Efficient and robust routing on mobile ad hoc networks

Guo, Xiaoguang

2005

Guo, X. (2005). Efficient and robust routing on mobile ad hoc networks. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/2643>

<https://doi.org/10.32657/10356/2643>

Nanyang Technological University

Downloaded on 26 Apr 2023 16:41:09 SGT

EFFICIENT AND ROBUST ROUTING ON MOBILE AD HOC NETWORKS

EFFICIENT AND ROBUST ROUTING ON
MOBILE AD HOC NETWORKS

GUO XIAO GUANG

GUO XIAO GUANG

SCHOOL OF COMPUTER ENGINEERING
NANYANG TECHNOLOGICAL UNIVERSITY

2005

2005

Efficient and Robust Routing on Mobile Ad Hoc Networks

Guo Xiaoguang

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Master of Engineering

2005

Acknowledgements

First of all, I would like to express my deep sense of gratitude to my supervisor, Dr. Hsu Wen-Jing for his invaluable, patient and inspiring guidance during the project and my thesis writing. Without his supervision and encouragement, it would have been impossible for me to make the progress documented in the thesis and to complete the thesis. I am strongly impressed by his guidance.

I also express sincere acknowledgement to the School of Computer Engineering, Nanyang Technological University for providing the research scholarship during the two years of my master studentship.

I would like to thank Dr. Hu Yahong for helping with the proof in Chapter 4. I would also like to thank Dr. Vee Voon Yee, Mr. Ye Rong and Chen Chuanyu for their advices in my research. My appreciation also goes to all friends who have helped me in my study.

I would also like to extend gratitude towards Ms. Tan Lay Choo and Mr. Lai Chee Keong for providing the system technical support and assistance.

Last but not the least, I would like to give my special thanks to my parents and my girlfriend. The love and support of them is the powerful force for me to complete this study.

Contents

Acknowledgements	i
List of Figures	v
List of Tables	viii
Abstract	viii
1 Introduction	1
1.1 Mobile Ad Hoc Networks	1
1.1.1 Background	1
1.1.2 Characteristics of MANETs	2
1.2 Routing on Mobile Ad Hoc Networks	3
1.3 Research Focus	5
1.3.1 Power-Efficient Issue	5
1.3.2 Robustness Issue	7
1.4 Objectives	8
1.5 Thesis Organization	9
2 Literature Study	10
2.1 A Review of MANET Routing Protocols	10
2.1.1 Topology-based Routing Protocol	11
2.1.2 Hierarchical Routing Protocol	20
2.1.3 Location-Aware Routing Protocol	26
2.2 Power Saving Mechanisms for MANETs	36
2.2.1 Power Management Mechanism	37
2.2.2 Power Control Mechanism	40

2.3	Chapter Summary	44
3	A Localized Topology Control Algorithm for Power Control and Routing on MANETs	45
3.1	Geometric Model	45
3.1.1	General Assumptions	45
3.1.2	Notations	46
3.1.3	The Model	46
3.2	Relay Topology Control Algorithm	56
3.2.1	Search for Relay Candidates	56
3.2.2	Relay Topology Control Algorithm for Power Control .	57
3.2.3	Relay Topology Control Algorithm for Power-Efficient Routing	64
3.3	Simulation Results	69
3.3.1	Simulation Environment	69
3.3.2	Simulation Results and Discussion	70
3.4	Chapter Summary	77
4	A Geographic Routing Protocol Based on Approximate Location Information for Mobile Ad Hoc Networks	79
4.1	FLIRP Algorithms	80
4.1.1	Network Partition and Addressing	80
4.1.2	Fisheye-like Location Information	82
4.1.3	FLI Setup and Update	84
4.1.4	Geographic Routing with Stepwise Refinement	86
4.2	Performance Analysis	87
4.2.1	Cost of FLI Maintenance	88
4.2.2	Performance of FLIRP Routing Path Length	90
4.3	Simulation Results and Evaluation	92
4.3.1	Simulation Setup	92
4.3.2	Protocol Traffic	94

4.3.3	Consistency of Location Information	98
4.3.4	Routing Performance	99
4.4	Chapter Summary	103
5	Conclusions and Future Work	104
5.1	Summary and Results	104
5.2	Discussion and Future Research	105
5.2.1	Extensions of RTC	105
5.2.2	Range Query Service Based on FLIRP	106
5.3	Concluding Remark	107
	Appendix	108
	Bibliography and References	109

List of Figures

1.1	Network architectures of two kinds of wireless networks . . .	2
1.2	The <i>topology control</i> approach	7
2.1	Classification of MANET routing protocols	11
2.2	The scope of fisheye	14
2.3	CGSR: Routing from node x to node y	22
2.4	HSR: an example of multilevel clustering	24
2.5	LAR: limited flooding of route request	27
2.6	Example for the expected region in DREAM	28
2.7	Greedy forwarding in GPSR	30
2.8	Location management in SLURP	33
2.9	Power saving mechanisms on MANETs	37
2.10	Node u find a neighbor v in the cone of α	42
3.1	Relay regions for node s in case of $k = 2$ and $k > 2$	47
3.2	Enclosure region $ER(s)$ of node s ($k > 2$)	50
3.3	Relay polygons formed in two general cases	51
3.4	Two kinds of relay polygons	52
3.5	The relay polygon of node x is a simple polygon	53
3.6	The relay polygon of node x is convex	53
3.7	Pairs of nodes in $\{A, B, C, D\}$ are neighbors of each other; . .	54
3.8	$\{A, B\}$ and $\{C, D\}$ are neighbors, but the other pairs of nodes in $\{A, B, C, D\}$ may be not neighbors;	55

3.9	Only $\{A, B\}$ and $\{C, D\}$ are neighbors, and all the other pairs of nodes in $\{A, B, C, D\}$ are not neighbors;	56
3.10	The pseudocode of search for relay candidates algorithm . . .	57
3.11	An example of RTC	58
3.12	The pseudocode of RTC algorithm	59
3.13	Connectivity analysis	62
3.14	“Blind Corner” of node i	64
3.15	Destination node t is only in $RR(s, r_1)$	66
3.16	Destination node t is in $RR(s, r_1) \cap RR(s, r_2)$	66
3.17	Failure of greedy forwarding ($k = 2$)	67
3.18	Greedy Power-Aware Routing algorithm	68
3.19	Network connectivity vs. Transmission radius	71
3.20	Network topologies resulting from various strategies	73
3.21	Average radius in transmission (The unit of node density is $10^{-4}/m^2$, which is same in the following)	74
3.22	Total number of links in Unit Disk Graph and Relay Graph .	74
3.23	Average node degree in Unit Disk Graph and Relay Graph . .	75
3.24	Power consumption in transmission and routing	76
3.25	Hop count in different cases	77
4.1	Network partitioning and addressing	81
4.2	Relationship between the size of a unit region and R_{max}	81
4.3	Fisheye-like location information	83
4.4	Central regions and coordinators	85
4.5	An example of stepwise refining geographic routing	86
4.6	Analysis of the cost of FLI maintenance	88
4.7	Nodes S and D in a same unit region.	91
4.8	The LCA of S and D is level- $(k + 1)$	91

4.9	Probabilities of connectivity when randomly place 2 to 6 nodes in a square of size from 150 to 500 meters on each side.	93
4.10	The average number of update packets per node per second does not vary much over network size	95
4.11	The average number of update packets per node per second of SLURP and FLIRP as functions of node speed. The network configuration is 400 nodes in a 2000×2000 area.	95
4.12	The average number of query packets per node per second as a function of the total number of nodes.	96
4.13	The average number of query packets per node per second as a function of node speed. The network configuration is 400 nodes in the 2000×2000 area.	97
4.14	Correctness ratio of location information for different network sizes.	98
4.15	Correctness ratio of location information for different orders of squares	99
4.16	FLIRP packets delivery rate vs. number of nodes.	100
4.17	The packets delivery rates of FLIRP and SLURP as functions of nodes speed. The network configuration is 400 nodes in a 2000×2000 area.	100
4.18	Routing latency as a function of the number of nodes. All the nodes are static.	101
4.19	Average routing path length as a function of the Euclidean dis- tance between source and destination nodes.	102
4.20	Average routing path lengths between FLIRP and GPSR. . .	102
5.1	An example of range query	106
5.2	Range query based on FLIRP	107

List of Tables

1.1	Power consumption comparison among different wireless cards	6
2.1	Comparisons of the characteristics of flat routing protocols . .	19
2.2	Comparisons of the characteristics of hierarchical routing protocols	26
3.1	Main parameters of simulation model	70
3.2	Thresholds of transmission radius in different node density . .	71
4.1	Network configurations with approximate density	93

Abstract

With the advance of wireless communication technologies and the increasing popularization of wireless devices, mobile ad hoc networks have attracted substantial interest in recent years. Due to the frequent changes in the network topology and the constrained power supply, routing in Mobile Ad Hoc Networks (MANETs) is difficult compared with the traditional routing protocols. Presently existing routing schemes are not well adapted to the mobile environment in terms of energy efficiency and routing robustness.

In this thesis, we present schemes for energy-efficient routing and for robust routing.

For energy-efficiency, a localized topology control scheme is proposed to achieve energy-efficient routing on mobile ad hoc networks. The distributed algorithm is simple yet efficient, and hence suitable for mobile ad hoc networks. With this algorithm, nodes can dynamically adjust their transmission power to construct an energy-efficient topology on the underlying ad hoc network. The power control approach not only reduces the power consumption in transmission, but also maintains the overall network connectivity so that the communication routes between nodes will not be disconnected.

For routing robustness, we present a geographic routing protocol inspired by the fisheye technique. A node maintains accurate location information of nearby nodes and approximate location information of faraway nodes. The further away a node x is with respect to a given node y , the less information about x will be maintained on y . The routing protocol is based on the approximate location information and greedy forwarding. Packet forwarding decisions can be made independently for each node. This protocol is adaptive and robust for the mobile environments without high update messages overhead.

This master thesis documents the project, the issues involved, our approach toward routing as well as the results obtained. We believe that the experience and results gained in this research could be useful for further work.

Chapter 1

Introduction

1.1 Mobile Ad Hoc Networks

1.1.1 Background

Wireless computer networks emerged in 1970's and have gained popularity since then. In the past decade, wireless networks have been adapted to support mobility of the nodes. Mobile nodes such as hand phones, PDAs, notebooks are now easily affordable and widely used in both personal and business life. With their increasing popularity and availability, there is a natural desire to share information and communicate among the mobile users.

However, in most existing wireless networks, the mobility is still restricted as they require the connectivity to the core network, hence they are not offering the true mobility. As shown in Figure 1.1(a), communication between two mobile nodes is done by connecting to the nearest base station(s), which are interconnected to form a wired backbone network. Due to mobility, the mobile node may move out of the range of one base station and into the range of another. A “hand-off” mechanism takes place from the former base station to the latter, which enables the node to communicate unceasingly with the other nodes.

CHAPTER 1. INTRODUCTION

Figure 1.1: Network architectures of two kinds of wireless networks

To further extend mobility, a new kind of wireless network called Mobile Ad Hoc Network (or MANET for short) came into being where the nodes can autonomously form the network themselves. In other words, MANETs interconnections are dynamically formed and changed as the nodes move around. As shown in Figure 1.1(b), MANET has no centralized administration, each node functions as a router, maintaining and discovering routes to the other nodes. Due to the limited transmission range of wireless network interfaces, multiple hops may be needed to exchange data between nodes in the network. MANET will play an important role in situations where temporary network connectivity is needed, such as military operations, conferences, emergency services [50], etc. These applications serve as a motivation to design new and effective routing strategies that work in constrained environments.

1.1.2 Characteristics of MANETs

Some of the prominent characteristics of MANET can be summed up as follows:

- *Dynamic Topology:* Nodes are free to move and connected to the network dynamically. As a result, the network topology will change rather unpredictably with time. Frequent movements may result in network partitioning and constraining network connectivity.

- *Resource Constraints:* Compared to the fixed networks, mobile ad hoc networks are also constrained in bandwidth and power supply.

Bandwidth Constraints: Although the bandwidth of mobile ad hoc networks has increased steadily over the past decade, it is still much less than the bandwidth available to the fixed networks. MANETs often operate in heterogeneous wireless environments with significantly varying bandwidth-delay characteristics. Besides, wireless links are affected by many error sources like fading, noise, interferences and path losses [57].

Power Constraints: The mobile nodes rely on batteries power or other portable power supplies for operations. The power supply constraint limits the transmission range and signal strength of the nodes. The protocols with power saving consideration will be more suitable for MANETs.

- *Limited Physical Security:* The security service in MANETs faces lots of challenges, such as wireless links, power constraints, and relatively poor physical protection of nodes in a hostile environment.

1.2 Routing on Mobile Ad Hoc Networks

Because of the unique features of MANET described in the previous section, a desirable MANET routing protocol should possess the following properties [11]:

- *Distributed operation:* A MANET routing protocol should be distributed in order to increase reliability. It is unacceptable to have a centralized routing protocol where all nodes are mobile. Each node should be intelligent enough to make routing decisions using other collaborative nodes.

- *Loop-freedom*: It is desirable to offer some assurance in packet delivery and avoid problems like packets looping in the network for an arbitrary period of time. Some protocols, such as DSDV [49], AODV [51], have been proposed to alleviate the packet looping problem.
- *Power efficiency*: Power efficiency is a very important consideration for the battery-powered mobile devices, such as handphones, PDAs, and notebooks. The disappearance of nodes due to shortage of power may cause the network to be partitioned and resulting in inaccessible nodes.
- *Unidirectional link support*: Wireless medium may cause a wireless link to be built up only in unidirection due to some physical factors, such as power constraints, and may not be possible to communicate bidirectionally. Thus, it is desirable to support the unidirectional links when designing a routing protocol for MANETs.
- *Security*: While security concerns exist within wired infrastructures and routing protocols as well, maintaining the physical security of the transmission media is harder in practice with MANETs.
- *Quality of Service*: A routing protocol should be aware of some QoS metrics, such as delay and throughput, of a source-destination pair connection. However, lacking a fixed infrastructure in MANETs makes QoS a challenging problem.
- *Scalability*: Some of the envisioned networks (e.g. mobile military networks) may be relatively large (e.g. ten or hundreds of nodes). Of course, the need for scalability is not unique to MANETs.

1.3 Research Focus

The limited power supply of mobile devices is a critical problem for MANETs, and frequent changes of nodes' location also pose another important challenge in MANETs.

1.3.1 Power-Efficient Issue

The studies in [62,67] show that significant power is consumed by the hardware components of a wireless devices, such as microprocessor, display devices, storage devices and wireless network interface card. Consequently, many power efficient designs are considered in the hardware components and their corresponding software strategies on the wireless devices. [38] provides a good survey on the software strategies of power management on portable computers. [31] presents some application-driven power management strategies for mobile communication. Incorporating low-power strategies into the design of network protocols may also result in significant additional power savings. In our research, we focus on the power efficient designs in terms of network protocols used for data communications.

A typical mobile device interface may operate in four modes: *transmit*, *receive*, *idle* and *sleep*. These modes represent different operations of one mobile device.

A wireless device consumes different amount of energy in these four modes. Table 1.1 summarizes the power consumption in different modes of some typical wireless cards. Here, the data of two kinds of WaveLAN PCMCIA cards is from the measurement results of [14], and data for Proxim RangeLAN2 card and Smart Spread is from their specification in [32,65].

From Table 1.1, we see that, the maximum power is consumed in the *transmit* mode, and the least in the *sleep* mode. Intuitively, this result provides two direct approaches to reduce the power consumption of wireless devices. One is to reduce the power consumption in the *transmit* mode; the

CHAPTER 1. INTRODUCTION

Wireless Card	Power Supply	Transmit	Receive	Idle	Sleep
Proxim RangeLAN2-7410	5V	265mA	130mA	N/A	2mA
Smart Spread	5V	150mA	80mA	N/A	5mA
WaveLAN (11Mbps)	4.74V	284mA	190mA	156mA	10mA
WaveLAN (2Mbps)	4.74V	280mA	204mA	178mA	14mA

Table 1.1: Power consumption comparison among different wireless cards

other one is to switch the wireless interface to *sleep* mode when no packet is detected.

Most existing works addressing the problem of power-efficient design are mainly in MAC layer and network layer. In the power-aware MAC protocols, nodes can adjust their transmission power, or turn-off their network interface to achieve power conservation. The power-aware routing protocols use power consumption as routing metric and find a route between source and destination which consumes the least power instead of the smallest number of hops. In addition to the layers in the existing network designs, like network layer and MAC layer, we introduce an approach called *topology control* to provide a strategy to integrate the different power-efficient strategies in different layers, especially in MAC and network layers. As shown in Figure 1.2, *topology control*

Figure 1.2: The *topology control* approach

works under the routing layer and above the MAC layer. It constructs a power-efficient topology on the underlining ad hoc network by adjusting the transmission range or shutting down idle nodes. *Topology control* also ensures that the power saving strategy does not affect the data communication to a great extent. For example, each node may adjust its transmission power to conserve power. However, the power control decisions on all nodes need to collectively guarantee global network connectivity so that communication routes between nodes will not be disconnected due to the power control.

Most existing solutions for power-efficient broadcast communications require knowledge of whole network topology to make decisions. Therefore, topology changes (mobility of nodes, or change in their activity status) must be propagated throughout the network. This approach entails high overhead for ad hoc networks. It is better to let each node decide its own behavior based only on the information from its neighboring nodes (on the knowledge of its 1-hop or 2-hops neighbors). Such protocols are said to be *localized* (or *distributed*) protocols.

1.3.2 Robustness Issue

Recent research has shown that routing in MANETs can be better achieved by using location information provided via GPS devices. However, the use of nodes' location information for routing poses evident problems in terms

of robustness. The accuracy of location information of the destination is an important problem to consider.

Designing location updates schemes to enable efficient routing in mobile ad hoc network appears to be more difficult than routing itself (see a recent survey [63]). For small networks, in the absence of any useful information about destination location, the following simple strategy can be applied. If message is reasonably “short”, it can be flooded by using an optimal broadcasting scheme [48, 53, 64]. If message is relatively “long”, then destination search (or route discovery [4]) can be initiated, which is a task of broadcasting short search message. Destination then reports back to source by routing a short message containing its position. The source then is able to route full message toward accurate position of destination.

However, in large networks, algorithms that assume that the location of destination is “reasonably” accurate will not be able to deal with eventual location deviation, and impose high mobility tracking overhead. Therefore robust routing schemes that adapt to the mobile environments are needed.

1.4 Objectives

This project concentrates on designing efficient and robust routing protocols for MANETs.

To address power-efficient issue, we present a localized topology control scheme to achieve power control and power-efficient routing. The main idea of the topology control scheme is that, nodes can reduce their transmission power to maintain the immediate neighbors more frequently than they do about the farther neighbors. The power control decisions are made through these immediate neighbors, which can construct a power-efficient topology on the underlining ad hoc network. Our algorithm not only reduces the transmission power, but also guarantees the global network connectivity so that

the communication routes between nodes will not be disconnected due to the power control. Moreover, the routing is operated on the topology so that a power-efficient end-to-end route can be formed locally.

To address robustness issue, we present a geographic routing protocol inspired by the fisheye technique [29]. Each node maintains accurate location information of nearby nodes, and approximate location information of faraway nodes. For each node, the accuracy levels of the other nodes' location information depend on their relative location in a hierarchical grid partition of the network. The routing protocol is based on the approximate location information and greedy forwarding [27]. It not only reduces the update messages overhead, but also is adaptive to the mobile environments.

