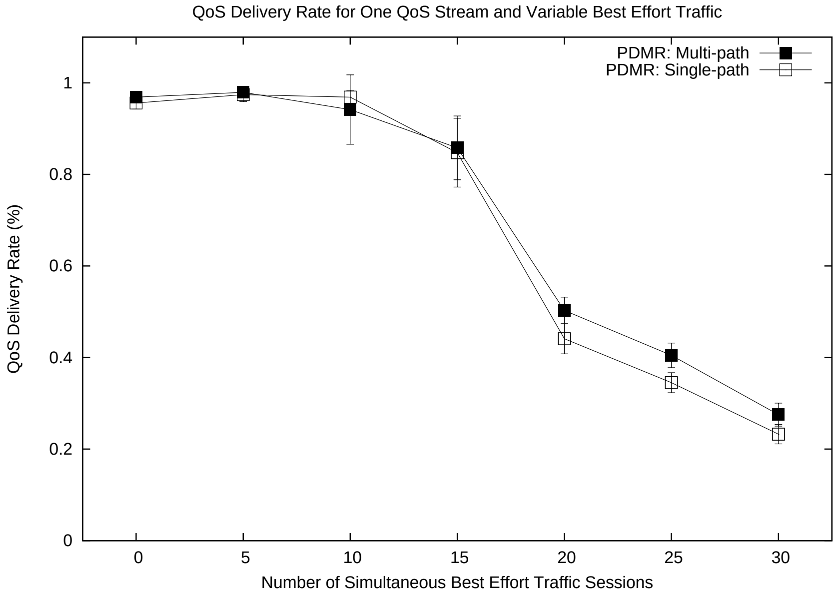

Fig. 5.9. QoS delivery rate for a static network with one QoS traffic stream as the number of simultaneous best effort traffic sessions varies

model. We note that confidence intervals increased in the mobile network results. Larger confidence intervals are likely due to variability in the results because of details in the protocol including:

- *LEAP Parameter Tuning.* LEAP parameters impact both the amount of routing overhead traffic in the network, and how accurate each node's local legend copy is.
- *Pure Source Routing.* PDMR only uses source routing, ignoring any potential more up-to-date information available at intermediate nodes.

We also note that the trends we discovered in both the static and mobile simulations are essentially identical (e.g., the delivery rates of shortest path and multi-path routing methods are statistically identical).

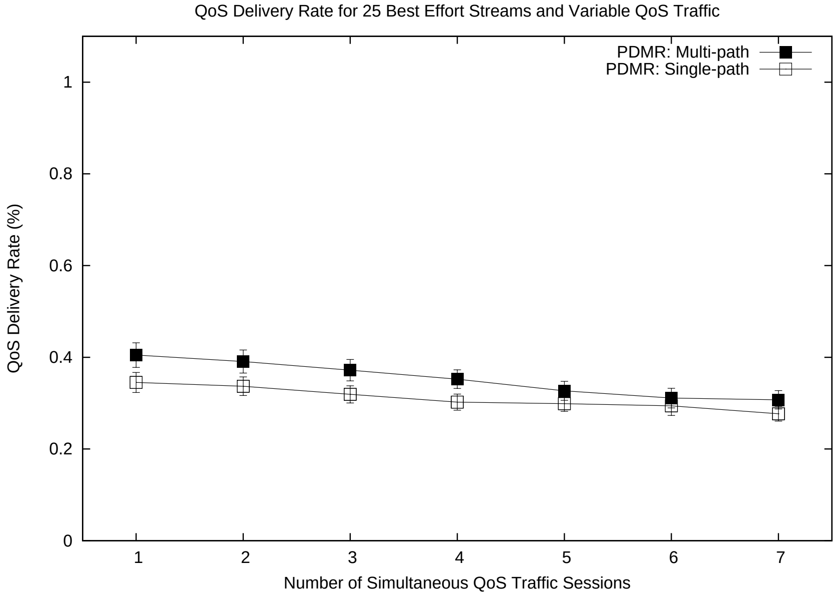

FIG. 5.10. QoS delivery rate for a static network with 25 best effort traffic sessions as the number of simultaneous QoS traffic sessions varies

