CONNECTIONLESS ROUTING PROTOCOLS FOR MOBILE AD-HOC NETWORKS

by

Yong Sun

B.Eng., Northwestern Polytechnical University, 1997

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

MATHEMATICAL, COMPUTER AND PHYSICAL SCIENCES

(COMPUTER SCIENCE)

THE UNIVERSITY OF NORTHERN BRITISH COLUMBIA

September 2004

Your file Votre référence
ISBN: 0-494-04693-7
Our file Notre référence
ISBN: 0-494-04693-7

Abstract

Routing is a fundamental problem in computer networks. Numerous protocols have been proposed in the literature to solve routing problem in mobile ad-hoc networks. We classify them into two categories: *connection-oriented routing protocols* and *connectionless routing protocols.*

In this thesis, we first expand and refine the existing classifications of the protocols for mobile ad-hoc networks. Then, we present a taxonomy of connectionless routing protocols. As a result of this classification, we design and present a new compulsory routing protocol to fill the vacuum of a class in the taxonomy. Subsequently, we survey and analyze the existing connectionless routing protocols. Next, we present a unified framework for connectionless routing protocols. Our framework reveals that many of the existing connectionless protocols are no more than particular cases of a general setting, based on a small set of basic principles. Finally, we propose three efficient connectionless semi-compulsory routing protocols. We conducted a simulation study and the results show that our protocols perform better than the recent protocols in their class.

Contents

List of Figures

List of Tables

Publications from this Thesis

1. Y. Sun, X. Cui, and K. Alagarsamy, Efficient Semi-compulsory Routing Protocols for Mobile Ad-hoc Networks, *Proceedings of IASTED Conference on Wireless Networks and Emerging Technologies*, Banff, Canada, July 2004.

2. X. Cui, Y. Sun, and K. Alagarsamy, Simple and Efficient Semi-compulsory Routing Protocols for Mobile Ad-hoc Networks, *Proceedings of International Conference of Wireless Networks*, Las Vegas, USA, June 2004.

3. Y. Sun and K. Alagarsamy, A Taxonomy of Connectionless Routing Protocols for Mobile Ad-hoc Networks, *Submitted for publication*, 2004.

4. Y. Sun and K. Alagarsamy, A Generalized Framework for Connectionless Routing Protocols in Mobile Ad-hoc Networks, *Under preparation.*

Acknowledgments

I would like to express my deep felt gratitude to my supervisor, Dr. Alex Alagarsamy Aravind, for his endless patience, constant support, and encouragement. I would especially like to thank him for his extensive comments on the thesis write up, as they have helped to improve my writing skills and have also helped me to present my research in a much better way. I would like to thank the other two members of the committee, Dr. J. Polajnar and Dr. J. Chen, for their comments and suggestions, especially during the thesis proposal. I would like to thank Dr. Kui Wu from University of Victoria for taking time out from his busy schedule to serve as my external examiner. Special thanks to Dr. Robert Tait, the Dean of Graduate Studies, and Dr. Waqar Haque, the Chair of Computer Science, for their support and encouragement.

And I am thankful to Xiang Cui for his help in the implementation and simulation of the proposed protocols. I must also acknowledge Robert Lucas of the Computer Technical Support Group, for all of his computer and technical assistance throughout my graduate program. Appreciation also goes out to Dr. Saif Zahir, Dr. Liang Chen, and Dr, Charles Brown, for all their encouragement and assistance which helped me along the way.

I would also like to thank my family for the support they provided me through my entire life.

Finally, I recognize that this research would not have been possible without the financial assistance of the University of Northern British Columbia Graduate Studies and the Department of Computer Science at the University of Northern British Columbia (Teaching Assistantships, Graduate Research Assistantships).

Chapter 1

Introduction

Mobile computing is a technology which enables people to connect their mobile computing devices to network whenever and wherever they go. Ubiquitous access to information, anytime and anywhere, will characterize whole new kinds of information systems in the 21st century. To cope with the demands of mobility and portability in using the computers, mobile computing technologies are being enabled by rapidly emerging wireless communication systems, based on radio and infrared transmission mechanisms. The history of wireless networks started in the 1960's[4] and the interest has been growing ever since.

Today most of the traditional wireless networks, such as cellular telephony, personal communications systems, wireless PBXs, and wireless local area networks, are supported by static infrastructure (also called backbone). The infrastructure consists of fixed base stations or access points, which are connected either through wires or by long range wireless transmissions to act as gateways and bridges in the network. The transmission range of a base station constitutes a cell. The mobile user is connected to the base station with the best signal quality to carry out the communications through the fixed infrastructure. When the mobile user moves out of the cell, it initiates a hand-off and switch to a new base station.

However, setting up of a fixed infrastructure is not always viable in ad-hoc situations such as battlefield, emergency search, rescue operation, etc. Therefore, the other type of network, infrastructureless network known as Mobile Ad-hoc Network (MANET) attracts more research interest recently.

1.1 Mobile Ad-hoc Network

A mobile ad-hoc network is a collection of mobile hosts with wireless interfaces forming a temporary network without the aid of any established infrastructure or centralized administration[13]. In such ad-hoc networks, all nodes are capable of movement and can be connected dynamically in arbitrary manner. Also, the frequent and unpredictable change of topology in the network does not favor the use of any centralized control. Thus, the responsibility of organizing and controlling the nodes in the network are mostly distributed among the nodes themselves.

In general, mobile ad-hoc networks have the following characteristics[9]:

- *Dynamic Topologies*: Nodes are free to move arbitrarily; thus, the network topology may change randomly and rapidly at unpredictable times, and may consist of both bidirectional and unidirectional links.

- *Bandwidth-constrained, Variable Capacity Links*: Wireless links have significantly lower capacity than their hardwired counterparts. Also, due to multiple access, fading, noise, interference conditions, etc. the wireless links have low throughput.

- *Energy-constrained Operation*: Some or all of the nodes in a MANET may rely on batteries or other exhaustible means for their energy. In this scenario, the most important system design criterion for optimization may be energy conservation.

- *Limited Physical Security*: Mobile networks are generally more prone to physical security threats than are fixed wired networks. There is increased possibility of eavesdropping, spoofing and denial-of-service attacks in these networks.

With no prerequisites of fixed infrastructure, mobile ad-hoc networks offer unique benefits and versatility for certain environments and applications. Wireless ad-hoc networks can be deployed easily and quickly. Thus, such networks can be used in situations where the fixed infrastructure is not available due to cost, security, or safety reasons. Such a network is tolerant of the failure or departure of nodes, because the network does not rely on the fixed infrastructure. Such advantages attracted immediate interest in its early use among military, police, and rescue operations in disaster areas. Recently, one of the most popular scenarios is communication within groups of people with laptops and other mobile devices in a small area, such as a conference or classroom, single building, convention center, recreation sites, etc.

1.2 Routing in Mobile Ad-hoc Networks

Communication is a key for any distributed system. Achieving efficient communication between mobile nodes is a challenging problem. *If a node in the ad-hoc networks wants to communicate to another node in the network, then it has to be achieved in a finite time.* This problem is called **message routing** or simply **routing**. The problem become trivial if the *destination (receiver)* is within the transmission range of the *source (sender)*. Otherwise, the message has to traverse through some intermediate mobile nodes before it reaches the destination, which is called *multi-hop routing*.

1.2.1 Desirable Properties of MANET Routing Protocols

Performance of any distributed application running on a networked environment heavily depends on the efficiency of the underlying message routing strategy. Designing an efficient message routing protocol for ad-hoc networks is a complex task. The routing protocols used in ordinary wired networks, which are usually built on periodic updates of the routes and also cause slow convergence to the topology changes, are not well suited for this kind of dynamic environment where the nodes are mobile and links are continuously being created and destroyed.

The following is a list of common desirable properties of MANET routing protocols[9]:

- *Adapting to frequent topology change.*
- *Distributed operation.*
- *Loop freedom.*
- *Minimal resource consumption.*
- *Security.*
- *QoS awareness.*

1.2.2 Routing Strategies

Many approaches in ad-hoc networks have been proposed with the goal of achieving efficient communications. One popular approach is, similar to wired networks, that the source of the message discovers and constructs an exact route between the communicating nodes. The messages are then transmitted through the established route. That implies that the source and the destination are connected when the communication occurs. Another approach is to forward the message to a neighbor in the right direction to the destination, and the neighbor then makes a similar decision regarding how to route the message in a greedy way. The simplest way of doing this is by flooding the message in all directions. The messages are generally saturated in the network and hence the messages reach the destination eventually. Obviously, flooding generates large numbers of duplicate copies and wastes the constrained network resource. Thereafter, in many variations of flooding, called *selective flooding*, the source node with additional information of the destination sends the message in the approximately right direction to the destination. In this approach the source node only considers the connectivity to its neighbor nodes.

The approaches outlined above mainly based on the connectivity to the destination or to the next hop. On receiving a message, the intermediate nodes look up for the next-hop and forward the message to it. After the forwarding, the delivered message is removed from the storage of the intermediate nodes. If there is no next-hop available, the communication is aborted and the undelivered messages are dropped. Recently, many routing protocols are proposed in the literature based on yet another approach in which the nodes actively carry the messages until the connectivity to the next-hop is established. This class of protocols are more effective for the networks where disconnection is often possible.

Based on the connectivity requirement of the network assumed by the routing protocols for MANETs, we classify them broadly into two classes: *connection-oriented routing protocols* and *connectionless routing protocols*. There are many surveys and taxonomies available for connection-oriented routing protocols[2, 19, 26, 35, 43, 44, 55, 56].

Our focus in this thesis is on connectionless routing protocols for mobile ad-hoc networks. However, for the sake of completeness we briefly review the existing classifications of connection-oriented routing protocols in terms of a new classification.

1.2.3 Classification of Connection-oriented Routing Protocols

We classify the connection-oriented routing protocols into two categories based on the type of connectivity requirement: *global connection-oriented routing protocols* - which require the establishment and maintenance of the entire route during the message transfer, and *local connection-oriented routing protocols* - which require the connectivity only to the next-hop in the route during the message transfer.

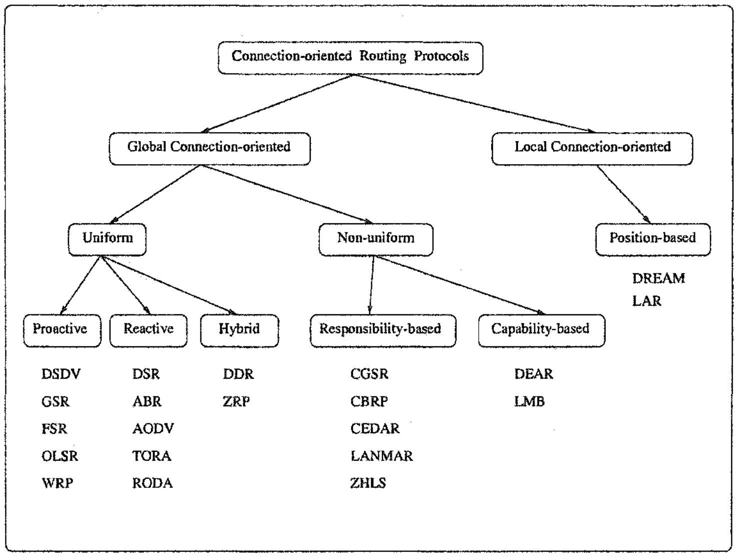

Figure 1.1: A Classification of Connection-oriented Routing Protocols

Global Connection-oriented Routing Protocols

The primary characteristic of the global connection-oriented routing protocols is that a route between the source and destination has to be constructed for their communication. This implies that a path between the source and destination is a prerequisite for their communication. That is, the protocol guarantees communication only if the source and destination is connected for a period of time long enough to discover the route and transmit the message over it. If that connectivity fails to be established, or fails to be maintained during the message delivery, then the communication between these two nodes is not possible.

Based on the node functionality, the global connection-oriented routing protocols can be again classified into **uniform** and **non-uniform** routing protocols. All the nodes in the uniform routing protocols have identical functionalities (capabilities and responsibilities), and all the nodes in the non-uniform routing protocols are not homogeneous.

In terms of when to initiate the route discovery, global connection-oriented routing protocols are categorized into three classes: (i) **proactive**, (ii) **reactive**, and (iii) **hybrid** protocols. In proactive protocols, also known as **table-driven** approaches, every node continuously maintains the complete routing information of the network. When a node needs to forward a packet, the route is readily available. Thus there is little delay until the route is determined. However, proactive protocols are not appropriate for a highly dynamic topology, as they continuously use a large portion of the scarce wireless resource to keep the routing information up-to-date. Examples of proactive routing protocols include Wireless Routing Protocol (WRP)[34], Global State Routing (GSR)[7], Destination-Sequenced Distance-Vector (DSDV) Routing[39], Optimized Link State Routing (OLSR)[29], and Fisheye State Routing (FSR)[24].

In reactive protocols, also known as **on-demand** approaches, the route discovery procedure is initiated only when it is required, and the route information is maintained as long as the route is used actively. Therefore, the route maintenance overhead is reduced. But the disadvantage is that this class requires more time to determine a route when needed. Some examples of reactive routing protocols are Dynamic Source Routing (DSR)[28], Associativity-Based Routing (ABR)[51], Ad-hoc On-demand Distance Vector Routing (AODV)[42], RODA[30], and Temporally Ordered Routing Algorithm (TORA)[40].

In hybrid routing protocols, each node proactively maintains the local routing information in one region and reactively initiates route discovery for the destination outside this region. Examples of hybrid routing protocols include Zone Routing Protocol (ZRP)[23] and Distributed Dynamic Routing Algorithm (DDR)[36].

Based on the non-uniformity of node functionality, we divide the non-uniform routing protocols into two classes: **responsibility-based** and **capability-based** routing protocols. The responsibility-based routing protocols utilize nodes with specialized responsibility, such as the *cluster heads, group leaders*, or *route gateways*, to coordinate the dissemination of local route information. Furthermore, the relative positions of the specialized nodes can provide directional guidance to routing between the regular nodes. Examples of responsibility-based routing protocols include Clusterhead Gateway Switch Routing (CGSR)[16], Core-Extraction Distributed Ad hoc Routing (CEDAR)[50], Zone-based Hierarchical Link State (ZHLS)[25], Cluster Based Routing Protocol (CBRP)[27], and Landmark Ad hoc Routing (LANMAR)[41].

The capability-based routing protocols utilize nodes with specialized capabilities, such as *transmission range, power supply, storage space, processing capacity, movement speed*, etc. With more powerful capabilities, these nodes take more part in the message delivery. Examples of capability-based routing protocols include Landmark Routing with Mobile Backbones (LMB)[53] and Device and Energy Aware Routing (DEAR)[1].

Local Connection-oriented Routing Protocols

The local connection-oriented routing protocols require the connectivity only to the next-hop in the route during the message transfer. Flooding is the simplest example of the local connection-oriented routing protocols. The source of a message need not know the complete route to the destination and just broadcasts the message to all its neighboring nodes. Through the hop-by-hop forward, the message can be flooded into the network and can eventually reach the destination in a greedy way. With additional information of the destination, the source can propagate the message in the right direction to the destination, achieving more efficient utilization of network resource.

To obtain such information of the other mobile nodes, based on the availability of positioning equipment, a group of **position-based** routing protocols is introduced. Each node obtains location and timing information from external devices such as GPS, and transmits its location coordinates to other nodes in the network. In practice, the location coordinates can be extracted as logical references in the grid-based networks. With the location information, the source can transmit messages in the direction of the destination using a greedy mode. Examples of position-based routing protocols include Distance Routing Effect Algorithm for Mobility (DREAM)[3] and Location-Aided Routing (LAR)[31].