1.5 Thesis Organization

The rest of this thesis is organized as follows:

- Chapter 2 presents a comprehensive literature review on MANET routing protocols and location services.
- Chapter 3 presents a localized topology control algorithm which not only conserves the transmission power for topology maintenance, but also reduces interferences among the nodes. A power-aware routing protocol based on the topology control algorithm is also proposed to achieve the power-efficient routing in the network.
- Chapter 4 presents a robust geographic routing scheme that not only reduces the update messages overhead, but also is adaptive to the mobile environments.
- Chapter 5 summarizes and discusses the work done in this thesis. Recommendations for future work are also presented.

Chapter 2

Literature Study

In Section 2.1, we classify and describe some existing routing protocols for mobile ad hoc network. In Section 2.2, we review existing power saving mechanisms for mobile ad hoc networks.

2.1 A Review of MANET Routing Protocols

We classify the existing routing protocols into two broad categories: topology-based routing and location-aware routing, as shown in Figure 2.1. In topology-based routing protocols, nodes use the information of links and network topology to make the routing decisions. While in location-aware routing protocols, nodes can determine their own location via the support of GPS devices or some other types of location services. The routing decision at each node is based on the location of its destination and neighboring nodes, instead of topology information of the network in flat or hierarchical routing protocols. The topology-based routing protocols can be further classified into two classes: flat routing and hierarchical routing, according to the network topology structure underlying the routing protocols. This taxonomy follows that of [20, 55].

Here we select a few well-known routing protocols, which either have been extensively evaluated by other researchers, or can briefly demonstrate each routing mechanism.

Figure 2.1: Classification of MANET routing protocols

2.1.1 Topology-based Routing Protocol

In this section, we will describe the main characteristics of the topology-based routing protocols, and compare them in terms of some conventional features, such as routing overhead, latency, scalability etc.

Flat Routing Protocol

Flat routing protocols adopt a flat addressing scheme, in which each node participating in routing plays an equal role. According to the routing strategy, these protocols can also be further classified into (A) proactive routing protocols and (B) reactive routing protocols. In the following, we will discuss them respectively

(A) *Proactive Routing Protocols*: The proactive routing protocols are also called *table-driven* routing protocols in the literature [59]. In these routing protocols, nodes are required to maintain consistent, up-to-date routing information about the network even if they are not currently used. The routing information is stored in one or more tables and refreshed by propagating updates throughout the network in order to maintain a consistent view of the network. Many proactive routing protocols have been proposed, e.g. [2, 8, 15, 22, 29, 41, 43, 49]. In the following, we will discuss two typical proactive routing protocols, namely DSDV [49] and FSR [29] in this section.

- ***Destination Sequenced Distance Vector Routing*** — Destination Sequenced Distance Vector Routing (DSDV) [49] is a table-driven protocol based on the classical Bellman-Ford routing mechanism [25]. The main improvement of DSDV is to guarantee loop-free routes in contrast to traditional distance-vector protocols.

DSDV basically is a distance-vector routing protocol where each node has a routing table that stores the next hop and the number of hops to all reachable destinations. Like distance-vector, DSDV requires that each node periodically broadcasts routing updates. In order to make it better suited for ad hoc networks, DSDV adopts some adjustments which consist of triggered updates that will take care of topology changes in the time between broadcasts. To reduce the amount of information in these packets, two types of update messages are defined: full and incremental. The full broadcast carries all available routing information and the incremental broadcast only carries the information that has changed since the last broadcast.

A new route broadcast contains the address of the destination, the number of hops to reach the destination, and the sequence number of the information received regarding the destination, as well as a new sequence number unique to the broadcast [49]. The route labeled with the most recent sequence number is always used. In the event that two updates have the same sequence number, the route with the smaller metric is used in order to optimize (shorten) the path. Nodes keep track of the setup time of routes, or the weighted average time that routes to a destination. By delaying the broadcast of a routing update by the length of the setup time, nodes can reduce network traffic and optimize routes by eliminating those broadcasts that would occur if a better route is discovered in the very near future.

Because DSDV is dependent on periodic broadcasts, it requires some time to converge before a route can be used. This convergence time is less critical

in a static wired network, where topology changes are infrequent. However, in an ad hoc network, the topology is expected to be very dynamic, thus a slow convergence of routes causes packets to be dropped when nodes move around. Periodic and triggered broadcasts also add a large amount of traffic to the network, especially in a high mobility network.

- ***Fisheye State Routing*** — Fisheye State Routing (FSR) [29] is an improvement of Global State Routing (GSR) [8], which is a link-state routing protocol that is very similar to DSDV and improves the link-state routing by avoiding flooding of routing messages. The update messages in GSR waste a considerable amount of network bandwidth. In FSR, each update message does not contain information about all nodes as in GSR. Instead, it exchanges information about closer nodes more frequently than it does about farther nodes thus reducing the update message size. So each node gets accurate information about neighbors and the level of detail and accuracy of information decreases with the distance from the node concerned. The scope is defined in terms of the nodes that can be reached in a certain number of hops, as shown in Figure 2.2. The center node has the most accurate information about all nodes in the inner circle and less accurate information about nodes in between the 2nd circle and the innermost circle and so on. Even though a node does not have accurate information about distant nodes, the packets are routed correctly because the route information becomes more and more accurate as the packet moves closer to the destination. FSR scales well to large networks as the overhead is controlled in this scheme.

FSR scales well to large networks, by keeping link state exchange overhead low without compromising route computation accuracy when the destination is near. By retaining a routing entry for each destination, FSR avoids the extra work of finding the destination (as in on-demand routing) and thus maintains low single packet transmission latency. As mobility increases, routes to remote

Figure 2.2: The scope of fisheye

destination become less accurate. However, when a packet approaches its destination, it finds increasingly more accurate routing instructions as it enters sectors with a higher refresh rate.

(B) *Reactive Routing Protocols*: Reactive routing protocols are also called *on-demand* routing protocols. Their routing strategies are different from the proactive routing protocols. Nodes are not required to update their routing information periodically. A routing path is built up only when there is a routing task awaiting. When a node requires to forward packets to a destination, it starts a route discovery process within the network until a route is found or all possible routes have been examined. Once a route has been established, it is maintained by a route maintenance process until the route is no longer desired or the destination is unreachable. Some on-demand routing protocols have been proposed in recent years, such as Dynamic Source Routing (DSR) [24], Ad-Hoc On Demand Distance Vector Routing (AODV) [51], Associativity-Based Routing (ABR) [66], Temporally Ordered Routing Algorithms (TORA) [44] and Signal Stability Routing (SSA) [13]. In the following, we will discuss DSR and AODV to demonstrate the on-demand routing mechanism, since they have been extensively evaluated in the MANET literature and considered by the Internet Engineering Task Force (IETF) MANET Working Group as the leading candidates for standardization.

• ***Dynamic Source Routing*** — Dynamic Source Routing protocol (DSR) [24] is a source-routed routing protocol. DSR allows nodes to dynamically discover a source route. The complete, ordered list of nodes in the route is carried in its header. In DSR, nodes are not required to broadcast routing messages periodically. Instead, they maintain route caches to contain the source routes that they are aware of, and update entries in the route caches when they find new routes. The protocol consists of two processes, namely *route discovery* and *route maintenance*, which work together to allow nodes to discover and maintain source routes to destinations in the network.

DSR requires no periodic packets of any kind at any level within the network. For example, DSR does not use any periodic routing advertisements, link status sensing, or neighbor detection packets, and does not rely on these functions from any underlying protocols in the network. So the network topology changes without affecting current routes are ignored, and no reactions are caused from the protocol. In response to a single route discovery, a node may learn and cache multiple routes to any destination. If the node does not have such a route, then it initiates the *route discovery* process by broadcasting a route request packet. The route request packet contains the address of the source and destination nodes, and a unique identification number. Each intermediate node checks whether it knows of a route to the destination. If it does not, it appends its address to the route record of the packet and forwards the packet to its neighbors.

To limit the number of route requests, a node processes the route request packet only if it has not already received the packet and its address is not contained in the route record of the packet. A route reply is generated when either the destination or an intermediate node with current information about the destination receives the route request packet. A route request packet reaching such a node contains the sequence of hops taken from the source

node to this node in its route record. Route reply can be sent by reversing the path only if links are guaranteed to be bi-directional. Otherwise, the *route discovery* is initiated and route reply is piggybacked on it.

In *route maintenance* process, two types of packets are used: route error packet and acknowledgement. When a node encounters a fatal transmission problem at its data link layer, it generates a route error packet. A node receiving a route error packet removes the associated hop from its route cache. All the routes containing the broken link are truncated at that point. The acknowledgement is used to verify the correct operation of the route links. It also includes passive acknowledgements in which a node hears the next hop forwarding the packet along the route. As the reply may result in reply storm, it is delayed for a random period of time.

In DSR, because of source routing, nodes need not maintain a complete view of the network in order to route the packets they forward. There is also no need for periodic routing advertisement messages, which leads to reduced routing overhead. This protocol has the advantage of learning routes by scanning for information in packets that are received. For example, node *A* routes from *A* to *C* through *B* means that *A* learns the route to *C*, but also that it will learn the route to *B*. The source route also means that *B* learns the route to *A* and *C* and that *C* learns the route to *A* and *B*. This form of active learning proves to be very useful and reduces overhead in the network. However, each packet carries a burden of the source route of the packet. This overhead increases when the packet has to go through more hops to reach the destination. This does not scale well to large networks when packets have to traverse through many hops to reach a destination. In addition, nodes only hold one route to their destination. When the route becomes invalid (due to topology change), a new route must be discovered. Therefore, DSR does not scale well to mobility as overhead (due to route requests) and latency (due to finding new routes) increase as mobility increases.

- ***Ad Hoc On-Demand Distance Vector*** — Ad hoc On-demand Distance Vector Routing (AODV) [51] is an improvement of DSDV [49] and DSR [24] protocols. AODV minimizes the number of broadcasts by creating routes on-demand as opposed to DSDV that maintains the list of all the routes. Like DSDV, it uses the sequence number to avoid the loop formation problem.

AODV retains the desirable feature of DSR that routes are discovered on demand. The route requests (RREQ) are forwarded in a manner similar to DSR. It reduces the need for the wide broadcasts by localizing the propagation of changes in the network. If the link status does not affect the ongoing communication, no broadcast occurs. Only when a source node tries to use the route with broken links, a global propagation is incurred to inform the nodes using the route with broken links. The route is then expired and removed from the corresponding nodes. However, the construction of routing table in AODV is different from DSR. During route request traversal, it sets up a reverse path towards the source node. When the destination replies, the route reply packets routes along the reverse path. An intermediate node may also send a RREQ packet when a more recent path is determined by the sequence number in this node. A new route request for a destination is assigned a higher destination sequence number than the known value. A routing table entry maintaining a reverse path is removed if a timeout interval is long enough to allow RREQ to come back in the worst case. A routing table entry maintaining a forward path is removed if it is not used in an active route timeout interval. If no packet is sent by using a particular routing table entry, the entry will be removed from the routing table.

When a link breaks, all the active neighbors are informed. The link failures are propagated using route error messages. When the source node receives the RREQ, it initiates a new route discovery. The connections with its neighbors

must be ensured by either using hello message periodically or getting the feedback from the lower layers. The absence of hello messages are used as an indication of link failure. Also, the failure to receive several MAC-level acknowledgement may be used as an indication of link failure. Since the route request broadcast may result in broadcast storm problem, the source node uses expanding ring search method by setting a Time-To-Live (TTL) value. TTL specifies the diameter of the subsection of the network in which RREQ is allowed to spread. If the route is not found within the timeout, a new TTL and timeout are set to search again. This is repeated till a threshold value of the TTL, the TTL is set equal to the network diameter and repeated before aborting the process.

Compared to the classical routing protocols like distance-vector and link-state protocols, AODV has greatly reduced the number of routing messages in the network by using a reactive approach. AODV only supports one route for each destination. This causes a node to reinitiate a route request query when its route breaks. This does not scale well as the number of route requests increase as mobility increases (topology changes in the network and links break). AODV also does not support unidirectional links. When a node receives a RREQ, it will set up a reverse route to the source by using the node that forwarded the RREQ as the next hop. This means that the route reply is unicasted back the same way the route request used.

Comparisons of Flat Routing Protocols: We summarize the main characteristics of these flat routing protocols in Table 2.1.

Proactive routing protocols require nodes in the networks to maintain consistent routing information of full networks frequently. Under light traffic and low mobility, this strategy leads to efficient routes and less latency for routing. When the number of nodes or mobility increases, however, the overhead of networks will increase respectively; moreover, a significant part of the available

CHAPTER 2. LITERATURE STUDY

Parameter	DSDV	WRP	FSR	DSR	AODV
Routing strategy	Proactive	Proactive	Proactive	Reac- tive	Reac- tive
Protocol type	Distance- vector	Distance- vector	Link-state	Distance- vector	Distance- vector
Routing metric	Shortest path	Shortest path	Shortest path	Shortest path	Shortest path
Frequency of updates	Periodi- cally and as needed	Periodi- cally and as needed	Periodi- cally	As needed	As needed
Use sequence numbers	Yes	Yes	Yes	No	Yes
Loop-free	Yes	Yes	Yes	Yes	Yes
Multicast capability	No	No	No	No	Yes

Table 2.1: Comparisons of the characteristics of flat routing protocols

bandwidth will be occupied and more battery power will be consumed. Thus the scalability of the network is limited with the proactive routing protocols. In contrast, reactive routing protocols react only when a communication is needed. Thus, the routing overhead is only incurred by the usage of discovery and maintenance of the routes. Under light traffic and low mobility, nodes occupy low bandwidth and consume little battery power. The networks can scale well to large networks. However, under heavy traffic and high mobility, since a flooding approach is used to forward route requests and to maintain the routing paths in reactive routing protocols, routing control overhead tends to grow significantly [12] and end-to-end latency of the route is also increased.

Proactive routing protocols have the higher routing efficiency under discontinuous traffic mode and high mobility, since the path to all the desired destinations are available in advance by the periodic beaconing. So no additional routing overhead is generated for finding a new destination or route. Nodes consume constant bandwidth and power regardless of the traffic conditions. However, the proactive routing protocols are not suitable for some resource critical applications, such as sensor networks, because of this feature.

More detailed performance comparisons of these flat routing protocols can be found in [3–5, 12, 20, 23, 59].

2.1.2 Hierarchical Routing Protocol

In the small mobile ad hoc networks, flat routing protocols are acceptable and operate well in the condition of light traffic and low mobility. When the network size increases, the flat routing protocols become infeasible because of heavy traffic. In order to solve this problem, some hierarchical routing protocols are proposed to provide scalable and efficient solutions. The basic idea of hierarchical routing is to organize the nodes in groups and then assign nodes different functionalities inside and outside a group. Each node in the hierarchical structure is included in a part of the network instead of the whole. So

both routing table size and update packet size are reduced, and thus control overhead is also reduced. In the literature, two ways to build hierarchy have been proposed. One is to group nodes that are close in geographic location into explicit clusters. In each cluster, there is a leading node (clusterhead) to communicate with other clusters on behalf of this cluster. The other one is to have implicit hierarchy, in which each node has a local scope and uses different strategies inside and outside the scope. Communications go through the overlapping scopes. This strategy is more flexible and can achieve more efficient overall routing performance. The following section discusses some typical hierarchical MANET routing protocols: CGSR [9], ZRP [19] and HSR [47].

- ***Clusterhead Gateway Switch Routing*** — Clusterhead Gateway Switch Routing (CGSR) [9, 10] is a typical cluster-based hierarchical routing protocol. CGSR uses as basis the DSDV routing algorithm described in the previous section. The protocol differs with DSDV in the network addressing and organization scheme employed. Instead of a “flat” network, CGSR is a clustered multihop mobile wireless network. By using a typical clustering algorithm, Least Clusterhead Change (LCC), mobile nodes are aggregated into clusters and a clusterhead is elected for each cluster. All nodes that are in the communication range of the clusterhead belong to a cluster. A gateway node is a node that is in the communication range of two or more clusterheads. It routes traffic from source to destination using a hierarchical clusterhead-to-gateway routing approach.

Figure 2.3 illustrates an example of this routing scheme. Using CGSR, each node must keep a “cluster member table” where it stores the destination clusterhead for each mobile node in the network. These cluster member tables are broadcast by each node periodically using the DSDV algorithm. Nodes update their cluster member tables on receipt of such a table from a neighbor. A packet sent by a node is first routed to its clusterhead, and then the packet

Figure 2.3: CGSR: Routing from node x to node y

is routed from the cluster head to a gateway to another clusterhead; the same forwarding process is repeated until the clusterhead of the destination node is reached. The packet is then transmitted to its destination.

In addition to the cluster member table, each node must also maintain a routing table which is used to determine the next hop in order to reach the destination. On receiving a packet, a node will consult its cluster member table and routing table to determine the nearest clusterhead along the route to the destination. Then, the node will check its routing table to determine the next hop used to reach the selected clusterhead.

CGSR adopts a framework among clusters for code separation, channel access, routing, and bandwidth by having a cluster head controlling a group of ad hoc nodes. This is a good approach when dealing with large ad hoc networks. It is very scalable because it uses the clustering approach that limits the number of messages that need to be sent. However, the cluster design is vulnerable to point failures. If a clusterhead goes down, then routing in the entire cluster is disturbed. Frequent clusterhead changes can adversely affect routing protocol performance since nodes are busy in clusterhead selection rather than packet relaying. In addition, routes between nodes in different clusters do not result in minimum hop paths.

- **Zone Routing Protocol** — The Zone Routing Protocol (ZRP) [18,19] is a hybrid of both the proactive and reactive schemes. For a node, the set

of nodes within a predefined radius r , form a zone. Routing inside the zone follows the proactive approach, while the routing across the zones follows the reactive approach.

Within the zone, the proactive protocol DSDV is used. Whenever a link changes, nodes within the zone will be informed. In this way, nodes always know how to reach a node in the same zone. For interzone routing, it is suggested to use the modified DSR protocol as follows. When a node needs a route to a node outside its zone, it performs a *bordercasting* by sending a route request to each node on the “border” of this zone. On receiving such a packet at a border node, it first checks its intrazone routing table for existence of a route to the requested destination node. If so, a route reply can be sent; otherwise, it performs another bordercasting in its zone. This is repeated until a route is found.

Although a modified source routing is used for interzone routing, the routing path only contains the border nodes that have to be traversed by the route request; the nodes across border nodes are not recorded in the path. Thus, when a data packet needs to be forwarded, the node will check its intrazone routing table to find the path to the next border node.

If there is a change in the state of a link, the information is dispersed within the zone. If a route failure occurs, the source node is informed as in DSR.

- ***Hierarchical State Routing*** — Hierarchical State Routing (HSR) [47] is a multilevel clustering-based LS routing protocol. It maintains a logical hierarchical topology by using the clustering scheme recursively. In HSR, a clusterhead is elected similar to other cluster-based algorithms. The clusterheads organize themselves into clusters again, elect a clusterhead and so on, to form the *hierarchy*. The nodes of a physical cluster broadcast their links information to each other. The clusterhead summarizes its cluster’s information and sends it to neighboring clusterhead via gateway as it is in CGSR [9]. As

Figure 2.4: HSR: an example of multilevel clustering

shown in Figure 2.4, these clusterheads are members of the cluster on a higher level and exchange their links information among that level and so on, up to the highest level. A node at each level floods to its lower level the information that it obtains after the algorithm has run at that level. So the lower level has a hierarchical topology information. Each node is identified by a hierarchical address, which may be so assigned that it reflects the clustering order from top to bottom. A gateway can be reached from the root via more than one path, so the gateway can have more than one hierarchical address. Thus, the hierarchical address is enough to ensure delivery from anywhere in the network to the node.