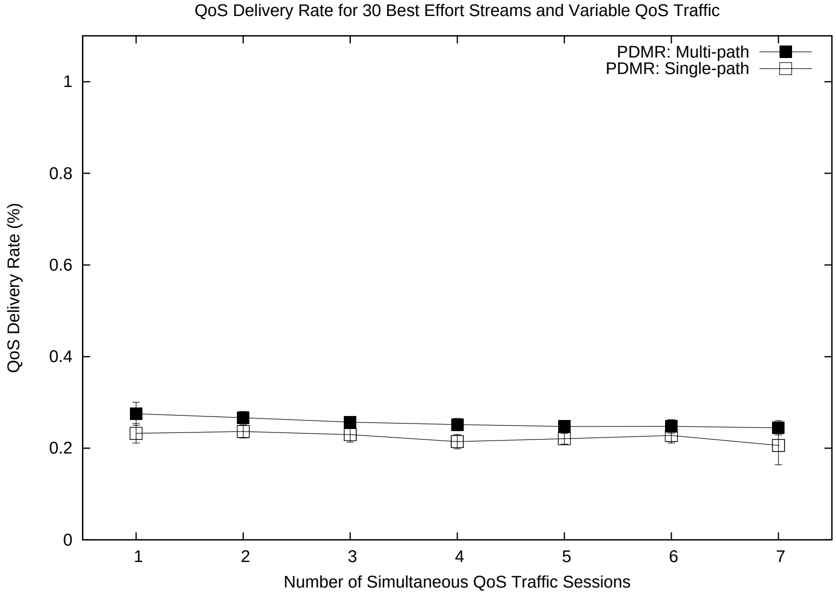

FIG. 5.11. QoS delivery rate for a static network with 30 best effort traffic sessions as the number of simultaneous QoS traffic sessions varies

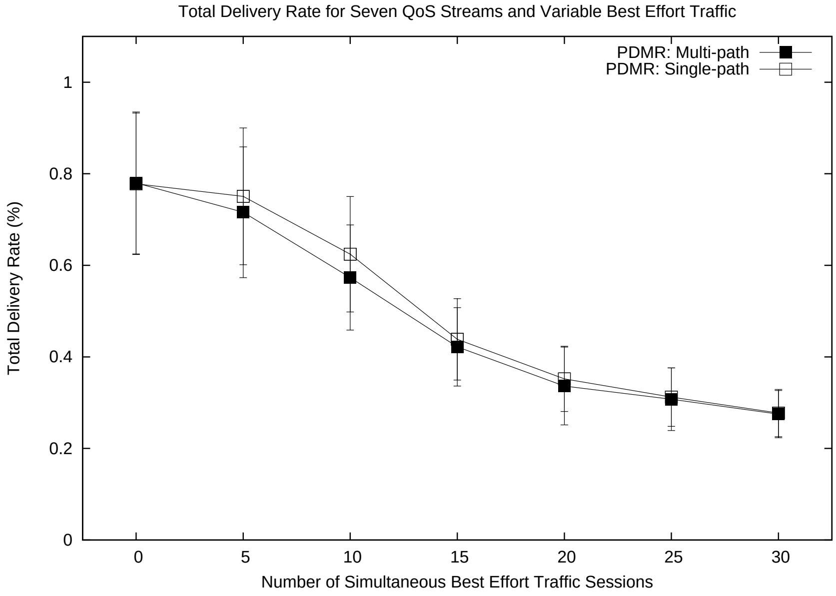

FIG. 5.12. Total delivery rate for a mobile network with seven concurrent QoS traffic sessions as the number of simultaneous best effort traffic sessions varies

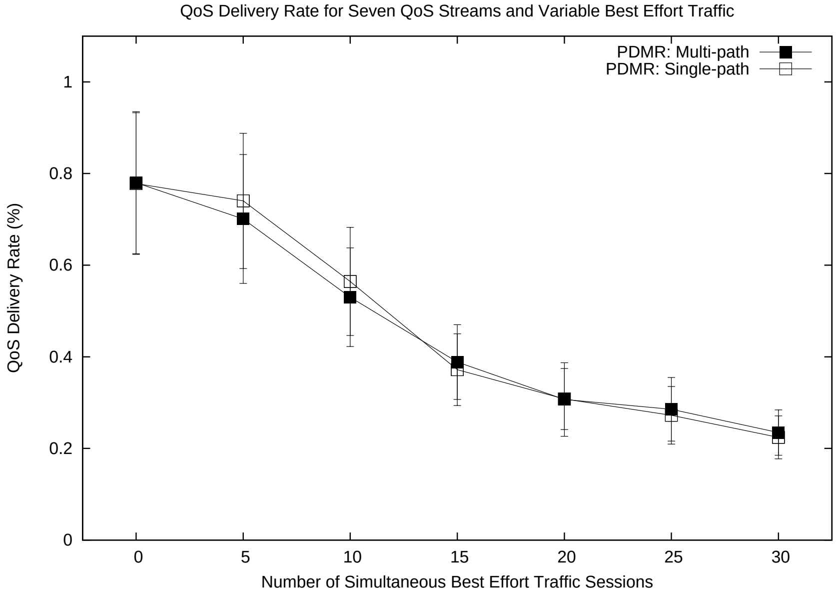

FIG. 5.13. QoS delivery rate for a mobile network with seven concurrent QoS traffic sessions as the number of simultaneous best effort traffic sessions varies

