



uOttawa

L'Université canadienne
Canada's university

**FACULTÉ DES ÉTUDES SUPÉRIEURES
ET POSTDOCTORALES**



**FACULTY OF GRADUATE AND
POSTDOCTORAL STUDIES**

Jingzhe Du

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

M.C.S.

GRADE / DEGREE

School of Information Technology and Engineering

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Geometric Routing Protocol in Disruption Tolerant Network

TITRE DE LA THÈSE / TITLE OF THESIS

Amiya Nayak

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

Paola Flocchini

Michel Barbeau

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

Geometric Routing Protocol in Disruption Tolerant Network

by

Jingzhe Du

A thesis submitted to
the Faculty of Graduate and Postdoctoral Studies
in partial fulfilment of
the requirements for the degree of
Master of Computer Science

Co-Supervisor: **Dr. Amiya Nayak**

Co-Supervisor: **Dr. Evangelos Kranakis**

Ottawa-Carleton Institute for Computer Science
Ottawa, Ontario, Canada

©Copyright
2008, Jingzhe Du



Library and
Archives Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-48450-0
Our file *Notre référence*
ISBN: 978-0-494-48450-0

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

Abstract

We describe Geometric Localized Routing (GLR) in Disruption (delay) tolerant networks (DTNs). Although DTNs do not guarantee the connectivity of the network all the time, geometric location information still could be used to make routing decisions in a store and forward way. When a message reaches a region where greedy forwarding is impossible for some time, geometric planar spanners, especially local Delaunay triangulation still could be used to provide a good routing graph with constant stretch factor and shorter paths during communication. Existing DTN routing protocols either focus on the improvement of message delivery latency and delivery ratio in an impractical way, or without achieving desirable delivery attributes. In this work, we design local distributed solutions to extract spanning trees from Delaunay triangulation (DT) in the direction from source to destination. Our protocol resorts to flooding packets along the trees and with high probability packets are delivered with low delay. The goals of Geometric Localized Routing are: 1) fast delivery with intelligence, 2) better storage utilization. 3) guaranteed delivery using face routing. Through the implementation in the NS-2 simulator, we have shown that the proposed routing protocol achieves high delivery ratio with satisfied attributes.

Acknowledgments

It is my great fortune to have pursued my Masters studies under the guidance of my supervisors, Dr. Amiya Nayak and Dr. Evangelos Kranakis. They introduced me to the subject area of this thesis, and guided me in every step with their profound knowledge, insights and wisdom. I express my sincere gratitude to them for their clever ideas, invaluable guidance, enthusiastic encouragement and strong support. Their high standards for research and genuine interests for intellectual challenges have stimulated me and will remain a source of inspiration to me.

I also feel deep gratitude to my family. I thank my wife Shouyan Zhang for her support for my Masters studies and the tremendous help in keeping the family running. I thank my sister, Zhaohui Du, for her suggestions on my studies. I cannot imagine how I can make it without the love, support and understanding of my father, Xuejian Du, and my mother, Hongrong Zhang. I dedicate this thesis to them.

Table of Acronyms

DGF Delivery Guaranteed Face Routing.

DT Delaunay triangulation.

DTLSR Delay Tolerant Routing for Developing Regions.

DTN Disruption (delay) tolerant network.

DTNRG Delay Tolerant Network Research Group.

FIMF Ferry-Initiated Message Ferrying.

GFG Greedy face greedy.

GG Gabriel graph.

GLR Geometric Localized Routing.

GPRS General Packet Radio Service.

GPS Global Positioning System.

GPSR Greedy perimeter stateless routing.

GPVFR Greedy path vector face routing.

IMEP Internet MANET Encapsulation Protocol.

IPNRG InterPlaNetary Internet Research Group.

k-LDT k-Local Delaunay triangulation.

k-LMWST k-Local minimum weight spanning tree.

LDT Local Delaunay triangulation.

LMWST Local minimum weight spanning tree.

MANET Mobile Ad Hoc Network.

MaxDLTR Max Distance Left to Right.

MaxDSTD Max distance source to destination.

MBR Model based routing.

MEED Minimum estimated expected delay.

MF Message Ferrying.

MidDLTR Mid Distance Left to Right.

MidDSTD Mid distance source to destination.

MinDLTR Min Distance Left to Right.

MinDSTD Min distance source to destination.

MST Minimum spanning tree.

NIMF Node-Initiated Message Ferrying.

PRoPHET Probabilistic Routing Protocol using History of Encounters and Transitivity.

RDG Restricted Delaunay Graph.

RNG Relative neighbor graph.

UDel Unit Delaunay triangulation.

UDG Unit Disk Graph.

Contents

Table of Acronyms	iii
1 Introduction	1
1.1 Overview	1
1.1.1 Mobile Ad Hoc Network (MANET)	1
1.1.2 Localized Geometric Spanner in MANET Routing	1
1.1.3 Disruption (Delay) Tolerant Network	2
1.2 Motivation	3
1.3 Previous Related Results	5
1.4 Objectives	8
1.5 New Methodology	8
1.6 Outline of Thesis	10
2 Localized Geometric Spanners	12
2.1 Introduction	12
2.1.1 Definitions	13
2.1.2 Geometric Spanner	15
2.2 Several Kinds of Geometric Spanners	17
2.2.1 Delaunay Triangulation	17

2.2.2	Minimum Spanning Tree	18
2.3	Localized Algorithms	18
2.3.1	Reasons to Use Local Algorithm	18
2.3.2	Localized Delaunay Triangulations	19
2.3.3	Localized Minimum Spanning Trees	21
2.4	Applications of Localized Geometric Spanners	25
2.4.1	Face Routing	25
2.4.2	Greedy Face Greedy	26
2.4.3	Other Face Routing Protocols and Pitfalls	27
3	Overview of DTN and Existing Routing Protocols	31
3.1	Introduction	31
3.1.1	Some Main DTN Applications	31
3.1.2	DTN Research Groups	33
3.1.3	Bundle Protocol	34
3.1.4	Characteristics of Disruption (Delay) Tolerant Network	34
3.2	Overview of Major DTN Routing Protocols	35
3.2.1	Epidemic Routing	35
3.2.2	Message Ferrying	38
3.2.3	Model Based Routing (MBR)	39
3.2.4	Interrogation-Based Relay Routing	40
3.2.5	MaxProp	40
3.2.6	MEED	41

3.2.7	DTLSR	41
3.2.8	Space and Time Routing	41
3.2.9	Spray and Wait	42
3.2.10	Utility Based Routing	43
3.2.11	Knowledge of Motion Profiles Approach	43
3.2.12	Erasur Coding Based Routing	44
3.2.13	PRoPHET	45
3.3	Summary of DTN Routing Protocols	46
3.4	Classification of DTN Routing Protocols	49
4	Proposed Solutions	51
4.1	Goals	51
4.2	System Design	52
4.2.1	Greedy Geometric Routing with Intelligently Controlled Flooding .	53
4.2.2	Delivery Guaranteed Face (DGF) Routing	60
4.2.3	Face Routing Planar Graph Selection	61
5	Implementation	63
5.1	Implementation Details	63
5.1.1	NS-2 and the Modification	63
5.1.2	Compute Delaunay Triangulation	65
5.1.3	Localized Delaunay Triangulation	66
5.1.4	MaxDSTD, MinDSTD and MidDSTD	66
5.1.5	Message Structure	67

5.1.6	Information Exchange Flow	68
5.1.7	Location Diffusion	70
5.1.8	Methods of Extracting Three Trees	71
5.1.9	Custody Transfer	73
5.1.10	Face Routing Using Planar Graph	74
6	Experimentation Results/Analysis	75
6.1	Simulation Environment	75
6.2	Simulation Results and Analysis	77
6.2.1	Store and Forward Mechanism	78
6.2.2	Location update	79
6.2.3	Delivery Latency Comparison	81
6.2.4	Delivery Ratio Comparison	83
6.2.5	Delivery with Custody Transfer	84
6.2.6	Hop Count Comparison	85
6.2.7	Storage Requirements	86
6.2.8	Face Routing Delay and Hop Comparison	87
6.2.9	Face Routing Energy Comparison	87
6.2.10	The Need for Planar Accuracy	89
6.3	Delay Tolerance Analysis	89
6.3.1	Previous Methods to Alleviate Delay	89
6.3.2	Our Proposal to Reduce Delay	90

7 Conclusion & Future Work	92
7.1 Future Directions	92
7.2 Conclusions	94
Bibliography	96

List of Figures

2.1	Relative neighbor graph (RNG)	15
2.2	RNG of random graph (radius 0.2, 50 nodes in unit square disk)	15
2.3	Gabriel graph (GG)	16
2.4	GG of random graph (radius 0.2, 50 nodes in unit square disk)	16
2.5	RNG is subset of GG	17
2.6	LMWST is a subset of RNG	23
2.7	LMWST is a superset of MST	24
2.8	Face routing definition	25
2.9	Illustration of face routing failure without planar	26
2.10	GFG routing	27
2.11	GFG routing loop	27
2.12	GPSR right hand rule	28
2.13	GPSR routing loop	29
2.14	GPVFR routing logic	30
2.15	GPVFR routing loop	30
3.1	Main organizations working on DTN topic	33

3.2	Bundle layer	34
3.3	Illustration of time evolving behavior of epidemic transmission	37
3.4	Epidemic routing	38
3.5	Local model and user profile example [54]	40
4.1	Topology of 50 nodes with radius 250,200,150,100 in 1000×1000 area.	55
4.2	Topology of 50 nodes with radius 10 in 1500×300 area	56
4.3	DGF routing	60
4.4	DGF routing: handling duplicate messages	61
4.5	Face routing using LMWST	62
5.1	Next hop selection	67
5.2	Distance to destination is smaller	72
5.3	Face routing demonstration	74
6.1	Delivery latency comparison (different route check interval)	79
6.2	Delivery latency comparison (50m radius)	82
6.3	Delivery latency comparison (100m radius)	83
6.4	Delivery latency comparison (various radius)	84
6.5	Delivery ratio comparison (50m radius)	85
6.6	The need for planar accuracy	89

List of Tables

3.1	Summary of Routing Mechanisms	47
6.1	Parameters of the simulations.	76
6.2	Message delivery results under location information availability	80
6.3	Message delivery ratio comparison (50m)	85
6.4	Hop counts (1980 messages, for geometric routing, 3 copies for 50m/100m and 1 copy for 150m/200m/250m)	86
6.5	Storage requirement, different message numbers (50m, 3copies)	86
6.6	Storage requirement, different radius (1980 messages, 3 copies for 50m/100m and 1 copy for 150m/200m/250m)	86
6.7	Message delivery comparison (250m radius, 150 messages, * same condition but not optimal)	87
6.8	Message delivery comparison (250m radius, 300 messages, * same condition but not optimal)	87
6.9	Message delivery comparison (250m radius, 600 messages, * same condition but not optimal)	87

6.10 Path comparison (250m radius, 600 messages, d is the distance between
two nodes in message transmission path) 88

Chapter 1

Introduction

1.1 Overview

1.1.1 Mobile Ad Hoc Network (MANET)

Mobile Ad Hoc Network (MANET) is a collection of autonomous mobile nodes connected by wireless channels without any pre-existing network infrastructure. Typically, some of these mobile devices are part of the network only while they can communicate with the rest of the network. Existing ad hoc routing protocols usually assume that there is always a connected path from message source to destination. In situations where network partitions exist, these routing protocols drop the message if a path could not be found and thus perform insufficiently in terms of message delivery.

1.1.2 Localized Geometric Spanner in MANET Routing

Geometric spanners could be constructed when global network location information is given for a static wireless network. Messages from source node to destination node could be properly delivered using these spanners (e.g., Delaunay triangulation or minimum spanning tree) by a specially designed routing algorithm, such as face routing [8].

With Global Positioning System (GPS), some mobile nodes could get the location information about themselves. These nodes could exchange this information with their neighbors when they come within radio communication range with each other. Although it is very difficult for a node in MANET to know the global network location information and connectivity, it still could construct local geometric spanners with only information provided by its neighboring nodes. The localized geometric spanners are expected to achieve global connectivity with desired spanning characteristics.

1.1.3 Disruption (Delay) Tolerant Network

While most localized geometric spanners working with MANET are highly connected, there are situations where network is disconnected at some places or times which will cause intermittent network communications. DTN is a specially designed network in which intermittent communications and anomalies have the least adverse impact on the network. There are several kinds of DTNs. Internet with IPv4 is homogeneous in itself because the IP layer is the same within the network. Other networks like General Packet Radio Service (GPRS) and Sensor Network differ from Internet by not using the IP protocol. Communication delays are inevitable between Internet and other networks because of the necessary network topology change. The high layer network proposed by Delay Tolerant Network Research Group (DTNRG) [1] to support communication between different networks is one type of DTN.

DTN could be considered as an overlay architecture over the existing network architecture. The goal of DTN is to ensure message delivery in mobile or extreme environments

by using store-and-forward approach. Messages (or packets), also called bundles, are sent from node to node by the incorporation of DTN layer even without continuous network connectivity.

1.2 Motivation

As current MANET routing protocols usually assume that networks remain connected, they hardly work well in mobile networks with disruptions. Therefore, the study of routing protocol study specially for DTNs with mobile nodes is needed. This is one important aspect of DTN research which is also the focus of our study. There are increased applications of DTNs in recent years, including military communications [36], inter-planetary networks [10], wildlife tracking [28] and intermittent Internet connection in under-developed countries (areas) [39], to name a few.

Various routing protocols have been proposed since the emergence of DTNs. Zhang [54] in his survey gives a detailed description of various routing protocols and categorizes them as deterministic and stochastic. Routing protocols in the first category determines the end-to-end path before packets are transmitted. Inside the second routing category (stochastic), there is one well known protocol called Epidemic routing [51]. By using this protocol, when packet arrives at a node, it is forwarded to all the neighbors of this node. This protocol shows high delivery rate, but it is impractical in terms of bandwidth, storage and energy usage. Some other protocols inside the second category make use of patterns of delivery probability and node movements to improve routing efficiency. Controlling of node movements and fault-tolerant coding techniques are also being exploited in the

stochastic case. However, neither one achieves small delivery latency, high delivery ratio and better resource utilization.

Although in DTN, the whole network may not be connected and the communication path between source and destination node may not exist, it is possible that parts of the network nodes are connected together. Whenever source and destination node are connected inside one part, the message is delivered quickly. If not, message should be sent to a node that lies in the same connected part with source but is most likely to find a path to the destination in the near future, e.g., a node that is closest to the destination. Network topology of partly connected network should be used to improve the efficiency of the message transmission.

With nodes moving, it is impossible for nodes to get the whole network topology information in most situations. Thus, any algorithm which uses global topology information to achieve global computation should be restricted to distance k neighborhood of a node in DTN. To construct planar subgraph, k needs to be greater than 1. As in some DTN applications, node could get its location information by GPS or localization algorithms. It could make geographical routing decisions using the location information of its own and its distance k neighbors. In geographical routing, packets are forwarded to those nodes which are closer to the destination. Unnecessary packets in other directions are avoided as a result. However, geographical routing suffers from the local minimum phenomenon, in which a packet gets stuck at a node that does not have a closer neighbor to the destination even without network partition. Maintaining planar subgraph of the underlying connectivity could deal with this problem, i.e., packets could be routed around a face of

the planar graph to get out of the local minimum, which was proposed by Bose *et al.* [8] and independently by Karp and Kung [29]. In DTNs, temporarily and partly connected parts of the whole network could still maintain individual planar subgraph to facilitate the routing process, and this planar subgraph should be constructed using only distance k ($k > 1$) neighborhood information.

1.3 Previous Related Results

The simplest solution to DTN routing is Epidemic routing [51]. The core idea of this protocol is that a node forwards a packet to every next-hop because it does not know which route will work, and if all nodes do this for every packet, high delivery ratio will be guaranteed. Many other new ideas have also been proposed to improve the efficiency of routing in recent years. Some DTN routing protocols use destination node's location information and forward messages in the direction of the receiver. Spray Routing [50] sends packets to the vicinity of the receiver's last known location and with high probability packets could be delivered. Shen *et al.* [45] designed interrogation based relay routing for ad hoc satellite networks where participating satellites exchange network topology information, i.e., orbit (location) information. In [5], the use of geographic information of the city increases the probability of delivery and reduces relay hop.

Routing protocols in MANET using location information were also heavily studied. In [12], Chavez *et al.* proposed one way to construct local planar spanner with irregular transmission ranges. For some parameter r ($0 < r \leq 1$), the transmission range of a node includes the entire disk around the node of radius at least r , and it does not include any

node at distance more than one. They construct a local minimum spanning tree with maximum degree of $3 + \frac{6}{\pi r} + \frac{r+1}{r^2}$ when $0 < r < 1$ (and at most 5 when $r=1$). For $k \geq 2$, the total weight of the spanner thus obtained is at most $\frac{kr+1}{kr-1}$ times the sum of the Euclidean lengths of the edges of a minimum weight Euclidean spanning tree. The resulting planar graph is a t -spanner which is constructed using information of nodes' distance 2 neighborhood. In [33], Li *et al.* proposed another algorithm that constructs a localized minimum spanning tree which has the maximum degree of 6. This local minimum spanning tree is built upon one-hop away neighbor information by applying Prim's algorithm [40] locally and it could still keep global connectivity. Locally constructed minimum spanning trees could work as the underlying routing graph for geographical routing protocols. Since minimum spanning trees have the characteristic of using shorter paths, it is expected that energy consumption could be reduced by using shorter range radios during communication. Geographical routing could use these spanners to deliver messages.

Under unit disk graph (UDG) model, Delaunay triangulation construction should only keep short edges (≤ 1). Local DT with only short edges should, have good stretch factor and should be constructed locally. Later in this thesis we will apply the localized DT in DTN. No similar work has been done in DTNs so far, but we could take local DT algorithm in MANET as a reference. Li *et al.* [34] provided one way to construct the localized Delaunay graph. They first define the $UDel(V)$ to be the graph obtained by removing all the edges that are longer than one unit from $Del(V)$ (Delaunay triangulation graph of set V). After that, they give a modified local algorithm to construct the localized $Del(V)$, the $LDel(V)$. Their algorithm further makes $LDel(V)$ a planar graph. The final planar

graph is proved to be a t -spanner of $UDG(V)$ with $t = \frac{4\sqrt{3}}{9}\pi$. Gao *et al.* [21] proposed another localized Delaunay graph, the Restricted Delaunay Graph (RDG). By first forming clusters on the whole node sets, they construct RDG on clusterheads and gateway nodes. The RDG is constructed and maintained in a distributed way with constant Euclidean and topological distance stretch factors.