Further, the nodes in the network are also partitioned into logical sub-networks and each node is assigned a logical address $\langle subnet, node \rangle$. Each subnetwork has a location management server (LMS). All the nodes of the sub-network register their logical address with the LMS. The LMS advertise these nodes' hierarchical addresses to the top level and the information is also sent to all LMS at lower layers. The transport layer sends a packet to the network layer with the logical address of the destination. The network layer finds the

hierarchical address of the destination from its LMS and then sends the packet to it. The destination LMS forwards the packet to the destination. Once the source and destination know each other's hierarchical addresses, they can bypass the LMS and communicate directly. Since logical/hierarchical address is used for routing, it is adaptable to network changes. Thus, the hierarchy reduces the overhead associated with the link state algorithm and reduces the entries in the routing table.

However, the protocol requires the location management servers with up-to-date information about the addresses. The failures of a LMS can affect the protocol. Thus, it is dependent upon a central entity. It requires the extra storage and computation to maintain the location database as well.

Comparisons of Hierarchical Routing Protocols: The key features of these hierarchical routing protocols are summarized in Table 2.2.

In hierarchical routing protocols, nodes are only required to maintain the information of nodes in its cluster or local scope, instead of the whole network in flat routing protocols. This can significantly reduce the routing overhead and power consumption for beaconing in contrast to the flat routing protocols. The reduction in overhead is also helpful to improve the scalability of the hierarchical routing protocols. Compared with implicitly hierarchical protocols, explicit hierarchical protocols will induce additional overhead in order to maintain the hierarchical structure under mobility. Moreover, as in HSR, more overhead is required to maintain the complex management for HID registrations and translations [45].

In most hierarchical routing protocols, the critical nodes (clusterheads or gateways) are elected by using some typical clustering algorithms [10] [16], which use lowest-ID, highest-connectivity (degree) as metrics. Under low mobility, the critical nodes remain relatively constant because of infrequent topology changes. However, because most routes go through these critical

Parameter	CGSR	ZRP	HSR
Routing strategy	Proactive	Hybrid	Proactive
Protocol type	Distance-vector	Distance-vector & Link-state	Link-state
Hierarchy	Explicit two levels	Implicit two levels	Explicit multiple levels
Loop-free	Yes	Yes	Yes
Routing metric	Via critical nodes	Local shortest path	Via critical nodes
Critical nodes	Yes(Clusterhead)	No	Yes(Clusterhead)

Table 2.2: Comparisons of the characteristics of hierarchical routing protocols

nodes, they draw heavily on the battery power of these nodes. Therefore, power-efficiency should be an important research issue for the cluster-based algorithms.

2.1.3 Location-Aware Routing Protocol

With the development of GPS devices nowadays, several location-aware routing protocols have been proposed to improve the performance of the routing protocols. In these protocols, each node determines its own location through

Figure 2.5: LAR: limited flooding of route request

the use of GPS or some other type of location services. The routing decision at each node is then based on the locations of its destination and neighbors. In the following, we review a few typical location-aware routing protocols with their location service schemes.

- **Location-Aided Routing** — Location-Aided Routing (LAR) protocol [30] is an improvement of the reactive routing protocols by using location information to enhance the route discovery phase. In LAR, the route discovery procedure also uses flooding approach to find routes as in DSR and AODV. However, in LAR, nodes can use location information to reduce the number of nodes desired to propagate the route requests. Thus, the route discovery is based on a “limited” flooding approach and the propagated region is also restricted in a certain area, called *request zone*. In LAR, two schemes have been proposed to determine the request zone [30].

In the first scheme, as illustrated in Figure 2.5 (a), when requiring a route from the source node S to the destination node D , the protocol defines a smaller forwarding zone instead of flooding the whole network. The forwarding zone seeks the smallest rectangle called *request zone* that can bound S ’s current location and D ’s possible location. The set of nodes in the request zone should be forwarded the route discovery packet. D ’s expected location is a gray circle zone called *expected zone* within the request zone and represents the possible

Figure 2.6: Example for the expected region in DREAM

region containing D at a certain time. In Figure 2.5 (a), node I , located in the forwarding zone, is responsible for broadcasting the route request packet, but node J , being outside the request zone, is not responsible for broadcasting the packet.

In the second scheme, as shown in Figure 2.5 (b), when the source node S is required to send packets to the destination node D , S first calculates the distance to the destination by using the location information of D . Let this distance be denoted by d_{SD} . The value of d_{SD} is stored in the route request and sent to its neighbors. When a node I receives the request, it compares the distance d_{SD} stored in the request with its own distance to the destination (denoted as L_{ID}). If L_{ID} is less than or equal to L_{SD} , the node will reply with a request message containing L_{ID} instead of L_{SD} . Otherwise, the request will be discarded. In Figure 2.5 (b), both nodes I and J will receive the request from node S , but only node I will reply to the request.

- **Distance Routing Effect Algorithm for Mobility** — Distance Routing Effect Algorithm for Mobility (DREAM) [1] is a proactive location-aware routing protocol. In DREAM, the routing table stored at each node contains the location information of other nodes. An entry in the routing table includes a node identifier, the direction and the distance to the node, as

well as a time value that indicates when this information was generated. As shown in Figure 2.6, when node S wants to send a packet to node D , it uses its own location information for D to obtain D 's direction of movement, and then transmits its packet to all of its one hop neighbors in the direction of D . Each successive hop repeats the procedure until D is eventually reached, if possible. The probability of finding D in the computed direction relies on how the location information is disseminated through the network. In DREAM, each node transmits control messages bearing its current location to all the other nodes. The frequency with which these control messages are transmitted is determined by the *distance effect* and the *mobility rate*.

The *distance effect* indicates that, the greater the distance between the two nodes, the slower they appear to be moving with respect to one another. Thus, nodes that are far apart are required to update each other's locations information less frequently than nodes that are closer together. This is realized by adding an "age" information to each control message which corresponds to how far from the sender that message travels.

The *mobility rate* indicates that, the faster a node moves, the more often it must update its location information. This allows each node to self optimize its dissemination frequency, thus transmitting location information only when needed and without compromising the route accuracy.

- ***Greedy Perimeter Stateless Routing*** — Greedy Perimeter Stateless Routing Protocol (GPSR) [27] is a novel geographic routing protocol without specific location service scheme. It uses the locations of mobile nodes and packets' destinations to make packets forwarding decisions. The location information can be provided by some specific location service schemes. GPSR uses the information about a node's immediate neighbors to make greedy forwarding decisions in the network topology. When a packet reaches a region where greedy forwarding cannot be used, the algorithm recovers by routing

Figure 2.7: Greedy forwarding in GPSR

around the perimeter of the region. By keeping state only about the local topology, GPSR has a better scalability than shortest path and other ad hoc routing protocols when the number of destinations increases.

GPSR has two main operating states: *Greedy* mode, and Planar *Perimeter* mode. Greedy mode is the default operational state, and the Planar Perimeter mode is invoked only when the Greedy mode fails.

In *Greedy* mode, each node will forward the data to its neighbor node closest to the destination, as illustrated in Figure 2.7 (a). This is achieved by examining the destination location contained within the data packet header, and checking its neighborhood table to determine which of its neighbors is closest to the destination. Then the source node x sends the data to the node y and the forwarding continues like this until the data reaches the destination node D . However, in some cases, the greedy mode cannot be continued. This happens when the sending node itself is the closest node to the destination in its neighborhood, as is shown in Figure 2.7 (b). This implies that the node cannot find a neighbor in the area common to both the node's neighbor range and the circle centered at the destination, as is shown in Figure 2.7 (b). This area is referred to as the “void” of the routing node. This implies a failure of

the greedy routing strategy, and GPSR must turn control over to the Perimeter mode.

The planar perimeter mode is used to avoid routing failures possible with greedy mode. The term planar perimeter refers to the fact that GPSR will forward around a void in the case of a *Greedy* mode failure. It chooses the direction based on the *right-hand rule*, that is, it automatically forwards to the first node it reaches when it looks in a counter-clockwise direction from the source node.

- ***Grid Routing with Grid Location Service*** — Grid Routing with Grid Location Service [33] is an approach that combines hierarchical and geographic routing. In Grid routing, a proactive distance vector routing protocol is used at the local level, while geographic routing is used for long-distance packet forwarding. The main idea of Grid routing is to have at least one location-aware node in each area where the proactive distance vector protocol is used. The location-aware nodes in this area may then be used as proxies: a location-unaware node uses the position of a location-aware node as its own position. Packets that are addressed to a location-unaware therefore arrive at a location-aware proxy and are then forwarded according to the information of the proactive distance vector protocol.

Because of the possible failure of geographic forwarding, a repair mechanism called Intermediate Node Forwarding (INF) is proposed in [33]. If a forwarding node has no neighbors that are closer to the destination than itself, it discards the packet and sends a notification to the sender of the packet. The sender of the packet then chooses a single intermediate position randomly for a circle around the midpoint of the line between the sender and the receiver. Packets have to traverse that intermediate position. If the packet is discarded again, the radius of the circle is increased and another random position is chosen. This is repeated until the packets are delivered to the destination or

until a predefined value has been reached and the sender assumes that the destination is unreachable.

Grid routing is based on a distributed location service, called Grid Location Service (GLS) [33], which can scale to large numbers of nodes. GLS is based on the idea that a node maintains its current location in a number of location servers distributed throughout the network. These location servers are not specially designated; each node acts as a location server on behalf of some other nodes. The location servers for a node are relatively dense near the node but become more sparse farther from node; this ensures that anyone near a destination can use a nearby location server to find the destination, while also limiting the number of location servers for each node. More details of GLS can be found in [33].

- ***Distributed Location Management Scheme (DLM)*** — The distributed location management scheme [72] is similar to GLS [33]. The entire network is partitioned into a hierarchical grid; for a specific node A in the network, a small subset of nodes, namely its location servers (each will be responsible for a region in the grid), will be chosen and periodically updated with A 's location. However, DLM uses a different approach to select such location servers. Instead of selecting the node in a partition at a certain hierarchy which has the “closest” ID to A 's ID as the location server, we use a *hash function* to map A 's ID directly to a set of locations in the network, and then select nodes at those locations as the location servers of A . Via this approach, location server selections may be made without scanning the entire region, while eliminating a considerable amount of update messages when nodes are mobile.

In DLM, a hierarchical addressing model is introduced to represent locations of nodes at different accuracy levels, while different location servers may carry location information of different levels of accuracy. Only a small set of location servers needs to be updated when the node moves.

Figure 2.8: Location management in SLURP

• **Scalable Location Update-based Routing Protocol (SLURP)** — SLURP [69] is developed to constantly maintain approximate location information about nodes in the network and to only find accurate routes to specific nodes when there are packets to be sent to them. This scheme is similar to the one adopted in DREAM [1]. Specifically, when a node needs to send a packet to some destination, it first determines the destination's approximate location and then uses a simple geographic routing protocol to send the packet to the destination. SLURP is based on a combination of (1) approximate geographical routing and (2) a simple static mapping procedure to maintain approximate location information for nodes.

In SLURP, the network area is divided into unit square regions. Each node is assigned a home region by hashing, and all the nodes in a region r act as the location servers for the nodes that are hashed into r . SLURP has eliminated the need to change location servers when nodes move. The location of a node is independent of its servers' location. Thus, if a node x wants to correspond with a nearby node y , it still needs to send a location query to y 's home region (which may be far away) to discover y 's location. As shown in Figure 2.8, when node D moved from region 16 to 25, it updates the new location to its location servers in region 12. When node S has packets sent to D , it needs

to query the location information of D from the nodes in region 12, and then sends the packets to node D .

The geographic routing of SLURP is based on MFR (Most Forward with fixed Radius) [21] without backwards progression. When the first packet is sent to a destination, the source does not know the precise coordinates of the destination. It only knows the region that the destination is currently located in. Thus, to route packets using MFR, the source determines the next hop using the coordinates of the center of the destination's current region as the location of the destination. When the packet reaches a node in the region, if that node has a cached route to the destination, it uses that route to forward the packet. Otherwise, it broadcasts a location discovery packet (as in DSR [24]) to nodes within the same region. The destination eventually receives this packet and generates a response that contains the route. Thus, MFR is used to get the packet to a region and then SLURP uses source routing to get the packet to the destination.

Comparisons of Location-aware Routing Protocols:

LAR use a reactive routing strategy, which can conserve bandwidth and power as inherited from reactive routing protocols. LAR further reduces routing overhead by restricting the propagation of route request packets. However, when there is no path available within the limited request zone or when location information is stale, LAR uses full area flooding approach as in the reactive routing protocol. Moreover, location information is used partially to reduce flooding reduction during the route discovery.

In DREAM, nodes also make data forwarding decisions based on the location information carried by the data packets. Due to the partial flooding of data packets toward the direction of the destination, multipaths can exist in DREAM and LAR protocols, and thus increase the robustness of the path and the probability of reception under mobility. LAR and DREAM have a same

CHAPTER 2. LITERATURE STUDY

communication complexity of $O(N)$ (where N denotes total number of nodes in the network), since they also use restricted directional flooding to forward packets. However, the control overhead increases with the size of network, and hence they scale less well to large networks with a large number of data communications.

In GPSR, the routing overhead is restricted only by the periodic beacon messages and a routing table, which is smaller than the ones in DREAM and LAR, since, in GPSR, the table is only to store location information of neighbors, while in DREAM and LAR, tables contain the location information of all the nodes in the network. Thus, GPSR is insensitive to the size of network in terms of scalability. However, in GPSR, the routing decisions are based on the location of destination. Thus, inaccurate location of destination causes packet loss and non-productive traffic.

Different from the other three geographic routing protocols, Grid routing provides hierarchical approach to routing, which achieves smaller storage and communication complexity. Thus, Grid scales better to large network with insensitivity of mobility. Due to the usage of a non-location-aware approach at the local level, Grid is more tolerant of location inaccuracy than GPSR.

GLS and DLM all use a hierarchical grid system with the smaller squares aggregated to form larger squares, giving rise to an infrastructure of overlapping squares of different sizes. This hierarchical grid structure is used to choose location servers that are well distributed in the system in GLS, and for addressing in DLM. DLM is similar to GLS except that in DLM location servers for a node are selected by applying a hash function with the node's ID, whereas in GLS the location server of a node is the node with the closest ID at the same level of the hierarchy.

SLURP has eliminated the need to change location servers when nodes move. The location of a node is independent of its servers' location. Thus,

if a node x wants to correspond with a nearby node y , it still needs to send a location query to y 's home region (which may be far away) to discover y 's location.

In all of the above schemes, the location information is updated by broadcasting to the entire network or some regions. When nodes have high mobility, they have to update frequently in order to maintain the accuracy of information. Too many update messages drain away more power and also decrease effective bandwidth of the network. Moreover, frequent node mobility makes location information stale and inconsistent which decreases the routing performance significantly [28].

2.2 Power Saving Mechanisms for MANETs

From the review in Section 1.2, we know that most of existing routing protocols for MANET are evaluated by using conventional metrics such as packet loss rates, routing overhead, and number of hops; few are considered in the light of power consumption. Since ad hoc networks are often deployed using battery-powered nodes, comparison and optimization of protocol energy consumption are important. In this section, we will review current existing power saving mechanisms on mobile ad hoc networks.

As shown in Figure 2.9, existing power saving mechanisms can be classified into two categories: *Power Management* and *Power Control*. Power management refers to the mechanism that saves energy by switching to *sleep* mode when the network interface is *idle*, while power control refers to the mechanism that reduces the power consumption during communication by using appropriate transmission power.

Figure 2.9: Power saving mechanisms on MANETs

2.2.1 Power Management Mechanism

Power management reduces energy consumption by turning off its radio interface when it has no communication activity. There are two kinds of power management approaches: one is *MAC-based Power Management*, and the other one is *Topology Control Power Management*.

MAC-based Power Management

This approach is to use the MAC layer based information to decide whether there is a frame transmission that is not destined to itself, and then turn off the radio interface module for one transmission frame.

Power Aware Multi-Access protocol (PAMAS) [60] is a typical example of the MAC-based power management protocol. The PAMAS protocol modifies the MACA protocol described in [26] by providing separate channels for RTS/CTS control packets and data packets. In PAMAS, a mobile node with packets to transmit sends a RTS (request-to-send) message over the control channel, and awaits the CTS (clear-to-send) reply message from the receiving node. The node enters a backoff state if no CTS arrives. However, if one CTS is received, then the node transmits the packet over the data channel. The receiving node transmits a "busy tone" over the control channel enabling users tuned to the control channel to determine that the data channel is busy.

Power conservation is achieved by allowing nodes that are not able to receive and send packets to turn off the wireless interface. The idea is that

a data transmission between two mobiles need not be overheard by all the neighbors of the transmitter. The use of a separate control channel allows for nodes to determine when and for how long to power off. A mobile node could power itself off when [60]: (i) it has no packets to transmit and a neighbor begins transmitting a packet not destined for it, and (ii) it does have packets to transmit but at least one neighbor-pair is communicating. Each mobile determines the duration that it could be powered off through the use of a probe protocol, the details of which are described in [60]. Theoretical bounds on power savings for random, linear, and fully connected topologies are also presented. The results from simulation and analysis show that between 10% and 70% power savings can be achieved for fully connected topologies.

Topology Control Power Management

MAC-based power management is feasible since all the idle nodes can operate in *sleep* mode and be waken up for data transmission whenever necessary. However, because of route setup phase, this approach may be inefficient and result in prolonged latency.

However, in topology control power management, the above issues can be addressed by always allowing a small set of nodes to be alive, which can form a virtual backbone for the network. Packets are delivered through the virtual backbone without cost of setup latency. This approach not only facilitates efficient data delivery at the same time of power conservation, but also guarantees the connectivity of the network. There are three schemes for topology control power management in the existing literature. They are presented as follows.

In adaptive energy conserving routing (AFECA) [70], each node maintains a count of the number of nodes within radio range, obtained by listening to transmissions on the channel. A node switches between sleeping and listening, with randomized sleep times proportional to the number of nearby nodes.

CHAPTER 2. LITERATURE STUDY

The net effect is that the number of listening nodes remains roughly constant, regardless of node density; as the density increases, more energy can be saved. AFECA's constants are chosen so that, with a high probability, the listening nodes form a connected graph and hence ad hoc forwarding works. An AFECA node does not know whether it is required to listen in order to maintain connectivity, so to be conservative AFECA tends to make nodes listen even when they could be asleep.

In geographical adaptive fidelity (GAF) [71], nodes use geographic location information to divide the world into fixed square grids. The size of each grid stays constant, regardless of node density. Nodes within a grid switch between sleeping and listening, with the guarantee that one node in each grid stays up to route packets.

In Span scheme [7], each node in the network running Span algorithm makes periodic, local decisions on whether to sleep or stay awake as a coordinator and participate in the forwarding backbone topology. To preserve capacity, a node volunteers to be a coordinator if it discovers, using information it gathered from local broadcast messages, that two of its neighbors cannot communicate with each other directly or through one or two existing coordinators. To keep the number of redundant coordinators low and rotate this role amongst all nodes, each node delays announcing its willingness by a random time interval that takes two factors into account: the amount of remaining battery energy, and the number of pairs of neighbors it can connect together. This combination ensures, with high probability, a capacity-preserving connected backbone at any point in time, where nodes tend to consume energy at about the same rate. Span does all this using only local information, and consequently scales well with the number of nodes. The results from simulation and analysis show that system lifetime with Span is more than a factor of two better than without Span, for a range of node densities, without much reduction in overall forwarding capacity.

All the protocols are proactive in constructing the “virtual backbone” without consideration of traffic characteristics. Maintenance of the routing backbone requires local exchange of messages which will consume a significant amount of energy, which is undesired.

2.2.2 Power Control Mechanism

In this mechanism, nodes can also use appropriate transmission power to reduce the power consumption during data transmission.

