

Matthias Witt

# Robust and Low-Communication Geographic Routing for Wireless Ad Hoc Networks

 Cuvillier Verlag Göttingen

# **Robust and Low-Communication Geographic Routing for Wireless Ad Hoc Networks**

Vom Promotionsausschuss der  
Technischen Universität Hamburg-Harburg  
zur Erlangung des akademischen Grades  
Doktor der Naturwissenschaften (Dr. rer. nat.)  
genehmigte Dissertation

von  
Matthias Witt

aus  
Reinbek

2008

## **Bibliografische Information der Deutschen Nationalbibliothek**

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

1. Aufl. - Göttingen : Cuvillier, 2008

Zugl.: (TU) Hamburg-Harburg, Univ., Diss., 2008

978-3-86727-733-4

Gutachter:

Prof. Dr. Volker Turau

Prof. Dr. Hermann Rohling

Tag der mündlichen Prüfung: 25.08.2008

© CUVILLIER VERLAG, Göttingen 2008

Nonnenstieg 8, 37075 Göttingen

Telefon: 0551-54724-0

Telefax: 0551-54724-21

[www.cuvillier.de](http://www.cuvillier.de)

Alle Rechte vorbehalten. Ohne ausdrückliche Genehmigung des Verlages ist es nicht gestattet, das Buch oder Teile daraus auf fotomechanischem Weg (Fotokopie, Mikrokopie) zu vervielfältigen.

1. Auflage, 2008

Gedruckt auf säurefreiem Papier

978-3-86727-733-4

## Acknowledgments

This dissertation is the result of four years of research in the field of wireless ad hoc networks at the Institute of Telematics, Hamburg University of Technology. I would like to thank all members of the Institute for supporting my work. My advisor, Prof. Dr. Volker Turau, provided an excellent supervision and has always been supportive of following high scientific standards and improving the quality of this dissertation. I am still amazed at his patience to read preliminary versions of this thesis over and over again, leaving tons of paper waste.

Special thanks go also to Christoph Weyer for many fruitful discussions and valuable ideas. Thanks to my brother Stefan and to Bernd Hauck for reading the manuscript and giving plenty of useful hints. Finally, I would like to thank my family for always supporting me. Without them, this work would not have been possible.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Ad Hoc Networks</b>	<b>5</b>
2.1	History of Ad Hoc Networks . . . . .	5
2.2	Characteristics of Wireless Ad Hoc Networks . . . . .	6
2.3	Routing in Wireless Ad Hoc Networks . . . . .	8
2.4	Design Challenges for Geographic Routing Algorithms . . . . .	10
2.4.1	Network Structure . . . . .	10
2.4.2	Characteristics of the Wireless Medium . . . . .	10
2.4.3	Capabilities of the Nodes . . . . .	10
<b>3</b>	<b>Geographic Routing Algorithms</b>	<b>13</b>
3.1	Beacon-based Algorithms . . . . .	13
3.1.1	Face Routing . . . . .	13
3.1.2	GPSR . . . . .	15
3.1.3	Other Algorithms . . . . .	16
3.1.4	Discussion . . . . .	17
3.2	Beacon-less Algorithms . . . . .	17
3.2.1	General Framework . . . . .	18
3.2.2	BLR: Beacon-Less Routing . . . . .	21
3.2.3	CBF: Contention-Based Forwarding . . . . .	21
3.2.4	IGF: Implicit Geographic Forwarding . . . . .	22
3.2.5	SIF: State-free Implicit Forwarding . . . . .	23
3.2.6	GeRaF: Geographic Random Forwarding . . . . .	24
3.2.7	MACRO: Integrated MAC/Routing Protocol . . . . .	24
3.2.8	BOSS: Beacon-less On Demand Strategy . . . . .	25
3.2.9	Discussion . . . . .	26
3.3	Three-dimensional Geographic Routing . . . . .	27
<b>4</b>	<b>Delivery Semantics</b>	<b>29</b>
4.1	Closeness . . . . .	30
4.2	Multiplicity . . . . .	31
4.3	Accept-outside . . . . .	32
4.4	Discussion . . . . .	33

<b>5</b>	<b>The Blind Geographic Routing Algorithm</b>	<b>35</b>
5.1	Basic Algorithm . . . . .	35
5.2	Timer Functions . . . . .	36
5.3	Recovery . . . . .	40
5.4	The 3D Version of BGR . . . . .	42
5.5	Avoidance of Simultaneous Forwarding . . . . .	44
5.6	Delivery Semantics . . . . .	44
5.7	Geocasting . . . . .	46
5.8	Duty Cycles . . . . .	48
5.9	Example . . . . .	49
<b>6</b>	<b>Location Errors</b>	<b>51</b>
6.1	Related Work . . . . .	51
6.2	Estimated Distance between Nodes . . . . .	52
6.3	Preparing BGR for Location Errors . . . . .	54
6.4	Preparing GPSR for Location Errors . . . . .	55
<b>7</b>	<b>Analytical Evaluation</b>	<b>59</b>
7.1	Expected Progress for Fixed Number of Nodes . . . . .	59
7.2	Expected Progress in Recovery Sector . . . . .	62
7.3	Expected Progress as Function of Network Density . . . . .	62
7.4	Probability of Successful Delivery . . . . .	63
7.4.1	No Recovery . . . . .	63
7.4.2	Turned Sector . . . . .	64
7.4.3	Augmented Sector . . . . .	64
7.4.4	Selected Values . . . . .	65
7.5	Delivery Probability for Spherical Sector . . . . .	65
7.6	Discussion . . . . .	69
<b>8</b>	<b>Simulation Results</b>	<b>71</b>
8.1	Unit Disk Graph Model . . . . .	72
8.1.1	Forwarding Areas . . . . .	73
8.1.2	Timer Functions . . . . .	75
8.1.3	Recovery . . . . .	76
8.1.4	ASF . . . . .	79
8.1.5	Geocasting . . . . .	79
8.1.6	Delivery Semantics . . . . .	82
8.1.7	Multi-flow Traffic . . . . .	83
8.1.8	Mobile Nodes . . . . .	84
8.1.9	Three-dimensional Topologies . . . . .	88
8.2	Location Errors . . . . .	89
8.3	Irregular Radio Model . . . . .	91

8.4	Combining Location Errors and Irregular Radio Model . . . . .	94
8.5	Discussion . . . . .	96
<b>9</b>	<b>Conclusion</b>	<b>97</b>
9.1	Summary . . . . .	97
9.2	Future Perspectives . . . . .	99
	<b>References</b>	<b>101</b>
	<b>Curriculum Vitae</b>	<b>113</b>





## List of Figures

3.1	Failure of greedy forwarding . . . . .	14
3.2	GG and RNG planarization . . . . .	14
3.3	Face routing example . . . . .	15
3.4	Forwarding areas: sector, circle, and Reuleaux triangle . . . . .	20
3.5	Values for computing the time interval . . . . .	20
3.6	Forwarding areas in CBF . . . . .	22
3.7	Priority regions in GeRaF . . . . .	24
4.1	Example where more than one node match the specified <i>closeness</i> semantics . . . . .	32
4.2	Example for <i>accept-outside</i> . . . . .	33
4.3	When the message gets stuck at node <i>A</i> , this node consumes the message if <i>accept-outside=true</i> , although node <i>B</i> is closer to the destination . . . . .	34
5.1	Different timer functions; $w = 40, d = 80, m = 0.5$ . . . . .	38
5.2	Timer functions; $w = 40, d = 80, m = 5.0$ . . . . .	38
5.3	Augmented sector strategy . . . . .	41
5.4	Maximum distance between the destination and any node within the turned sector, circle, and Reuleaux triangle (upper bound only) . . . . .	42
5.5	BGR forwarding example . . . . .	49
6.1	Real distance $d$ and estimated distance $e$ between two nodes . . . . .	53
6.2	Average percentage of fully connected nodes after planarization . . . . .	56
6.3	Average number of intersections per planar link . . . . .	57
7.1	Sector of angle $\beta$ . . . . .	60
7.2	Delivery probability in 2D for different recovery strategies . . . . .	66
7.3	Delivery probability in 2D for turned and augmented sector . . . . .	66
7.4	Delivery probability for spherical sector . . . . .	68
8.1	Delivery ratio for different forwarding areas . . . . .	73
8.2	End-to-end delay . . . . .	74
8.3	Stretch . . . . .	75
8.4	End-to-end delay for different timer functions . . . . .	76
8.5	Number of packets per generated message for different timer functions . . . . .	77

## List of Figures

8.6	Delivery ratio with and without recovery (Reuleaux triangle) . . . . .	78
8.7	Delivery ratio using augmented and turned sector . . . . .	78
8.8	Packet overhead using augmented and turned sector . . . . .	79
8.9	Delivery ratio with and without ASF . . . . .	80
8.10	Delivery ratio (percentage of nodes within geocast region that received the message) . . . . .	80
8.11	Packet overhead (number of packets divided by number of nodes) . . . . .	81
8.12	Average number of consumers for <i>at-least-one</i> semantics . . . . .	82
8.13	Influence of <i>accept-outside</i> on delivery ratio . . . . .	83
8.14	Average distance between destination location and location of consumer . . . . .	84
8.15	Delivery ratio at multi-flow traffic . . . . .	85
8.16	Stretch at multi-flow traffic . . . . .	85
8.17	Performance of BGR and GPSR in case of mobile nodes . . . . .	86
8.18	Modifications of GPSR for mobility . . . . .	87
8.19	Delivery ratio of BGR in 3D topologies . . . . .	88
8.20	Delivery ratio in 3D using different turning strategies . . . . .	89
8.21	Delivery ratio at location errors . . . . .	90
8.22	Delivery ratio at location errors for GPSR and fixed variants . . . . .	91
8.23	Packet reception rate in irregular radio model . . . . .	93
8.24	Delivery ratio using irregular radio model . . . . .	93
8.25	Delivery ratio with location errors and irregular radio model . . . . .	95

## List of Tables

2.1	Comparison of MANETs and sensor networks . . . . .	6
5.1	Values of $x(\tau)$ . . . . .	39
5.2	Threshold values up to which one timer function scatters better compared to linear timer function . . . . .	40
5.3	Maximum distance between the destination and any candidate node within the turned forwarding areas . . . . .	43
5.4	Sizes of forwarding areas/volumes as fraction of transmission area/volume . . . . .	43
6.1	Numerically calculated values of $E(G)$ and $E(H)$ assuming $d = 40$ . .	54
7.1	Calculated delivery probability (low network density) . . . . .	65
7.2	Volume of different 3D shapes . . . . .	68
8.1	Simulation parameters . . . . .	72
8.2	Radio parameters . . . . .	92



# 1 Introduction

Wireless ad hoc networks are decentralized networks of computer devices called *nodes* that communicate over wireless links and make dynamic forwarding decisions. They are classified into wireless sensor networks and mobile ad hoc networks (MANETs). The former consist of small-scale sensor nodes monitoring environmental data, typically deployed arbitrarily over an area with minimal configuration; the nodes in MANETs are portable devices like laptops, PDAs, or cellular phones. Applications for sensor networks include habitat monitoring, environmental surveillance, and healthcare applications; MANETs can be used for disaster relief operations where no communication infrastructure is available, or car-to-car communication.

Wireless ad hoc networks impose new challenges on routing algorithms for various reasons. A major problem is that topology changes caused by node failures, node removal or addition, and mobility are common. This is a big issue for routing algorithms that are based on neighborhood tables. Another point is that communication failures are likely. Anisotropic signal propagation and varying transmission ranges lead to unidirectional links, which affect unicast schemes based on neighborhood tables. Another important issue especially for sensor networks is energy efficiency, because energy is a limited resource and often battery replacement is virtually impossible. This demands for algorithms with low communication overhead, since the radio hardware is one of the most power-consuming units of the nodes.

In a wireless network where the nodes have information about their own locations, geographic routing can be performed, i. e., messages are forwarded toward the destination location using this positional information. Several geographic routing algorithms have been proposed in the literature. Most of them are beacon-based, i. e., the nodes exchange information about their neighborhood via beacon messages including ID and position. Forwarding decisions are made using this neighborhood information; the next hop is chosen from the neighborhood table, and the message is forwarded to the next hop via unicast. Apart from relying on bidirectional links (unless sophisticated neighborhood table management is performed), beacon-based routing algorithms suffer from inherent problems in case of frequent topology changes or high mobility caused by outdated neighborhood tables. In contrast to that, beacon-less algorithms, which operate with broadcast transmissions, do not face these problems, as they do not rely on static topologies.

The main goal of the thesis is the design and evaluation of a robust beacon-less geographic routing algorithm for wireless ad hoc networks called BGR (Blind Geographic Routing) that operates with as little communication overhead as possible and

## 1 Introduction

performs well under realistic conditions. Like other beacon-less algorithms, BGR does not use neighborhood tables, but forwards messages via broadcast. The general idea is that all nodes which receive this broadcast and are located within a designated forwarding area (determined by the forwarding node and oriented toward the destination) compete for becoming the next hop by starting a timer. The time is chosen dependent on the Euclidean distance of the node to the destination; the timer of the node closest to the destination expires first; this node forwards the message again. The other competing nodes notice this forwarding and cancel their timers.

This is the general scheme of beacon-less routing algorithms. BGR provides several significant improvements not found in previous algorithms. Like some other algorithms, BGR supports different recovery strategies in case of an empty forwarding area. In contrast to recovery strategies of other algorithms, the strategies of BGR neither have high communication overhead nor depend on regular communication models like the unit disk graph model. BGR also includes a novel strategy to avoid problems that arise when two or more nodes forward the message almost simultaneously. Furthermore, BGR is the first beacon-less geographic routing algorithm to support three-dimensional topologies. This is achieved by utilizing forwarding volumes instead of areas.

Additionally, the thesis proposes various delivery semantics for geographic routing, a topic that has not been addressed in the literature before. These semantics are *closeness* (how close must a node be located to the destination in order to consume the message), *multiplicity* (how many nodes may consume the message), and *accept-outside* (may a node where a message got stuck consume the message when the node is within transmission range from the destination).

BGR has been designed to tolerate location errors; simulation experiments confirmed a delivery ratio of over 95 % even for location errors in the order of magnitude of the transmission range. The width of the forwarding areas/volumes is adjusted according to the estimation of the standard deviation of the location error. Calculations of the estimated distance between nodes as a function of the real distance are presented in the thesis.

Furthermore, an analytical model for calculating the delivery probability of BGR is developed. An approximation of the delivery probability under the unit disk graph model (unit ball graph model in 3D) is calculated dependent on source-destination distance, transmission range, and network density.

Finally, extensive simulation studies show that the performance of BGR is over 95 % even in the case of mobility, radio irregularity, and location errors, while communication overhead is minimal. The results are directly compared with Greedy Perimeter Stateless Routing (GPSR), a well-elaborated beacon-based routing algorithm. Several recent variants of GPSR are investigated that were introduced partly in the literature and partly in this thesis. The goal of the variants was to improve the performance of GPSR for realistic scenarios. The simulations, however, show that BGR is still clearly superior.

In mobile networks, the delivery ratio of BGR even increases at higher node speeds. Although GPSR, in contrast to BGR, guarantees delivery under the unit disk graph model if the neighborhood graph is connected, GPSR shows significant deficiencies even under low mobility or radio irregularity. Small location errors do not degrade the performance of GPSR much; high location errors, however, lead to significantly worse delivery rates compared to BGR.

The thesis is organized as follows: Chapter 2 gives an introduction to wireless ad hoc networks. Chapter 3 provides an overview on geographic routing algorithms with a special focus on beacon-less algorithms. Additionally, a general framework for beacon-less geographic routing algorithms is developed. Novel delivery semantics for geographic routing are proposed and discussed in Chapter 4. The main contribution of the thesis is presented in Chapter 5, which gives a detailed description of the BGR algorithm. In Chapter 6, the influence of location errors on geographic routing algorithms is discussed. After presenting related work, improvements for BGR and GPSR are introduced on the basis of stochastic computations. An analytical model for calculating an approximation of the delivery probability of BGR is developed in Chapter 7. In Chapter 8, simulation results for BGR and GPSR are presented and evaluated. Finally, Chapter 9 gives a summary of the thesis and discusses future perspectives.





## 2 Ad Hoc Networks

This chapter gives an overview of wireless ad hoc networks. A classification of routing algorithms is presented and the superiority of geographic routing algorithms is emphasized. Finally, design challenges for geographic routing algorithms based on the characteristics of wireless ad hoc networks are discussed. This is the foundation for the BGR algorithm elaborated in this thesis.

### 2.1 History of Ad Hoc Networks

Research in the area of ad hoc networks started in 1973 with the PRNET (Packet Radio Network) project [JT87] of the U.S. Defense Advanced Research Projects Agency (DARPA). Focus of this project were testbed implementations and feasibility studies. Until 1986, networks of about 50 nodes had been realized. Apart from the PRNET project, little research, mainly of theoretical nature, has been done in the field of ad hoc networks until 1995, when the ACM launched the Annual International Conference on Mobile Computing and Networking (MobiCom), which was a driving force for many scientists to contribute to research of ad hoc networks.

Initially, the main focus of ad hoc network research was routing, since it was considered the most challenging problem, and the aspect of mobility was novel. Other research areas were quality of service and energy efficiency.

Research in the field of wireless sensor networks began around 1980 with the DARPA project DSN (Distributed Sensor Networks) [CK03]. Testbeds were built for military applications like ground surveillance or low-flying aircraft tracking. Processing was done with minicomputers available at that time, such as PDP-11 and VAX machines.

Around the year 2000, research interest in wireless sensor networks increased rapidly. Starting with the Smart Dust project [KKP99], various hardware platforms of small-scale sensor nodes were developed. Apart from hardware design, much research was done in developing algorithms for routing, data aggregation, localization, neighbor discovery, and related aspects. This research is still ongoing, since real-life deployments of sensor networks are rare until now.

Table 2.1: Comparison of MANETs and sensor networks (after [Wey06])

	MANETs	Wireless Sensor Networks
Number of nodes	Tens to hundreds	Hundreds to thousands
Node density	Low	High
Operation	Attended	Unattended
Active duty cycle	High	Very low
Redundancy	Low	High
Data rate	High	Low
Power supply	Rechargeable	Non-rechargeable
Mobility of nodes	Low to high mobility	Low
Communication	End-to-end	Many-to-one
Communication flows	Bidirectional	Predominantly unidirectional
Addressing	Address centric	Data centric

## 2.2 Characteristics of Wireless Ad Hoc Networks

In this section, the definition of wireless ad hoc networks and the related terminology used in this work are introduced. Characteristics and problems of wireless ad hoc networks are discussed.

A wireless ad hoc network is defined as follows:

**Definition 2.1** *A wireless ad hoc network is a network of computer devices called nodes, where the communication links are wireless. Each node is able to send packets to other nodes that are within transmission range.*

The network is called ad hoc, because the decision on which nodes forward a message is made dynamically depending on the current topology. Deployment of wireless ad hoc networks is possible with minimal configuration.