After the localized DT has been constructed in connected parts of the DTN, geographical routing could be used to forward packets in these connected parts of the network along the direction from source to destination. There are several existing geographical routing protocols. Kranakis *et al.* [31] proposed Compass Routing algorithm on geometric networks. In Compass Routing, given t as the destination node, source node u finds the next relay node v such that the angle $\angle vut$ is the smallest. Also in [31], Kranakis *et al.* proposed face routing algorithm, called Compass Routing II, which guarantees delivery. Greedy Routing [8] is another geographical routing protocol. It finds the next relay node v such that the distance $|vt|$ is the smallest. Karp *et al.* [29] also presented a geographical routing algorithm, called Greedy perimeter stateless routing (GPSR). Compass Routing II and GPSR route packet from source node to destination node in a greedy way.

In [22], Georgiou *et al.* proposed an algorithm to extract three spanning trees from localized Delaunay triangulation graph. The tree extraction is based on distance. In the Max Distance Left to Right (MaxDLTR) tree, each node is connected to a max distance neighbor to its right while in the Min Distance Left to Right (MinDLTR) and Mid Distance Left to Right (MidDLTR) tree, each node is connected to a Min and Mid (other than Max and Min if it has one) distance neighbor to its right respectively. They prove that

for $r_n \geq \sqrt{\frac{9 \ln n}{n\pi}}$, the trees MinDLTR, MaxDLTR and MidDLTR have an edge in common is at most $1/n$, asymptotically in n . The use of these three trees can be useful for fault tolerance.

1.4 Objectives

The main objective of this thesis is to investigate the existing DTN routing schemes and develop a more efficient mechanism, taking practical issues into consideration.

With DTN applications being widely used, the need of efficient routing is increasing. Current routing protocols either flood without control, or assume some prior knowledge of the whole network. The underlying routing graph is not often properly used to facilitate routing. Therefore, our aim is to develop a controlled-flooding scheme that can route messages quickly using a subgraph which has a shortest path that is t times the shortest path of the original graph.

1.5 New Methodology

As DTNs are more dynamic than the traditional MANETs, we propose adaptive routing mechanism that could intelligently route packets to the destination according to various network topologies. If the node communication range is short and the number of nodes is small, there is little opportunity that the network could be connected. More copies of the same message should be transmitted in the network to increase delivery opportunity and reduce delay. While in a network with high node density and long radius, single message copy is enough for a fast delivery. Multiple message copies approach

should be avoided in such a network because more copies could cause greater contentions and will slow down the message transmission as a result. The efficiency of the routing protocols should be improved by using such intelligent methods.

We also use geometric spanners which could be constructed locally in various connected parts of the whole network. A better routing graph is constructed which has good stretch factor if the whole network is connected, or even it is only partly connected under specified antenna radius. Multiple distinct routing paths (trees) are extracted from the geometric graph. The use of different routing paths increases the chance of delivery and decrease transmission time. Other factors, including the delivery rate and energy consumption are also taken into consideration during our research in this field.

Currently, there are several things that are considered in the development of our routing protocol scheme. First, our routing protocol should work in an efficient way when an end to end path really exists. Second, when the above method does not work, nodes should be able to deliver the packets to nodes that are moving towards the destination nodes and thus improve the delivery ratio and reduce delivery delay. Thirdly, when no actions could be taken, nodes turn into a store state, waiting further opportunity to appear. Lastly, when there is no opportunity for a greedy forwarding, nodes could use face routing to deliver message if there is a connection between the node which holds the message and the destination.

We are assuming that at anytime, source node knows the exact destination node position, just like the way used in GPSR [29]. As the destination node moves randomly and its position may change during message transmission process, destination location

update technique is, therefore, developed to counter this problem.

1.6 Outline of Thesis

In this thesis, we first define the concept of k-Local algorithm. By applying k-local minimum weight spanning tree and k-local Delaunay triangulation algorithms, k-LMWST and K-LDT are obtained. The properties of these two local geometric graphs are proved for further use in the routing process.

After that, we propose a novel Geometric-based Localized Routing (GLR) scheme. It uses the localized Delaunay triangulation technique to construct localized t-spanner. Spanning trees are then extracted from this geometric spanner, and message is transmitted along these different trees to reduce delay in the network with intermittent connections. A node turns into store state when the message could not be delivered because of network disruption, and the delivery process is restarted after specified delay. We also give out a novel face routing technique which guarantees delivery by slightly increasing the number of copies of the same message in transit. Face routing characteristics under both the local minimum weight spanning tree and local Delaunay triangulation are also evaluated.

Simulation results show that the proposed routing protocol is fast. It uses only limited storage and still achieves high delivery ratio. In the simulation, the proposed face routing technique achieves 100% delivery ratio when source and destination connection exists.

There are seven chapters in the thesis. The rest of the thesis is organized as follows. Chapter 2 gives out the description of local geometric spanners. Chapter 3 reviews some of the existing DTN routing protocols. Chapter 4 elaborates on our proposed solutions.

Following that, Chapter 5 presents our implementation details. Chapter 6 talks about the details of experiments results and analysis. Chapter 7 concludes by summarizing our findings and points out the possible future directions that we could work on.

Chapter 2

Localized Geometric Spanners

2.1 Introduction

In the disruption tolerant network with mobile nodes, it is difficult for a node to get the entire network topology. Global communication should be attained by using only local information. Any algorithm for global computation should be restricted to local computation in DTN. The final routing graph should be able to achieve desired global communication and spanning characteristics. At the beginning of this section, we first review some definitions and geometric spanners. Following that, k-local algorithm is defined, which is used in the following chapters. Based on the definition, the properties of the k-local Delaunay triangulation and k-local minimum weight spanning tree are proved. As face routing on geometric spanners is part of our routing logic, various previous face routing protocols are discussed.

2.1.1 Definitions

2.1.1.1 Unit Disk Graph (UDG)

To simplify the research of underlying network graph, Unit Disk Graph (UDG) model was introduced. UDG is an abstract model which could be defined as follows:

- It is a graph $G(V,E)$ with V the set of vertices and E the set of its edges.
- V : Vertices are the mobile nodes.
- E : Edges between vertices represent connectivity, i.e., two vertices u, v are connected if and only if their distance is at most 1.

Unit disk graphs have been used in computer science to model the topology of ad hoc wireless communication networks. It is assumed that all nodes are homogeneous and equipped with omnidirectional antennas to simplify the analysis of the network.

2.1.1.2 Planar Graph

A graph in which no two edges cross is called a planar. A planar graph is necessary for efficient face routing. Face routing solves the local minimum phenomenon, in which a packet gets stuck at a node that does not have a closer neighbor to the destination by using either the left hand rule or right hand rule.

2.1.1.3 Local Algorithm

Take any algorithm A such that on input node set V of UDG G , the algorithm outputs a graph $A(V)$ of V . k -Local Algorithm could be defined as follows:

Algorithm k -Local.

Input: Node set V of UDG G ;

Output: G' (subgraph of G with the same vertices);

- For any node u ,
 - u collects its distance k neighborhood $N_k(u)$.
 - u constructs $A(N_k(u))$.
- A link $\{u,v\}$ is accepted in G' if it is in $A(N_k(u))$ and $A(N_k(w))$, $\forall w \in N_1(u), u \in N_k(w)$ and $v \in N_k(w)$.

Here the last item is used instead of the definition “A link $\{u,v\}$ is accepted in G' if it is in both $A(N_k(u))$ and $A(N_k(v))$.” which was defined by Kranakis in his “Wireless Networking and Mobile Computing” lecture notes [2]. The definition modification is applied to guarantee that G' is a planar. Planar graph is necessary for the proposed face routing algorithm. It is easy to see that one node needs distance $k+1$ neighborhood information to apply k -Local Algorithm.

2.1.1.4 t-spanner

To reduce the complexity of original graph, simpler subgraphs are used in practice. These subgraphs are expected to be t -spanners. Given a graph H , the spanning sub-graph G of H is a t -spanner if the length of the shortest path connecting any two points in G is no more than t times the length of the shortest path connecting the two points in H . The constant ratio is called stretch factor. Delaunay triangulation is such a spanner.

In situations where there are several paths from source to destination, shorter path

always means faster and less energy consumption. So it is important for the planar subgraph to have a shortest path whose length is less than t times the shortest path length in the original graph with the same endpoints.

2.1.2 Geometric Spanner

2.1.2.1 Relative Neighbor Graph and Gabriel Graph

Relative neighbor graph (RNG) of a graph $G = (V, E)$, denoted by $\text{RNG}(G)$, is the set of all edges $uv \in E$ such that there is no vertex or point w where $uw \in E$, $wv \in E$ and $|uw| < |uv|$ and $|wv| < |uv|$. This could be shown in Figure 2.1 and the RNG of a random graph is shown in Figure 2.2.

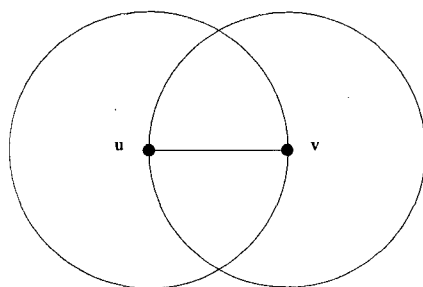


Figure 2.1: Relative neighbor graph (RNG)

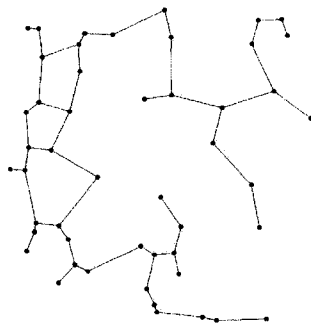


Figure 2.2: RNG of random graph (radius 0.2, 50 nodes in unit square disk)

Gabriel graph (GG) is a certain graph that connects a set of points in the Euclidean

plane. Two points u and v are connected by an edge in the Gabriel graph whenever the disk having line segment uv as its diameter contains no other points from the given point set. Figure 2.3 shows the GG edge. The GG of a random graph is shown in Figure 2.4.

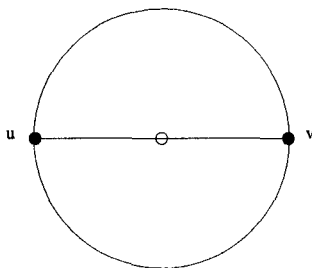


Figure 2.3: Gabriel graph (GG)

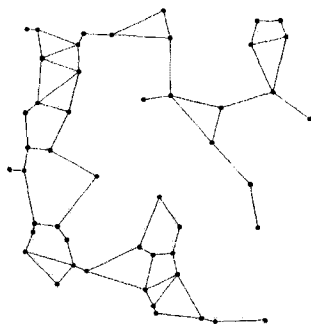


Figure 2.4: GG of random graph (radius 0.2, 50 nodes in unit square disk)

Both RNG and GG are planar graphs. RNG is a subset of GG (from Figure 2.5). However, Bose *et al.* [7] proved that the length stretch factors of these two graphs are $O(n)$ and $O(\sqrt{n})$ respectively. So they are not desirable in terms of the energy efficiency. Both RNG and GG are used to prove that our proposed k -local spanners have desired properties.

2.1.2.2 Ways of Obtaining Geometric Information

Nodes can obtain their location information either by GPS or localization algorithm[42].

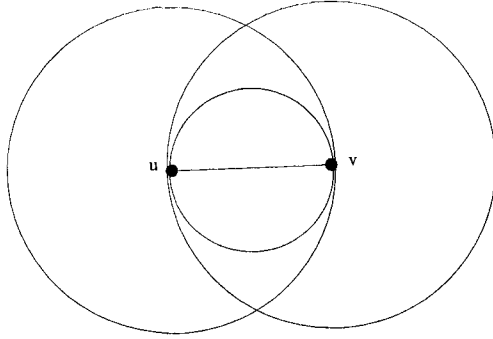


Figure 2.5: RNG is subset of GG

2.2 Several Kinds of Geometric Spanners

2.2.1 Delaunay Triangulation

Definition Delaunay triangulation (DT) is considered to be a spanner with constant stretch factor. Consider a triangulation τ of the point set P . For any triple (p, q, r) of points, let $S_{p,q,r}$ be the circle formed by the points p, q, r . τ is the Delaunay Triangulation of the point set P if for triple (p, q, r) of points, $S_{p,q,r} \cap P = \emptyset$. The Voronoi region of a point r is the collection of two dimensional points such that every point in this region is closer to r than to any other point of point set P . The union of all voronoi regions is the Voronoi diagram for P . DT is the dual graph of the Voronoi diagram.

UDel After construction of Delaunay triangulation over point set P , Unit Delaunay triangulation (UDel) could be extracted by removing edges that are longer than 1. UDel is connected if the original graph is connected.

Empty Circle Property Two points p_i and p_j are connected by an edge in the Delaunay triangulation, if and only if there is an empty circle passing through p_i and p_j .

(The proof could be easily obtained from the circumcircle property. If there is an empty circumcircle passing through p_i and p_j , then the center c of this circle is on the edge of the Voronoi diagram between p_i and p_j and it is equidistant from these points. So there is no other closer point inside the circle.)

Stretch Factor Dobkin *et al.* [15] proved that the shortest path of the Delaunay triangulation graph of a set P of N points in the plane is bounded by a constant factor c ($\leq ((1 + \sqrt{5})/2)\pi \approx 5.08$). Later Keil [30] improved this ratio to $\frac{2\pi}{3\cos(\frac{\pi}{6})} \approx 2.42$. Since there may exist partitions in DTNs, the DT could only be constructed in several different connected parts. The DT algorithm should work in a distributed way. Each node needs only consider all the nodes at distance at most k hops from it because of the dynamics of the DTNs.

2.2.2 Minimum Spanning Tree

A spanning tree of a network is a subnetwork that is a tree and contains all the network nodes. If the edges of the tree are weighted (e.g., representing power consumption or communication delay), a Minimum spanning tree (MST) is one with the minimum sum of edge weights. Two standard algorithms are Prim's algorithm and Kruskal's algorithm.

2.3 Localized Algorithms

2.3.1 Reasons to Use Local Algorithm

Although it is desirable to use global algorithm in the construction of underlying routing spanners, the cost of obtaining global information for every node may be very

high, or even impossible, especially in highly dynamic mobile ad hoc networks. By using only distance k ($k \geq 2$) neighboring nodes information, global algorithms could be modified to apply on every node's distance k neighborhood only so that every node could make routing decisions locally by collecting distance k neighboring nodes information around it. Properly designed local algorithms should have similar properties of the global algorithm, such as connectivity, stretch factor, etc.

2.3.2 Localized Delaunay Triangulations

2.3.2.1 LDel(v)

In [34], Li *et al.* defined k -localized Delaunay triangle and k -localized Delaunay graph as the following in their own way.

A triangle Δuvw satisfies k -localized Delaunay property if the interior of disk (u,v,w) does not contain any node of S that is a k -neighbor of u , v , or w ; and all edges of the triangle Δuvw have length no more than one unit. Triangle Δuvw is called a k -localized Delaunay triangle.

The k -localized Delaunay graph over a node set S , denoted by $LDel^{(k)}(S)$, has exactly all Gabriel edges and edges of all k -localized Delaunay triangles.

2.3.2.2 RDG

Restricted Delaunay Graph (RDG) [21] is another localized Delaunay triangulation defined by Gao *et al.* which has the following properties:

1. It is planar.
2. It is t -spanner.

3. The graph could be maintained by exchanging information with only neighboring nodes.

The algorithm works in the following way. First, nodes construct clusters according to [20]. Only clusterheads and gateway nodes are considered in the construction of the local Delaunay triangulation. Each node (clusterhead or gateway node) acquires its one-hop neighbors location and compute the corresponding Delaunay triangulation. Nodes will then make the graph planar by deleting the edges that are inconsistent. An edge of local graph will be kept only when all the neighboring nodes which see the two end nodes of this edge have this edge as part of their local graph. RDG is 1-Local DT according to our definition in Section 2.3.2.3.

2.3.2.3 LDT Definition and Its Properties

The Local Delaunay triangulation (LDT) could be obtained by using k-local algorithm (2.1.1.3, with $k=1,2,\dots,d$.) of DT construction. LDT is used in the proposed solution and has the following properties:

1. It is connected if the original graph is connected
2. It is planar
3. It is t-spanner, $t = \frac{4\sqrt{3}}{9}\pi$

Connected If an edge uv is in $UDel$, there must exist an empty circle passing through node u and node v (by empty circle property). If node u and v are in distance k range of node w , then the local Delaunay triangulation constructed by node w contains edge uv

because the empty circle property will hold locally if it holds globally. Edge uv is kept by all nodes that could see node u and node v in their distance k neighborhood. Edge uv is in the final LDT. So $UDel$ is a subset of LDT. If the original graph is connected, then $UDel$ is connected and LDT is connected. Similarly, we could prove that k -Local Delaunay triangulation (k -LDT) is a superset of $(k+1)$ -local Delaunay triangulation ($(k+1)$ -LDT) (with $k=1,2,\dots,d$). Compared with k -LDT, $(k+1)$ -LDT is more like DT with the increase on communication overhead.

Planar The proof is similar to the proof in [21].

Assume edge uv and edge wx intersects and both in the final local DT. Then $|uv| + |wx| \geq |uw| + |xv|$ and $|uv| + |wx| \geq |ux| + |wv|$. Either $|uw| \leq 1$ or $|xv| \leq 1$ and either $|ux| \leq 1$ or $|wv| \leq 1$. In either case, there will be one node that will see the other three node and hence the crossing edge would not exist in its local DT and would not exist in the final local DT. Contradicts with the assumption.

$\frac{4\sqrt{3}}{9}\pi$ **Spanner** In [34], Li *et al.* proved that $UDel(S)$ is a $\frac{4\sqrt{3}}{9}\pi$ spanner of UDG. As $UDel$ is a subset of LDT, it is clear that LDT is also a $\frac{4\sqrt{3}}{9}\pi$ spanner.

2.3.3 Localized Minimum Spanning Trees

2.3.3.1 Local Minimum Spanning Tree with Directed Edges

Li *et al.* [33] proposed a local minimum spanning tree algorithm for topology control. The local minimum spanning tree, which they call LMST, has directed edges, i.e., an edge uv belongs to local minimum spanning tree of node u and has the direction $u \rightarrow v$. Each

node u first collects its one-hop neighbors $N_1(u)$. Node u then computes the minimum spanning tree $MST(N_1(u))$ of the induced unit disk graph on its one-hop neighbors $N_1(u)$. Node u keeps a directed edge uv if and only if uv is an edge in $MST(N_1(u))$. They call the union of all directed edges of all nodes the local minimum spanning tree, denoted by G_0 . If only symmetric edges are kept, then the graph is called G_0^- , i.e., it has an edge uv if and only if both directed edge uv and directed edge vu exist. If the directions of the edges in G_0 are ignored, they call the graph G_0^+ , i.e., it has an edge uv if and only if either directed edge uv or directed edge vu exists. They prove that the graphs are connected, and have bounded degree 6.