Finally, we tested PDMR in single and multi-path modes with a further modification; rather than specify a destination for every source node, we specify a specific route length. During the route discovery process, the source node searches for a destination node that can be reached with a shortest path of the specified length. Observing performance at fixed route lengths gives an indication of the value of multi-path routing at various route lengths. For instance, multi-path PDMR offers no benefit for single hop routes, as routing is identical in single hop and single path routes. However, longer routes may offer more opportunity to discover disjoint routes.

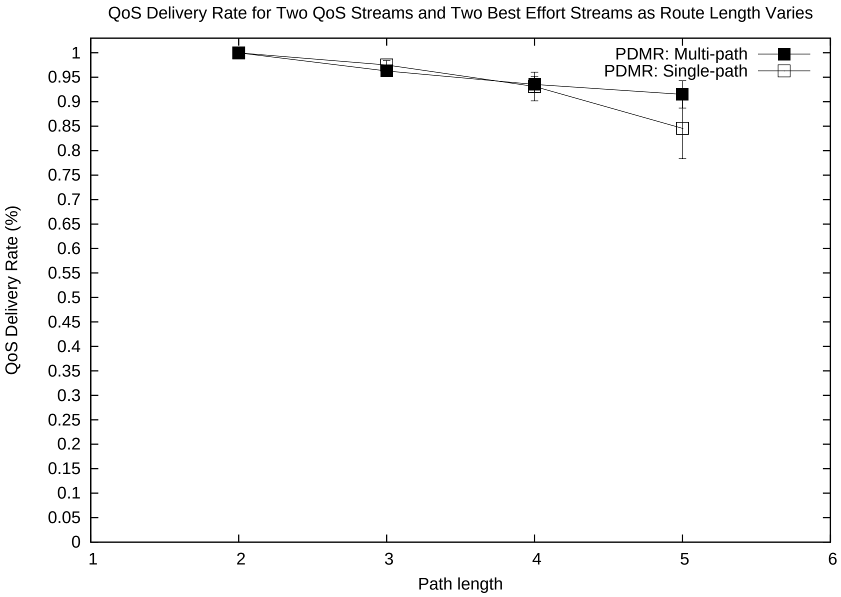

FIG. 5.14. QoS delivery rate for a mobile network with two concurrent QoS traffic sessions, two concurrent best effort traffic sessions, and various fixed route lengths

Figure 5.14 indicates delivery rates for a network with two concurrent QoS traffic sessions, two concurrent best effort traffic sessions, and various route lengths. Given that only four traffic sessions are present in the network at any given time, the large confidence intervals at a path length of five are not surprising. We suspect that the four traffic sessions will sometimes compete for the same network resources (e.g., when they are all routing through the same parts of the network), and will sometimes have no competition at all (e.g., when they are all routing through separate parts of the network). The varying competition for network bandwidth will lead to a large variance in delivery rates for small numbers of traffic sessions. On the other hand, with a large number of traffic sessions, the average available bandwidth along any given route will be much closer to constant, causing a decrease in delivery rate variance. Despite this fact, Figure 5.14 illustrates that single and multi-path PDMR perform nearly identically for short routes. The similar performance between single- and multi-path PDMR was caused by a light traffic load that is incapable of creating the bandwidth constrained situations in which multi-path PDMR offers benefits.