There is a growing interest of using directional antennas to enhance the routing performance. The protocol presented in [37] reduces the number of routing packets transmitted during the route discovery by limiting the query flood to a restricted region using directional transmission. In [15], Choudhury and Vaidya propose a directional MAC protocol *DiMAC* and evaluate the routing over directional antennas. The analysis shows that ad-hoc networks may achieve better performance using directional antennas. With this introduction to connection-oriented routing protocols, we next present the motivation for our thesis.

1.3 Motivation

In mobile ad-hoc networks, the arbitrary movements of mobile nodes may often result in temporary partitions in the network and therefore the presence of a stable connected path from the source to the destination or immediate connection to the next hop may not always be available. Recently, mobile ad-hoc networks called challenged networks or delay-tolerant networks[20] are getting more attention due to their suitability in various applications. In this class, the end-to-end disconnection may be more common than connection and therefore the message delivery cannot be guaranteed using the connection-oriented routing protocols mentioned above. This situation paved the way for the emergence of *connectionless routing protocols*. The basic idea behind the connectionless routing protocols is that instead of dropping messages or aborting the message communication when the connectivity to next-hop is not available, the protocol may actively enforce the intermediate nodes to *carry* the messages until such connectivity is established.

Recently, there are many connectionless routing protocols proposed in the literature, with minor variations in their functionalities. To the best of our knowledge, there is no survey or taxonomy available for connectionless routing protocols.

1.4 Contribution

This thesis contains the following contributions.

- The classification of routing protocols of MANET is first expanded in the higher level into connection-oriented and connectionless routing protocols. The expanded classification gives a comprehensive view of all the existing routing protocols for MANET.

- To complement the existing surveys and classifications of connection-oriented routing protocols[2, 19, 26, 35, 43, 44, 55, 56], we present a taxonomy of connectionless routing protocols[47]. This taxonomy exhibits the similarity and differences among the protocols in the connectionless class.

- As a result of our classification of connectionless routing protocols, we design and present a new connectionless routing protocol to fill the vacuum noted in the taxonomy.

- Next we identify the fundamental factors responsible for the performance behavior of connectionless routing protocols in mobile ad-hoc networks.

- Then, after elaborating these factors, we present a unified framework for connectionless semi-compulsory routing protocols[48]. This framework brings together the ideas and concepts scattered in various protocols of this class. Also, it can (i) provide a basis for developing theory and conduct experimental study of the protocols; (ii) aid to develop or design new protocols; and (iii) help the users to identify the suitable protocols for their applications to implement.

- We briefly sketch how the existing connectionless routing protocols fit in the framework.

- Then we present three new semi-compulsory routing protocols[49, 14], which are direct results of our experience in constructing the framework.

- Our protocols are simulated and compared with some well known protocols (Snake and Runners[13]) in their class.

1.5 Organization

The rest of the thesis is organized as follows. Chapter 2 presents a taxonomy of connectionless routing protocols for mobile ad-hoc networks. The routing protocols in various classes of the taxonomy are surveyed in Section 2.2. Another classification of connectionless routing protocols is presented in Section 2.3. Section 2.4 presents the summary of key ideas and observation of connectionless routing protocols. In Chapter 3, Section 3.1 identifies the fundamental factors of effective message communication in mobile ad-hoc networks. Section 3.2 presents a generic framework from which many connectionless routing protocols can be derived. Section 3.3 investigates the various potential concrete policies for the replaceable components of the framework. The analysis of existing protocols in our framework is sketched in Section 3.4. In Chapter 4, first we propose a new non-uniform-support compulsory routing protocol to fill the vacuum noted in our proposed taxonomy. Then we present three new semi-compulsory routing protocols in Section 4.2. An implementation strategy for our protocols is discussed in Chapter 5. Chapter 6 analyzes and compares the three new semi-compulsory protocols with Snake and Runners protocols, through a simulation study. Chapter 7 concludes this thesis and outlines directions for future research.

Chapter 2

Connectionless Routing Protocols

As mentioned in Section 1.2, the fundamental task of routing messages in most connection-oriented protocol is to establish a path between the source and destination. The intermediate nodes act as routers in order to forward the messages along the path. However, in highly dynamic ad-hoc networks, even if a valid path is established, a single link breakage will make the path invalid. In addition, the arbitrary movements of mobile nodes may often result in temporary partitions in the network and therefore the presence of a stable connected path from the source to the destination or immediate connection to the next hop may not always be available. In such environments, the end-to-end disconnection may be more common than connection and therefore the message delivery cannot be guaranteed using the connection-oriented routing protocols. This situation paved the way for the emergence of *connectionless routing protocols*.

The basic idea behind the connectionless routing protocols is that instead of dropping messages or aborting the message communication when the connectivity to next-hop is not available, the protocol may actively enforce the intermediate nodes to *carry* the messages until such connectivity is established. The message transmission concept *relay* is the key for connectionless routing protocols. Relay is a refinement of traditional *store-and-forward* mechanism and it works the same way as store-and-forward mechanism works when the next-hop to forward the message is connected. In addition, relay facilitates the nodes to *carry* the message when the next-hop is not connected.

Recently, there are many connectionless routing protocols proposed in the literature [47], with minor variations in their functionalities. Disconnected Transitive Communication (DTC)[10], Epidemic Routing[52], Improved Epidemic Routing[17], Probabilistic Routing[32], Optimistic Forwarding[8], Voilà Protocol[45, 46], Optimal Relay Path (LR1 and LR2)[33], Partitioning Avoidance[21], Snake and Runners protocols[13], Hierarchical Support Routing Protocol (HSRP)[11], and Message Ferrying[54] are examples of connectionless routing protocols for MANET. These protocols can be classified in various ways. In this thesis, we present two classifications for connectionless routing protocols. First we introduce some definitions.

2.1 Terminology

A mobile node can have the routing functionality but does not imply that it will always be willing to participate in the routing process. A node may not want to participate in routing for various reasons.

Definition 1 *A mobile node is said to be* **support node** *if its primary responsibility is routing. A mobile node which is not a support node is called* **regular node**.

The mobile nodes may be categorized based on the purpose of their mobility pattern.

Definition 2 *A mobile node is said to be* **compulsory** *if it adapts its mobility to satisfy the routing requirement.*

There may be nodes whose primary functionality is not routing but they might participate in routing on their way when necessary. With the above two definitions we can now introduce our first classification.

2.2 Taxonomy

The connectionless routing protocols can be classified as **uniform** protocols and **non-uniform** protocols based on the node functionalities. In uniform protocols all nodes are homogeneous and in non-uniform protocols the nodes may have varying functionalities (heterogeneous). First we will discuss uniform routing protocols.

The uniform routing protocols for mobile ad-hoc networks can be further classified into **compulsory** protocols and **non-compulsory** protocols, based on the *type* of mobile nodes. In compulsory protocols all nodes in the network are compulsory and in non-compulsory protocols no node is compulsory. The compulsory nodes have to modify their trajectories to fulfill the message transmission and the non-compulsory nodes rely only on their natural movements to transmit the messages.

A routing protocol is called **uniform** if all the nodes in the network have equal responsibility in the routing process. However, in many ad-hoc networks, all nodes need not have equal responsibility in the message delivery. Some nodes may not be willing to take part in routing due to many factors such as poor power supply, limited capability, privacy, etc. In such situations, only a subset of mobile nodes, i.e. *support*, take part in routing messages and therefore the routing protocols designed for such networks with heterogeneous nodes are called **non-uniform** routing protocols.

Again, based on the functionality of the support nodes, the non-uniform protocols can be further classified into **uniform support** protocols and **non-uniform support** protocols. In uniform support protocols all the support nodes are homogeneous and in non-uniform support protocols the support nodes need not be homogeneous. For example, the support may be organized in a hierarchical structure and the functionality in each level may be different.

The uniform support protocols can be further divided into **compulsory** protocols and **semi-compulsory** protocols based on the *intent* of the mobility pattern of the nodes in the network. The compulsory protocols require the mobility of all nodes, including support nodes and regular nodes, to be defined in order to carry out the message routing task. In case of semi-compulsory routing protocols only the mobility of support nodes are designed to carry out the message delivery and no constraint is placed on the mobility of the regular nodes.

Similar to uniform support case, non-uniform support protocols also can be divided into **compulsory** and **semi-compulsory** protocols based on the intent of the nodes mobility. We have identified a representative protocol in the literature for all the classes described above, except for the non-uniform non-uniform-support compulsory routing protocol. We will design a protocol (called TBSP) for this missing class by combining ideas from other existing protocols in section 4.1. The classification is depicted in Figure 2.1.

Next we survey the existing connectionless routing protocols in each class.

2.2.1 Uniform Compulsory Routing Protocols

In this class of routing protocols all nodes are *homogeneous* and *compulsory.* That means all nodes have identical functionality with respect to routing and each node

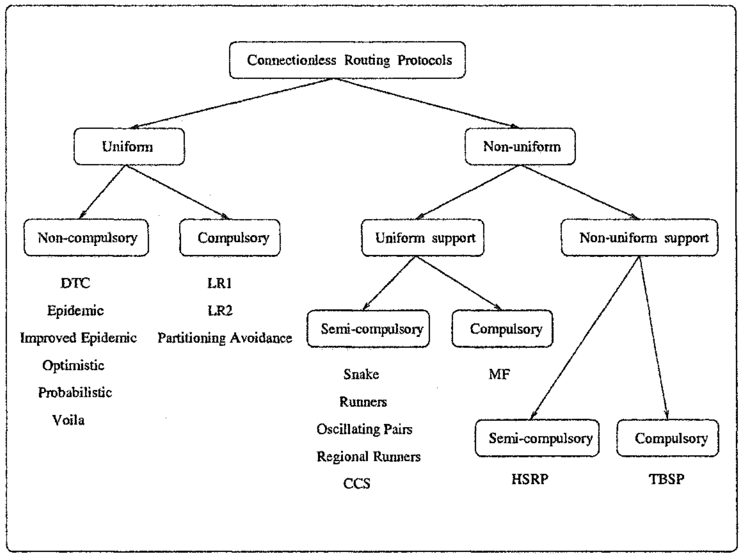

Figure 2.1: A Classification of Connectionless Routing Protocols

is required to change its normal mobility pattern when necessary to accomplish a routing path between the source and destination. This modification in the mobility pattern is enforced mainly to achieve the message communications across the mobile nodes as fast as possible. For example, if a mobile node following its trajectory near a highway carries a message to be delivered to a node in another city, then it could move closer to the highway so that the message can be transferred to a mobile node moving on the highway towards the destination city. This might speed up the delivery of that message.

Optimal Relay Path (LR1 and LR2)

First, we describe two protocols in this class proposed in [33]. For our reference we label them as LR1 and LR2. The basic idea behind these two protocols is that instead of waiting for network reconnection a node can actively change its location to achieve connectivity using the knowledge about the locations of other nodes. Thus, each node requires the knowledge about the motion and locations of all the other nodes. Protocol LR1 assumes such a knowledge is available. The second protocol LR2 does not assume that the movement of the other nodes is known. Instead, it uses the concept of *scope* (movement region) and each node maintains a minimal spanning tree to update the location information. When a node leaves or joins a scope it informs its location to its neighbors. The location update is then flooded to the entire network. In this way, the network can keep track of the mobile nodes if they communicate current location periodically.

Using the location and mobility information, both protocols estimate the optimal trajectory (shortest path) for the message to travel and then relay the message to the next node closer to or on the trajectory. The motion information of a node is updated according to the latest information at the most recent time in point. Each node carrying the message changes its normal movement in order to complete a routing trajectory. After relaying the message the node returns back to its original trajectory.

Partitioning Avoidance

In [21], an approach that utilizes *network survivability* concept to delay or avoid the network partitioning in ad-hoc wireless networks is proposed. With the unpredictable mobility of mobile nodes, the network can be separated into partitions. They define *separation link* as the connection between two nodes whose failure will create partitioning in the network. *Critical links* are the separation links that are about to fail (the distance between nodes forming the link is close to the communication or radio range). This approach uses depth first search (DFS) to detect the separation links in the network. It is assumed that each node knows its location and periodically exchanges the up-to-date location information with its neighbors. Once a node knows all of the surrounding critical links, two ways to delay or avoid their failure are used: (i) change the trajectory of one or both nodes forming the critical link; (ii) and bring in another node to reinforce the link. In the first way, two cases are considered. If the number of critical links around a node is one, the root node is moved t units towards the neighboring node forming the critical link. The value t can be either a constant or variable to make the critical link lose its critical status. If the number of critical links around a node is more than one, two possible actions can be taken by the root node: (a) do nothing, hoping that the neighboring nodes have only one critical link and they will take a favorable action to avoid the failure; (b) move in a direction to maximize the number of nodes it remains attached to.

In the second way, an outside node called *helper* is brought into reinforce the critical link. The root node with critical links broadcasts a help-seeking message carrying the following information: the destination where the helper should move to; and the importance of critical link. The helper is selected from the neighbors of the two nodes forming the critical link by minimizing the square root of summation of squares of the distances between the neighboring node and the two nodes forming the critical link. The determined helper then constantly change its location to keep itself in between the two nodes forming the critical link to ensure the connectivity of the two partitions.

Discussion

LR1 and LR2 introduce the concept relay to handle the temporary disconnection. Relay is a refinement of traditional store-and-forward mechanism. In addition to store-and-forward, it also carries the message when next-hop is not connected. The partitioning avoidance mechanism attempts to reinforce the established connection to avoid the network partitioning. These protocols work well for the applications where the network is almost connected. Because the time spent by a node deviating from the original trajectory is not too large. Otherwise, the errors in the location information increase due to cascading effect and that may lead to a kind of live-lock situation chasing each other. Also, a node has to deviate from its own trajectory too much to relay the message if the next-hop is not close and that might be unreasonable for many applications.

2.2.2 Uniform Non-compulsory Routing Protocols

In uniform non-compulsory protocols *all nodes are homogeneous* and *no node is compulsory.* That is, nodes need not change their trajectories for routing purpose. The basic idea behind this class of routing protocols is to take advantage of the *natural movements* of mobile nodes. A node carrying a message relays it when it eventually meets the next-hop. To increase the efficiency in message transmission and resource usage each node can extract and use the mobility information of the other nodes to select the next-hop(s). Many existing connectionless routing protocols fall into this uniform non-compulsory protocol class.

Epidemic Routing

The first set of protocols in this class we choose to discuss is called Epidemic Routing, proposed in [52], and its variations in [17].

Epidemic Routing uses flooding to distribute the messages to the nodes within a connected portion (called an *Island*) of the networks. In this way, messages are quickly distributed through connected portions of the network. When a node from an Island moves closer to another Island, its messages are transported and spread in that Island. At this point, the messages spread to an additional Island of nodes. A message is delivered when the receiver comes in contact with a node which holds the message.

Epidemic Routing supports the eventual delivery of messages to arbitrary destinations with minimal assumptions regarding the underlying topology and connectivity of the network. In fact, only *periodic pairwise connectivity* is required to ensure eventual message delivery. Multiple dead messages may float in the network due to flooding. To avoid this, each message is tagged with *hop-count* and the message is dropped when it crosses the number of hops equal to its hop-count. Computing the hop-count precisely is the key for reliable message delivery.