2.3.3.2 k-Local Minimum Weight Spanning Tree

We define k-Local minimum weight spanning tree (k-LMWST) by applying k-local algorithm (definition in Section 2.1.1.3) of minimum spanning tree. Node ids are used to break the balance when two edges have the same weight. Larger id numbers are assumed to have larger weight. We show that the constructed k-LMWST of UDG has interesting properties summarized in the following theorem. Local minimum weight spanning tree (LMWST) refers to our k-local minimum weight spanning tree definition in the following thesis.

Theorem 1. *If G is a UDG, by applying k-LMWST algorithm, the obtained graph has the following properties.*

- a) *It is a planar;*
- b) *It is a superset of minimum spanning tree.*

Proof. To prove part a), we first prove that the k -local minimum weight spanning tree is a subset of RNG.

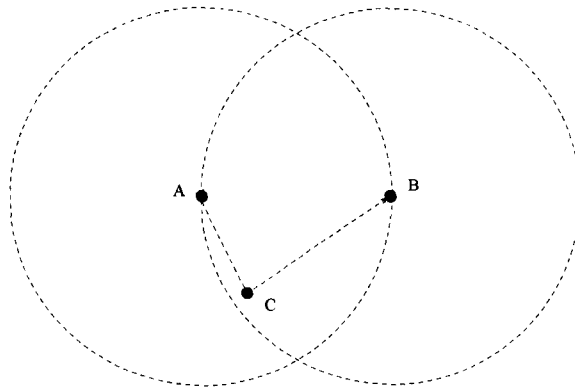


Figure 2.6: LMWST is a subset of RNG

Let C be a point in the lune of disk A and disk B (Figure 2.6). Since $|CA| < |AB|$ and $|CB| < |AB|$, the edge AB cannot be in the distance k local minimum weight spanning trees of both node A and node B . So edge AB could not be in the final local minimum weight spanning tree (by definition). We conclude by if an edge is not in RNG, then it is not in LMWST. LMWST is a subset of RNG.

As RNG is a subset of GG and GG is planar, LMWST must be planar. Otherwise there are crossing edges in LMWST and GG, which contradicts the fact that GG is planar.

We use similar method in [12] to prove part b). We argue by contradiction. Let the edge $\{u, v\}$ be retained in MST, but rejected in LMWST. Since $\{u, v\}$ was retained in MST, there is no other path in MST joining u and v and the edge length of $\{u, v\} \leq 1$. Since $\{u, v\}$ was rejected by LMWST, there exists a path, say p , in LMWST joining u and v using only edges smaller than $\{u, v\}$. Let $\{w, w'\}$ (see Figure 2.7) be an edge in p such that $\{w, w'\} \notin \text{MST}$. It follows that there is a path in MST joining w and w' using

only edges smaller than the edge $\{w, w'\}$. As this argument applies to each such edge of p , there must be a path in MST joining u and v using only edges smaller than $\{u, v\}$. This contradicts the fact that the edge $\{u, v\}$ was retained in MST .

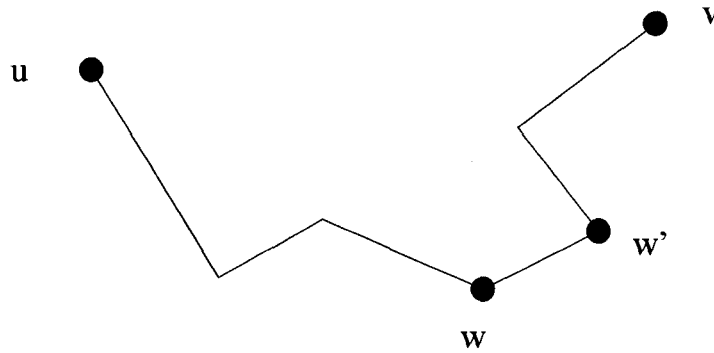


Figure 2.7: LMWST is a superset of MST

□

The above proof also applies to the relationship between k -LMWST and $(k+1)$ -local minimum weight spanning tree ($(k+1)$ -LMWST). We could show that $(k+1)$ -LMWST is a subset of k -LMWST (with $k=1,2,\dots,d$). Larger k will make the LMWST more like MST with the increase in communication costs.

Relationships of Geometric Graphs From the above discussion, the following relationships could be obtained among all the geometric graphs of MST, LMWST, RNG, GG, UDel, LDT.

$$\text{MST} \subseteq \text{LMWST} \subseteq \text{RNG} \subseteq \text{GG} \subseteq \text{UDeI} \subseteq \text{LDT}$$

2.4 Applications of Localized Geometric Spanners

Most localized geometric spanners are proposed in the scenario of MANET which normally has high node density and is highly connected. By applying face routing on these constructed spanners, messages could be delivered with certainty. Although global connectivity could not be guaranteed in DTNs, face routing logic could still be applied on different connected parts of the network.

2.4.1 Face Routing

2.4.1.1 Definition

In face routing, message has to follow a sequence of adjacent faces which are intersected by the straight line ST connecting the source node S with the destination node T . In Figure 2.8, message travels along Face A, Face B, Face C and Face D to the destination node T and never travels the same face twice. Face routing was first proposed by Kranakis *et al.* [31].

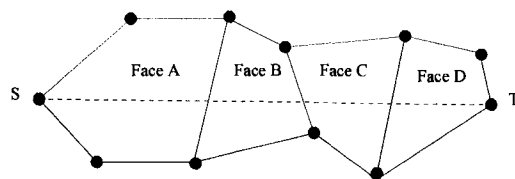


Figure 2.8: Face routing definition

2.4.1.2 The Need of Using Planar Graph

It is needed to use planar graph in face routing. Otherwise message delivery can not be guaranteed. In Figure 2.9, source S chooses the next hop B using the left hand rule, B choose C , C choose B and B choose S as the next hop. Although there exists

route $S \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow T$, the right route could not be found by the left hand rule without planar. By using planar graph, edge AC is eliminated and face routing could work properly.

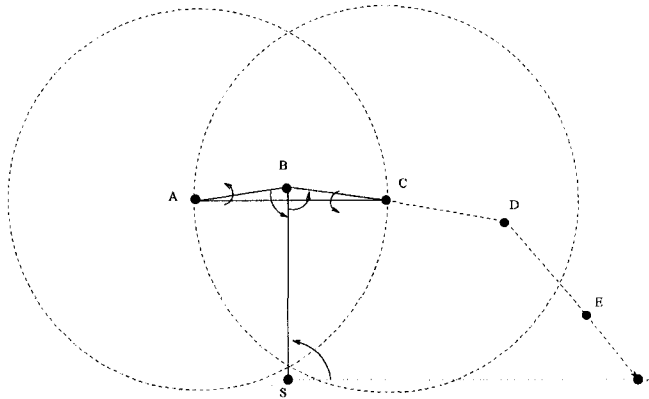


Figure 2.9: Illustration of face routing failure without planar

2.4.2 Greedy Face Greedy

Greedy face greedy (GFG) routing was proposed by Bose *et al.* [8]. The GFG routing logic works in the following way, as shown in Figure 2.10.

1. $P \leftarrow S$
2. *repeat*
3. *Let F be the face with P on boundary and intersecting PT*
4. *Traverse F until reaching an edge that intersects PT at some point $Q \neq P$*
5. $P \leftarrow Q$
6. *until $P=T$*

GFG works fine in most scenarios. But sometimes the routing logic could not find the destination node because it assumes routing could use either left hand rule or right hand rule upon an edge which intersects with the source to destination line segment. With

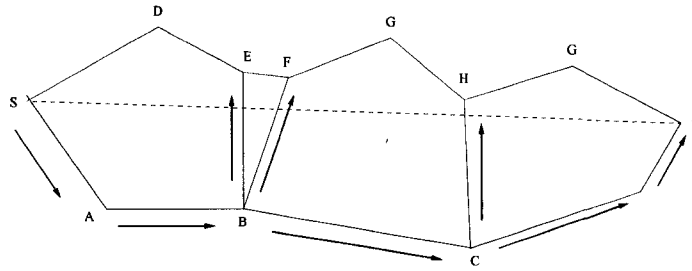


Figure 2.10: GFG routing

only one rule (right or left hand rule), routing could fail. Figure 2.11 gives out such an example.

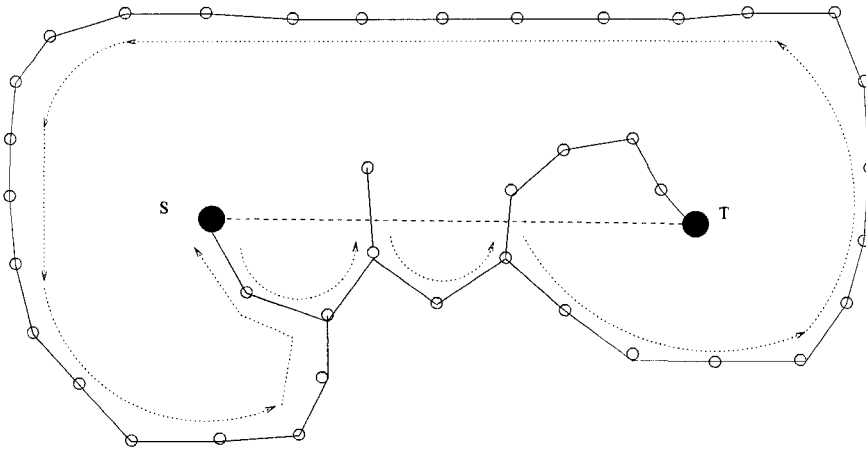


Figure 2.11: GFG routing loop

As shown in Figure 2.11, the graph is connected and there exists a path from source node S to destination node T. By using only right hand rule, routing fails.

2.4.3 Other Face Routing Protocols and Pitfalls

2.4.3.1 GPSR and Its Pitfalls

In [29], Karp *et al.* proposed greedy perimeter stateless routing (GPSR) for routing in MANET. They define right hand rule as the following (see Figure 2.12): The long-known right-hand rule states that when arriving at node x from node y , the next edge traversed

is the next one sequentially counterclockwise about x from edge (x,y) . The right hand rule which they defined is also referred as left hand rule by other face routing protocols because the routing logic could be considered as walking along the inner faces by using left hand touching the edges.

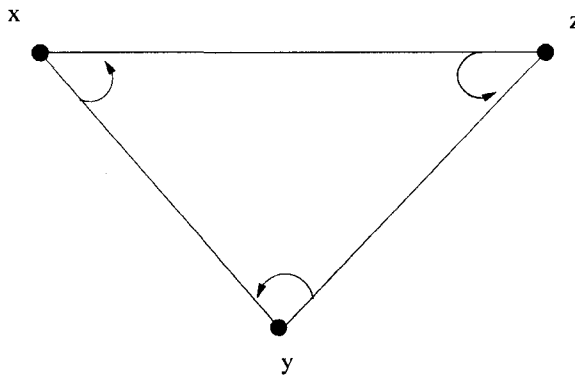


Figure 2.12: GPSR right hand rule

Their routing logic is defined as follows: On each face, the traversal uses the right-hand rule to reach an edge that crosses line \overline{ST} . At that edge, the traversal moves to the adjacent face crossed by \overline{ST} by using the right hand rule (see Figure 2.12).

It is known that GPSR face routing logic could fall in some pitfalls and messages could not find their destinations finally (see Figure 2.13). In figure (a), message travels from source node S along route $S \rightarrow A \rightarrow B \rightarrow A \rightarrow C \rightarrow A \rightarrow S$ according to the routing logic. Clearly, routing loop exists. In figure (b) message travels along route $S \rightarrow A \rightarrow B \rightarrow C \rightarrow B \rightarrow D \rightarrow B \rightarrow A \rightarrow S$, which is similar to the situation in figure (a) and exists loop.

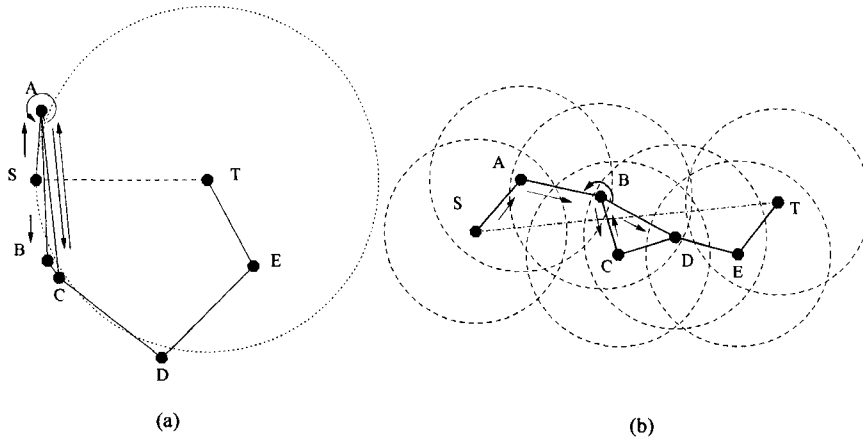


Figure 2.13: GPSR routing loop

2.4.3.2 GPVFR and Its Pitfalls

In [32], Leong *et al.* proposed Greedy path vector face routing (GPVFR). The routing logic defined in GPVFR is (see Figure 2.14):

1. Find the face F containing the line segment ST . Let A and B be the clockwise and anti-clockwise nodes. If $|AT| \leq |BT|$, forward the packet clockwise along F ; if $|AT| > |BT|$, forward the packet anti-clockwise.
2. If any adjacent edge intersects the line segment ST then set S to current node C and go to step 1.

As shown in [19], the routing logic of GPVFR could also fall in some pitfalls. In Figure 2.15, starting from source node v_1 to find the destination node T , message will travel along the routing path $v_1v_2v_3v_4v_5v_6v_1$ in both (a) and (b). Routing logic fails according to the above definition.

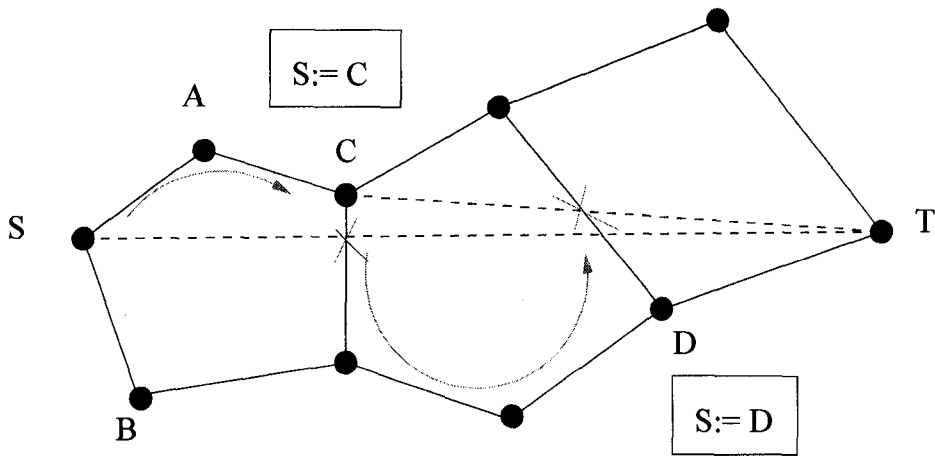


Figure 2.14: GPVFR routing logic

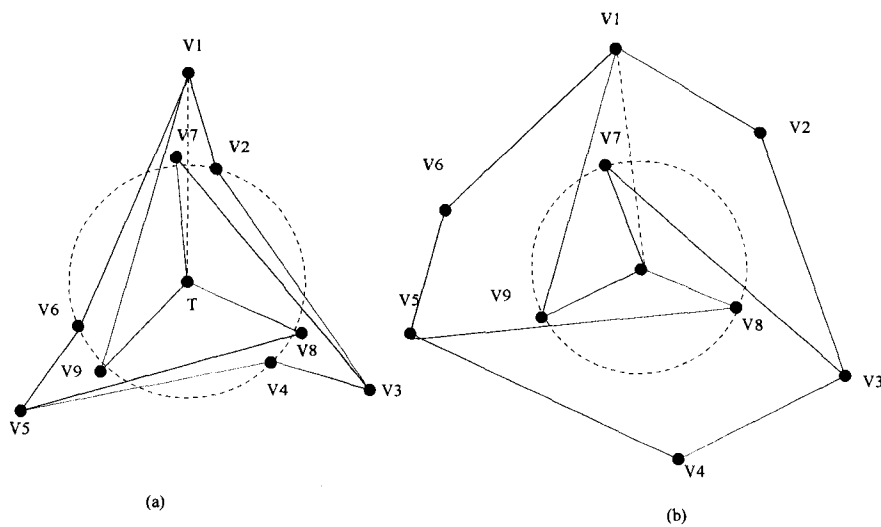


Figure 2.15: GPVFR routing loop

Chapter 3

Overview of DTN and Existing Routing Protocols

3.1 Introduction

3.1.1 Some Main DTN Applications

In situations where traditional networking solutions do not work well, the use of DTN solution provides an effective way for the proper functioning of many applications. In [16], Farrell and Cahill give out some primary application examples.

3.1.1.1 Environmental Monitoring

In many western European lakes, pollution of various kinds is becoming more and more of a problem. Monitoring the lake water quality is a necessary step toward the

protection of lake water quality. Comparing with the use of GSM, GPRS or 3G mobile communications, the DTN option with data mules is more environmentally friendly and the total cost is reduced.

3.1.1.2 Deep Space Applications

Over the coming decades, NASA, ESA and others are planning a series of robotic missions to Mars, the Moon and elsewhere. Given the light-speed delays involved, the communications in these missions should also be considered as another type of Delay Tolerant Networking.

3.1.1.3 Other Applications

The following is a list of some other DTN applications:

1. Noise Monitoring
2. DTN in the Developing World
3. Earthquake Monitoring
4. Undersea Acoustic Networking
5. Forest Fire Support
6. Glacial Sensor Network
7. Zebranet
8. Sami Network Connectivity

9. Tactical Military Networks

3.1.2 DTN Research Groups

There are several research groups (projects) working on the architectures and protocols of disruption (delay) tolerant network. DTNRG, DARPA DTN and InterPlaNetary Internet Research Group (IPNRG) are three of them. The following figure was used in [17] to depict the different but overlapping groups of people working on this topic.

