MULTIPATH “FRESNEL ZONE” ROUTING FOR WIRELESS AD HOC NETWORKS

Yibin Liang

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE
in
Electrical Engineering

Dr. Scott F. Midkiff, Chair
Dr. William H. Tranter
Dr. Luiz A. DaSilva

March 4, 2004
Blacksburg, Virginia

Keywords: wireless ad hoc networks, MANET, routing protocol, multipath routing

MULTIPATH “FRESNEL ZONE” ROUTING FOR WIRELESS AD HOC NETWORKS

Yibin Liang

(ABSTRACT)

Prior research in routing for wireless ad hoc networks has shown that multipath routing can enhance data delivery reliability and provide load balancing. Nevertheless, only a few multipath routing algorithms have been proposed and their interaction with transport layer protocols has not been thoroughly addressed in the literature.

In this work, we propose the multipath “Fresnel zone” routing (FZR) algorithm for wireless ad hoc networks. FZR constructs multiple parallel paths from source to destination based on the concept of “Fresnel zones” in a wireless network. The zone construction method assigns intermediate routers into different “Fresnel zones” according to their capacity and efficiency in forwarding traffic. The central idea in FZR is to disperse traffic to different zones according to network load and congestion conditions, thus achieving better throughput and avoiding congestion at intermediate routers. FZR differs from most existing multipath routing approaches in that both source and intermediate nodes use multiple forwarding paths. FZR also adopts a combination of proactive and on-demand (reactive) approaches to reduce control overhead and latency for packet delivery.

Simulation experiments have shown that FZR outperforms unipath distance vector routing, multipath distance vector (MDV) routing, and split multipath routing (SMR) algorithms in quasi-static wireless ad hoc networks. In our simulations, FZR achieves up to 100 percent higher average throughput using the User Datagram Protocol (UDP) and 50 percent higher average throughput using the Transmission Control Protocol (TCP). FZR can also provide better load balancing among different paths, improve network resource utilization, and enable fairer resource allocation among different data transmission sessions. Future work is needed to evaluate FZR in mobile scenarios.

To my mother and the memory of my father

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank Dr. Midkiff for accepting me as one of his M.S. students. He is always ready and patient to help me with questions and problems in research work. His invaluable advice and guidance have helped me develop this thesis and added to my experience as a graduate student.

I would also like to thank my other committee members, Dr. Luiz DaSilva and Dr. William Tranter for their time and cooperation in reviewing this thesis.

I thank all other colleagues in the Laboratory for Advanced Networking for their kindness and helps. I am grateful to all my friends who have helped me during my graduate career at Virginia Tech.

I wish to express my gratitude to the Intel Corporation for their financial support during my M.S. program through a “Wireless Curriculum Development and Instructional Laboratory” grant.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Overview and Motivation

Wireless communication has become one of the most vibrant areas of technology innovation. Cellular wireless networks have experienced dramatic global growth for the past decade. Recently, wireless local area networks are being rapidly deployed in industrial, commercial, and home networks. Several organizations are actively developing standards for future wireless networks. One important reason for their growing popularity is that wireless networks, to some extent, enable people to exchange information on the move anytime and anywhere in the world. As wireless devices become more inexpensive and widely available, communication networks will become more ubiquitous and far reaching in daily life.

To achieve truly ubiquitous communication, infrastructure-less networks, also known as ad hoc networks, come into play in addition to the widely deployed infrastructure-based networks. In cellular networks, users are connected via base stations and backbone networks. Although users can hand-off between base stations and roam among different networks, their mobility is limited within the coverage areas of the base stations. In ad hoc networks, communicating devices (nodes) can form arbitrary networks "on the fly" to exchange information without the need of pre-existing network infrastructure. Ad hoc networks, thus, can extend communication beyond the limit of infrastructure-based networks.

A fundamental problem in ad hoc networking is how to deliver data packets among nodes efficiently without predetermined topology or centralized control, which is the main objective of ad hoc routing protocols. Because of the dynamic nature of the network, ad hoc routing faces many unique problems not present in wired networks. In addition, ad hoc networking is inherently a multi-layer problem. For example, to vertically optimize protocol layers, ad hoc routing is often jointly considered with power control in the physical layer, multiple access control in the link layer, and quality of service (QoS) support for applications. Therefore, issues in multiple layers should be addressed with a cross-layer approach [67].

Most existing ad hoc routing protocols build and utilize only one single route for each pair of source and destination nodes. Due to node mobility, node failures, and the dynamic characteristics of the radio channel, links in a route may become temporarily unavailable, making the route invalid. The overhead of finding alternative routes may be high and extra delay in packet delivery may be introduced. Multipath routing addresses this problem by providing more than one route to a destination node. Source and intermediate nodes can use these routes as primary and backup routes. Alternatively, they can distribute traffic among multiple routes to enhance transmission reliability, provide load balancing, and secure data transmission.

Multipath routing appears to be a promising technique for ad hoc routing protocols. Prior work and some preliminary simulation results have demonstrated the potential performance improvement using multipath routing. However, most of the proposed multipath schemes still lack robust implementations and thorough testing. Furthermore, the interaction between multipath routing and end-to-end throughput has not been thoroughly studied in past work. More research is needed to address remaining issues in multipath routing for ad hoc networks.

1.2 Research Objectives

The objectives of this research are two-fold. The first objective is to gain insight into and experience with ad hoc networking with an emphasis on ad hoc routing. Research in ad hoc networking has been ongoing for decades, but there are still many open problems to be solved. A lot of research has focused on ad hoc routing, a fundamental problem that distinguishes ad hoc networking from wired networking. As an important aspect of this research, a comprehensive survey of multipath ad hoc routing is provided in Chapter 2.

The second goal is to develop an efficient multipath routing algorithm for ad hoc networks. Due to time constrains, it is not feasible to investigate all aspects of an ad hoc routing protocol or to design a complete protocol. The emphasis in this research is on improving the efficiency of data transmission and the throughput of an ad hoc routing protocol. This work focuses on the throughput of multipath routing in quasi-static wireless ad hoc networks and leaves potential extensions to mobile scenarios to future work.

In this research, we propose the multipath “Fresnel zone” routing (FZR) algorithm for wireless ad hoc networks. This data transmission mechanism utilizes multiple parallel paths from source to destination nodes to improve utilization and throughput of a network. FZR differs from most existing multipath extensions in that both source and intermediate nodes build and utilize multiple forwarding routes. The correct operation of the FZR protocol and its performance are evaluated through simulation. For comparison, some of the existing single shortest-path and multipath routing algorithms are also simulated.

1.3 Thesis Organization

This chapter provides an introduction to the research topic under investigation, namely multipath routing for ad hoc networks. The objectives and the scope of research efforts are also defined.

Chapter 2 provides background information and describes related research efforts. It begins with a brief review of ad hoc networks and some current research issues, followed by a discussion of ad hoc routing and reviews of several proactive and reactive routing protocols. This chapter also presents a comprehensive survey of existing ad hoc multipath routing.

Chapter 3 defines the problem under investigation and explains the research methodology. It also includes some discussion of the multipath routing approach used in this research.

The proposed "Fresnel zone" routing algorithm is the subject of Chapter 4. The first section describes the idea behind FZR and explains the concept of "Fresnel zones" in ad hoc networks. The rest of the chapter focuses on the operation of FZR.

Chapter 5 presents the simulation experiment and summarizes the simulation results. This chapter describes the simulation tool and the models used for the evaluation of the FZR protocol. Simulation parameters and factors are defined and assigned values.

Chapter 6 concludes the thesis with a summary of the research, including the significance of research results. Finally, the chapter offers recommendations for future work within this research area.

CHAPTER II

BACKGROUND AND PRIOR WORK

This chapter provides background and describes related research efforts. It begins with an introduction of wireless ad hoc networks. Section 2.2 gives a brief overview of ad hoc routing and explains several important concepts, including proactive versus reactive routing approaches and hierarchical routing. The next section describes some of the existing ad hoc routing protocols. Finally, Section 2.4 provides a literature review of multipath routing in wireless ad hoc networks.

2.1 Wireless Ad Hoc Networks

A wireless ad hoc network consists of a collection of geographically distributed nodes that communicate over a wireless medium, but have no fixed infrastructure available and have no predetermined network topology [67]. There is no central control in an ad hoc network and individual nodes are responsible for dynamically detecting the presence of other communication peers. The radio transmission range of a node is often limited; therefore, nodes are required to relay packets on behalf of others to facilitate communication across the network.

Research in ad hoc networking has been ongoing for decades. The history of wireless ad hoc networks can be traced back to the Defense Advanced Research Projects Agency (DARPA) packet radio network (PRNet), which evolved into the survivable adaptive radio networks (SURAN) program [21]. Ad hoc networks have played an important role in military applications and related research efforts, for example, the global mobile information systems (GloMo) program [44] and the near-term digital radio (NTDR) program [69]. Recent years have seen a new spate of industrial and commercial applications for wireless ad hoc networks, as viable communications equipment and portable computers become more compact and available.

Ad hoc networks have numerous potential applications. For example, the IEEE 802.11 wireless LAN standard family, as well as the European Telecommunications Standards Institute (ETSI) HIPERLAN standard, supports an ad hoc mode of operation for building simple ad hoc networks. Using multi-hop mesh-based ad hoc networks for wireless broadband access seems to be a promising alternative, as embraced by many startup companies like Rooftop Communications (now part of Nokia), Mesh Networks, and Radiant Networks. Some other potential applications include mobile conferencing, home networking, emergency and disaster relief systems, and personal area networks.

Wireless ad hoc networks can be broadly divided into two categories: quasi-static and mobile. In a quasi-static ad hoc network, nodes are static or portable. However, due to power controls and link failures, the resulting network topology may be dynamic. A typical sensor network is an example of a quasi-static ad hoc network. In mobile ad hoc networks (MANET), the entire network may be mobile, and nodes may move quickly relative to each other. A major technical challenge in a MANET is the design of efficient routing protocols to cope with the rapid topology changes.

Despite the long history of ad hoc networking, there are still quite a number of open problems. These challenges include power control, medium access control (MAC), quality of service (QoS) support, wireless security, and cross-layer optimization issues.

Ad hoc networking is inherently a multi-layer problem. The physical layer must adapt to the dynamic link characteristics and may need to support collision detection for the MAC layer. The MAC layer needs to minimize collisions, allow fair access in the presence of hidden or exposed terminals. The network layer needs to distribute topology control messages in an efficient manner and calculate up-to-date routes in the presence of rapid topology changes.

Since nodes in an ad hoc network are untethered to any infrastructure, individual nodes may have to rely on portable, limited power sources [67]. The idea of energy-efficiency, therefore, becomes important. An intuitive approach for saving energy is to reduce the power used by the radio transceiver, for example, to selectively send the receiver into a sleep mode or to use a transmitter with variable output power and select routes that require many short hops instead of a few long hops [72].

A similar multi-layer issue is that of security in ad hoc networks [90]. Due to the broadcast nature of radio communication, wireless networks are susceptible to eavesdropping, malicious jamming and interference, which a well-designed physical layer should be able to avoid. Because usually there are no central control and no trusted authorities in an ad hoc network, secure key distribution and management for data encryption and authentication become another problem.

2.2 Routing in Ad Hoc Networks

In wireless ad hoc networks, the communication range of a node is often limited and not all nodes can directly communicate with one another. Nodes are required to relay packets on behalf of other nodes to facilitate communication across the network. Since there is no pre-determined topology or configuration of fixed routes, an ad hoc routing protocol is used to dynamically discover and maintain up-to-date routes between communicating nodes.

Each node in an ad hoc network that is willing to relay packets resembles an intermediate router in the Internet. This observation motives attempts to adopt existing routing protocols for use in ad hoc networks [62]. Many Internet routing protocols are based on either link-state (LS) or distance vector (DV) algorithms, but most of them do not address the unique characteristics of wireless ad hoc networks. Numerous optimization techniques for existing LS or DV routing protocols as well as new routing concepts have been proposed during the past decades. Within the Internet Engineering Task Force (IETF), the mobile ad hoc networking (MANET) working group seeks to standardize routing protocols for ad hoc networks.

2.2.1 Proactive versus Reactive Approaches

Ad hoc routing protocols may generally be categorized as being either proactive or on-demand (reactive) according to their routing strategy [68]. Proactive protocols attempt to maintain a correct view of the network topology all the time and build routes from each node to every other node before they are needed. These protocols require each node to maintain one or more tables to store routing information, hence they are also called table-driven protocols. Any changes in topology are propagated through the network, so that all nodes know of the changes in topology. Examples include destination-sequenced distance-vector (DSDV) routing [64], optimized link state routing (OLSR) [11], fisheye state routing (FSR) [59], clusterhead gateway switch routing (CGSR), landmark ad hoc routing (LANMAR) [60], and topology dissemination based on reverse-path forwarding (TBRPF) [3, 55].

On-demand protocols, as the name suggests, only attempt to build routes when desired by the source node. In other words, network topology is detected as needed (on-demand). When a node wants to send packets to some destination but has no routes to the destination, it initiates a route discovery process within the network. Once a route is established, it is maintained by a route maintenance procedure until the destination becomes inaccessible or until the route is no longer desired. Examples include ad hoc on-demand distance vector routing (AODV) [61], associativity-based routing (ABR) [75], dynamic source routing (DSR) [33], lightweight mobile routing (LMR) [13], and the temporally ordered routing algorithm (TORA) [57].

Proactive protocols have the advantage that communication with arbitrary destination experiences minimal delay, but suffer the disadvantage of additional control overhead to update routing information at all nodes [62]. A common goal of proactive protocols is to reduce routing control overhead. On-demand protocols, however, try to maintain low control overhead by maintaining only those routes actually desired, but suffer initial latency during the route discovery phase. The on-demand route request broadcast may generate excessive traffic when the frequency of route discovery is high.

There also exist hybrid routing protocols that adopt both a proactive and an on-demand approach. For example, the zone routing protocol (ZRP) [25] uses a proactive approach to maintain routing information for nodes within a predefined zone and an on-demand approach to discover routes for nodes in different zones.

2.2.2 Clustering and Hierarchical Routing

Scalability is one of the important problems in ad hoc networking. Scalability in ad hoc networks can be broadly defined as the network's ability to provide an acceptable level of service to packets even in the presence of a large number of nodes in the network [67]. In proactive routing protocols, when the number of nodes in the network increase, the number of topology control messages increases non-linearly and they may consume a large portion of the available bandwidth. In reactive routing protocols, large numbers of route requests to the entire network may eventually become packet broadcast storms. Typically, when the network size increases beyond certain thresholds, the computation and storage requirements become infeasible. When mobility is considered, the frequency of routing information updates may be significantly increased, thus worsening the scalability issues.

One way to address these problems and to produce scalable and efficient solutions is hierarchical routing. Wireless hierarchical routing is based on the idea of organizing nodes in groups and then assigning nodes different functionalities inside and outside a group. Both the routing table size and update packet size are reduced by including in them only part of the network. For reactive protocols, limiting the scope of route request broadcasts also helps to enhance efficiency. The most popular way of building hierarchy is to group nodes geographically close to each other into clusters. Each cluster has a leading node (clusterhead) to communicate to other nodes on behalf of the cluster.

Example of hierarchical ad hoc routing protocols include clusterhead-gateway switch routing (CGSR) [8], hierarchical state routing (HSR) [58], zone routing protocol (ZRP) [25], and landmark ad hoc routing protocol (LANMAR) [23, 60].

2.2.3 Performance Evaluation

Corson and Macker, in RFC 2501, suggest several qualitative and quantitative criteria for the evaluation of mobile ad hoc routing protocols [12]. Some of the desirable qualitative properties of ad hoc routing protocols include distributed operation, loop-freedom, demand-based operation, proactive operation, security, sleep mode, and unidirectional link support. The quantitative metrics defined in RFC 2501 include the following:

- End-to-end data throughput and delay — external statistical measure of routing effectiveness.
- Efficiency — internal measure of routing effectiveness, including average number of data bits transmitted per data bit delivered, average number of control bits transmitted per data delivered, average number of control and data packets transmitted per data packet delivered.

- Route acquisition time — the time required to establish route(s) when requested, which is of particular concern with on-demand routing algorithms.
- Percentage of out-of-order delivery — important for transport layer protocols and application requiring in-order delivery.

For measuring a protocol's performance, the networking context should be considered. Some of the parameters considered in RFC 2501 include network size, network connectivity, topological rate of change, link capacity, fraction of unidirectional links, traffic patterns, mobility, and fraction and frequency of sleeping nodes.

2.3 Review of Ad Hoc Routing Protocols

This section presents brief descriptions for some of the existing ad hoc routing protocols. The purpose is to provide background information for further discussion. Comprehensive reviews and performance comparisons of ad hoc routing protocol have been presented in many publications [4, 5, 15, 28, 29, 37, 41, 62, 68].

2.3.1 Optimized Link State Routing

Optimized link state routing (OLSR) is a proactive routing protocol based on the link state algorithm [11]. The protocol uses multipoint relays (MPRs) [66] to reduce the number of broadcast packet transmissions and, also, the size of the link state update packets, leading to efficient flooding of control messages.

In OLSR, each node periodically broadcasts HELLO messages to all its neighbors. HELLO messages are only exchanged between immediate neighbors. A HELLO message contains the source node's own address and the list of neighbors known to that node. Upon receiving a HELLO message, a node can gather information describing its adjacent neighborhood and its two-hop neighborhood. Each node selects a set of one-hop neighbors that forward broadcasts of link state packets. This set of selected neighbor nodes is called the MPRs of that node. Other neighbor nodes that are not in the MPR set receive and process link state broadcasts from that node, but do not retransmit them further. The MPR set is selected in such a way that the set covers all nodes that are two hops away. It is desirable that the MPR set be minimal. Computing the optimal (minimal size) MPR set is an NP-complete problem [66]. Efficient heuristics are used in OLSR for the selection of MPRs.