MAC-based Power Control

MAC-based power control is proposed to enable the collision free access of the wireless channel when the transmission range is adjustable. Power Controlled Multiple Access Protocol (PCMA) [40] is a typical power controlled MAC protocol. PCMA generalizes the transmit-or-defer “on/off” collision avoidance model of current protocols to a more flexible “variable bounded power” collision suppression model. In PCMA, there are two main power saving mechanisms [40]: (1) A request-power-to-send(RPTS)/acceptable-power-to-send(APTS) handshake between the data sender and receiver, which is used to determine the minimum transmission power that will result in a successful packet reception at the receiver. The RPTS/APTS handshake occurs in the *data channel* and precedes the data transmission; (2) The noise tolerance advertisement is used by each active receiver to advertise the maximum additional noise power it can tolerate, given its current received signal and noise power level. The noise tolerance advertisement or busy tone is periodically pulsed by each receiver in the *busy tone channel*, where the signal strength of the pulse indicates the tolerance to additional noise. A potential transmitter can “sense the carrier” by listening to the busy tone for a minimum time period in order to detect the upper bound of its transmission power for all control and data packets.

Topology Control

MAC-based power control adjusts the transmission power locally, while topology control determines the transmission power of each node in the network so that the global minimum energy topology can be constructed. This approach not only reduces power consumption for transmission at nodes, but also decreases the chance of collisions, which reduce energy used in retransmission. However, the reduction of transmission power of nodes may decrease number of links in the network and partition the network. Thus, network connectivity is an important issue when designing the topology control algorithms. Several algorithms [6, 21, 56, 58, 68] have been proposed based on this mechanism. In the following, we will describe the main characteristics of these algorithms.

Rodoplu and Meng [58] proposed a minimum energy mobile wireless network protocol (MEMN for short), which is a distributed position-based network protocol optimized for minimum energy consumption in mobile wireless networks that support peer-to-peer communication.

The main idea of this protocol is that it is not required to consider all the nodes in the network to find the globally minimum power path to the desirable destination. Instead, using a very localized search, a node can eliminate some nodes in its relay regions from consideration and select only those few nodes in its immediate neighborhood to be the only potential candidates. The protocol runs in two phases. In the first phase, each node searches for its neighbors and selects these neighbors for which direct transmission requires less power than if an intermediate node is used to retransmit the message. This defines a graph called enclosure graph. In the second phase, each possible destination runs distributed loop-free variant of non-localized Bellman-Ford shortest path algorithm and computes the shortest path for each possible source. The same algorithm is run from each possible destination. However, this algorithm requires frequent distribution of the current topology of the entire network.

Figure 2.10: Node u find a neighbor v in the cone of α

With the presence of node mobility, such approaches would suffer from either out-of-date states or flooding of triggered updates. It also brings large message exchange overhead to the network, which by itself consumes nodal power.

This protocol has been improved by some other researchers in [34, 36, 37]. In [34], a network topology with smaller number of links than MEMN is constructed by using the minimum-energy property of MEMN. [34] has lower link maintenance costs than MEMN and can achieve a significant saving in energy consumption, and it is also computationally simpler than MEMN. In [36] and [37], some sparser spanners have been proposed to improve the efficiency of the results in [58], and these spanners have a linear number of edges and also preserve the path of the minimum power for any pair of nodes, while in [58], the time complexity of each node is in the cubic order of the number of its neighbors in the worst case.

Another topology control algorithm is proposed by Wattenhofer et al. [68]. The main idea of their algorithm is that each node adjusts its transmission power until it finds a neighboring node in each direction. The range of the direction is defined as a cone, as shown in Figure 2.10 [68].

Their algorithm, referred to Cone-Based Topology Control algorithm (CBTC for short), runs in two phases, which are summarized as follows: Starting with a small radius, each node (denoted by node u as shown in Figure 2.10) broadcasts a neighbor-discovery message. Each receiving node acknowledges this

broadcast message. Node u records all acknowledgments and the directions they came from. (Assume that the node can determine the direction of the sender when receiving a message). It then determines whether there is at least one neighbor in every cone of α degrees, centered on node u . In this first phase, node u continues the neighbor discovering process by increasing its transmission radius (operational power) until either the above condition is met or the maximum transmission power is reached. They prove in [68] that, for α smaller than or equal to $2\pi/3$, the algorithm guarantees maximum connected node set. For α smaller than or equal to $\pi/2$, they also can guarantee power-efficient routes. In the second phase, the algorithm performs a redundant edge removal process without impacting the connectivity. This phase is designed to reduce the node degrees, which help in reducing interference and enhancing throughput [21]. Redundant edge removal is carried out without deteriorating the minimum power routes of the network.

In [35], some improvements have been proposed for CBTC. It is shown to be power-efficient to use a tight upper bound $5\pi/6$ on the cone degree for the algorithm to preserve connectivity, and three optimizations to the basic algorithm in [68], which are the shrink-back operation, asymmetric edge removal, and pairwise edge removal.

Power-Aware Routing

In power-aware routing protocols, nodes use power consumption as routing metrics and find routes between sources and destinations which consume the least power.

In [61], the power-aware routing protocol is proposed to use a function $f(A)$ to denote node A 's reluctance to forward packets and to choose a path that minimizes the sum of $f(A)$ for nodes on the path. This routing protocol [61] addresses the issue of energy-critical nodes. As a particular choice for f , [61] proposes $f(A) = 1/g(A)$ where $g(A)$ denotes the remaining lifetime ($g(A)$ is

normalized to be in the interval $[0,1]$). Thus, reluctance grows significantly when lifetime approaches 0. The other metric used in [61] is aimed at minimizing the total energy consumed per packet. However, [61] merely observes that the routes selected when using this metric will be identical to routes selected by minimum hop count, since the energy consumed in transmitting (and receiving) one packet over one hop is considered *constant*. For each of the two proposed power consumption metrics (cost and hop count), [61] assigns weights to nodes or edges and then refers to nonlocalized Dijkstra's algorithm for computing shortest weighted path between source and destination. We also observed that the validation of power-aware metrics in [61] was done on random graphs where each pair of nodes is joined by an edge with a fixed probability p .

In [42], a power-aware routing optimization (PARO) is proposed by J. Gomez to consider the problem of power aware routing in wireless networks where all nodes are located within the maximum transmission range of each other. PARO uses a packet forwarding technique where immediate nodes can be elected as redirectors on behalf of source-destination pairs with the goal of reducing the overall transmission power.

2.3 Chapter Summary

In this chapter, we have surveyed the existing typical MANET routing protocols. This includes a classification according to the network topology underlying routing protocols. An overview of these MANET routing protocols has been given and the features of them have been summarized and highlighted. Moreover, existing power saving mechanisms have been classified and features of them have also been highlighted.

Chapter 3

A Localized Topology Control Algorithm for Power Control and Routing on MANETs

In this chapter, we describe our localized topology control algorithm for power control and power-efficient routing in details. Some important properties of the mobile ad hoc networks, such as network connectivity and loop-freedom, are also analyzed theoretically in this chapter. Finally, we evaluate our algorithm empirically and discuss the simulation results.

3.1 Geometric Model

3.1.1 General Assumptions

We model a Mobile Ad Hoc Network as a directed graph $G = (V, E)$, where V, E denote, respectively, the set of nodes and connections among these nodes in the network. Moreover, all n nodes in the set V are in a two-dimensional area, where no two nodes coincide at the same location.

We adopt a common *path loss model* to measure the power consumption P of transmission, in which the signal strength falls as a function of $1/D^k$, where

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

D denotes the distance between two nodes, and k (referred to as power coefficient henceforth) denotes a constant determined from field measurements, whose value typically lies between 2 and 4 [57]. In a more precise model, the path loss may depend on the height of the transmit antenna as well as the transmit-receiver separation [57]. In this report, we assume that the mobile devices have similar antenna heights so that the variation in the third dimension can be ignored.

We also assume that, in the network, each node can get its own geographical location through the support of GPS; there also exists a location service system, such as GLS [33], to provide the location information of destinations for source nodes in the network. In order to achieve power control, each node has the ability to adjust its transmission power, up to a maximum value that is identical for all nodes in the network. Here, we also assume the existence of an underlying MAC layer to resolve multiple channel access and interference problems.

3.1.2 Notations

The following conventions will be adopted throughout the report.

3.1.3 The Model

Since the required transmit power increases quickly as a function of D^k , relaying packets through intermediate nodes may in some cases result in lower power consumption than communicating directly. To investigate the implications of location information on power-efficient transmission, we assume that $k \geq 2$ is a given value and present the following model that defines the relay region of a node.

Consider three nodes s , r and t on a plane, where s denotes the sender and t denotes the destination node. If s uses r as its next hop to forward packets to t , node r is referred to as a relay node of s .

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

$d(x, y)$	The Euclidean distance between node x and node y .
$x \rightarrow y$	There exists a link from node x to node y .
$P(x)$	The transmission power of node x . P_{max} denotes the maximum transmission power for each node, and hence $0 \leq P(x) \leq P_{max}$.
$P(x \rightarrow y)$	The transmission power required for a link from node x to node y .
$R(x)$	The transmission radius of node x with $P(x)$. R_{max} is the maximum transmission radius, and hence $0 \leq R(x) \leq R_{max}$.
$N(x)$	The set of neighbors of node x . A <i>neighbor</i> of node x is defined as a node who can receive the packets sent by node x with transmission power $P(x)$. $N_{max}(x)$ denotes the maximal set of neighbors of node x with the maximum transmission power P_{max} . So the following relationship exists: $N(x) \subseteq N_{max}(x)$.

Definition 3.1 (*Relay Region*) [58]: For a given sender-relay node pair (s, r) , the relay region $RR(s, r)$ of the ordered pair (s, r) is defined as

$$RR(s, r) = \{x | d^k(s, r) + d^k(r, x) \leq d^k(s, x), x \neq r\}$$

Figure 3.1: Relay regions for node s in case of $k = 2$ and $k > 2$

The shaded areas in Figure 3.1 illustrate the relay regions of (s, r) with the power coefficient $k = 2$ and $k > 2$ respectively. This notation of *relay region*

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

is first defined by Rodoplu in [58]. Some properties of *relay region* are also proved in this paper. Here, we summarize them in the Property 3.1, 3.2, 3.3.

Property 3.1 (*Asymptotic Behavior of Relay Regions*) [58]

For a given sender-relay node pair (s, r) , let $B(s, r)$ be the boundary of $RR(s, r)$. Let the relay node r be located at the origin and the sender node s be located on the $+x$ axis. Let (x_a, y_a) represent a point on $B(s, r)$. For power coefficient $k > 2$, $y_a \rightarrow \pm\infty$ when $x_a = d(s, r)/2$. For $k = 2$, $B(s, r)$ is given by $x_a = 0$.

Property 3.2 (*Properties of Relay Regions*) [58]

Let s be the sender node. Relay regions of node s have the following properties.

- (i) For any $u \neq s$, $u \notin RR(s, u)$.
- (ii) If $u \in RR(s, t)$, then $t \notin RR(s, u)$.
- (iii) If $u \in RR(s, t)$ and $v \in RR(s, u)$, then $t \notin RR(s, v)$.

Property 3.3 (*Distance Properties of Relay Nodes*) [58]

Let s be the sender node, r the relay node, and t the destination node. If $t \in RR(s, r)$, then $d(s, t) > d(r, t)$ and $d(s, t) > d(s, r)$.

Lemma 3.1 (*Angle Property of Relay Nodes*)

Let s be the sender node, r the relay node, and t the destination node. (1) If $t \notin RR(s, r)$, then the angle $\angle srt < \pi/2$. (2) If the angle $\angle srt \geq \pi/2$, then $t \in RR(s, r)$.

Proof: (1) If $t \notin RR(s, r)$, by Definition 3.1, we have $d^k(s, r) + d^k(r, t) > d^k(s, t)$ for $k \geq 2$, and hence $d^{2k}(s, t) < (d^k(s, r) + d^k(r, t))^2$. In $\triangle srt$, we have $d^2(s, t) = d^2(s, r) + d^2(r, t) - 2d(s, r)d(r, t)\cos\angle srt$, and hence we can get $(d^2(s, r) + d^2(r, t) - 2d(s, r)d(r, t)\cos\angle srt)^k < (d^k(s, r) + d^k(r, t))^2 \leq (d^2(s, r) + d^2(r, t))^k$ (see Appendix for proof). So we get $d^2(s, r) + d^2(r, t) -$

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

$2d(s, r)d(r, t) \cos \angle srt < d^2(s, r) + d^2(r, t)$, and hence $\cos \angle srt > 0$. Then $\angle srt < \pi/2$.

Statement (2) is just the contrapositive of statement (1), so it is also true according to (1). ■

Definition 3.2 (*Relay Candidate*): For a given node s , and node $r \in N(s)$. Node r is said to be a relay candidate of node s if $\nexists x \in N(s)$, such that $r \in RR(s, x)$. The set of relay candidates of node s is said to be relay set of node s and denoted by $RC(s)$.

Intuitively, for each node s , its relay candidates are the set of nodes that can relay messages for s to the other nodes more power-efficiently than direct transmission. Here, if a node j is a relay candidate of node i , and node l lies in the relay region $RR(i, j)$, node l is said to be *dominated* by node j . For those nodes that are dominated by node l for node i , they are also dominated by node j .

Lemma 3.2 (*Power Efficiency of Relay Candidates*)

Given a node s and node $r \in RC(s)$, for any $t \in RR(s, r)$, (1) $P(s \rightarrow r \rightarrow t) \leq P(s \rightarrow t)$; (2) if $t \notin RR(s, r')$ for any other relay candidate r' of s , then $P(s \rightarrow r \rightarrow t) < P(s \rightarrow r' \rightarrow t)$.

Proof: As given by the transmission power model, the signal strength falls as $1/d^k, k \geq 2$. Assume that $P = c \cdot d^k$, where c is a positive constant.

(1) By assumption, we have $P(s \rightarrow t) = c \cdot d^k(s, t)$ and $P(s \rightarrow r \rightarrow t) = c \cdot d^k(s, r) + c \cdot d^k(r, t)$. Since $t \in RR(s, r)$, by Definition 3.1, we have $d^k(s, r) + d^k(r, t) \leq d^k(s, t)$, and hence $c \cdot d^k(s, r) + c \cdot d^k(r, t) \leq c \cdot d^k(s, t)$. So we get $P(s \rightarrow r \rightarrow t) \leq P(s \rightarrow t)$.

(2) Since $t \in RR(s, r)$, as proved in (1), we have $P(s \rightarrow r \rightarrow t) \leq P(s \rightarrow t)$. Similarly, since $t \notin RR(s, r')$, we have $P(s \rightarrow r' \rightarrow t) > P(s \rightarrow t)$. Therefore, $P(s \rightarrow r \rightarrow t) < P(s \rightarrow r' \rightarrow t)$. ■

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

From the above lemma, we know that relaying packets using relay candidates may result in lower power consumption than communicating directly or via other neighbors. Thus the relay candidates play an important role in the power-efficient routing.

Definition 3.3 (*Enclosure Region*): For a given node s lying in a bounded region D , the Enclosure Region $ER(s)$ of node s is defined by

$$ER(s) = D - \bigcup_{r \in RC(s)} RR(s, r)$$

where $RC(s)$ denotes the set of relay candidates of node s , and $\bigcup RR(s, r)$ denotes the overall relay regions of node s . The shadowed area in Figure 3.2 illustrates the enclosure region $ER(s)$ of node s .

Figure 3.2: Enclosure region $ER(s)$ of node s ($k > 2$)

Definition 3.4 (*Relay Polygon*): For a given node s , let $r_0, r_1, \dots, r_{n-1}$ be the list of relay candidates of node s , sorted according to the clockwise direction with respect to the node s . The relay polygon $RP(s)$ is formed by linking every two consecutive relay candidates in this sequence, i.e.

$$RP(s) = \bigcup_{0 \leq i < n} \overline{r_i r_{i+1 \pmod n}}$$

Figure 3.3 illustrates the relay polygons (as shown in solid lines) in two general cases. As shown in Figure 3.3(a), if node s is internal to its relay

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL
AND ROUTING ON MANETS

Figure 3.3: Relay polygons formed in two general cases

polygon, we say the enclosure region $ER(s)$ of node s is *bounded*; otherwise, as shown in Figure 3.3(b), if node s is external to its relay polygon, we say the enclosure region $ER(s)$ of node s is *unbounded*. Obviously, the enclosure of one node may be bounded only when the number of its relay nodes is more than two.

Lemma 3.3 *For any arbitrary node x on the plane, the relay polygon $RP(x)$ is a convex polygon.*

Proof: We first prove that the relay polygon $RP(x)$ is a simple polygon. For a given node x , nodes $r_1, r_2, \dots, r_j, r_{j+1}, \dots, r_m$ are its relay candidates. For contradiction, assume that $RP(x)$ is not a simple polygon. According to the definition of *simple polygon* in [52], there must exist at least two crossing edges in the polygon $RP(x)$. By the construction rule of relay polygon in Definition 3.4, the relay candidates $r_1, r_2, \dots, r_j, r_{j+1}, \dots, r_m$ are sorted according to the clockwise direction with respect to the source node x . As shown in Figure 3.4 (a), when node x is internal to its relay polygon, the edges linking any two consecutive nodes r_j, r_{j+1} in the relay polygon lie in the different fan regions formed by these consecutive nodes and source node, like $\widehat{r_j x r_{j+1}}$. So there does not exist any other relay candidates in the fan regions and hence these edges cannot intersect each other. As shown in Figure 3.4 (b), when

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL
AND ROUTING ON MANETS

(a): node x is internal to $RP(x)$ (b): node x is external to $RP(x)$

Figure 3.4: Two kinds of relay polygons

node x is external to its relay polygon, the edges linking any two consecutive nodes r_j, r_{j+1} (except the last edge $\overline{r_m r_1}$) in the relay polygon also lie in the different fan regions. So these edges cannot intersect each other, and hence one of the crossing edges must be the last edge in the sequence of relay polygon, e.g. $\overline{r_m r_1}$. Assume that $\overline{r_j r_{j+1}}$ intersects $\overline{r_m r_1}$ at point w , as shown in Figure 3.5. So nodes r_j and r_{j+1} lie in the two sides of line $\overline{r_m r_1}$, as shown in Figure 3.5. Since nodes r_1 and r_m are the first and last ones in the sequence of relay candidates in relay polygon $RP(x)$, node r_j lies in the triangle $\triangle r_1 x r_m$ that is contained by the circle with diameter $\overline{r_1 x}$ or $\overline{r_m x}$. Without loss of generality, we assume that node r_j lies inside the circle with diameter $\overline{r_1 x}$. So the angle $\angle r_1 r_j x > \pi/2$. By Lemma 3.1, r_1 must be in the relay region $RR(x, r_j)$, which contradicts to the assumption that node r_1 is a relay candidate of node x . So we can conclude that the relay polygon $RP(x)$ is a simple polygon.

Next, we show that the relay polygon of node x is a convex polygon. For contradiction, assume that the relay polygon is a non-convex polygon. By the definition of *nonconvex polygon* [52], there exist two points p, q interior to the polygon $RP(x)$, such that $\overline{pq}$ is not entirely contained in $RP(x)$. Assume that $\overline{pq}$ intersects $\overline{r_{j-1} r_j}$ and $\overline{r_j r_{j+1}}$ at points g and h respectively, and the segment $\overline{gh}$ of line $\overline{pq}$ is not contained in the relay polygon $RP(x)$. As shown

Figure 3.5: The relay polygon of node x is a simple polygon

in Figure 3.6, in the triangle $\triangle gr_jh$, the angle $\angle gr_jh$ is less than π , and hence the complementary of $\angle gr_jh$ is greater than π . So at least one of the angles $\angle r_{j-1}r_jx$ and $\angle r_{j+1}r_jx$ must be greater than $\frac{\pi}{2}$. Without loss of generality, we assume that $\angle r_{j-1}r_jx > \pi/2$. Then, by Lemma 3.1, r_{j-1} must be in the relay region $RR(x, r_j)$. This contradicts to the condition that r_{j-1} is a relay candidate of node x . So combining with the fact that $RP(x)$ is a simple polygon, we conclude that the relay polygon $RP(x)$ of node x is a convex polygon.