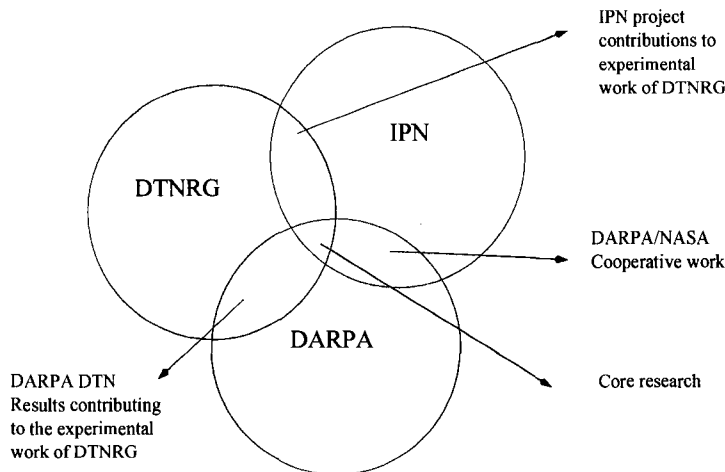


Figure 3.1: Main organizations working on DTN topic

DTNRG (Delay Tolerant Network Research Group) [1] aims to design and implement architectures and protocols for networks, where no end-to-end connectivity can be assumed and which differ in the characteristics of the known Internet. The so-called “bundle” protocol was developed by DTNRG to be one main contender for a standard DTN protocol.

3.1.3 Bundle Protocol

The mainstream DTN approach treats bundle protocol as an overlay network that can sit on top of nearly any combination of networks and still provide an approximation of end-to-end connectivity. In bundle protocol, each packet transmitted is called a “bundle” and contains all of the signalling as well as the data required to transit the overlay network. The following figure shows where the bundle layer locates inside the protocol stack, as given in [53].

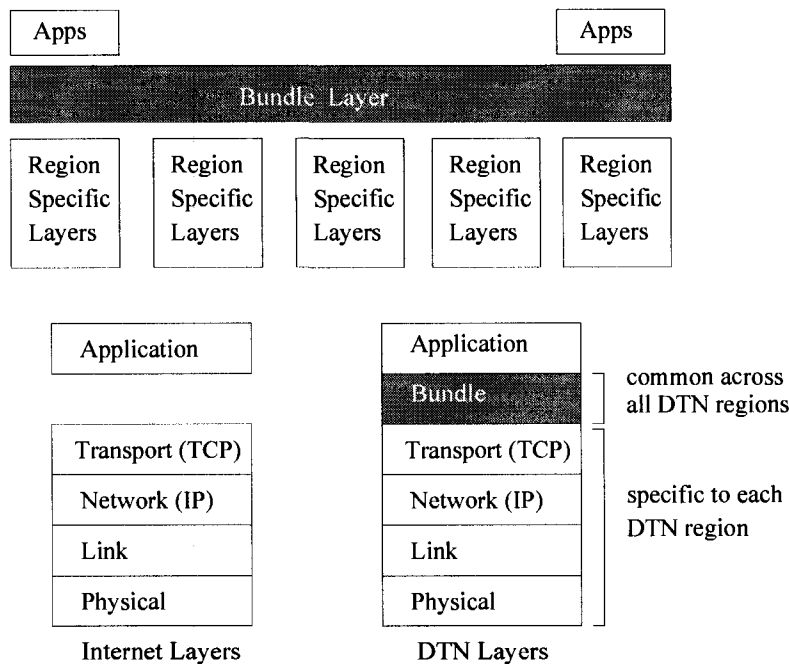


Figure 3.2: Bundle layer

3.1.4 Characteristics of Disruption (Delay) Tolerant Network

Traditional Internet protocols could not work on the disruption (delay) tolerant network because of the special characteristics of DTNs.

- **Intermittent Connectivity:** end-to-end path between source and destination does not always exist due to node movement.
- **Long Delay:** long distance between nodes causes long propagation delays.
- **Asymmetric Data Rates:** different network topology or different transmission direction may use different data rate.
- **High Error Rates:** extreme environments may introduce high error rates during transmission.

3.2 Overview of Major DTN Routing Protocols

Disruption (Delay) tolerant networks are also referred to as intermittently-connected networks, sparse networks, and disconnected networks which are the networks without constant end-to-end connections. There has been extensive research on routing in disruption (delay) tolerant networks.

3.2.1 Epidemic Routing

Epidemic routing [51] is based on selecting carriers that have the task to find the destination. These carriers serve as a form of mobile agents which distribute many copies of same message to a large number of nodes with the hope that one of these intermediate nodes will reach the destination. Every receiving node becomes a new carrier for that message. It attempts to give all nodes a copy of every message through random exchanges

between nodes. Since this is similar to how some diseases spread, it was called epidemic routing, as proposed by Vahdat *et al.* [51]. This approach can achieve high delivery ratio. If it is provided with infinite bandwidth and buffer resources, it will deliver all the messages that can possibly be delivered in the minimum amount of time without prior knowledge about the network. But it is very expensive in terms of the number of transmissions and buffer space and it does not appear that this approach can scale as the number of messages and nodes in the network increase.

The process of epidemic forwarding could be depicted in Figure 3.3. Suppose at time t_1 , source node S has a message for destination node D. When node 1 and node 2 come into communication range (radius r) of node S at time t_2 , S sends out the message to node 1 and 2 respectively. Node 2 relays the message to node 3 at time t_3 and node 3 delivers the message to destination D at time t_4 .

Epidemic routing relies on carriers of messages coming into contact with another node through node mobility. When two nodes come into communication range with each other, they exchange the list of messages they hold (called summary vector) first. Messages that the other node does not have are exchanged following that. The process is shown in Figure 3.4.

One apparent drawback of this routing protocol lies in that the messages will never be cleared. To clear the messages which have already been delivered to the destinations, some kind of acknowledgement has to be developed. K. Harras and K. Almeroth in their paper [24] present several approaches to solve this issue. Active receipt or passive receipt is generated to clear the already delivered messages. In the active mode, nodes will send

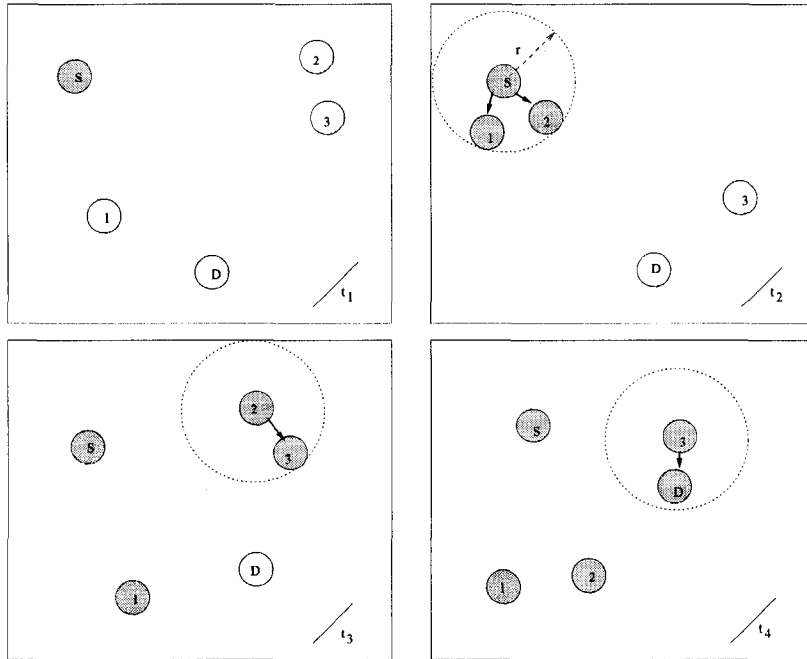


Figure 3.3: Illustration of time evolving behavior of epidemic transmission

active receipts to notify all the nodes they meet that some messages have already been delivered to the destination. While in the passive mode, nodes will only send receipts when there are some other nodes trying to send messages to them which are known to have been delivered already. But no matter what the situation is, more messages are generated in the network and how to stop the broadcasting of the receipt messages is another question.

Various protocols [13, 11, 46, 49] have been proposed to improve the efficiency of epidemic routing in recent years. While storage and bandwidth are reduced in these protocols, many copies of each message are still transmitted inside the network which will never be cleared.

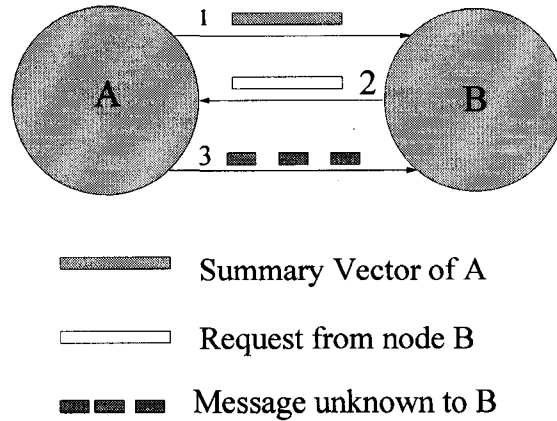


Figure 3.4: Epidemic routing

3.2.2 Message Ferrying

Instead of letting the mobile host wait passively for the network to reconnect, controlling node mobility is another way to limit delivery delays in DTNs. Controlling moving agents is distinctively different from other active schemes for routing in the sparse networks, in the sense that in this model special mobile nodes called Message Ferries, or mobile agents are used to facilitate connectivity among nodes. The role of the message ferries (or simply ferries), is to visit the nodes in the network and deliver data among them. This is an attractive model because it allows serving a variety of networks and also allows achieving explicit end-to-end message delivery characteristics. Additionally, it takes the burden of message routing away from the nodes, which may be desirable when nodes have limited energy and storage resources. Message Ferrying (MF) [55] is one such protocol. The main idea behind the MF approach is to introduce non-randomness in the movement of nodes and exploit such non-randomness to help deliver data. Two variations of MF, depending on whether ferries or nodes initiate proactive movement are introduced.

In the Node-Initiated Message Ferrying (NIMF) scheme, ferries move around the deployed area according to known routes and communicate with other nodes that they meet. Nodes move closer to ferries when trajectory control logic asks them to do so for the purpose of sending or receiving.

In the Ferry-Initiated Message Ferrying (FIMF) scheme, ferries move proactively to meet nodes that have messages to send or to receive. The requesting nodes notify ferries using long range radio. Ferries move closer to the requesting nodes according to the requests and exchange messages with these nodes using short range radio to reserve energy. After the message exchange actions, ferries move back to their original trajectories.

3.2.3 Model Based Routing (MBR)

In [5], Becker and Schiele proposed a Model based routing (MBR) approach. The model includes three parts: the location model (Fig.3.5), the user profiles and the sensor receptions. The location model contains information such as road maps and building charts. Location information of different places of the buildings and road maps are also included. They assume that mobile devices are typically carried by human beings which follow some specific motion patterns. As shown in the figure, a user could stay at a given location at a given moment in time with a known probability during the office hours. This user profile is collected before routing. With the information of the possible receiver locations, the location model and user profiles, source node could select some relay nodes that have higher probability to move near or to the destination location.

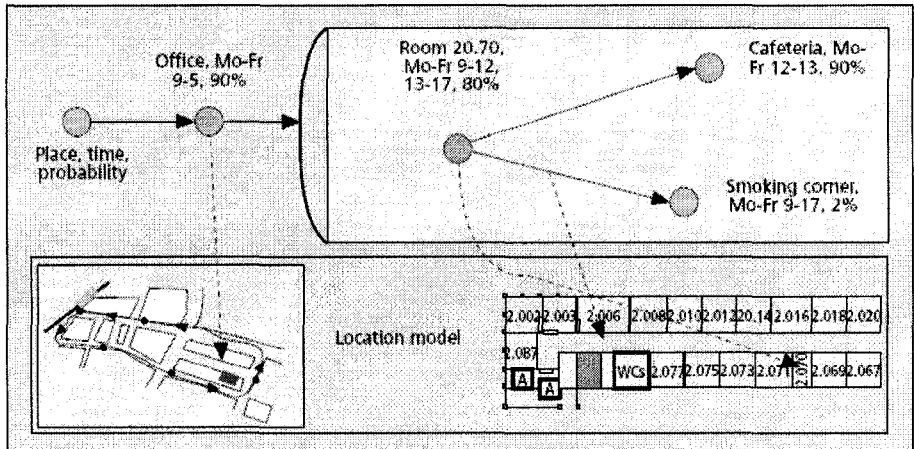


Figure 3.5: Local model and user profile example [54]

3.2.4 Interrogation-Based Relay Routing

In [45], Shen *et al.* proposed interrogation based relay routing protocol. Satellite location and orbit information is exchanged during the process of interrogation. When two ad hoc satellites come closer to each other, they engage in an interrogation process to exchange orbital and routing information between each other in the form of queries and responses. This information is used by the nodes to select a next hop node. Clearly, every satellite node knows its own location information well in advance.

3.2.5 MaxProp

In [9], Burgess *et al.* proposed MaxProp, a protocol specially designed for communication among vehicles. MaxProp is based on prioritizing both the schedule of packets transmitted to other peers and the schedule of packets to be dropped. These priorities are based on the path likelihoods to peers according to historical data and also on several

complementary mechanisms, including acknowledgments, a head-start for new packets, and lists of previous intermediaries.

3.2.6 MEED

In [27], contact history among nodes are exploited to make routing decisions and increase delivery probability. Minimum estimated expected delay (MEED) is calculated and the contact that has the least MEED value is chosen as the next hop.

3.2.7 DTLSR

In [14], Delay Tolerant Routing for Developing Regions (DTLSR) is modeled closely on classic link state algorithms. As the network state changes, link state announcements are flooded throughout the network. Each node maintains a graph representing its current view of the state of the network, and uses a shortest path computation (e.g., Dijkstra) to find routes for messages. The authors designed this routing protocol [14] specially for delay tolerant routing in developing regions which have comparatively stable and predictable underlying topological structures. Because a node may have an incomplete or inaccurate view of the network, MEED approach is adopted when the best paths are calculated.

3.2.8 Space and Time Routing

In [37], routing path is constructed in space and time dimensions with the assumption that all contacts among nodes could be accurately predicted over the time interval of T . The explicit node movements create paths over time that include the possibility of a node

carrying a message before forwarding to another suitable node. In most of these networks, the mobility of nodes is predictable either over a finite time horizon or indefinitely due to periodicity in node motion. Merugu *et al.* [37] propose a new space-time routing framework for these networks leveraging the predictability in node motion. Specifically, they construct space-time routing tables where the next hop node is selected from the current as well as the future neighbors. Unlike traditional routing tables, their space-time routing tables use both the destination and the arrival time of message to determine the next hop node. They devise an algorithm to compute these space time routing tables to minimize the end-to-end message delivery delay. Their routing algorithm is based on a space-time graph model derived from the mobility of nodes.

The time layers in the space-time graph comes from time-expanded graph [18] which translates a dynamic network flow over time to a “static” network flow. Floyd-Warshall algorithm is applied to the derived graph to compute the shortest path for a scheduled message in the proposed routing protocol.

In this routing protocol, an end-to-end path over time (may be time dependant) is determined and the routing table is also established well in advance with the almost full knowledge of the network.

3.2.9 Spray and Wait

According to the protocol proposed by Spyropoulos *et al.* [48], nodes will spray and wait using either two hop or binary spray and wait method. According to the two hop relaying method, the source is allowed to spread up to a maximum number of copies

within the network. Each time source node encounters some other node with no copy of the message, it gives other node one message copy until it has only one copy left (for the destination node only). Each of these nodes carrying a message copy performs direct transmission, i.e., forwards the message only to its destination. In the binary spray and wait method, the source of a message initially starts with L copies. Any node A that has $n(>1)$ message copies (source or relay), and encounters another node B (with no copies), hands over to B $\lfloor n/2 \rfloor$ and keeps $\lfloor n/2 \rfloor$ for itself. When it is left with one copy, only direct transmission is allowed. Binary spray and wait is faster than the two hop relaying method and induces no more transmissions than those of the latter.

3.2.10 Utility Based Routing

In [47], the authors define a utility function $U_X(Y)$, maintained by node X for destination node Y which indicates how useful node X might be in delivering a message to a node Y . Routing decisions are made according to the value of the utility function. Node A forwards to another node B a message destined to a node D , if and only if $U_B(D) > U_A(D)$.

3.2.11 Knowledge of Motion Profiles Approach

In [23], knowledge associated with the motion and availability profiles of the devices making up the disruption tolerant network is exploited. By assuming source node A has full knowledge of the characteristic profiles for all hosts, it can compute a message delivery path from A to some destination host H if such a path exists. This algorithm for message passing depends on knowledge that is global with respect to space and time, i.e., the

motion and availability profiles for all hosts. Path trees are constructed according to the global knowledge.

3.2.12 Erasure Coding Based Routing

In [52], erasure coding technique is used to increase the probability of delivery and reduce delay. Erasure codes operate by converting a message into a larger set of code blocks such that any sufficiently large subset of the generated code blocks can be used to reconstruct the original message. Suppose the original message is encoded into k blocks at the beginning, by using erasure coding, the message is encoded into n ($n > k$) blocks such that if k or more of the n blocks are received, the original message could be successfully recovered. Here, $r = n/k$ is called the replication factor and determines the level of redundancy. The above method was called as (n, k) -threshold scheme by Shamir [44] and Blakley [6] independently in the field of cryptography.

Wang *et al.* suggested in [52] that the source node would code the original message using erasure coding with replication factor r and the coded message blocks are equally split among the first kr contacts (k has the same meaning as above). The kr nodes with message blocks only deliver to the destination. The original message could be decoded as soon as k contacts deliver their data to the destination.

When $k=1$, the erasure coding approach has the same effect as the simple replication approach, which is, the source node sends the message to the first r nodes it meets which deliver the message only to destination node.

Simulation results show that using erasure coding ($k > 1$) improves the worst case delay

than simple replication ($k=1$).