The multipoint relay selector (MPRS) set for a node is the set of neighbor nodes that select this node as their MPR nodes. All nodes with a non-empty MPRS set periodically generate link state updates called topology control (TC) messages. These TC messages are diffused to other nodes in the network via MPR nodes. A TC message contains the addresses of the source node and its MPRS set nodes. When a node receives the MPRS set information, the node uses it to create a partial topology graph of the network, made up of all reachable nodes in the network and the set of links between a node and its MPR selectors. Using this partial topology graph, it is possible to compute optimal shortest paths from a node to any reachable destination in the network [66].

OLSR improves over a standard link state algorithm in two aspects. First, the link state update is reduced in size since it includes only links for a node's MRPS set. Secondly, only MPR nodes forward link state updates, thus minimizing flooding of control messages. OLSR is particularly suited to large and dense networks. When the network is sparse, every neighbor of a node becomes a multipoint relay. The OLSR then reduces to a pure link state protocol. Performance evaluation of OLSR through experiments and simulation was presented in [10] and [38].

2.3.2 Dynamic Source Routing

Dynamic source routing (DSR) [32, 33] is an on-demand routing protocol for wireless ad hoc networks. DSR is based on the concept of source routing, in which a source node indicates the sequence of intermediate routes in a data packet's header. Like other on-demand routing protocols, operation of DSR can be divided into two procedures: route discovery and route maintenance.

Each node in the network maintains a route cache for the source routes that it has learned. When a node needs to send a packet to some destination, it first checks its route cache to determine whether it already has an up-to-date route to the destination. If no route is found, the node initiates the route discovery procedure by broadcasting a route request message to neighbor nodes. This route request message contains the address of the source and destination nodes, a unique identification number generated by the source node, and a route record to keep track of the sequence of hops taken by the route request message as it is propagated through the network. Upon receiving a route discovery request, an intermediate node checks whether its own address is already listed in the route record of the route request message. If the node has not seen or processed this route request, it appends its address to the route record and forwards the route request to its neighbors.

When the destination node receives the route request, it appends its address to the route record and returns it to the source node within a new route reply message. If the destination already has a route to the source, it can use that route to send the reply. Otherwise, it can use the route in the route request message to send the reply. The first case is beneficial in situations where a network might be using unidirectional links and it might not be possible to send the reply using the same route taken by the route request message. If symmetric links are not supported, the destination node may initiate its own route discovery message to the source node and piggyback the route reply on the new route request message.

Route maintenance is accomplished through the use of route error messages and acknowledgement messages. If a node that is part of some route detects a link failure when forwarding data packets, it creates a route error message and sends it to the source of the data packets. The route error message contains the address of the node that generates the error and the next hop that is unreachable. When source node receives the route error message, it removes all routes from its route cache that have the address of the node in error. It may initiate a route discovery for a new route if needed. In addition to route error message, acknowledgements are used to verify the correct operation of links. Such acknowledgements include passive acknowledgements, where a node is able to hear the next hop forwarding the packet along the route.

To reduce the route search overhead, an important optimization is allowing an intermediate node to send a route reply to the source node if it already has an up-to-date route to the destination. Promiscuous listening used by DSR helps nodes to learn route updates without actually participating in a route discovery process.

2.3.3 Ad Hoc On-Demand Distance Vector Routing

Ad hoc on-demand distance vector (AODV) routing [61, 63] adopts both a modified on-demand broadcast route discovery approach used in DSR and the concept of destination sequence number adopted from destination-sequenced distance-vector routing (DSDV) [64].

When a source node wants to send a packet to some destination and does not already have a valid route to that destination, it initiates a path discovery process and broadcasts a route request (RREQ) message to its neighbors. The neighbors in turn forward the request to their neighbors until the RREQ message reaches the destination or an intermediate node that has an up-to-date route to the destination. In AODV, each node maintains its own sequence number, as well as a broadcast ID. Each RREQ message contains the sequence numbers of the source and destination nodes and is uniquely identified by the source node's address and a broadcast ID. AODV utilizes destination sequence numbers to ensure loop-free routing and use of up-to-date route information. Intermediate nodes can reply to the RREQ message only if they have a route to the destination whose destination sequence number is greater or equal to that contained in the RREQ message.

During the process of forwarding the RREQ messages, an intermediate node automatically records the address of the neighbor from which it received the first copy of the RREQ message, thereby establishing a reverse path. Additional copies of the same RREQ message are discarded. Once the RREQ message reaches the destination or an intermediate node with a fresh route, the destination or the intermediate node responds by sending a route reply (RREP) packet back to the neighbor from which it first received the RREQ message. As the RREP message is routed back along the reverse path, nodes along this path set up forward path entries in their routing tables. Intermediate nodes forward the first RREP message for a given source node towards the destination and subsequent RREP messages that contain either a greater destination sequence number than the previous RREP or the same destination sequence number with a smaller hop count.

A route maintenance procedure is invoked when a node detects a link failure or a change in neighborhood. If a source node moves, it can reinitiate the route discovery procedure to find a new route to the destination. If a node along the route moves, its upstream neighbors notify the move by sending a link failure notification message to each of its active upstream neighbors. These nodes in turn forward the link failure messages to their upstream neighbors until the source node is reached.

2.4 Multipath Ad Hoc Routing

The term multipath routing has been used in the literature to describe routing approaches that establish and utilize multiple paths for each pair of source and destination nodes. Multipath routing for high-data-rate and heterogeneous networks has been studied extensively in the literature. Previous work has focused on using multiple paths to achieve load balancing and provide fault tolerance. Proposed multipath routing schemes for load balancing include selecting alternate routes to avoid congestion in primary routes in connection-oriented high-data-rate networks [2, 20, 27] and distributing traffic load to multiple routes in connection-less networks [51, 70, 74]. A synthesis method for fault-tolerant multipath routing was discussed in [26]. The diffusing algorithm for shortest multipath (DASM) [88] and the multipath distance vector algorithm (MDVA) [79] appear to be the first multipath distance vector routing algorithms. A modified link-state algorithm to support multiple paths is described in [80]. An on-demand multipath routing scheme was proposed in [71] to support QoS requirements. In [6], an efficient packet forwarding mechanism for multipath routing is discussed. Kaur *et al.* put forward an evolutionary framework to adopt multipath routing in the Internet [34].

In addition to providing advantages for some traditional wired networks, multipath routing seems to be a promising technique to improve the resilience of wireless ad hoc networks. Due to the effect of unreliable wireless links and frequent topology changes, communicating pairs of nodes often experience route disruptions. Repairing broken routes may incur heavy penalties for sending control traffic through the network and extra delay in packet delivery. An intuitive solution to this problem is to provide redundant routes in case a primary one fails. Few existing ad hoc routing protocols support such a feature. For example, the TORA [57] routing protocol provides loop-free multiple alternate paths by maintaining a destination-oriented directed acyclic graph (DAG) for each node in the network. However, TORA selects these alternate paths arbitrarily and does not provide any mechanism to evaluate their quality. Also, TORA may not perform well in comparison to other on-demand protocols according to some simulation studies [5, 14].

2.4.1 Summary of Existing Multipath Routing Extensions

Several different multipath routing mechanisms have been studied in prior work. The multipath extension to DSR (MDSR) presented in [52, 53] maintains alternate disjoint routes to the destination, which are used when the primary one fails. It is demonstrated by simulations that the multipath extension scheme can reduce the frequency of route request query floods. In AODV-BR [40], an extension of AODV, backup routes are established with a modified route reply procedure. Ad hoc on-demand multipath distance vector routing (AOMDV) [49, 50] computes multiple loop-free and link-disjoint paths during the route discovery process. Multipath source routing (MSR) [81, 82] proposes a weighted round-robin heuristic-based scheduling strategy to distribute load among multiple paths. Split multipath routing (SMR) [42] relies on a modified route request procedure to discover maximally disjoint paths. However, the load is distributed across only two routes per session. Another multipath extension to DSR, proposed in [84], uses node coloring techniques to find two disjoint paths during the query phase of the route discovery process.

While most multipath extensions to DSR and AODV prefer disjoint paths, new path selection criteria have recently been adopted in several multipath schemes. The neighbor-table-based multipath routing (NTBMR) scheme builds non-disjoint paths by maintaining a two-hop neighbor table and a route cache in every node [87, 86]. The authors of NTBMR introduced an analysis model to compute the reliability of non-disjoint and disjoint multipath routing and argued that the non-disjoint paths perform better and provide more redundancy. A similar path selection criterion is adopted in the redundancy based multipath routing (RBMR) [35], which aims to establish a route that contains more redundant paths towards the destination. In [22], braided multipath routing is introduced and compared with disjoint multipath routing in wireless sensor networks. Meshed multipath routing (M-MPR) [17] uses meshed paths and selective forwarding on all intermediate nodes to achieve better load distribution in sensor networks. Several optimal and heuristic algorithms to compute minimum energy node-disjoint and link-disjoint paths are developed in [73]. There also exist location-based routing algorithms that support multiple paths [47, 54].

2.4.2 Path Selection Criteria

The problem of selecting an optimal set of paths from all available paths for certain optimization objectives under various constraints is nontrivial and often debated. Path reliability, power consumption, and throughput are some common optimization objectives for using multipath routing.

The disjointness between paths is one criterion often used in selecting paths. Paths are *node-disjoint* if they share no common nodes except the source and destination nodes. Similarly, paths are *link-disjoint* if they share no common links. Node-disjoint paths are also link-disjoint. Braided paths relax the requirement for node disjointness, which means that alternate paths in a braid are partially overlaid with the primary path, i.e., they are not completely disjoint. As illustrated in Figure 1, A-B-C-D-E-I and A-F-G-H-I are node-disjoint paths, and A-F-G-H-I and A-F-J-K-L-I are braided paths. In [56], the disjoint path set selection protocol (DPSP) was proposed for selecting disjoint paths. Although many multipath routing schemes prefer disjoint paths, others adopt non-disjoint paths. The route discovery procedure proposed in [17] builds a mesh of forward links. Meshed multiple paths are also adopted in [87, 35].

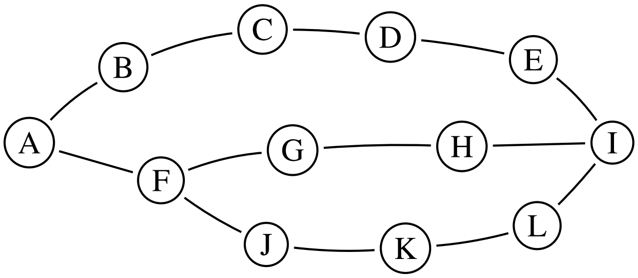

Figure 1: Disjoint and braided multiple paths

2.4.3 Path Discovery

In proactive routing protocols, each node has a complete view of the network topology and all possible paths to a destination can be constructed using a modified link state or distance vector algorithm (e.g., see [79, 80]). On-demand protocols, however, avoid periodic broadcast of full topology information to reduce routing overhead. When a route to a destination is required, a route request is flooded to the network until an up-to-date route to the destination is found. To avoid unnecessary route request broadcast, intermediate nodes usually drop duplicate route requests. Another important optimization for on-demand protocols is the use of route caches on intermediate nodes. However, such optimization techniques reduce the chance of discovering multiple paths.

A modified procedure to discover multiple routes was first proposed in [52]. In this multipath extension scheme to DSR, the destination node caches subsequent route queries from a source, but replies to only a selected set of them. The path taken by the first route query to the destination is regarded as the primary path. This primary path is used to determine which subsequent queries have link-disjoint paths from the primary route. Only disjoint paths are chosen, as backup routes are desired in this case. All alternate paths are stored on the source node and, optionally, on the intermediate nodes on the primary path. This scheme does not modify the route request forwarding procedure on intermediate nodes, nor does it offer any analysis model or optimization for discovering multiple paths. The discovery of multiple disjoint paths is not guaranteed even if they exist.

Follow-up work in [42] proposed another modification for DSR to find maximally disjoint paths. Instead of dropping every duplicate route request, intermediate nodes forward duplicate route requests that arrive from different incoming links and have equal or smaller hop-counts than the first received route request. In this way, the destination has more disjoint paths to select from, but the overhead in the route request process is substantially increased. In [84], another heuristic modification of the route request procedure in DSR is used to find two disjoint paths. It marks the nodes in a path as either white or black. The source node sends two route request packets marked with different colors to request two disjoint paths. The destination replies to the first black and the first white route requests. The overhead in the route discovery process is also substantially increased in this case.

In AODV, each route advertisement arriving at a node may potentially define an alternate path to the source or the destination. The difficulty in selecting multiple next-hop routes on intermediate nodes is that they may lead to routing loops. AOMDV [49] uses “advertised hop count” to avoid routing loops. In AOMDV, each node maintains multiple next-hop routes and their hop counts to the destination. A node advertises all available next-hop routes and the maximum hop count of these routes to other nodes. It is proved that the protocol can guarantee loop-freed routes by only accepting alternate routes with lower hop counts. Other heuristics are used in AOMDV to select disjoint paths from alternate paths.

In summary, how to effectively discover multiple paths with desired properties and, at the same time, limit the overhead of route request broadcast is an important trade-off in designing on-demand multipath routing algorithms. Most existing on-demand multipath discovery mechanisms are based on modifications of the route request and reply procedures. Their operations are usually verified by simulations, but more analysis and performance results are needed to evaluate their effectiveness.

2.4.4 Applications

The primary use of multipath routing is to provide backup routes on the source and intermediate nodes. Nasipuri and Castañeda [53] show that multipath routing can increase the lifetime of routes and reduce the frequency of route queries for on-demand routing protocols. By reducing the chance of route disruption, multipath routing effectively increases the packet delivery ratio.

Multipath routing can also provide load balancing and improve link utilization in ad hoc networks. The MDSR extension distributes load among multiple paths based on the measurement of round-trip time. Simulation results show that packet delivery ratio, end-to-end delay, and the throughput of TCP and UDP are significantly improved [52, 53]. Similar results are also reported by Lee and Gerla [42].

A novel framework for multipath routing was proposed in [76, 77] to use multiple paths simultaneously by splitting the information among multiple paths. Each data packet is split into multiple fragments and distributed over multiple paths. Even if some of the fragments are lost, the data packet can still be recovered at the destination by using a diversity coding scheme to protect the data packet. The same concept is also adopted in [19] for a multipath routing scheme for wireless sensor networks.

Besides fault tolerance and load balancing, multipath routing can also be used in many other applications, such as QoS support for real-time wireless applications [16, 89] and multipath transmission for image and video information [7, 43]. Secure data transmission schemes were proposed in [9, 48, 39], where data packets are disseminated using multiple different paths. Each intermediate router only relays portion of the data of a transmission session, thus offering a certain degree of security against eavesdropping.

2.4.5 Performance Issues

Performance studies of multipath routing have been presented in [65, 85]. However, most of the proposed schemes still lack robust implementations and thorough testing. A number of issues remain to be resolved, including improvements for path selection criteria and path discovery procedures. A performance study of multipath schemes in [46] shows that out-of-order packets may adversely affect TCP performance. The effect of multipath routing on lower and upper layers is not yet well understood.

2.5 Summary

Ad hoc networks have many unique features, which make them suitable for a broad range of applications. One major issue in ad hoc networking is ad hoc routing. Many ad hoc routing protocols have been proposed in the literature. Recently, multipath approaches to ad hoc routing were proposed. Preliminary simulation results have shown that multipath routing can provide significant performance improvements in ad hoc networks.

CHAPTER III

PROBLEM STATEMENT AND METHODOLOGY

This chapter defines the scope of the problem under investigation and the research methodology. Specifically, we consider multipath ad hoc routing in this research. While most past work focused on improving the intrinsic performance of an ad hoc routing protocol, such as reducing control overhead and improving resilience to dynamic network topology changes, we focus on optimizing an important extrinsic performance metric, the aggregate throughput of an ad hoc network as measured at the transport layer.

The first section presents the problem definition, the performance metrics considered, as well as assumptions concerning the underlying network. Section 3.2 discusses the optimization approach and methodology. A comparison of unipath and multipath routing in Section 3.3 motivates some of the ideas presented in this work.

3.1 Problem Definition

Many existing ad hoc routing protocols primarily aim to achieve low control overhead and cope with dynamic topology changes in ad hoc networks, but few have addressed the interaction between routing and transport layer throughput performance. More importantly, most proposed protocols build routes in an ad hoc fashion without considering the underlying link characteristics or the capacity of the network, often resulting in non-optimal path selection and poor transport layer throughput. Some multipath routing algorithms for ad hoc networks have been proposed in the literature, but most of their optimization goals are limited to improving path resilience and reliable packet delivery. Furthermore, a comprehensive performance evaluation of multipath routing has not been published to our knowledge. In short, current ad hoc routing protocols have not fully addressed performance issues related to throughput.

3.1.1 Network under Investigation

We consider routing in a quasi-static wireless ad hoc network. It is assumed that nodes are randomly scattered within a bounded flat terrain and that the network is fully connected and non-partitioned. In other words, each node can communicate with at least one other node and each node participating in the network is reachable from all other nodes directly or via one or more intermediate nodes. We limit the size of the network so that clustering and hierarchical routing will not be essential.

A limited radio resource, frequency spectrum, is shared by all nodes for exchanging data. Ideally, each host should have fair access to the radio resource. For traffic patterns, each node in the network has equal probability to communicate with any other node in the network. For simplicity, we consider one-way constant rate Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) traffic between pairs of source and destination nodes.

3.1.2 Design Objective

The main objective in this research is to design an efficient data transmission and routing scheme for ad hoc networks to maximize the transport layer throughput for a given system capacity. We define the throughput metrics as follows. Suppose K transmission sessions are active between source-destination pairs $\{S_1D_1, S_2D_2, ..., S_KD_K\}$ and measured throughput at the destinations is $\{T_0, T_1, ..., T_K\}$, then the *aggregate transport throughput* of the network, T_A, is defined as the sum of the measured transport throughput T_i, or

$$T_A = \sum_{i=1}^{K} T_i$$

If we measure the traffic transmitted or forwarded by node N_n for all K session as $\{T_{0,n}, T_{1,n,}..., T_{K,n}\}$, the *aggregate network throughput*, T_N, can be expressed as the sum of the throughput P_n ($P_n = T_{0,n} + T_{1,n} + ... + T_{K,n}$) on all N nodes as follows,

$$T_N = \sum_{n=1}^{N} P_n = \sum_{n=1}^{N} \sum_{i=1}^{K} T_{i,n}$$

We define the *system capacity* of an ad hoc network as the maximum achievable aggregate network throughput. Suppose each node in the network is transmitting at the maximum possible rate and each node shares the medium equally with its neighbors, then the sum of the transmission rate of all nodes gives an estimate of the upper bound of the system capacity.