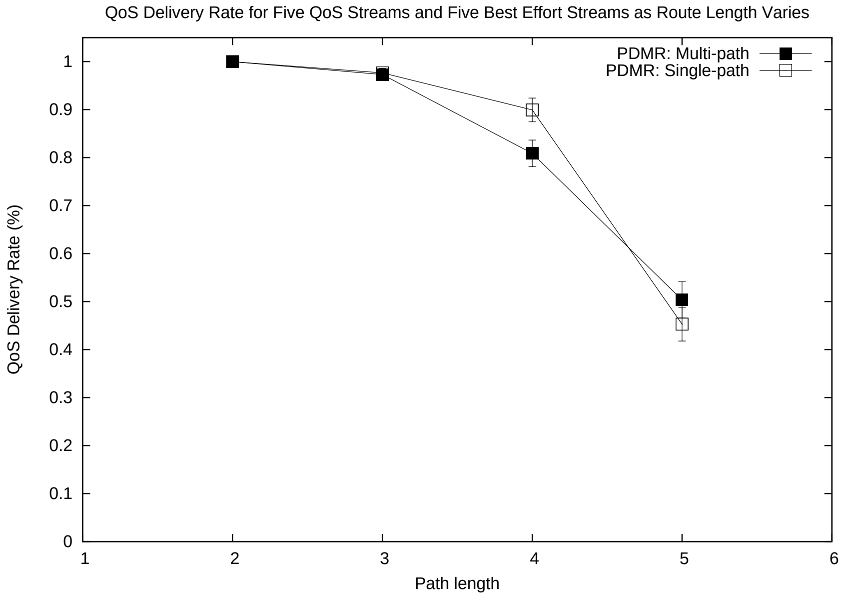

FIG. 5.15. QoS delivery rate for a mobile network with five concurrent QoS traffic sessions, five concurrent best effort traffic sessions, and various fixed route lengths

Figure 5.15 indicates that paths of length four performed best with single path routing rather than multi-path routing, at least under the traffic conditions used to generate these results. The reason for this involves the algorithm used to choose disjoint routes in multi-path PDMR. First, a shortest path is discovered. In single path mode, this path is used. In multi-path mode, the algorithm searches for a second path, disjoint to the first. This path, as previously mentioned, will be *at least* as long as the first path. Using this second path in situations where traffic is not severe enough to warrant multi-path routing only adds complexity to the routing process.

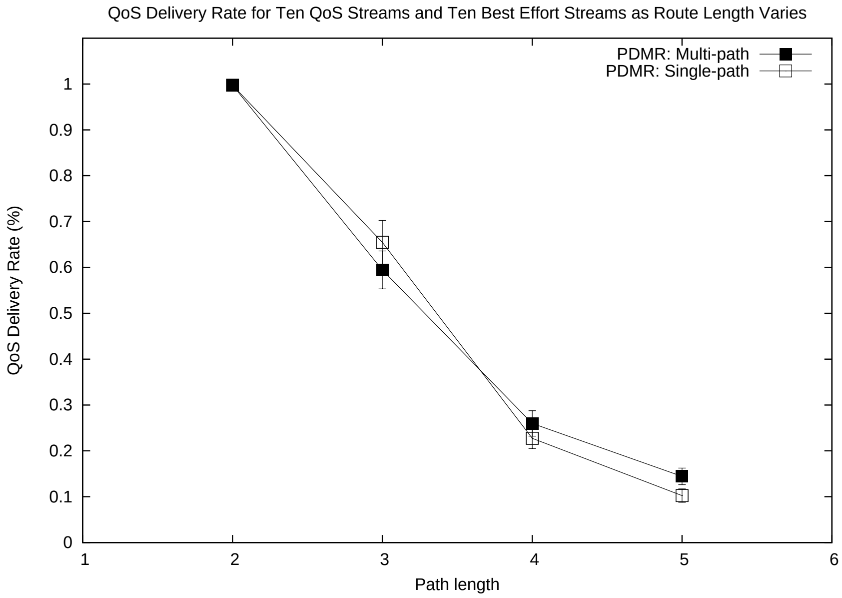

FIG. 5.16. QoS delivery rate for a mobile network with ten concurrent QoS traffic sessions, ten concurrent best effort traffic sessions, and various fixed route lengths

Figure 5.16 indicates that with ten concurrent QoS traffic sessions and ten concurrent best effort traffic sessions, multi-path routing performed slightly better than single path routing when the path length of the route was greater than four. This was likely due to two factors:

1. The difference in first and second path lengths in multi-path routes becomes less significant with longer routes, as the difference is proportionally smaller.
2. Longer routes give more opportunity for higher quality disjoint routes, in terms of the disjoint path hierarchy.

However, Figure 5.16 does not indicate as large a difference between single- and multi-path PDMR as Figure 5.15. The two types of PDMR performed similarly because the high traffic severity had a significantly higher impact on delivery rates than did the benefits of multi-path routing.

5.4 Contrived Network Model

Sections 5.2 and 5.3 include results that imply no sustained difference in the performance of single- and multi-path routing. In this section, we investigate whether a contrived scenario will illustrate the benefits of multi-path routing. Table 5.4 lists the simulation parameters, and Figure 5.17 illustrates the static network topology for this contrived scenario. In Figure 5.17, the dotted lines represent best effort traffic sessions, while the solid lines represent the potential paths for a QoS traffic session. Two best effort sessions are present in each simulation. One session is routed from source S$_{BE1}$ to destination D$_{BE1}$ and the other from source S$_{BE2}$ to destination D$_{BE2}$. A single QoS session is simulated in which QoS traffic is routed from source S$_{QoS}$ to destination D$_{QoS}$.