Improved Epidemic Routing

The routing strategies proposed in [17] are the variations of Epidemic routing. First variation, from the epidemic routing, is on *mobility*. The node movements are modeled as *discrete* steps using discrete probability distributions. The probability distribution is not restricted to uniform, it can be *any distribution*. The nodes use *pauses* between steps and nodes within a transmission range are able to communicate. The second variation is on the *message drop strategy*. The message drop strategies of this protocol are based on buffer size. Finite buffer size is imposed on nodes and the selective dropping of messages is applied when the buffer limitation is reached. The paper explores various message drop strategies. These drop strategies form an implicit routing protocol by deciding which packets to drop from the buffer. This protocol assures more message deliveries, as compared to Epidemic routing.

Probabilistic Routing

Recently in [32], yet another non-compulsory protocol called Probabilistic Routing is proposed. The intuition behind this protocol is that most nodes usually do not move around randomly and the movement patterns are thus *likely to be predictable.* Based on this observation, a probabilistic metric called *delivery predictability* is established at each node for each known destination. The delivery predictability indicates the chance of that node delivering the message to the destination. When a node encounters another node, they exchange information about the delivery predictabilities they have and update their own information accordingly. Based on this information, a decision is then made on whether or not to forward a certain message to this node. Delivering messages to more nodes is also indicated in the paper.

Disconnected Transitive Communication (DTC)

The routing protocol called Disconnected Transitive Communication (DTC), proposed by Chen and Murphy in [10], also belongs to this class. The basic idea of DTC is to relay the message to another node as *close* to the destination as possible, where closeness is defined to be the likelihood of being in contact with the destination earlier than the source. There are two subtasks to be carried out by this protocol: (i) relay the message through the network, constantly getting closer to its destination and (ii) estimate how often a node searches for next-hop to relay the message. The first task is achieved through a concept called *utility,* which describes the usefulness of a host as the next-hop for a message and the second task is achieved by properly defining *rediscovery interval.*

The utility of a host reflects the possibility that it will meet the destination of the message before the message becomes dead. Every message carries a *time to live* value, and when this expires, the message becomes dead and therefore must be dropped from the system. Five characteristics of mobile nodes are identified and used as components for the utility calculation. These include the list of nodes *most recently noticed*, the list of nodes *most frequently noticed*, the *future plan* of a host, the *power level*, and the *rediscovery interval*. The paper discusses two ways of computing the utility: (i) host collects the needed information from the cluster (connected component), computes the utility of each node in the cluster and decides the next-hop; (ii) host computes its own utility value and broadcasts it to the cluster. The nodes in the cluster compute their utility values, compare with the host's utility value and respond only to those nodes whose utility values are higher than that of the host's. This way the host can identify the next-hop with higher utility value for that message and relay the message to that node.

The rediscovery interval (RDI) defines how often a node invokes its search for next-hop(s). The protocol chooses to discover the next hop *periodically*, where the period is *tunable* by the application and is able to approximately detect changes in cluster membership without adding a great deal of network overhead. The interval is shortened when cluster membership changes are more frequent, and lengthened when changes are infrequent (RDI doubles each time a utility probe is completed).

Optimistic Forwarding

Another uniform non-compulsory routing protocol is by Chen, Kung, and Vlah[8]. They present two protocols: *pessimistic* and *optimistic*. These two routing methods are distinguished by how long the messages are stored at intermediate nodes before they are forwarded to the next-hop. In pessimistic forwarding, a message is dropped whenever there is no connection to next-hop. In optimistic forwarding, messages are never dropped due to disconnection with next-hop. Instead, the intermediate nodes carry the messages while waiting for the opportunities to be forwarded further.

Voilà Protocol

Shah and Hutchinson proposed the Voilà Protocol[45, 46] to deliver messages between disconnected hosts. Instead of designing a new routing protocol, they locate Voilà Protocol between Transport Layer and Network Layer by using any existing routing protocols such as AODV, DSR, etc. When the source node is not able to discover and construct a route to the destination in another partitioned component using the routing protocol, it uses Voilà protocol to select the carrier node in every direction that comes in contact with the destination within a certain period of time and disseminate the message to the boundary of the partition. The carrier nodes carry the undelivered messages waiting for the connection to the destination.

Discussion

The utility component or predictability to identify next-hop and message dropping strategies are the keys for the efficiency of the protocols in this class. It gives flexibility for the designers to choose the functions suitable for a particular system or application. On the other hand, the selection of such specific system or application makes the protocol less generic.

Chen, Kung, and Vlah have conducted simulation study by dropping off messages after predetermined times called *pessimistic delay* and observed that the optimistic forwarding takes advantage of predictable node movement on the highway to relay messages. The experiments show that node mobility improves end-to-end transmission if messages are delayed rather than dropped immediately in case of disconnection with the next-hop and the improvement is higher for traffic scenarios with more relative movement.

In probabilistic routing, estimating the delivery predictability for each neighbor is the key to the performance of the protocol. Another problem is to determine how many nodes to be chosen to relay the message. Choosing a large number might increase the probability of quick and reliable delivery, but more resources are to be wasted.

Voilà Protocol complements the connection-oriented routing protocols in case of disconnected networks. If a message cannot be delivered to the destination, the intermediate nodes carry the undelivered message and initiate the route discovery later.

2.2.3 Non-uniform Uniform-support Compulsory Routing Protocols

In this class, support nodes are *homogeneous* and both support nodes and regular nodes are *compulsory*.

Message Ferrying (MF)

The Message Ferrying (MF) protocol proposed in [54] falls into this class. In the MF scheme, the nodes are classified either as *message ferries* or *regular nodes* based on their roles in the message communication. The main idea of the MF scheme is that it introduces *non-randomness* in the mobility of the ferry nodes and exploits such non-randomness to help message delivery. Ferries take responsibility of relaying messages to regular nodes. Ferries move around the deployed area according to *known routes*, collect messages from the sending nodes and deliver messages to their destinations or other ferries. With knowledge about ferry routes, the regular nodes can *adapt* their trajectories to meet the ferries and transmit or receive messages. By using ferries as relays, the MF scheme provides regular connectivity in a disconnected network and improves message delivery performance without global knowledge of each node's location. Ferry scheme is defined using the following metrics: *Ferry selection or designation, Number of ferries, Ferry mobility, Ferry coordination, Regular nodes mobility,* and *Regular nodes coordination.*

Discussion

The above protocol is simple because it is based on the conventional ferry or bus routing strategy. But the receivers do not know whether they have messages for them in the ferries so they cannot decide their movements towards ferry route to receive the messages.

2.2.4 Non-uniform Uniform-support Semi-compulsory Routing Protocols

In this class, the support nodes are *compulsory* whereas the regular nodes are *non-compulsory.* In semi-compulsory protocols, only the mobility of support nodes have to be defined in order to carry out routing. The *snake* and *runners* protocols presented and analyzed in [13] are the examples of the semi-compulsory routing protocols with uniform support.

Snake Protocol

The snake protocol forces only the support nodes of the network to move in a coordinated way for message routing purpose. The main idea is that a snake-like sequence of support nodes always remain *pairwise adjacent* and move in a way determined by the snake head. The head moves by executing a *random walk* over the entire area and the other nodes follow the same route. The number of support nodes is predefined. At the initial phase, a leader election is conducted to choose the snake head. Assume that the support nodes are MS_1 (head), MS_2, MS_3, ..., MS_n. The head MS_1 randomly chooses its new direction to move. Before leaving the current location, MS_1 sends a message to MS_2 that states the new direction of movement. MS_2 then will change its direction as per instruction of MS_1 and will propagate the message to MS_3. In general, MS_i will follow the order of MS_{i-1} after transmitting the new direction to MS_{i+1}. The speeds are assumed to be the same. The protocol is implemented using three subprotocols: *sensor subprotocol* - to notify the sender that it may send its message(s); *motion subprotocol* - to implement snake's random motion; and *synchronization subprotocol* - to transmit incoming messages to the members of the support. The messages are stored in every node of the support and when a receiver comes within the transmission range of a support node, the receiver is notified and the message is then forwarded. After delivering a message, a control message is flooded across the support to remove the duplicate messages.

Runners Protocol

In runners protocol, instead of maintaining pairwise adjacency between support members, all support nodes sweep the entire area by independent random walk. When two runners meet they exchange messages.

Next, we briefly describe three new semi-compulsory routing protocols, Oscillating Pairs Protocol[49], Regional Runners Protocol[49], and Center Concentrated Support Protocols[14]. The complete protocols and their analysis of these protocols are given in Section 4.2.

Oscillating Pairs Protocol

In oscillating pairs protocol, the network area is divided into parallel stripes as scopes. A pair of support nodes is assigned to each scope and they always maintain connectivity between them. The support pair in each scope oscillates between the top and the bottom of their scope. When the support pairs of adjacent scopes meet, they can exchange their messages.

Regional Runners Protocol

The regional runners protocol can be viewed as a variation of runners protocol. The difference is that the runners do not move randomly in the entire network. The network area is divided into overlapping subregions as scopes. The support nodes are assigned to each scope and restricted within their respective scopes.

Center Concentrated Support Protocols

The center concentrated support family includes four protocols. We studied one protocol called CCS1. In CCS1, the network area is divided into center region (CR) and outer region (OR) of equal size. More support nodes are assigned in CR and they are restricted to make random walk within CR. The remaining support nodes are deployed in OR initially and they are allowed to make random walk in the entire network region.

Discussion

In snake protocol, only the snake head moves randomly and the movements of the other support nodes are determined by the head node. In case of runners protocol, every node of the support moves (runs) independently over the entire area and the messages are exchanged between support nodes when they meet. The maintenance of pairwise adjacency in the snake boosts the efficiency of message transfer across the support, but it takes longer time for the support (snake) to sweep the entire network area. Independent random movements of runners help the support to sweep the area faster, but increase the complexity of message management task of the protocol due to the increased randomness in the mobility of the support nodes. The analysis and simulation study of the three new protocols with snake and runners protocols will be presented in Chapter 6.

2.2.5 Non-uniform Non-uniform-support Semi-compulsory Routing Protocols

In this class, the network contains both support nodes and regular nodes. The support nodes are *compulsory* whereas the regular nodes are *non-compulsory*. The support nodes may have differing functionalities (*non-uniform*). For example, one might divide the support nodes into two category: *city support* - responsible for message routing within a city and *inter-city support* - responsible for message routing between cities and along the highway. In this example, the mobility pattern of city support could be different from the mobility pattern of inter-city support. The basic idea behind this class of protocols is that instead of using a single large support for the whole network, it employs a different subset of support nodes in an organized structure (such as hierarchical, clustered, etc.) suitable for the nature of the network and its application.

Hierarchical Support Routing Protocol (HSRP)

The hierarchical support routing protocol (HSRP) proposed in [13] is a semi-compulsory protocol with non-uniform support. The network is abstracted into *city graphs* which are connected by highways across specific *city access ports*. Support nodes are divided into two categories based on their mobility patterns: *city mobile nodes* with *random routes* and *highway mobile nodes* with *non-random routes*. City nodes are deployed in each city and perform the mobility described in [13]. The highway nodes move only on the interconnection highways passing frequently between the access ports. The messages within the city are relayed using local mobility pattern and the messages for another city are relayed through highway nodes. A city node exchanges messages with a highway node when they meet at a city access port.

Discussion

Taking advantage of the regular traffic of the mobile users across the interconnection highway is the attractive feature of this protocol. It implicitly assumes that the mobile nodes in such regular traffic take the responsibility in routing and therefore they are also considered as support nodes. However, it is not obvious to see an application where this type of arrangement is common.

2.2.6 Non-uniform Non-uniform-support Compulsory Routing Protocols

In this class, the network contains both support nodes and regular nodes. All the nodes are *compulsory.* Again, like non-uniform non-uniform-support semi-compulsory case, the support nodes may have differing functionalities (*non-uniform*). An example of this would be the classification of support nodes into regional or local support nodes and highway support nodes.

We are not aware of any existing routing protocol for this class. Therefore, we will design and present a new protocol called three-base support routing protocol (TBSP) in section 4.1. Next we present another classification based on message copies.

2.3 Another Classification

Another way to classify the connectionless routing protocols is based on *how many copies of a message* are kept in the system during its relay. If a protocol retains exactly one copy of the message during its relay, then we say that the protocol is **single-copy based**. Otherwise, the protocol is called **multiple-copy based.**

In single-copy based protocols, since only one message is relayed at any time, the availability of that message in the network is restricted to one node. Therefore, this class of protocols need more sophisticated mechanisms to *identify the next-hop* to relay the message to increase its delivery probability. The popular next-hop identification mechanisms used in the literature are:

- **Utility**[10]: Uses the history and future plan of node movement, and the system parameters. Five characteristics include: Most recently noticed, Most frequently noticed, Future plans, Power, and Rediscovery Interval.

- **Meeting Likelihood**[17]: Describes the likelihood of a pairwise meeting.

- **Delivery Predictability**[32]: Indicates the probability of encountering a certain node.

- **Optimal Relay Path**[33]: With the knowledge about the trajectories of the other nodes to compute the *optimal trajectory* for relaying a message with the least time.

In multiple-copy based protocols, the node floods the message either fully or selectively to increase the availability of message in the system so as to increase its delivery probability. On the other hand, due to the existence of multiple copies of the messages in the network, the multiple-copy based protocols require *message replica control* tasks such as message flooding, dead message (garbage) removal, etc. In Epidemic Routing[52], each message is tagged with *hop-count* and the message is dropped when it reaches the number of hops equal to its hop-count. Disconnected Transitive Communication (DTC)[10] uses a similar way to remove dead messages in the network. Every message carries a *time to live* (TTL) value, and when this expires, the message becomes dead and therefore be dropped from the system. These two mechanisms work proactively and the value of hop-count or TTL need to be specific for particular networks. In [17], various mechanisms used for dead message removal are presented:

- **Drop-Oldest** (DOA): The packet that has been in the network longest is dropped.

- **Drop-Least-Encountered** (DLE): The packet is dropped based on the estimated likelihood of delivery.

- **Drop-Least-Recently-Received** (DLR): The packet that has been in the node's buffer longest is dropped.

- **Drop-Random** (DRA): The packet to be dropped is chosen at random.

The drop strategies of Drop-Oldest and Drop-Least-Encountered perform best in [17]. Also, in case of selective flooding, a node need to identify *which nodes to relay* the message.

The classification based on message copies is given in Figure 2.2.

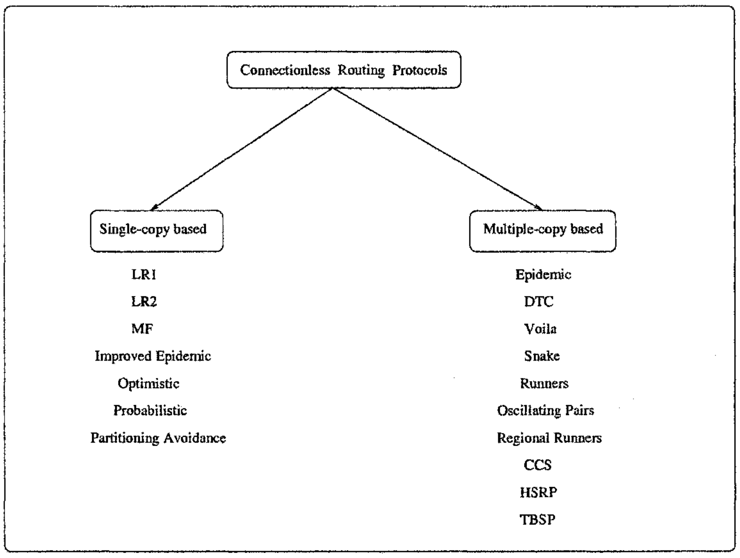

Figure 2.2: A Classification of Connectionless Protocols based on Message Copies

2.4 Summary and Analysis

In this section we briefly analyze and summarize the basic ideas, advantages, and limitations of the existing connectionless routing protocols.