3.1.3 Assumptions Concerning Capacity

In reality, it is not feasible to directly compute or measure the system capacity of an ad hoc network, since the system capacity is a function of many factors in multiple protocol layers. Several factors may affect the maximum transmit and receive rate of a node. From the physical layer perspective, radio coverage range of a node, the use of a directional antenna pattern, and frequency channel allocation schemes may contribute to the effective data rate supported by the wireless links. In the wireless multiple access layer, the effective goodput may be influenced by traffic patterns from higher layers. Here, goodput is defined to be the maximum effective data rate measured at the receiver, excluding error packets and retransmissions. Some other limiting factors include the choice of un-slotted or slotted transmission, use of a collision avoidance mechanism, and random or polling access methods.

Researchers have defined several theoretical capacity bounds for ad hoc networks [24, 45], but they are not applicable to our problem. In this research, we do not attempt to determine the system capacity theoretically. Instead, we make some general assumptions concerning the capacity of the underlying physical and medium access control (MAC) layer rather than rely on simulation results of specific wireless physical and MAC protocols. Assuming each node participating in the network has equal share of the wireless medium, a simple ideal capacity distribution among ad hoc nodes is that the transmit capacity of each node is inversely proportional to its number of neighbors. This implies that the MAC layer can effectively minimize collisions between neighbor nodes and allocate the shared channel bandwidth fairly among the neighboring nodes. It is also assumed that each node can receive all data transmitted by neighbor nodes. Under these assumptions, the system capacity can be estimated as the sum of the maximum transmit rate of all nodes. These assumptions are considered for the optimization problem in this research.

3.2 Approach and Methodology

If we consider the system capacity of an ad hoc network as constant for a given topology, the problem of maximizing the aggregate transport throughput includes two aspects. First, for a given traffic pattern and maximum efficiency, we maximize the aggregate network throughput. In other words, we should use the network capacity as much as possible. Second, we maximize the efficiency of the ad hoc routing, defined as $\eta = T_A/T_N$. If packet drops occur, the efficiency will be less than one. Also, an increase in path lengths will reduce the efficiency. To increase the efficiency, the routing schemes should try to avoid congestion and packet drops on forwarding nodes and reduce path lengths. Link capacity and congestion condition should be considered in routing and trade-offs may exist with respect to path length.

The routing approach is the optimization variable in this problem. We assume that the physical and MAC layer performance will not be affected by higher layers. A natural choice for the routing approach for the problem is multipath routing. Multiple paths tend to use more available links from the source to destination, thus maximizing the aggregate network throughput. Using multiple paths may achieve load balancing among different paths and avoid congestion.

We use discrete event simulation to evaluate the performance of the proposed approach and to compare performance to existing multipath routing protocols. The simulation modules include abstract physical and MAC layer modules, routing modules for different routing approaches, IP network layer modules, TCP/UDP modules, and traffic generators.

3.3 Unipath versus Multipath Routing

The traditional single shortest path (unipath) routing approach faces many challenges in wireless ad hoc networks. Due to mobility, power control, and frequent link failures, routes in wireless ad hoc networks are much more dynamic than in wired networks. To reduce control overhead while maintaining up-to-date routing information, various techniques have been adopted. For example, OLSR broadcasts only partial topology information and uses multipoint relay nodes to reduce flooding of link state messages [11]. On-demand protocols such as AODV only try to find routes to a destination when required [61], thus avoiding periodic exchange of topology information across the whole network. Nevertheless, the costs of finding alternative routes after route disruption may still be high and extra delay in packet delivery is introduced. In [18], experimental results from wireless test beds have demonstrated that shortest single-path routing often chooses routes that have significantly less capacity, resulting in poor performance.

Unipath routing schemes sometimes do not respond well to network congestion. Packet queuing and packet drops introduce extra costs in the transmission path and waste system capacity. For multiple simultaneous connections sharing some links along their transmission paths, available bandwidth to each connection is reduced and congestion may occur. Regardless of the actions taken, e.g., queuing, random packet dropping, or packet prioritizing, the transmission costs will increase. Sending data along a single shortest path often results in creating congestions in some links while keeping some other potential links idle. Shortest path routing may not be the best choice in that case.

To illustrate the difference between unipath and multipath routing, we can make analogies between digital communication networks and transportation systems. Infrastructure backbone networks resemble interstate highways, access networks resemble in-state country roads, and ad hoc networks resemble streets in some metropolitan cities. Usually highways have higher capacity and utilization, while country roads have lower capacity and utilization. Now consider the traffic and routes in highways and country roads. Since there are no or only a few traffic lights on highways, there is usually no congestion leading to queuing delays. Thus, the variance of expected travel time is small. Finding routes between two remote towns in many cases is trivial. One first identifies the shortest or fastest interstate highways, and then selects the shortest in-state routes to the highway.

The traffic and routes in metropolitan cities are quite different. The capacities of roads are usually of the same order of magnitude and all roads have high utilization, especially during rush hours. Many intersections of roads (routers) have traffic lights (queues), and expected travel delays, therefore, have high variance. Usually there is more than one route between two places with approximately equal distance. However, determining the optimum (fastest) routes from the topology is non-trivial. Many drivers rely on real-time radio broadcast of traffic information to select routes.

For the same source and destination, routes are chosen dynamically and tend to use all possible combinations.

The above comparison gives us some hints on how to use multiple routes in ad-hoc networks. First, since there is no coordination between hosts in selecting routes, optimum routes cannot be solely inferred from topology. The actual costs of routes will be a function of the choice of routes of all connections. Information about the status and potential congestion of routes can help in deciding routes. Second, routes between end hosts should be selected dynamically and it is desirable to utilize all available routes dynamically to spread out the traffic to improve capacity utilization. Third, queuing increases the variance in the estimated cost of routes and makes the network more dynamic, thus making it more difficult to predict good routes prior to transmission. Indefinite queuing delays also affect end-to-end performance.

3.4 Summary

This chapter outlines the problem of optimizing the aggregate transport throughput performance of ad hoc routing protocols. Multipath routing is a promising technique for addressing this problem. However, most current ad hoc multipath routing extensions have not explicitly considered the transport layer throughput in route construction. In the next chapter, we define a multipath routing approach to address the problem.

CHAPTER IV

FRESNEL ZONE ROUTING

In this chapter, we propose a multipath routing approach based on “zone construction” in a wireless ad hoc network. The first section explains the “zone construction” method, which is an analogy of the Fresnel zones concept in wave propagation. The central idea in “Fresnel zone” routing (FZR) is to disperse traffic to different zones according to network load and congestion conditions, thus achieving better throughput and avoiding congestion on intermediate routers. The detailed operation of FZR is described in Section 4.2, including path construction, transmitting and forwarding, path maintenance, congestion control, and load balancing.

4.1 Overview

To address the unique features and requirements of ad hoc routing, researchers have developed many new routing algorithms, such as AODV [61], TBRPF [55], and FSR [59]. However, most of these algorithms still embrace the canonical view of routing for wired networks. They construct routes mainly based on the network topology, rather than the network capacity and end-to-end data transmission requirements. Packets in a data session often transverse a single fixed path. The bandwidth of the dedicated routers in the network core is statistically shared by many different data sessions. In this "route-centric" approach, routing is regarded as a separate network layer function and it is up to higher layers at the network edges to handle congestion in the network core.

Wireless ad hoc networks often render such a “route-centric” approach inappropriate. The topologies and routes change often so that maintaining consistent routes between communicating pairs is more difficult and inefficient. Unlike wired networks, there are usually no predetermined links or routes in wireless ad hoc networks, in other words, there are no dedicated routers or network core to carry traffic. As discussed in the previous chapter, the traditional shortest path routing approach faces many issues in such wireless ad hoc networks. Recently, people have begun to take a more unified view of routing and data transmission. For example, the “directed diffusion” routing algorithm [31] is a “data-centric” dispersive routing approach for wireless sensor networks. Before we go into the details of FZR, let us first review the concept of Fresnel zone in wave propagation.

4.1.1 Fresnel Zones in Wave Propagation

The concept of Fresnel zones is a central one in wave propagation. The zone construction was first proposed by the French engineer Augustin Jean Fresnel in 1818 in an attempt to explain diffraction phenomena using Huygens' principle [30]. The notion of a Fresnel zone is illustrated by the following scenario (See Figure 2). Consider a source of spherical wave at P_1. Surface S is the wave front at time t. According to the Huygens'-Fresnel's principle, the total field at P_2 can be viewed as the mutual interference of the secondary waves generated by the wave front S. Let us draw spheres centered at P_2 of radii $r_0, r_1, ..., r_n$ so that $r_1 - r_0 = r_2 - r_1 = ... = r_n - r_{n-1} = \lambda/2$, where λ is the wavelength, $r_0 = Q_0P_2$, and Q_0 is the point of intersection of P_1P_2 with S. The spheres divide the wave front surface into a number of zones $z_1, z_2, ..., z_n$ called spherical Fresnel zones. While the first zone is in the shape of a spherical segment, all others are spherical annuli. The path difference $r_n - r_{n-1}$ for each couple of adjacent Fresnel zones z_n and z_{n-1} is $\lambda/2$. Therefore, their secondary waves will radiate out of phase. We can also apply the zone construction to a surface perpendicular to P_1P_2. If plane S is moved along the line of sight P_1P_2 as illustrated in Figure 3, the zone boundaries are rotational ellipsoids around the Z axis with foci in points P_1 and P_2. For all three cases, the Fresnel zones are determined by the following equation.

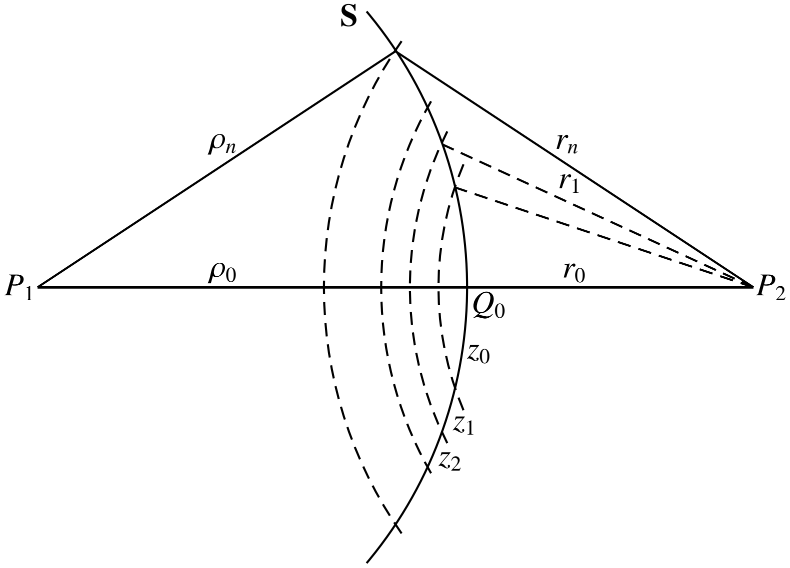

Figure 2: Spherical Fresnel zones

$$\rho_n + r_n = \rho'_{n-1} + r'_{n-1} + \lambda/2 = ... = \rho_0 + r_0 + n\lambda/2 = \text{constant}$$

Fresnel zones have found many applications in optics, acoustics, radio wave engineering, geophysics, and other fields. The zone construction provides easy methods for determining the field

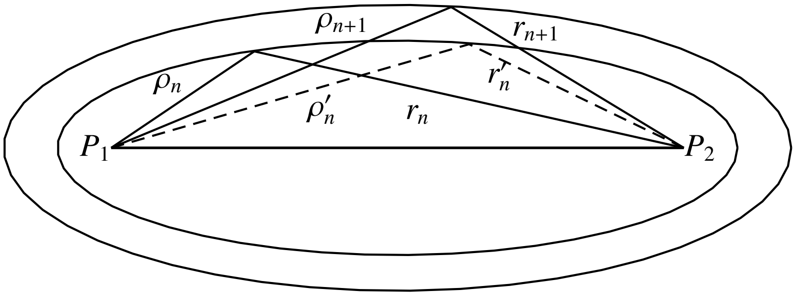

Figure 3: Rotational Fresnel zones

strength. The total field strength at P_2 can be calculated by summing the strength produced by each zone. Fresnel zones help to visualize how energy is transmitted in wave propagation, and they also prove to be a useful concept when one wants to control the transmit and receive patterns. In microwave links, the first Fresnel zone carries most of the energy, and, therefore, it must be clear of any obstacles.

4.1.2 "Fresnel Zones" in Ad Hoc Routing

We define "Fresnel zones" in ad hoc networks in a similar manner. Consider a source node S and a destination node D in an ad hoc network. The minimum hop distance between S and D is L hops. We divide all intermediate nodes that are reachable from node S and node D into different "Fresnel zones." If the shortest path from S to D via a given node has $L + n - 1$ hops, this node lies in the n-th "Fresnel zone." The sum of minimum hop distances from this node to source S and to destination D is $L + n - 1$ hops. The first zone includes all nodes on the direct shortest paths from the source to the destination. Figure 4 shows the first five zones for an example network topology. For dense networks, the zone construction has the shape of ellipsoids with foci in the source and destination nodes. We can make the following observations from the zone construction.

- All intermediate nodes on the direct shortest paths from S to D lie within the first zone and there exists at least one shortest path. When multiple shortest paths exist, they can be either disjoint or non-disjoint. Shortest paths usually have smaller delays and it is desirable to forward traffic along shortest paths. However, the number of available shortest paths may often be limited and the amount of traffic that can be forwarded may be limited by the capacity of one or more intermediate nodes.
- Paths via the second zone with hop distance $L + 1$ can be either disjoint or non-disjoint with the paths in the first zone. For the disjoint case, all intermediate nodes on the path lie within the second zone. For the non-disjoint case, the non-disjoint paths consist of nodes on the paths in the first zone, with hop distance L, and nodes on the paths in second zone , with hop

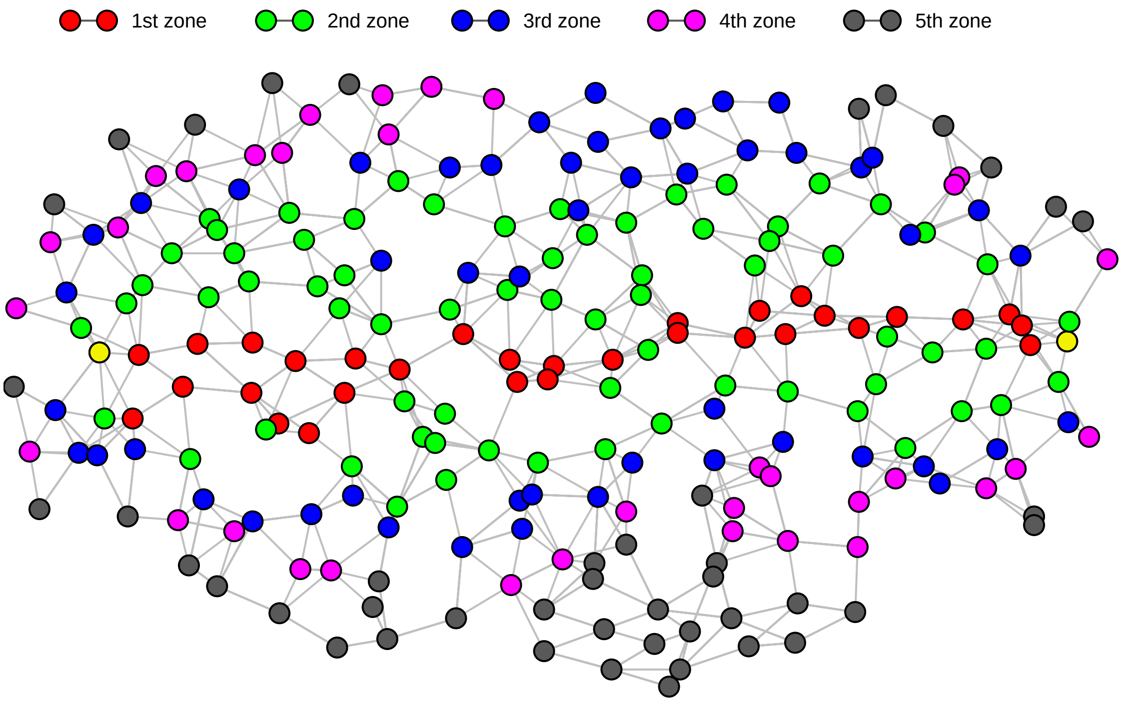

Figure 4: Illustration of "Fresnel zones" in an ad hoc network

distance $L + 1$, and there exists at least one node on the forward path in the second zone that is exactly one hop away from the first zone. Paths via second zones can be used as alternate paths when the shortest paths in the first zone become congested.

- Similarly, paths via the third zone with hop distance $L+2$ can be either disjoint or non-disjoint with the paths in the first and second zones. The non-disjoint paths via the third zone include partial paths from the first and second zones. Paths via the third zone have larger delays, but they can provide more alternative paths. This observation can be generalized to higher number zones.
- An intermediate node can belong to multiple paths from the source to destination that have different sets of intermediate nodes. If we take into account all shortest paths via nodes in different zones, these paths form meshed forward paths with the shape of ellipsoids.

It is also possible to use other metrics in zone construction, such as the average packet forwarding delays and the reciprocal of the link bandwidth. The zone construction method is similar and the idea is to group intermediate nodes according to the efficiency in forwarding data. In this research, we focus on using hop count as a simple metric. Considering that the transmission and propagation delay are relatively low compared to queueing delays in wireless networks, the average total delay on a forward path normally increases linearly with the number of hops. Hence, it is a reasonable choice to use hop count as the metric to measure the quality of paths.