In single-path mode, PDMR routes QoS traffic through one of the paths indicated by the solid lines. In multi-path mode, PDMR uses both paths indicated by the solid lines. To minimize LEAP traffic interference, we ended the legend traversal at 100 seconds. At 100 seconds all traffic sessions are started and continued until the end of

Input Parameters	
Number of Nodes	6
Simulation Duration	1,250 seconds
Simulation Area Size	175 m x 175 m
Transmission Range	100 m
Simulator	
Simulator	NS-2 (version 2.26)
Medium Access Protocol	IEEE 802.11 (SWAN)
Link Bandwidth	11 Mbps
Number of trials	10
Confidence Interval	95%

Table 5.4. Simulation parameters.

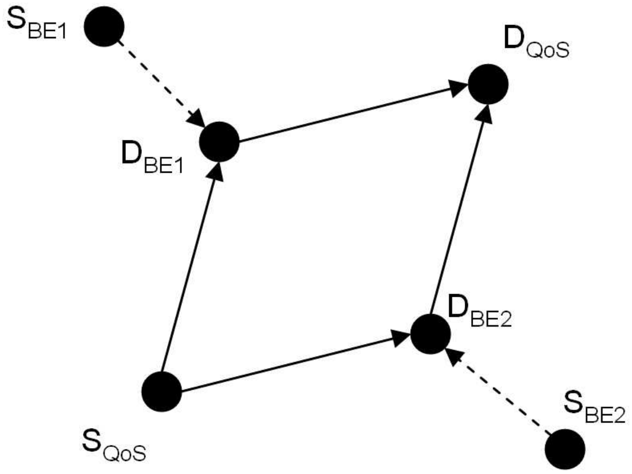

Fig. 5.17. Network topology for the contrived scenario.

the simulation. Each of the two simulated best effort sessions was comprised of 512 byte packet payloads, with a constant bit rate of 20,480 bytes per second (40 packets per second). The QoS traffic session was comprised of 160 byte packet payloads at varying constant bit rates. All packets include 20 bytes of non-PDMR headers and 18 bytes of PDMR header.

Figure 5.18 shows the delivery rates for QoS traffic at various QoS transmission rates. At rates under 7 kB/s, multi-path routing and single-path routing performed nearly identically. In these cases, both routing methods discovered paths that had enough available bandwidth to deliver comparable amounts of QoS traffic. Between 7 kB/s and 12 kB/s multi-path routing performed significantly higher than single-path routing. In these cases, neither single path had enough available bandwidth for the traffic requirements; however, sufficient bandwidth for the traffic requirements was available on the two disjoint paths. At transmission rates above 12 kB/s, the performances of multi-path and single-path routing are again similar. This convergence in delivery rates is due to the decreasing significance of the bandwidth advantage in multi-path routing. The amount of increased available bandwidth provided by multi-path routing does not increase as network usage increases. Therefore, as traffic severity increases, the relative impact of multi-path routing decreases. We note that this contrived network is relatively small, but still illustrates the benefits of multi-path routing. Larger networks should offer more opportunities to obtain benefits from multi-path routing.

Figure 5.19 shows the throughput rates for QoS traffic at various transmission rates. This figure illustrates the same trends as Figure 5.18 due to the correlation between delivery rates and throughput rates. However, Figure 5.19 indicates the linear growth in throughput (i.e., a constant delivery rate) stops at approximately 5 kB/s.

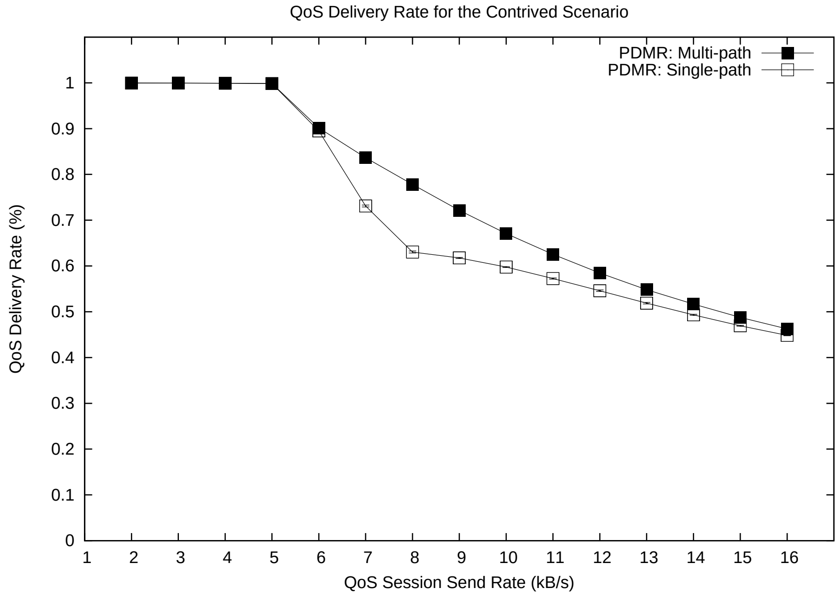

FIG. 5.18. Delivery rates for the contrived scenario.