There are two types of wireless ad hoc networks, namely mobile ad hoc networks (MANETs) and wireless sensor networks. Table 2.1 lists the different characteristics of these network types.

MANETs became a research topic in the 1990s. The nodes are portable devices like laptops, PDAs, or cellular phones. Possible communication technologies are Wi-Fi (based on IEEE 802.11), Bluetooth, and ZigBee (based on IEEE 802.15.4). The network may operate standalone, or be connected to the Internet. Example scenarios are disaster relief operations in areas where no communication infrastructure is available, or car-to-car communication in so-called *VANETs* (vehicular ad hoc networks).

Wireless sensor networks gained research interest around the year 2000. They are networks of small-scale devices called *sensor nodes* consisting of one or more sensors, a microcontroller, one or more radio transceivers, and a power supply, usually

a battery. Applications for sensor networks include environmental monitoring and surveillance, industrial and health monitoring. A common scenario is to arbitrarily distribute the nodes over an area, like, for instance, by dropping them from an airplane. The sensor nodes may also be attached to people, animals, or mobile objects such as cars, for example to gather interactivity information.

Since a sensor network is usually intended to operate unattendedly, battery replacement is not possible. Therefore, energy efficiency is a critical design issue for sensor network applications. This is achieved by utilizing power saving modes and turning off unneeded components. Using duty cycles, i. e., switching components off and on, is possible for components that need not be active permanently. For instance, sensors can be turned on from time to time if the phenomenon to be observed occurs for a period of time that is long enough. The microcontroller can enter a low power mode when it is not needed. Turning off the transceiver, however, is critical, because no packets can be received when it is off. Power consumption in idle mode, when the transceiver is listening for incoming packets, is usually almost as high as in sending or receiving mode. Modern chips like the Chipcon CC2500 by Texas Instruments [TI] have a wake-on-radio feature, i. e., they periodically wake up from sleep mode and listen for incoming packets (see also [MMF<sup>+</sup>07]). This relieves the application of controlling the radio duty cycles.

Definition 2.1 on the facing page does not imply that the transmission range is fixed or known. However, this is often presumed when wireless ad hoc networks are studied. This is motivated by the assumption of isotropic signal propagation when using omnidirectional antennas. This model is called *unit disk graph model*:

**Definition 2.2** *The unit disk graph model assumes the transmission range of all nodes to be fixed (the unit). The transmission area of each node is circular; two different nodes can directly communicate with each other if and only if the distance between them does not exceed the transmission range.*

In three-dimensional topologies, the equivalent is the *unit ball graph model*; each node has a transmission volume, which is spherical.

The unit disk/ball graph model does not reflect realistic topologies adequately [ZHKS04]. Even when using omnidirectional antennas, signal propagation is highly anisotropic. The main reasons are reflection, diffraction, scattering, obstacles, and hardware calibration.

The implications of anisotropic signal propagation are manifold and not reflected in the unit disk/ball graph model:

- **Unidirectional links** occur frequently because of different transmission ranges of the nodes; this causes severe problems for routing algorithms. When building neighborhood tables based on beacon messages received from the neighbors, it is not possible to send packets to a neighbor whose link is unidirectional. Advanced neighborhood protocols identify bidirectional links through message

exchange and only use these for data propagation. However, this leaves many links unused; often the majority of the links is unidirectional [TWW06].

- **Unreliable communication** is common in wireless networks. Packets may get corrupted or lost. Reasons are noise, interference, distortion, and environmental conditions.
- **Unstable links** are a direct consequence of unreliable communication. When sending subsequent packets over the same link, some packets may get lost, others may be delivered. The fraction of delivered packets is a measure for the link stability. It may vary over time due to environmental changes.

Obstacles are another issue why the unit disk/ball graph model does not reflect reality in an adequate way. They can result in nodes not noticing each other despite being geographically close.

In addition to communication characteristics changing over time, the network topology can also change in wireless ad hoc networks. Factors are:

- Nodes may fail completely; for instance, they run out of energy, or they are physically destroyed.
- New nodes may be added to the network, or nodes are taken away.
- The nodes are mobile. The degree of mobility can be within a broad range, from very infrequent movements caused by blasts to very high mobility, e. g., when the nodes are attached to people, animals, or cars.

A frequently used communication pattern in sensor networks is measuring and sending data to a designated node called *sink* (or *base station*) periodically or when encountering special events, like sensing data that exceed a threshold. Periodic sending of data from different source nodes to one sink is called *convergecast*. Some applications even make use of multiple sinks. This facilitates *anycast* algorithms: data can be sent to the nearest sink.

### 2.3 Routing in Wireless Ad Hoc Networks

Routing protocols for wireless ad hoc networks must take into account topology changes and communication failures. Many different protocols have been proposed; a survey is given in [LK03]. Existing protocols can be classified into categories, which will now be discussed.

**Proactive** (or table-driven) protocols maintain routing information. Each node keeps its own routing table with information how to route packets to all possible destinations. The tables have to be consistent and up-to-date, so topology changes must be propagated. This has negative impact on scalability. The advantage, on the

other hand, is that routes are already established when needed, thus a low latency can be achieved. Because of the high overhead needed to maintain routing information and the lack of scalability, proactive protocols are commonly regarded as not suitable for ad hoc networks. A prominent proactive protocol is Destination-Sequenced Distance-Vector Routing (DSDV) [PB94].

**Reactive** (or on-demand) protocols establish routing paths only when needed. If a packet is to be delivered and no route has been set up yet, a route discovery is triggered. Since the network topology can change over time, route maintenance is also an important part of reactive protocols. The lower overhead to maintain routes is an advantage over proactive protocols; the drawback is higher latency to build new routes. Dynamic Source Routing (DSR) [JM96] and Ad-hoc On-Demand Distance Vector Routing (AODV) [PR99] are well-known reactive protocols.

**Hybrid** protocols try to combine the advantages of proactive and reactive protocols while overcoming their disadvantages. Usually, this is achieved by performing hierarchical routing. The Zone Routing Protocol (ZRP) [HPS02], for instance, divides the network into overlapping zones of variable size. The routes within the zones are maintained proactively, whereas inter-zone routes are discovered reactively. The need for route discovery and maintenance still leaves communication overhead.

**Geographic** protocols forward messages based on location information rather than node addresses. Each node knows its own position, either via GPS or some localization algorithm. Geographic protocols are the main topic of this thesis; they are discussed in detail in Chapter 3. Geographic information makes routing easier, because route discovery is not necessary; messages can be forwarded toward the destination using location information of the nodes. Especially in wireless sensor networks, the need for location information is not an extra requirement, because measured data without information about the location is commonly useless, so the nodes have to carry this information anyway.

**Multicast** protocols support routing to multiple destinations at the same time. Many existing multicast protocols are variants of unicast protocols. The majority of multicast protocols are reactive. The protocols are categorized into tree-based and mesh-based approaches.

**Power-aware** protocols are designed to reduce energy consumption for data transmission. This is achieved by sending at low transmission power over shorter distances. The energy needed to transmit a signal over a distance  $d$  is roughly proportional to  $d^\alpha$ , where  $\alpha \geq 2$  is the path loss exponent. Thus, when  $\alpha = 2$ , doubling the distance requires four times more energy for transmission. This means that the transmission over two hops needs half as much energy than over one hop in the optimal case. The cost, however, is a higher latency.

## **2.4 Design Challenges for Geographic Routing Algorithms**

When designing a geographic routing algorithm for wireless ad hoc networks, the characteristics of ad hoc networks must be reflected in the algorithm. These are the network structure, the characteristics of the wireless medium as well as the capabilities of the nodes. In the following, they are discussed in detail.

### **2.4.1 Network Structure**

Routing algorithms must be able to accommodate to topology changes quickly. The network structure can change due to nodes being taken away, new nodes in the network, or node mobility. Thus, it is not advisable to make routing decisions based on neighborhood tables, because these may be outdated at forwarding time. This suggests that a broadcast-based forwarding strategy is preferable, where nodes that receive the broadcast run a contention phase to determine the next hop. This way, topology changes are reflected in routing decisions immediately, which minimizes the chance of delivery failures caused by changes of the topology.

### **2.4.2 Characteristics of the Wireless Medium**

The characteristics of the wireless medium are important to consider. Unidirectional and unstable links, which are common, suggest that unicast schemes are inappropriate. They result in frequent delivery failures while on the other hand not incorporating unidirectional links. Broadcast-based schemes are better suited due to the redundancy of possible next hops. Many existing routing algorithms use unicast forwarding schemes and therefore do not exploit the broadcast characteristic of wireless networks.

Furthermore, anisotropic signal propagation makes algorithms inadequate that rely on the unit disk graph or unit ball graph model. This is the case for many graph-based algorithms, especially face routing (see Chapter 3).

### **2.4.3 Capabilities of the Nodes**

The capabilities of the nodes must be taken into account when designing a routing algorithm. Energy efficiency is crucial especially in wireless sensor networks, since energy is a scarce resource in sensor nodes. Usually, battery replacement is not feasible; thus, the hardware components must be used sparingly. This involves small memory requirements and avoidance of complex computations. One of the most power-consuming units of the nodes is the radio hardware. Therefore, it is important to put the transceiver into sleep mode frequently. The wake-on-radio feature, however, makes the incorporation of radio duty cycles in the application unnecessary, because the transceiver itself provides energy saving in idle mode. To minimize

## 2.4 Design Challenges for Geographic Routing Algorithms

energy consumption, a routing algorithm has to operate with as little communication as possible. The communication overhead induced by control messages must be kept low. Message-intensive algorithms such as neighborhood protocols should be avoided. Algorithms should also balance energy consumption between the different nodes in order to avoid the formation of hot spots [WWT05].

For geographic routing algorithms, the nodes must be aware of their locations. This location information is obtained via GPS or a localization algorithm like SeRLoc [LP04] or APIT [HHB<sup>+</sup>03]. It is inaccurate in real deployments. In extreme cases, the error can be in the order of magnitude of the transmission range [LP04]. Routing algorithms must account for this; they have to be designed to tolerate location errors.





## 3 Geographic Routing Algorithms

Geographic routing algorithms for ad hoc networks are classified into two categories: beacon-based and beacon-less algorithms. The key difference is that in beacon-based algorithms, nodes carry information about their neighborhood and make forwarding decisions based on this information, while beacon-less algorithms operate without neighborhood information, and messages are forwarded via broadcast using a contention scheme.

This chapter provides an overview of relevant geographic routing algorithms. First, beacon-based algorithms are introduced, which represent the traditional approach of geographic routing. The major focus of this chapter is on the second section, which introduces beacon-less algorithms. All of them are quite novel. Because this thesis deals with beacon-less algorithms, a general framework is presented at first, to which the single algorithms adhere. After this, the state of the art in the field of beacon-less geographic routing is discussed.

### 3.1 Beacon-based Algorithms

In beacon-based algorithms, the nodes exchange information about their one-hop neighborhood. This information is provided by periodically broadcasting short beacon messages that contain node ID and location. Each node keeps a neighborhood table. When a message is to be forwarded, the next hop is chosen from this table. The most common strategy is *greedy forwarding*, which means that the neighbor closest to the destination is chosen. This method fails when there is no node closer to the destination than the forwarder. Early algorithms considered only greedy forwarding [Fin87].

#### 3.1.1 Face Routing

Greedy forwarding fails when there is no neighbor closer to the destination than the current forwarder. Figure 3.1 on the next page shows an example. The gray area, which is empty, contains all nodes within transmission range from the forwarder that are closer to the destination than the forwarder. Plain greedy algorithms fail to deliver the message in this case. Advanced algorithms make use of a recovery strategy. The most prominent strategy is face routing, initially used in Compass Routing II [KSU99] and GFG [BMSU99], later in GPSR [KK00] and GOAFR<sup>+</sup> [KWZZ03].

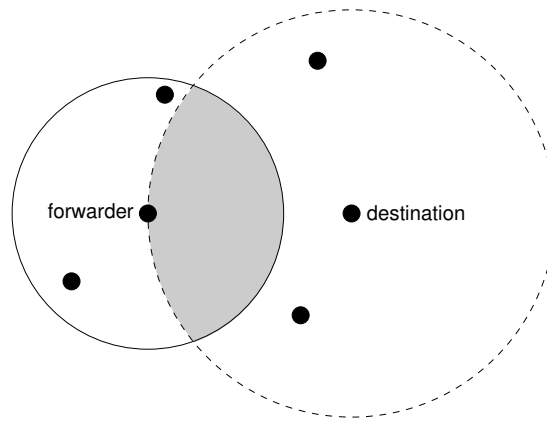


Figure 3.1: Failure of greedy forwarding

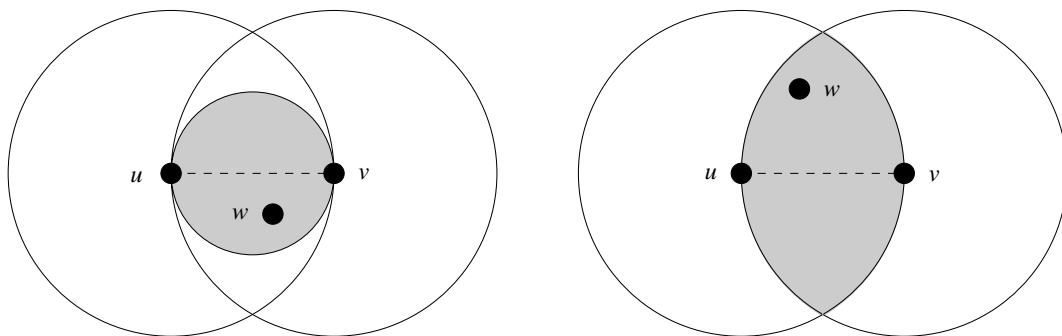


Figure 3.2: GG (left) and RNG (right) planarization

Face routing requires a planar (i. e., crossing-free), two-dimensional graph in order to work. Since typical network graphs have many intersections, a distributed planarization algorithm has to be run in order to remove crossing links. Commonly used planar graphs are the Gabriel Graph (GG) [GS69] and the Relative Neighborhood Graph (RNG) [Tou80]. Both graphs can be constructed in a distributed manner based on one-hop neighborhood information only. The resulting graph is connected if the original network graph is connected. The network graph has to be a unit disk graph.

Figure 3.2 illustrates how distributed GG and RNG planarization works. The link between nodes  $u$  and  $v$  is removed if there exists a node  $w$  within the gray area, otherwise it is retained. From the fact that the RNG area contains the GG area follows that the Relative Neighborhood Graph is a subgraph of the Gabriel Graph. Another planar graph is the Restricted Delaunay Graph (RDG) [GGH<sup>+</sup>01], but it is hard to compute and requires additional network communication.

To understand how face routing works, consider Figure 3.3 on the next page. Messages are routed along faces (enclosed polygonal regions) of the planar graph us-

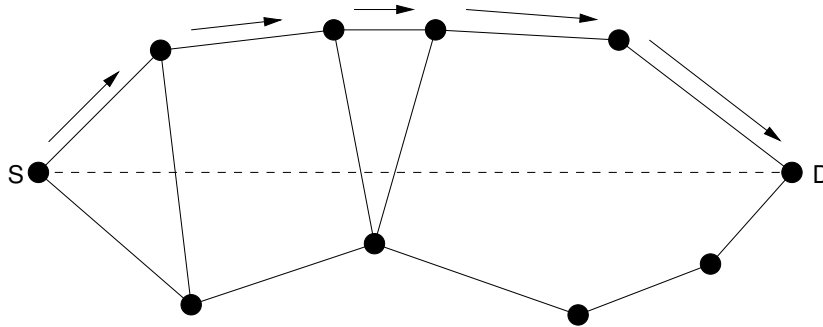


Figure 3.3: Face routing example

ing the right-hand rule, i. e., clockwise. When a link is encountered that crosses the source-destination line  $\overline{SD}$  (dashed in Figure 3.3), the face is changed, so that an adjacent face closer to the destination is traversed. Existing face routing algorithms have different rules to determine the node at which a face change is performed. These rules are outlined in [KGKS05b]. In the example shown in Figure 3.3, a face change is performed as soon as the next traversed link would cross the line  $\overline{SD}$ .

Algorithms that combine greedy and face routing usually operate in greedy mode whenever possible, i. e., they perform face routing only when the message reaches a node  $N$  that has no neighbor closer to the destination, and switch back to greedy mode as soon as a node is reached that is closer to the destination than  $N$ . Face routing on a connected planar graph has been shown to always reach the destination [BMSU99].

### 3.1.2 GPSR

Greedy Perimeter Stateless Routing (GPSR) [KK00] is the best-known beacon-based geographic routing algorithm. It combines greedy mode, which is used whenever possible, and the so-called *perimeter mode*, which is in fact face routing.

Neighborhood tables are obtained by periodically broadcasting beacons, which include ID and position of the sending node. The beacon interval  $B$  is a configurable parameter. To avoid synchronization effects [FJ94], the beacon transmission is jittered by 50% of  $B$ , so that the time between two beacon transmissions is uniformly distributed in  $[0.5B; 1.5B]$ . A node is deleted from the neighborhood table after a timeout interval of  $4.5B$ , which is three times the maximum time between two beacons. This means that three consecutive beacons must have been missed in order to delete a neighbor from the table. To reduce the number of beacons, GPSR processes implicit beacons. These are regular data packets that are regarded as beacons so that the scheduling of the next regular beacon is delayed. The network interfaces are required to operate in promiscuous mode, so that every node within transmission range receives packets regardless of the designated receiver. When a node sends a data packet, it resets its beacon timer.

If the neighborhood table contains a node closer to the destination  $D$ , the message is forwarded in greedy mode. Otherwise, it is forwarded in perimeter mode using face routing until a node is reached that is closer to  $D$  than the node  $P$  which entered perimeter mode. A face change is performed at the first link that crosses the line  $\overline{PD}$  at a point closer to  $D$  than the node where the current face was entered. Perimeter mode operates either on the Gabriel Graph or the Relative Neighborhood Graph.

#### 3.1.3 Other Algorithms

This section gives a short outline on other geographic routing algorithms based on neighborhood information. For a broader discussion, refer to [AKK04, AY05, GS03].

Compass Routing [KSU99] forwards the message to the neighbor with the smallest angle between forwarder-neighbor line and forwarder-destination line. This algorithm is not loop-free. Compass Routing II (also introduced in [KSU99]) is the first face routing algorithm in the literature. All faces are required to be convex. The Delaunay triangulation is proposed as planar graph; however, no algorithm for construction is given.

Greedy-Face-Greedy (GFG) [BMSU99] is the first algorithm that combines greedy forwarding and face routing. The Gabriel Graph is used as planar graph.

Location Aided Routing (LAR) [KV00] is a collection of routing algorithms that build paths with route requests and restricted flooding.

Geographical and Energy Aware Routing (GEAR) [YGE01] forwards messages based upon a cost function that includes distance to destination and remaining energy. If possible, a neighbor that is closer to the destination is selected. GEAR also supports disseminating messages within a target region using restricted flooding.