4.1.3 “Fresnel Zones” for Data Forwarding

In an ad hoc network, if we opt to use all available loop-free paths from the source to the destination for data forwarding, how should we construct the forward paths from all possible path combinations? When multiple forward paths exist, how should we distribute the traffic among different paths? To answer these questions, we consider two factors in data forwarding for each intermediate node: (i) the delays in the forward path and (ii) the maximum amount of traffic that can be effectively forwarded via this node.

The zone construction provides a convenient way to construct loop-free routes according to the effectiveness in forwarding traffic of intermediate nodes. We define first-order paths as the shortest paths from the source to the destination via the first zone that have hop distance of L. Similarly, second-order paths are the shortest paths via one or more intermediate nodes in the second zone that have hop distance of $L + 1$. Paths with different orders are a subset of all loop-free paths from the source to the destination and these paths have different efficiency in forwarding data packets. Since the direct shortest (first-order) paths usually have lower costs and delays, we should first try to maximize the utilization of the first-order paths. When the first-order paths become congested, we can distribute traffic to higher-order paths.

The zone construction can help to decide how different intermediate nodes should cooperate in forwarding data packet, without explicitly determining all possible path combinations. The idea is to confine and forward data packets within the “Fresnel zones,” and the intermediate nodes decide how to forward the data packet based on available forward paths and bandwidth. In the first zone, each intermediate node has at least one forward path with minimum hop distance. Intermediate nodes can distribute the traffic among all available next hops. When an intermediate node in the first zone is adjacent to a node in the second zone, it can also forward packet using alternative paths via the second zone. For intermediate nodes in the second zone, it is desirable to use available forward paths within the second zone that are disjoint from the paths in the first zone. Alternatively, they can forward data packets back to the first zone or to forward paths via the third zone.

By distributing traffic to all available paths within the first zone and to higher-numbered zones, we effectively disperse traffic to more intermediate nodes, thus reducing the bandwidth requirement on forwarding nodes and the chance of congestion. Using meshed forward paths and providing multiple forward paths on intermediate nodes can reduce packet loss due to route disruption. Even though an individual path may change frequently, the zone construction may remain relatively constant. At the same time, there are important trade-offs in dispersing traffic to more intermediate nodes and to higher-numbered zones. Paths via higher-numbered zones have larger delays and require more resources for forwarding packets. To avoid potential interference among multiple communication sessions, it is also desirable to limit the total number of zones in use. The proposed

FZR algorithm uses the only the first two zones.

4.2 Detailed Operation of FZR

This section describes the detailed operation of the "Fresnel zone" routing algorithm. The functions of FZR include path construction, packet forwarding, path maintenance, and congestion control.

4.2.1 "Fresnel Zone" Path Construction

FZR uses only the first two zones for data forwarding. Since the first zone contains all nodes on the direct shortest paths from source to destination, we can identify all nodes in the first zone for given source-destination pairs if we know the shortest distances between all pairs of nodes. In FZR, each node maintains a distance table for recording the distance to all other nodes. Each record in the table includes the following fields: *destination address*, *sequence number*, *distance in hops*, and *next hops*. The *next hops* field contains the addresses of the next hop neighbors on the shortest paths. If the source and intermediate nodes send packets via all of these next hop neighbors, all nodes the packets will transverse are in the first zone.

To exchange distance information, each node periodically broadcasts HELLO messages to its neighbors. The HELLO message contains one or more distance records. Each distance record in the HELLO message contains the following fields: *destination address*, *sequence number*, and *hop distance*. When broadcasting a new distance record, a node sets its own address as the *destination address* and copies its sequence number to the distance record. The *hop distance* is initially set to one. Upon receiving a distance message, a node inspects the distance records and updates its distance table accordingly.

- If the distance record to the destination has a more recent sequence number, the node updates the sequence number and hop distance in the distance table. The source address of the HELLO message is added as a next hop node for the distance record. If the hop distance to the destination is changed, other old next hop entries are removed.
- If the distance record has an identical sequence number and distance as in the distance table, but is received from a different neighbor, this neighbor is added as another next hop entry.
- If the distance record contains a shorter distance, but identical sequence number as in the distance table, the node updates the distance and removes other old next hop entries.

Upon receiving a distance record for a node with a more recent sequence number or with a shorter distance, a node increases the distance in the record by one hop and forwards the distance record to its neighbors. It is possible that distance records with shorter distances arrive after records with longer distances. To avoid unnecessary broadcasts, these update messages are constructed after some delay. When distance records from all nodes are propagated through the network, each node has up-to-date distance information to all other nodes. The HELLO messages are also used to discover a node's neighborhood. When a node does not receive any HELLO messages from a neighbor node before a timeout occurs, it removes this node from its neighbor list. All forward paths and distance information via this neighbor are invalidated.

The above procedures are similar to, but not the same as, those used by the distance vector algorithm. The distance information updates are initiated by the destination nodes and propagated through the network, but are not periodically generated by all nodes. Using destination sequence numbers eliminate the count-to-infinity problem. Multiple next hops on the shortest forward paths to a destination are recorded. However, no routing decision is made to select from these forward paths.

FZR discovers alternate paths via second zones with an on-demand approach. The path discovery procedure can be initiated by the source node or intermediate nodes in the first zone. To avoid the situation where packets are routed to a single node in the second zone and then back to the first zone, we opt to find alternate paths that contain at least two intermediate nodes in the second zone. This is achieved by having both the originating node and the one-hop neighbors keep track of the forward paths discovered. Each node uses a forward path table to store the paths via nodes in the second zone. The forward path table contains the following fields: *source address*, *destination address*, and *next hops*. The procedure is described as follows.

- The source or intermediate node sends a path request message to its neighbors, specifying its own address as the originating node address, the source and destination address, the distance between the source and destination, and the shortest hop distance requirement to the destination. The distance requirement is set to one plus the shortest distance between the originating node and the destination node. The distance requirement field is necessary to avoid routing loops. Sequence numbers and the originating node address uniquely identify a path request message.
- When a one-hop neighbor receives a path request message, it first determines whether it is in the second zone by comparing the advertised distance in the message and its sum of distances to the source and destination. If the node is in the second zone and its distance to the destination is equal to the hop distance requirement, it processes and forwards the path request message to its neighbors, decreasing the hop distance requirement by one. A node is the neighbor of the originating node when the source address of the received path request message is identical to the originating node address field.

- When a two-hop neighbor in the second zone receives the path request message, it sends back a path reply message to the originating node via the upstream neighbor. The path reply message contains addresses of the originating node, the source node, and the destination node. Duplicate path request messages are not processed.
- The upstream one-hop neighbor in turn forwards the path reply to the originating node. This neighbor also records the forward path for this source-destination pair in its forward path table. It will also mark the forward paths with flags to indicate that it is a path via the second zone. Subsequent path reply messages from different two-hop neighbors to this node are processed, but not forwarded to the originating node.
- Upon receiving a path reply message, the originating node records the address of the node from which it received the path reply message. Subsequent replies from different neighbors are also processed. In this way, the originating node can discover one or more forward paths via the second zone.
- If a node in the second zone receives a different path request message from another node in the first zone and has up-to-date forward paths for the requested source and destination pair, it generates a path reply without forwarding the path request message to further neighbors.
- If a neighbor in the first zone receives a path request message from an upstream neighbor, it sends its own path request message to discover alternative paths. In this way, all intermediate nodes in the first zone try to discover alternative paths via the second zone.

The above procedure can be modified to discover paths that have more than two disjoint nodes in the second zone. To reduce the storage requirement for forward paths, only the first two forwarding nodes keep a record for the forward path.

One strategy for constructing paths via the second zone is to discover available alternate paths from the source and all intermediate nodes when the source node initiates communication to the destination node. This strategy requires more memory to store routing information and more bandwidth for the path request and reply messages, but has the advantage of acquiring more alternative paths. A relatively conservative strategy is to discover alternative paths only when congestion and route disruption occurs. Another strategy is to discovery alternative paths on the source or intermediate nodes that have only one forward path available. Such alternate paths can be used simultaneously or as backup paths.

4.2.2 Transmitting and Forwarding

The main idea in FZR is to spread traffic to multiple forward paths to reduce congestion and to improve utilization of network capacity. We adopt a simple approach for distributing traffic to multiple forwarding paths. If multiple forward paths via either the first or second zone are available, traffic is distributed proportionally to all forward paths. When forward paths via both the first and the second zones are available, an intermediate node distributes load via the first and second zones according to a ratio, which can be adjusted dynamically according to available bandwidth and congestion conditions.

Since shortest forward paths are set up by exchanging HELLO messages between neighbors, a source node can readily send packets to a destination node along all shortest paths in the first zone. On each intermediate node, packets are distributed randomly to all available paths with equal probability. To use forward paths via the second zone, the source node must initiate the path discovery procedure to the destination. After discovering alternate paths via the second zone, source and intermediate nodes begin to send packets via both the first and second zones. The detailed procedure for packet forwarding is outlined below.

- The source and intermediate nodes in the first zone check whether forward paths via the second zone exist for the source and destination pair by querying the forward path table. If such paths do exist, the node chooses to use the paths via the second zone with probability $1-\gamma$, where γ ($\gamma \leq 1$) is the fraction of traffic assigned to the first zone. Otherwise, the node distributes all traffic to the first zone. If the path request is not yet initiated, the source node sends the first path request.
- If forward paths for the source and destination pair exist on intermediate nodes in the second zone, the node uses only these forward paths. These paths in the forward path table are flagged as second-order paths. Other intermediate nodes in the second zone will simply distribute traffic to all available shortest paths to the destination.
- After determining the type of forward paths to use, the forwarding node sends packets to all available paths with equal probability. This is achieved by selecting the paths according to random numbers generated for each packet.

The ratio of traffic in the first and second zone can be adjusted dynamically by each intermediate node. There is a trade-off between dispersing traffic and using shortest paths. Some methods to adjust γ are discussed in later sections. It is also possible to adjust the ratio of traffic among different forward paths within the same zone, but the forwarding operations are more complicated.

4.2.3 Path Maintenance

FZR uses meshed multiple paths to provide resilience to route disruption and robust data forwarding. Since the zone construction and path discovery rely on accurate distance and shortest path information, it is important to detect link failures and route disruptions. Although we focus on quasi-static ad hoc networks in this work, FZR has the potential to be used in mobile networks.

Distance updates are periodically generated by destination nodes and propagated through the network. Due to its broadcast nature, it is desirable to reduce the frequency of the distance update broadcast. Instead, the distance maintenance procedure is used when a link failure or route disruption is detected.

- When a new neighbor node is discovered, a node sends a new distance message containing all the up-to-date distance records to this neighbor. The neighbor updates its distance table and forwards distance updates to further neighbors if needed.
- When a node detects that a neighbor is lost, it invalidates all shortest paths via this neighbor. If this neighbor provides the only shortest path to a destination, the distance advertised by this node is no longer correct. Therefore, it sends a path maintenance message to all of its neighbors. Each neighbor checks its distance records to determine whether its distance to the destination node is changed, i.e., if the broken link provides the only shortest path to the destination. The neighbor, in turn, propagates this update to further neighbors if needed.
- Forward paths via the second zones may become obsolete when the distance among nodes changes and a node moves into other zones. To detect such changes, those nodes that store forward paths via the second zone also remember the distance between the source and destination nodes. When changes in distance to either source or destination are detected, the forward paths are invalidated and a new path request is sent.
- Local path maintenance is used to repair link failures and route disruptions. If a link failure in the forward paths via the second zone is detected, the node in the first zone removes this node from the next hop list in the forward path table. The node initiates a new path discovery procedure to discover new paths via the second zone. If a node in the second zone finds that the forward paths to the next hop breaks, it removes the node from the forward paths. If a route to the destination breaks, the node broadcasts a route break message to the upstream nodes in the first zone, which update their forward table entries and initiate new a path discovery procedure when necessary.

4.2.4 Congestion Control and Load Balancing

The goal of FZR is to disperse traffic onto more intermediate nodes and to, thus, alleviate congestion on intermediate nodes. There are trade-offs between dispersing traffic and improving the utilization of nodes on shortest paths. The ratio of traffic sent via the first zone and the second zone is used to adjust the dispersiveness of traffic and provide certain degree of load balancing. This ratio can be adjusted dynamically according to available bandwidth and congestion conditions. FZR uses heuristic methods to dynamically adjust this parameter on intermediate routers.

- The initial traffic ratio is set according to the number of available forward paths via the first and second zone. Suppose there are m forward paths in the first zone and n paths in the second zone, an intermediate node sets the traffic ratio to $\gamma = km/(km + n)$, where k is a constant. A larger value of k will distribute more traffic via the first zone.
- Each intermediate node can optionally monitor the traffic forwarded by its neighbors to estimate the forwarding capacity for an individual path. The ratio can be adjusted according to the estimate.
- When a node drops a certain amount of traffic from an upstream node, it can send a congestion notification to the upstream node, which in turn fine tunes the forwarding ratio.

4.3 Summary

In this chapter, we propose “Fresnel zone” routing for wireless ad hoc networks based on the “zone construction” in wireless ad hoc networks. The “zone construction” method offers a convenient way to group intermediate nodes according to their effectiveness in forwarding packets. The central idea in FZR is to disperse traffic to more intermediate paths within the first and second “Fresnel zones.”

FZR constructs meshed multiple forwarding paths, and, therefore, inherently offers a form of load balancing and congestion control within the network layer. However, some assumptions about network congestion for certain transport layer protocols, in particular TCP, may be violated. We discuss these performance issues in next chapter.

CHAPTER V

SIMULATION AND RESULTS

This chapter presents the simulation experiments and results. The goal is to verify the correct operation of FZR and evaluate its performance using discrete event simulation. The first section summarizes the simulation environment and implementation details of the models for each layer. Section 5.2 describes the simulation parameters, performance metrics considered, and experimental procedures. Simulation validation and verification are also discussed. Section 5.3 presents the simulation results and compares different routing schemes in wireless ad hoc networks.

5.1 Simulation Models

In this work, we build simulation models using the Objective Modular Network Testbed in C++ (OMNeT++) package [78], an object-oriented discrete event simulation environment. OMNeT++ is capable of simulating communication networks, as well as "complex IT systems, queueing networks and hardware architectures" [78]. The basic component in OMNeT++ is a simple module, which implements the module behavior and handles simulation events. A compound module may contain one or more simple or compound modules, but it does not handle simulation events. An OMNeT++ simulation model consists of hierarchically nested modules. Modules communicate by exchanging messages. Besides discrete event simulation, OMNeT++ also has built-in support for generating pseudo-random numbers, collecting simulation statistics, and configuring multiple simulation runs.

5.1.1 Top-level Modules

In our simulations, the system has one top-level *adhocnet* compound module. The *adhocnet* module represents the whole ad hoc network and contains *adhocnode* compound modules (See Figure 5). The *adhocnet* module is responsible for setting up the network topology and individual *adhocnode* modules. Each *adhocnode* module represents an ad hoc node in the network. The *adhocnode* modules include application, network, link, and physical layer modules. Modules for different layers within a node exchange messages directly with modules for adjacent layers. Neighboring nodes exchange messages directly via the physical layer modules.

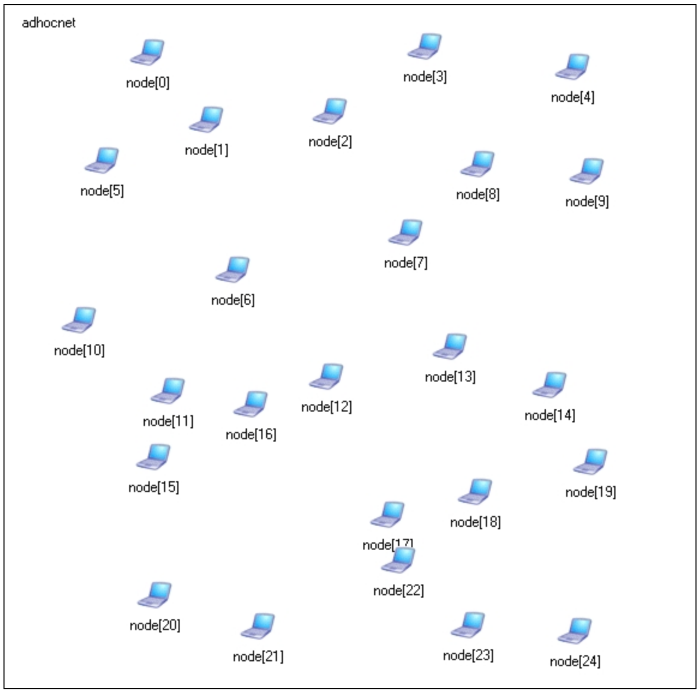

Figure 5: Ad hoc network module

5.1.2 Physical Layer Module

The *abstractphy* module implements the abstract physical layer. It maintains a list of all its neighbors, and, thus, can send frames directly to them. When the *abstractphy* module receives a data frame from the link layer, it adds a header to the frame and sends it out to neighboring nodes. Upon receiving a frame, the *abstractphy* module removes the header and forwards the frame to the link layer.

If the wireless medium has bandwidth BW shared by N_b adjacent nodes, the abstract physical layer can send frames at a constant rate that is inversely proportional to the number of neighbors,

$$R_s = \frac{BW}{N_b}$$

The physical layer can receive at a rate not less that the sum of the transmission rate of its neighbors. Therefore, it can receive all frames from its neighbors. This abstract model assumes that neighboring nodes can maintain perfect synchronization and transmission scheduling and that there are no collisions or retransmissions at the physical layer. The abstract model does not consider frame drops or frame errors.

5.1.3 Link Layer Modules

The link layer compound module consists of two sub-modules, the *link* module and the *outputQueue* module (See Figure 6). The main functions of the link layer module are to handle packets from the network layer and the physical layer and to handle the packet output queue. For simplicity, there is no input buffering in the link layer. IP datagrams from the network layer first go to the output queue. If the queue is full, incoming packets are dropped. The *link* module will keep fetching datagrams from the output queue unless it is empty. The *link* modules adds headers to the IP datagram to form a frame, sends the link frame down to the physical layer, and waits until the physical layer has finished transmitting the frame. When a frame arrives from the physical layer, the *link* module removes the header and sends the IP datagram up to the network layer.