At this point, traffic conditions begin to approach network saturation, illustrating the point where multi-path routing becomes beneficial. We note that for both Figure 5.18 and Figure 5.19, the confidence intervals calculated for these data points were too small to be visible on the graphs.

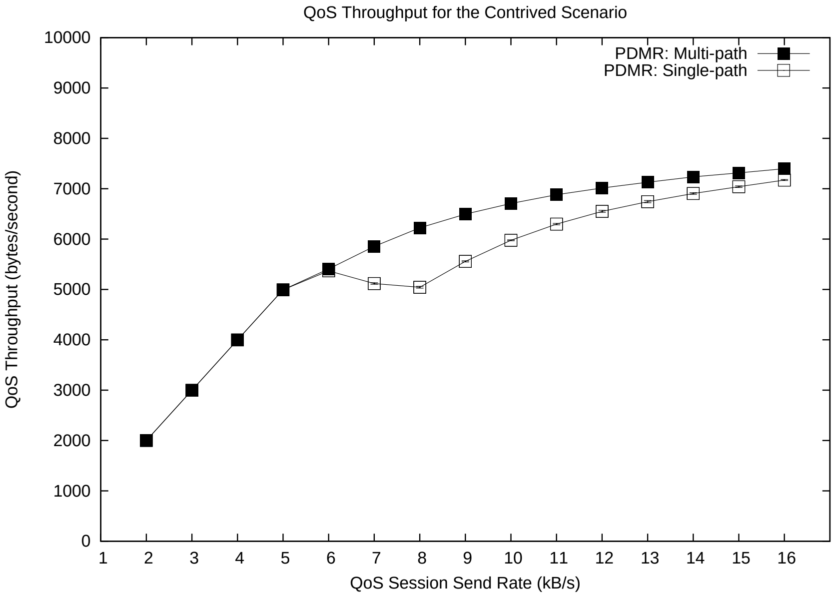

FIG. 5.19. Throughput rates for the contrived scenario.

Chapter 6

CONCLUSIONS AND FUTURE WORK

In this study, we evaluated the Proactive Disjoint Multi-path Routing protocol (PDMR) in both single path and multi-path modes in an effort to determine the value of multi-path routing in terms of QoS delivery rates. PDMR employs the LEAP information dissemination protocol to enable multi-path source routing at every node in the network. We developed an algorithm capable of discovering various types of multi-path routes based on the relative position of the two paths comprising a route in the network. We identified the following theoretical requirements for benefiting delivery rates by using disjoint multi-path routes:

1. The source and destination nodes are not bandwidth constrained.
2. Intermediate nodes on the multi-paths are under load such that a) no single path can satisfy the bandwidth requirement, and b) considered together, their bandwidth capacity is sufficient to satisfy the bandwidth requirement.

In an attempt to create these conditions, we simulated both static and mobile networks with various levels of traffic severity in terms of QoS and best effort traffic. To further classify network conditions, we generated results regarding route type frequencies for various network sizes. We observed a steady increase in the availability of disjoint routes, as network density increased, implying the feasibility of multi-path routing in dense networks. However, the results indicate that there is marginal or no improvement provided by multi-path routing in the scenarios we simulated. Due to the apparent low frequency of the two requirements in the simulated scenarios, we developed a contrived scenario to verify the positive impact of multi-path routing. The results generated from these simulations imply that there is value to multi-path routing in certain traffic scenarios. We provide our plans for future work in Section 6.2. Section 6.1 presents our conclusions.

6.1 Conclusions

In some high traffic scenarios, multi-path routes do offer slight improvements in QoS delivery rates over those observed in single path routing, as shown in Figure 5.8. This is due to the load balancing effect multi-path routing has on a network. Load balancing evenly distributes the load for a single traffic session across two disjoint routes. However, our simulation results illustrate that traffic load has far more impact on delivery rates than on the type of routing. In other words, though multi-path routing is useful in some situations, these situations are those in which low delivery rates are acceptable.

Although we observed an improvement in delivery rates for multi-path routing and certain traffic scenarios, the improvement was not observed in most of our simulations. This implies that although multi-path routing may be slightly beneficial in some situations, it generally offers no benefit in terms of delivery rate.