GEDIR [SL01] is a variant of greedy forwarding that selects the neighbor closest to the destination regardless of the distance between forwarder and destination. Thus, the message can be forwarded to a neighbor that is farther away from the destination than the forwarder. A loop can only occur between two consecutive nodes, so the algorithm can be made loop-free. In a variant with restricted flooding, message delivery is guaranteed.

GOAFR<sup>+</sup> [KWZZ03] is another algorithm that combines greedy forwarding and face routing on the Gabriel Graph. Face routing is improved so that the algorithm is asymptotically optimal in the worst case.

Greedy Distributed Spanning Tree Routing (GDSTR) [LLM06] is an algorithm that has been designed to work completely without face routing. Instead, so-called *hull trees* are built and maintained, which are spanning trees in that a convex hull is assigned to each node; the convex hull contains the locations of all descendant nodes. When greedy routing fails, routing on the hull trees is performed. Building, maintaining, and repairing the trees imposes an extra overhead. Unlike other algorithms, GDSTR does not assume the unit disk graph model; however, bidirectional links are still required.

### 3.1.4 Discussion

Beacon-based algorithms have been studied extensively in terms of simulation and analytical evaluation. The main problem is that the vast majority of these studies is based on the unit disk graph model and hence does not face the problem of unidirectional or unstable links, which are by far more common than reliable, bidirectional links in real sensor network deployments [TRV<sup>+</sup>05, TWW06, GEW<sup>+</sup>02, CACM03, TWV06].

The usage of advanced neighborhood protocols [WSBC04, MGLA02] instead of simple beaconing schemes could partially solve this problem by identifying bidirectional links and only using these for routing. The problem is that stable bidirectional links are rare and therefore these strategies tend to a quick formation of hot spots, while many nodes would be excluded from routing paths. Additionally, in low-density networks, the resulting neighborhood graphs are likely to be disconnected.

Another issue is the high communication overhead necessary to build the neighborhood tables. Even when there is no data traffic, packets must be exchanged to keep neighborhood information up-to-date. In systems with moderate message flow, the number of beacons is extraordinarily high compared to the number of data packets. Apart from raised energy consumption, the probability of collisions increases.

Finally, there is always some latency between the instant when the neighborhood table was updated and the instant when a data message is forwarded. If the topology changed in the meantime, the forwarding decision is based on outdated neighborhood information, and the message may be forwarded to a node that is no longer reachable. This may be caused by node failures or, to a worse extent, node movements. The policy of GPSR to keep nodes in the neighborhood tables until three consecutive beacons have been missed is especially fatal, because nodes stay in the neighborhood tables for a long time after having moved out of transmission range. Delivery failures are very likely in this case.

## 3.2 Beacon-less Algorithms

Like beacon-based algorithms, beacon-less routing algorithms also make use of greedy mode, but instead of forwarding the message to the neighbor closest to the destination, a neighbor closest to the destination within a specific area called *forwarding area* becomes the next hop. Since no neighborhood tables are maintained, they cannot use unicast for this purpose; they use broadcast instead. The closest neighbor is not known a priori; instead, it is determined during the forwarding process itself. Nodes within the forwarding area compete for becoming the next hop using a timer-based contention scheme. The timer of the node closest to the destination expires first, whereupon this node forwards the message. The key issue is that the next forwarder is selected without additional communication.

Beacon-less routing algorithms are opportunistic algorithms [SWW05]. The idea of opportunistic algorithms is to choose the next hop at the time of packet transmis-

sion, making use of the redundancy of nodes. Whereas some opportunistic algorithms like ExOR [BM04] use neighborhood information, beacon-less algorithms generally operate without storing neighborhood information.

#### 3.2.1 General Framework

In the following, a general framework for beacon-less routing algorithms is provided and an integrative terminology is established. The general scenario is point-to-point communication in a two-dimensional topology from a source node  $S$  to a destination location  $D$ . Extensions for three-dimensional topologies are straightforward and will be discussed in Section 5.4. Existing beacon-less algorithms assume that there is a node at location  $D$ ; however, BGR drops this assumption by supporting different delivery semantics.

The concept of a *forwarding area* is fundamental for beacon-less geographic routing. Nodes located within the forwarding area can preferably mutually communicate with each other; the key idea is that one node within the forwarding area forwards the message, while the other nodes notice this and do not forward the message. Forwarding areas are described using geographical constraints, such that a node can determine whether it belongs to a forwarding area or not using only its location information.

An important metric is the *packet progress* of a forwarding candidate  $C$ . It is defined as the difference between the distance from  $S$  to  $D$  and the distance from  $C$  to  $D$ . Thus, the packet progress is positive if  $C$  is closer to  $D$  than  $S$ .

The basic routing algorithm, not yet incorporating delivery semantics, can be described as follows:

1. Initially,  $S$  is the forwarder. It stores  $D$  in the header of the message  $M$ .
2. The forwarder selects a forwarding area and stores its description in the header of  $M$ .
3. The forwarder broadcasts  $M$ .
4. A node  $N$  that receives  $M$  acts as follows:
  - If  $N$  is not located within the forwarding area,  $M$  is ignored.
  - Otherwise, if  $N$  is located at the destination  $D$ ,  $N$  broadcasts a CANCEL packet and accepts  $M$ .
  - Otherwise, if  $N$  has a contention timer running for  $M$ , the timer is canceled and  $M$  is ignored.
  - Otherwise,  $N$  becomes a forwarding candidate and starts a contention timer depending on the packet progress toward  $D$ .

5. A node which receives a CANCEL packet belonging to a message for that it has a contention timer running, cancels the timer, because the message has reached  $D$ .
6. The first node whose contention timer expires continues with step 2.

The algorithm contains some degrees of freedom. A specific algorithm must define

- shape and width of the forwarding area,
- a recovery strategy for the case that the forwarding area is empty, and
- the duration of the contention timer.

The forwarding area is placed such that the position of the forwarder is farther away from  $D$  than any other point within the forwarding area, and it is symmetric about the line from forwarder to  $D$ . The forwarding area has to be small enough so that all nodes within it can communicate with each other (under the unit disk graph model). This is necessary, because the other nodes must cancel their contention timers when one node forwards the message. The forwarding area should be formed such that the probability that it contains the node with largest progress toward the destination is high. Thus, conditions for a qualified forwarding area are to

1. contain many forwarding candidates with small distance to  $D$ ,
2. contain few forwarding candidates with large distance to  $D$ ,
3. be sufficiently large in order to contain as many forwarding candidates as possible,
4. contain at least one node, and
5. have a description which allows a simple membership test.

Figure 3.4 on the next page shows examples for forwarding areas: a  $60^\circ$  sector, a circle, and a Reuleaux triangle. The Reuleaux triangle meets the first condition best, the sector the second one, and the circle the third one. The fourth condition is impossible to guarantee for arbitrary topologies. All three forwarding areas conform to the last condition, since membership tests for them can be done with some efficient calculation.

The timer function should prefer nodes with large packet progress, but on the other hand differentiate the timer durations at different nodes in order to avoid collisions and simultaneous forwarding.

To express the timer function, different values can be considered (see Figure 3.5 on the following page):

- $c$ : distance between candidate node and destination



### 3 Geographic Routing Algorithms

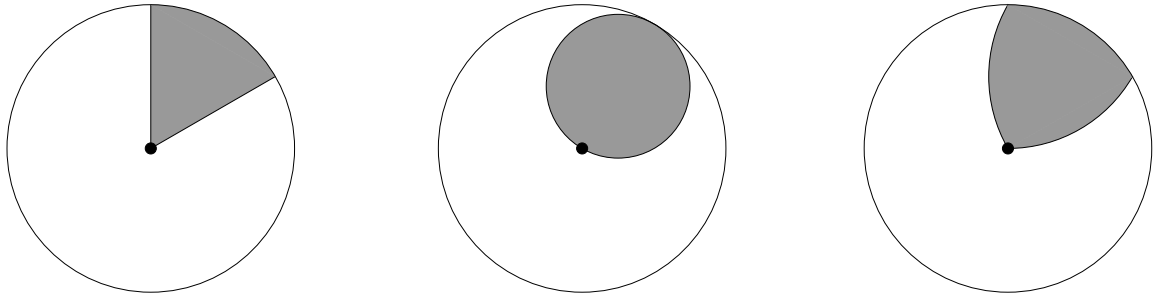


Figure 3.4: Forwarding areas: sector, circle, and Reuleaux triangle

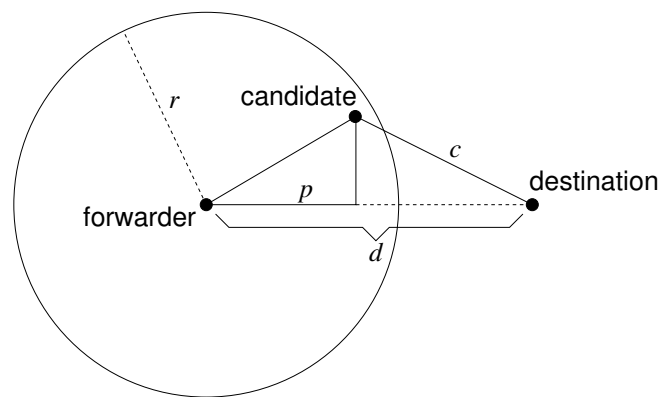


Figure 3.5: Values for computing the time interval

- $d$ : distance between forwarder and destination
- $p$ : distance between forwarder and projection of candidate node's position on the straight line from forwarder to destination
- $r$ : transmission range

To allow for a recovery strategy in case the forwarding area turns out to contain no node (or no node responds), there has to be a maximum waiting time  $m$ . The choice of  $m$  is crucial for the message delivery time. There is a trade-off between delivery time and packet collisions: The forwarding latency depends on  $m$ , but if it is set too low, packet collisions will arise frequently, because many nodes may try to forward the same message almost simultaneously. The value of  $m$  can be set dependent on the node density. In high-density networks, the probability of collisions is higher; therefore,  $m$  should not be chosen too low. In low-density networks, lower values of  $m$  are possible and lead to faster message delivery.

### 3.2.2 BLR: Beacon-Less Routing

BLR [HBBW04, HB03] discusses all three above-mentioned forwarding areas. For the timer  $t$ , three different functions are proposed, which have influence on the latency in different ways:

1.  $t = m \cdot \left( \frac{r-p}{r} \right)$
2.  $t = m \cdot \left( \frac{p}{r} \right)$
3.  $t = m \cdot e^{\sqrt{p^2+d^2}-1}$

The first function causes a linear decrease of the computed delay toward the destination. The second function favors nodes close to the forwarder; the intention is to reduce energy consumption. The third function is an example for an advanced timer function that is intended to reduce collisions by spreading the time intervals near the forwarder (this function also favors nodes close to the forwarder). However, nodes far away from the forwarder compute similar delay intervals, which causes more collisions if these nodes forward the message simultaneously.

Due to the fact that a node notices which node has forwarded the message, BLR has an option to send subsequent messages to the same destination directly to the next hop via unicast. This can be done without delay and with adjusted transmission power. Admittedly, to accomplish this, a routing table has to be stored, which has a negative impact on the scalability. To account for new nodes, an expiration time is determined; once it is expired, messages are sent via broadcast again. Optionally, there is a promiscuous mode, in which nodes within the forwarding area process unicast packets that are not sent to them and start a timer for the case that the receiver of the packet is no longer available.

If the forwarder does not notice another node forwarding the packet within the time interval  $m$ , the forwarding area is considered to be empty, and a recovery strategy is triggered: The node broadcasts a request, and all neighbor nodes reply and send their positions. If one of the neighbor nodes is closer to the destination, it is chosen as the next hop; otherwise, the forwarding continues using the right-hand rule on a Gabriel Graph until a node closer to the destination is found.

### 3.2.3 CBF: Contention-Based Forwarding

CBF [FWMH03] divides the forwarding process into two phases: *contention* and *suppression*. In the contention phase, a node is determined as the next hop; in the suppression phase, it suppresses other candidates from also forwarding the message.

The contention phase works timer-based using the following timer function:

$$t = m \cdot \left( 1 - \frac{d-c}{r} \right).$$

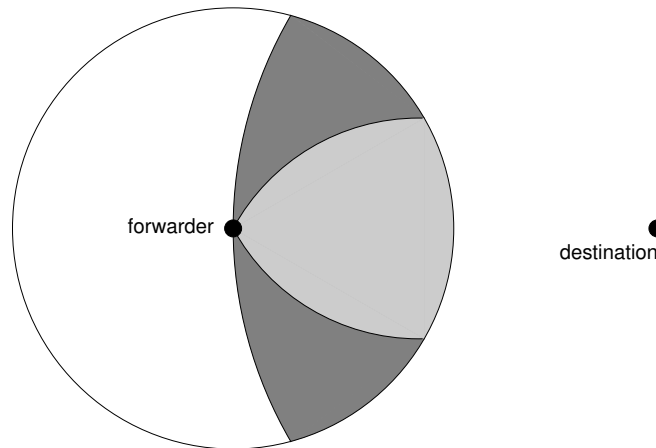


Figure 3.6: Forwarding areas in CBF: Reuleaux triangle and adjacent regions with positive progress

The timer is only started when  $d > c$ . Note that the computed time is non-negative and not greater than  $m$ , because  $0 \leq d - c \leq r$ .

For the second phase, the suppression phase, two alternatives are proposed: *area-based suppression* and *active selection*. Area-based suppression operates with a forwarding area. The areas that are discussed are the circle and the Reuleaux triangle; the latter is favored due to Condition 1 on page 19. If the forwarding area turns out to be empty, the forwarder sends up to two more broadcasts with different forwarding areas; these are the areas left and right to the original forwarding area with positive progress (i. e., where nodes are closer to the destination than the forwarder, see Figure 3.6). If these areas are also empty, another recovery strategy is necessary, but its details are left open.

In contrast to that, active selection works as follows: The forwarder broadcasts a *Request To Forward* (RTF), whereupon all nodes with positive progress start a timer and send back a *Clear To Forward* (CTF) when the timer expires. The forwarder chooses the next hop among all nodes from which it has received a CTF. It sends the data packet via unicast to this node. Other nodes cancel their timers when noticing a CTF. Through active selection, all forms of packet duplication are excluded. The cost is a packet overhead of factor three, a higher collision probability, and possible transmission failures due to the unicast scheme.

#### 3.2.4 IGF: Implicit Geographic Forwarding

IGF [BHSS03] is a combined routing/MAC protocol that shifts the timer-based selection of the next hop to the MAC layer. The MAC protocol IEEE 802.11 is modified in the sense that instead of an RTS an *Open Request To Send* (ORTS) is broadcast. Nodes in the forwarding area start a timer and send a CTS on expiration, if no other

node has sent a CTS. Nodes outside the forwarding area and nodes that overhear a CTS from another node set their NAV (Network Allocation Vector) timers to avoid collisions with subsequent packets. The remaining process is identical to the original 802.11 protocol: The forwarder unicasts the data packet to the selected hop, which acknowledges with an ACK packet.

As forwarding area, the sector is used. The node density is assumed to be high enough so that the forwarding area contains at least one node. An optional enhancement is a shift of the forwarding area, if the forwarder has not received any CTS.

The scheduled time is computed by an advanced function using three parameters:

- distance to the destination,
- available energy of the node,
- additional random delay.

Nodes with little remaining energy resources increase the time interval, so that nodes with more energy are favored.

### 3.2.5 SIF: State-free Implicit Forwarding

Strongly influenced by IGF, SIF [CDV05a] is also a cross-layer protocol that combines MAC handling and routing. It is likewise based on IEEE 802.11 with a broadcast RTS. The forwarding area is the area of positive packet progress; this is the area that contains all nodes which are closer to the destination than the forwarder. Nodes within this area can be farther away from each other than the transmission range  $r$ , but they only need to perform carrier sensing to detect the transmission by another node, which is claimed to be possible since the maximum distance of nodes within the forwarding area is less than  $2r$ ; the carrier sensing range is assumed to be adjustable up to  $2.2r$  using 802.11 radios. The timer function is a weighted function which comprises the packet progress, remaining energy, and a random value.

When the forwarder does not receive any CTS, the RTS is re-transmitted up to a threshold value (e. g., three times) in order to cope with transient failures. If this also fails, two different recovery strategies are proposed:

- The transmission power is gradually increased and the RTS is re-transmitted. However, not all transceiver hardware supports adjustable transmission power.
- The forwarder marks itself as dead end and does not forward subsequent messages to the same destination. However, simulation results by the same authors show that this method is ineffective [CDV05b]. In low-density networks, it performs even worse than routing without any recovery at all.

Obviously, end-to-end delay is high in sparse topologies, because at first the RTS is broadcast several times and afterwards a recovery strategy is triggered in which the RTS is re-transmitted again, each time waiting for the maximum timer interval.

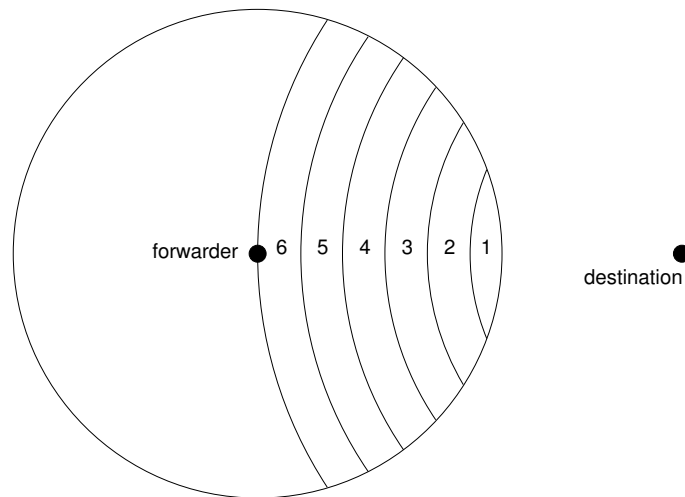


Figure 3.7: Priority regions in GeRaF

#### 3.2.6 GeRaF: Geographic Random Forwarding

The principle of GeRaF [Zor04] differs from the other algorithms. The contention timers are replaced by time slots assigned to different regions of the forwarding area. Collision avoidance, which is integrated in the protocol, is performed on the MAC layer. The nodes periodically sleep and wake. Like in SIF, the forwarding area is the area of positive packet progress, which is divided into several priority regions ordered by distance to the destination (see Figure 3.7). After the forwarder broadcasts a RTS, a time slot is assigned to each region for sending a CTS, ordered by priority. As soon as one node sends a CTS, nodes in other slots stop competing. So the next hop is located within the highest-priority region that contains non-sleeping nodes. If it contains more than one non-sleeping node, all of them will send a CTS simultaneously, which will lead to a collision. This collision is detected, and a collision resolution algorithm is executed: All nodes involved decide with probability 0.5 if they send again in the next slot or not. This is repeated until only one node is left. If there is a slot in which no node decides to send, the decision is repeated in the following slot.

When all regions of the forwarding area are empty, the forwarder retries later, when another set of nodes is awake. The duty cycles and the fact that only nodes with positive packet progress participate in the contention process imply that GeRaF is only suitable for high network densities [CMN<sup>+</sup>05].