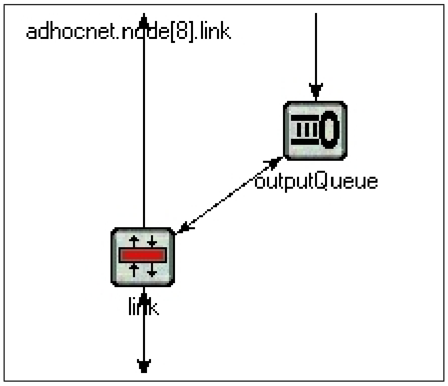

Figure 6: Link layer module

5.1.4 Network and Transport Layer Modules

The network and transport layers are implementations of the standard TCP/IP protocol stack, adopted from the IPv4 implementation in OMNeT++ [83]. The network layer module consists of the *proc* module, the *localDeliver* and the *send* modules (See Figure 7). The *send* module receives TCP and UDP data segments from the transport layer, adds IP headers to the data segments, and then sends the IP datagram to the *proc* module. The *localDeliver* module forwards arriving IP

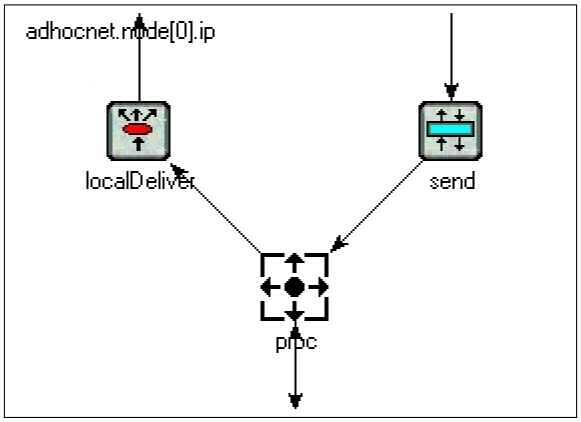

Figure 7: Network layer module

datagrams to the upper layer. The *proc* module is responsible for routing IP datagrams. It includes several routing agent implementations, which are described later. Packets from the transport layer as well as from the link layer are handled by the routing agents. The simplified IP layer does not support fragmentation of IP datagrams.

The TCP module is taken from the IPv4 implementation in OMNeT++. The TCP module implements several TCP variants, including delayed acknowledgements, fast retransmissions, fast recovery, and the NewReno algorithm [1]. For simplicity, socket support is not used for both TCP and UDP applications.

5.1.5 Application Layer Modules

Application layer modules include the constant rate UDP traffic generator client and sink server and the one-way TCP traffic generator client and sink server. The UDP and TCP sink servers receive data from the client and report the total number of bytes received and the overall throughput.

The UDP client generates constant rate UDP packets with fixed packet length. The interval between packet transmissions are uniformly distributed random numbers in range (t_{min}, t_{max}), where t_{min} is the smallest interval and t_{max} is the largest interval. The TCP server and client also include procedures to set up and close the connection.

5.1.6 Routing Agents

To evaluate the performance of FZR, we compare its operation and performance with other algorithms, including the single shortest-path distance vector routing, multipath distance vector routing (MDV), and split multipath routing (SMR) [42].

Since static network topologies are considered in this work, routes are pre-computed before the simulation of data transfers. The routing agents compute routes from each node to all other nodes in the network and save them in routing configurations, which are loaded by the network layer module at the beginning of a simulation run. Overhead in routing control messages is not considered in evaluating the throughput. It should be noted that the performance and operation may be different from actual implementation, for example, the SMR algorithm generates an excess amount of broadcast traffic and may affect data transmission sessions, if all of the nodes in the network initiate route requests to other nodes at the same time.

Distance vector routing is the traditional shortest path routing algorithm. Each node maintains one forward path entry for each destination. The routes are readily computable from a static network topology.

In MDV, all possible next hops on the shortest forward paths for a destination are stored on the routing table. When multiple forwarding paths are available, traffic are distributed equally into all available paths. This scheme is equivalent to FZR using only the first zone for data forwarding. It should be noted that this multipath scheme is different from the AOMDV algorithm [49].

The split multipath routing (SMR) is a dynamic source routing algorithm. It uses a modified route discovery procedures to discover two maximal disjoint paths for each source and destination pair. Traffic is distributed equally into two paths.

5.2 Experimental Design

This section describes the simulation experiments for the evaluation of four routing schemes, namely distance vector (DV), MDV, SMR, and FZR. We first list the performance metrics, fixed parameters, and factors for the simulation and then briefly discuss simulation validation and verification.

5.2.1 Performance Metrics

The aggregate transport throughput is the primary performance metric for routing optimization. The measured TCP or UDP throughput for all transmission sessions averaged over multiple simulation runs are used to compute the aggregate transport throughput for each combination of network topology and traffic pattern. Other performance metrics include average end-to-end packet delay and packet delivery ratio.

5.2.2 Parameters and Factors

Since the simulation models involve multiple protocol layers, the simulation space will increase exponentially if we adjust all parameters in different layers. To focus on the comparisons of different routing approaches, parameters for the physical (PHY) and link layer, as well as some for other layers, are fixed. These parameters are summarized in Table 1.

Table 1: Fixed Simulation Parameters

Parameter	Values	Note
N	100	Number of nodes in the network
DIM	1 km × 1 km	Terrain dimensions
BW	10 Mbps	Bandwidth shared by adjacent nodes
T_d	10 ms	PHY and propagation delays
H_l	128 bits	Link frame header size
L_q	500 Kbytes	Link layer queue size
MSS	1460 bytes	TCP maximum segment size
L_{UDP}	500 bytes	Fixed UDP data segment size

Network topology, routing schemes, and traffic patterns are the factors considered for the simulation. Since the number of multiple paths available depends on the actual network topology and network congestion may be affected by both routing and traffic patterns, we evaluate the performance of each routing scheme using different topologies and different traffic patterns.

We divide the topologies into three categories, low, medium, and high connectivity. For each category, we use a total of ten different random topologies in the simulation experiments. The random topologies are generated as follows. We divide the square terrain into equal-size grids and then place one node randomly inside each grid. The topology connectivity is adjusted by using different radio ranges for all ad hoc nodes. The average number of neighbors in low, medium and high connectivity topologies are approximately four, five, and six neighbors, respectively. All topologies are generated randomly and only those without partitions are used. The connectivity in terms of average number of neighbors are summarized in Table 2 for different radio ranges R (in meters). The connectivity also affects the system capacity.

Table 2: Connectivity (Degree) for Generated Topologies

Seed	1	2	3	4	5	6	7	8	9	10
R = 135m	3.94	4.22	4.24	4.12	3.98	4.38	4.14	3.96	4.08	4.40
R = 150m	4.98	5.38	5.30	4.86	5.20	5.22	5.16	5.16	5.34	5.44
R = 165m	6.20	6.48	6.48	6.24	6.52	6.32	6.38	6.36	6.46	6.68

For traffic generation, we use three traffic loads, low, medium, and high. The traffic load is adjusted by changing the number of transmitting and receiving pairs. There are four, eight, and twelve simultaneous data sessions for the low, medium, and high load conditions, respectively. Since the network topology is randomly generated, there is no need to randomly select the communicating pairs. Instead, the transmitting and receiving pairs are manually selected and applied to all topologies. Heuristics are used to ensure that the communicating nodes spread across the network, *i.e.*, that they are not located closely in the network.

For each combination of topology and traffic pattern, five repetitions with different random seeds are carried out. In each simulation run, all statistics are collected for a duration of at least 100 seconds after a start-up time of 10 seconds. This is to ensure that there will be enough samples for each statistical measurement and that all simulation runs are in steady state after the start-up time. For each simulation run, a trace file is checked to verify that all transmission sessions are active throughout the simulation interval.

5.2.3 Validation and Verification

The implementation of each individual simulation model is first verified. The modular design in OMNeT++ makes it possible to test modules for each layer separately. The following techniques are used.

- Trace files generated during simulation runs are checked to ensure that the module behavior and event handling are correctly implemented. By manually comparing the trace with the correct procedures, the correct implementation of each module is verified.
- Related performance metrics are measured at multiple layers for verification. For example, the total number of packets transmitted, received, and dropped are computed for each session. Throughput in the transport layer, link layer, and physical layer are compared.
- OMNeT++ provides a GUI environment for debugging a simulation. It can run the simulation step-by-step, in batch mode, or advance to a specific simulation time. When a simulation is paused, the simulation status, future events, module messages, and statistics variables can be inspected. The GUI environment is helpful in verifying the simulation models.

After the verification of each module, the system module is tested with different network topology and traffic patterns using the above techniques.

The simulation results are validated against known or analytical results. For routing operation, the generated routes are checked manually for the network topology. The routing tables on different nodes are also checked for consistency and routing loops. The TCP throughput T is compared with a simple analytic model given in [36].

$$T = \frac{0.75 \times W \times MSS}{RTT}$$

Here, W is the congestion window size, MSS is the maximum segment size, and RTT is the round-trip time. Table 3 compares the simulation and theoretical TCP throughput of a single data session for five different topologies. In these experiments, fast retransmission and fast recovery options for TCP are disabled and unipath distance vector routing is used. The simulation results match the theoretical values computed using the average congestion window size $W \cdot MSS$ and measured RTT.

Table 3: Measured and Theoretical TCP Throughput (Kbytes/s)

Topology	$W \cdot MSS$ (bytes)	RTT (s)	Throughput (Kbytes/s)	
			Simulation	Theoretical
1	76648	1.83	31.0	31.4
2	76648	1.91	32.0	30.0
3	76648	1.73	30.8	33.3
4	78256	1.56	35.5	37.5
5	78256	1.53	35.5	38.5

5.3 Results

In this section, we summarize the simulation results for FZR, as well as unipath distance vector routing, multipath distance vector (MDV) routing, and split multipath routing (SMR). Results for both the UDP and TCP transport protocols are presented. Graphs of the results are provided in the appendix.

5.3.1 UDP Throughput

We use constant bit rate (CBR) traffic to estimate the maximum throughput that can be achieved among communicating pairs. Each source node transmits at the rate of 2 Mbps or half of its maximum transmission rate, whichever is smaller. It should be noted that due to packet drops and congestion, there may be unfairness among different data sessions and the measured throughput may depend on the actual traffic pattern and network topology.

Table 4 summarizes the aggregate UDP throughput observed for all topologies with different connectivity under different traffic load conditions. The average (Avg) and standard deviation (Std) are calculated over ten random topologies with five replications per topology. The table also shows the percentage improvement (Imp) for using multipath schemes over unipath distance vector routing. On average, FZR outperforms all other schemes for all cases.

Table 4: Aggregate UDP Throughput Results (Mbps)

Connectivity	Traffic	DV		SMR			MDV			FZR		
		Avg	Std	Avg	Std	Imp	Avg	Std	Imp	Avg	Std	Imp
Low	L	2.20	0.7	2.74	0.5	24%	2.52	0.4	14%	3.35	0.2	52%
	M	3.02	0.6	3.29	0.6	9%	3.67	1.0	21%	4.40	0.8	46%
	H	3.79	0.9	4.22	1.5	11%	4.95	2.0	31%	5.35	2.3	41%
Medium	L	1.70	0.3	2.65	0.5	56%	2.18	0.3	28%	3.36	0.4	98%
	M	2.98	0.6	4.05	1.0	36%	4.01	0.6	34%	5.19	0.5	74%
	H	4.04	1.2	5.22	2.4	29%	5.61	1.4	39%	6.42	0.7	59%
High	L	1.95	0.1	2.71	0.1	39%	2.62	0.3	34%	3.32	0.0	70%
	M	2.97	0.3	4.11	0.3	38%	4.18	0.6	41%	5.30	0.4	78%
	H	4.65	1.5	5.53	0.8	19%	6.16	0.8	33%	6.35	0.6	37%

Graphs showing all UDP throughput results are included in the appendix. Figures 9 through 11 show the results for the low connectivity topology. In most topologies, FZR provides the highest throughput. While SMR and MDV also provide performance improvements, their results are more dependent on the actual topology. FZR provides approximately 50 to 150 percent throughput improvement for low traffic load conditions. For medium load traffic load conditions, FZR again provides consistent improvement in throughput. When the traffic load is high and the network itself becomes congested, the advantages of multipath algorithms become less prominent, but still offer some throughput improvement.

Figures 12 through 14 show the results for the medium connectivity topologies. Again, all multipath routing algorithms provide performance improvements. It should be noted that FZR performs better for medium connectivity topologies better than for low connectivity topologies. This suggests that FZR is able to take advantage of the additional connections between neighboring nodes.

Figures 15 through 17 show the results for the high connectivity topologies. All multipath routing algorithms provide performance improvements as in the other cases.

5.3.2 TCP Throughput

Table 5 summarizes the aggregate TCP throughput results for all topologies with different connectivity under different traffic load conditions. The traffic patterns and routing configuration are the same as for the experiments examining UDP throughput. The average and standard deviation are calculated over ten random topologies. The table also shows the percentage improvement for using multipath schemes over unipath distance vector routing.

Table 5: Aggregate TCP Throughput Results (Mbps)

Connectivity	Traffic	DV		SMR			MDV			FZR		
		Avg	Std	Avg	Std	Imp	Avg	Std	Imp	Avg	Std	Imp
Low	L	2.61	0.5	2.90	0.3	11%	2.65	0.4	2%	3.28	0.3	26%
	M	4.23	0.6	4.62	0.6	9%	4.73	1.2	12%	4.96	0.8	17%
	H	6.10	0.9	6.27	1.2	3%	6.72	1.6	10%	6.73	1.5	10%
Medium	L	2.05	0.5	2.64	0.6	29%	2.28	0.5	11%	3.11	0.6	52%
	M	3.81	0.4	4.34	0.6	14%	4.52	0.6	19%	4.81	0.5	26%
	H	5.66	0.7	6.38	0.6	13%	6.48	0.9	15%	6.29	0.7	11%
High	L	2.14	0.1	2.64	0.1	23%	2.58	0.2	21%	3.10	0.1	45%
	M	3.74	0.4	4.28	0.3	14%	4.52	0.4	21%	4.90	0.1	31%
	H	5.78	0.3	6.14	0.5	6%	6.63	0.6	15%	5.96	0.6	3%

Figures 18 through 20 show throughput results for the low connectivity topologies. It can be seen from the results that while multipath routing can provide more potential throughput improvements, especially when there is congestion on certain nodes, the improvement is less significant than for UDP. The throughput is limited by the TCP congestion control mechanism, which sends data at a rate less than the maximum available bandwidth on the paths.

Two limiting factors in TCP's congestion control may adversely affect throughput. The first is TCP slow start when some of the packets are dropped. Usually fast retransmission can be used to compensate for random packet losses, but fast retransmission is disabled in the simulation (See

Section 5.3.3). The second limiting factor is higher variance in the round-trip delay measurement. Out-of-order packet delivery makes the round-trip delay measurement in TCP less accurate with multipath routing.

Figures 21 through 26 show the TCP throughput results for the medium and high connectivity topologies. FZR and other multipath routing approaches all yield better performance than unipath distance routing.

5.3.3 TCP Performance with Fast Retransmission

In our simulation, we observe performance degradation for TCP over multipath routing when the fast retransmission option is enabled, which is not unexpected. In multipath routing, packets travelling along different paths may experience different queueing delays with large variance and, therefore, out-of-order packet delivery is common in multipath routing. This is verified by examining the TCP trace at both the sender and receiver.

Figure 8 compares the sender's congestion window growth in two data session, with and without fast retransmission. The network topology, traffic patterns, and other settings are the same for these two data sessions. There are no timeouts or packet drops during the data sessions and the congestion window grows exponentially in the slow start phase when the fast retransmission option is disabled. When the fast retransmission and fast recovery options are used, we see that the sender invokes the fast retransmission procedures four times, even though there are no packet drops or timeouts. Duplicate acknowledgements packets have prevented the sender from increasing its congestion window appropriately, resulting in much lower throughput. The duplicate acknowledgement packets may be introduced by out-of-order delivery of data packets from the sender to

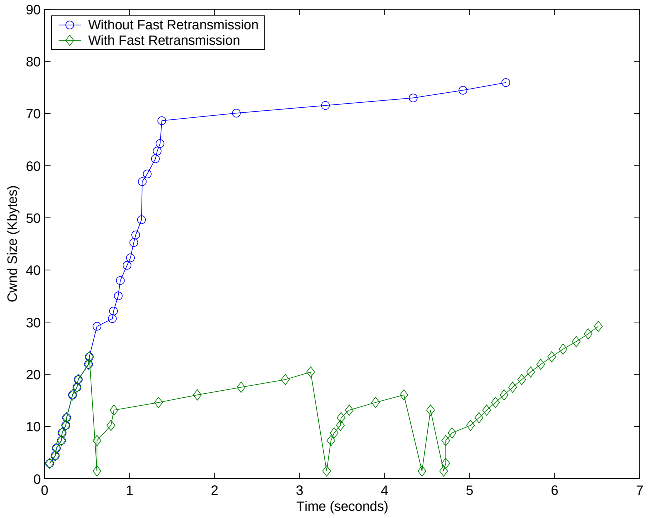

Figure 8: TCP congestion window size updates with multipath routing

the receiver or may be caused by out-of-order delivery of acknowledgement packets.

Simulation results also show that the TCP delayed acknowledgement option may reduce the number of duplicate acknowledgements. When the number of out-of-order data segments is small, the delayed acknowledgements may help to improve performance. However, when the traffic load on intermediate routers is high and there is large variance in queueing delays, the delayed acknowledgement will not prevent the sender from invoking unnecessary fast retransmissions.