3.2.13 PRoPHET

In [35], Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) is proposed. PRoPHET first establishes a probabilistic metric called delivery predictability, $P(a,b) \in [0, 1]$, at every *node a* for each known destination *node b*. This indicates how likely it is that this node will be able to deliver a message to that destination. The calculation of the delivery predictability has three parts. The first thing to do is to update the metric whenever a node is encountered, so that nodes that are often encountered have high delivery predictability. The calculation is shown in the Eq.3.1, where $P_{init} \in (0, 1]$ is an initialization constant.

$$P_{(a,b)} = P_{(a,b)old} + (1 - P_{(a,b)old}) \times P_{init} \quad (3.1)$$

If a pair of nodes does not encounter each other in a while, they are less likely to be good forwarders of messages to each other, thus the delivery predictability values must age, being reduced in the process. The aging equation is shown in Eq.3.2, where $\gamma \in (0, 1)$ is the aging constant, and k is the number of time units that have lapsed since the last time the metric was aged. The time unit used can differ, and should be defined based on the application and the expected delays in the targeted network.

$$P_{(a,b)} = P_{(a,b)old} \times \gamma^k \quad (3.2)$$

The delivery predictability also has a transitive property, which is based on the observation that if node A frequently encounters node B, and node B frequently encounters node C, then node C probably is a good node to forward messages destined for node A too. Eq.3.3 shows how this transitivity affects the delivery predictability, where $\beta \in [0, 1]$ is a scaling constant that decides how large impact the transitivity should have on the delivery predictability.

$$P_{(a,c)} = P_{(a,c)old} + (1 - P_{(a,c)old}) \times P_{(a,b)} \times P_{(a,b)} \times \beta \quad (3.3)$$

After the calculation of delivery predictability, a message is sent to the other node if the delivery predictability of the destination of the message is higher at the other node when two nodes meet. The first node does not delete the message after sending it as long as there is sufficient buffer space available (since it might encounter a better node, or even the final destination of the message in the future). If buffers are full when a new message is received, a message must be dropped according to the FIFO queue management system.

3.3 Summary of DTN Routing Protocols

A summary of these routing protocols, together with the proposed approach in Chapter 4, is shown in Table 3.1.

Table 3.1: Summary of Routing Mechanisms

Protocols	Disadvantages	Advantages
Epidemic Routing [51]	<ol style="list-style-type: none"> 1. Needs infinite buffer space 2. Infinite control messages 	Has high delivery ratio and desirable delay
Message Ferrying [55]	<ol style="list-style-type: none"> 1. Special nodes are needed 2. Needs long range radios in FIFM 3. Needs node trajectory control in NIFM 4. All nodes have global knowledge on special nodes in NIFM 5. Not suitable for random network 	Efficient in data delivery and energy consumption
Model Based Routing [5]	<ol style="list-style-type: none"> 1. Extra user profiles need to be collected beforehand 2. Central location service required 3. Only a sketch of the algorithm is given 	Messages in transit are reduced
Interrogation Based Relay Routing [45]	<ol style="list-style-type: none"> 1. Global knowledge is needed 2. Performance in other networks may not be the best because it is specially designed for satellite communication 	Route could be selected in a better way
MaxProp [9]	<ol style="list-style-type: none"> 1. May experience long delays 	Better storage usage with ACK in message exchange
MEED [27]	<ol style="list-style-type: none"> 1. Long contact history among nodes is needed 	Suitable for relative stable networks
DTLSR [14]	<ol style="list-style-type: none"> 1. Link state flooding is needed 2. Link may not be available at the time of making route decisions 3. Estimation based scheme may experience long delays 	Better storage usage

Table 3.1 (continued): Summary of Routing Mechanisms

Protocols	Disadvantages	Advantages
Space and Time Routing [37]	1. Global knowledge is needed	Best route could be used
Spray and Wait [48]	1. Needs infinite buffer space in certain situations 2. May experience long delivery delays	Flooding is controlled
Utility Function Based [47]	1. May experience high delays when utility function not accurate 2. Estimation may not be accurate	Active approach to facilitate delivery
Knowledge of Motion Profiles Approach [23]	1. Global knowledge is needed 2. Path tree construction is not easy to scale	Best route selection is possible
Erasur Coding Based Routing [52]	1. Needs infinite buffer space 2. Not optimal in terms of delivery latency 3. Erasure coding ability is required	Best worst case delay is achieved
PRoPHET [35]	1. Needs infinite buffer spaces 2. Multiple copies without control	Reasonable buffer utilization among nodes
Proposed Solution (GLR) in Chapter 4	1. Computation is complex	Efficient storage utilization Fast delivery High delivery ratio

3.4 Classification of DTN Routing Protocols

In the survey of Zhang [54], routing protocols are divided into deterministic and stochastic. Farrell *et al.* [16] supplemented the above classification with some schemes presented at a DTNRG meeting [3]. The following subsections give out the list of schemes given by them.

Oracle Scheme The full network topology information is known at some node or sets of nodes under this scheme. Thus the routing process could be properly arranged by using this available knowledge. The classic example is space networking where spacecraft ephemeris and communication schedules are carefully planned and known far in advance. In Zhang's survey [54], the oracle scheme is called deterministic. Knowledge of motion profiles approach [23], Interrogation based relay routing [45] and Space and time routing [37] all belong to this scheme.

Model-Based Schemes Model of the network could be established because in some DTNs, nodes move in more or less a predictable way. Thus it is possible to build a model of the network and thereby gain some information that can be used to improve routing. Model based routing [5] belongs to this scheme.

Epidemic Schemes Nodes will forward messages to every next hop because it is not sure which route will work in this scheme. Epidemic [51] and Spray and wait [48] belong to this scheme.

Estimation Schemes Nodes improve upon epidemic routing by estimating the probability of forwarding message to the next hop which will leads to the successful delivery to the destination. PRoPHET [35], Utility based routing [47], MEED [27] and DTLSR [14] all belong to this scheme.

Erasur Coding Scheme Packets are encoded with redundancy so that the packets could be transmitted in a more reliable manner. Erasure coding [26] belongs to this scheme.

Node Movement Control Scheme In this scheme, some relay nodes are added to the network to increase the successful delivery rate. These nodes are called data mules or message ferries. The movement of these data mules are controlled in order to improve the efficiency of this scheme. Message Ferrying [55] belongs to this scheme.

Chapter 4

Proposed Solutions

Our proposed solutions use geometric information to construct localized Delaunay triangulation. A store and forward mechanism is added according to the special characteristics of the DTN. Controlled multi copies of same message are transmitted in the network to reduce delay. The routing protocol adjusts the number of message copies according to the connectivity characteristic intelligently.

4.1 Goals

The following goals are kept in mind in designing the proposed solutions:

- Low delay
- High delivery ratio
- Limited memory requirement

4.2 System Design

We noticed the need of intelligence in DTNs. Our Geometric Localized Routing (GLR) first decides the needed number of message copies intelligently based on different situations. More message copies are sent into network when the network is sparse and the node communication range is short. Then GLR constructs localized Delaunay triangulation (LDT, Section 2.3.2.3) by collecting distance two neighborhood related information. From there on, different spanning trees are extracted from this localized planar spanner. These trees provide fault tolerance as well as disruption and delay tolerance. A message travels along an extracted spanning tree in the direction from the source to destination node and at each step the message moves closer to the destination node. A store and forward mechanism is introduced when no actions could be taken temporarily. When there are messages in its storage, a node checks regularly to see if further opportunities appear to extract source to destination trees because of the movement of nodes. Message storage space is freed after the message has been properly delivered to the next hop. During the geometric routing, custody transfer is used to guarantee that message would not be lost during transmission, that is, the message receiver notifies the message sender it has taken control of the delivered message. The sender could free the corresponding message storage space when custody transfer acknowledgement is received. When greedy forwarding path has been not available for some time, nodes try face routing on extracted planar graph to avoid the local minimum problem. The planar graph could be LDT or LMWST. It depends on the application's needs to choose either LDT or LMWST, i.e., it is better to

choose LDT if reducing the delivery latency is needed and LMWST should be chosen if energy conservation is the first priority.

Compared with epidemic routing, GLR achieves better storage utilization. It should also be faster because contention could be avoided by allowing only a few number of same message in transit. Simulations are also performed under storage limitations to see the comparison between GLR and the epidemic one.

4.2.1 Greedy Geometric Routing with Intelligently Controlled Flooding

4.2.1.1 Intelligence

In our GLR routing, flooding is controlled by intelligence. At times when a source node has message for a destination node, intelligently deciding the number of duplicate message copies is the first thing that needs to be done. As shown in Algorithm 1, the more sparse the network is, the more number of copies of same message are transmitted in the network. Any node can calculate the network connectivity and density of nodes by using the number of nodes, the radius range and the area of a given region. So the node knows the possibility of connection. In [22], Georgiou *et al.* show that for any positive real number s , the network $G(P, r_n)$ with a set P of n nodes and radius r_n is connected with probability of at least $1 - \frac{1}{s}$, for $r_n \geq \sqrt{\frac{\ln n + \ln s}{n\pi}}$. The higher the radius range, the more likely the network could be connected. If the network is dense and the network could be connected at some time, single copy is enough for a fast delivery. Multiple message copies

should be avoided. Otherwise, great contentions will lead to long delay. If it is impossible for a network to be connected, multiple copies approach should be used.

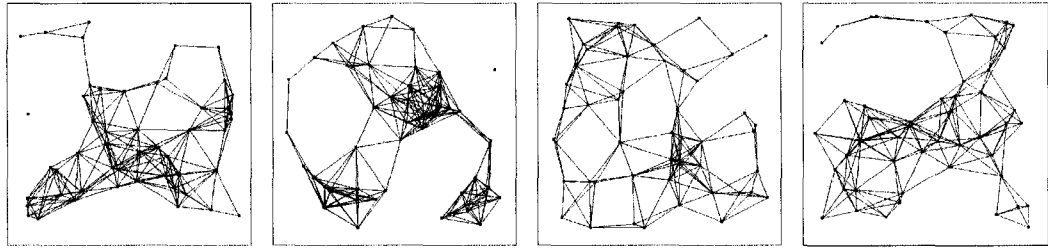
Algorithm 1 Decision Making with Intelligence

```
1: procedure INTELLIGENCEDECISIONMAKING      ▷ Refer Section 5.1.4 for detail
2:   if Network is sparse then
3:     Decide the number of message copies needed
4:     Send multiple copies of same message into network
5:   else
6:     Use single copy
7:   end if
8: end procedure
```

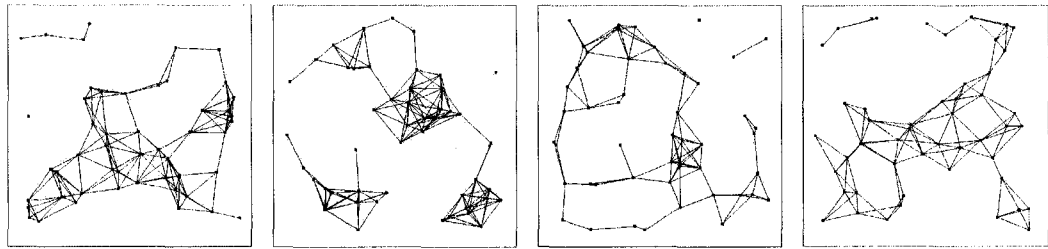
We generate four different scenarios and calculate the connection edges of 50 randomly generated nodes within range 1000×1000 with radius 250, 200, 150 and 100 respectively. Only the edges less than the radius are kept which represent connectivity. The final results are shown in Figure 4.1.

These figures clearly show that when the radius is 250, the networks are either connected or only one node is disconnected with other nodes. When the radius is 200, most nodes of the network are connected together although partition exists. In both scenarios, the possibility that source node could send directly to the destination node is high. Even when at some time the destination node could not be reached directly, with nodes moving, it is very likely that a new path will emerge. The node that holds the message waits and resends it when network topology changes with node movement. Single message copy has great opportunity to reach the destination quickly in most scenarios.

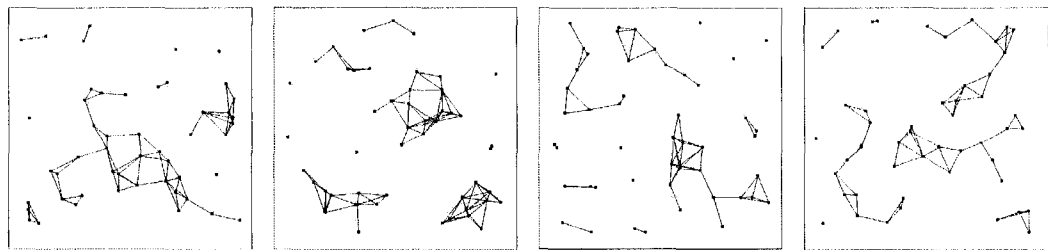
If the scenario is as shown in (c), part of the network is connected and this part connection could also be used. During the routing process, packet is delivered toward



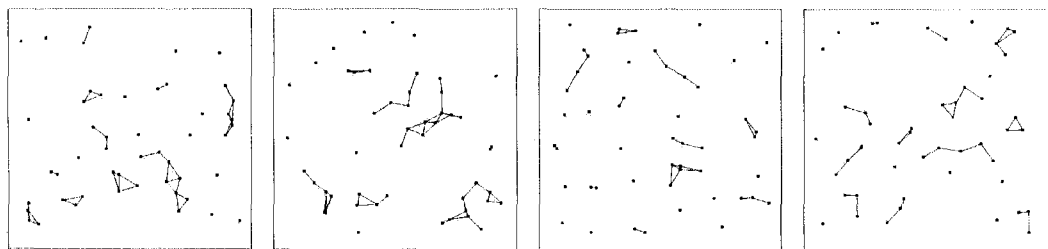
(a) radius 250



(b) radius 200



(c) radius 150



(d) radius 100

Figure 4.1: Topology of 50 nodes with radius 250,200,150,100 in 1000×1000 area.

destination in one connected part until it could not go forward anymore. Then the node that holds the message just turns into wait state and forwards the message later. To avoid local minimum, face routing could be applied on the partly connected network when greedy forwarding does not work. The increase of message copies could increase delivery opportunity and reduce delay.

In scenario (d), the possibility of network connection is very low. Compared with other three scenarios, more copies of same message should be sent into network to increase the delivery probability and decrease delivery latency. Figure 4.2 shows another extreme situation with 10 meter low radius range, 50 nodes in 1500×300 area. No two nodes could have connection. Even the number of nodes was increased to 2000, still only a few connections could be observed. This should be the reason why it takes 55 hours for epidemic routing using the same parameters to route packets while still not having delivered more than 10% of the packets.

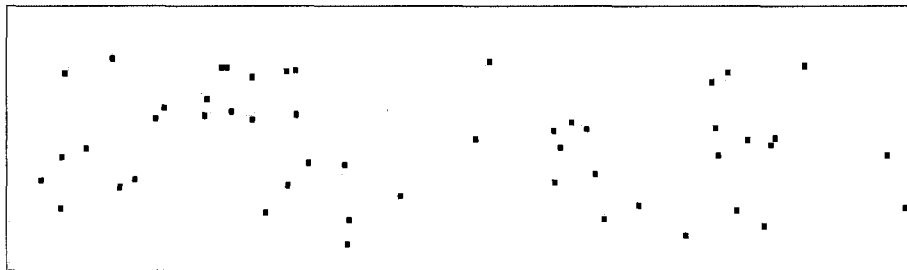


Figure 4.2: Topology of 50 nodes with radius 10 in 1500×300 area

4.2.1.2 Geometric Routing with Controlled flooding

We construct localized Delaunay triangulation by using the k-local algorithm as mentioned in Section 2.1.1.3. Before we greedily route the packet to the next node closer to the destination, three trees are extracted using methods similar to MaxDLTR, MinDLTR and MidDLTR [22] from the underlying geometric spanner. The difference lies in that the direction is not the left to right direction. Rather it is source to destination direction. We call the trees as Max distance source to destination (MaxDSTD), Min distance source to destination (MinDSTD) and Mid distance source to destination (MidDSTD) respectively. In MaxDSTD, each node is connected to a neighbor that makes Max progress (e.g., closest to the destination node) to the destination while in MinDSTD and MidDSTD, each node is connected to a neighbor that makes Min and Mid (other than Max and Min if it has one) progress to the destination. By using the extracted trees, message travels from the source to the direction of destination along different paths. Three trees provide faster delivery in the network with disruption compared with only one tree being used. We analyze this in Section 6.2.2. If only MaxDSTD tree is used, it is the same as single copy approach. Unlike the MaxDSTD tree and the MinDSTD tree, the MidDSTD tree has more options, i.e., a node may have several mid distance neighbors which make progress to the destination and any one could be selected in the tree. If more than three identical message copies are needed by source node's intelligence decision process, multiple MidDSTD trees could be extracted.

The geometric routing process is shown in Algorithm 2.

Algorithm 2 Geometric Routing with Controlled Flooding

```
1: procedure GREEDYGEOMETRICROUTING
2:   while Messagecount  $\neq$  0 do
3:     Collect neighboring nodes information
4:     Construct LDT
5:     for Every Message do
6:       if There are neighbors closer to destination then
7:         Messagecount  $\leftarrow$  Messagecount - 1
8:         Release message storage space
9:         if Source node then
10:          if Need more copies then
11:            Extract MaxDSTD, MinDSTD and MidDSTD tree neighbors
12:            Set message maxflag  $\leftarrow$  1
13:            Set message minflag  $\leftarrow$  1
14:            Set message midflag  $\leftarrow$  1
15:            Deliver message to these neighbors
16:          else
17:            Extract MaxDSTD neighbor
18:            maxflag  $\leftarrow$  1
19:            Deliver message to MaxDSTD neighbor
20:          end if
21:        else
22:          if Message maxflag = 1 then
23:            Extract MaxDSTD neighbor
24:            Deliver message to MaxDSTD neighbor
25:          end if
26:          if Message minflag = 1 then
27:            Extract MinDSTD neighbor
28:            Deliver message to MinDSTD neighbor
29:          end if
30:          if Message midflag = 1 then
31:            Extract MidDSTD neighbor
32:            Deliver message to MidDSTD neighbor
33:          end if
34:        end if
35:      end if
36:    end for
37:    StartTimer(checkinterval)
38:  end while
39: end procedure
```

Messagecount reflects the number of messages a node has. Node initiates the geometric routing process if there are messages in its storage area. LDT (defined in Section 2.3.2.3) is constructed first. Messages are treated differently if their corresponding destinations are different. If a node finds closer neighbors for a message, it removes the message from message storage area and decreases the *Messagecount*. Source decides the number of identical message copies needed, extracts corresponding tree next hops from LDT (only MaxDSTD if single copy approach is chosen) and sets flags of messages. Relay node only needs to route message following the tree specified by the message flag.

When node comes to a state that there is no other actions could be taken for a message, it stores the message and forwards it later. Timer is used when node enters into store state. When timer expires (the interval is *checkinterval*), node will check neighboring nodes location to see if any change occurs. When its relevant location with neighboring nodes has changed and new path emerges in the locally constructed trees, this node will send out the stored message. Destination node location is acquired during message delivery through the *Collect neighboring nodes information* process. This process is needed because the destination node may move far out of the original area and geometric routing may fail otherwise.

We collect data on how the routing protocol uses the underlying geometric graph finally in the network with disruption. Our properly designed protocol shows that the delivery latency is desirable. We presents this during the simulation in Chapter 6.