Figure 3.6: The relay polygon of node x is convex

■

Since the *point location problem* [52] in convex polygon is simpler than in non-convex polygon, Lemma 3.3 can help us to simply the judgement of point location in our algorithm. It's also the reason that we only consider the closure

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

of polygon formed by nodes in $RC(i)$, not in $N(i)$.

Definition 3.5 (*Relay Graph*): Given a network $G = (V, E)$, the relay graph of this network, denoted by $RG = (V', E')$, is defined as

$$RG = \{(V', E') | V' = V, E' = \bigcup_{i \in V'} \bigcup_{j \in RC(i)} (i, j)\}$$

In the following, we will prove that the graph RG is a planar graph with no crossing edges, which can help us to apply our location-based routing algorithm on the graph.

Lemma 3.4 *The graph RG is a planar graph with no crossing edges.*

Proof: Assume that there are two crossing edges $\overline{AB}$ and $\overline{CD}$ in Relay Graph, as shown in Figure 3.7. $d(A, B)$ and $d(C, D) \leq R_{max}$. We divide into three possible cases:

Case 1: Pairs of nodes in $\{A, B, C, D\}$ are neighbors of each other;

Figure 3.7: Pairs of nodes in $\{A, B, C, D\}$ are neighbors of each other;

In Figure 3.7, for node A , node B is a relay candidate of A and nodes C, D are A 's neighbors. By the definition of relay candidates, node B is not in the relay regions of any other neighbors of A , such as $RR(A, C), RR(A, D)$. By Lemma 3.1, we get $\angle ACB, \angle ADB < \pi/2$. Similarly, $\angle CAD, \angle CBD < \pi/2$. Combining these inequalities, we get $\angle ACB + \angle ADB + \angle CAD + \angle CBD < 2\pi$, which violates the basic property of quadrangles that the sum of all internal angles of any quadrangles is 2π .

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

Figure 3.8: $\{A, B\}$ and $\{C, D\}$ are neighbors, but the other pairs of nodes in $\{A, B, C, D\}$ may be not neighbors;

Case 2: $\{A, B\}$ and $\{C, D\}$ are neighbors, but the other pairs of nodes in $\{A, B, C, D\}$ may be not neighbors;

As shown in Figure 3.8, we list four possible cases. Case (a) and (b) can be proved by a similar approach in Case 1. For case (c), assume that node C is a neighbor of nodes A and B , but D is not a neighbor of A and B . Node B must be not in the relay region $RR(A, C)$. So we get that $\angle ACB < \pi/2$. Similarly, we also get $\angle CAD, \angle CBD < \pi/2$. Since node D is not a neighbor of nodes A, B , that means nodes A, B cannot receive beacons from node D , so $d(A, D), d(B, D) > d(A, B)$. Then the angle $\angle ADB$ must be the smallest angle in triangle $\triangle ADB$ because the edge $\overline{AB}$ that $\angle ADB$ faces is the smallest one in the triangle $\triangle ADB$, so $\angle ADB < \pi/2$. Combining all these four inequalities, $\angle ACB + \angle ADB + \angle CAD + \angle CBD < 2\pi$. This also violates the basic property of quadrangles. Similarly, case (d) can also be proved by this way.

Case 3: Only $\{A, B\}$ and $\{C, D\}$ are neighbors, and all the other pairs of nodes in $\{A, B, C, D\}$ are not neighbors;

In Figure 3.9, since node C is not a neighbor of nodes A, B , so $d(A, C), d(B, C) > d(A, B)$. Then the angle $\angle ACB$ must be the smallest angle in triangle $\triangle ACB$, and $\angle ACB < \pi/2$. Similarly, all the other three angles $\angle ADB, \angle CAD, \angle CBD < \pi/2$. We can also get the following inequalities $\angle ACB + \angle ADB + \angle CAD + \angle CBD < 2\pi$, which violates the basic property of quadrangles.

Figure 3.9: Only $\{A, B\}$ and $\{C, D\}$ are neighbors, and all the other pairs of nodes in $\{A, B, C, D\}$ are not neighbors;

Finally, we can conclude that the graph RG is a planar graph without crossing edges. ■

3.2 Relay Topology Control Algorithm

In this section, we will describe the Relay Topology Control (RTC) algorithm in details. It consists of three main parts: *Search for Relay Candidates*, *RTC for power control* and *RTC for power-efficient routing* algorithms.

3.2.1 Search for Relay Candidates

In this section, we propose a distributed algorithm to find the relay candidates of each node. In our relay topology control algorithm, the *Search for Relay Candidates* algorithm is the basis for other operations (power control, routing, etc.)

The pseudocode of this algorithm is shown in Figure 3.10, where i denotes the node executing the algorithm, and $N(i)$, $RC(i)$ denote the set of neighbors and relay candidates of node i respectively, when using transmission power $P(i)$. This algorithm is executed in each node to find its relay candidates.

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

```

Search_relay_candidates (i)
{
    RC(i) = N(i);
    for each j ∈ N(i)
        for each k ∈ N(i) - {j}
            if k ∈ RR(i, j)
                RC(i) = RC(i) - {k};
}

```

Figure 3.10: The pseudocode of search for relay candidates algorithm

3.2.2 Relay Topology Control Algorithm for Power Control

In this section, we propose a localized topology control algorithm called *Relay Topology Control (RTC)* algorithm. In RTC, each node uses less transmission power to broadcast, and dynamically adjusts its transmission power in order to guarantee the network connectivity. The power control decision is based on whether a node is surrounded by its relay candidates (internal to its relay polygon). In the following, we will describe the details of our algorithm and analyze the network connectivity after applying RTC.

A. The Algorithm

The details of the Relay Topology Control algorithm are described in the following.

Initially, each node i broadcasts beacons with a transmission power $P(i) = P_0$. After receiving replies from the other nodes, node i finds its relay candidates by executing the *Search for Relay Candidates* algorithm (described in Section 3.2.1) and calculates its relay polygon $RP(i)$. Then node i can judge whether itself lies inside of $RP(i)$. This can be regarded as a *point-location problem* in computational geometry [52]. If node i is not internal to $RP(i)$, it will continue to grow the transmission power until it becomes internal to its

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

Figure 3.11: An example of RTC

relay polygon, or until the maximum transmission power P_{max} is reached. In the following, we give an example to explain how RTC works.

Example: As shown in Figure 3.11, for a given node s , assume that the nodes a, b, c, e and f are all its relay candidates when node s broadcasts with the maximum transmission power P_{max} . Let $P_1(s)$ be a value satisfying relation $0 < P_1(s) < P_{max}$. Suppose that, when node s broadcasts with power $P_1(s)$, it can detect the presence of nodes a, b and c . Since s is not internal to the relay polygon formed by nodes a, b, c , node s increases its transmission power to $P_2(s)$, where $0 < P_1(s) < P_2(s) \leq P_{max}$. Presently, node s can detect nodes a, b, c, f . Moreover, it is internal to the relay polygon formed by these nodes. So node s keeps the transmission power $P_2(s)$ during this beacon interval (we assume that each node will beacon once in each time interval). ■

The pseudocode of algorithm RTC is given in Figure 3.12, where i denotes the node executing the algorithm, $P(i)$ denotes the current transmission power of node i and P_{max} denotes the maximum transmission power.

B. Connectivity Analysis of RTC

With the RTC algorithm, instead of always using the maximum transmission power, we may reduce the transmission power and still maintain network topology. This approach not only reduces power consumption in the network

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

```

RTC(i)
{
  P(i) = P0 (Initially, P0 = Pmax);
  while P(i) ≤ Pmax
  {
    Beaconsing;
    Search_relay_candidates(i);
    Compute RP(i);
    if i ∈ RP(i)
      return;
    else increase P(i);
  }
}

```

Figure 3.12: The pseudocode of RTC algorithm

topology maintenance operations, but also decreases the probability of interference. However, reduction of transmission power leads to fewer neighbors maintained by the nodes in the network compared with using the maximum transmission power. This may cause the network to be partitioned and routes to be broken. In this section, we will prove two theorems about the connectivity of network: (1) for each node in a MANET, maintaining all its relay candidates is sufficient to achieve the same network connectivity as using the maximum transmission power; (2) under some conditions, the network connectivity can still be preserved by maintaining a proper subset of the relay candidates.

We model a network, in which each node broadcasts with the maximum transmission power P_{max} , as a graph denoted by G_{max} . We assume that G_{max} is strongly connected. Let G be the graph constructed by our RTC algorithm. So our goal is to analyze whether G can preserve the connectivity of G_{max} .

A graph is *strongly connected* if there is a path between any pairs of nodes in the graph. By assumption, the network is connected as long as nodes can communicate with all its neighbors found by using maximum transmission

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

power. In RTC, a node x may not communicate with all its neighbors in $N_{max}(x)$, but only a subset $N(x)$ of them. If the nodes in $N_{max}(x) - N(x)$ can be connected through this subset of nodes, the network is still connected.

Here we are only required to consider the case where the relay enclosure of a given node is bounded, because when the enclosure is unbounded, according to our RTC algorithm, the node will operate with the maximum transmission power P_{max} , and hence maintain its connectivity.

Lemma 3.5 *Suppose that s is a given node and node $v \in N_{max}(s)$. There exists a path from s to v through a relay candidate of s .*

Proof: Since node $v \in N_{max}(s)$, according to *Search for Relay Candidates* algorithm, there are two cases for v : one is that v is a relay candidate of s if it is not dominated by any other neighbors of s . In this case, there exists a direct link from s to v ; the second case is that v is dominated by u_1 , one of the relay candidates of s , i.e. $v \in RR(s, u_1)$.

Since $v \in RR(s, u_1)$, according to Property 3.3, we know that $d(u_1, v) < d(s, v) \leq R_{max}$ and hence $v \in N_{max}(u_1)$. Similarly, there also exist two cases: (1) v is a relay candidate of u_1 , then there is a direct link from u_1 to v . Combining with the link from s to u_1 , the path indeed exists and the lemma holds for this case; (2) If v is not a relay candidate of u_1 , there must exist a relay candidate who dominates v (here we denote this relay candidate by u_2) and $v \in RR(u_1, u_2)$. In this case, we can derive that $d(u_2, v) < d(u_1, v) < d(s, v) \leq R_{max}$ and $v \in N_{max}(u_2)$. In general, we may assume that there exists a set of nodes $\{u_0, u_1, \dots, u_l\}$ ($u_0 = s$ and $u_l = v$) such that $u_i \in RC(u_{i-1})$ for $i \in [1, l]$.

In the following, we will prove this lemma by mathematical induction.

Base case: when $l = 1$, this case has been shown and the lemma holds.

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

Inductive step ($l = m$): Assume that the lemma holds when $l = m$, that is, there exists a path from s to v via u_1 . We need to prove that the lemma is still true in the case that $l = (m + 1)$.

Inductive step ($l = (m + 1)$): when $l = (m + 1)$, by the hypothesis, there exists a path from u_1 to v via u_2 . Since $u_1 \in RC(s)$, there exists a link from s to u_1 . So we conclude that there exists a path from s to v via u_1 . The lemma also holds in this case.

By mathematical induction, we conclude that the lemma is true. ■

Theorem 3.1 *Assume that G_{max} is a connected graph. If each node maintains all its relay candidates, the induced graph G stays connected.*

Proof: Consider any edge (i, j) in G_{max} , if (i, j) is not in G , by Lemma 3.5, there is a path from i to j through a relay candidate of i . In other words, the connectivity in G_{max} is always retained through such paths, and the theorem is true. ■

From Theorem 3.1, we know that relay candidates are the dominating nodes which can guarantee the connectivity of networks. However, in the RTC algorithm, a node may not maintain all its relay candidates. So, in the following, we will give the conditions that guarantee the connectivity of network when not all relay candidates are maintained by the nodes in the network.

Theorem 3.2 *Let node i denote an arbitrary node in the network, r_m, r_{m+1} be any two consecutive relay candidates in $RP(i)$. Let $\angle r_m i r_{m+1}$ be the angle formed by $\overline{r_m i}$ and $\overline{r_{m+1} i}$ with degree α . If G_{max} is connected, then the graph G will also be connected under the following conditions: (1) $0 < \alpha \leq \frac{2}{3}\pi$, or (2) $\frac{2}{3}\pi < \alpha < \pi$, and $d(r_m, r_{m+1}) \leq 2 \sin \alpha R_{max}$*

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

Figure 3.13: Connectivity analysis

Proof: Let node j be a neighbor of i , which is not in the current transmission range $R(i)$ but in the maximum transmission range R_{max} of node i . By assumption, node i is internal to its relay polygon, so node j must lie in one acute-angled fan region formed by two consecutive relay candidates r_1, r_2 of node i , like $\widehat{piq}$ in Figure 3.13.

By assumption, node j is not reachable by node i when using current transmission power. From Theorem 3.1, we know that the network is connected if node j can be reached through the relay candidates currently maintained by node i . So our aim is to show that node j is in the maximum transmission range of one of the relay candidates currently maintained by node i .

Here we use a line $\overline{ik}$ that bisects the angle $\angle r_1ir_2$ to aid our proof. By construction, $\angle r_1ik = \angle r_2ik = \frac{\alpha}{2}$.

(1) when $0 < \alpha \leq \frac{2}{3}\pi$. In $\triangle ijr_1$,

$$d^2(j, r_1) = d^2(i, j) + d^2(i, r_1) - 2d(i, j)d(i, r_1) \cos \angle jir_1.$$

Without loss of generality, assume that $\angle jir_1 \leq \angle jir_2$, so $\angle jir_1 \leq \frac{\pi}{3}$, and

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

hence $\cos \angle jir_1 \geq \frac{1}{2}$. Since $d(i, r_1) < d(i, j)$, we get

$$\begin{aligned} d^2(j, r_1) &= d^2(i, j) + d^2(i, r_1) - 2d(i, j)d(i, r_1) \cos \angle jir_1 \\ &\leq d^2(i, j) + d^2(i, r_1) - d(i, j)d(i, r_1) \\ &\leq d^2(i, j) \\ &\leq R_{max}^2 \end{aligned}$$

Thus $d(j, r_1) \leq R_{max}$. So the theorem is true in this case.

(2) when $\frac{2}{3}\pi < \alpha < \pi$, and $d(r_1, r_2) \leq 2 \sin \alpha R_{max}$, in $\triangle r_1jr_2$, we get

$$d^2(r_1, r_2) = d^2(r_1, j) + d^2(r_2, j) - 2d(r_1, j)d(r_2, j) \cos \angle r_1jr_2.$$

Since $\frac{2}{3}\pi < \alpha < \pi$, $-2 \cos \alpha > 1$. Without loss of generality, assume that $d(r_1, j) \leq d(r_2, j)$, so we have

$$\begin{aligned} d^2(r_1, r_2) &= d^2(r_1, j) + d^2(r_2, j) - 2d(r_1, j)d(r_2, j) \cos \angle r_1jr_2 \\ &> d^2(r_1, j) + d^2(r_2, j) + d(r_1, j)d(r_2, j) \\ &\geq 3d^2(r_1, j) \end{aligned}$$

Thus $3d^2(r_1, j) < d^2(r_1, r_2) \leq 4 \sin^2 \alpha R_{max}^2 < 3R_{max}^2$, Thus $d(r_1, j) < R_{max}$.

So the theorem is true in this case.

Combining all the cases, we conclude that the theorem is true. ■

In the basic RTC algorithm, the network is not always guaranteed to be connected. As shown in Figure 3.14, when node j is located in the shaded area, in which it is out of the maximum transmission range R_{max} of both node r_1 and r_2 , as well as the current transmission range $R(i)$, thus node i cannot communicate with it by using the current transmission power $P(i)$ (the corresponding range is $R(i)$). We call the shaded areas as the “Blind Corners” of node i .

In order to solve this problem, we propose a checkup operation for the RTC algorithm. After executing the RTC algorithm, if the enclosure region $ER(i)$

Figure 3.14: “Blind Corner” of node i

of node i is bounded, it checks all the angles formed by any two consecutive relay candidates on the relay polygon $RP(i)$. If the angles are all no more than $\frac{2}{3}\pi$, it will keep the current transmission power for beaconing; If one of the angles α is more than $\frac{2}{3}\pi$, node i will compare the distance between these two consecutive relay candidates with $2 \sin \alpha R_{max}$. From the previous section, we know that if the distance is more than $2 \sin \alpha R_{max}$, the “Blind Corner” exists. So node i is required to beacon with the maximum transmission power P_{max} to check whether these “Blind Corners” contain any other nodes. If some new nodes are found in the “Blind Corners”, node i will add these nodes into $N(i)$ and judge whether they are relay candidates of node i . During this process, if some new relay candidates are found, the transmission power of node i will increase to some value that can maintain the new relay candidates. In order to guarantee the global connectivity of network, this checkup operation will be done after beaconing in each time interval.

3.2.3 Relay Topology Control Algorithm for Power-Efficient Routing

In this section, we present a location-based power-aware routing protocol based on RTC algorithm. Here, location information is used to achieve efficient and stateless routing when accommodating node mobility. The routing operates on

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL
AND ROUTING ON MANETS

the *Relay Graph* that is constructed by the RTC algorithm. So the packets are relayed through the relay candidates, which not only reduce the transmission power but also form a locally optimal power-efficient end-to-end route. Like most location-based algorithms [27, 30], the geographic forwarding may fail when the route falls into a “void” [27]. When the forwarding fails, we adopt a similar approach in GPSR [27] to continue the routing until the packets reach the destination node. Our routing protocol is distributed and loop-free, which is suitable for the mobile wireless networks.

In our routing protocol, a greedy heuristic algorithm is adopted to make the locally optimal choice for choosing the next forwarding node. We call our routing protocol *Greedy Power-Aware Routing* (GPAR for short). In the following, we will describe the details of GPAR.

Let node s denote the sender and node t the destination. Node s first searches for its relay candidates in its neighbor set by using the *Search for Relay Candidates* algorithm. Node s checks all the nodes in its neighbor set $N(s)$ to choose the relay candidates who can dominate node t . There are three cases when choosing a relay candidate as the next forwarding node.

Case 1: If the destination node t is dominated by only one neighbor r_1 of s , as shown in Figure 3.15, we have to see whether it is a relay candidate of node s . If node r_1 is a relay candidate in $RC(s)$, then it will be chosen as the next hop to forward packets to. If r_1 is not a relay candidate in $RC(s)$, we let r_1 be the destination and apply the greedy algorithm to find a suitable relay candidate in $RC(s)$ who can dominate node r_1 in its relay region.

Case 2: If the destination node t is dominated by several neighbors of s , e.g. $r_1, r_2, \dots, r_m$, as shown in Figure 3.16, we adopt a greedy approach to make routing decisions, and choose the neighbor r_k ($1 \leq k \leq m$) that has a minimum power in the routing path $s \rightarrow r_k \rightarrow t$ among all relay candidates. If r_k is a relay candidate in $RC(s)$, r_k is chosen as the next hop to forward

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

Figure 3.15: Destination node t is only in $RR(s, r_1)$

packets to; if r_k is not a relay candidate in $RC(s)$, we let r_k as the destination to run the greedy algorithm again until a suitable relay candidate is found in $RC(s)$ which covers r_k in its relay region.