Presence of stable connectivity in the network will certainly increase the efficiency of message delivery. However, maintaining such a stable connectivity is difficult or impractical for many situations such as highly mobile networks, sparse mobile networks, and the networks where mobile nodes are non-uniformly distributed. Leaving the mobile nodes to follow their natural movements may often result in partitioned networks. Achieving communication in such disconnected networks is either difficult or impossible. Therefore, replacing the strong connectivity assumption that *the next-hop or the destination is connected whenever the communication occurs* by weaker and feasible assumptions such as *predetermined pairwise connectivity* or *periodic pairwise connectivity* are more appealing. Because they are comparatively easier to ensure and often good enough for efficient message communications. Such conditions cannot support delay sensitive applications such as interactive multimedia which require low message delay. However, this environment is suitable for the delay tolerant applications.

It is apparent that the *next-hop(s) identification policies* and efficient *message drop policies* are the main factors for the performance of many connectionless routing protocols. The next-hop identification policies based on *utility* introduced in [10], *meeting likelihood* introduced in [17], *delivery predictability* introduced in [32], and *optimal relay path* introduced in [33] are interesting, yet further study is required to understand their performance under various conditions. Similarly, the message drop strategies introduced in [10, 17, 52] are attractive, but require experimental study to understand their performance. Next, we analyze non-compulsory, compulsory, and semi-compulsory routing protocols.

The advantage of non-compulsory routing protocols is that each node need not change its normal behavior (movement) compulsorily to assist in message delivery. The protocols take advantage of the natural movements of mobile nodes to carry the messages. However, if a node resides in a remote area within the network, for example the Swedish Lapland described in [18], and there is not any node moving closer to this node, then the message communication to this separated node is not possible.

Compulsory protocols may increase the efficiency of message delivery due to the active participation of every node by changing their mobility to deliver the message. But forcing every node to change its mobility pattern is too restrictive and impractical in many situations. Semi-compulsory routing protocols fall between the two extreme cases, by choosing a subset of mobile nodes (support nodes) to be dedicated for routing purpose.

Snake protocol and Runners protocol are popular semi-compulsory protocols. The maintenance of pairwise adjacency in the snake boosts the efficiency of message transfer across the support, but it takes longer time for the support (snake) to sweep the entire network area. Independent random movements of runners help the support to sweep the area faster, but increase the complexity of message management task of the protocol due to the increased randomness in the mobility of the support nodes. In [13], the snake protocol and runners protocol are implemented and compared through experiments which measure the message delay between sender-receiver pairs, the total number of message copies stored in the support structure, and the message delivery rate. The experiments show that the runners protocol overall outperforms the snake protocol. Through the analysis, it is apparent that the mobility patterns of the support nodes significantly affect the performance of the semi-compulsory protocols in ad-hoc networks. Specific mobility patterns of the support nodes can be applied to achieve better performance. The hierarchical support routing protocol[13] is appealing, in which the highway mobile node idea is similar to ferry idea introduced in [54], but requires support nodes with heterogeneous capability to achieve the hierarchy.

Another issue which need to be investigated for semi-compulsory routing protocols is how to select the support nodes and how to define the size of the support (i.e., the number of the support nodes). In [13], the number of the support nodes is predefined assuming that the network is fixed and known in advance. However, the ad-hoc network may have significant and unpredictable changes on traffic, node population, node deployment, and area size. Even an optimal support size for the initial network may become not satisfactory for the current network. Hence, in order to achieve the minimal packet delay and minimal resource expense in ad-hoc networks which are dynamically changing, an adaptive approach is required to make the support size converge to the optimal size. An adaptive compulsory protocol to reactively control the support size is presented in [11].

Chapter 3

Connectionless Semi-compulsory Routing Protocols

Based on our classification presented in Chapter 2 and the definition of semi-compulsory routing protocols, we observe the following.

Observation 1 *In one extreme of semi-compulsory class, if all nodes are compulsory then the protocols become simply compulsory. On the other extreme of the class, if no node is compulsory then the protocols become non-compulsory.*

Thus, we can consider semi-compulsory class as a general case where compulsory and non-compulsory are its extreme cases. Also, most efficient systems fall between these two extreme cases and thus fit properly into semi-compulsory class. Therefore, here after in this thesis, we focus on semi-compulsory protocols.

First we will identify the fundamental factor to achieve efficient communications for connectionless routing protocols, and then propose a unified framework which achieves the identified requirements.

From the definition given earlier, the proper semi-compulsory routing protocols assumed to have two types of mobile nodes: *regular nodes* and *support nodes.* The support nodes are compulsory, whereas the regular nodes are not. That is, only the support nodes move in such a way to facilitate effective message communication in the network.

3.1 Fundamental Characteristics

In the traditional wired networks, if a new node is added then its connectivity to the network is *stable* and therefore the message communication to any other node in the network is always *guaranteed* under normal circumstances. This is not the case in the wireless mobile networks. Though the mobile nodes are formally added to the network, the connectivity of any node to the network is normally on and off due to node mobility. Therefore, to guarantee message communication with an acceptable delay the support nodes must be aware of the possible *locations* of the mobile nodes in the network. That is, the knowledge of the *region* in which the mobile nodes can move is essential for guaranteed message routing across the entire network. We refer this region as *network region or area.*

To achieve efficient communication in mobile ad-hoc networks,

(i) the sender needs some support node in its transmission range, preferably as soon as possible, to *transfer* the message, then

(ii) the message has to be effectively *relayed* across the support, and finally

(iii) on the other side, a support node carrying the message has to move closer to the receiver, as quickly as possible, to *deliver* the message.

From these requirements, we identify the following two factors as primarily responsible for effective communication.

1. Frequent network area *coverage* by the support nodes, to establish prompt connectivity with the regular nodes.

2. *Connectivity* of the support, to relay the message within it efficiently.

Network connectivity is a fundamental requirement even for stationary networks, but the network area coverage is unique to mobile ad-hoc networks due to the mobility of the nodes. It is apparent that the connectivity of the support and its coverage of the network area are related. However, achieving both tasks simultaneously are two conflicting goals for the support when its size in number is limited. Therefore, the performance of any routing protocol in this context mainly depends on proper trade-off between these two factors. Next, we analyze the various mechanisms of these two factors.

3.1.1 Network Coverage

The network coverage by the support has essentially two subtasks: *initial deployment of support* and *support mobility*. These two tasks can be achieved by using two separate functions. The initial deployment function is usually computed once but the mobility function is computed either continuously or discretely. These functions could be either **deterministic** or **stochastic** (based on some probability distribution). Here we have four possible combinations:

1. **Deterministic Coverage** (both deployment and mobility are deterministic): The simplest deterministic deployment is grid-based, where support nodes are

located at the center of each grid. The size of grid is based on the transmission range of the support nodes. Suppose that the network area is X^*Y, and the transmission range is R, then the number of support nodes to cover the entire network is $\frac{X*Y}{R^2}$. To minimize the support size, we can increase the grid size and allow the support nodes to move around or across the grid in a deterministic way. Another example would be the popular transport system. The routes and ports are deterministic.

2. **Deterministic-Stochastic Coverage** (deterministic deployment and stochastic mobility): With the above grid-based deployment, the support nodes may be allowed to move around or across the grid using some probability distribution. Another example would be city taxi. Each taxi will have predetermined starting point and after that their mobility is driven by the demand, which is stochastic.

3. **Stochastic-Deterministic Coverage** (stochastic deployment and deterministic mobility): May be used for rescue missions or some data collections where nodes are sprayed from an airplane and then their mobility might follow predefined instructions.

4. **Stochastic Coverage** (both deployment and mobility are stochastic): In many situations, deterministic deployment is not always effective or feasible, due to various reasons. An example would be, after spraying mobile sensor nodes from an airplane the mobility of the nodes is stochastic.

The selections of suitable distribution for deployment and mobility are application dependent. Stochastic mobility pattern are popularly used in most studies.

3.1.2 Support Connectivity

Since a message normally travels across multiple support nodes before reaching the destination, support connectivity is the key factor for overall performance of the semi-compulsory routing in mobile ad-hoc networks. In some systems, maintaining the connectivity of support is handled as a separate task under the name **topology maintenance**. We consider it as a part of routing task. There are three popular ways of maintaining connectivity:

- **Always Connected:** In this case, all support nodes are always connected. So that when a support node receives a message from one neighbor, it has, at least one, another neighbor to forward. This constant connectivity requires that the support nodes move in a coordinated way to maintain it. The snake-like mobility pattern, presented in [13], is an example of always connected support.

  The advantage of the always connected support simplifies the task of message relay and message exchange policies. But it constrains the mobility of the support nodes very much.

- **Deterministically Connected:** In this case, the connectivity of a support node to another support node is approximately known. The simplest example would be **periodic connectivity**, used in city transport system. Here each support node moves to a specified location at predetermined time so that it can meet the other support nodes there, possibly to exchange the messages between them. The support presented in [54] is another example for deterministically connected support, where the support nodes called *message ferries* move around the deployed area following the known routes.

  The advantage of deterministically connected support is that the communication time between any two support nodes can be mostly predictable. Predictability is an attractive property for most practical applications.

- **Stochastically Connected:** In the above two cases, the mobility of the support has to be deterministically constrained to achieve the connectivity. If the mobility pattern of the support nodes is stochastic, then such a deterministic constraint on the connectivity may not be possible or effective. In such situations the likelihood of meeting can be measured only in terms of probabilities. Also, the explicit constraint on the mobility increases the complexity of the protocol. The independent movements of support nodes may result in uniform node distribution in the deployed area. Given a sufficient period of time, any support node can meet another support node with some probability. Hence, the basic idea of this class mobility pattern stems from the concept of **eventual meeting** of the support nodes. To increase the meeting likelihood, a constraint on the area in which a support node can make random movement can be placed.

  The advantage of this class of connectivity is its simplicity, but may require unpredictable resources (energy and storage space).

3.2 A Framework for Connectionless Semi-compulsory Routing Protocols

In this section, we present a unified hierarchical framework for connectionless semi-compulsory routing protocols in mobile ad-hoc networks. This framework unifies many existing routing protocols and therefore they can be derived easily from the framework, by specific implementation of its component policies.

3.2.1 Overview

The network area is normally assumed in theory as a *geometrical region* and in practice as a *geographical region.* The geometrical regions are usually *n-dimensional regular geometric structures* (where n=1, 2, or 3), and the popular geographical regions modeled are *cities connected by highways, some forest areas, countries, etc.* For a better management and to utilize the *locality of mobility* for message routing, the network region is normally divided into subregions, which we refer as *scopes.* For example, the mobile users in a province or state can be considered as a collection of cities and towns connected by highways. The mobile nodes in a city mainly move around their city or town. Each city or town with part of the highways connected to it can be considered as a scope of the network. Thus the routing task in mobile ad-hoc networks has three basic functions: (i) *scope establishment*, (ii) *routing within the scope*, and (iii) *routing across the scopes.* Each of these functions can be further synthesized and refined to obtain the complete framework.

We make the following assumptions for our framework.

Assumption 1 *Each node has a unique identifier.*

Assumption 2 *A regular node can send or receive message when there is a support node within its transmission range.*

3.2.2 Framework

We characterize the framework (F) for the semi-compulsory routing protocols in mobile ad-hoc networks in two levels **L1** and **L2.**

L1: This level consists of three major policy components ($F =< SEP, Intra\_SRP, Inter\_SRP >$):

- **Scope Establishment Policy** (*SEP*)**:** This policy defines the scopes (subregions) in the network and determines the support for the scopes.

- **Intra-Scope Routing Policy** (*Intra_SRP*)**:** This policy accomplishes routing within each scope.

- **Inter-Scope Routing Policy** (*Inter_SRP*)**:** This policy accomplishes routing across the scopes.

The scope establishment policy has two subcomponents, namely *scope definition policy* and *support assignment policy* for each scope. The intra-scope routing policy has four main components: *node mobility* within its scope, *next-hop(s) identification* to relay the message when necessary, *message acceptance policy*, and *message drop policy.* Inter-scope routing policy has similar mobility and message acceptance components. But it does not normally require exclusive message drop policy because it can be achieved through intra-scope message drop policy. Similarly, it does not require exclusive next-hop(s) identification policy because the meeting between support nodes of different scopes are normally designed specifically to exchange the messages between themselves. This refinement gives the policies for level **L2** in the framework.

L2: The scope establishment policy has two subcomponents ($SEP =< SDP, SAP >$):

P1: **Scope Definition Policy** (*SDP*)**:** This policy defines the boundary for each scope.

P2: **Support Assignment Policy** (*SAP*)**:** This policy first *determines* support size (number of support nodes) and then *identifies* and subsequently *deploys* the support nodes.

The intra-scope routing policy has four components ($Intra\_SRP =< NHIP$, $MDP, MP1, MAP1 >$):

P3: **Next-hop Identification Policy ($NHIP$):** This policy *identifies* the next-hop(s) to relay the message, among its neighbors.

P4: **Message Drop Policy (MDP):** The excess or undeliverable messages are removed using the *message drop policy.*

P5: **Mobility Policy ($MP1$):** The mobility policy $MP1$ of a node defines the mobility pattern of that node within its scope.

P6: **Message Acceptance Policy ($MAP1$):** The message acceptance policy $MAP1$ of a node defines what messages it has to accept to relay them further.

Finally, the inter-scope routing policy has two components ($Inter\_SRP =< MP2, MAP2 >$):

P5': **Mobility Policy ($MP2$):** The mobility policy $MP2$ of a node defines the mobility pattern of that node in order to exchange messages with other scopes.

P6': **Message Acceptance Policy ($MAP2$):** The message acceptance policy $MAP2$ of a node defines what messages it has to accept from the support nodes of other scopes to relay them further.

The mobility policy $MP2$ determines when to move, which location in a neighboring scope to move, and how long a support node waits or pauses there to exchange messages with the support nodes of other scopes.

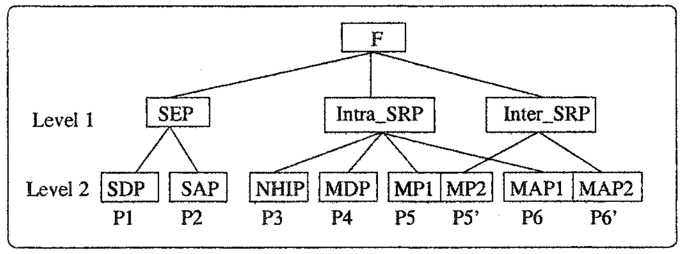

Figure 3.1: Hierarchy of Policies in the Framework

The hierarchy of policies described above is depicted in Figure 3.1.

The policies in the leaf nodes of the hierarchical tree are the *substitutable components* of the framework. By choosing a particular implementation for these policies, a connectionless routing protocol can be obtained. Some of the representative policies for the above mentioned components of the framework will be identified and listed next.

3.3 Analysis

In this section, we study some of the implementations of the substitutable components $P1, P2, ..., P6, P5', P6'$ to derive routing protocols for disconnected mobile ad-hoc networks.

3.3.1 Scope Definition

A scope is a subregion within the entire network area defined for the purpose of effective management of routing. The simplest case for defining scopes would be to consider the entire region as a single scope. If the network region is a regular geometric shape (n-dimensional cube or sphere) then the natural choice for the scope is its scaled down shape. Also, the transmission range could be incorporated in defining the boundary for a scope. In the network of cities connected by highways (also referred as *city graph*), each city with its associated highways can be considered as a scope. Scopes may overlap to increase the connectivity. Establishing fixed inter-scope meeting points, we refer as *ports*, would help to facilitate the meeting between nodes from different scopes. Once the scope is established, it could be either fixed or changing over the period of time. For example, like nomadic communities[18] move from location to location together, groups of mobile nodes collectively move from one point to another. Each group forms a *moving scope* whose size and boundary keep changing. The interconnections and ports between scopes also change accordingly.