#### 3.2.7 MACRO: Integrated MAC/ Routing Protocol

MACRO [FGL<sup>+</sup>05] is another protocol that combines MAC and routing functionality. The nodes run duty cycles, but need not be synchronized. When a packet is to be forwarded, the forwarder broadcasts several wake-up messages until all forward-

ing candidates are switched on. Then, a so-called *go-message* is broadcast, which triggers the contention phase: Each candidate starts a timer depending on the packet progress divided by the transmission power needed to reach this node. This metric is called *weighted progress*; the next hop is chosen such that the weighted progress is maximized. The intention is to reduce energy consumption by using low transmission power. When the timer expires, the candidate sends its weighted progress factor to the forwarder, which collects these messages and then estimates on the basis of a probabilistic function if a better next hop can be found by increasing the transmission power. If this is the case, the process starts again with a higher transmission power. When the forwarder does not expect any further improvement, it forwards the data message to the chosen next hop.

Although simulation studies presented in [FGL<sup>+</sup>05] show promising results in terms of energy consumption, MACRO is only suitable for high-density networks. Besides, the influence of unidirectional links has not been studied yet; it can be expected that they will cause severe problems due to the inherent need for message exchange between forwarder and possible candidate nodes.

### 3.2.8 BOSS: Beacon-less On Demand Strategy

BOSS [SMPR07] is a protocol that tries to account for realistic physical layers. It does so by broadcasting the data message first; all receiving nodes start a contention timer and broadcast a RESPONSE message when it expires. Nodes noticing this message cancel their timers, but only when the message has been sent by a node with positive progress. The forwarder sends a SELECTION message to the first node that has sent a RESPONSE message, whereupon all other nodes cancel their timers and the selected node forwards the message again.

The data message is broadcast first because it is the longest message and therefore the probability of unsuccessful transmission is much higher than for the following control messages. Nodes that did not receive the data message successfully do not take part in the contention process. On the other hand, nodes that did receive the data message will likely be receiving the control messages also.

Similar to GeRaF, the transmission area is divided into regions. Each node within a region computes the same contention timer value, but adds a random value that is small enough not to interfere with a value from another region. The idea behind this timer scheme is to reduce collisions caused by nodes with similar progress toward the destination.

When only nodes with negative progress send RESPONSE messages, all these messages are collected by the forwarder, which then changes to perimeter mode, builds a planar graph, and forwards the message using face routing. As usual, perimeter mode is left when a node is reached which is closer to the destination than the node that started perimeter mode.

### 3 Geographic Routing Algorithms

Although BOSS has explicitly been designed to work with realistic physical layers, it is completely based on the unit disk graph model. The timer scheme assumes that all receiving nodes are within the fixed transmission area. This is a severe drawback of the algorithm.

#### 3.2.9 Discussion

All the presented protocols have several drawbacks. First, some of them (CBF, SIF, and GeRaF) require a very high network density, because they only forward to nodes that are closer to the destination than the forwarder. The optional recovery strategy of IGF (shift of the sector), in contrast, includes candidates with larger distance to the destination than the forwarder. The same applies to BLR, MACRO, and BOSS; BLR and BOSS use face routing as recovery strategy, while MACRO collects messages from all nodes within transmission range.

A severe problem of all algorithms is that the recovery strategies are not robust or impose a high communication overhead. BLR and BOSS use face routing, which has drawbacks, because it requires additional communication for collecting neighborhood information, and transient communication failures result in delivery failures, since unicasts are used. Additionally, it is not suitable for realistic networks, because it is based on the unit disk graph model. In the recovery strategy of CBF, the backup forwarding areas are small compared to the main forwarding area. As already mentioned above, they contain only nodes with positive packet progress. The recovery strategy of IGF is merely optional and not investigated in depth. The two recovery strategies proposed by SIF are impracticable, as discussed in Section 3.2.5 on page 23. In both GeRaF and MACRO, all candidate nodes within a predetermined zone respond, which (intentionally) leads to collisions and high communication costs.

All of the presented algorithms impose a high communication overhead. Face routing of BLR and BOSS, active selection of CBF, and (on MAC layer) request-based RTS/CTS mechanisms of IGF, GeRaF, SIF, MACRO, and BOSS all require exchanging several messages. Apart from the message overhead, unicasts are used in this case, which are subject to fail.

Moreover, problems that arise when the contention timers of two nodes expire almost simultaneously are considered by none of the algorithms (with the exception of collision avoidance schemes in GeRaF, MACRO, and BOSS). Another issue is that all algorithms except MACRO assume that the transmission range  $r$  is fixed and known. Nodes outside the nominal transmission area do not take part in the contention process, although they could potentially forward the message, which would lead to a higher delivery ratio especially in sparse networks. The simulation studies and analytical evaluations are always based on the unit disk graph model, which is unrealistic since all links are bidirectional. Hence, the influence of unidirectional links and transient transmission failures has not been studied yet.

Additionally, no algorithm addresses the problem of location errors and studies their influence on performance and delivery ratio. Finally, the sender must know the exact position of the destination, and there must be a node exactly at the destination location. The option to route a message to a node in the vicinity of a specific location is not provided.

### 3.3 Three-dimensional Geographic Routing

All geographic routing algorithms discussed so far assume Euclidean two-dimensional network topologies. Location information is held as  $x$  and  $y$  coordinates. This is sufficient when the network is deployed in a plane, e. g., for environmental monitoring in a large area. In some cases, however, the network can become three-dimensional. Application scenarios include networks within buildings, underwater networks, or even networks in space. Especially underwater sensor networks have gained research interest recently [HYW<sup>+</sup>06, PM05, APM05]. To enable 3D routing, the existing algorithms have to be extended. However, it is not sufficient to just add the  $z$  coordinate. This section describes the challenges that arise when making the step from 2D to 3D.

Current research on three-dimensional networks is focused on connectivity and coverage [AH06, HTL04, Rav04]. Little work has been published that addresses 3D routing. In [PM05], two routing algorithms for underwater networks are proposed based on link metrics. The first algorithm, which is delay-insensitive, only selects nodes that are closer to the destination, and therefore is subject to fail for sparse networks. The second algorithm is delay-sensitive, but uses a centralized approach.

Face routing is not directly extensible to 3D network graphs. The faces to be traversed are determined by the line from source to destination. However, in 3D graphs, this line does not determine the faces [KFO05]. Thus, 2D face routing algorithms are not directly applicable to 3D. A heuristic variant of face routing, which does not guarantee delivery, but has been shown to perform well in simulations, is proposed in [KFO05] and further elaborated in [AFO06b] and [AFO07]. Greedy routing and partial flooding are combined in [AFO06a]. Several randomized 3D routing algorithms are presented in [FW08]. Some challenges in designing 3D algorithms are discussed in [PPKS06]. Beacon-less 3D routing algorithms have not been proposed yet. The non-existence of a local 3D geographic routing algorithm with guaranteed delivery for unit ball graphs has been proven recently in [DKN08].

A problem with three-dimensional topologies is that more nodes are needed for network coverage than in two-dimensional topologies. For a quantitative comparison, suppose that the same average number of neighbors is to be achieved in a 2D topology of area  $a^2$  and a 3D topology of volume  $a^3$ . The transmission range  $r$  is fixed, i. e., the unit disk graph (or, in 3D, unit ball graph) model is assumed. If the total number of nodes in the 2D topology is  $n$ , then the average number of neighbors is  $\frac{n\pi r^2}{a^2}$  (ignoring border effects). For this value to be the average number of neighbors in the 3D



### 3 Geographic Routing Algorithms

topology, the total number of nodes has to be

$$\frac{n\pi r^2}{a^2} \cdot \frac{a^3}{\frac{4}{3}\pi r^3} = \frac{3an}{4r}.$$

This means that in a 3D topology, the total number of nodes has to be by a factor of  $\frac{3a}{4r}$  larger than in a 2D topology.

## 4 Delivery Semantics

All geographic routing algorithms discussed in Chapter 3 assume that the location of the destination is known to the sender and that there is a node exactly at the destination location, the node locations being published for instance by a distributed location service like GLS [LJDC<sup>+</sup>00]. The location service, however, imposes an extra overhead. Besides, the destination locations may be incorrect due to location errors. Additionally, nodes may fail or move, so it is not always possible to keep track of the current state.

The position of the destination is indeed known to the sender when messages are only routed to pre-defined static locations such as base stations. For pure data gathering applications, this scheme is sufficient. However, in more advanced ad hoc network applications, it may be necessary to send messages to arbitrary locations. For example, when the user actively wants to spread queries into the network targeting an area of interest [TWW05], or when the user seeks services in a specific region, the routing protocol must support destination locations without knowing the exact locations of the nodes in the vicinity of the destination. A node (or some nodes) in the vicinity of the destination location shall receive the message. Thus, when receiving a packet, a node has to decide whether it is a suitable destination for this packet; additionally, if only one node shall consume the message, the nodes have to agree about the consumer. Since the concrete strategy is application-dependent, different delivery semantics are needed.

Although there are different strategies in the literature for finding a suitable destination node, each algorithm uses only a single strategy. Some algorithms perform a limited flooding in the vicinity of the destination [HBBW04], which also requires the sending of many packets. An interesting alternative is utilized by GHT [RKS<sup>+</sup>03], where the message is routed to the node that is closest to the destination location. This is achieved by taking advantage of the perimeter mode of GPSR: The destination location is traversed using the right-hand rule; when the message has looped around the location, the first node that receives the message for the second time in perimeter mode declares itself as destination and consumes the message.

SenriGan [IT07] is a modification of GPSR targeted at sensor networks in which the nodes are equipped with cameras, each covering its own sensing area. The destination location is a point to be observed, not a location of a node. Similar to GHT, the destination is traversed in perimeter mode, whereupon information of the sensing areas of the traversed nodes is collected in the message, until the message has looped around the perimeter. There may be more than one node whose sensing area covers

## 4 Delivery Semantics

the destination point, and there are several ways of sending back the reply; for instance, the data from the node with smallest distance to the destination could be sent back.

*Anycast* is a well-known technique for routing messages to any out of several possible destinations. Within the scope of sensor networks, anycast can be used when multiple sinks exist; messages are routed to the nearest sink [IDL99, TTS05, HSS06]. However, anycast routing schemes do not use geographic routing and consequently are highly application-specific.

Other algorithms include time in delivery semantics. *Mobicast* [HLR03] is a spatiotemporal multicast scheme that supports moving destination areas. Since the semantics proposed in this thesis are not dependent on time, such schemes are not covered.

This chapter provides a taxonomy of delivery semantics and proposes three parameters, which can be combined independently. An application that sends messages has to set these parameters according to the specific requirements. A routing algorithm must then deliver the message with respect to the desired semantics. The proposed delivery semantics are independent from concrete routing algorithms. The semantics were introduced in [WT06a].

Basically, there are three questions that have to be answered when determining the destination for a message in a wireless ad hoc network:

1. How close to the destination must a node be in order to consume the message?
2. Is it acceptable that multiple nodes consume the message, or should only one node consume it?
3. When the message gets stuck at a node  $A$  because there is no node that is as close to the destination as desired, should node  $A$  drop the message or consume it if  $A$  is sufficiently close (within transmission range) to the destination?

In the following, these questions are discussed in detail.

### 4.1 Closeness

Assume a sensor network for environmental monitoring, where neither the user nor the base station know the exact positions of the nodes, because the nodes have been spread ad hoc and did not exchange topology information. The user requests the sensor data for a specific location and initiates a query, having this location as destination position. However, it is not known whether there is a sensor node exactly at this location. Indeed, this is extremely improbable. Thus, a node has to consume the message if its distance to the destination location lies within a designated limit of tolerance  $t$ . Using Euclidean metrics, the destination area is a disk (ball in 3D) with radius  $t$ . This approach raises the probability that a destination node is found. The limit can be

set depending on the network density. Assume that the network consists of  $n$  nodes spread uniformly in a plane of surface area  $A > \pi t^2$ . Then the probability that the destination area is empty is

$$\left(1 - \frac{\pi t^2}{A}\right)^n,$$

not regarding border effects. For this probability to be less than  $p$ , the limit of tolerance  $t$  has to be set to a value such that

$$t > \sqrt{\frac{A}{\pi} \left(1 - p^{\frac{1}{n}}\right)}.$$

Consider, on the other hand, the case that the message containing the measured data is sent from the sensor node to the sink that initiated the query. The message is explicitly addressed to the sink, so it is known that there is a node at the designated location, provided the sink is stationary. In this case, the limit of tolerance can be set to zero; this also ensures that no other node than the sink consumes the message.

Another possibility is to route the message to the node that is nearest to a given destination location, like in GHT.

Following from these examples, three different semantics are proposed for the parameter *closeness* in order to describe destination locations: *exact*, *nearby*, and *nearest*. In the *nearby* case, a limit of tolerance has to be given.

The *nearby* and *nearest* semantics make routing to mobile nodes possible. In *exact* semantics (de facto used by all existing geographic routing protocols), messages can only be routed to stationary nodes. Hence, the incorporation of the *closeness* semantics adds a substantial benefit to geographic routing schemes, as no stationary destination node is required.

## 4.2 Multiplicity

If the *nearby* semantics is used, there may be more than one potential destination node (see Figure 4.1 on the next page for an example). The routing protocol may ensure that only one of these nodes consumes the message; however, this leads to more overhead, since the nodes must somehow agree on the winner, which comes at additional communication costs and is subject to race conditions. But this is not necessary in cases when it is not important that only one node receives the message. For example, if the sensor node at (or nearby) a specific location shall be set to alarming mode, because the user wants to pay special attention to this location, it is not crucial that only one node receives the message. To account for node failures, it is even better if multiple nodes receive it.

On the other hand, there are cases when only one node shall receive the message. In general, this is the case when the nodes provide different services. For instance, when the message is addressed to the sink, no other node is allowed to consume it.

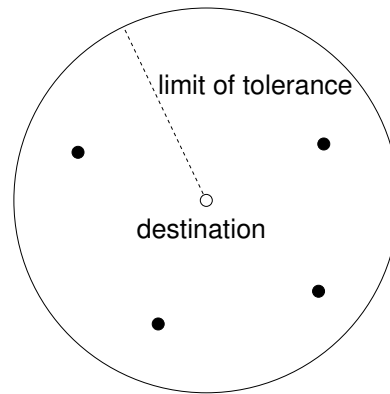


Figure 4.1: Example where more than one node match the specified *closeness* semantics

Another example are sensor/actor networks, where a message is sent to activate an actor, and it must be guaranteed that only one actor is activated.

Inspired by semantics for Remote Procedure Calls (RPCs), the following semantics are useful for the multiplicity of the destination:

- *exactly-one*: the message must be consumed by exactly one node,
- *at-most-one*: the message must be consumed by zero or one node,
- *at-least-one*: the message must be consumed by one or more nodes,
- *all*: the message must be consumed by all nodes meeting the *closeness* semantics,
- *maybe*: the message may reach a node or not.

Note that these semantics are not meant for describing *how many times* a node receives the message. This aspect of delivery semantics is orthogonal to the ones discussed here. The semantics specify *how many nodes* receive the message.

### 4.3 Accept-outside

Consider the case that there is no node that matches the specified *closeness* semantics, but the message has reached a node that is already in proximity of the destination. Figure 4.2 on the facing page illustrates an example: The destination location is within the transmission range, the current node *A* does not yet match the *closeness* semantics, but no closer node can be found, either because closer nodes do not exist or because they are currently not reachable. Now, there are two possibilities: Either the message is regarded as non-deliverable and dropped, or node *A* declares itself as destination

and consumes the message. The strategy to be applied depends on what has higher priority: that a node which exactly matches the *closeness* semantics receives the message, or that any node receives it at all. For example, if data at a specific location shall be measured, it may be tolerable if merely data near this location are measured; however, if the sink is the destination, it is not acceptable when another node consumes the message. This demands for a Boolean parameter *accept-outside*.

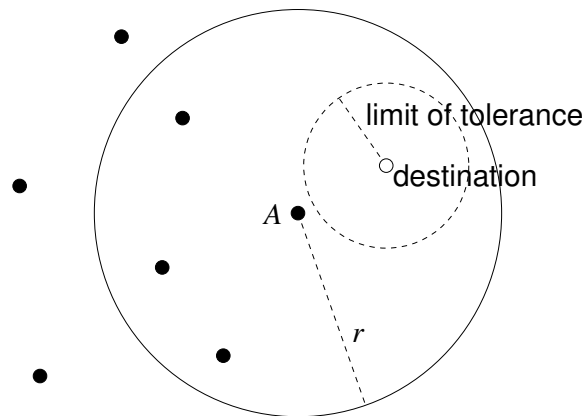


Figure 4.2: Example for *accept-outside*: The message gets stuck at node A, but the destination location is within transmission range

This parameter can be used in combination with *exact* and *exactly-one* semantics to deliver the message to a node that is close to a location, for instance when the *nearest* semantics is not available. This is not possible with *nearby* semantics, because the limit of tolerance must be chosen sufficiently high in order to guarantee that the area of tolerance contains at least one node. However, in *nearby* semantics the message is consumed by any node within the area of tolerance, so this will likely happen as soon as the message enters the area of tolerance, although there may be closer nodes. With *accept-outside* semantics, on the other hand, the message will not be consumed until it gets stuck, so it is typically closer. Note that this semantics does not guarantee that the node closest to the destination consumes the message. This is not covered by the proposed semantics. Figure 4.3 on the next page shows a counterexample.

## 4.4 Discussion

Not all combinations of these parameters seem to be useful. For instance, in *exact* semantics with *accept-outside=false*, the semantics *at-least-one* and *exactly-one* seem to yield the same results (when there are no nodes with identical positions). However, the routing protocol may behave differently; for example, it may deliver the messages faster or send fewer packets in *at-least-one* semantics.

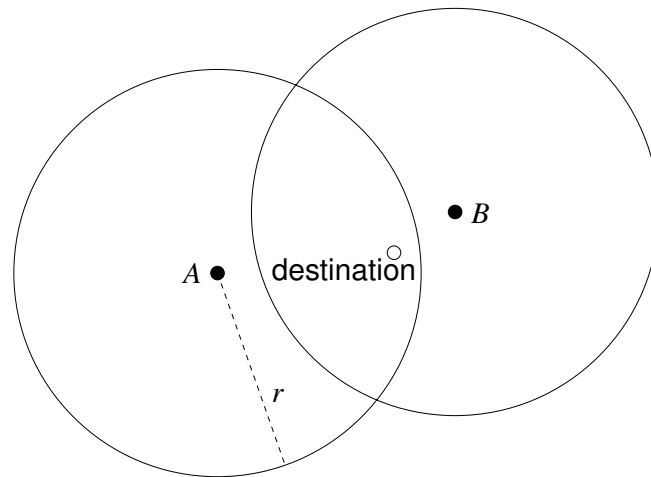


Figure 4.3: When the message gets stuck at node  $A$ , this node consumes the message if  $accept-outside=true$ , although node  $B$  is closer to the destination

An important issue is that it is not guaranteed that the same nodes consume subsequent messages sent to the same location, especially when the nodes are mobile. The messages may be forwarded on different paths and hence reach different nodes, even in static networks. Only when  $closeness=exact$  or  $multiplicity=all$ , with  $accept-outside=false$  in both cases, all messages are consumed by the same nodes, provided that the network is static, under an ideal radio model.