5.3.4 UDP Packet Delivery Ratio

The average UDP packet delivery ratio over ten topologies of medium connectivity are shown in Figure 27 in the appendix. Since there is no congestion control mechanism in UDP, the network becomes more congested and the packet delivery ratios are lower as the traffic load increases. FZR effectively reduces network congestion and, thus, has the highest packet delivery ratio for all cases. Figure 28 shows the UDP packet delivery ratio for different connectivity levels. Multipath routing schemes achieve higher packet delivery ratio than unipath distance vector routing and has better performance with more connectivity.

5.3.5 TCP End-to-end Delay

Figure 29 and 30 show the average TCP end-to-end delay and variance. The delay samples are collected during simulations runs. The average delay and variance are computed for all sessions over ten random topologies with medium connectivity. Since the transmission and propagation delays are small as compared to queuing delays, the end-to-end delay is a good indication of the congestion condition on intermediate nodes. FZR has the lowest end-to-end delays, but the difference is less prominent when the traffic load is very high and most of the nodes in the network are congested. All multipath routing schemes provide lower end-to-end delays. However, they have larger variance in end-to-end delays.

5.4 Summary

The correct operation of the FZR protocol and its performance are evaluated through simulations. We present the performance results under different conditions. Performance results for unipath distance vector routing and split multipath routing are included for comparison. The simulation results have verified the correct operation of FZR and shown an improvement in throughput for UDP and, to a lesser extent, for TCP with FZR.

CHAPTER VI

SUMMARY AND CONCLUSIONS

6.1 Summary

In this research, we investigated options for improving the throughput, as seen by the transport layer, in wireless ad hoc networks through the use of multipath routing. Many traditional routing protocols do not work well in wireless ad hoc networks, since some of the design assumptions do not hold in a wireless environment, in particular, wireless ad hoc networks tend to have many shortest paths.

Prior work has shown that multipath routing appears to be a promising technique for wireless ad hoc networks. The first use of multipath routing is to provide redundant paths in case the primary ones fail. Simulation results have shown improvement over unipath routing in terms of path reliability and packet delivery ratios. Some other work has attempted to use multipath routing for load balancing, redundant transmissions, and security.

Although multipath routing has shown many potential benefits and improvement over traditional unipath routing, the design of multipath routing schemes is more difficult. Only a few multipath extensions for existing ad hoc routing protocols have been proposed, for example, split multipath routing (SMR) [42] and ad hoc on-demand multipath distance vector routing (AOMDV) [49, 50]. Most of the existing multipath extensions adopt a reactive approach and use network-wide broadcast to discover multiple paths. However, the chances of discovering multiple paths are reduced by optimization techniques to reduce route request flooding. Some of the proposed schemes (e.g., SMR) broadcast redundant route requests to discover maximal disjoint paths, but at the cost of generating an excessive amount of broadcast messages. Other techniques have been proposed, but no simple and efficient techniques for constructing multiple paths have been proposed so far as we know. The other difficult question is the criteria for selecting multiple paths. Most existing approaches have opted for using maximally disjoint paths, but some have argued that non-disjoint paths may work better.

In an attempt to solve some of the issues in multipath routing, this work proposed a new multipath routing approach, “Fresnel zone” routing, for wireless ad hoc networks. FZR builds multiple parallel paths based on the construction of "Fresnel zones” in an ad hoc network. The zone construction provides an easy way to coordinate packet transmissions and construct multiple paths of different lengths from source to destinations. The main design idea in FZR is to disperse traffic in the network, although this concept is not new. To evaluate the performance of FZR, simulations are carried out using discrete-event simulation models.

6.2 Conclusions

The operation and performance improvement of FZR are demonstrated by simulation experiments. The results show that FZR provides improved throughput for both UDP and TCP over existing multipath algorithms. The advantages of FZR include the following.

- FZR uses simple and effective procedures to construct multiple forward paths. The combination of proactive and reactive approaches makes it possible to avoid routing latency while maintaining low control overhead. Unlike most on-demand protocols, route requests in FZR are confined within zones and unnecessary network-wide broadcasts are avoided.
- Simulation results show improved aggregate transport throughput for most quasi-static scenarios. FZR provides better resource utilization on intermediate routers and, at the same time, lowers the chance of congestion.
- By reducing the chance of congestion, FZR effectively increases the packet delivery ratio for UDP packets. FZR also reduces TCP end-to-end delay caused by congestion and queuing on intermediate routers.

At the same time, like other multipath routing algorithms, FZR may generate a large number of out-of-order packets. An assumption in TCP fast retransmission is that out-of-order acknowledgements imply packet loss and congestion. The TCP sender initiates fast retransmissions after receiving triple duplicate acknowledgements. In our study, we have shown that using TCP fast retransmission over multipath routing causes unnecessary retransmissions and prevents the sender from increasing its congestion window size in an optimal manner.

6.3 Contributions

In this research, we proposed the concept of “Fresnel zones” in wireless ad hoc networks. While most past work has used disjointness as the primary multipath selection criterion, this work proposed using efficiency and capacity of intermediate nodes for selecting multiple parallel paths. The zone construction method offers an intuitive way to construct multiple forward paths according to underlying network capacity and end-to-end transmission requirements. Compared to existing multipath extensions, the “Fresnel zone” path construction is more efficient and suitable for ad hoc networks. The concept of zone construction can be extended using different path metrics.

Another important aspect of this research is the design of the FZR algorithm, whose operation is verified using discrete-event simulation. While prior work has focused on improving path resilience using multipath routing, the FZR addresses the transport layer throughput problem in ad hoc routing with a unique "data-centric" approach. Results show that FZR achieves better throughput and generates less routing overhead.

6.4 Future Work

This research has demonstrated the potential performance gains in FZR in a quasi-static environment, but further investigation for mobile scenarios is needed. FZR has the potential to be extended to mobile networks. However, the zone construction method has not considered the effect of rapid topology changes.

In this work, we use abstract physical and MAC layer models to verify the operation of FZR. It will be useful to demonstrate the performance of FZR using detailed models for existing physical and MAC specifications, such as the IEEE 802.11 standard. Prototype implementation of the FZR routing protocol will be useful to verify its operation. It will enable more accurate evaluation of the control overhead, resilience to topology changes, and actual throughput with dynamic wireless channels.

Another aspect for improvement is the investigation of the inherent congestion control mechanism in multipath routing. The goal of FZR is to avoid congestion within the network, therefore how to dynamically select paths according to congestion conditions is an important aspect of FZR. Since the congestion mechanism of TCP does not work well with multipath routing, an improvement to the existing TCP implementation may be worth investigating.

REFERENCES

[1] M. Allman, V. Paxson, and W. Stevens, “TCP congestion control,” IETF RFC 2581, April 1999. [Online]. Available: http://www.ietf.org/rfc/rfc2581.txt

[2] S. Bahk and M. El Zarki, “Dynamic multi-path routing and how it compares with other dynamic routing algorithms for high speed wide area networks,” in *Proceedings of Communications Architectures and Protocols Conference*, August 1992, pp. 53–64.

[3] B. Bellur and R. Ogier, “A reliable, efficient topology broadcast protocol for dynamic networks,” in *Proceedings of IEEE INFOCOM*, vol. 1, March 1999, pp. 178–186.

[4] A. Boukerche, “A simulation based study of on-demand routing protocols for ad hoc wireless networks,” in *Proceedings of 34th Annual Simulation Symposium*, April 2001, pp. 85–92.

[5] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, “A performance comparison of multi-hop wireless ad hoc network routing protocols,” in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, October 1998, pp. 85–97.

[6] J. Chen, P. Druschel, and D. Subramanian, “An efficient multipath forwarding method,” in *Proceedings of IEEE INFOCOM*, vol. 3, 1998, pp. 1418–1425.

[7] J. Chen and S.-H. Chan, “Multipath routing for video unicast over bandwidth-limited networks,” in *Proceedigns of IEEE GLOBECOM*, vol. 3, November 2001, pp. 1963–1967.

[8] C.-C. Chiang and M. Gerla, “Routing and multicast in multihop, mobile wireless networks,” in *6th International Conference on Universal Personal Communications Record*, vol. 2, October 1997, pp. 546–551.

[9] L.-P. Chou, C.-C. Hsu, and F. Wu, “A reliable multipath routing protocol for ad-hoc network,” in *Proceedings of 10th IEEE International Conference on Networks (ICON)*, August 2002, pp. 305–310.

[10] T. Clausen, G. Hansen, L. Christensen, and G. Behrmann, “The optimized link state routing protocol, evaluation through experiments and simulation,” in *Proceedings of IEEE Symposium on Wireless Personal Mobile Communications*, September 2001. [Online]. Available: http://hipercom.inria.fr/olsr/wpmc01.ps

[11] T. Clausen and P. Jacquet, “Optimized link state routing protocol,” IETF RFC 3626, October 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3626.txt

[12] S. Corson and J. Macker, “Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations,” IETF RFC 2501, January 1999. [Online]. Available: http://www.ietf.org/rfc/rfc2501.txt

[13] S. M. Corson and A. Ephremides, "A distributed routing algorithm for mobile wireless networks," *Wireless Networks*, vol. 1, no. 1, pp. 61–81, February 1995.

[14] S. R. Das, R. Castañeda, J. Yan, and R. Sengupta, "Comparative performance evaluation of routing protocols for mobile ad hoc networks," in *Proceedings of 7th International Conference on Computer Communications and Networks (IC3N)*, October 1998, pp. 153–161.

[15] S. R. Das, R. Castañeda, and J. Yan, "Simulation-based performance evaluation of routing protocols for mobile ad hoc networks," *Mobile Networks and Applications*, vol. 5, no. 3, pp. 179–189, September 2000.

[16] S. De and S. Das, "Dynamic multipath routing (DMPR): An approach to improve resource utilization in networks for real-time traffic," in *Proceedings of Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2001, pp. 23–30.

[17] S. De, C. Qiao, and H. Wu, "Meshed multipath routing: An efficient strategy in sensor networks," in *Proceedings of Wireless Communications and Networking (WCNC)*, vol. 3, March 2003, pp. 1912–1917.

[18] D. S. J. De Couto, D. Aguayo, B. A. Chambers, and R. Morris, "Performance of multihop wireless networks: Shortest path is not enough," *ACM SIGCOMM Computer Communication Review*, vol. 33, no. 1, pp. 83–88, January 2003.

[19] S. Dulman, T. Nieberg, J. Wu, and P. Havinga, "Trade-off between traffic overhead and reliability in multipath routing for wireless sensor networks," in *Proceedings of Wireless Communications and Networking Conference (WCNC)*, vol. 3, 2003, pp. 1918–1922.

[20] M. El Zarki and S. Bahk, "A distributed routing scheme for networks of the future," in *Proceedings of Second IEEE Workshop on Future Trends of Distributed Computing Systems*, September-October 1990, pp. 81–87.

[21] J. A. Freebersyser and B. Leinerr, "A DoD perspective on mobile ad hoc networks," in *Ad Hoc Networking*, C. E. Perkin, Ed. Addison-Wesley, 2001, pp. 29–51.

[22] D. Ganesan, R. Govindan, S. Shenker, and D. Estrin, "Highly-resilient, energy-efficient multipath routing in wireless sensor networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 4, pp. 11–25, October 2001.

[23] M. Gerla, X. Hong, and G. Pei, "Landmark routing for large ad hoc wireless networks," in *Proceedings of IEEE GLOBECOM*, vol. 3, November-December 2000, pp. 1702–1706.

[24] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, March 2000.

[25] Z. J. Haas and M. R. Pearlman, "The performance of query control schemes for the zone routing protocol," *IEEE/ACM Transactions on Networking*, vol. 9, no. 4, pp. 427–438, August 2001.

[26] Y. Hatanaka, M. Nakamura, Y. Kakuda, and T. Kikuno, “A synthesis method for fault-tolerant and flexible multipath routing protocols,” in *Proceedings of Third IEEE International Conference on Engineering of Complex Computer Systems*, September 1997, pp. 96–105.

[27] M. Heusse and Y. Kermarrec, “A new routing policy for load balancing in communication networks,” in *Proceedings of ACS/IEEE International Conference on Computer Systems and Applications*, June 2001, pp. 267–272.

[28] X. Hong, M. Gerla, Y. Yi, K. Xu, and T. J. Kwon, “Scalable ad hoc routing in large, dense wireless networks using clustering and landmarks,” *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 5, pp. 3179–3185, May 2002.

[29] X. Hong, K. Xu, and M. Gerla, “Scalable routing protocols for mobile ad hoc networks,” *IEEE Network*, vol. 16, no. 4, pp. 11–21, July-August 2002.

[30] H. D. Hristov, *Fresnel Zones in Wireless Links, Zone Plate Lenses and Antennas*. Artech House Publishers, 2000.

[31] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: A scalable and robust communication paradigm for sensor networks,” in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, August 2000, pp. 56–67.

[32] D. B. Johnson and D. A. Maltz, “Dynamic source routing in ad hoc wireless networks,” in *Mobile Computing*, T. Imielinski and H. Korth, Eds. Kluwer Academic Publishers, 1996, vol. 353.

[33] D. B. Johnson, D. A. Maltz, and Y.-C. Hu, “The dynamic source routing protocol for mobile ad hoc networks (DSR),” IETF Internet Draft, April 2003, work in progress. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-09.txt

[34] H. T. Kaur, S. Kalyanaraman, A. Weiss, S. Kanwar, and A. Gandhi, “BANANAS: An evolutionary framework for explicit and multipath routing in the Internet,” in *Proceedings of the ACM SIGCOMM Workshop on Future Directions in Network Architecture*, August 2003, pp. 277–288.

[35] S. Kim, W. Noh, and S. An, “Multi-path ad hoc routing considering path redundancy,” in *Proceedings of Eighth IEEE International Symposium on Computers and Communication (ISCC)*, March 2003, pp. 45–50.

[36] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*. Addison-Wesley, 2000.

[37] D. Lang, “A comprehensive overview about selected ad hoc networking routing protocols,” Master’s thesis, Technische Universit at Munchen, Germany, 2003, [Online]. Available: http://wwwbib.informatik.tu-muenchen.de/infberichte/2003/TUM-I0311.pdf.

[38] A. Laouiti, P. Muhlethaler, A. Najid, and E. Plakoo, “Simulation results of the OLSR routing protocol for wireless networks,” in *Proceedigns of 1st Mediterranean Ad-Hoc Networks Workshop (Med-Hoc-Net)*, 2002. [Online]. Available: http://hipercom.inria.fr/olsr/medhoc.ps

[39] C. Lee, X.-H. Lin, and Y.-K. Kwok, “A multipath ad hoc routing approach to combat wireless link insecurity,” in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 1, April 2003, pp. 448–452.

[40] S.-J. Lee and M. Gerla, “AODV-BR: Backup routing in ad hoc networks,” in *Proceedings of Wireless Communications and Networking Conference (WCNC)*, vol. 3, September 2000, pp. 1311–1316.

[41] S.-J. Lee, C.-K. Toh, and M. Gerla, “Performance evaluation of table-driven and on-demand ad hoc routing protocols,” in *Proceedigns of IEEE Personal Indoor and Mobile Radio Communications (PIMRC) Conference*, September 1999, pp. 297–301.

[42] S. Lee and M. Gerla, “Split multipath routing with maximally disjoint paths in ad hoc networks,” in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 10, June 2001, pp. 3201–3205.

[43] P. Leelapornchai and T. Stockhammer, “Progressive image transmission applying multipath routing in mobile ad hoc networks,” in *Proceedings of International Conference on Image Processing*, vol. 1, September 2002, pp. 553–556.

[44] B. Leiner, R. Ruth, and A. R. Sastry, “Goals and challenges of the DARPA GloMo program,” *IEEE Personal Communications*, vol. 3, no. 6, pp. 34–43, December 1996.

[45] J. Li, C. Blake, D. S. De Couto, H. I. Lee, and R. Morris, “Capacity of ad hoc wireless networks,” in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, July 2001, pp. 61–69.

[46] H. Lim, K. Xu, and M. Gerla, “TCP performance over multipath routing in mobile ad hoc networks,” in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 2, May 2003, pp. 1064–1068.

[47] X. Lin, M. Lakshdisi, and I. Stojmenovic, “Location based localized alternate, disjoint, multipath and component routing schemes for wireless networks,” in *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing*, October 2001, pp. 287–290.

[48] W. Lou and Y. Fang, “A multipath routing approach for secure data delivery,” in *Proceedings of IEEE MILCOM*, vol. 2, October 2001, pp. 1467–1473.

[49] M. K. Marina and S. R. Das., “On-demand multipath distance vector routing in ad hoc networks,” in *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, March 2001, pp. 14–23.

[50] M. K. Marina and S. R. Das, “Ad hoc on-demand multipath distance vector routing,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 6, no. 3, pp. 92–93, July 2002.

[51] S. Murthy and J. Garcia-Luna-Aceves, “Congestion-oriented shortest multipath routing,” in *Proceedings of IEEE INFOCOM*, vol. 3, March 1996, pp. 1028–1036.

[52] A. Nasipuri and S. Das, “On-demand multipath routing for mobile ad hoc networks,” in *Proceedings of Eight International Conference on Computer Communications and Networks*, October 1999, pp. 64–70.

[53] A. Nasipuri, R. Castañeda, and S. R. Das, “Performance of multipath routing for on-demand protocols in mobile ad hoc networks,” *Mobile Networks and Applications*, vol. 6, no. 4, pp. 339–349, August 2001.

[54] D. Niculescu and B. Nath, “Trajectory based forwarding and its applications,” in *Proceedings of the 9th Annual International Conference on Mobile computing and Networking*, September 2003, pp. 260–272.

[55] R. Ogier, F. Templin, and M. Lewis, “Topology dissemination based on reverse-path forwarding (TBRPF),” IETF Internet Draft, October 2003, work in progress. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-manet-tbrpf-11.txt

[56] P. Papadimitratos, Z. J. Haas, and E. G. Sirer, “Path set selection in mobile ad hoc networks,” in *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing*, June 2002, pp. 1–11.

[57] V. D. Park and M. S. Corson, “A highly adaptive distributed routing algorithm for mobile wireless networks,” in *Proceedings of IEEE INFOCOM*, vol. 3, April 1997, pp. 1405–1413.