4.2.2 Delivery Guaranteed Face (DGF) Routing

As mentioned in section 2.4.2, greedy face greedy routing can not guarantee delivery. To solve this problem, two copies approach should be adopted, i.e., upon a crossing edge, left hand rule and right hand rule are applied at the same time.

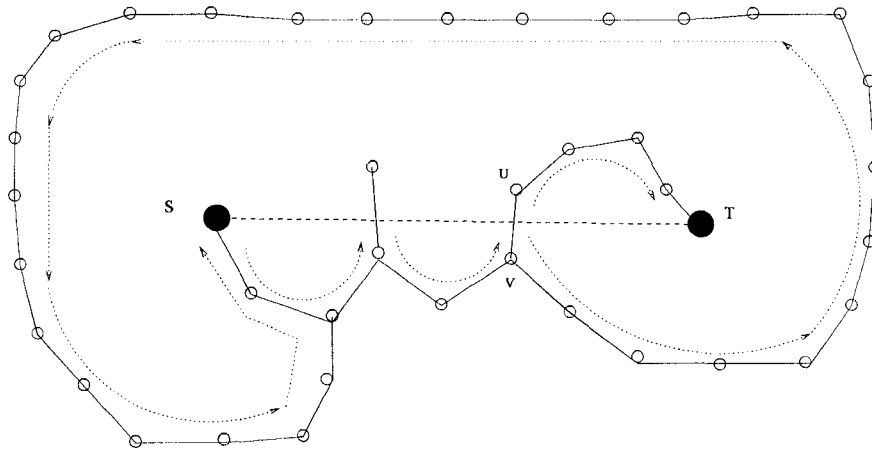


Figure 4.3: DGF routing

In Figure 4.3, when edge UV crosses line ST, one message traverses to the next hop using left hand rule from node U above line ST and another same copy traverses to the next hop using right hand rule from node V (see dashed line).

A node keeps a record of all the messages it received for some time. If the same message arrives, it is just discarded. This could be shown in Figure 4.4. By using both left and right hand rules, node S sends out message along edge SA and SD respectively. Suppose the message reaches node B first, B will deliver the message to node E in the other side of the crossing edge. The message will be delivered further along edge BF and EF. When the message from node D reaches node E, node E will discard it seeing the same message already being transmitted.

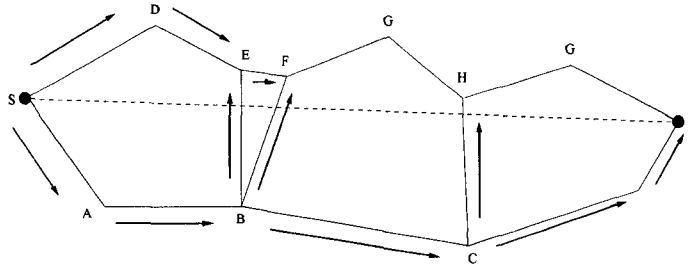


Figure 4.4: DGF routing: handling duplicate messages

To deal with the situation that message goes along face edges that never ends up at the destination node, hop count is used to terminate the message. After exceeding a specified maximum hop count, message is considered as going along the wrong direction and is discarded thereafter.

4.2.3 Face Routing Planar Graph Selection

Two planar graphs are extracted from the original UDG, i.e., LMWST and LDT (with $k=1$). Both are planar graphs with different properties. The LMWST is the graph which has desirable total weight while the LDT is the graph with constant stretch factor. Depending on system requirements, face routing could select LMWST for better energy usage or select LDT for fast delivery.

In Figure 4.5, although there is direct connection between source node S and destination node T, edge ST would not exist in LMWST while it exists in LDT. By using left hand rule, face routing goes along from node B to node E. The tree branch from node E to node H is explored and back to node E. Following that, the face routing goes from node I to node T. In LDT, message is delivered from node S to node T directly. More hops normally mean longer delay.

Even by using right hand rule, face routing from S to T in the LMWST still travels two hops, one hop more than LDT in Figure 4.5.

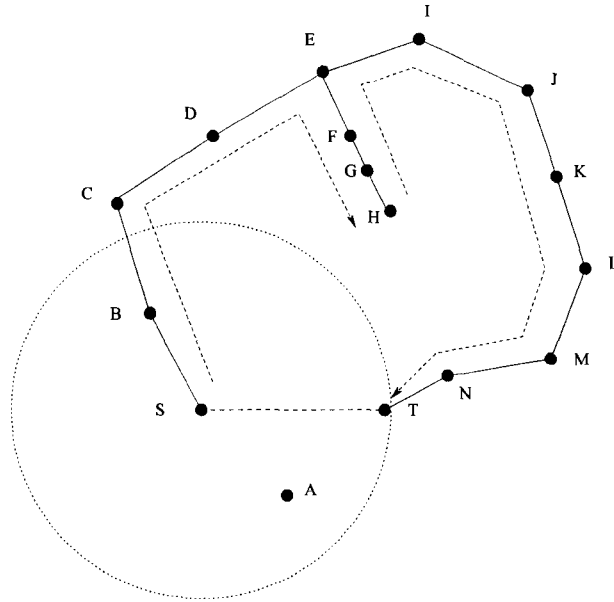


Figure 4.5: Face routing using LMWST

Chapter 5

Implementation

5.1 Implementation Details

The implementation includes two main parts. GLR algorithm is implemented first. As a complement to GLR, DGF is also implemented. Since these two algorithms are used at different stages, they are designed separately.

5.1.1 NS-2 and the Modification

We utilize the NS-2 network simulator [4], with the CMU Monarch Project wireless and mobile NS-2 extensions, to study the effects of employing GLR routing and DGF face routing.

5.1.1.1 Internet MANET Encapsulation Protocol

Internet MANET Encapsulation Protocol (IMEP) layer is used in the design of the proposed routing system. Location information is added to the IMEP packet for the quick and accurate neighborhood information exchange. The mechanisms contained in the IMEP are:

- Link/Connection Status Sensing
- Control Message Aggregation
- Broadcast Reliability
- Network-layer Address Resolution
- Security Authentication

The first three mechanisms are implemented in NS-2 and are used in the neighboring nodes information collection in the proposed routing protocol. Routing protocol could look up the neighborhood information by using the interface provided by the IMEP layer. There are four types of packets in the IMEP layer: namely, Beacon type, Ack type, Hello type and Object type. The IMEP message structure is shown in the form below:

```
<IMEP message> ::= <IMEP_VERSION> <BLOCK_FLAGS> <IMEP_LENGTH>
```

```
[Ack Block]
```

```
[Hello Block]
```

```
[Object Block]
```

The Beacon packet contains only the header, including the IMEP_VERSION, BLOCK_FLAGS and IMEP_LENGTH. By modifying the header format, each node could add its location information to the header and exchange this information with its neighbors in the process of Link/Connection Status Sensing.

5.1.2 Compute Delaunay Triangulation

The computation of Delaunay triangulation of the set of nodes is based upon the definition. Whenever no other node is inside the circle formed by three nodes, the triangle of these three nodes is part of the Delaunay triangulation.

5.1.2.1 Circumcircle

One important Delaunay triangulation computation is to decide whether a point is within a triangle's circumcircle. In two dimensions, one way to detect if point D lies in the circumcircle of point A, B and C is to evaluate the following determinant (A_x and A_y represent the x and y coordinates of point A, similar for B, C and D):

$$\begin{vmatrix} A_x & A_y & A_x^2 + A_y^2 & 1 \\ B_x & B_y & B_x^2 + B_y^2 & 1 \\ C_x & C_y & C_x^2 + C_y^2 & 1 \\ D_x & D_y & D_x^2 + D_y^2 & 1 \end{vmatrix}$$

If A, B and C lie in counter-clockwise order, the determinant is positive if and only if D lies in the circumcircle formed by A, B and C.

5.1.3 Localized Delaunay Triangulation

In Disruption tolerant network, any global algorithm should be modified into local for its proper functioning. After the calculation of Delaunay triangulation locally, nodes inside DTN should perform information exchange with neighboring nodes only to get desired planar spanner.

5.1.4 MaxDSTD, MinDSTD and MidDSTD

Multiple copies of same message should be sent from source to the direction of destination if nodes use short range radius and network is sparse to increase delivery probability and reduce delay. The three tree branch extraction is shown below.

After the construction of localized Delaunay triangulation and the planarization process, three trees are extracted from this graph, namely the max distance source to destination (MaxDSTD) tree, the min distance source to destination tree (MinDSTD) and the mid distance source to destination tree (MidDSTD).

To intelligently adjust the number of same messages in transit, source node could enable MinDSTD, MidDSTD or disable them whenever needed. Extra copies could also be added if network is really sparse.

The calculation of which neighbor is the next hop is described using Figure 5.1. Node A is the MaxDSTD next hop from the source node S to destination node T direction, node B is the MinDSTD next hop and node C is the MidDSTD next hop.

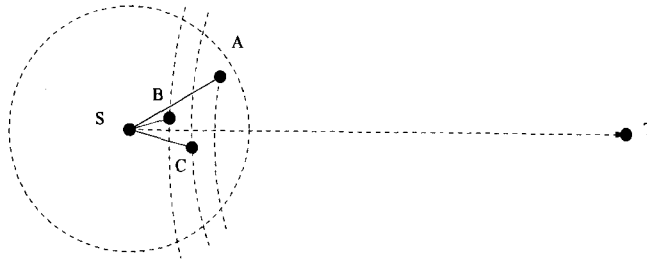


Figure 5.1: Next hop selection

5.1.5 Message Structure

If a message could not be delivered to the destination node directly during message transmission process, it is saved in node's storage area which is called *Store*. Some extra control information also needs to be saved together with message body. The message structure has the following form.

Message Structure
Source ID
Destination ID
Source Location
Destination Location
Message Count
Hop Count
Tree Branch Info
Time Stamp
Message Body

Each source node gives the message which it generates a message count, starting from 1 and increment by 1 after that. When identical copies of message reach destination, destination node distinguishes message by the *Source ID* and *Message Count* and discards the duplicate message. *HOP Count* field is used in face routing to deal with wrong face routing, i.e., face routing goes to the wrong direction. *Tree Branch Info* field indicates which branch this node should take to find the next hop when multiple copies approach is adopted and the *Time Stamp* field indicates the time when message was generated.

5.1.6 Information Exchange Flow

In dynamic disruption tolerant network, the construction of localized Delaunay triangulation is performed whenever message needs to be sent to the destination. The node with message initiates the message transmission process. Suppose node u is the node which has message in its $Store_u$, it starts the information exchange process. Algorithm 3 is the corresponding pseudo code.

First, u inquires neighboring nodes' information, including their locally constructed Delaunay triangulations and their neighbors. After receiving the inquiry from u , neighboring node, e.g., w calculates its own Delaunay triangulation LDT_u^w after collecting its own neighboring nodes ID set $N_1(w)$ and corresponding location information. The LDT_u^w and $N_1(w)$ are given to u for further planar construction. 1-LDT (Section 2.3.2.3) is the default in the implementation. In Algorithm 3, $N_1(w)$ should be replaced with $N_k(w)$ if k-LDT is the underlying routing graph.

The planar process is given in Algorithm 4.

Algorithm 3 Localized Delaunay Triangulation Construction

```
1: procedure INQUIRY ▷ For node u
2:   if  $Store_u \neq \emptyset$  then
3:     Broadcast("Ask for information from neighbors")
4:   end if
5: end procedure

6: procedure NEIGHBORINGREPLY( $u$ ) ▷  $\forall w \in N_1(u)$ 
7:   while Receiving inquiry do
8:     Collect neighboring nodes ID
9:     Calculate  $LDT_u^w$ 
10:    Send  $N_1(w)$  and  $LDT_u^w$  to node u
11:   end while
12: end procedure

13: procedure CONSTRUCTPLANARLDT ▷ For node u
14:   Collect neighboring nodes ID
15:   Calculate  $LDT_u$ 
16:   while Receiving reply from node w do
17:      $recvneighbors_u^w \leftarrow N_1(w) + w$ 
18:     PlanarProcess( $LDT_u^w, LDT_u, recvneighbors_u^w$ ) ▷ see Algorithm 4
19:   end while
20: end procedure
```

Algorithm 4 The Planar Process

```
1: procedure PLANARPROCESS( $LDT_u^w, LDT_u, recvneighbors_u^w$ )
2:   for all  $uv \in LDT_u$  do
3:     if  $(u \in recvneighbors_u^w) \cap (v \in recvneighbors_u^w) \cap (uv \notin LDT_u^w)$  then
4:        $LDT_u^- = \{uv\}$ 
5:     end if
6:   end for
7: end procedure
```

For each received localized Delaunay triangulation LDT_u^w from neighboring node w , node u checks every edge in its own localized Delaunay triangulation LDT_u . If the two end nodes of the checked edge are in the node set of the neighboring node w 's neighboring node set (include w itself) $recvneighbors_u^w$ and this edge is not in the received localized Delaunay triangulation LDT_u^w , u deletes this edge from its localized Delaunay triangulation LDT_u . When every node u calculates the resulting LDT_u according to Algorithm 3 and 4, the resulting graph is the planar LDT.

When LMWST is adopted as the underlying routing graph for the face routing logic, the above information exchange flow could also be applied on LMWST construction by replacing Delaunay triangulation algorithm with minimum spanning tree algorithm.

5.1.7 Location Diffusion

For the proper functioning of geometric routing, destination node's location accuracy is required in making routing decisions. Although it is assumed in the GLR that source node knows the destination location information at the beginning, it is possible that destination node moves far away from its initial location during the process of message delivery, especially in a network with long delays and disruptions.

To solve this issue, location information diffusion is applied in the process of message exchange. Nodes exchange their location information whenever two nodes coming into communication range with each other. The location information is recorded, together with the time stamp. Each node keeps a table of other nodes' location information together with their IDs and time stamps. Message holder adds destination location information

in the packet which is used to collect neighboring nodes' information. A neighboring node updates the destination location information if the message holder has more recent destination location than its own and notifies the message holder if it has more recent destination location than that of the message holder has.

For best location accuracy, location tables should be exchanged whenever two nodes meet each other. Since this will add extra overhead in the routing protocol, it is not adopted in GLR considering the above mechanism is enough for the proper function of geometric routing.

5.1.8 Methods of Extracting Three Trees

There are two methods of finding nodes which make progress to the destination before extracting trees. Both of them define the meaning of closer and choose the next node closer to the destination in its own way.

The simpler way is to choose a node that is close to the destination in distance (see Figure 5.2). If one of the neighboring node is closer to the destination than the current node itself and there exists an edge with these two nodes as endpoints after the planar process, this neighboring node is used to calculate the Max, Min and Mid next hop. In the figure, node A and node B is closer to node T (destination) than node S while node C is further away than node S (source node) according to this definition. The simpler way is used as default in the simulation.

Another way is to choose a node that is close to the destination in the source to destination direction. If a line is drawn across the neighboring node and this line is

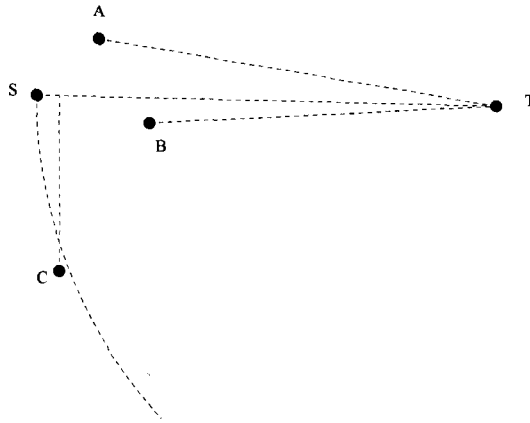


Figure 5.2: Distance to destination is smaller

perpendicular to the line drawn from source to destination, there is a intersection point of these two lines. A node is closer to destination means that the distance from the intersection point to the destination is shorter than the current node. Also an edge with the current node and the neighboring node as endpoints should exist after the planar process. Max, Min and Mid next hop are extracted after closer neighbors are collected. Also in Figure 5.2, node C is closer to the destination than node S according to this definition.

The distance from the destination node and one of the closer neighboring node is considered as Max when there is no other node which has shorter distance to the destination than this neighboring node. A closer node is considered as Min when there is no other node which has longer distance to the destination than this neighboring node. A closer node is considered as Mid when this neighboring node's distance to the destination is not Max nor Min (or Max if the number of neighboring nodes is less than three).

After three next hop neighboring nodes are selected, message is delivered using these

nodes accordingly. If none of these neighboring nodes are available, the current node saves the message in its *Store*, waits for a while and forwards the message later.

5.1.9 Custody Transfer

In the proposed routing solution, custody transfer is used to ensure that message would not be deleted by the sender, unless the corresponding receiver has notified the sender that it has received correctly. Two storage areas are designed to distinguish between the messages which have not been sent and the messages which have been sent and without receiving reply. The *Store* is the place where messages are waiting to be sent while messages that are just sent are saved in the *Cache*.

Whenever a node successfully receives a message, it will notify the sender that the message has been received correctly. This notification contains information regarding the source node, destination node, message count (the i^{th} message generated by the source node) and the extracted tree branch information (it is needed because messages in different tree branches follow different routing paths). After receiving reply from the receiver, sender will check the *Cache* and delete the corresponding message from it.

In the case that a message was lost during transfer or reply was not received properly, after staying in the *Cache* for specified time, the message is moved from *Cache* to *Store* for another round of transfer rescheduling, may or may not choose the same next hop this time when position, neighboring nodes and destination location change.

5.1.10 Face Routing Using Planar Graph

Before the DGF face routing process, planar graph is extracted from the original graph. Either LDT or LMWST is constructed using 1-local algorithm (Section 2.1.1.3), depending on the application's needs. Nodes make routing decisions locally.

When source node has message to send (see Figure 5.3), it chooses either left hand neighbor (the angle $\angle TSA$ counterclockwise is no more than 180°) or right hand neighbor otherwise. When node A is chosen as the left hand next hop, node A continues to use left hand rule to find the next hop, which is B. As line segment AB crosses with line segment ST, all A's neighbors are calculated to see if they also cross ST. The neighboring node which crosses with ST and with the cross point which is the closest to destination node is chosen as the next hop instead of B. In Figure 5.3, line segment AC crosses with ST at X and $|XT|$ is the smallest. When node C is chosen as the next hop, it uses a different rule than its previous node (if A use left hand rule, then C use right hand rule and vice visa). Also for the purpose of guaranteed delivery, node A sends another message copy to node D using left hand rule. The process continues until message reaches the destination.