## 5 The Blind Geographic Routing Algorithm

In this chapter, the Blind Geographic Routing (BGR) algorithm is described in detail. It is called *blind* because all forwarding decisions are made without any neighborhood information. The BGR algorithm constitutes the main contribution of this thesis. An early version of BGR is presented in [WT05]. In contrast to the algorithms described in Section 3.2, BGR does not assume a constant transmission range for all nodes; the parameter  $r$  is merely an estimation of the transmission range that is needed for constructing the forwarding areas. Better accuracy of  $r$  results in better performance of the algorithm. BGR also has a robust recovery strategy and a method to circumvent problems that arise in simultaneous forwarding situations. Furthermore, BGR is the first beacon-less geographic routing algorithm that supports three-dimensional topologies. The protocol is completely broadcast-based, which makes it robust against transmission failures and mobility; furthermore, this eliminates the need for unique node IDs.

As a routing protocol, BGR resides in OSI layer 3 (network layer). Special MAC layer functionality is not needed. Multiplexing like TDMA or FDMA is not necessary, because BGR is contention-based and robust against packet collisions and transmission failures. Carrier sensing, however, is useful to reduce the number of collisions; so it should be provided by the MAC layer.

### 5.1 Basic Algorithm

As BGR makes use of forwarding areas (in 2D), a precise definition of a forwarding area is necessary. Henceforth, let a forwarding area be an area which includes the forwarder and is characterized by the following parameters:

- shape of the area,
- position of the forwarder,
- width,
- location.

Different shapes have been introduced in Section 3.2.1. The width (or diameter) is a function of  $r$ ; it is equal to  $r$ , unless location errors are taken into account, which will be discussed in Chapter 6. If not in recovery mode, the location is given such that the



## 5 The Blind Geographic Routing Algorithm

forwarder is the farthest point from the destination within the forwarding area, and the forwarding area is symmetric about the line from forwarder to destination.

The requirement that all nodes within the forwarding area can mutually communicate with each other is not included in the definition, because the transmission range need not be fixed or known. Apart from that, transient communication failures may also occur, so that this requirement is not revisable. However, the width of the forwarding area has to be chosen such that mutual communication is feasible with a sufficiently high probability.

The algorithm starts as follows: The source node stores the position of the destination and a unique message ID in the packet header. The message ID consists of the position of the source node and a sequence number (or a time stamp). The source node selects a forwarding area and broadcasts the packet. In addition, it starts a recovery timer  $t_r$ , which expires after  $t_{\max} + a$ , where  $t_{\max}$  is the maximum possible contention timer duration of any node within the forwarding area and  $a$  is a short additional delay which is added for not missing packets that are sent shortly before the deadline.

A node that receives this broadcast first checks if it is a destination node for this message. Details about that follow in Section 5.6.

If the node is not the destination, it checks if it has a contention timer or a recovery timer running for this packet. When this is the case, the timer is discarded and the packet is ignored.

Otherwise, the node checks if it is located within the forwarding area stored in the packet, and if it has not forwarded the same message before. If one of these conditions does not hold, the packet is ignored (in order to prevent routing loops, double forwarding is not performed). To detect packet duplicates, the IDs of the last  $v$  forwarded messages have to be stored in a separate list. It is assumed that  $v$  is large enough so that messages whose IDs have been removed from the list are not present in the network anymore. Unless network traffic is very high,  $v$  can be chosen small. Thus, the storage of the IDs does not result in serious memory overhead.

When none of the above apply, the node becomes a forwarding candidate and starts a contention timer  $t$ . Different timer functions are discussed in the following section. When the contention timer expires, the candidate declares itself as next hop and forwards the packet like described above.

### 5.2 Timer Functions

The purpose of the contention timer is to find the best forwarding candidate while on the other hand reducing the probability of collisions, which arise when two or more nodes compute similar timer values. The parameters  $m$ ,  $d$ , and  $c$  have been introduced on page 20;  $w$  is the width of the forwarding area (forwarding volume in 3D, see Section 5.4). For small values of  $c$ , the timer function should have the following properties:

- The calculated value should be small in order to account for fast message delivery, and
- for different values of  $c$ , the function should scatter the calculated values in order to avoid simultaneous forwarding of several nodes.

Additionally, the function must have the following properties to cover the complete co-domain from 0 to  $m$ :

$$\begin{aligned} t(d-w) &= 0 \\ t(d) &= m \end{aligned}$$

The following five contention timer functions fulfill these properties:

$$\begin{aligned} \text{square root : } t(c) &= m \cdot \sqrt{1 - \frac{d-c}{w}} \\ \text{logarithmic : } t(c) &= \log \left( (1 - e^m) \frac{d-c}{w} + e^m \right) \\ \text{linear : } t(c) &= m \cdot \left( 1 - \frac{d-c}{w} \right) \\ \text{exponential : } t(c) &= (m+1)^{1 - \frac{d-c}{w}} - 1 \\ \text{quadratic : } t(c) &= m \cdot \left( 1 - \frac{d-c}{w} \right)^2 \end{aligned}$$

Figure 5.1 on the following page shows the function graphs for the relevant range of  $c$  for  $m = 0.5$ . For values of  $c$  between  $d-w$  and  $d$ , the order in which the functions are listed above corresponds to the order of the resulting values, with the quadratic function computing the smallest values. The square root function scatters the timer values best for small values of  $c$ , while the quadratic function does this for large values of  $c$ . For large values of  $m$ , the logarithmic function computes the longest time. Figure 5.2 on the next page shows the graphs for  $m = 5$ .

In the following, the scattering of the timer functions for small differences in the values of  $c$  is investigated. A good scattering is important to avoid simultaneous packet forwarding.

Due to the monotonicity of the timer functions, the following property holds for two values  $c_1$  and  $c_2$ :

$$|c_1 - c_2| > x(\tau) \Leftrightarrow |t(c_1) - t(c_2)| > \tau, \quad (5.1)$$

i. e., for a given time difference  $\tau$ , a node distance  $x(\tau)$  is to be found that guarantees that the computed values differ by more than  $\tau$ .

Without loss of generality let  $c_1 \leq c_2$ . Then Equation (5.1) can be expressed as follows:

$$c_2 - c_1 > x(\tau) \Leftrightarrow t(c_2) - t(c_1) > \tau. \quad (5.2)$$

## 5 The Blind Geographic Routing Algorithm

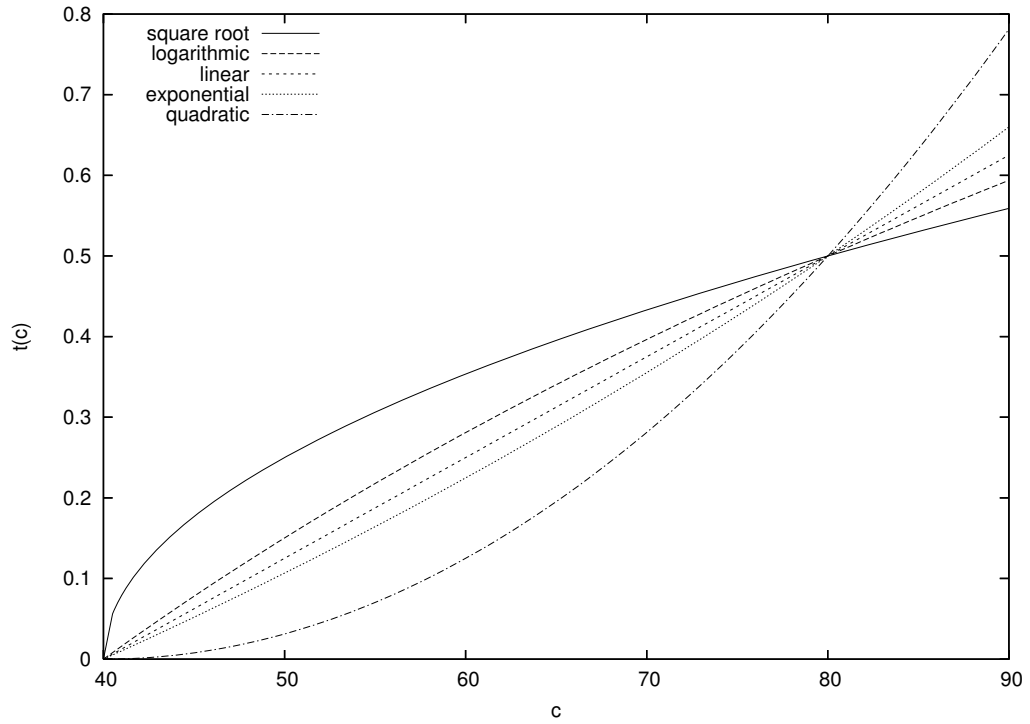


Figure 5.1: Different timer functions;  $w = 40$ ,  $d = 80$ ,  $m = 0.5$ ;  $c > 80$  corresponds to nodes farther away from the destination than the forwarder

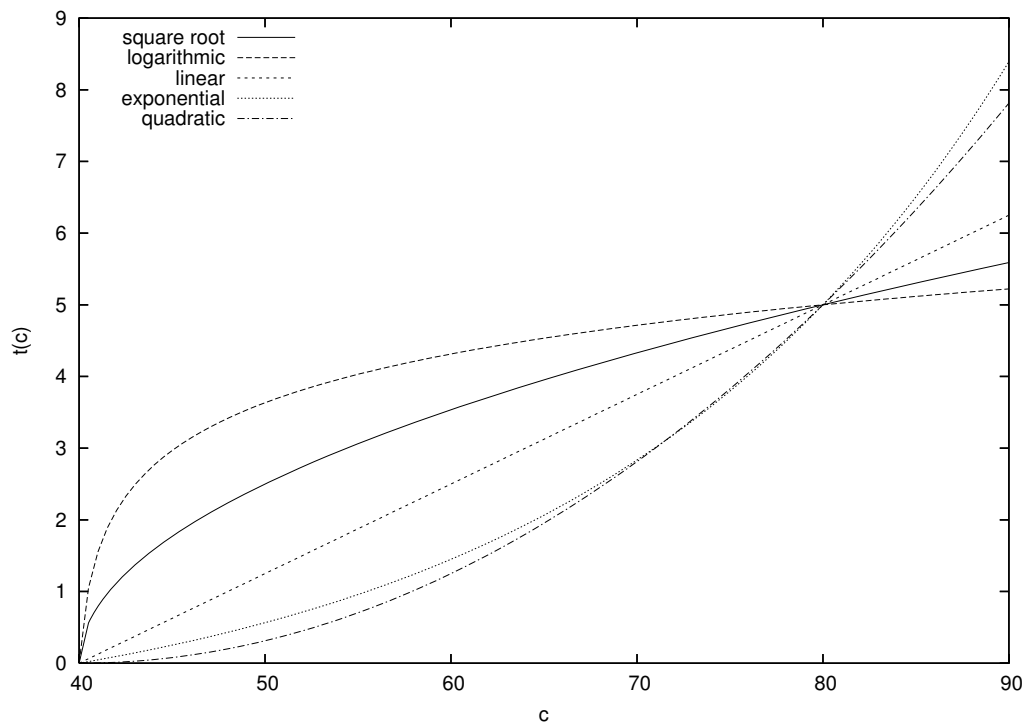


Figure 5.2: Timer functions;  $w = 40$ ,  $d = 80$ ,  $m = 5.0$

Dependent on the second derivative of the timer function being positive or negative, the inequality

$$t(c_2) - t(c_1) \leq t'(c_i)(c_2 - c_1) \quad (5.3)$$

holds, with  $i = 1$  for negative and  $i = 2$  for positive second derivative. If the second derivative is zero (which is the case for the linear timer function),  $i$  can be chosen 1 or 2.

From Equations (5.2) and (5.3) follows (assuming  $t'(c_i) \neq 0$ , which holds for all considered functions when  $c > d - w$ )

$$\tau < t(c_2) - t(c_1) \leq t'(c_i)(c_2 - c_1) \Leftrightarrow \frac{\tau}{t'(c_i)} < c_2 - c_1. \quad (5.4)$$

With Equation (5.1), this gives  $x(\tau) = \frac{\tau}{t'(c_i)}$  for the node distance. Table 5.1 shows the values of  $x(\tau)$  for the five timer functions.

Table 5.1: Values of  $x(\tau)$

Function	$x(\tau)$
square root	$\frac{2\tau w \sqrt{1 - \frac{d-c_1}{w}}}{m}$
logarithmic	$\tau \frac{(1-e^m)(d-c_1) + we^m}{m}$
linear	$\frac{\tau w}{m}$
exponential	$\frac{\tau w}{\log(m+1) \cdot (m+1)^{1 - \frac{d-c_2}{w}}}$
quadratic	$\frac{\tau w}{2m(1 - \frac{d-c_2}{w})}$

The scattering of the timer functions can be compared by equating the values of  $x(\tau)$ . Equating two functions gives the threshold value for  $c$ , up to which the timer function that computes higher values scatters better. The values equated to the linear timer function are listed in Table 5.2 on the next page. The concrete values for  $m = 0.5$  are also shown. The results indicate that the square root function scatters better than the linear function when the difference between  $d$  and  $c_1$  is larger than  $0.75w$ ; otherwise, the linear function scatters better. The other functions can also be equated; however, this results in rather complex formulas.

The timer function to be chosen for a specific application depends on the network density and the delay requirements of the application. Functions with larger scattering at low candidate-destination distances are better suited for high-density networks,

## 5 The Blind Geographic Routing Algorithm

Table 5.2: Threshold values up to which one timer function scatters better compared to linear timer function

Function	$x_f(\tau) = x_{\text{linear}}(\tau)$	For $m = 0.5$
square root	$d - c_1 = \frac{3}{4}w$	$d - c_1 = 0.75w$
logarithmic	$d - c_1 = \left( \frac{e^m}{e^m - 1} - \frac{1}{m} \right) w$	$d - c_1 \approx 0.541w$
exponential	$d - c_2 = \left( \frac{\log \log(m+1) - \log m}{\log(m+1)} + 1 \right) w$	$d - c_2 \approx 0.483w$
quadratic	$d - c_2 = \frac{1}{2}w$	$d - c_2 = 0.5w$

because there will likely be several nodes with small distance to the destination. However, these functions compute higher timer values, which result in higher end-to-end delay. If these concerns are not crucial, the linear function is the best choice, because it is the easiest one to compute.

### 5.3 Recovery

On expiration of the recovery timer  $t_r$ , the forwarding area is assumed to be empty. For this case, a recovery strategy is needed. A survey of recovery strategies has been carried out by [CV07]. However, all strategies discussed there have severe limitations, which make them unsuitable for BGR: need for topology information (planar-graph-based, geometric, and cost-based techniques), high communication overhead (flooding-based), impracticability or ineffectiveness (heuristic). Therefore, a novel recovery strategy is used by BGR inspired by the one used by CBF (cf. Section 3.2.3): The forwarding area is turned by  $60^\circ$ ; the turning direction is chosen randomly. The message is broadcast again and a new recovery timer  $t_r$  is started. If this timer also expires, the forwarding area is turned in the other direction and the message is sent a third time. If this also fails, the message is regarded as undeliverable and dropped. The advantage compared to the strategy of CBF is that the forwarding area is not smaller in size than in the first attempt and therefore contains more potential forwarding candidates. The fact that the turned forwarding area overlaps with the one of the first attempt when using the circle or Reuleaux triangle is not a disadvantage: due to transient communication failures, nodes in the overlapping area may not receive the packet until the second attempt. The forwarding area could also be turned by  $120^\circ$  in a fourth attempt, but studies revealed that this increases performance only marginally, while in most situations being useless.

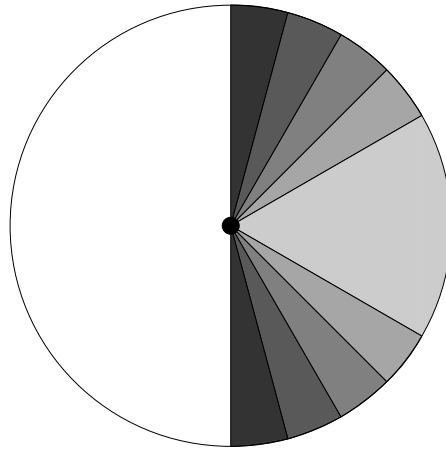


Figure 5.3: Augmented sector strategy: the sector is augmented by  $30^\circ$  in each attempt

When using the sector as forwarding area, BGR supports another novel recovery strategy: Instead of turning the sector, it is augmented by  $30^\circ$  in each recovery step (see Figure 5.3). Recovery is done up to four times, so in the last try, the entire half-circle serves as forwarding area. The disadvantages of this strategy are that the areas of the different steps overlap, and that not all nodes within the augmented sectors can communicate with each other (in the unit disk graph model). The former issue, however, turns into an advantage when transient communication failures arise: The inclusion of the sector from the previous try lowers the chance that areas are considered empty which in fact are not. The latter problem is not a big issue when applying irregular radio models, since transmission ranges vary anyway. A downside of this strategy is a higher communication overhead, because recovery is triggered up to four times and packet duplicates occur more often. The benefit is higher reliability especially in the presence of communication failures.

A drawback of the recovery strategies of BGR is that the forwarder broadcasts the data packet multiple times. It would be more energy-efficient to broadcast it only once; all nodes within any forwarding area of all attempts receive and store the data packet, but only the nodes within the first forwarding area take part in the contention process. When the forwarder recovers, it just broadcasts a small control packet containing a description of the recovery forwarding area instead of sending the entire data packet again. Although being more energy-efficient in recovery mode, this strategy has two downsides due to which BGR does not use it:

- It is less robust, because a node must successfully receive two packets (data and control packet) in order to take part in the contention process. If the data packet is sent every time, all receiving nodes are immediately prepared to forward it.

## 5 The Blind Geographic Routing Algorithm

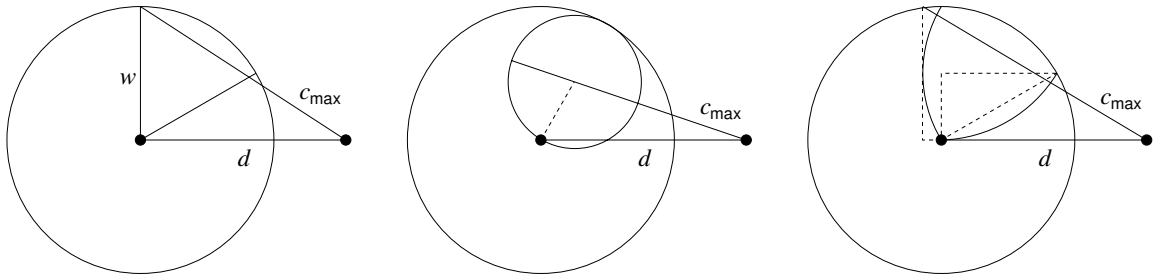


Figure 5.4: Maximum distance between the destination and any node within the turned sector, circle, and Reuleaux triangle (upper bound only)

- The receiving nodes must start another timer to discard the data packet eventually.