3.3.2 Support Assignment

The support assignment policy is to determine the support size and then identify and deploy the support nodes. The semi-compulsory protocols only use a small sized support to achieve efficient message communication in finite amount of time. The size of the support, i.e., the number of support members, significantly effects the performance of semi-compulsory protocols, since only the support nodes take the responsibility of message relay.

The following parameters can be used to determine effective support size:

- T_o - the network topology including the network area, node population and node deployment,
- T_f - the traffic generated by the regular nodes,
- R_c - the average resource consumed by the routing process,
- D_l - the average message delay and
- R_t - the overall message delivery rate.

The parameters can vary with the particular requirements in various environments.

The support assignment policy could be either **static** and **adaptive**.

1. **Static:** In this case, the support size and the support nodes are predetermined in the beginning, using a deterministic function.

2. **Adaptive:** The ad-hoc networks may experience significant unpredictable changes in traffic, node population, and area size. Even an optimal support size for the initial network may become not satisfactory for the current network. Along with the topology and traffic changes, the performance of the previous support will vary. The message delay and delivery rate may increase or decrease. In addition, some support nodes may experience the overflow of their message storage. Simply dropping the extra messages is not a good idea. Furthermore, some support nodes may fail to carry on the routing responsibility due to their own reasons. Therefore, in order to achieve a better protocol performance in ad-hoc networks which are dynamically changing, the support size and support nodes need to be adaptively determined based on some system parameters. The paper [11] presents a routing protocol which adapts to its support size.

After the support size is identified, the second crucial issue is which mobile nodes should be elected as support nodes.

In the previous work, a mobile ad-hoc network is usually assumed to be homogeneous, where each mobile node has the identical capability. The traditional hierarchical protocols utilize specialized nodes, such as *cluster heads, group leaders*, or *route gateways*, to coordinate the routing process. Many leader election schemes have been proposed in the clustering literature. Two distributed clustering protocols, *Lowest-ID* protocol and *Highest-Degree* protocol are widely used due to their simplicity.

The parameters that can be considered for choosing a cluster head or group leader are generally *power supply, transmission range, movement speed, memory space, processing capability, etc.* The selection formula might become complex if privacy issue which effects the node's *willingness* to take part in the routing is also considered. For example, some nodes even with high-level capability may not be willing to participate in the message relay.

In this thesis, we just make a first step in choosing a cluster head or leader including the willingness factor. For that we introduce a concept called *availability degree*, to evaluate the probability of a node participation in routing. That is, the node with high availability degree has the higher chance of becoming a support node. The availability degree has two components, we call them as: *survivability* and *willingness.* Survivability is based on resource factors and willingness may be expressed in probability, between 0 and 1. A more quantitative analysis and specific formulas may be derived using these factors, which is beyond the scope of this thesis.

3.3.3 Support Mobility

The success of establishing proper connectivity and covering the network area mainly depends on the mobility of the support nodes. Many models to characterize or simulate the mobility of nodes are proposed in the literature[6]. Mobility model is to generally mimic the movements of some real mobile nodes. The mobility model for $MP2$ is usually simple and deterministic. Varieties of mobility models are available for $MP1$. Mobility models can be classified as either *traces* or *synthetic models*[6].

1. **Traces:** Those mobility patterns that are observed in real life. Traces provide accurate information, when it is collected for large population over a long period,

but normally hard to obtain.

2. **Synthetic Models:** Attempt to realistically represent the behaviors of mobile nodes without using traces. Synthetic models are again classified into *entity mobility models* and *group mobility models.*

   The popular entity models are: *random walk mobility model and its derivatives, random way-point mobility model, a boundless simulation area mobility model, Gauss-Markov mobility model, a probabilistic version of the random walk mobility model*, and *city section mobility model.* The main group mobility models are: *exponential mobility model, column mobility model, nomadic community mobility model, pursue mobility model,* and *reference point group mobility model.* A detailed discussion of these models is given in [6].

3.3.4 Next-hop(s) Identification

When a node receives a message, it can relay it in two ways:

1. **Single-copy based:** simply relays the original message.
2. **Multiple-copy based:** duplicates and relays the copies to more than one node to increase the reachability of the message to its destination.

The case 2 will result in multiple copies in the network, whereas in case 1 the network will contain at most one copy of a message at any time. After receiving a message, the support node should identify exactly one support neighbor in case of single-copy based routing and more than one support neighbors in case of multiple-copy based routing, to relay that message. For multiple-copy based routing, the simplest case would be that the support node relays the copies to all support nodes that it meets. This achieves the highest probability of the messages spread in the support. Using such flooding-like mechanism to spread the undelivered messages generates large numbers of duplicate copies of the messages. For efficient utilization of the limited storage space and reduction of the traffic overhead between the support nodes, better next-hop identification methods need to be used. With some kind of guidance information, the support nodes can identify the support nodes which might move closer to the actual destination of the message.

How often the process of identification invoked is based on rediscovery interval. It could be periodic where the period is tunable or/and adaptive (whenever a new neighbor joins, etc.).

3.3.5 Message Acceptance

When two support nodes establish connectivity, they exchange their stored messages according to the message acceptance policy. The exchanged messages include the messages for the regular nodes in the same scope or the others scopes. When a node relays a certain message to a neighbor, the receiving node may choose to ignore, drop, or deny the acceptance of that message due to various reasons such as resource constraints, security concerns, etc. We categorize the message acceptance policies into two types based on whether the two meeting support nodes are from the same scope or not.

1. **Message Acceptance Policy ($MAP1$):** The message acceptance policy $MAP1$ of a node defines what messages it has to accept in order to relay within its scope. Normally the message exchange task is combined with the next-hop(s) identification task. On receiving a new message from the regular node, the support node decides the next-hop among its neighboring support nodes, nego-

tiates for message acceptance if necessary, and then forwards the message. In a simple message exchange case, when two support nodes meet, they compare the message lists of both sides and make up what the other side lacks.

2. **Message Acceptance Policy ($MAP2$):** The message acceptance policy $MAP2$ of a node defines what messages it has to accept from the support nodes of other scopes in order to relay within its scope and across its scope. When a support node meets another support from the other scopes, it only accepts the messages for the regular nodes in its scope and the messages which have to be transitively relayed to the other scopes.

3.3.6 Message Drop

A node may drop dead or likely to be dead messages to increase the utilization of the network resources. The message drop policy is applicable only in the multiple-copy based case. There are two situations in which the messages could be removed:

1. **Immediately after the delivery of message to its destination** - the support node which delivers the message to the destination sends an explicit request to the other support nodes to delete the copies.

2. **Independent of the message delivery to its destination** - the support node might drop the messages based on some predetermined parameters.

   There are two popular parameters used to drop a message in case 2:

   2.1 **Hop-count** - the number of hops that the message has traveled in the network.

Table 3.1: Analysis of Existing Protocols in the Framework

Routing Protocol	Main Components			
	Support Assignment	**Mobility**	**Copies to be relayed**	**Message Drop**
DTC[10]	Adaptive	Non-compulsory	Multiple	Time-to-live
Epidemic[52]	Adaptive	Non-compulsory	Multiple	Hop-count
Voilà[45]	Adaptive	Non-compulsory	Multiple	Time-to-live
Improved Epidemic[17]	Adaptive	Non-compulsory	Single	N/A
Probabilistic[32]	Adaptive	Non-compulsory	Single	N/A
Optimistic[8]	Adaptive	Non-compulsory	Single	N/A
LR1[33]	Adaptive	Compulsory	Single	N/A
LR2[33]	Adaptive	Compulsory	Single	N/A
Partitioning Avoidance[21]	Adaptive	Compulsory	Single	N/A
Snake[13]	Static	Semi-compulsory	Multiple	After Delivery
Runners[13]	Static	Semi-compulsory	Multiple	Not mentioned
Oscillating Pairs[49]	Static	Semi-compulsory	Multiple	Not mentioned
Regional Runners[49]	Static	Semi-compulsory	Multiple	Not mentioned
CCS[14]	Static	Semi-compulsory	Multiple	Not mentioned
HSRP[11]	Static	Semi-compulsory	Multiple	Not mentioned
Message Ferrying[54]	Static	Compulsory	Single	N/A

2.2 **Time to live** - predefined period of time starting from existence of that message in the network.

3.4 Analysis of Existing Protocols with respect to our Framework

Based on the components of our framework, we present a higher level classification of existing connectionless routing protocols of mobile ad-hoc networks, in Table 3.1.

From the table, we can derive many commonalities and differences among the existing protocols. For example,

- The protocols in [10, 52, 45, 17, 32, 8, 33, 21] differ with the protocols in [13, 49, 14, 11, 54] in support assignment policy.

- Though the protocols in [10, 52, 45] belong to the same class with respect to support assignment, mobility, and copies to be relayed, they differ in message drop policy.

- The protocols in [17, 32, 8, 33, 21, 54] use single copy to be relayed and therefore no message drop policy is needed.

- Though the protocols in [17, 32, 8] and the protocols in [33, 21] have the same support assignment and copies to be relayed policies, they differ in mobility policy.

- The protocols in [13, 49, 14, 11] belong to the same class with respect to all the four policies, support assignment, mobility, copies to be relayed, and message drop in higher level. However, they have differing behaviour and performance.

- The protocols in [13, 49, 14, 11] differ with the protocol in [54] in mobility, copies to be relayed, and message drop policies.

Chapter 4

New Connectionless Routing Protocols

In this chapter, we present four simple connectionless routing protocols. As mentioned in Section 2.2.6, the first protocol called three-base support protocol (TBSP) is proposed to fill the vacuum in the taxonomy. The other three protocols called oscillating pairs protocol, regional runners protocol, and center concentrated support protocols are strictly in semi-compulsory class.

4.1 Three-Base Support Protocol (TBSP)

We use the ideas mainly from MF[54] and HSRP[11] to design TBSP. Message Ferrying scheme deploys ferry nodes moving around the network area along the known routes to collect and deliver messages. The regular nodes are compulsory to move closer to the ferry route to send or receive messages. In HSRP, the network area is divided into city scopes. City support nodes operate in each scope and the message relay between different scopes relies on the mobility of highway support nodes.

TBSP inherits its non-uniform support property from HSRP and its compulsory property from MF.

4.1.1 The Protocol (TBSP)

- The network is modeled as city graph used in [13] and a mobile node could be either support node or regular node.

- The support nodes are classified into three categories: *city-taxi, city-bus*, and *inter-city-bus.* City-taxis do random movement, like runners in [13], aiming to cover the entire city, collecting and delivering messages along its way. City-buses follow regular routes within the city, like message ferries in [54], collecting and delivering messages in the city. Inter-city-buses follow the predetermined routes and schedules between cities, like highway mobile users, collecting and delivering messages between different cities. When the support nodes of the three classes meet, they exchange the undelivered messages.

- If a regular node which has a message to send cannot meet a city-taxi in finite time, it moves closer to its neighboring bus to deliver the message. That means a regular node is compulsory only if it changes its original movement to move close the bus route. Unlike in MF scheme, the regular nodes need not periodically move to the bus routes to collect the messages designated to itself. That is, the message delivery from the support nodes to the regular nodes does not require the trajectory modification of the regular nodes. It relies on the pairwise meeting of the regular node and the city-taxi or city-bus.

The protocol inherits its simplicity from the conventional message ferrying, highway mobile vehicle, and random runners. It eliminates the weakness of message ferry routing strategy that the uncertainty in the mobility of message receivers. We leave further analysis and performance study for future research.

4.2 New Semi-compulsory Routing Protocols

The motivation for our semi-compulsory routing protocol is mainly derived from the analysis of snake and runners protocols. The main idea of snake protocol is that the support nodes are organized in a snake-like sequence and move in a way determined by the snake's head. The support nodes always remain *pairwise adjacent*. The messages are stored in every node of the support and when a receiver comes within the transmission range of a node of the support, the receiver is notified and the message is then forwarded. After delivering a message, a control message is flooded across the support to remove the duplicate messages.

In runners protocol, instead of maintaining pairwise adjacency between members, all support nodes sweep the entire area by independent random walk. When two runners meet they exchange undelivered messages. In runners protocol, the connectivity constraint is relaxed and the runners are allowed to make *independent random movements* to sweep the network area faster. This increases the chance of message delivery faster, but the uniformity of network coverage and the frequency of connectivity are still not guaranteed.

We believe that by carefully defining scope and then selecting appropriate support and its mobility pattern, the connectivity and coverage of the network can be improved very much. And that will in turn increase the efficiency of the routing protocol. Based on this intuition, we propose three simple semi-compulsory protocols called *Oscillating Pairs Protocol*[49], *Regional Runners Protocol*[49], and *Center*

Concentrated Support Protocols[14]. Thus the objective of these three protocols is to assure the basic requirements for a guaranteed and effective routing (the frequent network coverage of support nodes while maintaining proper connectivity among them), discussed in section 3.1.

The idea common to all protocols is that the network area is divided into smaller scopes and the mobility (and hence responsibility) of the support nodes is localized within their scopes. These protocols mainly differ in the way of the scope definition, support assignment, and support mobility, are determined. For all our protocols, we assume that the network area size and support size are fixed.

4.2.1 Oscillating Pairs Protocol

For simplicity, we assume that the network area is approximated to rectangular region.

- We divide the network area into parallel stripes as scopes, whose size is based on the transmission range of the support nodes.

- A pair of support nodes is assigned to each scope and they always maintain the connectivity between them. For our discussion, without loss of generality, we assume that the stripes are vertical and the support pair in the i^{th} stripe is denoted by $< ls_i, rs_i >$ (left and right support of the scope i).

- The support pair in each scope oscillates between the top and the bottom of their scope, with common speed, to *cover* the scope, as shown in Figure 4.1. Note that, for the optimal coverage, the support nodes need not move to the boundaries of their scope. They move only close enough to communicate with the regular nodes on the boundaries. This is indicated by a dotted rectangle within the network region.

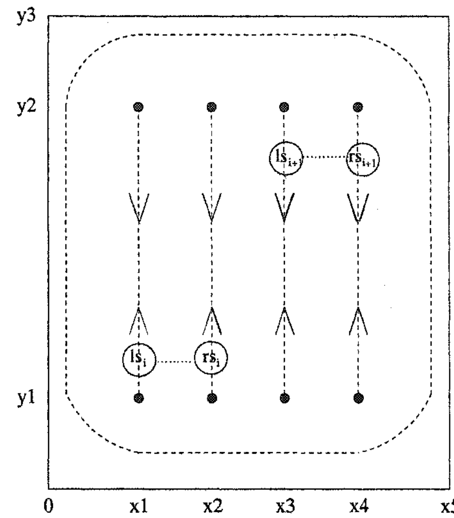

Figure 4.1: The Mobility Pattern of Oscillating Pairs

The connectivity of the support pair of each scope is maintained to pass the message from one side of the scope to the other side. If the support pairs of adjacent scopes i and $i+1$ are at the same height, then rs_i and ls_{i+1} can communicate and exchange messages.

Analysis

The protocol guarantees deterministic coverage of the network by moving along the fixed route repeatedly. That is, any point in the network can be swept by some support node at a definite time. It maintains the connectivity of support nodes within the scope, like in the snake protocol, and establishes periodic connectivity with the support pairs in its neighboring scopes.