The results in figures 5.3 and 5.4 illustrate that multi-path routes will, on average, be longer than single path routes. A higher average route length implies a higher average number of retransmissions per packet delivery, which also implies more traffic in the network. As seen in Figure 5.15, in situations where the criteria for multi-path benefits cannot be met, these negative effects of multi-path routing can negatively impact QoS and overall delivery rates.

Although the results from Sections 5.2 and 5.3 do not show any significant performance differences between single-path and multi-path routing, figures 5.18 and 5.19 in Section 5.4 illustrate that multi-path routing will improve delivery and throughput rates in certain conditions. This improvement was not observed in the earlier simulations due to the infrequent occurrence of the traffic conditions required to obtain benefits from multi-path routing. Possible reasons for the scarcity of these conditions are discussed in Section 6.2.

Given that there are positive and negative benefits to multi-path routing, each to a varying degree depending on the amount of traffic in the network, multi-path routing can be, but is not always, a benefit to delivery rates. In most situations multi-path routing should not be used as, generally, the traffic severity necessary to obtain the benefits of multi-path routing is not common. However, results suggest that protocols developed to leverage multi-path routing may be beneficial in some situations.

6.2 Future Work

The lack of significant improvement in delivery rates when employing multi-path routing may have been caused by the scenarios we chose to simulate. Although various types of multi-path routes were available, as shown in figures 5.1 and 5.2, the criteria for an improvement in multi-path routing were not met with a frequency sufficient for significant delivery rate improvement. The following items are our plans for future work in this area:

- *Network Region Size.* Our simulated network produced routes with an average length of approximately 2-3 hops (as shown in Figure 5.3). Thus, the disjoint paths found were relatively close to each other. One area of future work is to increase the simulated network region, which will also increase the average route lengths. This effect should increase the number of disjoint path options, allowing multi-paths to leverage network resources in different parts of the network.

- *Node Density.* Figures 5.1 and 5.2 indicate that 100 nodes in a 300m x 600m network provides sufficient network density for multi-path route discovery. However, it is not clear what effect an increase in node density may have on delivery rates. Although a higher density implies more direct shortest paths, it also implies a higher frequency of disjoint paths. Thus, we plan to investigate whether an increase in density will lead to a high frequency of the valuable strict link disjoint routes.

- *Protocol Parameters.* As mentioned in Section 5.1, PDMR lacks features such as intermediate node route improvements and route failure messages. Although these were assumed to have little or no impact on the relative performance of single- and multi-path routing, we plan to investigate their impact. The additional components of PDMR we plan to evaluate further are:

  - maximum packet retransmissions,
  - packet acknowledgment policy,
  - SWAN admission control parameters, and
  - LEAP traversal parameters.

- *Traffic Scenario.* Although we attempted to simulate traffic at various levels of severity, it is possible that multi-path routing may respond to different best effort or QoS packets sizes, transmission intervals, or traffic session durations. We plan to simulate these other types of traffic scenarios to determine what traffic scenarios gain benefits from multi-path routing.

- *Number of Multi-Paths.* PDMR currently only uses two disjoint paths in a multi-path route. PDMR's performance in delivery rate may improve if a multi-path route included more than two disjoint paths. We plan to extend PDMR to enable the discovery and use of more than two disjoint paths. We also plan to evaluate when more than two disjoint paths is beneficial.

- *Route Discovery Algorithm.* PDMR parameters such as the maximum bandwidth threshold can be adjusted and may change performance. Additionally, a component such as a signal strength filter may be employed. We plan to investigate modifications and potential augmentations of our current PDMR protocol that may improve the performance of multi-path routing.

Finally, as described in Chapter 2, many multi-path routing protocols leverage multi-paths to prevent excessive routing overhead by using the additional paths as backups in case of a route failure. We plan to investigate alternative path use policies as they may be another means to leverage the value of multi-path routing. Furthermore, we plan to investigate the load balancing capabilities of multi-path routing, as load balancing may offer benefits to energy constrained networks such as wireless sensor networks.

REFERENCES

[1] J.A. Stine and G. de Veciana. “A Paradigm for Quality-of-Service in Wireless Ad Hoc Networks Using Synchronous Signaling and Node States,” *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 7, pp. 1301-1321, 2004.

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

[3] S. Lee, G. Ahn, X. Zhang, and A. Campbell. “Evaluation of the INSIGNIA Signaling System,” *Proceedings of Networking*, pp. 311-324, 2000.

[4] G. Ahn, A. Campbell, A. Veres, and L. Sun. “SWAN,” draft-ahn-swan-manet-00.txt, Work in Progress, October 2002.