Although the recovery strategies of BGR cannot guarantee delivery even in unit disk graphs, they are robust against failures and thus a good heuristic. Performance evaluation will show that in realistic settings, BGR outperforms beacon-based routing schemes that guarantee delivery in unit disk graphs.

From the property  $t(d) = m$  of the timer functions follows that the recovery timer has to be set to  $m + a$  when not in recovery mode. When the forwarding area is turned, however, there can be candidate nodes that are farther away from the destination than the forwarder. This results in contention timers set to a value greater than  $m$ . For the computation of the recovery timer, the maximum distance between the destination and any node within the shifted forwarding area is needed (see Figure 5.4). The resulting values are listed in Table 5.3 on the facing page. The exact value for the Reuleaux triangle is rather complicated to compute; however, the upper bound is sufficient for practical usage. The duration of the recovery timer when turning the forwarding area is computed with the timer function substituting  $c$  by  $c_{\max}$ . For the linear timer function, this yields

$$t_r = m \cdot \left( 1 - \frac{d - c_{\max}}{w} \right) + a.$$

### 5.4 The 3D Version of BGR

To operate in 3D space, the forwarding areas have to be converted into forwarding volumes by constructing the solid of revolution around the forwarder-destination axis [WT07]. Therefore, the 2D sector becomes a spherical sector, the circle becomes a sphere, and the Reuleaux triangle becomes the solid of revolution of a Reuleaux triangle. Note that this is different from the Reuleaux tetrahedron, whose diameter is slightly larger than the radius of the intersecting spheres from which it is constructed.

The additional dimension of the destination location leads to more possible routing directions, which results in lower delivery rates. This can easily be seen when

Table 5.3: Maximum distance between the destination and any candidate node within the turned forwarding areas

Forwarding Area	$c_{\max}$
Sector	$\sqrt{d^2 + w^2}$
Circle	$\frac{w}{2} + \sqrt{d^2 + \frac{w^2}{4} - \frac{1}{2}dw}$
Reuleaux triangle (upper bound)	$\sqrt{\left(d + \left(1 - \frac{\sqrt{3}}{2}\right)w\right)^2 + w^2}$

Table 5.4: Sizes of forwarding areas/volumes as fraction of transmission area/volume

	Sector (Sph. sector)	Circle (Sphere)	Reuleaux triangle
2D	$\frac{1}{6} \approx 0.167$	$\frac{1}{4} = 0.25$	$\frac{1}{2} - \frac{\sqrt{3}}{2\pi} \approx 0.224$
3D	$\frac{1}{2} - \frac{\sqrt{3}}{4} \approx 0.067$	$\frac{1}{8} = 0.125$	$\frac{1}{2} - \frac{\pi}{8} \approx 0.107$

considering the fraction of the transmission area/volume that is covered by the forwarding areas/volumes. Table 5.4 indicates that in 3D, the forwarding volumes cover only about half as much of the transmission volume as the corresponding forwarding areas in 2D. As a consequence, the 3D version of BGR performs recovery up to four times per hop in contrast to two times in the 2D version. The first forwarding volume is obtained by turning the forwarder-destination axis by  $60^\circ$  downward first (exception see below), then in the opposite direction (mirrored about this axis), then turned by  $90^\circ$  about this axis, and in the last try mirrored again. A problem is that there are gaps between the turned forwarding volumes, which is not the case for the 2D forwarding areas (with the exception of the circle, which leaves two small gaps). Also, the overlapping regions are larger than in the 2D case. There are no gaps when the augmented spherical sector is used as forwarding volume. Only with this recovery strategy, the entire half-sphere is covered after four times of recovery. Thus, this strategy is especially interesting for the 3D case.

The  $z$  coordinate of a node can give a hint in which direction a message should be forwarded first in recovery mode. When the node is located close to the ground, the forwarding volume should not be turned downward first, because the message is likely to get stuck when it comes close to the network boundary, where the number of



## 5 The Blind Geographic Routing Algorithm

neighbors is lower on average. In this context, the ground is the lowest possible  $z$  value of any node in the network. For outdoor networks, this may be the earth's surface, for underwater networks the sea bottom, etc. BGR applies the following strategy: If the  $z$  coordinate of the node is larger than the estimated transmission range  $r$ , the forwarding volume is turned down at the first and third retry, and up at the second and fourth retry (on different axes); otherwise, it is done the other way round.

### 5.5 Avoidance of Simultaneous Forwarding

Another problem arises when two or more candidate nodes forward the packet nearly simultaneously due to similar contention timer values. Nodes within the intersection of the forwarding areas selected by these forwarders start a contention timer when receiving the first packet and cancel it when receiving the second one, since they assume that the next hop has been found. For the same reason, the recovery timers are also canceled, so the packet is lost (see Section 5.9 for an example).

The solution for this problem is to store the hop count in the packet. When a packet is received and a contention or recovery timer is running for this packet, the hop count of the received packet is compared with the hop count of the stored packet. If they are equal, the timer is not canceled. This method is called *Avoidance of Simultaneous Forwarding* (ASF).

### 5.6 Delivery Semantics

The implementation of the delivery semantics discussed in Chapter 4 is based on CANCEL packets, which are sent by nodes that consume the message. Nodes that receive a CANCEL packet stop processing the associated data packet. As for the multiplicity parameter, BGR supports *exactly-one* and *at-least-one*. The *at-most-one* and *maybe* semantics are not really necessary, since *exactly-one* is a stronger variant that can be used instead. The *all* semantics, which covers all nodes meeting the specified closeness semantics, is covered by geocasting; this is discussed in Section 5.7. The *nearest* semantics is not implemented, because it is not possible to achieve without large communication overhead.

The following pseudo code describes the concrete steps that are performed when a node receives a packet for which it is a destination node (regarding the closeness semantics):

```

if contention timer running for associated packet then
  cancel timer
  ignore packet
else if packet has not been received before then
  if node is within forwarding area or
    multiplicity=at-least-one then
    compute delay using timer function
    if delay<limit or closeness=exact then
      delay:=limit
    end if
    if multiplicity=at-least-one then
      delay:=0
    end if
    schedule destination timer to expire
      after delay (wait:=false)
  else
    // ignore packet
  end if
else if packet already consumed and
  multiplicity=exactly-one then
  send CANCEL packet with parameter force:=true
end if

```

The value *limit* is a short delay (near zero). If it is set too small, the CANCEL packet may be sent too early, i. e., other candidate nodes may receive it before receiving the actual data packet. The flag *wait* indicates if the node is currently waiting to consume the message after sending a CANCEL packet. In some delivery semantics, the message can be consumed immediately after reaching a destination node; in other semantics it must be assured that only one node consumes it; thus, after sending a CANCEL packet, the node waits for a short additional time. If it receives a CANCEL packet from another node which is closer to the destination than itself, it will drop the message.

In *exactly-one* semantics, when a node already consumed the message and receives a CANCEL packet for it afterwards, the competing node has to be forced to cancel the message immediately. Thus, the node sends a CANCEL packet with parameter *force=true*, whereupon the other node drops the message.

When the destination timer expires, the following code is executed (*a* is the additional delay introduced in Section 5.1):

## 5 The Blind Geographic Routing Algorithm

```
if wait=true then
  consume packet
else
  broadcast CANCEL packet
  if multiplicity=exactly-one and
    (closeness=nearby or accept-outside=true) then
    schedule destination timer to expire
      after  $2a$  (wait:=true)
  else
    consume packet
  end if
end if
```

Finally, a node which receives a CANCEL packet performs the following steps:

```
if destination timer running for associated packet and
  multiplicity=exactly-one and
  (force=true or sender of CANCEL packet is closer
  to destination than this node) then
  cancel destination timer
end if
if contention timer running or recovery timer running
  for associated packet then
  cancel timer
end if
if multiplicity=exactly-one and
  packet has been consumed by this node then
  if force=false then
  send CANCEL packet (force:=true)
  else
  // other node has also consumed the same packet
  end if
end if
```

### 5.7 Geocasting

Geocasting is a variant of multicasting. A message is delivered to all nodes within a geographical area specified by the sender. This area is called *geocast region*. Initially,

geocasting was introduced as an integration into the Internet Protocol [NI97]. Later, it was adopted for ad hoc networks. A survey of early geocasting protocols can be found in [Mai04].

The first geocasting protocols for ad hoc networks used limited flooding. For instance, [KV02] operates with forwarding zones; nodes within the zone re-broadcast the packet, while nodes outside ignore it. These flooding-based protocols do not guarantee delivery, because the network graph within the geocast region may consist of several connected components that are connected with each other only through nodes outside the geocast region.

Three geocasting protocols that guarantee delivery under the unit disk graph model are presented in [Sto04]. Two of them are based on face routing. In the first algorithm, face routing is started by the nodes at the border of the geocast region. The second algorithm performs a depth-first search face tree traversal. In the third algorithm, the geocast region is divided into regions of diameter equal to the transmission range. The message is routed to the center of each region, from where limited flooding is started within the respective region.

BGR also supports geocasting. It is based on an intelligent variant of flooding. Face routing is not used, because it is not in line with the overall philosophy of BGR: face routing requires a unit disk graph and reliable unicast transmission. Since BGR has been designed to work with irregular propagation, transient communication failures, and unidirectional links, face routing is not acceptable. Additionally, face routing is not possible in 3D networks.

The geocasting scheme of BGR has not been designed with the goal of guaranteed delivery to all nodes within the geocast region. In the presence of transmission failures, this would only be achievable with high communication overhead. BGR, however, raises a claim to be a low-communication protocol. Guaranteed delivery to all nodes is not considered essential, because dynamic topology changes are common anyway in ad hoc networks. Thus, the strategy of BGR is conforming to the overall characteristics of ad hoc networks.

Since the geocasting scheme of BGR is flooding-based, it requires a medium to high network density. If the geocast region consists of several connected components, the message will be spread in only one of them, unless abnormal effects such as multiple paths occur caused by irregular propagation.

The scheme works as follows: When a geocast message is to be sent, the sender puts a description of the geocast region into the packet header and sets the center of this area as destination location, which is used to route the message toward the geocast region. The center can be determined using any reasonable measure, such as the center of gravity. If the center is located outside the geocast region, which is possible when the region is not convex, another point within the geocast region has to be set as destination. The choice of this point is up to the sender.

If the sender is not located within the geocast region, the message is routed toward the destination location designated in the message header until it reaches a node

## 5 The Blind Geographic Routing Algorithm

within the geocast region. At this node (or these nodes, if more than one node within the geocast region receive the message), the geoflooding phase starts. If the sender is already located within the geocast region, the geoflooding phase starts immediately. Note that the delivery semantics discussed in Chapter 4 are irrelevant for geocasting.

A node where geoflooding within the geocast region begins, broadcasts the message immediately, without setting any timers. This is done so that other nodes, which are not located within the geocast region, cancel their timers and do not forward the message anymore. The packet is broadcast in the mode GEOFLOOD.

A node that receives this GEOFLOOD broadcast checks if it is also located within the geocast region. If this is not the case, the packet is ignored. Otherwise, a contention timer is started according to the contention timer function already discussed. When this timer expires, the node continues geoflooding; when the node receives another GEOFLOOD packet in the meantime, the timer is canceled. This reduces communication, because not all nodes within the geocast region flood the message. The risk is that some nodes may not receive the message. In high-density networks, however, this is unlikely to occur.

### 5.8 Duty Cycles

Since conventional transceivers consume approximately as much energy in idle mode as in sending or receiving mode, it is important for energy efficiency to turn off the transceiver periodically. The sleep and awake phases are called *duty cycles*. Prominent examples for duty cycle protocols are GAF [XHE01] and STEM [STS02]. In GAF, the network is divided into virtual grids that are small enough so that all nodes within a grid can communicate with each other. Only one node in a grid needs to be awake to perform routing tasks. STEM uses a separate radio channel with a low duty cycle to perform wakeup. The data channel is turned off when inactive.

The biggest issue of GAF is that it is not prepared for irregular transmission ranges and transient link failures. It is completely based on the unit disk graph model. The redundancy of nodes, which helps against transient link failures, is neutralized by turning off all nodes except one in each grid. STEM, on the other hand, needs special hardware with a dual radio.

As discussed in Section 2.2, modern transceivers have a wake-on-radio feature, which eliminates the need for application-controlled duty cycles. There may be cases where duty cycles are desired, e. g., when wake-on-radio is not available. The infrastructureless operation of BGR makes synchronization of duty cycles among the nodes unnecessary. Thus, every node can control its own duty cycles independent from the cycles of the other nodes. When a packet is broadcast at a time when nodes within the forwarding area/volume are sleeping, these nodes do not participate in the contention process. This is the same as if these nodes had not received the packet due to transmission failures, and it does not lead to a failure of the algorithm if the

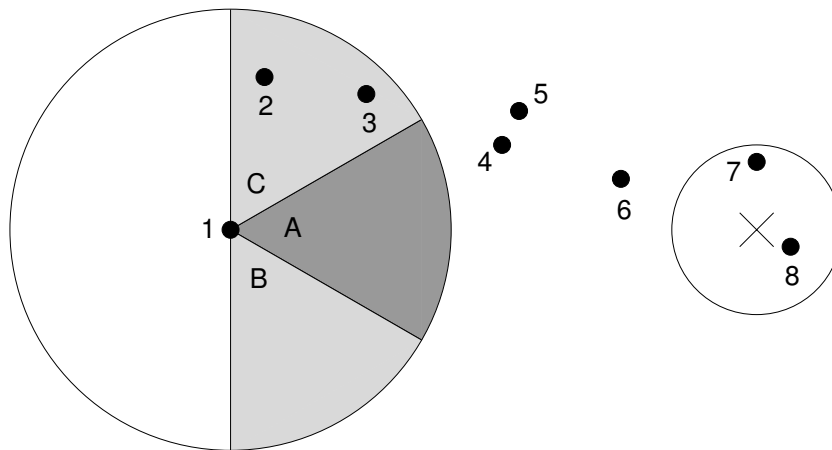


Figure 5.5: BGR forwarding example (explanation see text)

network is dense enough so that there are nodes within the forwarding area/volume that are awake at forwarding time. Of course, it is important that the sleeping intervals of the nodes do not overlap too much. This could be achieved by varying the cycle lengths in a randomized way. Another important thing is that a node must not sleep when it has a timer running, because it would then miss packets forwarded by another node.

## 5.9 Example

An example scenario demonstrates the functionality of the BGR algorithm. Figure 5.5 shows the topology; node 1 is the sender of the message, and the center of the right circle constitutes the destination location. The right circle itself comprises the destination area, i. e., the area which meets the *closeness* semantics that the sender chooses. The parameter *closeness* is set to *nearby*; the limit of tolerance is the radius of this circle. The parameter *multiplicity* is set to *exactly-one*. The value of *accept-outside* is irrelevant in this example, since the destination area is not empty. The forwarding area is the sector.

Node 1 begins by broadcasting the message using Sector A as forwarding area. Nodes 2 and 3 receive the broadcast, but since they are not located within the forwarding area, they ignore it. Because Sector A is empty, no node will respond, so eventually the recovery timer of node 1 expires. It turns the forwarding area by  $60^\circ$  to the right or left randomly. Assume that the former possibility is chosen, then Sector B becomes forwarding area and the message is broadcast again. Since this area is also empty, the recovery timer expires again and node 1 broadcasts the message a third time using Sector C. Now, nodes 2 and 3 receive the message and start contention timers depending on their distances to the destination. As node 3 is closer to the destination, its contention timer expires first and the node forwards the message. Nodes

## 5 The Blind Geographic Routing Algorithm

1 and 2 notice this forwarding and cancel their respective timers. Nodes 4 and 5 also receive the packet, and because they are located within the forwarding area, they start contention timers. Because they have nearly the same distance to the destination, their timers expire almost simultaneously, and both nodes forward the message. Node 6 receives both messages. Assume that it receives the message from node 4 first. It starts a contention timer. When it receives the message from node 5, it notices that an ASF situation has occurred, because the hop counts of both messages are the same. Therefore, it does not cancel its contention timer. If it had canceled the timer, the message would be dropped silently at this point.

When the contention timer of node 6 expires, it forwards the message again. Since both nodes 7 and 8 are potential destination nodes, they start destination timers depending on their distance to the destination location. As node 8 is closest, its timer expires first, and it broadcasts a CANCEL packet and starts a very short ( $2a$ ) destination timer ( $wait=true$ ). When this timer expires, it consumes the message. Nodes 6 and 7, which receive the CANCEL packet, cancel their timers, and the algorithm is completed.

## 6 Location Errors

Most existing geographic routing algorithms were initially designed for nodes with exact location information. Studies about performance in case of location errors were executed merely a posteriori for some algorithms; thus, these studies did not have influence on the design of the algorithms. However, it is important that geographic routing algorithms tolerate location errors, because they do occur in real ad hoc network deployments. In [SMP02], five main sources of location errors are identified: measurement, finite precision, objective function-specific, intractable optimization tasks, and localized algorithms. Studies of localization schemes show location errors up to the order of magnitude of the transmission range [LP04].

This chapter presents related performance studies of greedy and face routing in case of location errors. Afterwards, improvements for BGR and GPSR are developed based on analytical calculations and simulation experiments. Similar results are presented in [WT06b].

### 6.1 Related Work

An evaluation of greedy forwarding in case of location errors can be found in [HHB<sup>+</sup>03]. Through simulation, it was found out that delivery rate and path length remain acceptable up to location errors of about 40 % of the transmission range. Other modes than the greedy mode were not investigated.

Location errors in GPSR have been studied in [KLH04]. Greedy and perimeter mode were investigated separately. In plain greedy mode, a high packet drop rate due to false dead ends was observed. The drop rate increases with higher network density. Values up to 50 % were observed at location errors of  $0.2r$  in dense networks ( $r$  is the transmission range). Furthermore, the impact on the optimal path rate was investigated. The simulations showed that up to 53 % of the paths were non-optimal; these results, however, are not very significant, since they say little about the actual path lengths. It is a difference whether the path is merely 1 % or 100 % longer than the optimal path. A simple Boolean value (optimal or non-optimal path) is not enough for a clear understanding.

Regarding the perimeter mode, a phenomenon called *planar graph collapse* has been studied, which means that an edge is not removed due to location errors, but it should be. Since this is not the only possible planarization error, this analysis is not sufficient either. In perimeter mode, a packet drop rate up to 28 % was observed at location errors of  $0.2r$ .



A fix for GPSR in case of location errors has been proposed by Seada et al. [SHG04], who found out that most of the failures are due to incorrectly removed edges. Therefore, they proposed that, before a node  $u$  removes an edge  $(u, v)$ , it sends a message to  $v$ , who responds only if it also sees the neighbor  $w$ . Only when  $u$  receives a positive response, the edge is discarded. This modification results in a much higher success rate in their simulations. The position error in their simulations, however, is uniformly distributed between zero and the maximum error, which is not an appropriate model. For modeling errors, Gaussian distributions should be used.