Figure 3.16: Destination node t is in $RR(s, r_1) \cap RR(s, r_2)$

Case 3: If the destination node t is dominated by none of its neighbors of s , we cannot find a neighbor of s that can be relayed power-efficiently. As shown in Figure 3.17, the shaded areas denote the “blind areas” of relay regions of sender s . When the destination node t lies in these areas, it is not dominated by any relay candidates and the greedy forwarding fails.

We adopt a similar approach as proposed in GPSR [27] to find a path to the destination when the greedy forwarding fails. In GPSR, a packet would be forwarded in “Greedy” mode initially. When the greedy forwarding fails, the forwarding strategy would be switched into “Perimeter” mode, in which a

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL
AND ROUTING ON MANETS

Figure 3.17: Failure of greedy forwarding ($k = 2$)

right-hand rule is designed to forward packets in this mode. In GPSR [27], it was shown that the perimeter forwarding is loop-free and efficient in finding a feasible path. In our routing protocol, there are also two forwarding modes: “greedy”, and “perimeter”. Initially, a packet is set as “greedy” mode and forwarded to its relay candidates by GPAR algorithm presented in previous section. The GPAR routing continues until it fails. There are situations that may cause routing failures, such as incorrect forwarding decisions in the sender nodes, or inaccurate location information of the destination. In order to mitigate the effects of inaccurate location information, the relay node first updates the location of destination when it can not locate the next hop using the GPAR algorithm, then proceeds to examine whether greedy forwarding is possible after the location update. If it is still impossible, the relay node will switch the packet into the “Perimeter” mode. Similar to GPSR, the packet in this mode will be forwarded according to the right-hand rule on a planar graph (*Relay Graph*) until either the destination is reached or a greedy forwarding would be applied again.

The pseudocode of full GPAR algorithm, executed on each of the forwarding nodes, is presented in Figure 3.18.

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL
AND ROUTING ON MANETS

```

GPAP(s, t)
{
  Candidate(s, t) =  $\emptyset$ ;
  for  $n_i \in N(s), i \in \{1, 2, \dots, j\}$ 
    if  $t \in RR(s, n_i)$ 
      Candidate(s, t) = Candidate(s, t)  $\cup \{n_i\}$ 
  if there exist Candidate(s, t) =  $\{r_1, r_2, \dots, r_m\}$   $m \leq j$ 
    such that  $t \in RR(s, r_1) \cap RR(s, r_2) \cap \dots \cap RR(s, r_m)$ 
    then choose the candidate  $r_h, h \in \{1, 2, \dots, m\}$  such that
       $P(s \rightarrow r_h \rightarrow t) = \min P(s \rightarrow r_i \rightarrow t)$  for  $i \in \{1, 2, \dots, m\}$ 
    if  $r_h \in RC(s)$ 
      return  $r_h$ ;
    else
      GPAP(s, r_h);
  else
    Greedy forwarding fails;
}

```

Figure 3.18: Greedy Power-Aware Routing algorithm

Theorem 3.3 *The route produced by the greedy routing algorithm is loop-free.*

Proof: Let node s denote a sender, and t denote a destination node. Assume that this routing algorithm is not loop-free and there exists a loop from node s to t . Let $t_1, t_2, \dots, t_m$ be the nodes on that loop path. From the routing algorithm we have:

$$t_m \in RR(s, t_{m-1}), t_{m-1} \in RR(s, t_{m-2}), \dots, t_1 \in RR(s, t_m).$$

From Definition 3.1, we get:

$$\begin{cases} P(s \rightarrow t_{m-1}) + P(t_{m-1} \rightarrow t_m) \leq P(s \rightarrow t_m) \\ P(s \rightarrow t_{m-2}) + P(t_{m-2} \rightarrow t_{m-1}) \leq P(s \rightarrow t_{m-1}) \\ \dots \\ P(s \rightarrow t_m) + P(t_m \rightarrow t_1) \leq P(s \rightarrow t_1) \end{cases}$$

Combining all these inequalities, we get:

$$P(s \rightarrow t_m) + P(t_m \rightarrow t_1) + P(t_1 \rightarrow t_2) + \dots + P(t_{m-1} \rightarrow t_m) \leq P(s \rightarrow t_m).$$

So $P(t_m \rightarrow t_1) + P(t_1 \rightarrow t_2) + \dots + P(t_{m-1} \rightarrow t_m) \leq 0$, which is impossible.

The routing algorithm is loop-free. ■

3.3 Simulation Results

In this section, we simulate our algorithms to measure the impact on some performance issues, such as network connectivity, power consumption, etc.

3.3.1 Simulation Environment

Our simulation is implemented based on Simjava [54], a discrete event simulation package for Java. The simulation emphasizes on the topology control algorithm and packet forwarding protocol while assuming other low level parameters at reasonable values. Our simulations provide a rather wide range of variables such as node density, adjustable transmission radius as well as variations of routing methods. The details of the simulation environment are described in the following.

Simulation Model — We test five sets of nodes – with number 20, 60, 100, 140 and 200 – which are located in an area of 800×600 by a randomized uniform distribution. This covers both sparse and dense network settings. We choose a sufficiently large radio transmission radius to provide strong connectivity in a static network. For each density, 30 independent samples are tested, and most of the nodes are generated before simulation starts and the rest are generated randomly during the simulation time. We simulate the traffic flow in a pattern where each node will send a packet over the network to a randomly chosen destination. The packets are dropped when a timeout is encountered. The nodal delay per packet (for example, MAC layer processing time, the computation time) is set at 0.5s, which is adequate for the study but may not be realistic. The original packets are scheduled at a random uniform distribution from 0s to 10s. A simulation is ended when no future event occurs.

The main parameters are summarized in the following table.

Computational Methodology — In our simulation, the energy is measured as a quadratic function of transmission distance ($P \sim d^2$, where d de-

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

Parameter	Value
Network Region	$800 \times 600 (m^2)$
Number of Nodes	20, 40, 80, 120, 160
Number of sample files	30
Nodal delay per packet	0.5s
Scheduled time of packets	$0 \sim 10s$

Table 3.1: Main parameters of simulation model

notes the packet transmission radius). The hop number is counted for all packets in a simulation. The final result is averaged over data from 30 independent samples.

3.3.2 Simulation Results and Discussion

A. Connectivity

In this section, the purpose of the experiment is to study the effect of varying nodes' transmission radius on network connectivity. We vary nodes' transmission radius in the range from 50 meters to 400 meters and test the network connectivity in the five cases of node density. The transmission radius is step-incremented at 50 meters. We assume that no nodes fail or disappear under any circumstances. The simulation results are shown in the Figure 3.19.

From Figure 3.19, we can see that increasing transmission radius may result in increasing of the network connectivity. Moreover, the network connectivity will not further increase when certain thresholds of transmission radius is exceeded in different cases. Therefore, the networks in different cases will achieve strong connectivity when the transmission radius of nodes in this network reaches or exceeds the threshold. In our simulation, we treat the threshold of transmission radius in each node density as an *optimal* transmission radius

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

Figure 3.19: Network connectivity vs. Transmission radius

for this case since the threshold is the smallest transmission radius which can ensure strong connectivity of the network. We summarize the thresholds of transmission radius of each case in Table 3.2.

Number of nodes	20	40	80	120	160
Node density ($\times 10^{-4}/m^2$)	0.4167	0.833	1.67	2.50	3.33
Transmission radius(m)	370	250	200	150	100

Table 3.2: Thresholds of transmission radius in different node density

From Table 3.2, we see that, the sparser the network, the larger its optimal transmission radius requires. In our simulation, we choose 400 meters as the maximum transmission radius for each node because it can ensure the network connectivity in any cases in our simulation.

In the following, we will measure the performance of our algorithms under three topology control scenarios (denoted by Max, Fixed and RTC), in which their transmission radii are maximum transmission radius, optimal transmission radius and the radius by RTC respectively. The Max scenario represents

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

the common case of transmission in MANET because in most routing protocols, nodes always use the maximum transmission radius to broadcast packets with few power-efficient considerations. The Fixed scenario represents a near-optimal case of transmission in MANET because the values of transmission radius are the smallest transmission radii that still ensure strong connectivity of the network. However, all these optimal values are measured from the experimental results with certain parameters and may be not applicable in the realistic environments. Here we just use them to evaluate the results of our topology control algorithm.

B. Efficiency of RTC Algorithm

In this section, the purpose of the experiment is to study the efficiency of the *Relay Topology Control* algorithm. Two kinds of graphs constructed in different topology control scenarios are shown in Figure 3.20; one is the *Unit Disk Graph* in which each node connects to all its neighbors, and the other one is the *Relay Graph* as defined in the previous section. Figure 3.20 shows that varying the transmission radius will have more effect on the unit disk graph, and less on the relay graph. These results can also be derived from Figure 3.22(a).

We measure the average radius in each node density by using the RTC algorithm. Figure 3.21 shows that, when the node density is low, the average transmission radius of RTC is less than the fixed transmission radius. With the increase of node density, the average transmission radius of RTC is only slightly more than the fixed one.

Figure 3.22(a) shows that using RTC algorithm can reduce the number of links in the network significantly, and hence reduce the cost of topology maintenance for each node. Moreover, the results in Figure 3.22(b) show that RTC algorithm can achieve approximately the same effect as the other two topology control strategies.

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL
AND ROUTING ON MANETs

Figure 3.20: Network topologies resulting from various strategies

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETs

Figure 3.21: Average radius in transmission (The unit of node density is $10^{-4}/m^2$, which is same in the following)

We also measure the average node degree when using the three kinds of topology control scenarios. The experimental results are shown in Figure 3.23(a) and Figure 3.23(b). The node degree of the mobile ad hoc networks should not be too large because a large node degree typically implies that a node has to communicate with more distant nodes directly, which increases interference and collision. The average degree should not be too small either because that tends to increase the overall network power consumption as longer paths have to be taken. So we believe that average node degree is an important metric for mobile ad hoc network topology.

When using RTC and Fixed transmission radii, the average node degree can be decreased significantly compared with using maximum transmission power. The average node degree in RTC is very close to the one in Fixed scenarios

(a): Total number of links in Unit Disk (b): Total number of links in Relay Graph

Figure 3.22: Total number of links in Unit Disk Graph and Relay Graph

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

(a): Average node Degree in Unit Disk (b): Average node Degree in Relay Graph

Figure 3.23: Average node degree in Unit Disk Graph and Relay Graph

when the node density is not very high. However, the difference between them becomes larger when the node density increases. In RTC algorithm, the nodes that lie near the boundaries of the network contribute more to the average degree statistics. This is because most of such nodes are not internal to their enclosures, but they are required by the RTC algorithm to increase their transmission radius to check whether they are internal to their relay polygons until the maximum transmission radius is reached. So we remark that the average node degree can be reduced if we know the boundary of the network. When a node finds itself is very near the boundaries of the network region, it is not required to use the maximum transmission radius. We leave the research of further reduction of node degrees to the next stage.

C. Power Consumption

In this section, the purpose of the experiment is to evaluate the effect of RTC and GPAR algorithms on power consumption.

In our simulation, we only simulated a static network. If mobility is low, a proactive approach to update the network topology may be used. If the mobility is high, an on-demand approach to update the network topology may be the only viable way to keep the update control traffic low.

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETs

(a): Power consumption for transmission (b): Power consumption of GPSR and GPAR

Figure 3.24: Power consumption in transmission and routing

In our simulation, we do not simulate the process of adjusting to the right transmission radius. It is adjusted to the right transmission radius immediately after executing RTC algorithm.

We first test the power consumption for transmission when using different topology control scenarios. The results are shown in Figure 3.24(a). In the max topology control scenario, each node always broadcasts with the maximum transmission power. So the average power consumption for transmission remains as a constant and is most significant. The average power consumption for transmission in the RTC algorithm is reduced significantly compared to the maximum transmission case, and tends closely to the topology in Fixed scenario. From the results, we can see that our RTC is effective in reducing power consumption for transmission.

Figure 3.24(b) shows the average power consumption for routing when using GPSR and GPAR routing protocols in different topology control scenarios. From the figure, we can see that our topology control algorithm also reduces the power consumption significantly in routing process and tends closely to the Fixed scenario. We also tested our GPAR protocol in these different scenarios. Since GPAR routing protocol is operated only on relay graph, and the difference of number of links of relay graph is not obvious when using different topology control scenarios, the difference of power consumption of GPAR in

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL AND ROUTING ON MANETS

Figure 3.25: Hop count in different cases

these cases will also be non-obvious. So, here we just test the scenario of maximum transmission radius. The result shows that GPAR is more power-efficient than all the other cases.

In multihop wireless network, hop count can be used as a metric to measure routing latency when the impact of congestion is not significant. Here we test the total hop count in each scenario. Figure 3.25 shows that GPAR uses the largest hop count in all cases. Therefore, the latency in GPAR will increase significantly in contrast to GPSR. A possible reason of this result is that, in GPSR, each node always sends packets to the node who is nearest to the destination but far from the sender, while in GPAR, this direct forwarding is replaced by routing through more nodes to achieve the locally minimum power transmission. Due to the mobility feature of MANET, long latency may decrease the successful delivery rate of packets and cause more retransmissions which further increase the latency. In [39], Luan proposed a strategy which can achieve a good balance between power consumption and latency in GPSR. However, her result is based purely on experiments. How to theoretically analyze the relations between these metrics will be our future research.

3.4 Chapter Summary

In this chapter, a distributed topology control algorithm has been proposed to improve the power efficiency of routing algorithms in mobile ad hoc networks.

CHAPTER 3. A LOCALIZED TOPOLOGY CONTROL ALGORITHM FOR POWER CONTROL
AND ROUTING ON MANETS

In the algorithm, each node makes local decisions about its transmission power and these local decisions collectively retain global connectivity. The power control approach not only reduces the power consumption in transmission, but also retains the global network connectivity so that the communication routes between nodes will not be disconnected due to the power control. We also apply our topology control algorithm in some geographic routing protocols to show that the routes in the multihop network are efficient in power consumption. Finally, simulation results demonstrate significant performance improvement.

Part of this chapter has been published in [17].

Chapter 4

A Geographic Routing Protocol Based on Approximate Location Information for Mobile Ad Hoc Networks

Recent research has shown that routing can be more efficient by using location information in Mobile Ad Hoc Networks (MANETs). Geographic routing protocols of MANETs can achieve good scalability with less routing overhead compared with the topology-based routing protocols. They are based on the location services to provide location information. However, frequent changes of nodes' locations increase the cost of maintaining accurate location information, and also decrease the robustness of routing. So how to make geographic routing efficient and robust without the heavy overhead is still a challenge in MANETs. In this chapter, we propose a geographic routing protocol (called Fisheye-like Location Information based Routing Protocol (FLIRP)). The protocol is inspired by the fisheye technique, in which each node maintains accurate location information of nearby nodes, and approximate location information of faraway nodes. For each node, the accuracy levels of the other nodes'

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

location information depend on their location in a hierarchical grid partition of the network. The routing protocol is based on the approximate location information and greedy forwarding. It not only reduces the update messages overhead, but also is adaptive to the mobile environments. The simulation results indicate that our protocol is well suited to the ad hoc networks where nodes move at high speed.

4.1 FLIRP Algorithms

Our scheme is inspired by the fisheye technique proposed by Kleinrock and Stevens in [29], where the technique was used to reduce the size of information required to represent graphical data. The eye of a fish captures higher level of detail of the scenes near its focal point. The level of detail decreases as the distance increases. In our protocol, the network area is partitioned into a hierarchy of grids. The fisheye approach is translated to maintaining accurate location information about the immediate neighborhood of a node in the same unit square region, with progressively less details as the size of squares increases. In the following, we will describe the FLIRP algorithms in details.

4.1.1 Network Partition and Addressing

We model MANET as a directed graph $G = (V, E)$, where V, E denote, respectively, the set of nodes and connections among these nodes in the network. Moreover, all N nodes in the set V are in a two-dimensional area, where no two nodes coincide at the same point. We assume that each node has a unique ID and can get its own geographical location through the support of GPS or some other means.

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

(a): Entire network partitioning (b): Addressing method for partitioning regions

Figure 4.1: Network partitioning and addressing

Network Partition

Similar to GLS [33], we partition the network area into a quadtree-like hierarchy of squares. As shown in Figure 4.1 (a), the entire region referred to as the order-0 region, is partitioned into 2×2 order-1 regions, which are further partitioned into 2×2 order-2 regions, and so on. Regions with the highest order in the network are said to be *unit regions*. In Figure 4.1 (a), the order-3 regions are the unit regions.

Figure 4.2: Relationship between the size of a unit region and R_{max}

We assume that the maximum transmission range R_{max} of all nodes in the network are identical. In order to avoid message flooding, a unit region in the network is covered by the maximum transmission range of the nodes in the region. As shown in Figure 4.2, the unit region is a $\sqrt{2}R_{max}/2 \times \sqrt{2}R_{max}/2$ square. Thus in a unit region, each node can get the location information of

the other nodes by one-hop broadcasting, this can avoid messages flooding in the unit region and reduce the cost of region maintenance.

Addressing of Hierarchical Regions

With the quad-tree partitioning method for the entire network, the location of a node can be identified by the region it resides in. In order to address nodes, we label the regions according to their orders and locations. As shown in Figure 4.1 (b), the four order-1 regions are identified by 1,2,3,4 in clockwise direction. The same labeling approach is adopted for regions at different orders. Finally, the address of a given region can be represented like an IP address. For example, the addresses of the shaded regions R_1 and R_2 in Figure 4.1 (b) are (1.3.4) and (3.1) respectively. If two regions are in a same order- n region, the two regions are said to be *sibling regions in order- n* . For instances, the regions in {1,2,3,4}, or in {1.1,1.2,1.3,1.4}, or {1.1.1,1.1.2,1.1.3,1.1.4} are all sibling regions in different orders. In our protocol, we assume that all nodes know the same global partitioning of the network, and they can also get the unit region address of a node according to its current location.

4.1.2 Fisheye-like Location Information

In Fisheye State Routing (FSR) [46], each node exchanges information about nearby nodes more frequently than it does about farther nodes, thus reducing the update message size. So each node gets accurate information about its neighbors and the level of detail and accuracy of information decreases with the distance from the node concerned. However, using this approach, nodes still need to flood the entire network to collect the information of all the other nodes in the network, which decreases efficiency of the network. In FLIRP, we propose a localized update mechanism to avoid message flooding. It is a proactive information update mechanism, in which each node exchanges information with its neighbors (nodes within its radio range) periodically.

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

A node a need not maintain the accurate location information of all the nodes in the network. Like a fisheye, the closer the nodes to node a , the more accurate location information that node a has about them. However, different from fisheye technique, we use a quadtree-like hierarchy to represent the accuracy levels of the location information instead of physical distances. Let r represent the *lowest common ancestor (LCA)* in the hierarchy containing nodes x and y . If r is of the highest order, then x and y are in the same unit region and they keep the accurate location information of each other; otherwise, x just keeps the ID of the largest subregion under the LCA that contains y as an approximate location of y . We call these location information *Fisheye-like Location Information (FLI)*.

Figure 4.3: Fisheye-like location information

Example: In Figure 4.3, node S and four other nodes A, B, C, D reside in the same order-2 region (4.2) which is a unit region. Node S maintains accurate location information of these four nodes. For the nodes residing in the sibling order-2 regions of region (4.2) such as nodes E, F, G , S only maintains the addresses of the order-2 regions they residing in, which are (4.1), (4.3), (4.4) for E, F, G respectively. Similarly, for nodes H, I, J who are residing in the sibling order-1 regions, node S keeps their order-1 regions' addresses (1), (2), (3) respectively. ■

4.1.3 FLI Setup and Update

At startup, each node (such as node S) periodically sends out beacons to announce its existence and location. Other nodes within the radio range will receive the beacons and record S as a neighbor stored in their neighbor tables. In FLIRP, each node maintains two kinds of tables: one is the *neighbor table*, in which a node stores the IDs and location information of other nodes within its radio range. The other table is the *FLI table*, in which a node stores the fisheye-like information of other nodes in the network.