[5] G. Ahn, A. Campbell, A. Veres, and L. Sun. “Supporting Service Differentiation for Real-Time and Best Effort Traffic in Stateless Wireless Ad Hoc Networks (SWAN),” *IEEE Transactions on Mobile Computing*, vol. 1, no. 3, pp. 192-207, September 2002.

[6] S. Lee and M. Gerla. “Split Multi-path Routing with Maximally Disjoint Paths in Ad hoc Networks,” *Proceedings of the International Conference on Communications (ICC)*, vol. 10, pp. 3201-3205, June 2001.

[7] G. Koh, D. Oh, and H. Woo. “A Graph-Based Approach to Compute Multiple Paths in Mobile Ad Hoc Networks,” Lecture Notes in Computer Science, vol. 2713/2003, pp. 323-331, August 2003.

[8] A. Nasipuri, R. Castaneda, and S. Das. “Performance of Multi-path Routing for On-Demand Protocols in Mobile Ad Hoc Networks,” *ACM/Baltzer Mobile Networks and Applications (MONET)*, vol. 6, pp. 339–349, 2001.

[9] B. Villela and O. Duarte. “Maximum Throughput Analysis in Ad Hoc Networks,” *Lecture Notes in Computer Science - Networking 2004*, vol. 3042, pp. 223-234, May 2004.

[10] J. Chen, P. Druschel, and D. Subramania. “An Efficient Multi-path Forwarding Method,” *Proceedings of IEEE INFOCOM*, pp. 1418-1425, 1998.

[11] I. Cidon, R. Rom, and Y. Shavitt. “Analysis of Multi-Path Routing,” *IEEE/ACM Transactions on Networking*, vol. 8, no. 6, pp. 885-896, 1999.

[12] R. Ogier, V. Rutenburg, and N. Shacham. “Distributed Algorithms for Computing Shortest Pairs of Disjoint Paths,” *IEEE Transactions on Imformation Theory*, vol. 39, no. 2, pp. 443-455, 1993.

[13] D. Sidhu, R. Nair, and S. Abdallah. “Finding Disjoint Paths in Networks,” *Proceedings of ACM SIGCOMM*, pp. 43-51, 1991.

[14] N. Taft-Plotkin, B. Bellur, and R. Ogier. “Quality-of-Service Routing Using Maximally Disjoint Paths,” *Proceedings of the IEEE International Workshop on Quality of Service*, pp. 119-128, 1999.

[15] S. Vutukury and J. Garcia-Luna-Aceves. “An Algorithm for Multi-path Computation Using Distance Vectors with Predecessor Information,” *Proceedings of IEEE International Conference on Computer Communications and Networks (IC3N)*, pp. 534-539, 1999.

[16] W. Zauman and J. Luna-Garcia-Aceves. “Loop-Free Multi-path Routing Using Generalized Diffusing Computations,” *Proceedings of IEEE INFOCOM*, pp. 1408-1417, 1998.

[17] P. Papadimitratos, Z. Haas, and E. Sirer. “Path Set Selection in Mobile Ad Hoc Networks.” *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc)*, pp. 1-11, 2002.

[18] D. Johnson and D. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks,” In Mobile Computing, edited by Tomasz Imielinski and Hank Korth, Chapter 5. Kluwer Academic Publishers, pp. 153-181, 1996.

[19] M. Mirhakkak, N. Schultz, and D. Thomson. “Dynamic Quality-of-Service for Mobile Ad Hoc Networks,” *Proceedings of the 1st ACM International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc)*, pp. 137-138, 2000.

[20] N. Bauer, M. Colagrosso, and T. Camp, “An Agile Approach to Distributed Information Dissemination in Mobile Ad Hoc Networks”, *Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2005.

[21] W. Navidi, T. Camp, and N. Bauer. “Improving the Accuracy of Random Waypoint Simulations Through Steady-State Initialization,” *In Proceedings of the 15th International Conference on Modeling and Simulation (MS)*, pp. 319-326, 2004.

[22] C. Perkins. “Ad Hoc On Demand Distance Vector (AODV) routing,” IETF RFC 3561, November 1997.

[23] S. Corson and J. Macker. “Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations,” IETF RFC 2501, January 1999.

[24] M. Marina and S. Das. “On-Demand Multi-path Distance Vector Routing for Ad Hoc Networks,” *In Proceedings of the International Conference for Network Protocols (ICNP)*, pp.14-23, November 2001.

[25] The VINT Project. The network simulator - ns-2. http://www.isi.edu/nsnam/ns/. Page accessed on January 1st, 2005.