The maximum message delay can be computed as follows. Let T be the time taken for the support node to travel from one end to the other end. A regular node can meet a support pair within $2T$ units of time. A support pair can meet the support pair of its neighboring scopes within T units of time. If there are k scopes, then within $2 * 2T + (k-1) * T = (k+3) * T$ units of time the message can reach the receiver.

The protocol covers the area uniformly and periodically and assures deterministic connectivity of support nodes, therefore, assures guaranteed message delivery.

Next we present another routing protocol which does not maintain any stable connectivity but assures uniform stochastic connectivity and coverage across the entire network.

4.2.2 Regional Runners Protocol

This protocol may be viewed as a variation of runners protocol presented in [12, 13].

- The network region is divided into smaller subregions as scopes. A single or a group of support nodes is assigned to each scope, as shown in Figure 4.2A.

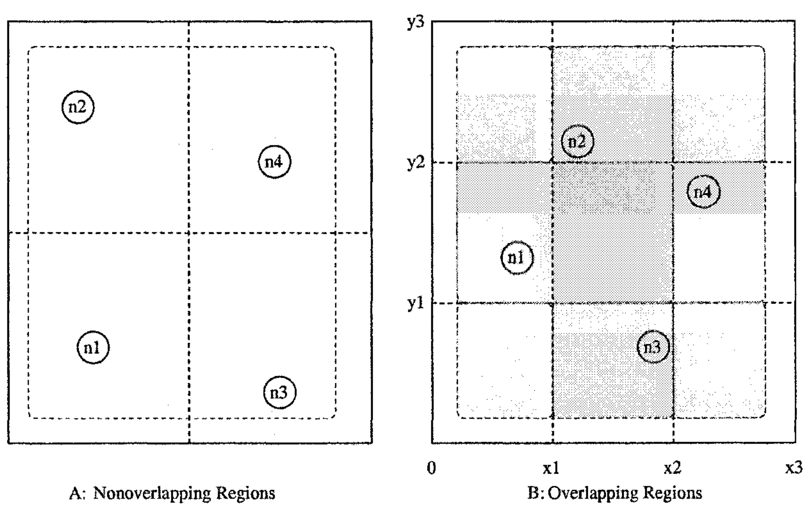

Figure 4.2: The Mobility Pattern of Regional Runners

- The support nodes make independent random walks and exchange messages when two support nodes meet, similar to runners protocol. But in our protocol the movements of support nodes are restricted within their respective scopes. Therefore we call the support nodes in this protocol as **regional runners**.

Meeting between two regional runners of the same scope facilitates the message routing within that scope and meeting between two regional runners from adjacent scopes facilitates the message routing between their scopes.

Analysis

Observation 2 *Dividing the entire region into the smaller subregions and restricting the regional runners within their scopes avoid the possibility of clustered random movements and hence increase the uniformity in the network area coverage. However, a regional runner can meet another runner from its neighboring scope only if both move closer to their common boundary.*

From Observation 2, it is easy to see that the connectivity in the setup given in Figure 4.2A is weaker. This weakness is reflected in the performance of the routing protocol that we verified through simulation.

To increase the connectivity among the regional runners, the protocol is refined to overlap the subregions as shown in Figure 4.2B. In this case the scopes for the regional runners $n1, n2, n3$ and $n4$, respectively, are ((0, 0), (x2, 0), (x2,y2), (0,y2)), ((0, y1),(x2, y1), (x2, y3), (0,y3)) ((x1, 0), (x3, 0), (x1,y2), (x3,y2)), and ((x1, y1),(x3, y1), (x3, y3), (x1,y3)).

The scopes are made to *overlap* in order to increase the probability of the *support connectivity* between adjacent scopes. Hence with initial uniform deployment of runners, the restricted overlapping movement of runners within their scopes assures both the uniform stochastic connectivity and uniform stochastic coverage of the network. And this increases the efficiency of message delivery that we verified through simulation.

4.2.3 Center Concentrated Support Protocols

In this section we present four simple protocols called Center Concentrated Support (CCS) protocols in which one protocol is studied through simulation.

In this thesis, we use the Random Way-point Mobility Model as the random mobility pattern of the mobile nodes. In [6], Camp, Boleng and Davies present the simulation results of various mobility models. In Random Way-point Mobility Model, a mobile node begins by staying in one location for a certain period of time (i.e., a pause time). Once this time expires, the mobile node chooses a random destination in the simulation area and a speed that is uniformly distributed between some *minspeed* and *maxspeed*. The mobile node then travels toward the newly chosen destination at the selected speed. Upon arrival, the mobile node pauses for a specified time period before starting the process again. In the Random Way-point Mobility Model, the probability of a mobile node choosing a new destination that is normally located in the center of the simulation area, or a destination which requires travel through the middle of the simulation area, is high. That is, in the Random Way-point Mobility Model, the clustering of mobile nodes occurs near the center of the simulation area. Thus, the mobile nodes appear to converge, disperse, and converge again repeatedly.

We derive the CCS protocols based on the following simple observations.

Observation 3 *In the Random Way-point Mobility Model, if the total network region is divided into center region (CR) and outer region (OR), then CR has the higher probability of visits by mobile nodes than OR.*

Observation 4 *Maintaining better connectivity in the region where more visits are probable, that is CR, would increase the efficiency of message delivery.*

We assume that the network area is divided into center region (CR) and outer region (OR) and the number of support nodes in the network is *fixed*. In our protocols, the support nodes in CR and OR are maintained differently in such a way that the support nodes in CR would aim to provide better connectivity in CR and the support nodes in OR are concerned only about the network coverage. We list some of the choices, each giving a routing protocol for mobile ad-hoc networks.

CCS1: More nodes are assigned in CR and they are restricted to make random walk within CR. This is to maintain better connectivity in CR. The remaining nodes are deployed in OR initially and they are allowed to make random walk in the entire network region.

CCS1': More nodes are assigned in CR and they are restricted to make random walk within CR. The remaining nodes are deployed in OR initially and they are allowed to make random walk within OR.

CCS2: More nodes are assigned in CR and they follow snake protocol within CR. This is to maintain constant connectivity in CR. The remaining nodes are deployed in OR initially and they are allowed to make random walk in the entire network region.

CCS2': More nodes are assigned in CR and they follow snake protocol within CR. The remaining nodes are deployed in OR initially and they are allowed to make random walk within OR.

As shown in Figure 4.3, the network area is divided into Center Region and Outer Region. The support nodes n1, n2, n3, n4 and n5, are restricted to move in CR to achieve better connectivity in CR, and the others support nodes n6, n7, and n8, can randomly move in the entire network area for the network coverage.

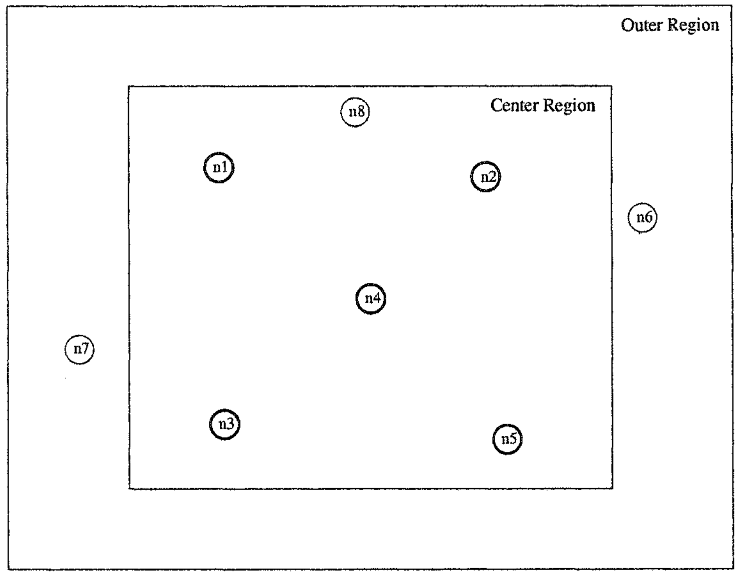

Figure 4.3: The Mobility Pattern of Center Concentrated Support

Table 4.1: Coverage and Connectivity of the Support

Routing Protocol	*Network Coverage*	*Support Connectivity*
Snake	Stochastic	Constant
Oscillating Pairs	Deterministic	Periodic
Runners	Stochastic	Stochastic
Regional Runners	Uniformly Stochastic	Uniformly Stochastic
CCS1	Non-uniformly Stochastic	Non-uniformly Stochastic

We conducted a limited simulation study on CCS1 and found that it performs better than both snake and runners protocols. We are continuing the performance study on other protocols.

4.2.4 Summary

The support characteristics of Snake, Runners, Oscillating Pairs, Regional Runners, and CCS1 are summarized in Table 4.1.

Chapter 5

Implementation

In this Chapter, we discuss a way of implementing our protocols. First, we design the data structures and variables used in our protocols.

Each node has a unique node ID. Each regular node keeps a message queue to temporarily store the generated messages when waiting for support nodes. Each support node needs to store all undelivered messages in a message table, and keep a list of receipts to inform the other support nodes to remove the delivered messages from their storage. The message table is indexed by the sender ID, the receiver ID, and the message sequence number. The receipt list only keeps the summary of the delivered messages, i.e., the message indexes. When a support node receives a new message from a regular node, it stores the message in its message table. If a message is forwarded to the designated receiver, the support node removes the message from the message table and inserts its summary into the receipt list.

We identify the following components in the routing protocols:

1. **Regular Nodes**:

    - Message Generation: We assume a uniform message generation. That is, the sender-receiver pairs are randomly selected and the message generation is in uniform distribution.

- Neighbor Sense: Sense the neighboring support nodes.
- Message Sending: Send generated messages to the neighboring support node.
- Message Receiving: Receive the new messages for itself from neighboring support nodes.

2. **Support Nodes**:

- Neighbor Sense: Sense the neighboring nodes, including support and regular nodes.
- Mobility Policy: Define the mobility pattern of support nodes.
- Message Acceptance:
  - Message Receiving: Receive new messages from regular nodes.
  - Message Sending: Send the undelivered messages to the designated regular node.
  - Message Synchronization: Exchange the undelivered messages between the support nodes.
- Message Drop: Drop the delivered messages, and the dead messages when the storage space overflows.

Because only the support nodes perform the message storage and forwarding, if a regular node has messages to send, this node needs to sense the support nodes around itself. It is not guaranteed that each regular node has a wireless link to some support node at any time instance. When some member of the support enters the transmission range of the sender, the sensing mechanism notifies the sender to send out the messages. On the other hand, each support node carrying the undelivered messages needs to have the full knowledge of its neighboring nodes. While a new regular node enters the transmission range of a support node, the support node checks whether there are undelivered messages for this regular node and then forwards these messages to it. When two support nodes are within transmission range of each other, they initiate the message synchronization procedure to exchange the undelivered messages.

To sense the neighboring nodes, all nodes including regular and support nodes broadcast beacon messages to indicate their existence. The regular node which has a message to send starts the transmission only if it is within the transmission range of some support node. Each support node listens to the beacon messages to sense all its neighboring nodes.

When the sensing mechanism indicates that a regular node has a message to send, the support node receives this message and store it in the message table. When a new regular node enters the transmission range of a support node, the support node looks up the undelivered messages for this regular node in its message table. After forwarding the messages to the designated regular node, the support node removes it from its message table and inserts the message summary into its receipt list.

When two support nodes meet, they exchange all stored messages and receipts to make sure the information is up-to-date. A simple message exchange is applied on the connected support nodes in the oscillating pairs protocol. On receiving a new message from the regular node, the support node forwards the message to its connected support node. After forwarding a message to its receiver, the support node notifies its connected support node to remove the message from its storage. This can assure the message synchronization between the support nodes of an oscillating pair.

For the message exchange between the oscillating pairs, whose connectivity is periodic, or the support nodes in the regional runners protocol, whose connectivity is eventual, we borrow the ideas presented in [12, 52] and propose a three-phase scheme to carry on the message exchange. When two support nodes meet, they compare the message tables of both sides and make up what the other side lacks. The message exchange is accomplished by the following three phases:

1. The support node with the higher id S_2 sends the summary of its message table, denoted by T_2, which only includes the keys of the table, and the receipt list R_2 to the support node with the lower id S_1. The support node S_1 combines them with its own to compute the new set T_1 and R_1: $R_1 = R_1 \cup R_2$, and $T_1 = (T_1 \cup T_2)$ - R_1.

2. The support node S_1 then sends the new set R_1 and the messages not stored in T_2 to the support node S_2, and request the messages not stored on S_1.

3. The support node S_2 sends the requested messages to S_1.

This implementation is exercised in our simulation study that we present next in Chapter 6.

Chapter 6

Simulation Study

To compare performance of our semi-compulsory routing protocols with the performance of snake and runners protocols, we conducted a limited simulation study using ns-2[38] developed at University of California at Berkeley. The simulator ns-2 is an object-oriented, discrete event driven network simulator written in C++ and OTcl. The ns-2 is extended with radio propagation that models signal capture and collision. The simulator also models node mobility, allowing for experimentation with ad-hoc routing protocols that must cope with frequently changing network topology. Finally, the ns-2 implements the IEEE 802.11 Medium Access Control (MAC) protocol.

We implemented our semi-compulsory routing protocols using the ns-2 packet-level simulator. We examined the C++ class hierarchy and derived the new routing agent classes in order to make them suitable for our simulation.

6.1 Experimental Setup

Our experiments are to evaluate the routing performance of semi-compulsory protocols in disconnected ad-hoc networks. Therefore, we deploy the fixed number of regular nodes in comparatively large area, which makes the connectivity of the network weak. The detailed experimental setup is listed as follows:

- **Network area size** - 1200m × 1200m and 1800m × 1800m.
- **Transmission range** - 250 meters.
- **Regular Nodes:**
  - *Number of nodes* - 50,
  - *Initial deployment* - uniformly random, and
  - *Mobility* - move according to the "random way-point" model adopted in the rectangular field. The node chooses a random destination and a random speed between 0 and 20 m/sec; and moves to that destination with chosen speed; then pauses for some random seconds after reaching the destination; and repeats the process.
- **Support Nodes:**
  - *Number of nodes* - 4, 6, 8, 10, 12, 14, 16, and 20, and
  - *Initial deployment and Mobility* - defined by the routing protocols.
- **Message:**
  - *Generation* - in the time period of 1000 seconds, 5000 data messages are generated between the regular nodes in a uniform distribution which means each node generates a new message approximately every 10 seconds, and
  - *Source and destination* - random selection.

- **Experiment Stop Point** - until 5000 data messages relayed to the designated receivers.

- **Run Times** - all five protocols are run 10 times with identical environment conditions to obtain the average performance metrics.

6.2 Performance Metrics

When comparing routing performance, the metrics play an important role. In [5, 44], the following metrics that have been often used to evaluate the routing performance of connection-oriented routing protocols are discussed:

- *Packet Delivery Ratio*: The packet delivery ratio is the ratio between the number of packets sent by the source and the number of packets actually received by the destination. This metric is important because it measures the loss rate of packet delivery. It is desired that the routing protocol achieves a high level of delivery ratio to carry out the successful packet communication between the source and destination.

- *Routing Overhead*: The routing overhead is the total number of routing packets transmitted during the simulation. This metric shows the utilization efficiency of wireless bandwidth which is often limited in wireless system.

- *Average Message Delay*: The average packet delay is the time interval between the time when a data packet is given to the network layer at the source and the time when the packet arrives at the network layer of the destination.