[58] G. Pei, M. Gerla, X. Hong, and C.-C. Chiang, “A wireless hierarchical routing protocol with group mobility,” in *Proceedings of Wireless Communications and Networking Conference*, vol. 3, September 1999, pp. 1538–1542.

[59] G. Pei, M. Gerla, and T.-W. Chen, “Fisheye state routing: A routing scheme for ad hoc wireless networks,” in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 1, June 2000, pp. 70–74.

[60] G. Pei, M. Gerla, and X. Hong, “LANMAR: landmark routing for large scale wireless ad hoc networks with group mobility,” in *Proceedings of the 1st ACM International Symposium on Mobile Ad Hoc Networking and Computing*, November 2000, pp. 11–18.

[61] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc on-demand distance vector (AODV) routing,” IETF RFC 3561, July 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3561.txt

[62] C. E. Perkins, *Ad Hoc Networking*. Addison-Wesley, 2001.

[63] C. Perkins and E. Royer, “Ad-hoc on-demand distance vector routing,” in *Proceedings of Second IEEE Workshop on Mobile Computing Systems and Applications*, February 1999, pp. 90–100.

[64] C. E. Perkins and P. Bhagwat, “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers,” in *Proceedings of the Conference on Communications Architectures, Protocols and Applications*, August-September 1994, pp. 234–244.

[65] P. P. Pham and S. Perreau, “Performance analysis of reactive shortest path and multi-path routing mechanism with load balance,” in *Proceedings of IEEE INFOCOM*, vol. 1, March-April 2003, pp. 251–259.

[66] A. Qayyum, L. Viennot, and A. Laouiti, “Multipoint relaying: An efficient technique for flooding in mobile wireless networks,” INIRIA Technical Report RR-3898, 2000.

[67] R. Ramanathan and J. Redi, “A brief overview of ad hoc networks: Challenges and directions,” *IEEE Communications Magazine*, vol. 40, no. 5, pp. 20–22, May 2002.

[68] E. Royer and C.-K. Toh, “A review of current routing protocols for ad hoc mobile wireless networks,” *IEEE Personal Communications*, vol. 6, no. 2, pp. 46–55, April 1999.

[69] R. Ruppe, S. Griswald, P. Walsh, and R. Martin, “Near term digital radio (NTDR) system,” in *Proceedings of IEEE MILCOM*, vol. 3, November 1997, pp. 1282–1287.

[70] Y. Seok, Y. Lee, Y. Choi, and C. Kim, “Dynamic constrained multipath routing for MPLS networks,” in *Proceedings of Tenth International Conference on Computer Communications and Networks*, October 2001, pp. 348–353.

[71] C. K. Siew, G. Wu, and G. Feng, “On-demand QoS multipath routing,” in *Proceedings of the 8th International Conference on Communication Systems (ICCS)*, vol. 1, November 2002, pp. 589–593.

[72] S. Singh, M. Woo, and C. S. Raghavendra, “Power-aware routing in mobile ad hoc networks,” in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, October 1998, pp. 181–190.

[73] A. Srinivas and E. Modiano, “Minimum energy disjoint path routing in wireless ad-hoc networks,” in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, September 2003, pp. 122–133.

[74] X. Su and G. de Veciana, “Dynamic multi-path routing: Asymptotic approximation and simulations,” *ACM SIGMETRICS Performance Evaluation Review*, vol. 29, no. 1, pp. 25–36, June 2001.

[75] C.-K. Toh, “Associativity-based routing for ad hoc mobile networks,” *Wireless Personal Communications*, vol. 4, no. 2, pp. 103–139, March 1997.

[76] A. Tsirigos and Z. Haas, “Multipath routing in the presence of frequent topological changes,” *IEEE Communications Magazine*, vol. 39, no. 11, pp. 132–138, November 2001.

[77] Z. Tsirigos, A. Haas, “Multipath routing in mobile ad hoc networks or how to route in the presence of frequent topology changes,” in *Proceedings of IEEE MILCOM*, vol. 2, October 2001, pp. 878–883.

[78] A. Varga *et al.*, “Objective Modular Network Testbed in C++ (OMNeT++),” March 2004. [Online]. Available: http://www.omnetpp.org

[79] S. Vutukury and J. Garcia-Luna-Aceves, “MDVA: A distance-vector multipath routing protocol,” in *Proceedings of IEEE INFOCOM*, vol. 1, April 2001, pp. 557–564.

[80] S. Vutukury and J. J. Garcia-Luna-Aceves, “A simple approximation to minimum-delay routing,” *SIGCOMM Computer Communication Review*, vol. 29, no. 4, pp. 227–238, October 1999.

[81] L. Wang, Y. Shu, M. Dong, L. Zhang, and O. W. W. Yang, “Adaptive multipath source routing in ad hoc networks,” in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 3, 2001, pp. 867–871.

[82] L. Wang, L. Zhang, Y. Shu, and M. Dong, “Multipath source routing in wireless ad hoc networks,” in *Proceedings of Canadian Conference on Electrical and Computer Engineering*, vol. 1, March 2000, pp. 479–483.

[83] K. Wehrle, J. Reber, D. Holzhausen, V. Boehm, V. Kahmann, U. Kaage, and X. T. Nguyen, “IP-Suite: IPv4 ONMeT++ Implementation,” Institut fuer Telematik and Universitaet Karlsruhe, March 2004. [Online]. Available: http://ctieware.eng.monash.edu.au/twiki/bin/view/Simulation/IPv4Suite/

[84] J. Wu, “An extended dynamic source routing scheme in ad hoc wireless networks,” in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS)*, January 2002, pp. 3832–3838.

[85] K. Wu and J. Harms, “Performance study of a multipath routing method for wireless mobile ad hoc networks,” in *Proceedings of Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, August 2001, pp. 99–107.

[86] Z. Yao, J. Jiang, P. Fan, Z. Cao, and V. O. K. Li, “A neighbor-table-based multipath routing in ad hoc networks,” in *Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference*, vol. 3, April 2003, pp. 1739–1743.

[87] Z. Yao, Z. Ma, and Z. Cao, “A multipath routing scheme combating with frequent topology changes in wireless ad hoc networks,” in *Proceedings of International Conference on Communication Technology Proceedings (ICCT)*, vol. 2, April 2003, pp. 1250–1253.

[88] W. Zaumen and J. Garcia-Luna-Aceves, “Loop-free multipath routing using generalized diffusing computations,” in *Proceedings of IEEE INFOCOM*, vol. 3, April 1998, pp. 1408–1417.

[89] J. Zhao, B. Zhou, and J. Wu, “Quality of service based multipath routing for supporting real time wireless applications,” in *Proceedings of International Conferences on Info-tech and Info-net*, vol. 2, October-November 2001, pp. 122–127.

[90] L. Zhou and Z. J. Haas, “Securing ad hoc networks,” *IEEE Network*, vol. 13, no. 6, pp. 24–30, November-December 1999.

APPENDIX

GRAPHS SHOWING ALL RESULTS

A.1 UDP Throughput

The throughput results for UDP are summarized in the following figures according to network connectivity and traffic loads. Each figure compares throughput results based on ten random topologies for four routing schemes, namely unipath distance vector (DV) routing, split multipath routing (SMR), multipath distance vector (MDV) routing, and "Fresnel zone" routing (FZR). Refer to Section 5.3 for discussion concerning the results.

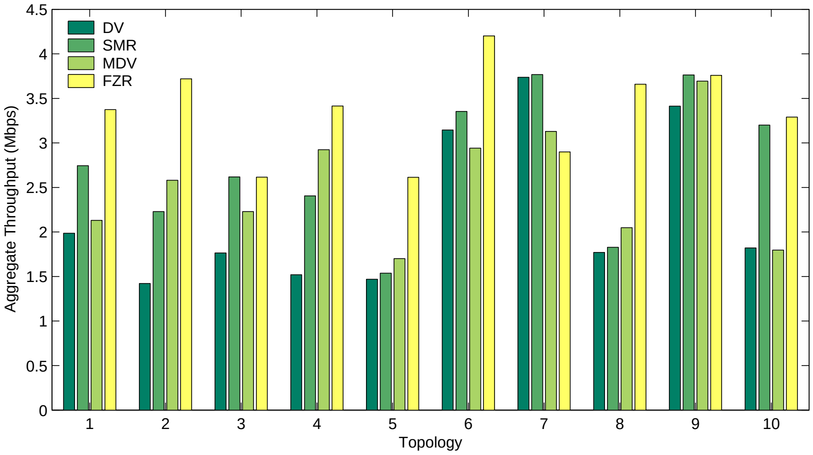

Figure 9: UDP throughput for low connectivity and low traffic load

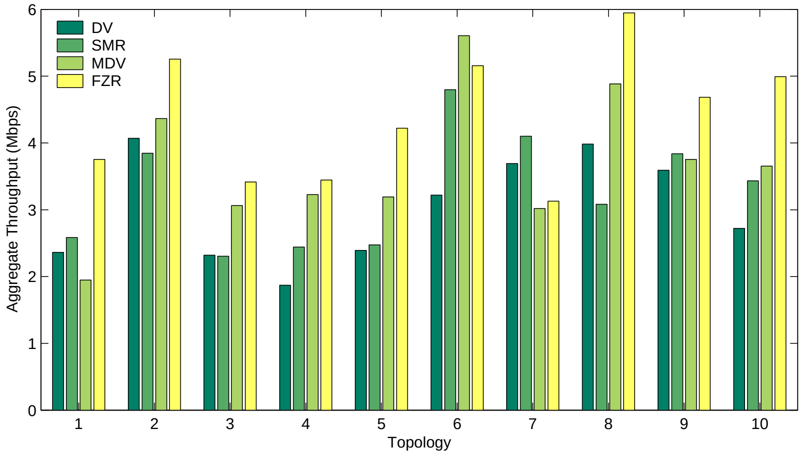

Figure 10: UDP throughput with low connectivity and medium load

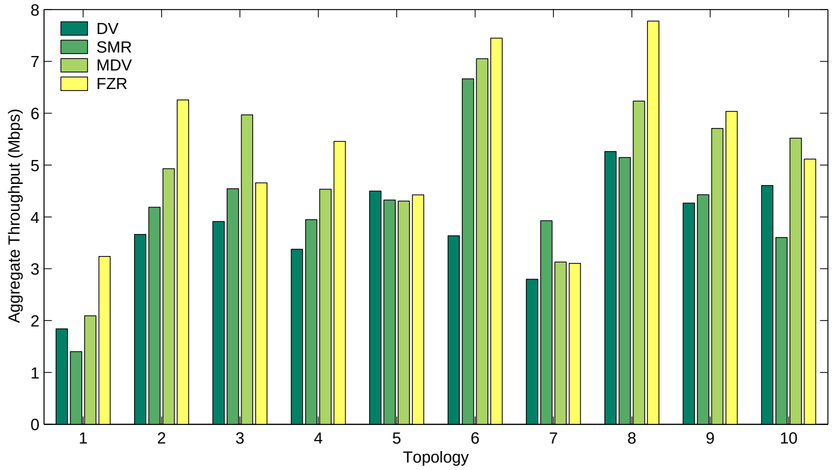

Figure 11: UDP throughput with low connectivity and high load

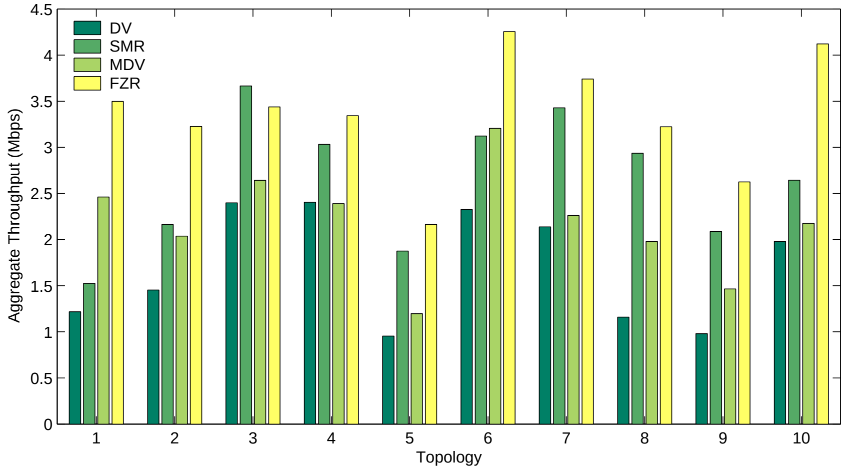

Figure 12: UDP throughput with medium connectivity and low load

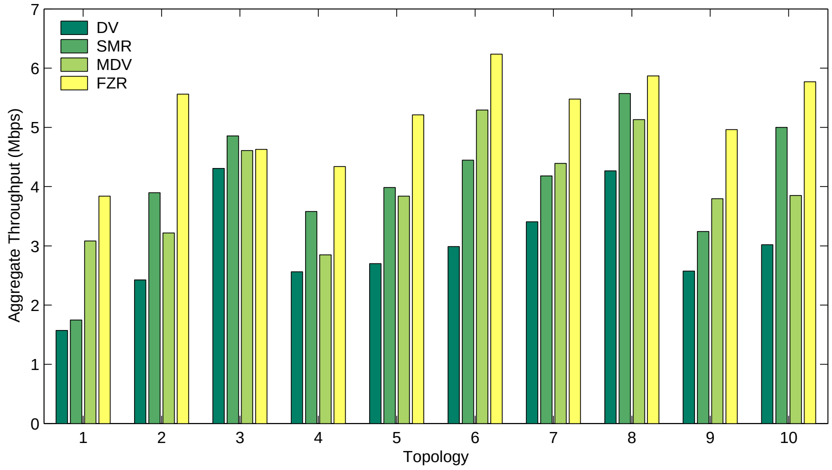

Figure 13: UDP throughput with medium connectivity and medium load

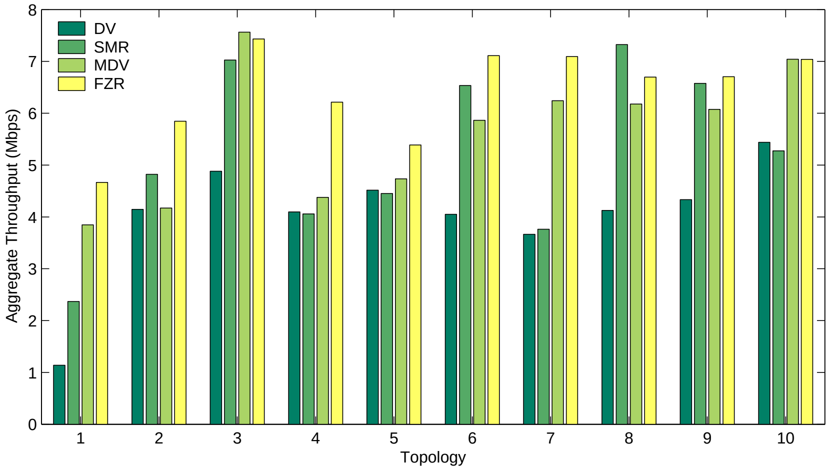

Figure 14: UDP throughput with medium connectivity and high load

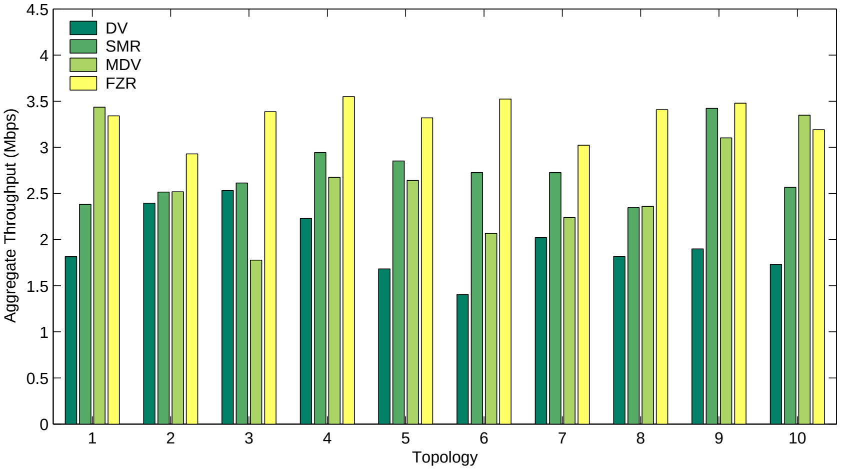

Figure 15: UDP throughput with high connectivity and low load

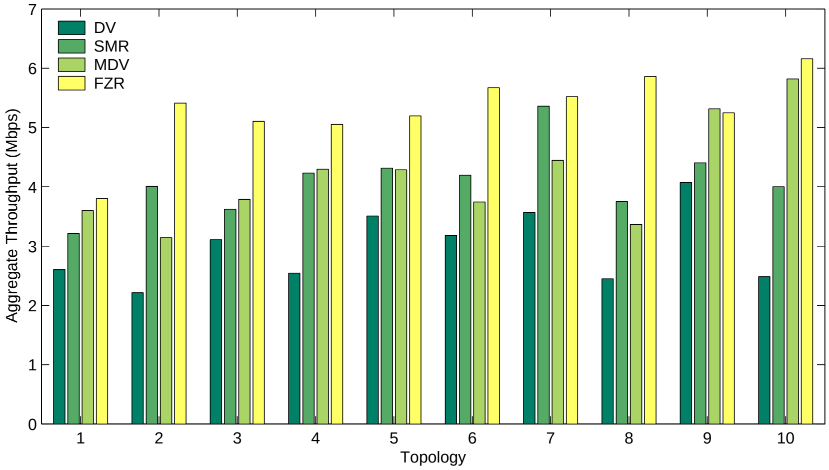

Figure 16: UDP throughput with high connectivity and medium load

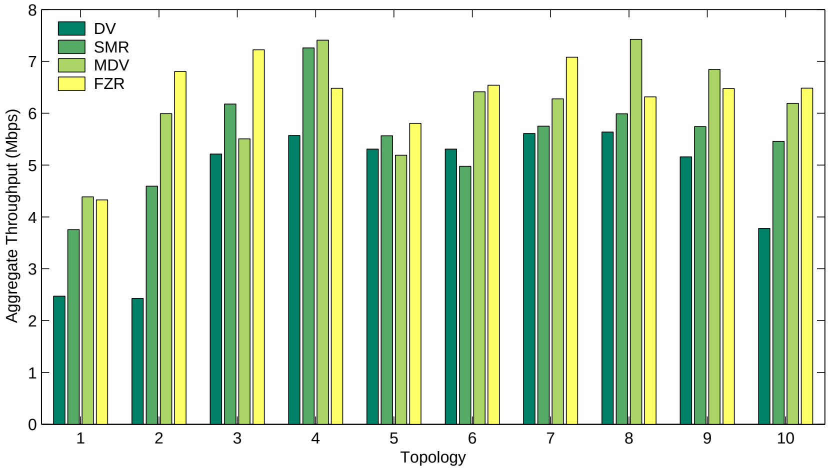

Figure 17: UDP throughput with high connectivity and high load

A.2 TCP Throughput

The throughput results for TCP are summarized in the following figures according to network connectivity and traffic loads. Each figure compares throughput results based on ten random topologies for four routing schemes. Refer to Section 5.3 for discussion concerning the results.