Another study can be found in [SWR05]. Here, a geographic routing protocol is analyzed that uses greedy mode where possible and flooding to route around obstacles and voids. Hence, this protocol is very energy-consuming. Analytical computations and simulation runs reveal that performance starts dropping at location errors of about 20 % of the transmission range; when using two-hop neighborhood information, however, this can be improved up to 40 %. Unfortunately, they also use a uniform distribution of the location error.

MER (Maximum Expectation within transmission Range) [KS06] is a geographic routing algorithm that was explicitly optimized for networks with inaccurate location data. The greedy mode is changed in such a way that not the neighbor with maximum progress is selected, but the neighbor that maximizes a special objective function. The algorithm was designed for unit disk graph networks and minimizes the probability of incorrect backward progress.

### 6.2 Estimated Distance between Nodes

This section presents calculations that are needed in the subsequent sections, where BGR and GPSR are prepared for location errors.

In this work, the location errors in two-dimensional topologies are modeled following a two-dimensional Gaussian distribution  $\mathcal{N}^2(0, \sigma^2)$ . In three-dimensional topologies, a three-dimensional Gaussian distribution  $\mathcal{N}^3(0, \sigma^2)$  is used. This implies that the average distance between real and estimated location of each node follows a Rayleigh distribution in 2D and a Maxwell-Boltzmann distribution in 3D. The expected value is  $\sqrt{\pi/2} \sigma \approx 1.253 \sigma$  in 2D and  $\sqrt{8/\pi} \sigma \approx 1.596 \sigma$  in 3D, respectively.

In the following, the expected value for the estimated distance  $e$  between two nodes assuming the real distance  $d$  is calculated. This is first done for the 2D case. For the calculation, we assume that one node has an error-free location at  $(d, 0)$ ; the other node is located at  $(0, 0)$  with its estimated position  $(x, y)$  being distributed following the sum of the two Gaussian distributions of both nodes, which is  $\mathcal{N}^2(0, 2\sigma^2)$ .

The function  $g(x, y)$  calculates the estimated distance  $e$  (see Figure 6.1 on the next page):

$$g(x, y) = \sqrt{y^2 + (d - x)^2}.$$

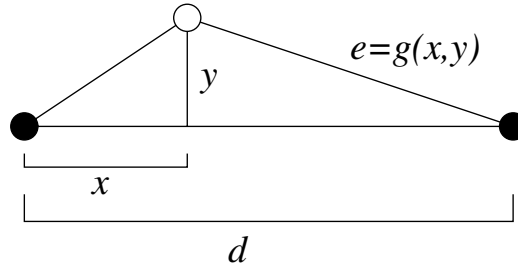


Figure 6.1: Real distance  $d$  and estimated distance  $e$  between two nodes (represented as black circles; the white circle denotes the estimated position of the left node)

The random variable  $G$  describes the estimated distance  $e$ ; the two-dimensional Gaussian probability density function is

$$f(x, y) = \frac{1}{4\pi\sigma^2} e^{-\frac{x^2+y^2}{4\sigma^2}}.$$

The expected value of  $G$  is

$$\begin{aligned} E(G) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) f(x, y) dx dy \\ &= \frac{1}{4\pi\sigma^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \sqrt{y^2 + (d-x)^2} e^{-\frac{x^2+y^2}{4\sigma^2}} dx dy. \end{aligned}$$

Unfortunately, this integral cannot be solved analytically. But if  $d$  and  $\sigma$  are fixed, the value can be calculated numerically. The only necessary value in the next sections is the result for  $d = r$ , thus the value can be stored in the nodes as a constant.

In the three-dimensional case, the estimated distance  $e$  is calculated with the function  $h(x, y, z)$ :

$$h(x, y, z) = \sqrt{y^2 + z^2 + (d-x)^2}.$$

The estimated distance  $e$  is described by the random variable  $H$ ; the three-dimensional Gaussian probability density function is

$$f_{3D}(x, y, z) = \frac{1}{8\pi^{\frac{3}{2}} \sigma^3} e^{-\frac{x^2+y^2+z^2}{4\sigma^2}}.$$

Table 6.1: Numerically calculated values of  $E(G)$  and  $E(H)$  assuming  $d = 40$ 

$\sigma$	5	10	15	20	25	30	35	40
$E(G)$	40.63	42.60	46.18	51.28	57.47	64.38	71.77	79.49
$E(H)$	41.25	45.00	51.07	58.86	67.75	77.31	87.32	97.61

The expected value of  $H$  is

$$\begin{aligned}
 E(H) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, y, z) f_{3D}(x, y, z) \, dx \, dy \, dz \\
 &= \frac{1}{8\pi^{\frac{3}{2}} \sigma^3} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \sqrt{y^2 + z^2 + (d-x)^2} e^{-\frac{x^2+y^2+z^2}{4\sigma^2}} \, dx \, dy \, dz.
 \end{aligned}$$

Again, this integral can only be solved numerically. Table 6.1 lists the numerically calculated values of  $E(G)$  and  $E(H)$  in case  $d = 40$ .

### 6.3 Preparing BGR for Location Errors

This section explains how BGR is prepared to operate in case of location errors. The standard deviation  $\sigma$  of the error is needed; it can be estimated if the exact value is not available.

The results from the previous section indicate that, on average, the calculated distance between two nodes is higher than the real distance. This means that a significant number of nodes falsely assumes to be located outside the forwarding area/volume when they are in fact within it. The solution for this problem is to stretch the assumed transmission range. This is done by enlarging the forwarding areas/volumes: Not the transmission range  $r$  is taken as width  $w$  of the forwarding areas/volumes, but the estimated distance of two hypothetical nodes whose distance is equal to  $r$ . This value is to be calculated numerically as shown in the previous section. It can be calculated once when configuring the network and stored in the nodes as a constant, so the nodes do not need to perform any calculations to get this value. The enlargement of the forwarding areas/volumes also leads to a better dispersion of the calculated delays.

Because in case of location errors it is impossible to guarantee that all nodes within the forwarding area/volume notice each other, multi-path flows can occur when more than one node forwards the message. Another problem is that the message is not necessarily routed in the direction toward the destination. This leads to detours, which can result in total delivery failures in the worst case. However, this problem is not

specific to BGR; all geographic routing algorithms suffer from it when location information is inaccurate.

## 6.4 Preparing GPSR for Location Errors

The GPSR fix from Seada et al. [SHG04] for networks with location errors has already been discussed in Section 6.1. An attempt to adopt this fix in the original ns-2 GPSR implementation revealed that it is not applicable because of far too many packet collisions, even with random backoff delay in the order of magnitude of one minute. The problem is that GPSR triggers a re-planarization every time a new neighbor is detected. Therefore, in the network setup phase, in which new neighbors are added within a short period of time, planarization is performed so often that the number of messages generated by this fix is enormous. Seada et al. used an ideal MAC and physical layer without packet loss, hence they did not face this problem.

To make it applicable, the fix from Seada et al. can be modified: No packets are exchanged during the planarization phase; instead, the planarization is done using two-hop neighborhood information. The IDs of the neighbors are added to the periodically sent beacon messages. When receiving a beacon, not the entire neighborhood of this neighbor needs to be stored, but only those which are located within the Gabriel circle. This reduces memory requirements.

Originally, a re-planarization is triggered when either a new node is detected or a node has been removed from the neighborhood. This policy is not sufficient anymore: re-planarization must be done when the neighborhood of a neighbor has changed. So the following must be added to the fix in order to make it work correctly: When the neighborhood of a node changes, a flag is set in the next three beacon messages that forces the receiving nodes to re-planarize. (The flag is sent multiple times, as beacons can get lost. The number of three was chosen, because this is the number of beacons that must have been missed in order to remove a node from the neighborhood table.) For this period of three beacons, the use of implicit beacons is disabled. Implicit beacons are regular data packets that are regarded as beacons so that the scheduling of the next regular beacon is delayed, but since they do not contain neighborhood information, they cannot be used during the planarization phase. This leads to a small increase of the total number of packets.

The proposed fix has two shortcomings:

1. More memory space is required to store two-hop neighborhood information.
2. The beacon messages are much longer, which leads to more energy consumption and delivery failures.

To solve both issues, a probabilistic approach can be applied: When a node  $u$  sees a node  $w$  within the Gabriel circle  $(u, v)$  (see Figure 3.2 on page 14), the decision whether the link  $(u, v)$  should be removed is based on the question whether node  $v$

## 6 Location Errors

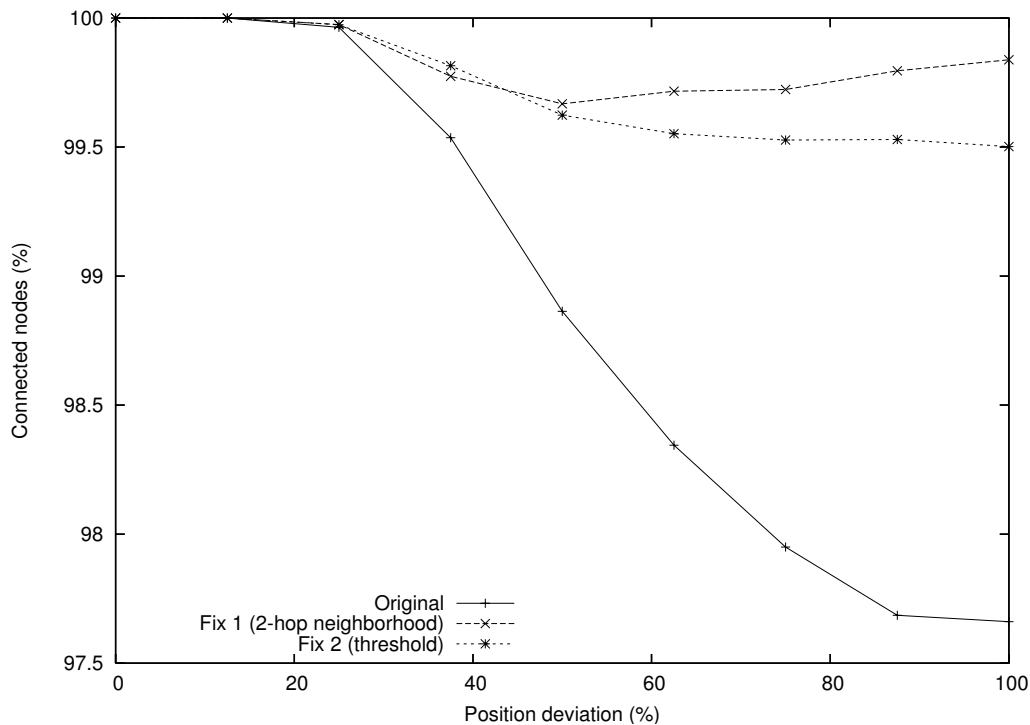


Figure 6.2: Average percentage of fully connected nodes after planarization; position deviation plotted as fraction of transmission range

also sees node  $w$ . This is the case when the real distance between  $v$  and  $w$  is at most the transmission range  $r$ . Thus, the threshold value  $t$  for removing links is the estimated value of  $r$ , again calculated numerically as shown in Section 6.2. The decision of node  $u$  whether to remove the link to node  $v$  is based on the estimated distance  $e$  between  $v$  and  $w$ . The link is removed if  $e \leq t$ . Since node  $u$  knows that node  $v$  will remove the link only if the distance between  $u$  and  $w$  is not above  $t$ , the link is kept when this distance is above  $t$ . Thus, the link is only removed if both the distance between  $u$  and  $w$  and between  $v$  and  $w$  do not exceed the threshold  $t$ . This leads to a more consistent planarization.

Statistical analysis revealed that, unlike claimed in [SHG04], the problem of disconnection due to incorrect edge removal is *not* the main problem of location errors in face routing. To show this, the planarization of 1000 randomly generated topologies was analyzed. In Figure 6.2, the average percentage of fully connected nodes after the planarization phase is depicted. These results show that only a few nodes are isolated. Moreover, the two fixes increase the number of connected nodes.

Instead, the main problem is intersection of links. Figure 6.3 on the next page depicts the average number of intersections per planar link. Note that the two fixes described so far result in more planar links than the original GPSR, because a condition is evaluated before a link is removed. Thus, more links are retained. To avoid

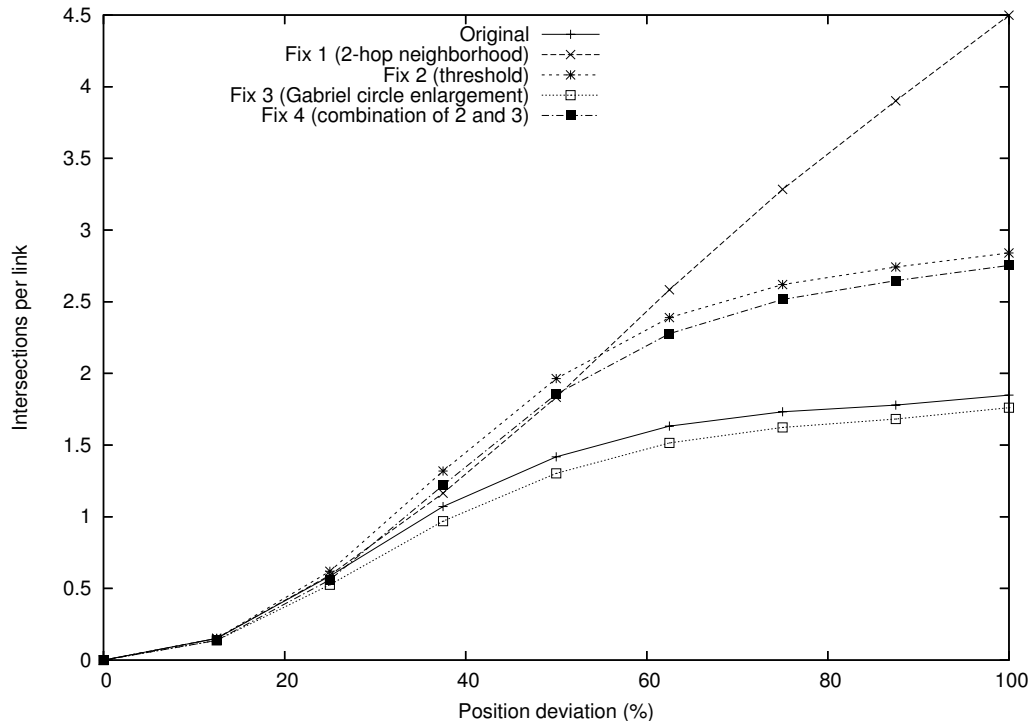


Figure 6.3: Average number of intersections per planar link

the intersection of links, however, *fewer* links should be retained. This is achieved through a third fix, which enlarges the Gabriel circle by  $\frac{\sigma}{2}$ . Simulation experiments showed that a further enlargement would lead to too many isolated nodes. This fix results in fewer link intersections, as can be seen from Figure 6.3.

The combination of fixes 2 and 3 leads to a fourth fix, which both enlarges the Gabriel circle and removes links based on the threshold. The intention of this fix is to combine the advantages of the previous fixes.

The performance of the fixes will be evaluated in Chapter 8.



## 7 Analytical Evaluation

This chapter presents probability calculations for BGR under the unit disk graph model (unit ball graph model in 3D). Formulas are constructed to approximate the probability of successful delivery for a given network density, transmission range, and source-destination distance. As forwarding area, the sector is used (spherical sector in 3D). Investigated recovery strategies are turning and augmentation of the sector.

In Sections 7.1 to 7.3, the expected progress within sectors of different angles and orientations is calculated. Section 7.4 presents formulas for the delivery probability using different recovery strategies (no recovery, turned sector, and augmented sector). Finally, in Section 7.5, the delivery probability in three-dimensional networks is calculated.

### 7.1 Expected Progress for Fixed Number of Nodes

First, the two-dimensional sector is considered. As a first step, the expected progress in a sector is calculated for a fixed number of nodes located within the sector. The results of this section are based on calculations from [HBBW04, HB03], but have been generalized for arbitrary sectors. Due to symmetry reasons, it is sufficient to consider only one half of the sector. Figure 7.1 on the next page shows a general half-sector of angle  $\beta$ . The x-axis points toward the destination; the transmission range is normalized to one. The sector  $\alpha$  does not belong to the forwarding area; thus, it represents a gap between the two half-sectors forming the forwarding area. This will be used for the augmented sector later. For the common  $60^\circ$  sector,  $\alpha$  is zero and  $\beta$  is  $30^\circ$ .

Progress is defined as the x value of the node within the forwarding area that forwards the packet, i. e., the projection on the x-axis. If exactly one node is located within the sector  $\beta$ , the density function for the progress can be constructed by expressing the diameter of the sector at any point  $x$  as a function of  $x$  ( $\phi := \alpha + \beta$ ):

$$\tilde{f}(x) = \begin{cases} x(\tan \phi - \tan \alpha) & (0 \leq x \leq \cos \phi) \\ \sqrt{1-x^2} - x \tan \alpha & (\cos \phi < x \leq \cos \alpha) \\ 0 & (\text{otherwise}) \end{cases}$$



## 7 Analytical Evaluation

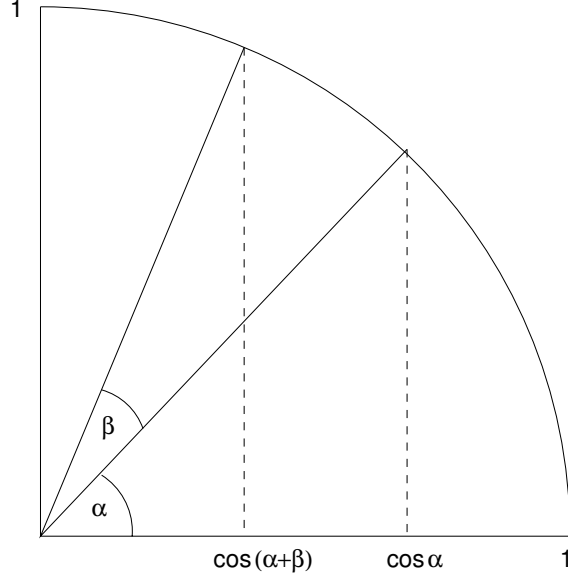


Figure 7.1: Sector of angle  $\beta$

For the density function  $f$ , the area of the sector must be normalized to one. Integration of function  $\tilde{f}$  yields:

$$\begin{aligned}
 A &= \int_0^{\cos \phi} x(\tan \phi - \tan \alpha) dx + \int_{\cos \phi}^{\cos \alpha} (\sqrt{1-x^2} - x \tan \alpha) dx \\
 &= (\tan \phi - \tan \alpha) \int_0^{\cos \phi} x dx + \int_{\cos \phi}^{\cos \alpha} \sqrt{1-x^2} dx - \tan \alpha \int_{\cos \phi}^{\cos \alpha} x dx \\
 &= \frac{1}{2} \phi - \frac{1}{2} \alpha = \frac{\beta}{2}
 \end{aligned}$$

The x-axis has to be stretched by the factor  $\sqrt{\frac{1}{A}} = \sqrt{\frac{2}{\beta}}$ . Hence, the density function  $f$  for packet progress within a normalized sector is:

$$f(x) = \begin{cases} x(\tan \phi - \tan \alpha) & (0 \leq x \leq \sqrt{\frac{2}{\beta}} \cos \phi) \\ \sqrt{\frac{2}{\beta}} - x^2 - x \tan \alpha & (\sqrt{\frac{2}{\beta}} \cos \phi < x \leq \sqrt{\frac{2}{\beta}} \cos \alpha) \\ 0 & (\text{otherwise}) \end{cases}$$

Suppose there are  $n$  nodes located within the half-sector, then the potential progress of the nodes can be expressed as independent and identically distributed (i. i. d.) random variables  $X_i (i \leq n)$ , which are distributed according to the density function  $f$ . The distribution function of each  $X_i$  is obtained by integrating  $f$ :

$$F_{X_i}(x) = \begin{cases} 0 & (x < 0) \\ \int_0^x t(\tan \phi - \tan \alpha) dt & (0 \leq x \leq \sqrt{\frac{2}{\beta}} \cos \phi) \\ \sqrt{\frac{2}{\beta}} \cos \phi & \\ \int_0^{\sqrt{\frac{2}{\beta}} \cos \phi} t(\tan \phi - \tan \alpha) dt & \\ + \int_{\sqrt{\frac{2}{\beta}} \cos \phi}^x (\sqrt{\frac{2}{\beta}} - t^2 - t \tan \alpha) dt & (\sqrt{\frac{2}{\beta}} \cos \phi < x \leq \sqrt{\frac{2}{\beta}} \cos \alpha) \\ \sqrt{\frac{2}{\beta}} \cos \phi & \\ 1 & (\text{otherwise}) \end{cases}$$

$$= \begin{cases} 0 & (x < 0) \\ \frac{1}{2}x^2(\tan \phi - \tan \beta) & (0 \leq x \leq \sqrt{\frac{2}{\beta}} \cos \phi) \\ \frac{x\beta\gamma \cos \alpha + 2 \cos \alpha \arctan \frac{x}{\gamma} - x^2 \phi \sin \alpha - 2 \cos \alpha(\frac{\pi}{2} - \phi)}{2\beta \cos \alpha} & (\sqrt{\frac{2}{\beta}} \cos \phi < x \leq \sqrt{\frac{2}{\beta}} \cos \alpha) \\ 1 & (\text{otherwise}) \end{cases}$$

with  $\gamma := \sqrt{\frac{2}{\beta} - x^2}$ .