- *Path Optimality*: The optimal path is usually defined as the shortest path between the source and destination. The path optimality is to decrease the difference between the number of hops a packet actually took to reach its destination and the length of the shortest path that physically existed through the network when the packet was originated.

However, the metrics required depend on the structure and properties of the network. The above metrics are to solve the problems and different aspects of connection-oriented routing protocols. They may not be suitable for connectionless routing protocols in disconnected ad-hoc environments. For example, in connectionless semi-compulsory protocols, the source does not attempt to construct the complete path to the destination. Therefore, the source does not have to collect and store the routing information. The bandwidth for routing purpose is only consumed by the message exchange between the support nodes. On the other side, the undelivered messages are stored on the support nodes. We are interested in the storage consumption on the support nodes. Hence, in our simulation, we use the following three performance metrics:

- *Average Message Delay*: For each data message, we calculate the message delay as the time difference between its creation at its source and receipt at its destination. The delay consists of several smaller delays that add together. These delays may include the waiting time spent in message queue, forwarding delay, and propagation delay (the time for the travel through the medium).

- *Message Delivery Ratio*: The message delivery ratio is the rate of the total number of messages successfully received by the destinations to the total number of messages generated by the sources.

- *Average Number of Message Copies*: In this thesis, we examine the average number of duplicate copies stored on each support node.

6.3 Result Analysis

In this section, we use the three metrics mentioned above to compare the performance of the five protocols: Snake, Runner, Oscillating Pairs, Regional Runners, and CCS1. First we conduct the experiments on the four protocols: Snake, Runner, Oscillating Pairs, and Regional Runners for two different area sizes, $1200m \times 1200m$ and $1800m \times 1800m$. Figures 6.1 & 6.2 highlight the relative performance of the four protocols.

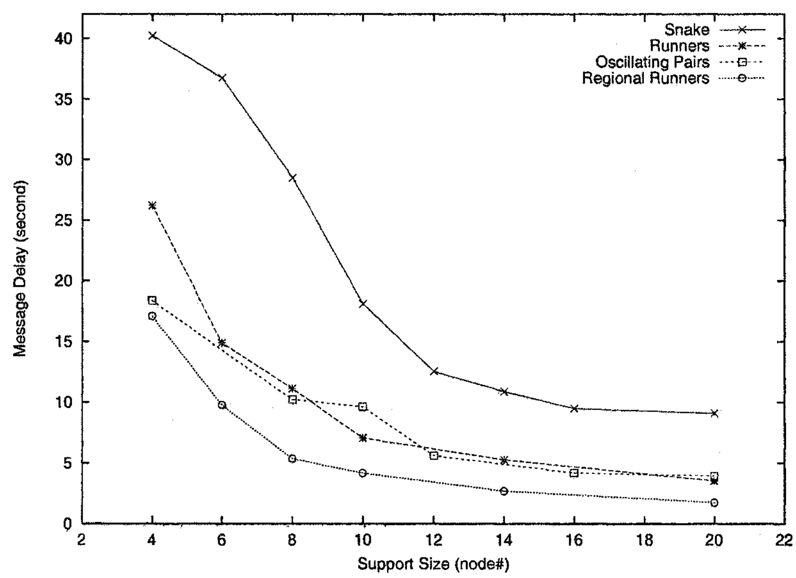

Figure 6.1: Average Message Delay vs. Support Size (Area Size 1200m × 1200m)

From Figures 6.1 and 6.2, we list the following observations about the average message delay.

Observation 5 *The relative performance of snake and runners protocols indicated in [12] is confirmed.*

That is, the message delay of runners protocol is less than that of snake protocol.

Observation 6 *For all four protocols, the average message delay drops rather quickly when the support size is small, but after some threshold value the improvement is very little.*

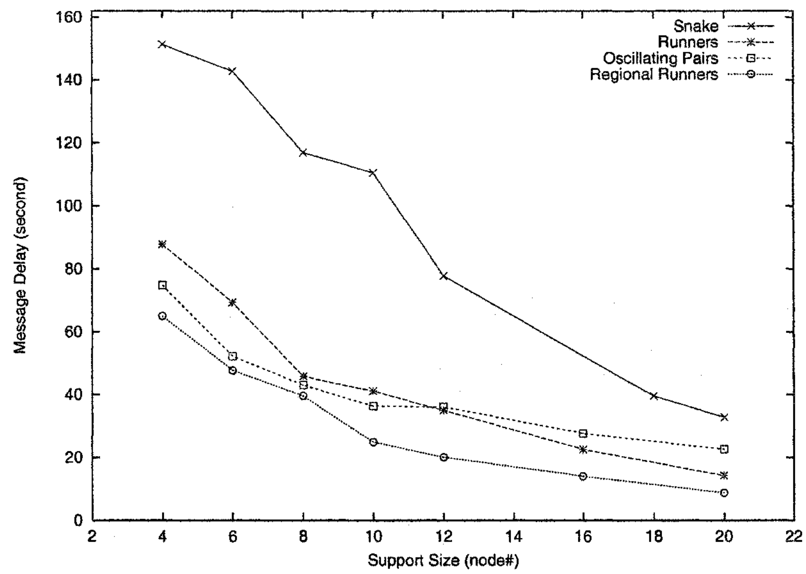

Figure 6.2: Average Message Delay vs. Support Size (Area Size 1800m × 1800m)

If the message delivery relies on a small sized support, the probability of a support node meeting the source or destination is very low. When the support size increases, the support nodes achieve better network coverage which decreases the meeting time between the regular nodes and support nodes. In addition, it guarantees the support connectivity which speeds up the spread of messages across the support. Hence, the message delay drops quickly. However, when the support nodes almost cover the entire network area, the source and destination can meet the support nodes in a short time anyway. Therefore, increasing the support size after such threshold value will not improve the performance much.

Observation 7 *The oscillating pairs protocol and regional runners protocol outperform the snake protocol.*

Observation 8 *The oscillating pairs protocol performs better than the runners protocol when the support size is small.*

However, when the support size increases, the curve of the oscillating pairs proto-

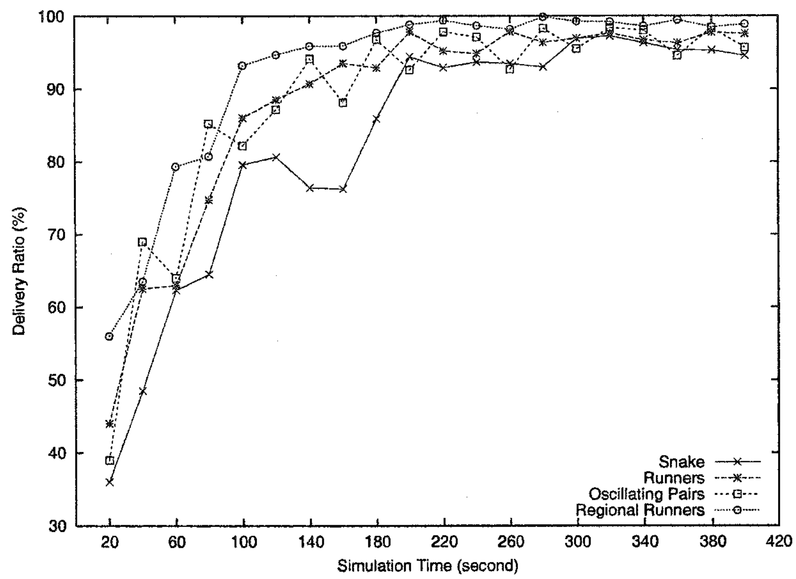

Figure 6.3: Delivery Ratio vs. Simulation Time (Area Size 1200m × 1200m)

col drops more slowly than that of the runners protocol. After some threshold value, the average message delay is similar to or even worse than the runners protocol.

Observation 9 *The regional runners protocol performs better than the runners protocol.*

As expected, the regional runner protocol achieves a more efficient trade-off between network area coverage and support connectivity. We conclude that the regional runners protocol outperforms the other three protocols in its class.

In Figures 6.3, we compare the delivery ratio in the first 400 seconds with the area size of $1200m \times 1200m$ and support size of 10.

Observation 10 *As the simulation goes on, the delivery ratios of all four protocols increase and reach high levels.*

As mentioned in the experimental setup, the regular nodes generate new messages in a uniform distribution. In the beginning of the simulation, few messages are successfully

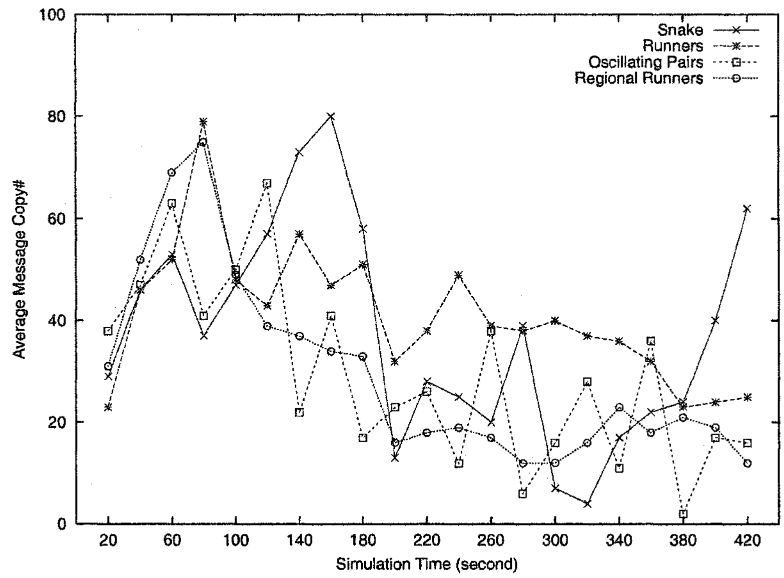

Figure 6.4: Average Message Copy Number vs. Simulation Time (Area Size 1200m × 1200m)

delivered to the destinations. The delivery rate is low at the beginning. As the simulation goes on, the messages stored on the support nodes are relayed to their destinations, which increases the delivery rate.

Observation 11 *The snake protocol is slower in delivering messages than the other three protocols.*

As shown in Figures 6.1 and 6.2, the message delay of the snake protocol is the worst which implies that the support nodes take longer time to meet the source and destination. Hence, the number of successfully delivered messages is less than the other protocols at any moment.

Observation 12 *Obviously, the regional runners protocol increases the delivery ratio much faster than the other three protocols, and reaches the highest level. It performs the best.*

This observation confirms the performance of the regional runners regarding the message delay.

Observation 13 *An interesting observation is that the delivery ratio of the oscillating pairs protocol fluctuates.*

This could be explained by the fact that the oscillating pairs periodically oscillate within their scopes.

Another metric to evaluate the performance is to measure the average number of message copies on support nodes, shown in Figure 6.4.

Observation 14 *In the beginning phase, the average message numbers of all four protocols increase quickly, then drop after some threshold values.*

Observation 15 *The regional runners protocol outperforms the other three protocols.*

Observation 16 *The oscillating pairs protocol performs better than the runners protocol, but similar to the curve in delivery ratio graph, the average message number is also fluctuating.*

If the overall message delay is worse, the undelivered messages will remain for a longer period within the support until the destination is encountered. Therefore, the average number of undelivered messages increases while more messages are still pending.

Observation 17 *The snake protocol performs worst and the curve has no convergence.*

The snake protocol is distinguished from the other three protocols by its support connectivity. When receiving a new message, the support node forwards this message to all the other support nodes. After delivering a message to its destination, the support node notifies the other support nodes to remove this delivered message using an explicit control message. Since the support in snake protocol is always connected, a new message spreading across the support and dropping a delivered message happen quickly. Therefore, the number of messages in the snake protocol fluctuates randomly.

To compare the CCS1 with snake and runners protocols, we conduct a limited simulation study in the same environment as above. We use the message delay as the parameter to compare the performance of the three protocols: Snake, Runners, and CCS1. We conduct the experiment for two different area sizes, $1800m \times 1800m$ and $2400m \times 2400m$. Figures 6.5 & 6.6 highlight the relative performance of Snake, Runners, and CCS1.

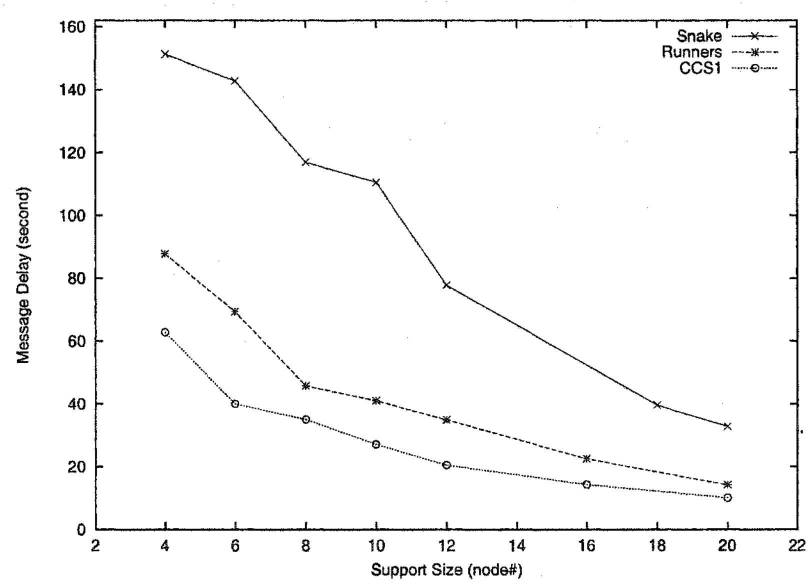

Figure 6.5: Average Message Delay vs. Support Size (Area Size 1800m × 1800m)

Since the support is constantly connected in snake protocol, the flooding of messages in the support increases the number of messages in the support exponentially and therefore its performance drops even worst. Therefore, for the size 2400m × 2400m, we only compare Runners and CCS1 in Figure 6.6.

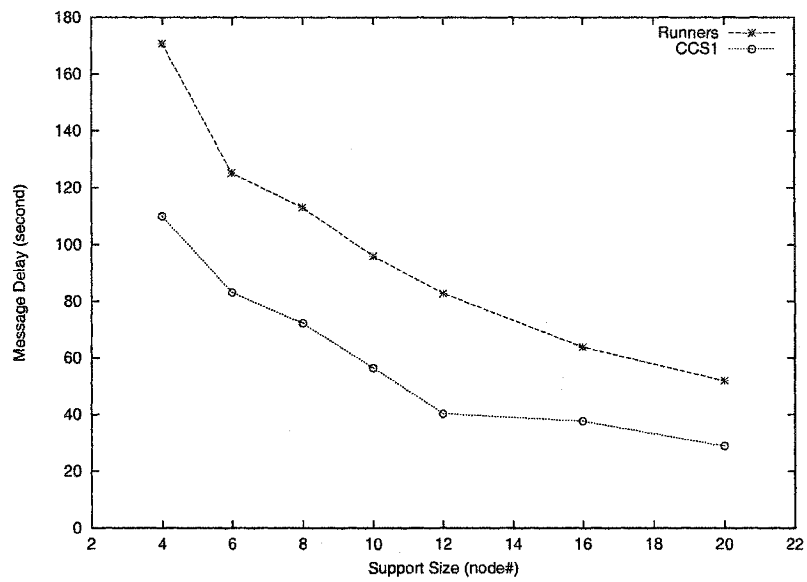

Figure 6.6: Average Message Delay vs. Support Size (Area Size 2400m × 2400m)

From Figures 6.5 and 6.6, we list the following observations.

Observation 18 *For all three protocols, the average message delay drops rather quickly when the support size is small, but after some threshold value the improvement is very little.*

Observation 19 *The protocol CCS1 performs better than snake and runners protocols.*

As expected, one of the basic protocols of the center concentrated family achieves a more efficient trade-off between network area coverage and support connectivity. We conclude that the center concentrated protocol would outperform snake and runners protocols.