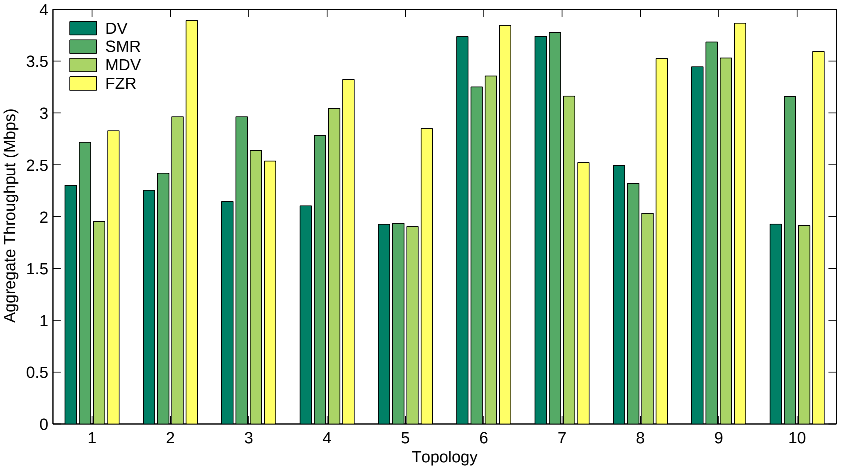

Figure 18: TCP throughput with low connectivity and low load

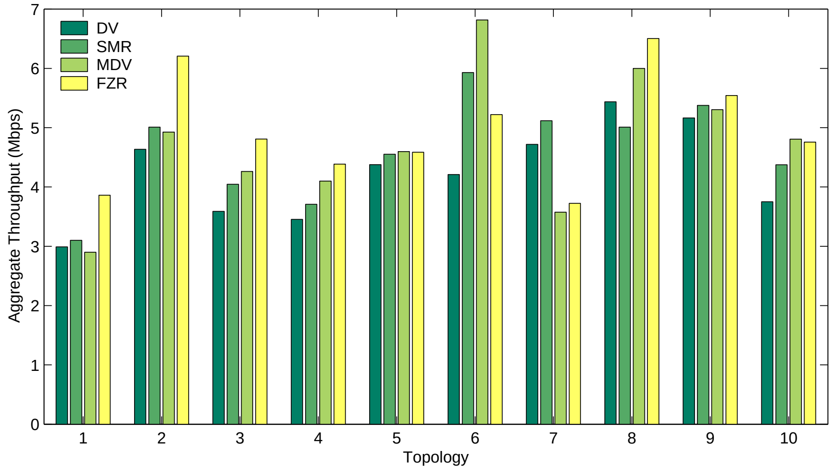

Figure 19: TCP throughput with low connectivity and medium load

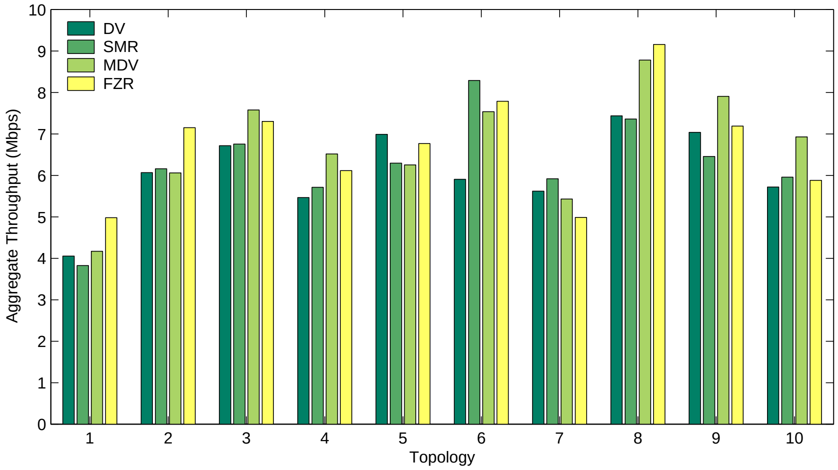

Figure 20: TCP throughput with low connectivity and high load

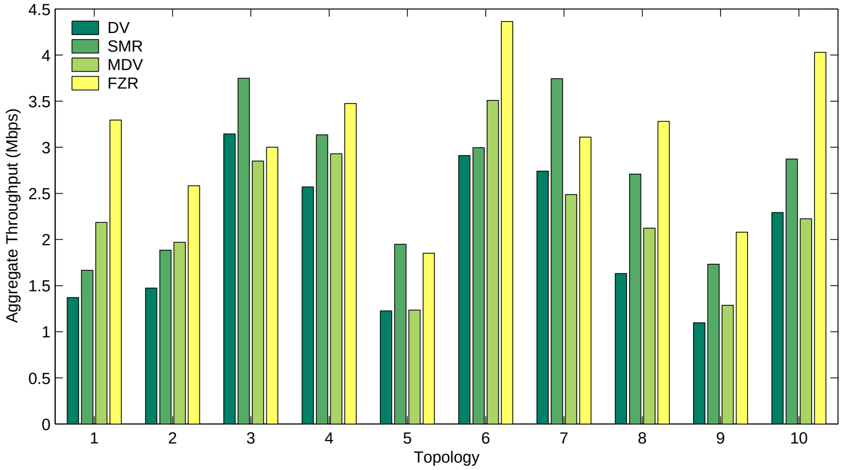

Figure 21: TCP throughput with medium connectivity and low load

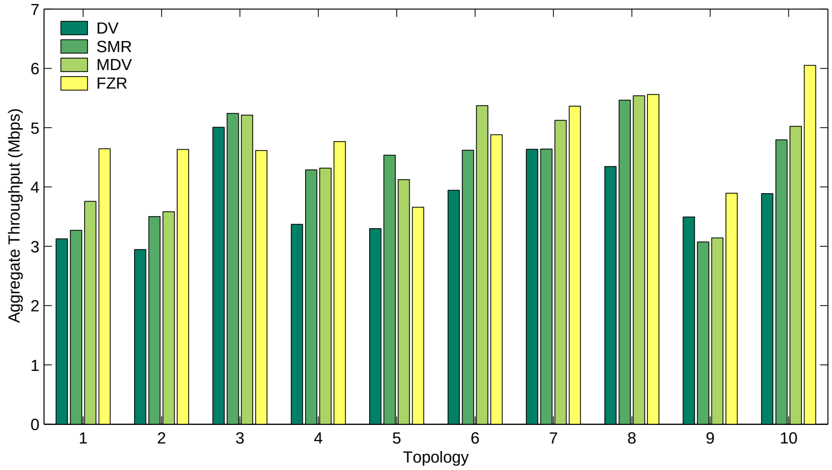

Figure 22: TCP throughput with medium connectivity and medium load

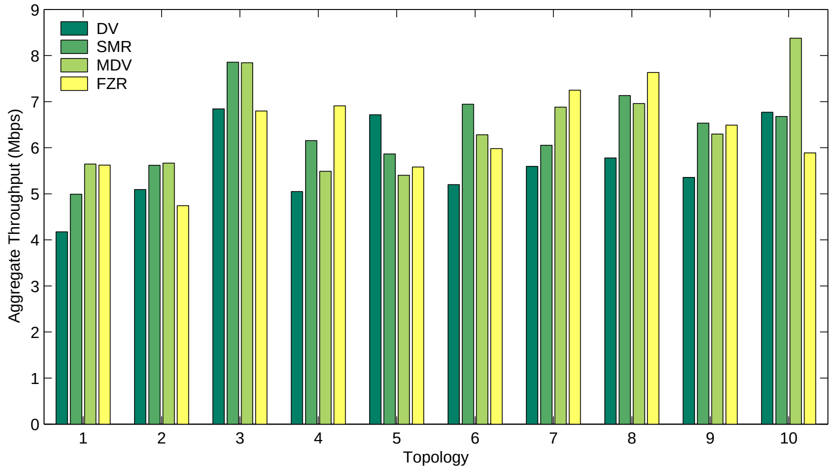

Figure 23: TCP throughput with medium connectivity and high load

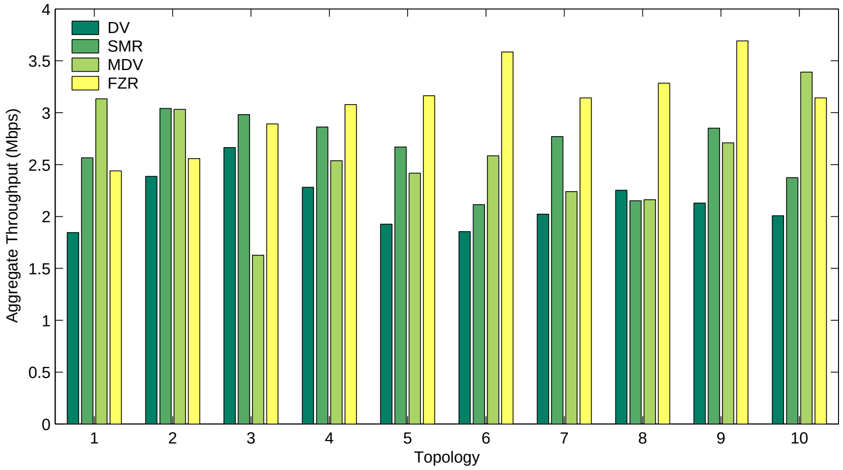

Figure 24: TCP throughput with high connectivity and low load

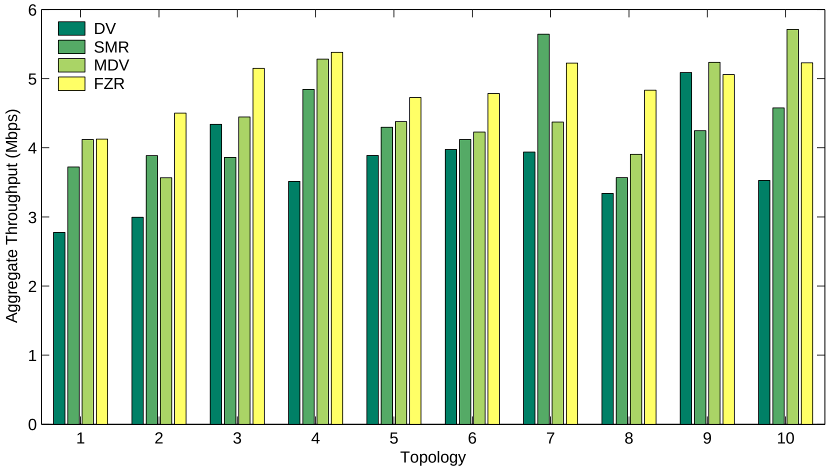

Figure 25: TCP throughput with high connectivity and medium load

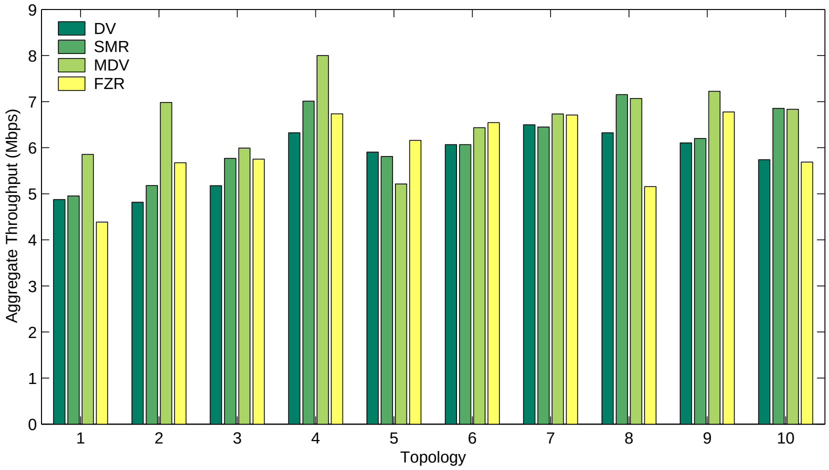

Figure 26: TCP throughput with high connectivity and high load

A.3 UDP Packet Delivery Ratio

The average UDP packet delivery ratio based on ten topologies for different traffic load and connectivity are shown in Figures 27 and 28.

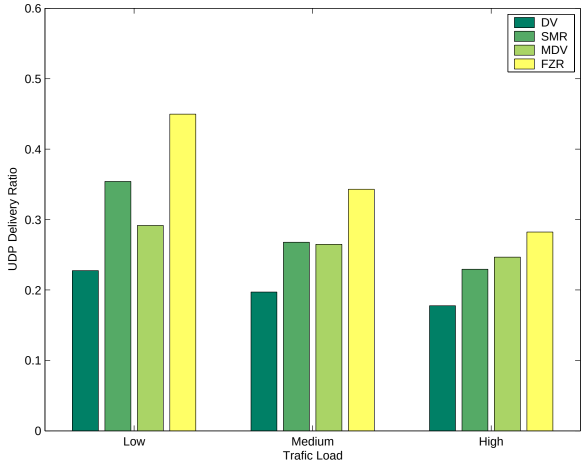

Figure 27: UDP packet delivery ratio for different traffic loads

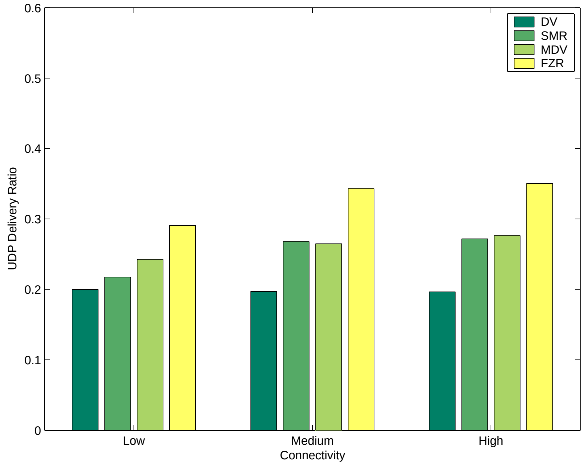

Figure 28: UDP packet delivery ratio for different connectivity

A.4 TCP End-to-end Delays

The average TCP end-to-end delays and variance based on ten topologies of medium connectivity are shown in the following figures.

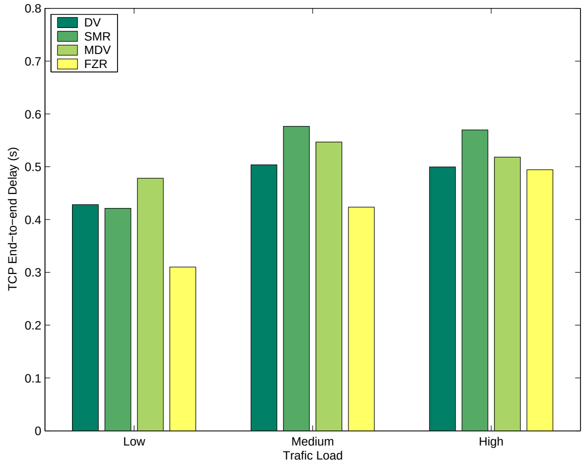

Figure 29: Average TCP end-to-end delays

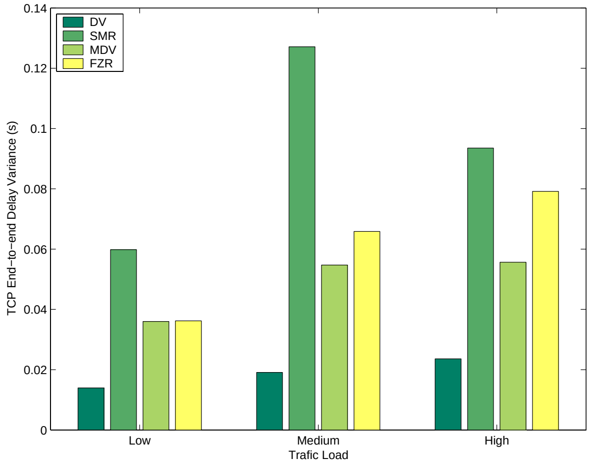

Figure 30: Variance in TCP end-to-end delays

VITA

Yibin Liang was born on February 20, 1977 in the city of Shunde in Guangdong province, China. He earned his Bachelor's degree in Electrical Engineering from Beijing Information Technology Institute in July 2000. After graduation, he worked for the Midea Corporation as a network system engineer. He later worked for the Beijing TRS Information Technology, Ltd. as a software engineer, before joining the MSEE program at the Bradley Department of Electrical and Computer Engineering at Virginia Tech in Fall 2001. He has worked in Virginia Tech's Mobile and Portable Radio Research Group (MPRG) and Laboratory for Advanced Networking (LAN). His research interests include wireless communications, digital signal processing and computer networks.

Liang is a member of the Institute of Electrical and Electronics Engineers (IEEE) and its Communications Society. He is a member of the Phi Kappa Phi honor society.