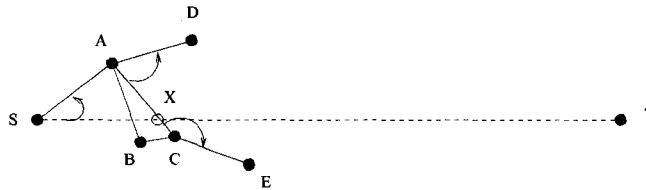


Figure 5.3: Face routing demonstration

Chapter 6

Experimentation Results/Analysis

6.1 Simulation Environment

The proposed Geometric Localized Routing is implemented using the NS-2 [4] simulator. This simulation environment includes full simulation of the IEEE 802.11 physical and MAC layers, which makes the simulation better reflect the real world. NS-2 simulator also models node mobility, providing a frequently changing network topology for experimentation on DTN routing protocols. The motion pattern follows the random waypoint model: a node chooses a random destination within the simulated region, chooses a random velocity within the configured range and then moves to the destination at the chosen velocity. Upon reaching the destination, it pauses for a configured time and then repeats the process. For the propagation model, we choose *Two Ray Ground* which considers both the direct path and a ground reflection path.

In the simulation, we choose parameters according to Table 6.1.

Parameter	Value
Number of mobile nodes	50
Mobility	0-20m/s(uniform distribution)
Transmission range	50-250m
Data rate	1 Mbps
Propagation model	<i>Two Ray Ground</i>
Simulation time	3800-10000 seconds
Link layer queue length	150
Topology size	1500m ×300m
Pause time	0 seconds
Packet payload size	1000 bytes
Antenna model	OmniAntenna

Table 6.1: Parameters of the simulations.

During the simulation, various scenarios are adopted. When comparing with Epidemic routing [51], a subset of the 50 nodes act as sources and destinations, with each of 45 nodes sending packets to other 44 nodes. Packets are generated every second.

In our simulations, we make the following assumptions:

- Source knows the true destination location.
- Nodes have synchronized clock.

For the simulation results, all points in the figures, as well as numbers in all the tables are obtained as an average of 10 different runs with 10 different network topologies and movement patterns. The confidence intervals for numbers are calculated at 90% confidence level.

The following phases are implemented for all the experiments.

Initial Phase At the beginning, node gathers distance two neighborhood information and construct localized Delaunay triangulation. When node radius is larger than 100

meters, only MaxDSTD tree is used. Otherwise, three trees are extracted when source node has packet to send. For relay nodes, the source and destination node location information is obtained through the header of the packet. One of the three trees from source to destination is used when one of the three paths from the source to destination going through this node exists.

Location Diffusion Phase In the implementation, each mobile node has a geometric routing agent layered on top of the Internet MANET Encapsulation Protocol (IMEP) layer inside the NS-2 simulator, which notifies changes of the node's neighborhood information to the geometric routing agent. During message delivery process, location information among neighboring nodes is also exchanged by adding node location and node ID to the packet.

6.2 Simulation Results and Analysis

During the simulation, fast delivery is achieved by using the proposed routing protocol to meet both partition and moving characteristics of DTN. Since we intelligently decide the number of message copies which are needed, i.e., use three spanning trees in the packet delivery process when node radius is short to increase delivery opportunity and use single copy when node radius is long to reduce contention, message delivery delays are reduced in both situations.

Since DTNs will unavoidably introduce network partition, some nodes will enter into store and forward state during the routing process. The store and forward mechanism

is implemented during the simulation. Simulation results show that we achieve desirable storage utilization.

As face routing is needed to solve the local minimum, in the simulation, Delivery Guaranteed Face Routing (DGF) technique is implemented on both the LMWST and LDT underlying graphs. Delivery hop counts and energy consumption of *Two Ray Ground* model are compared between the two different local graphs.

In our protocol, nodes do not keep the packets when they forward these packets to other nodes. Custody transfer ensures that packets are not lost during transmission. Delivery ratio under various simulation parameters are gathered and the correctness of this protocol is proved.

6.2.1 Store and Forward Mechanism

When delivery route is not available at the time of message sending, node will store the message in its storage area, waiting for a while and trying to see if possible routes appear later. The node which has messages to send needs the destination node location as well as its neighboring nodes location information to make routing decisions. Neighboring nodes location information is obtained by asking, and waiting for all the replies coming in. In this process, new messages may be received, but only could be stored at the end of the message queue, waiting for its turn to be transmitted. When there are messages in store, route availability is checked periodically. The check interval affects the delivery latency also, although not too much. The more frequent route check is performed, the more control messages are in transit, but delivery latency is reduced in general. Figure

6.1 describes this tradeoff. Simulations are performed at 1980 messages and 100 meter scenario.

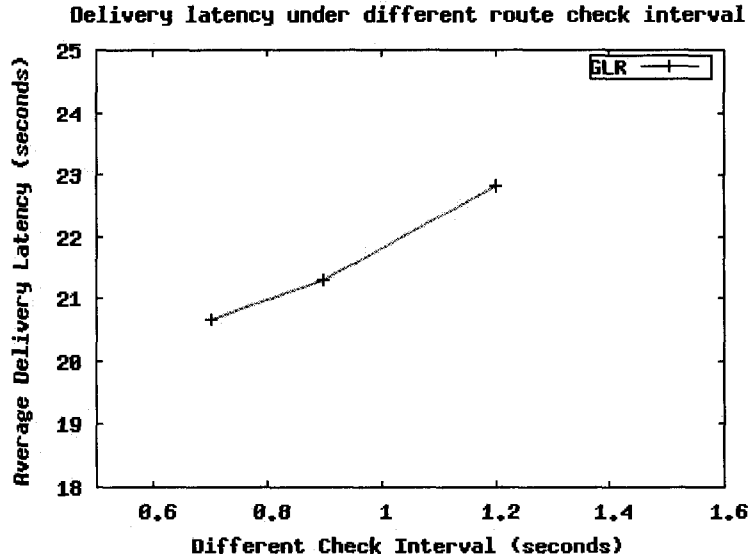


Figure 6.1: Delivery latency comparison (different route check interval)

The following simulations use 0.9 second as the default check interval.

6.2.2 Location update

In the geometric routing, destination node's location information is important in making routing decisions. Delivery latency differs greatly when the location availability is different, especially in the DTNs when destination nodes could move far away from their original places when the messages destined for them were generated. Four different situations are considered in the simulation. The first one assumes all nodes in the path from source to the destination node know exactly the destination location. The second and third assume only the source node knows the destination node location and includes the x

and y coordinates of the destination node in the messages. The last situation assumes no node knows the destination location information well in advance. The simulation results are presented in Table 6.2.

Number of Copy	Destination Location	Delivery Rate	Average Latency (seconds)	Average Hops	Storage (number of messages)
1 copy	All nodes know	100%	120.2±8.5	14.9±0.3	38.3±1.4
3 copies	Only source knows	100%	149.7±9.6	17.3±0.4	43.6±1.4
1 copy	Only source knows	100%	156.1±11.2	18±0.3	40.3±2
3 copies	No nodes know	99.9%	212.4±16.6	23.1±0.5	50.9±3.8

Table 6.2: Message delivery results under location information availability

Location information is exchanged locally throughout the network. Whenever two nodes come into communication range of each other, they exchange their location information. In the message delivery process, the destination location is updated when the message owner knows more recent destination location information than its own.

The above results show that three copies approach with source knows destination location is slower than the one copy approach with all nodes know destination location. But it performs better than the one copy approach with source knows destination location. This reflects the fact that the controlled flooding really reduces latency.

Although source node knows nothing about the destination node location in the last situation, relay nodes could adjust the destination location in the process of delivery (random location is given at the beginning). Because of unknown destination location, delivery latency of the last approach is the longest and not all messages are delivered to

the destination nodes within specified time frame (3800 seconds in the simulation) as a result.

The impact of location inaccuracy and solution In the process of delivery, the destination node could move away from the original place where it stayed when the message destined to it was generated. When the message reaches a node that is closest to a stale destination location which is contained in the packet header, no neighboring node could be selected as the next relay because this node is the closest.

To avoid longer delays, a new value is assigned to the destination location so that the node which is closest to the wrong location could deliver it out to another node to increase the delivery probability.

6.2.3 Delivery Latency Comparison

Comparison under 50m Radius Although Ramanathan *et al.* call epidemic routing protocol as “unbeatable” [41], simulation results show it is beatable in practice. In epidemic routing, all nodes flood all the messages they hold to all their neighbors if possible. This approach costs significant storage. It looks like this protocol delivers messages faster than all other possible approaches at the beginning. Our routing protocol outperforms this epidemic one in delivery latency when messages in transit increase as shown in Figure 6.2.

The increased contention may be the reason why epidemic routing slows down when messages increase.

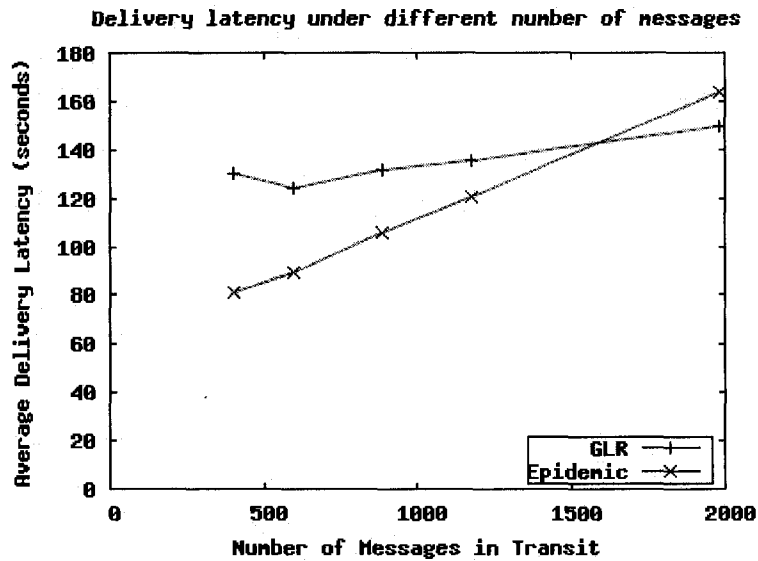


Figure 6.2: Delivery latency comparison (50m radius)

Comparison under 100m Radius In 100 meter radius, the delivery latency of our routing protocol also outperforms epidemic routing. Figure 6.3 shows the result.

Comparison under different Radius Under different radius, the delivery latency of our routing protocol also outperforms epidemic routing in 1980 messages scenario. Figure 6.4 shows the result.

For the radius 150, 200 and 250 meters scenario, single copy approach is used. The use of more copies in these situations increases contention severely and leads to longer delay.

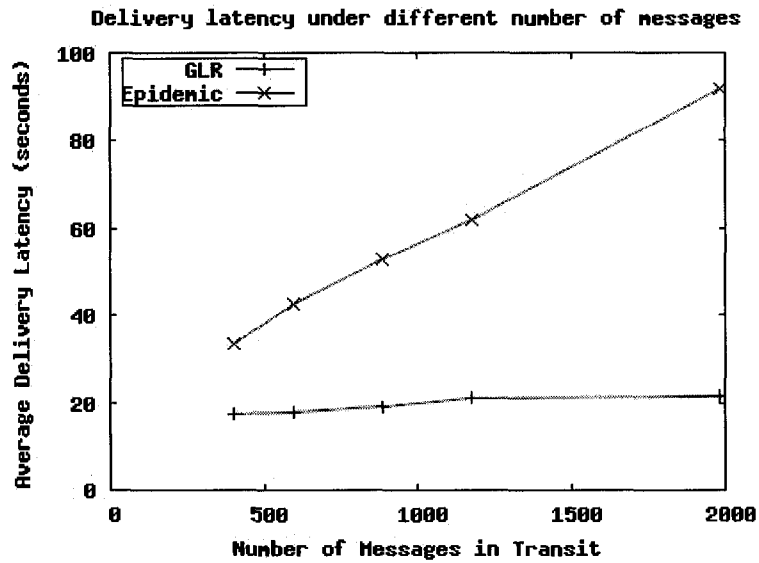


Figure 6.3: Delivery latency comparison (100m radius)

6.2.4 Delivery Ratio Comparison

The proposed DTN routing protocol achieves 100% delivery ratio, the same as the epidemic one when storage is unlimited. When storage drops below 200 in the 1980 messages in transit scenario, the delivery ratio of epidemic routing begins to drop, while the proposed approach still keeps 100% delivery ratio even when storage drops to 100 per node.

Figure 6.5 is the simulation results of the delivery ratio comparison between the proposed solution and epidemic routing.

In the epidemic routing, messages will never be dropped when nodes are assumed to have unlimited buffer space. The reason for keeping all messages is because nodes do not know whether a message has been delivered to the destination. When storage is limited

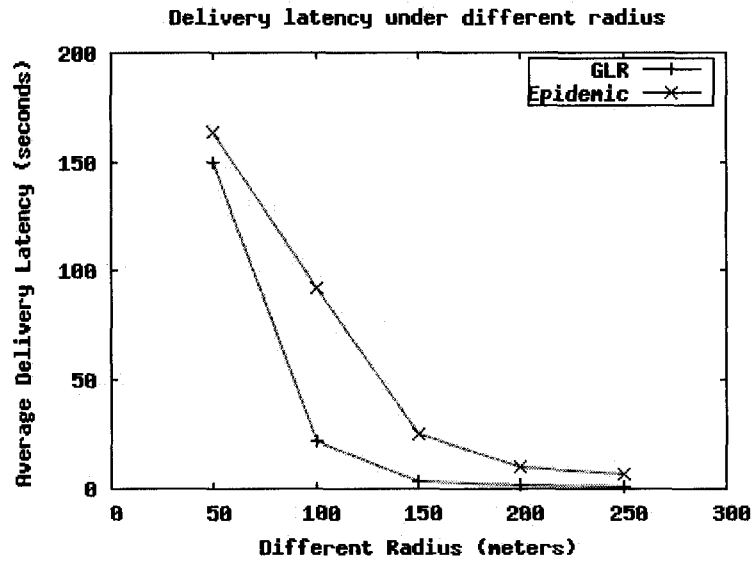


Figure 6.4: Delivery latency comparison (various radius)

and the storage space is fully occupied, old messages are dropped when new messages come in. In the proposed solution, message is deleted from storage area when it has been delivered to the next hop node and acknowledgement is received from that node. Because flooding is controlled in the proposed solution, less storage space is needed in total. Under limited storage situation, less messages are dropped in our solution compared with epidemic routing, as shown in Figure 6.5.

6.2.5 Delivery with Custody Transfer

Without custody transfer, messages could be delivered with high probability but without guarantee. With custody transfer, message is discarded only when message sender receives the acknowledgement from the receiver. No message is lost in this situation. Table 6.3 gives out a short description of the delivery ratio comparison.

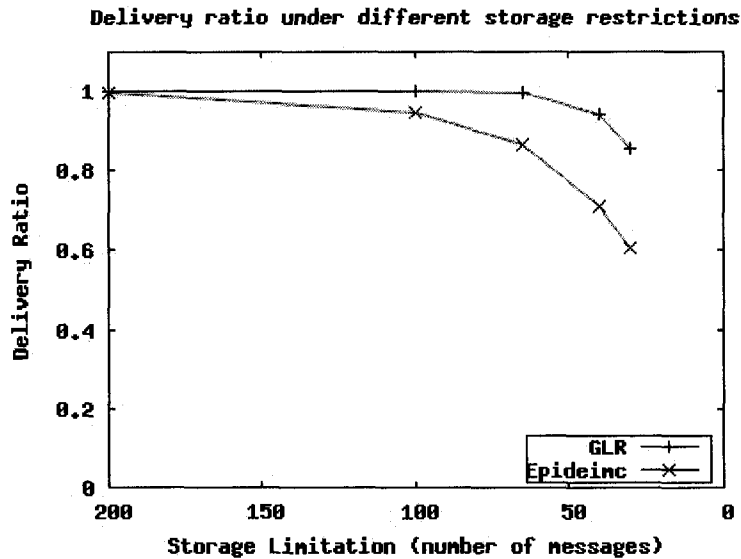


Figure 6.5: Delivery ratio comparison (50m radius)

Simulation Scenario (Number of Messages)	890 messages
Delivery ratio without custody	86.1%±1%
Delivery ratio with custody	100%

Table 6.3: Message delivery ratio comparison (50m)

The delivery ratio would be changed at different scenarios. But one thing is clear, hardly could all messages be delivered without custody transfer because of contention or node movement.

6.2.6 Hop Count Comparison

In the proposed routing protocol, every node which has messages in its storage checks regularly to see if there are any other nodes which are closer to the destination. In the epidemic routing, nodes exchange messages only when two nodes come into communication range of each other and no message is exchanged even if relative positions changed.

As a result, the geometric routing protocol travels along more hops than the epidemic routing. Table 6.4 shows the simulation results.

Simulation Scenario	250m	200m	150m	100m	50m
GLR routing	3.4±0.04	4.1±0.05	5.23±0.13	8.75±0.13	17.32±0.4
Epidemic routing	3.19±0.14	3.64±0.07	4.58±0.07	4.92±0.06	3.92±0.05

Table 6.4: Hop counts (1980 messages, for geometric routing, 3 copies for 50m/100m and 1 copy for 150m/200m/250m)

6.2.7 Storage Requirements

Storage is required in the process of delivery when node is busy or delivery path to the destination is not available. When the number of messages in transit increases, the number of average storage also increases. It is also true that the longer the radius, the smaller the storage size. This could be verified through the following tables.

Number of messages	400	600	890	1180	1980
max peak storage (number of messages)	39±4.67	43.9±3.38	49.1±2.97	59.9±7.17	69±5.82
average peak storage (number of messages)	21.31±0.59	25.77±1.05	30.2±1.23	37.28±2.82	43.64±1.42

Table 6.5: Storage requirement, different message numbers (50m, 3copies)

Simulation Scenario	250m	200m	150m	100m	50m
max peak storage (number of messages)	6.9±4.29	14.3±4.81	24.3±4.54	48.4±6.52	69±5.82
average peak storage (number of messages)	1.76±0.72	3.28±1.06	8.36±0.95	25.82±1.37	43.64±1.42

Table 6.6: Storage requirement, different radius (1980 messages, 3 copies for 50m/100m and 1 copy for 150m/200m/250m)

6.2.8 Face Routing Delay and Hop Comparison

Face routing under local minimum weight spanning tree and local Delaunay triangulation are implemented. Simulation results under randomly generated scenario files show that face routing using LDT outperforms LMWST in both the delivery delay and the number of hop count. Delivery ratios using the proposed face routing strategy under both graphs are 100% guaranteed. Table 6.7, 6.8 and 6.9 are the face routing comparisons between LDT and LMWST.