After the neighbor table is set up, each node will have a full node list of the unit region it is residing in. In order to explain how nodes exchange their node list effectively, we give some definitions in the following.

Assume that the highest order is h . In a order- i ($0 \leq i < h - 1$) region, the four unit regions that are surrounding the center of the order- i region are defined as *central regions in order- i* . In Figure 4.4, the shaded unit regions are all central regions in different orders. In each central region, the node that is closest to the center of the order- i region acts as a *coordinator* to generate and exchange the node list of the order- $(i + 1)$ region it residing in, such as nodes A1-A4 in the order-1 region (4), nodes B1-B4 in the order-0 region. These coordinators like A1-A4 and B1-B4 are called *sibling coordinators*. If there exist no nodes in a central region, the coordinator will be selected from the other unit sibling regions closest to the center. Each coordinator is in charge of generating a node list of the order- $(i + 1)$ region it is residing in. The node list is denoted by $NL(addr)$, where $addr$ is the address of the region it is in charge of. As in Figure 4.4, nodes A1-A4 will generate the node lists $NL(4.1), NL(4.2), NL(4.3), NL(4.4)$, and nodes B1-B4 will generate the node lists $NL(1), NL(2), NL(3), NL(4)$ independently. A node list only contains all the nodes' IDs in its responsible region, and broadcasts within this region. A same node list will not be rebroadcasted by a node who has received it.

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

Figure 4.4: Central regions and coordinators

After nodes setup their neighbor tables, the coordinators A1-A4 generate node lists of the order-2 regions they are in charge of. They exchange these nodes lists with each other and broadcast them to the other nodes in their responsible regions. When a node receives the node list, it stores the list into its FLI table with a timestamp. When nodes B1-B4 receive all the node lists from other three sibling order-2 regions, they will generate nodes lists of the order-1 regions they are responsible for. Likewise, the node lists will be exchanged. After the node lists in all levels are exchanged, each node will maintain location information of the other nodes in the network.

When a node moves away from the previous unit region, it will send out a beacon to update its new location. If the beacon can still be received by some nodes in that previous unit region, those nodes can conclude that the node has moved away and can safely remove the node from their neighbor table after receiving its beacons. Otherwise, the absence of periodic beacons triggers neighbor expiration. If a node does not receive beacons from a known neighbor for certain time, the node removes the neighbor from its neighbor table. When a coordinator finds its node list changed, such as new nodes join, or some nodes move out, it will incorporate the changed parts of node list into its periodic update packets and broadcast to its neighbors. This approach can reduce the

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

size of update message and save network bandwidth. The update message will be exchanged with other sibling coordinators. This update mechanism implies that if a node only moves within a order- i region, information will be updated only within this region, and no other regions need to be updated. As shown in Figure 4.4, if node S moves within the region (4.1), only the nodes in region (4.1) need to update respectively. If node S moves out of region (4.1) and enters the new region (4.4), the nodes in region (4) have to update.

4.1.4 Geographic Routing with Stepwise Refinement

As described in the preceding section, if the destination and source nodes are not in a same unit region, the sender only maintains an ID of the region the destination node residing in. So the center of the region will be regarded as the destination. As a packet gets closer to the destination, the intermediate nodes will have more accurate location information of the destination. Similar to geographic forwarding, the greedy forwarding suffers from the “void” problem [27]. When the greedy forwarding fails, a backup forwarding strategy such as perimeter routing in GPSR [27] is adopted to solve the problem.

Example: In Figure 4.5, when node S sends query packets to the destination node D , it first looks up its routing table and finds that S only maintains the

Figure 4.5: An example of stepwise refining geographic routing

address (2) of the order-1 region D residing in. As shown in Figure 4.5 (a), then S forwards packets towards the center of region (2), i.e. O_1 . By greedy forwarding, S selects node A that is closer to O_1 as its next forwarding node. Since nodes A,B,C are in a same order-1 region with S, they all maintain the same location information of node D. The query packets are forwarded from node S to C with a same destination O_1 . As shown in Figure 4.5 (b), when packets reach node E (a neighbor of node C) which is in the same order-1 region with D, node E knows D's location (2.3) more accurately than the previous four nodes. So node E forwards packets towards the center O_2 of region (2.3) via nodes F and G. In the same way, when the packet reaches node G, it forwards the packet towards the center O_3 of region (2.3.2) as shown in Figure 4.5 (c). Finally, the packet reaches the destination node D via node H as shown in Figure 4.5 (d). After the destination node D receives the query packet, it can get the location information of node S contained in the query packets and send the reply packets with its current location information to the sender node S by using geographic forwarding directly. ■

The stepwise refining geographic routing is actually based on GPSR protocol [27]. However, approximate location information instead of accurate location information is used to adapt to mobile environments. In FLIRP, if nodes in the network are static or in low speeds, they can use the successive geographic routing as a location query in the first attempt. After getting the accurate location information of the destination, they can forward the subsequent packets directly. If node mobility is high, they can still rely on the stepwise refining routing.

4.2 Performance Analysis

In this session, we will analyze the FLI maintenance cost and space required of the table recording each node's location in FLIRP.

(a): Relationship between a region and its circles
(b): Average distance travelled by a node within a region

Figure 4.6: Analysis of the cost of FLI maintenance

In the network, there are N nodes uniformly distributed in a two-dimensional space of area A with an average of $\lambda = N/A$ nodes per unit network area. The transmission range of each node is unique and circular with the radius r_t . A denotes area of the network. v denotes the average speed of nodes in the network. In every t_b units of time, each node broadcasts update messages to its neighbors.

4.2.1 Cost of FLI Maintenance

In FLIRP, a node needs to send update messages in two cases: one is the periodic update of its current location, the other one is when the node moves into a new region, it will generate a location update to announce its new location. So the cost of location maintenance C_u is $O(C_p + C_m)$ where C_p is the cost of periodic update and C_m is the cost of update due to node mobility.

Since each node beacons once in t_b , the number of messages for the periodic update per node per second C_p is

$$C_p = \frac{1}{t_b}.$$

which is a constant determined by the beaconing interval t_b .

When a node moves into a new region, it sends a location update message to its neighboring nodes. In order to analyze the mobility cost per node per

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

second C_m , we must know the average time t_m of one node crossing a region. Here we use $\bar{d}$ to denote the average distance traversed by a node within a region.

In Figure 4.6 (a), the area of a square region with l of each side can be surrounded by its circumcircle and inscribed circle. To simplify the analysis, we assume a circular region of radius r instead of the square region, where r satisfies

$$\frac{l}{2} < r < \frac{l}{\sqrt{2}}$$

As mentioned early, $l = \frac{R_{max}}{\sqrt{2}}$. So we can get

$$\frac{\sqrt{2}}{4}R_{max} < r < \frac{1}{2}R_{max}.....(1)$$

As illustrated in Figure 4.6 (b), once a node enters a region, it travels for some time within the region before exiting. The distance the node travels while in the region can be seen to be

$$d = 2l \sin \theta$$

where θ is the angle the velocity vector makes with the tangent at the point of entry into the region. The average distance $\bar{d}$ traveled by a node within a circular region is given by

$$\bar{d} = \frac{2}{\pi} \int_0^{\frac{\pi}{2}} 2r \sin \theta d\theta = \frac{4r}{\pi}.$$

Therefore, the average time t_m spent in a circular region is given by

$$t_m = \frac{\bar{d}}{v} = \frac{4r}{\pi v}.$$

So the mobility update cost per node per second C_m in terms of number of messages is given by

$$C_m = \frac{1}{t_m} = \frac{\pi v}{4r}.....(2)$$

Combining (1) and (2), we get

$$\frac{\pi v}{2R_{max}} < C_m < \frac{\pi v}{\sqrt{2}R_{max}}$$

So $C_m \propto v$

Finally, we get the cost of location maintenance $C_u = \Theta(\frac{1}{t_b} + v)$. When t_b is constant, the cost of FLI maintenance is $\Theta(v)$.

4.2.2 Performance of FLIRP Routing Path Length

In FLIRP, each node uses the stepwise refining geographic routing to forward packets. Since such geographic routing scheme is based on the approximate location information maintained in each node, the performance of routing path may be not as good as the typical geographic routing schemes with accurate location information. In this section, we try to analyze the routing path performance of FLIRP and find an upper bound for it.

Lemma 4.6 *For any pair of source node S and destination node D , the length of FLIRP routing path is upper-bounded by the length of the diagonal of the entire region.*

Proof: We prove this by mathematical induction on the level of LCA of S and D .

Base case: Assume that both S and D are in a unit region. In this case, it is clear that the diagonal of the unit region has the maximal length among all the line segments connecting S and D , as shown in Figure 4.7.

Inductive step (LCA in level- k): Assume that the lemma holds when the LCA of S and D is a level- k region, we need to prove that the lemma is still true in the case that the LCA is a level- $(k + 1)$ region.

Figure 4.7: Nodes S and D in a same unit region.

Inductive step (LCA in level- $(k + 1)$): Suppose A and O are the end points of a diagonal of a level k region R_S , and S is any point in a level- $(k + 1)$ (as long as it doesn't fall in the same level- k region where the destination D lies in). X is the point where the packet from S enters the level- k region R_D containing D . Extend the line segment XS and call the intersection of XS and the level $k + 1$ boundary Y . Thus, $|SX| \leq |XY|$ holds and the equality holds if and only if S is coincident with Y . Consider the following two cases:

Figure 4.8: The LCA of S and D is level- $(k + 1)$.

(a) X and Y are in a same level k region. In this case, definitely $|XY|$ is no more than the length of the diagonal of the level k region which is equal to $|AO|$. The equality holds if and only if X and Y are the end points of the diagonal.

(b) X and Y are not in a same level k region. Figure 4.8 gives an example. Add the auxiliary line XZ as shown in Figure 4.8. So $|XY| = |XZ|/\cos \alpha$. When $\alpha = \frac{\pi}{4}$, $|XY|$ gives the maximal value, i.e. exactly when X coincides with O , and Y coincides with A . In other words, $|XY| \leq |AO|$.

Therefore, in both cases, $|SX| \leq |AO|$. With analysis, it is also clear that AO is the longest path for a packet to travel in order to enter the level k region where D resides in.

Combining this result with the hypothesis, we can claim that a packet moving along the diagonal of the entire region travels the longest path. So the lemma holds.

■

4.3 Simulation Results and Evaluation

The section presents simulation study of the scalability and robustness of FLIRP. Good scalability means that protocol overheads of each node do not rise quickly as functions of the total number of nodes. While good robustness means that protocol overheads of each node do not rise quickly as a function of the average node speed. Details of the simulations are described in the following.

4.3.1 Simulation Setup

The simulation is implemented based on Simjava [54], a discrete event simulation package for Java. Each node can get its current location (denoted by a coordinate (x, y) in a plane) anytime anywhere. The IEEE 802.11 radio and MAC model are used by each node with a radio range of 250 meters. The nodes are placed at uniformly random locations in a square network. Five

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

Size of networks	No. of nodes	Network density	Highest- order of hierarchy
700×700	50	102.04	2
1000×1000	100	100	2
1400×1400	200	102.04	3
1700×1700	300	103.81	3
2000×2000	400	100	3

Table 4.1: Network configurations with approximate density

network configurations are shown in Table 4.1. The size of each network region is chosen to maintain an average density of around 100 nodes per square kilometer as in SLURP [69]. The size of a unit region is from 175 to 250 meters on a side since the probability of connectivity is very high within this radio range, as shown in Figure 4.9. From Table 4.1, we can see that orders of the square regions in the network configurations are different. For the networks of 50 and 100 nodes, the grid hierarchies go up to order-2. For 200,300 and 400 nodes, they go up to order-3.

Figure 4.9: Probabilities of connectivity when randomly place 2 to 6 nodes in a square of size from 150 to 500 meters on each side.

Each node moves using the *random waypoint* mobility model [4]. In this

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

model, a node chooses a random destination and moves toward it with a constant speed uniformly distributed between zero and a maximum speed (up to 30 m/s in the simulations). When the node reaches the destination, it chooses a new destination and begins moving towards it immediately without any “pause” time.

The nodal delay per packet (for example, MAC layer processing time, the computation time) is set randomly from 0 to 0.1 second. The beacon interval is 1 seconds which means each node will send a HELLO message in every second after simulations run. A neighbor will be expired by a node if no beacons are received in 3 seconds. For each simulation, 100 independent samples are tested.

4.3.2 Protocol Traffic

In this section, we will evaluate FLIRP performance in terms of protocol traffic.

Location Update Cost

As mentioned in FLIRP, each node generates two types of location update messages: one is the periodic message which is just sent to its neighbors and not forwarded to other nodes, and the other is the update message due to node mobility, which is broadcasted to the other nodes.

Cost of location update with respect to number of nodes - Figure 4.10 presents the results of number of location update packets per node per second when the network size changes. We see that varying the number of nodes has little effect on the cost of location update. However, when the node speed increases, the cost of location update also increases respectively. When the nodes are static in the network, there are no mobility updates and only periodic updates are needed. When the node speed increases, the mobility updates also increase since a node moves into a new region more frequently. In

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

Figure 4.10: The average number of update packets per node per second does not vary much over network size

FLIRP, each node only updates to its neighbors by broadcasting. The network size will have less effects on the cost of update.

Figure 4.11: The average number of update packets per node per second of SLURP and FLIRP as functions of node speed. The network configuration is 400 nodes in a 2000×2000 area.

Cost of location update with respect to node speed - Figure 4.11 shows the result of location update cost in FLIRP and SLURP. The result shows that the location update cost of SLURP is 10%-25% higher than that of FLIRP, and increases quickly when nodes move faster. While the cost of location update of FLIRP increases moderately as node speed increases. This is because, in SLURP, each node has to forward update packets to the location servers in its home region, and the forwarding overhead increases

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

with node mobility. While in FLIRP, nodes use a localized update approach to maintain the variable location information about the other nodes relative to their distance. So this can significantly reduce the network overhead and achieve better scalability.

Cost of Location Query

As mentioned, in a low mobility environment, we use stepwise refining geographic routing to query the location information of destination in the first try. In this section, we study the cost of location query of FLIRP by varying the network size and node speed. In this experiment, we will select pairs of nodes and assign half of them as source nodes and the others as destination nodes. The source nodes send query messages to get the location information of their responding destination nodes. When the destination nodes receive the query messages, they reply by using greedy forwarding directly. The number of location query packets per node consists of the query packets and the reply packets per node.

Figure 4.12: The average number of query packets per node per second as a function of the total number of nodes.

Cost of location query with respect to number of nodes - Figure 4.12 shows the average number of location query packets originated or forwarded per node per second as a function of the total number of nodes.

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

The cost of location query increases when the number of nodes increases, since query packets need to be forwarded via more hops when the network size increases. The experimental results show that, although node speed affects the cost of location query, the effect is not significant.

Figure 4.13: The average number of query packets per node per second as a function of node speed. The network configuration is 400 nodes in the 2000×2000 area.

Cost of location query of FLIRP and SLURP with respect to node speed - Figure 4.13 shows the cost of location query in FLIRP and SLURP. When the node mobility is low, the cost of location query of SLURP is smaller than that of FLIRP. When node mobility increases, the query cost of SLURP increases more significantly than FLIRP. In SLURP, nodes send the query packets to the centers of the destination nodes' home regions, while in FLIRP, the query packets are forwarded to the centers of a large grid regions if the destination nodes are far from the source nodes. Since the forwarded region in SLURP is smaller than FLIRP, the number of hops forwarded in the query routes of SLURP is smaller than that of FLIRP when the node mobility is low. With the increasing of node mobility, in SLURP, more and more query packets fail due to the out-of-date location information maintained in the home regions, and the retransmissions of query packets cause the query cost increase. While in FLIRP, node mobility has little impact on location query and the cost of location query increases slowly.

4.3.3 Consistency of Location Information

The inconsistencies of location information may affect the performance and even the correctness of geographic routing [28]. So the correctness of location information is important. We define a “correctness ratio” as the ratio of the number of the correct location information of other nodes over the total number of instances of location information maintained by each node. In FLIRP, each node maintains the accurate location information of other nodes residing in a same unit region. For the nodes in different unit regions, it only maintains the corresponding IDs of these regions. For a node A , we judge a location information of node B is correct if the following two cases are satisfied: if both A and B reside in the same unit region, then A must have accurate information of B in its routing table; otherwise, A need only have the correct ID of the region that contains B .

Figure 4.14: Correctness ratio of location information for different network sizes.

Figure 4.14 presents the average correctness ratio of location information in different number of nodes and speed. The correctness ratio in each configuration is high, and stays relatively constant when varying the number of nodes. The correctness ratio decreases with the node speed increasing, and the differences become smaller when the number of nodes increases.

Figure 4.15 presents the average correctness ratio of location information for different orders of squares in cases of 200, 300 and 400 nodes. The correctness ratio decreases when the order of region increases, and the differences

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

(a): 1400×1400 , 200 nodes (b): 1700×1700 , 300 nodes (c): 2000×2000 , 400 nodes

Figure 4.15: Correctness ratio of location information for different orders of squares

enlarge with increasing of the orders. This is because FLIRP takes advantage of the distance effect. The size of the region increases as the order of region decreases. It is less likely for a node to move out of the boundary of larger regions, so the information remains accurate for a longer period of time.

4.3.4 Routing Performance

A location service can affect the routing performance in several ways. First, the stale information will cause nodes to choose a sub-optimal path. Second, the inconsistency of information at intermediate nodes may cause routing loop. Third, the stale and inconsistent information may provide no forwarding locations of the destination nodes, which causes routing failures. In the following, we will evaluate FLIRP routing performance in terms of packets delivery rate, routing latency and routing path.

Packets Delivery Rate

We evaluate the packet delivery rate with respect to the network size and node speed. In this experiment, we select pairs of nodes and assign half of them as source nodes and the others as destination nodes. Each source node sends 10 packets to its destination node. A routing packet will be dropped when the timeout is encountered. If a routing packet fails, it will not be retransmitted, so a successful packet delivery means success on the first attempt.

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

Figure 4.16: FLIRP packets delivery rate vs. number of nodes.

Figure 4.17: The packets delivery rates of FLIRP and SLURP as functions of nodes speed. The network configuration is 400 nodes in a 2000×2000 area.

Figure 4.16 presents the packet delivery rate of FLIRP in various number of nodes and speed. The packet delivery rate of FLIRP is very high. Although node density and speed can decrease the packet delivery rate, the impacts are not significant. In Figure 4.17, we compare the packet delivery rates of SLURP and FLIRP protocols. FLIRP can achieve higher packet delivery rate than SLURP. Moreover, the packet delivery rate of SLURP appears to decrease more quickly than FLIRP, as nodes move faster.

In FLIRP, a node only maintains the IDs of squares containing faraway nodes. The farther the nodes with the source node, the larger size of regions' IDs are maintained. Nodes moving within larger regions have no impact on the packets delivery rate. Most failures are due to either inaccurate and stale

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

information caused by nodes' moving out of squares or nodes not being correctly updated because of delayed or lost location updates. The result suggests that the FLIRP is robust in highly mobile environments.

Routing Latency

In multihop wireless network, hop counts can be used to measure routing latency when the impact of congestion is not significant.

Figure 4.18: Routing latency as a function of the number of nodes. All the nodes are static.