Chapter 7

Conclusion and Future Directions

7.1 Conclusion

Recently, mobile ad-hoc computing has received increasing attention in the research community. Routing is a fundamental problem in any network of computing systems. Connectionless routing in ad-hoc networks is a growing area of research. Different approaches have been proposed in order to solve routing problem in disconnected ad-hoc networks.

In this thesis, we first classified and surveyed the existing connectionless routing protocols. We feel that this survey and classification can complement the existing surveys and classifications to give a wider view of the various existing routing protocols for mobile ad-hoc networks. As a result of our classification of connectionless routing protocols, we designed and presented a new compulsory routing protocol to fill the vacuum noted in our taxonomy. Next, we identified the fundamental factors responsible for the performance of routing protocols in mobile ad-hoc networks. Then, after elaborating these factors, we presented a unified framework for connectionless routing protocols. The framework brings together the ideas and concepts scattered in various protocols of this class. Finally, we proposed three connectionless semi-compulsory routing protocols derived from the framework and conducted a simulation study. The simulation results show that our protocols perform better than the popular protocols in their class.

7.2 Future Directions

There are many directions in which the work presented in this thesis can be expanded. We outline some here.

- In Section 4.1, we proposed a non-uniform-support compulsory routing protocol, TBSP. We only outlined the basic idea of this protocol, and have not conducted theoretical or experimental analysis. Certainly, further study is needed to determine the optimal numbers of nodes in each category of support nodes and the bus route schedule which is an interesting future research.
- In Section 4.2.3, we proposed four simple CCS routing protocols in which only one is studied through simulation. Further analysis (analytical and simulation) on these protocols is an interesting future work.
- Although the five semi-compulsory routing protocols, snake, runners, oscillating pairs, regional runners and CCS1, are analyzed and evaluated through simulations, more mobility patterns of support nodes, which closely reflect practical applications, need to be investigated.
- In our simulation, the mobility pattern of regular nodes is uniformly random. However, in many realistic applications, the movement of mobile nodes emerges

as group mobility, which is in non-uniform distribution. The routing performance should vary in such situations. Therefore, more variations of our protocols derived from the framework need to be investigated to cope with particular environments.

- In this thesis, we assumed that the network size is fixed and the divided scopes are also predefined for our protocols. If the network size and boundary keep changing, the adaptive scope definition and support assignment could be good ideas for further research. The corresponding policies in our framework need more sophisticated mechanisms to achieve the adaptiveness.

List of Acronyms

ABR	-	Associativity-Based Routing
AODV	-	Ad-hoc On-demand Distance Vector routing
CBRP	-	Cluster Based Routing Protocol
CCS	-	Center Concentrated Support Protocols
CEDAR	-	Core Extraction Distributed Ad Hoc Routing Algorithm
CGSR	-	Clusterhead Gateway Switch Routing
DDR	-	Distributed Dynamic Routing
DEAR	-	Device and Energy Aware Routing
DFS	-	Depth First Search
DiMAC	-	Directional MAC Protocol
DREAM	-	Distance Routing Effect Algorithm for Mobility
DSDV	-	Destination-Sequenced Distance-Vector routing
DSR	-	Dynamic Source Routing
DTC	-	Disconnected Transitive Communication
FSR	-	Fisheye State Routing
GPS	-	Global Positioning System
GSR	-	Global State Routing
HSRP	-	Hierarchical Support Routing Protocol
LANMAR	-	Landmark Ad hoc Routing
LAR	-	Location-Aided Routing
LMB	-	Landmark Routing with Mobile Backbones
MAC	-	Medium Access Control
MAP	-	Message Acceptance Policy
MANET	-	Mobile Ad-hoc NETworks
MDP	-	Message Drop Policy
MF	-	Message Ferrying scheme
MP	-	Mobility Policy
NHIP	-	Next-Hop Identification Policy
NS	-	Network Simulator
OLSR	-	Optimized Link State Routing
QoS	-	Quality of Service
RODA	-	Dynamic Routing Protocol Using Dual Paths to Support Asymmetric Links

RDI	-	Rediscovery Interval
SAP	-	Support Assignment Policy
SDP	-	Scope Definition Policy
SEP	-	Scope Establishment Policy
TBSP	-	Three-Base Support Protocol
TORA	-	Temporally Ordered Routing Algorithm
TTL	-	Time To Live
WRP	-	Wireless Routing Protocol
ZHLS	-	Zone-based Hierarchical Link State
ZRP	-	Zone Routing Protocol

Bibliography

[1] A. Avudainayagam, W.J. Lou, and Y.G. Fang, DEAR: A Device and Energy Aware Routing protocol for heterogeneous ad hoc networks, *Journal of Parallel and Distributed Computing*, 63(2), 2003, 228-236.

[2] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, A Review of Routing Protocols for Mobile Ad-hoc Networks, *Ad Hoc Networks*, 2, 2004, 1-22.

[3] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward, A Distance Routing Effect Algorithm for Mobility (DREAM), *Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking*, 1998.

[4] Bell Labs, Lucent Bell Labs' Top Innovations, *http://www.bell-labs.com/about/history/timeline.html.*

[5] J. Broch, D. A. Maltz, D. B. Johnson, Y.C. Hu, and J. Jetcheva, A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols, *Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking*, 1998.

[6] T. Camp, J. Boleng, and V. Davies, A Survey of Mobility Models for Ad Hoc Network Research, *Wireless Communication & Mobile Computing*, 2(5), 2002, 483-502.

[7] T.-W. Chen and M. Gerla, Global State Routing: A New Routing Scheme for Ad-hoc Wireless Networks, *Proceedings of IEEE Internationl Conference on Communications*, 1998, 171-175.

[8] Z.D. Chen, H.T. Kung, and D. Vlah, Ad hoc relay wireless networks over moving vehicles on highways, *Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networking & Computing*, 2001.

[9] S. Corson and J. Macker, Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, *IETF draft-ietf-manet-issues-02.txt*, 2002.

[10] X.C. Chen and A. L. Murphy, Enabling Disconnected Transitive Communication in Mobile Ad Hoc Networks, *Principles of Mobile Computing*, 2001.

[11] I. Chatzigiannakis and S. Nikoletseas, An Adaptive Compulsory Protocol for Basic Communication in Highly Changing Ad-hoc Mobile Networks, *International Parallel and Distributed Processing Symposium*, 2002.

[12] I. Chatzigiannakis, S. Nikoletseas, and P. Spirakis, An Experimental Study of Basic Communication Protocols in Ad-hoc Mobile Networks, *Proceedings of the fifth Annual Workshop on Algorithmic Engineering*, Lecture Notes in Computer Science, Vol. 2141, 2000, 159-171.

[13] I. Chatzigiannakis, S. Nikoletseas, and P. Spirakis, Distributed Communication Algorithms for Ad-hoc Mobile Networks, *Journal of Parallel Distributed Computing*, 63(1), 2003, 58-74.

[14] X. Cui, Y. Sun, and K. Alagarsamy, Simple and Efficient Semi-compulsory Routing Protocols for Mobile Ad-hoc Networks, *Proceedings of International Conference of Wireless Networks*, 2004.

[15] R. R. Choudhury and N. H. Vaidya, Ad Hoc Routing Using Directional Antennas, *Technical Report*, 2002.

[16] C.-C. Chiang, H.-K. Wu, W. Liu, and M. Gerla, Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel, *Proceedings of IEEE Singapore International Conference on Networks*, 1997, 197-211.

[17] J. Davis, A. Fagg, and B. Levine, Wearable computers as packet transport mechanisms in highly-partitioned ad-hoc networks, *International Symposium on Wearable Computing*, 2001.

[18] A. Doria, M. Uden, and D. P. Pandey, Providing Connectivity to the Saami Nomadic Community, *Proceedings of International Conference on Open Collaborative Design for Sustainable Innovation*, 2002.

[19] L. M. Feeney, A Taxonomy for Routing Protocols in Mobile Ad Hoc Networks, *SICS Technical Report T99/07*, 1999.

[20] K. Fall, A Delay-Tolerant Network Architecture for Challenged Internets, *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, 2003.

[21] D. Goyal and J. Caffery, Jr., Partitioning Avoidance in Mobile Ad Hoc Networks Using Network Survivability Concepts, *Proceedings of IEEE Symposium on Computers and Communications*, 2002.

[22] M. Grossglauser and D.N.C. Tse, Mobility Increases the Capacity of Ad Hoc Wireless Networks, *IEEE/ACM Transactions on Networking*, 10(4), 2002, 477-486.

[23] Z. J. Haas, M. R. Pearlman, and P. Samar, Zone Routing Protocol (ZRP), *IETF MANET Internet Draft draft-ietf-manet-zone-zrp-04*, 2002.

[24] A. Iwata, C.-C. Chiang, G. Pei, M. Gerla, and T.-W. Chen, Scalable Routing Strategies for Ad Hoc Wireless Networks, *IEEE Journal on Selected Areas in Communications, Special Issue on Wireless Ad Hoc Networks*, 17(8),1999, 1369-1379.

[25] M. Joa-Ng and I.-T. Lu, A Peer-to-Peer Zone-Based Two-Level Link State Routing for Mobile Ad Hoc Networks, *IEEE Journal on Selected Areas in Communications, Special Issue on Wireless Ad Hoc Networks*, 17(8), 1999, 1415-1425.

[26] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, Routing protocols for mobile ad-hoc networks - a comparative performance analysis, *Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking*, 1999.

[27] M. Jiang, J. Li, and Y.C. Tay, Cluster Based Routing Protocol (CBRP), *IETF MANET Internet Draft draft-ietf-manet-cbrp-spec-01.txt*, 1999.

[28] D. B. Johnson and D. A. Maltz, Dynamic Source Routing Protocol in Ad Hoc Wireless Networks, *Mobile Computing, edited by Tomasz Imielinski and Hank Korth*, chapter 5, pages 153–181, 1996.

[29] P. Jacquet, P. Muhlethaler, A. Qayyum, A. Lanouiti, L. Viennot, and T. Clausen, Multi-Scope Routing Protocols Flat Routing Protocols Optimized Link State Routing (OLSR), *IETF MANET Internet Draft draft-ietf-MANET-olsr-02.txt*, 2000.

[30] D. Kim, C.K. Toh, and Y. Choi, RODA: A New Dynamic Routing Protocol Using Dual Paths to Support Asymmetric Links in Mobile Ad-hoc Networks, *Proceedings of Ninth International Conference on Computer Communications and Networks*, 2000.

[31] Y.-B. Ko and N. H.Vaidya, Location-Aided Routing (LAR) in Mobile Ad Hoc Networks, *Proceedings of ACM/IEEE International Conference on Mobile Computing and Networking*,66-75, 1998.

[32] A. Lindgren, A. Doria, and O. Schelen, Probabilistic Routing Intermittently Connected Networks, *Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networking & Computing*, 2003.

[33] Q. Li and D. Rus, Communication in Disconnected Ad-hoc Networks Using Message Relay, *Journal of Parallel Distributed Computing*, 63(1),2003, 75-86.

[34] S. Murthy and J.J. Garcia-Luna-Aceves, An Efficient Routing Protocol for Wireless Networks, *ACM Mobile Networks and Applications Journal, Special Issue on Routing in Mobile Communication Networks*, 1996.

[35] M. Mauve, J. Widmer, and H. Hartenstein, A Survey on Position-Based Routing in Mobile Ad-Hoc Networks, *IEEE Network Magazine*, 15(6), 2001, 30-39.

[36] N. Nikaein, H. Labiod, and C. Bonnet, Distributed Dynamic Routing Algorithm (DDR) for Mobile Ad Hoc Networks, *Proceedings of the First Annual Workshop on Mobile Ad Hoc Networking & Computing*, 2000.

[37] A. Nasipuri, J. Mandava, H. Manchala, and R. E. Hiromoto, On-Demand Routing Using Directional Antennas in Mobile Ad Hoc Networks, *Proceedings of International Conference on Computer Communications and Networks*, 2000.

[38] The VINT project, The ns Manual, *http://www.isi.edu/nsnam/ns/doc/*.

[39] C. E. Perkins and P. Bhagwat, Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, 1994, 234-244.

[40] V. Park and M.S. Corson, A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks, *Proceedings of Conference of the IEEE Communications Society*, 1997.

[41] G.Y. Pei, M. Gerla, and X.Y. Hong, LANMAR: Landmark Routing Protocol for Large Scale Wireless Ad Hoc Networks with Group Mobility, *Proceedings of IEEE/ACM Workshop on Mobile Ad Hoc Networking & Computing*, 2000.

[42] C.E. Perkins and E.M. Royer, Ad-hoc On-Demand Distance Vector Routing, *Proceedings of Second IEEE Workshop on Mobile Computing Systems and Applications*, 1999, 90-100.

[43] S. Ramanathan and M. Steenstrup, A Survey of Routing Techniques for Mobile Communications Networks, *Mobile Networks and Applications*, 1(2), 1996, 89-104.

[44] E. M. Royer and C-K. Toh, A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks, *IEEE Personal Communications*, 6(2), 1999, 46-55.

[45] R. Shah, Delivering Messages in Disconnected Mobile Ad Hoc Networks, *MS Thesis*, UBC, Canada, 2004.

[46] R. Shah and N. C. Hutchinson, Delivering Messages in Disconnected Mobile Ad Hoc Networks, *International Conference on Ad Hoc Networks and Wireless*, LNCS 2865, 72-83, 2003.

[47] Y. Sun and K. Alagarsamy, A Taxonomy of Connectionless Routing Protocols for Mobile Ad-hoc Networks, *Submitted for publication*, 2004.

[48] Y. Sun and K. Alagarsamy, A Generalized Framework for Connectionless Routing Protocols in Mobile Ad-hoc Networks, *Under preparation*, 2004.

[49] Y. Sun, X. Cui, and K. Alagarsamy, Efficient Semi-compulsory Routing Protocols for Mobile Ad-hoc Networks, *Proceedings of IASTED Conference on Wireless Networks and Emerging Technologies*, 2004.

[50] R. Sivakumar, P. Sinha, and V. Bharghavan, CEDAR: a Core-Extraction Distributed Ad hoc Routing algorithm, *IEEE Journal on Selected Areas in Communications, Special Issue on Wireless Ad Hoc Networks*, 17(8), 1999.

[51] C.-K. Toh, A Novel Distributed Routing Protocol to Support Ad-Hoc Mobile Computing, *Proceedings of IEEE International Phoenix Conference on Computers and Communications*, 1996.

[52] A. Vahdat and D. Becker, Epidemic routing for partially connected ad hoc networks, *Technical Report CS-200006*, Duke University, 2000.

[53] K.X. Xu, X.Y. Hong, and M. Gerla, Landmark routing in ad hoc networks with mobile backbones, *Journal of Parallel and Distributed Computing*, 63(2), 2003, 110-122.

[54] W.R. Zhao and M. Ammar, Message Ferrying: Proactive Routing in Highly-Partitioned Wireless Ad Hoc Networks, *Proceedings of International Workshop on Future Trends in Distributed Computing Systems*, 2003.

[55] Z.J. Haas, J. Deng, B. Liang, P. Papadimitratos, and S. Sajama, Wireless Ad Hoc Networks, *http://www.ece.cornell.edu/~haas/wnl/html*, 2001.

[56] X. Zou, B. Ramamurthy, and S. Magliveras, Routing Techniques for Wireless Ad Hoc Networks-Classification and Comparison, *SCI*, 2002.