Simulation Scenario	delay (seconds)	hop count
LDT	16.59±1.19*	5.11±0.26
LMWST	72.51±8.82*	18.8±2.12

Table 6.7: Message delivery comparison (250m radius, 150 messages, * same condition but not optimal)

Simulation Scenario	delay (seconds)	hop count
LDT	16.13±0.57*	5.28±0.14
LMWST	67.17±6.57*	18.48±1.58

Table 6.8: Message delivery comparison (250m radius, 300 messages, * same condition but not optimal)

Simulation Scenario	delay (seconds)	hop count
LDT	16.63±0.68*	5.28±0.1
LMWST	67.71±5.65*	18.14±1.37

Table 6.9: Message delivery comparison (250m radius, 600 messages, * same condition but not optimal)

6.2.9 Face Routing Energy Comparison

Although face routing delivery latency using LMWST is larger than the delivery latency of using LDT, the energy consumption using LMWST is less than that of the LDT.

We calculate the sum of path distance's fourth power, which is shown in Table 6.10 below.

Simulation Scenario	average sum of d^4 (per message)
LDT	$(2.96219 \pm 0.4)e+09$
LMWST	$(1.36382 \pm 0.36)e+09$

Table 6.10: Path comparison (250m radius, 600 messages, d is the distance between two nodes in message transmission path)

It is known that in the *Two Ray Ground* propagation model, the relation between the power used to transmit packets, P_t and the received power P_r can be characterized as

$$P_r = \frac{P_t G_t G_r h_t^2 h_r^2}{d^4 L} \quad (6.1)$$

where G_t is the antenna gain of the transmitter, G_r is the antenna gain of the receiver, λ is the wave length, d is the distance between the antenna of the transmitter and that of the receiver, and L is the system loss, h_t is the antenna height of the transmitter and h_r is the antenna height of the receiver.

According to Equation 6.1 and Table 6.10 it is clear that the minimum energy consumption for message transmission in LMWST is less than the minimum energy consumption in LDT if the propagation model is *Two Ray Ground*, only accounts for 46% energy consumption of LDT if radius is adjusted properly after knowing the distance between two neighboring nodes.

6.2.10 The Need for Planar Accuracy

During simulation, it is noticed that some messages could not be delivered to their destinations properly when neighboring nodes information are not collected accurately, even if paths exist. This is due to the reason that not all neighboring nodes reply to the node which holds message during LMWST or LDT construction process. In the next hop selection process that follows, wrong decision is made as a result. In Figure 6.6, when message travels from source node S to node B by the right hand rule, node B would collect neighboring nodes' information to construct local planar graph. If node C would not reply properly due to some reason, message is sent back to node A and wrong decision is made. So planar accuracy is important to face routing. We use enhanced neighboring information collection process which checks neighboring information several times in the implementation and delivery is guaranteed.

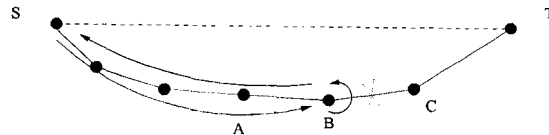


Figure 6.6: The need for planar accuracy

6.3 Delay Tolerance Analysis

6.3.1 Previous Methods to Alleviate Delay

Since network disruption and communication delay in DTN networks are unavoidable, it is better to understand the characteristic of corresponding communication system before

the routing design of the disruption (delay) tolerant network. In satellite communication, the satellite is reachable only every 3 hours for 15 minutes. Space communication within our solar system could introduce speed-of-light delays up to tens of minutes. Scheduled contacts could be arranged to achieve optimal intermittent communication results in these situations.

Several projects [39, 9, 25, 38, 43] aim at providing Internet service for small villages. Data is transferred from the Internet in the cities to Kiosks installed in small villages using mobile vehicles. The disconnection between the vehicles and access points in the cities is on the order of 75 seconds to hours. Kiosk sends request to vehicle when the vehicle passing by and vehicle brings data to Kiosk when it drives back. Delivery delay is a round trip time. The main work of these projects is on the efficiency of applications.

To achieve fast delivery, epidemic routing uses wide spread messages flooding within the whole network. This approach established a desirable benchmark for other protocols in terms of delivery rate and delay, but the protocol itself is not practical.

6.3.2 Our Proposal to Reduce Delay

Intelligence is our first step towards fast delivery. For highly connected network, single copy approach is adopted to avoid contention and reduce delay. GLR quickly delivers packets along the planar tree while the unnecessary packets flooding is avoided. Just as the 150m/200m/250m radius situations in the simulation, single copy guarantees fast message delivery. Identical multiple message copies are sent into network to increase delivery opportunity and reduce delay when network is sparse. The number of copies

is decided by intelligence according to network density and node communication range. In particular, our approach uses three spanning trees extracted from localized Delaunay triangulation. The direction of the three localized trees is from the source to destination. These three trees send packets from source to the vicinity of the destination using three different paths. With node mobility, one of these paths would reach the destination with high probability than only one path being utilized.

Although our simulation uses only three copies approach, it is possible to extract more trees out of the original graph when node radius is very short. Thus the whole network delivery delay is minimized and this approach works better than other approaches with high probability.

When node enters a local minimum, the proposed DGF face routing technique reduces transmission delay by using multiple copies upon an edge which intersects with source node to destination node line segment. The DGF routing also guarantees delivery if path exists.

Chapter 7

Conclusion & Future Work

7.1 Future Directions

Further improvements on DTN routing protocol “attributes” like stretch factor, delivery probability and energy consumption are still needed in future protocol research to push forward the study in this area. The velocity and direction of nodes’ movement could be exchanged to improve the accuracy of nodes’ location prediction in the geometric greedy routing process.

UDG assumes that the radius of all the nodes are the same. Sometimes it would be more efficient when some nodes would use different radius. In DTNs, the need to use longer radius at some time seems necessary when the delivery of messages are urgent. To save energy, clusters should be formed together and clusterheads with longer radius should be responsible for communicating among clusters. Increased radius with clusters would be an interesting research topic.

Researches in disruption (delay) tolerant network usually assume all nodes cooperate in a nice way. But it is true that network participants are heterogeneous. There exist not only non-cooperation nodes, but also malicious nodes. It is critical for some applications to perform in a safe, secure and guaranteed data transmission way even if various situations appear. With the disruption and delay characteristics of the network, traditional ad hoc network security is not sufficient. But current security study in MANETs could work as a reference for similar research in DTNs. Security would be an important aspect of the future research.

Internet is always an interesting resource and the connection between Internet and DTNs seems unavoidable. How to effectively connect DTN with Internet should be thoroughly studied.

The increase of packet copies into the network could be utilized in a more efficient way. Not only source node could send to destination node, but also some destination node could query the source content. The mechanism that is widely used in peer-to-peer system could be utilized into the DTNs.

Another important aspect about disruption tolerant network is that some specifications and formal program framework have been developed by DTNRG and DARPA DTN research group respectively. The incorporation of bundle layer into DTN is one of them. Many control information in the messages are used in the bundle layer. It would be beneficial if research would further take these formal specifications and framework into consideration.

7.2 Conclusions

Disruption (delay) tolerant network is gaining more attention and rapidly becoming an interesting research direction in wireless mobile networks. Routing in such network has been carefully studied in the last couple of years. However, there has been no routing protocol which achieves desirable delay and better resource consumption. Partly connected local planar spanners have never been studied in this area.

We proposed a novel mechanism called Geometric Localized Routing (GLR). GLR uses local neighborhood location information to construct localized planar spanners. Then geographical routing is used to greedily deliver message to the node closer to the destination. Spanning trees are employed to achieve better delay tolerance. We use store and forward way to deliver message upon partition. Delivery Guaranteed Face (DGF) routing strategy is designed to guarantee delivery when local minimum exists. Our simulation results show that GLR outperforms epidemic routing in randomly generated network.

In the thesis, we first provide an overview of underlying routing graphs that could be constructed and maintained locally. After that we talk about the routing on top of the geometric graph. We propose intelligent greedy routing with controlled flooding technique which effectively uses an underlying routing graph with good stretch factor and achieves high delivery ratio. Further simulation work proves the correctness of these mechanisms. We also discuss the future research directions in DTNs.

Our work is unique in a number of ways. First, our Geometric Localized Routing is the first to apply localized Delaunay triangulation in DTNs. Second, unlike other approaches,

it works intelligently to some extent to be able to work efficiently in various scenarios.

Finally, our implementation achieves desirable delivery attributes in a practical way.

As Zhang [54] pointed out in his survey, wireless networks with high mobility and short radio range will become a common phenomenon. Before the benefits of DTNs could be fully utilized, it is imperative that key issues be fully understood and studied. And it is expected that more research interests would arise upon deeper study in this field.

Bibliography

- [1] DTNRG Website, <http://www.dtnrg.org>, Accessed July 8, 2008.
- [2] E. Kranakis's Website, <http://www.carleton.ca/~kranakis>, Accessed July 10, 2008.
- [3] IETF-65 DTNRG group meeting proceedings, Dallas, TX, USA, 2006, <http://www.ietf.org/proceedings/06mar/DTNRG.html>, Accessed June 15, 2008.
- [4] The Network Simulator, NS-2, <http://www.isi.edu/nsnam/ns/>, Accessed July 8, 2008.
- [5] C. Becker and G. Schiele, "New mechanisms for routing in ad hoc networks through world models," *4th Plenary Workshop Pisa, Italy*, 2001.
- [6] G. Blakley, "Safeguarding cryptographic keys," *Proceedings of AFIPS 1979 National Computer Conference*, vol. 48, no. 313-317, p. 390, 1979.
- [7] P. Bose, L. Devroye, W. Evans, and D. Kirkpatrick, "On the spanning ratio of gabriel graphs and beta-skeletons," *Proceedings of the Latin American Theoretical Infocomatics (LATIN)*, pp. 479–493, 2002.

- [8] P. Bose, P. Morin, I. Stojmenović, and J. Urrutia, “Routing with Guaranteed Delivery in Ad Hoc Wireless Networks,” *Wireless Networks*, vol. 7, no. 6, pp. 609–616, 2001.
- [9] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, “MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks,” *Proc. IEEE INFOCOM, April*, 2006.
- [10] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss, “Delay-tolerant networking: an approach to interplanetary Internet,” *Communications Magazine, IEEE*, vol. 41, no. 6, pp. 128–136, 2003.
- [11] B. Burns, O. Brock, and B. Levine, “MV routing and capacity building in disruption tolerant networks,” *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 1, 2005.
- [12] E. Chavez, S. Dobrev, E. Kranakis, J. Opatrny, L. Stacho, and J. Urrutia, “Local construction of planar spanners in unit disk graphs with irregular transmission ranges,” *Lecture Notes in Computer Science*, vol. 3887, pp. 286–297, 2006.
- [13] J. Davis, A. Fagg, and B. Levine, “Wearable computers as packet transport mechanisms in highly-partitioned ad-hoc networks,” *International Symposium on Wearable Computing*, pp. 141–148, 2001.
- [14] M. Demmer and K. Fall, “DTLSR: delay tolerant routing for developing regions,” *Proceedings of the 2007 workshop on Networked systems for developing regions*, 2007.

- [15] D. Dobkin, S. Friedman, and K. Supowit, “Delaunay graphs are almost as good as complete graphs,” *Discrete and Computational Geometry*, vol. 5, no. 1, pp. 399–407, 1990.
- [16] S. Farrell and V. Cahill, *Delay-and Disruption-Tolerant Networking*. Artech House, Inc. Norwood, MA, USA, 2006.
- [17] S. Farrell, V. Cahill, D. Geraghty, I. Humphreys, and P. McDonald, “When TCP Breaks: Delay-and Disruption-Tolerant Networking,” *IEEE INTERNET COMPUTING*, pp. 72–78, 2006.
- [18] L. Ford and D. Fulkerson, *Flows in networks*. Princeton University Press Princeton, NJ, 1962.
- [19] H. Frey and I. Stojmenovic, “On delivery guarantees of face and combined greedy-face routing in ad hoc and sensor networks,” *Proceedings of the 12th annual international conference on Mobile computing and networking*, pp. 390–401, 2006.
- [20] J. Gao, L. Guibas, J. Hershberger, L. Zhang, and A. Zhu, “Discrete Mobile Centers,” *Discrete and Computational Geometry*, vol. 30, no. 1, pp. 45–63, 2003.
- [21] J. Gao, L. Guibas, J. Hershberger, L. Zhang, and A. Zhu, “Geometric spanners for routing in mobile networks,” *Selected Areas in Communications, IEEE Journal on*, vol. 23, no. 1, pp. 174–185, 2005.

- [22] C. Georgiou, E. Kranakis, R. Marcelín-Jiménez, S. Rajsbaum, and J. Urrutia, “Distributed Dynamic Storage in Wireless Networks,” *International Journal of Distributed Sensor Networks*, vol. 1, no. 3, pp. 355–371, 2005.
- [23] R. Handorean, C. Gill, and G. Roman, “Accommodating transient connectivity in ad hoc and mobile settings,” *Lecture Notes in Computer Science*, vol. 3001, pp. 305–322, 2004.
- [24] K. Harras and K. Almeroth, “Transport layer issues in delay tolerant mobile networks,” *Proceedings of IFIP-TC6 Networking, May, 2006*.
- [25] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, “CarTel: a distributed mobile sensor computing system,” *Proceedings of the 4th international conference on Embedded networked sensor systems*, pp. 125–138, 2006.
- [26] S. Jain, M. Demmer, R. Patra, and K. Fall, “Using redundancy to cope with failures in a delay tolerant network,” *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 109–120, 2005.
- [27] E. Jones, L. Li, J. Schmidtke, and P. Ward, “Practical Routing in Delay-Tolerant Networks,” *IEEE TRANSACTIONS ON MOBILE COMPUTING*, pp. 943–959, 2007.

- [28] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein, “Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet,” *ACM SIGPLAN Notices*, vol. 37, no. 10, pp. 96–107, 2002.
- [29] B. Karp and H. Kung, “GPSR: greedy perimeter stateless routing for wireless networks,” *Proceedings of the 6th annual international conference on Mobile computing and networking*, pp. 243–254, 2000.
- [30] J. Keil and C. Gutwin, “The delaunay triangulation closely approximates the complete euclidean graph,” *Proc. 1st Workshop Algorithms Data Structure (LNCS 382)*, 1989.
- [31] E. Kranakis, H. Singh, and J. Urrutia, “Compass routing on geometric networks,” *Proc. 11 th Canadian Conference on Computational Geometry*, pp. 51–54, 1999.
- [32] B. Leong, S. Mitra, and B. Liskov, “Path Vector Face Routing: Geographic Routing with Local Face Information,” *Network Protocols, 2005. ICNP 2005. 13th IEEE International Conference on*, pp. 147–158, 2005.
- [33] N. Li, J. Hou, and L. Sha, “Design and analysis of an MST-based topology control algorithm,” *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, vol. 3, 2003.
- [34] X. Li, G. Calinescu, and P. Wan, “Distributed construction of a planar spanner and routing for ad hoc wireless networks,” *INFOCOM 2002. Twenty-First Annual*

Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, vol. 3, 2002.

- [35] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 19–20, 2003.
- [36] P. Marshall, "The disruption tolerant networking program," 2005. <http://www.darpa.mil/sto/solicitations/DTN/briefs.htm>, Accessed July 8, 2008.
- [37] S. Merugu, M. Ammar, and E. Zegura, "Routing in Space and Time in Networks with Predictable Mobility," *College of Computing Technical Reports, Georgia Institute of Technology*, at <http://smartech.gatech.edu/handle/1853/6492>, 2004. Accessed July 25, 2008.
- [38] J. Ott and D. Kutscher, "Drive-thru Internet: IEEE 802.11 b for" automobile" users," *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 1, 2004.
- [39] A. Pentland, R. Fletcher, and A. Hasson, "DakNet: Rethinking Connectivity in Developing Nations," *Computer*, vol. 37, no. 1, pp. 78–83, 2004.
- [40] R. Prim, "Shortest connection networks and some generalizations," *Bell System Technical Journal*, vol. 36, no. 6, pp. 1389–1401, 1957.

- [41] R. Ramanathan, P. Basu, and R. Krishnan, "Towards a formalism for routing in challenged networks," *Proceedings of the second workshop on Challenged networks CHANTS*, pp. 3–10, 2007.
- [42] A. Savvides, C. Han, and M. Strivastava, "Dynamic fine-grained localization in Ad-Hoc networks of sensors," *Proceedings of the 7th annual international conference on Mobile computing and networking*, pp. 166–179, 2001.
- [43] A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav, "Low-cost communication for rural internet kiosks using mechanical backhaul," *Proceedings of the 12th annual international conference on Mobile computing and networking*, pp. 334–345, 2006.
- [44] A. Shamir, "How to share a secret," *Communications, ACM*, vol. 22, pp. 612–613, 1979.
- [45] C. Shen, G. Borkar, S. Rajagopalan, and C. Jaikaeo, "Interrogation-based relay routing for ad hoc satellite networks," *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*, vol. 3, 2002.
- [46] T. Small and Z. Haas, "Resource and performance tradeoffs in delay-tolerant wireless networks," *Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 260–267, 2005.
- [47] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Single-copy routing in intermittently connected mobile networks," *Sensor and Ad Hoc Communications and Net-*

works, 2004. *IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, pp. 235–244, 2004.

- [48] T. Spyropoulos, K. Psounis, and C. Raghavendra, “Spray and wait: an efficient routing scheme for intermittently connected mobile networks,” *Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 252–259, 2005.
- [49] K. Tan, Q. Zhang, and W. Zhu, “Shortest path routing in partially connected ad hoc networks,” *Global Telecommunications Conference, 2003. GLOBECOM’03. IEEE*, vol. 2, 2003.
- [50] F. Tchakountio and R. Ramanathan, “Tracking Highly Mobile Endpoints,” *Proceedings of the 4th ACM international workshop on Wireless mobile multimedia*, pp. 83–94, 2001.
- [51] A. Vahdat and D. Becker, “Epidemic routing for partially connected ad hoc networks,” *Duke University*, at <http://www.cs.duke.edu/~vahdat/ps/epidemic.pdf>, 2000. Accessed July 25, 2008.
- [52] Y. Wang, S. Jain, M. Martonosi, and K. Fall, “Erasure-coding based routing for opportunistic networks,” *Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 229–236, 2005.

- [53] F. Warthman, “Delay-Tolerant Networks (DTNs): A Tutorial,” *Interplanetary Internet Special Interest Group*, at http://www.ipnsig.org/reports/DTN_Tutorial11.pdf, May 2003. Accessed July 17, 2008.
- [54] Z. Zhang, “Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges,” *Communications Surveys & Tutorials*, *IEEE*, vol. 8, no. 1, pp. 24–37, 2006.
- [55] W. Zhao, M. Ammar, and E. Zegura, “A message ferrying approach for data delivery in sparse mobile ad hoc networks,” *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pp. 187–198, 2004.