Figure 4.18 shows the routing latency of FLIRP and SLURP in number of hops for various network configurations. The routing latencies in FLIRP and SLURP are proportional to the number of nodes. The latency in SLURP is higher than FLIRP. Moreover, the latency in SLURP increases faster than FLIRP when the number of nodes increases.

The routing latencies in FLIRP and SLURP became longer when the number of nodes increases, because when the network size increases, a node will travel more hops to forward query packets to the destination node. In SLURP, if a node has packets for another node, it has to query the location information of destination firstly, and then forwards the packets to it by geographic routing. While in FLIRP, a node can make the routing decisions immediately without querying the destination's location. So the routing latency in FLIRP is shorter than SLURP.

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

Routing Path Length

Figure 4.19: Average routing path length as a function of the Euclidean distance between source and destination nodes.

Figure 4.19 shows how the distance that query packets travel compares with the direct distance between the source and the destination. The graph shows that the distance traveled by a routing packet between two nodes is proportional to the actual distance between these two nodes. Our simulation agrees with a theoretical analysis in [33] which predicts that, with a sufficiently dense uniform distribution, the query distance is proportional to the distance to the destination.

Figure 4.20: Average routing path lengths between FLIRP and GPSR.

Figure 4.20 presents the distance that FLIRP routing packets travel compares with the GPSR routing path between the source and the destination.

CHAPTER 4. A GEOGRAPHIC ROUTING PROTOCOL BASED ON APPROXIMATE
LOCATION INFORMATION FOR MOBILE AD HOC NETWORKS

The result shows that the FLIRP path length is around 10%-20% longer than the GPSR routing path length, and the difference increases with the network size. Recall that in GPSR, packets are forwarded by actual location information of the destinations. When the network size increases, the square sizes in different orders also increase which causes the routing packets to travel more hops if the source nodes are far from the destination nodes.

4.4 Chapter Summary

In this chapter, we have proposed a scalable and robust geographic routing framework using a fisheye-like location information service, in which each node maintains approximate location information of most other nodes in the network and only stores accurate location information of its nearby nodes. FLIRP not only reduces the update messages overhead, but also is adaptive to the mobile environments. The theoretical and simulation results all indicate that our protocol is well suited to large-scale ad hoc networks where nodes can move at high speed.

Part of this chapter has been accepted by Globecom'2004.

Chapter 5

Conclusions and Future Work

In this chapter, we summarize the work that has been done in the thesis. Discussion and recommendations for future work are presented.

5.1 Summary and Results

The main work in the thesis may be summarized as follows:

- A localized topology control algorithm (RTC) has been proposed. This algorithm shows the following features: (1) It is a distributed algorithm that is simple yet efficient and hence suitable for mobile ad hoc networks; (2) It can minimize power consumption in the transmission process, which indirectly increases the network lifetime. At the same time, the network connectivity can be guaranteed; (3) It can construct a network topology with small node degree, so that interference is minimal and hence throughput is sufficient. A power-aware routing algorithm (GPAR) has been proposed. This routing algorithm is derived from GPSR [27], yet it can achieve better power saving than GPSR protocol.
- We have presented a geographic routing protocol (called Fisheye-like Location Information based Routing Protocol (FLIRP)). This routing

protocol is based on the approximate location information and greedy forwarding. It not only reduces the update messages overhead, but also is adaptive to the mobile environments. The simulation results indicate that our protocol is well suited to the ad hoc networks where nodes move at high speed.

5.2 Discussion and Future Research

Despite partial success, the thesis can be enhanced in the following areas:

5.2.1 Extensions of RTC

- **Power Consumption vs. End-to-end Latency** — Our algorithm tends to choose routes with a higher number of hops, each of which having small distance, and it incurs longer end-to-end delays for the packet delivery. This can be seen from the experimental results in Chapter 3. End-to-end latency is an important metric to influence the performance of MANETs. In the future research, we will attempt to optimize and balance these two metrics.
- **Node degree** — In our topology control algorithm, nodes that lie near the boundaries of the network contribute more to the average degree statistics. This is because, most of such nodes are not internal to their enclosures, but they are required by the ETC algorithm to increase their transmission radius to check whether they are internal to their relay polygons until the maximum transmission radius is reached. The average node degree can be reduced if we know the boundary of the network. When a node finds itself to be very near the boundaries of the network region, it is not required to use the maximum transmission radius.

- **Robustness of path** — In our algorithm, the overall topology maintained is sparse. Although it lowers the cost of topology maintenance, it also lowers the number of choices in setting up paths and decreases the robustness of path. As a result, a link or node may be overwhelmed by multiple requests. So how to increase the robustness of a connection should be useful future work.

5.2.2 Range Query Service Based on FLIRP

In the location service schemes, nodes always try to get location information of some desired destinations. While in range query service, nodes try to find which nodes are in a specified range. In Figure 5.1, a node P inquires about all the nodes in the rectangle range $ABCD$ in a range query service.

In order to get the node information, we could flood the query packet to all the nodes in the network. For large networks, this approach is very inefficient. Another alternative approach is to send the query packet only to the limited range by geographic forwarding. For instance, node P can regard the center of range $ABCD$ as its destination to forward the query packet. However, without location information, it is difficult to judge whether the query packet has covered the entire region of $ABCD$.

Figure 5.1: An example of range query

Figure 5.2: Range query based on FLIRP

As shown in Figure 5.2, with FLIRP, P knows that range $ABCD$ is within region (1) and (2). It can send query packets to region (1) and (2) respectively. When the query packets reach these regions, the intermediate forwarding nodes will have more accurate information about the regions $ABCD$ intersects, such as the shaded regions in Figure 5.2. Then the query packets will be forwarded to these regions until reaching them. Since the nodes in the shaded regions know of the nodes in these regions, they can check which nodes are in the range $ABCD$ accordingly.

5.3 Concluding Remark

In this thesis, we have investigated two important issues of mobile ad hoc networks: energy-efficiency and mobility-robustness. The research done so far is just a first step of such exploration. We sincerely hope that our work will be a useful basis for future research and developments of mobile ad hoc networks.

Appendix

Auxiliary proof for Lemma 3.1 (1): $(d^n(s, r) + d^n(r, t))^2 \leq (d^2(s, r) + d^2(r, t))^n$, for $n \geq 2$.

Proof: Assume that $d(s, r) = C \cdot \cos \gamma$ and $d(r, t) = C \cdot \sin \gamma$, for $0 < \gamma < \pi/2$. So $d(s, r), d(r, t) < C$.

By mathematical induction, (1) when $n = 2$, obviously, $(d^n(s, r) + d^n(r, t))^2 = (d^2(s, r) + d^2(r, t))^n$; (2) when $n = k - 1$, assume that $(d^{k-1}(s, r) + d^{k-1}(r, t))^2 \leq (d^2(s, r) + d^2(r, t))^{k-1}$; (3) for $n = k$, since $(d^{k-1}(s, r) + d^{k-1}(r, t))^2 \cdot (d^2(s, r) + d^2(r, t)) \leq (d^2(s, r) + d^2(r, t))^{k-1} \cdot (d^2(s, r) + d^2(r, t))$, we have $(d^{k-1}(s, r) + d^{k-1}(r, t))^2 \cdot C^2 \leq (d^2(s, r) + d^2(r, t))^k$. So $(d^2(s, r) + d^2(r, t))^k \geq (C \cdot d^{k-1}(s, r) + C \cdot d^{k-1}(r, t))^2 > (d^k(s, r) + d^k(r, t))^2$. So the inequality also holds when $n = k$.

Combining (1)(2)(3), we can conclude that the inequality is true. ■

Bibliography and References

- [1] S. Basagni, I. Chalamtac, V. R. Syrotiuk, and B. A. Woodward. A distance routing effect algorithm for mobility (DREAM). In *Proceedings of the Fourth Annual International Conference on Mobile Computing and Networking (MOBICOM '98)*, pages 76–84, 1998.
- [2] B. Bellur and R. G. Ogier. A reliable, efficient topology broadcast protocol for dynamic networks. In *IEEE INFOCOM'99 New York, NY,*, March 1999.
- [3] A. Boukerche. Performance comparison and analysis of ad hoc routing algorithms. In *IEEE Conference on Performance, Computing and Communications*, pages 171–178, 2001.
- [4] 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 *Mobile Computing and Networking*, pages 85–97, 1998.
- [5] J.C. Cano and P. Manzoni. A performance comparison of energy consumption for mobile ad hoc network routing protocols. In *ACM International Symp. on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2000.
- [6] J.H. Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *INFOCOM (1)*, pages 22–31, 2000.
- [7] B.J. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. In *Mobile Computing and Networking*, pages 85–96, 2001.

BIBLIOGRAPHY AND REFERENCES

- [8] X. Chen, L. Qi, and D. Sun. Global and superlinear convergence of the smoothing Newton method and its application to general box constrained variational inequalities. *Mathematics of Computation*, 67(222):519–540, 1998.
- [9] C. Chiang and M. Gerla. Routing and multicast in multihop, mobile wireless networks. *Proceedings of IEEE ICUPC'97, San Diego, CA*, pages 546–551, Oct. 1997.
- [10] C.C. Chiang, H.K. Wu, W. Liu, and M. Gerla. Routing in clustered multihop, mobile wireless networks. In *IEEE Proceeding Singapore International Conference Networks*, pages 197–211, 1997.
- [11] S. Corson and J. Macker. Mobile ad hoc networking(manet): Routing protocol performance issues and evaluation considerations. *draft-ietf-manet-issues-02.txt*, October 1998.
- [12] S. R. Das, C. E. Perkins, and E. E. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *INFOCOM (1)*, pages 3–12, 2000.
- [13] R. Dube, C. Rais, K. Wang, and S. Tripathi. Signal stability based adaptive routing (ssa) for ad hoc mobile networks. *IEEE Personal Communication*, pages 36–45, Feb. 1997.
- [14] L. M. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. *IEEE INFOCOM'2001*, 2001.
- [15] J. J. Garcia-Luna-Aceves and M. Spohn. Source-tree routing in wireless networks. In *ICNP*, pages 273–282, 1999.
- [16] M. Gerla and J. Tsai. Multiclustet, mobile, multimedia radio network. *ACM/Baltzer Journal of Wireless Networks*, 1(3):255–265, 1995.
- [17] X.G. Guo and W.J Hsu. A localized topology control algorithm for power-efficient routing on mobile ad hoc networks. In *Proceedings of 5th IEEE International Conference on Mobile and Wireless Communications Networks*, pages 124–128, Singapore, October 2003. World Scientific.

BIBLIOGRAPHY AND REFERENCES

- [18] Z. J. Haas and M. R. Pearlman. The performance of query control schemes for the zone routing protocol. In *SIGCOMM*, pages 167–177, 1998.
- [19] Z.J. Haas and M.R. Pearlman. The zone routing protocol for ad-hoc networks. *Internet Draft RFC* (<http://www.ietf.org/charters.html/manet-charter.html>), November 1997., 1997.
- [20] X.Y Hong, K.X Xu, and M. Gerla. Scalable routing protocols for mobile ad hoc networks. *IEEE Network*, pages 11–21, 2002.
- [21] T.C. Hou and V.O. K. Li. Transmission range control in multihop packet radio networks. *IEEE Transactions On Communications*, 34(1):38–44, January 1986.
- [22] P. Jacquet. Optimized link state routing protocol. In *draft-ietf-manet-olsr-05.txt*, Internet Draft, IETF MANET Working Group, Nov. 2000.
- [23] H. Jiang and J. J. Garcia-Luna-Aceves. Performance comparison of three routing protocols for ad hoc networks. In *10th IEEE Conference on Computer Communications and Networks*, pages 547–554, 2001.
- [24] D. B Johnson and D. A Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.
- [25] L. R. Ford Jr. and D. R. Fulkerson. *Flows in Networks*. Princeton Univ. Press, 1962.
- [26] P. Karn. Maca - a new channel access method for packet radio. In *Proc. of Computer Networking Conf.*, 1990.
- [27] B. Karp and H.T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Mobile Computing and Networking*, pages 243–254, 2000.
- [28] Y. Kim, J. Lee, and A. Helmy. Impact of location inconsistencies on geographic routing in wireless networks. In *Proceedings of the 8th international workshop on Modeling analysis and simulation of wireless and mobile systems*, pages 124–127. ACM Press, 2003.

BIBLIOGRAPHY AND REFERENCES

- [29] L. Kleinrock and K. Stevens. Fisheye: A lenslike computer display transformation. Technical report, UCLA, Computer Science Department, 1971.
- [30] Y.B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. In *Mobile Computing and Networking*, pages 66–75, 1998.
- [31] R. Kravets and P. Krishnan. Power management techniques for mobile communication. In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom'98)*, 1998.
- [32] Range LAN. <http://www.proxim.com>.
- [33] J. Li, J. Jannotti, D. D. Couto, D. Karger, and R. Morris. A scalable location service for geographic ad-hoc routing. In *Proceedings of the 6th ACM International Conference on Mobile Computing and Networking (MobiCom '00)*, pages 120–130, August 2000.
- [34] L. Li and J. Halpern. Minimum-energy mobile wireless networks revisited. In *Proceedings of IEEE Conference of Communications (ICC '01)*, June, 2001.
- [35] L. Li, J. Halpern, V. Bahl, Y. M. Wang, and R. Wattenhofer. Analysis of a cone-based distributed topology control algorithm for wireless multihop networks. In *ACM Symposium on Principle of Distributed Computing (PODC)*, August, 2001.
- [36] X.Y. Li and P.J. Wan. Constructing minimum energy mobile wireless networks. In *ACM Symposium on Mobile Ad Hoc Networking and Computing MobiHoc*, 2001.
- [37] X.Y. Li, P.J. Wan, and Y. Wang. Power efficient and sparse spanner for wireless ad hoc networks. In *IEEE International Conference on Computer Communications and Networks (ICCCN01)*, 2001.
- [38] J. Lorch and A. Smith. Software strategies for portable computer energy management. *Personal Communications Magazine*, 5(3):60–73, 1998.

BIBLIOGRAPHY AND REFERENCES

- [39] L. Luan and W.J. Hsu. Power-efficient geographic routing for manets. Technical report, Center for Advanced Information System, Nanyang Technological University, Singapore, 2002.
- [40] J. Monks, V. Bharghavan, and W.W. Hwu. A power controlled multiple access protocol for wireless packet networks. In *INFOCOM*, pages 219–228, 2001.
- [41] S. Murthy and J. J. Garcia-Luna-Aceves. An efficient routing protocol for wireless networks. *Mobile Networks and Applications*, 1(2):183–197, 1996.
- [42] M. Naghshineh, J. Gomez, A. T. Campbell, and C. Bisdikian. Conserving transmission power in wireless ad hoc networks. *ICNP'2001*, 2001.
- [43] R. G. Ogier. Topology broadcast based on reverse-path forwarding (tbrpf). In *draft-ietf-manet-tbrpf-05.txt, Internet-Draft, MANET Working Group*, March 2002.
- [44] V. D. Park and M. S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *INFOCOM (3)*, pages 1405–1413, 1997.
- [45] G.Y. Pei and M. Gerla. Mobility management for hierarchical wireless networks. *Mobile Networks and Applications*, 6(4):331–337, 2001.
- [46] G.Y. Pei, M. Gerla, and T.W. Chen. Fisheye state routing in mobile ad hoc networks. In *ICDCS Workshop on Wireless Networks and Mobile Computing*, pages D71–D78, 2000.
- [47] G.Y. Pei, M. Gerla, X.Y. Hong, and C.C. Chiang. A wireless hierarchical routing protocol with group mobility (WHIRL). Technical Report 990006, 25, 1999.
- [48] W. Peng and X.C. Lu. On the reduction of broadcast redundancy in mobile ad hoc networks. In *Proceedings of the First Annual Workshop on Mobile and Ad Hoc Networking and Computing*, pages 129–130, Boston, USA, August 2000.

BIBLIOGRAPHY AND REFERENCES

- [49] C. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, pages 234–244, 1994.
- [50] C. E. Perkins. *Ad Hoc Networking*. Addison-Wesley, 2000.
- [51] C. E. Perkins and E. M. Royer. Ad-hoc on-demand distance vector routing. In *MILCOM '97 panel on Ad Hoc Networks*, Nov. 1997.
- [52] F. P. Preparata and M. I. Shamos. *Computational Geometry: an introduction*. Springer-Verlag, New York.
- [53] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying: An efficient technique for flooding in mobile wireless networks. In *Proceedings of Hawaii International Conference on System Sciences*, 2002.
- [54] McNab R. and F.W. Howell. Using java for discrete event simulation. In *Proceedings of Twelfth UK Computer and Telecommunications Performance Engineering Workshop (UKPEW)*, pages 219–228, University of Edinburgh, UK, 1996.
- [55] R. Rajaraman. Topology control and routing in ad hoc networks: A survey. *To appear in SIGACT News*, June 2002.
- [56] R. Ramanathan and R. Hain. Topology control of multihop wireless networks using transmit power adjustment. In *INFOCOM (2)*, pages 404–413, 2000.
- [57] T. S. Rappaport. *Wireless Communications: Principles and Practice*. Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [58] V. Rodoplu and T. Meng. Minimum energy mobile wireless networks. *IEEE Communications*, 17(8):1333–1344, August 1999.
- [59] E. Royer and C. Toh. A review of current routing protocols for ad-hoc mobile wireless networks. *IEEE Personal Communications*, pages 46–55, April 1999.

BIBLIOGRAPHY AND REFERENCES

- [60] S. Singh and C. Raghavendra. PAMAS: Power aware multi-access protocol with signalling for ad hoc networks. *ACM Computer Communications Review*, 1999.
- [61] S. Singh, M. Woo, and C. S. Raghavendra. Power-aware routing in mobile ad hoc networks. In *Mobile Computing and Networking*, pages 181–190, 1998.
- [62] M. Stemm and R.H. Katz. Measuring and reducing energy consumption of network interfaces in hand-held devices. *IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Science*, 1997.
- [63] I. Stojmenovic. *Location updates for efficient routing in ad hoc networks, in Handbook on Wireless Networks and Mobile Computing*. Wiley, 2002.
- [64] I. Stojmenovic, M. Seddigh, and J. Zunic. Internal node based broadcasting in wireless networks. In *Proceedings of Hawaii International Conference on System Sciences*, January 2001.
- [65] Adcon Telemetry. <http://www.adcon.com>.
- [66] C.K. Toh. A novel distributed routing protocol to support ad hoc mobile computing. In *IEEE 15th Annual International. Phoenix Conference Computer and Communicaition*, pages 480–486, 1996.
- [67] S. Udani and J. Smith. Power management in mobile computing (a survey). *University of Pennsylvania*, <http://www.cis.upenn.edu/~udani/paper.html>, 1996.
- [68] R. Wattenhofer, P. Bahl L. Li, and Y. M. Wang. Distributed topology control for power efficient operation in multihop wireless ad hoc networks. In *INFOCOM*, pages 1388–1397, 2001.
- [69] S.C. M. Woo and S. Singh. Scalable routing protocol for ad hoc networks. *ACM Wireless Networks*, 7:513–529, 2001.
- [70] Y. Xu, J. Heidemann, and D. Estrin. Adaptive energy-conserving routing for multihop ad hoc networks. Research Report 527, USC/Information Sciences Institute, October 2000.

- [71] Y. Xu, J. S. Heidemann, and D. Estrin. Geography-informed energy conservation for ad hoc routing. In *Mobile Computing and Networking*, pages 70–84, 2001.
- [72] Y. Xue, B. Li, and K. Nahrstedt. A scalable location management scheme in mobile ad-hoc networks. In *Proceedings of 26th IEEE Annual Conference of Local Computer Networks*, pages 102–111, Tampa,FL, November 2001.