The distribution of the maximum of the  $X_i$  is  $F_{\max_{i \leq n} X_i}(x) = (F_{X_i}(x))^n$ . The expected value, which describes the expected progress in the presence of  $n$  nodes, is

$$\begin{aligned}
 E(\max_{i \leq n} X_i) &= \int_0^{\infty} (1 - F_{\max_{i \leq n} X_i}(x)) dx - \underbrace{\int_{-\infty}^0 F_{\max_{i \leq n} X_i}(x) dx}_0 \\
 &= \int_0^{\infty} (1 - (F_{X_i}(x))^n) dx \\
 &= \int_0^{\sqrt{\frac{2}{\beta}} \cos \alpha} 1 dx - \int_0^{\sqrt{\frac{2}{\beta}} \cos \alpha} (F_{X_i}(x))^n dx \\
 &= \sqrt{\frac{2}{\beta}} \cos \alpha - \int_0^{\sqrt{\frac{2}{\beta}} \cos \phi} \left( \frac{1}{2}x^2(\tan \phi - \tan \beta) \right)^n dx \\
 &\quad - \int_{\sqrt{\frac{2}{\beta}} \cos \phi}^{\sqrt{\frac{2}{\beta}} \cos \alpha} \left( \frac{x\beta\gamma \cos \alpha + 2 \cos \alpha \arctan \frac{x}{\gamma} - x^2 \phi \sin \alpha - 2 \cos \alpha(\frac{\pi}{2} - \phi)}{2\beta \cos \alpha} \right)^n dx.
 \end{aligned}$$

## 7 Analytical Evaluation

These integrals are not solvable analytically, therefore the values have to be computed using numerical integration. Note that the expected progress is not normalized, i. e., the value is between 0 and  $\sqrt{\frac{\beta}{2}}$ . To normalize it, it has to be multiplied with  $\sqrt{\frac{\beta}{2}}$ .

### 7.2 Expected Progress in Recovery Sector

The expected progress in the recovery sector (turned by  $60^\circ$ ) is a special case, because it is not symmetric about the x-axis. It can be computed similarly to Section 7.1. The function for calculating the diameter of the sector is

$$\tilde{f}(x) = \begin{cases} \sqrt{1-x^2} - \frac{1}{\sqrt{3}}x & (0 \leq x \leq \frac{\sqrt{3}}{2}) \\ 0 & (\text{otherwise}) \end{cases}$$

Integrating this function yields

$$\int_0^{\frac{\sqrt{3}}{2}} (\sqrt{1-x^2} - \frac{1}{\sqrt{3}}x) dx = \frac{\pi}{6}.$$

The density function is obtained by stretching the x-axis by  $\sqrt{\frac{6}{\pi}}$ :

$$f(x) = \begin{cases} \sqrt{\frac{6}{\pi} - x^2} - \frac{1}{\sqrt{3}}x & (0 \leq x \leq \frac{3}{\sqrt{2\pi}}) \\ 0 & (\text{otherwise}) \end{cases}$$

Integrating  $f$  leads to the distribution function:

$$F_{X_i}(x) = \begin{cases} 0 & (x < 0) \\ \frac{1}{2}x\sqrt{\frac{6}{\pi} - x^2} + \frac{3\arcsin\frac{\sqrt{6\pi}}{6}x}{\pi} - \frac{\sqrt{3}}{6}x^2 & (0 \leq x \leq \frac{3}{\sqrt{2\pi}}) \\ 1 & (\text{otherwise}) \end{cases}$$

The expected progress is calculated according to Section 7.1 and similarly yields integrals not analytically solvable.

### 7.3 Expected Progress as Function of Network Density

In this section, the expected progress is calculated as a function of the network density  $d$  and the transmission range  $r$ . This can also be found in [HBBW04, HB03].

The nodes are assumed to be distributed according to a homogeneous Poisson point process, so the probability that there are exactly  $k$  nodes within a sector consisting of two half-sectors of angle  $\beta$  is

$$P(X = k) = e^{-d\beta r^2} \frac{(d\beta r^2)^k}{k!}$$

and the probability that the sector is empty is

$$P(X = 0) = e^{-d\beta r^2}.$$

Together with the results from Section 7.1, this leads to the expected progress depending on  $d$  and  $r$ :

$$\tilde{EP}(d, r) = \sqrt{\frac{\beta}{2}} e^{-d\beta r^2} \sum_{k=1}^{\infty} \frac{(d\beta r^2)^k}{k!} E(\max_{i \leq k} X_i).$$

More important for the following calculations is the expected progress under the condition that the sector contains at least one node, which is obtained by dividing this value by the probability that the sector is not empty:

$$EP(d, r) = \frac{\sqrt{\frac{\beta}{2}} e^{-d\beta r^2}}{1 - e^{-d\beta r^2}} \sum_{k=1}^{\infty} \frac{(d\beta r^2)^k}{k!} E(\max_{i \leq k} X_i).$$

## 7.4 Probability of Successful Delivery

### 7.4.1 No Recovery

For a given source-destination distance  $z$ , the probability that a message reaches the destination can be approximated with the following recursive formula:

$$P(z, d, r) = \begin{cases} 1 & (z \leq r) \\ (1 - e^{-d\frac{\pi}{6}r^2}) \cdot P(z - r \cdot EP(d, r), d, r) & (\text{otherwise}) \end{cases}$$

If the destination is located within the transmission area of the source, the delivery probability is equal to one. Otherwise, the delivery probability is calculated as the probability of a non-empty sector multiplied with the delivery probability of a potential node with the expected progress. This is not quite accurate, since the distance from this potential node to the destination can be slightly larger than  $z - r \cdot EP(d, r)$ , because the calculated progress is based on the projection to the source-destination line; it is, however, a valuable approximation especially for large values of  $z$  compared to  $r$ .

Another issue why this is to be regarded merely as an approximation is that only the expected progress is used for further calculations instead of the progress distribution.

The approximation of the delivery probability can also be calculated in a non-recursive way. The probability of successful delivery is the probability of a non-empty sector to the power of the number of hops, under the assumption that the progress of each hop is given by  $EP(d, r)$ . The closed formula, which is equivalent to the recursive one, is:

$$P(z, d, r) = \begin{cases} 1 & (z \leq r) \\ (1 - e^{-d\frac{\pi}{6}r^2})^{\lceil \frac{z-r}{r \cdot EP(d, r)} \rceil} & (\text{otherwise}) \end{cases}$$

### 7.4.2 Turned Sector

When using the turning of the sector as recovery strategy, the following recursive formula approximates the probability of successful delivery:

$$\begin{aligned}
 P_{\text{turn}}(z, d, r) &= \begin{cases} 1 & (z \leq r) \\
 (1 - e^{-d \frac{\pi}{6} r^2}) \cdot P_{\text{turn}}(z - r \cdot EP(d, r), d, r) \\
 + e^{-d \frac{\pi}{6} r^2} (1 - e^{-d \frac{\pi}{6} r^2}) \cdot P_{\text{turn}}(z - r \cdot EP_{\text{rec}}(d, r), d, r) \\
 + e^{-2d \frac{\pi}{6} r^2} (1 - e^{-d \frac{\pi}{6} r^2}) \cdot P_{\text{turn}}(z - r \cdot EP_{\text{rec}}(d, r), d, r) & (\text{otherwise}) \end{cases} \\
 &= \begin{cases} 1 & (z \leq r) \\
 (1 - e^{-d \frac{\pi}{6} r^2}) \cdot P_{\text{turn}}(z - r \cdot EP(d, r), d, r) \\
 + (e^{-d \frac{\pi}{6} r^2} - e^{-d \frac{\pi}{2} r^2}) \cdot P_{\text{turn}}(z - r \cdot EP_{\text{rec}}(d, r), d, r) & (\text{otherwise}) \end{cases}
 \end{aligned}$$

Here,  $EP_{\text{rec}}$  describes the expected progress within the turned recovery sector.

For the case  $z > r$ , the term consists of three sub-terms, according to the law of total probability. The first sub-term corresponds to the original sector and is identical to the non-recovery case covered in Section 7.4.1. The second sub-term corresponds to the first recovery sector, which is selected when the original sector is empty while the recovery sector is not. The third sub-term corresponds to the second turned sector; for this to be selected, both the original and the first recovery sector must be empty while at the same time the second recovery sector is not empty.

Unfortunately, a closed formula cannot be constructed in a similar way as for the non-recovery case, since this is a double-recursive formula.

### 7.4.3 Augmented Sector

This subsection presents results for the alternative recovery strategy, in which the sector is not turned but augmented in steps of  $30^\circ$  (both  $15^\circ$  to the left and to the right) up to four times. The expected progress is written as  $EP_{\alpha, \beta}$ , where  $\alpha$  and  $\beta$  are given as depicted in Figure 7.1 on page 60. Again, the formula is constructed according to the law of total probability:

$$P_{\text{aug}}(z, d, r) = \begin{cases} 1 & (z \leq r) \\
 (1 - e^{-d \frac{\pi}{6} r^2}) \cdot P_{\text{aug}}(z - r \cdot EP_{0, \frac{\pi}{6}}(d, r), d, r) \\
 + e^{-d \frac{\pi}{6} r^2} (1 - e^{-d \frac{\pi}{12} r^2}) \cdot P_{\text{aug}}(z - r \cdot EP_{\frac{\pi}{6}, \frac{\pi}{12}}(d, r), d, r) \\
 + e^{-d \frac{\pi}{4} r^2} (1 - e^{-d \frac{\pi}{12} r^2}) \cdot P_{\text{aug}}(z - r \cdot EP_{\frac{\pi}{4}, \frac{\pi}{12}}(d, r), d, r) \\
 + e^{-d \frac{\pi}{3} r^2} (1 - e^{-d \frac{\pi}{12} r^2}) \cdot P_{\text{aug}}(z - r \cdot EP_{\frac{\pi}{3}, \frac{\pi}{12}}(d, r), d, r) \\
 + e^{-d \frac{5\pi}{12} r^2} (1 - e^{-d \frac{\pi}{12} r^2}) \cdot P_{\text{aug}}(z - r \cdot EP_{\frac{5\pi}{12}, \frac{\pi}{12}}(d, r), d, r) & (\text{otherwise}) \end{cases}$$

Table 7.1: Calculated delivery probability (low network density)

Source-dest. distance $z$ [m]	No recovery	Turned sector	Augmented sector
60	0.875	0.998	0.998
80	0.765	0.996	0.996
100	0.765	0.996	0.996
120	0.669	0.993	0.994
140	0.585	0.992	0.992
160	0.585	0.991	0.991
180	0.512	0.989	0.990

#### 7.4.4 Selected Values

This section presents values and graphs for selected network scenarios. Table 7.1 lists results for the parameters  $r = 40$  m and  $d = \frac{150}{246^2}$  nodes per  $m^2$ , which is a low density of 12.5 nodes per transmission area on average. The calculated delivery probability is very high when recovery is used, even at this low density. This shows that BGR provides high delivery rates, although it does not guarantee delivery.

It is also noticeable that the calculated probability is not strictly monotonic for variable  $z$  values. The function decreases stepwise. This is due to the recursiveness of the function and is especially evident in the closed formula, where the ceiling function is used.

Figure 7.2 on the following page shows the probability functions for different recovery strategies as graphs. The network density is  $d = \frac{150}{224^2}$  nodes per  $m^2$ , which corresponds to 15 nodes per transmission area. Since the graphs for turned and augmented sector are nearly identical, they are repeated separately in Figure 7.3 on the next page. The stepwise decrease is especially evident in these graphs.

It can be concluded that recovery is essential to ensure a high delivery ratio. With recovery turned on, regardless of the actual strategy, BGR offers a high delivery probability even for large source-destination distances.

## 7.5 Delivery Probability for Spherical Sector

In the following, the delivery probability in 3D topologies is studied. The density function for the expected progress in a spherical sector is constructed by calculating the area of all points within the spherical sector that have equal progress. Using the

## 7 Analytical Evaluation

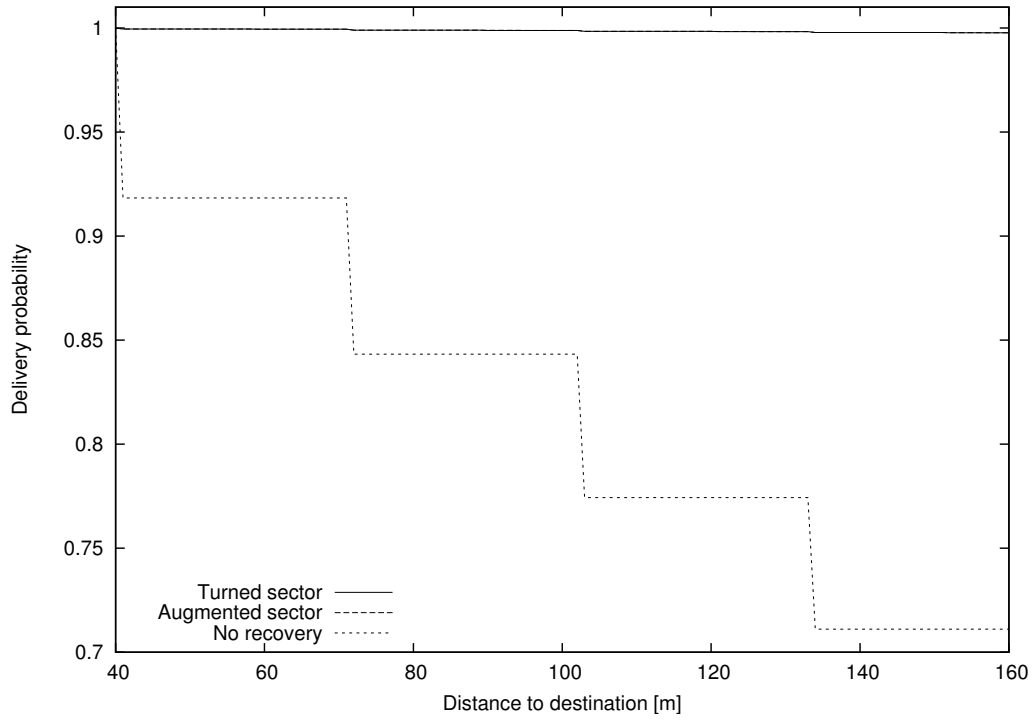


Figure 7.2: Delivery probability in 2D for different recovery strategies (medium network density); curves for turned and augmented sector are nearly identical

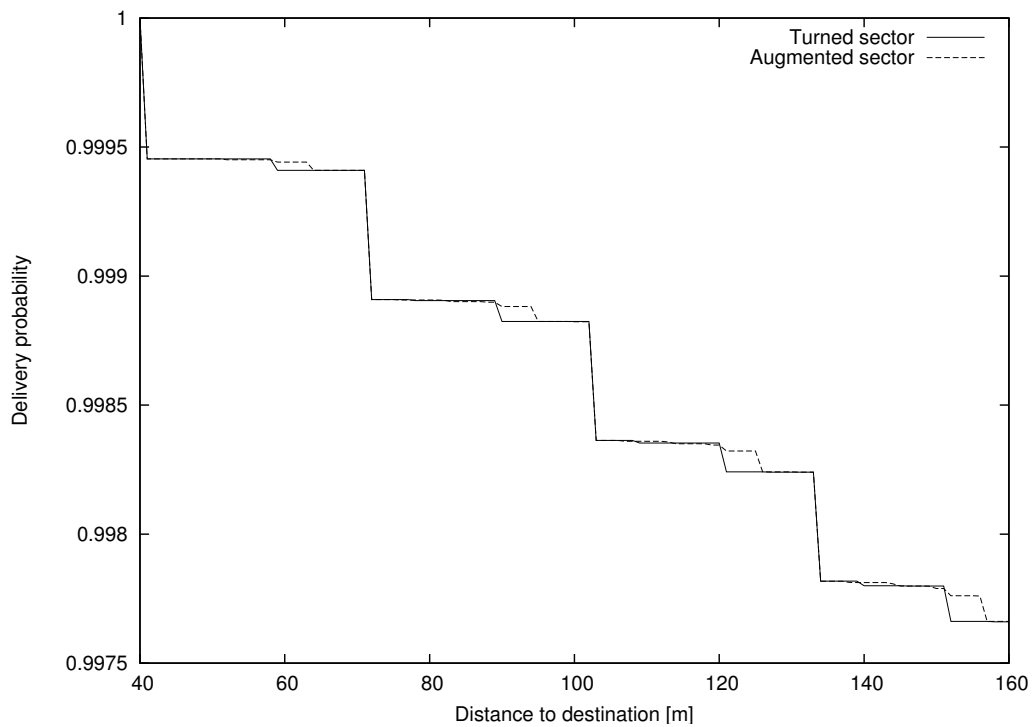


Figure 7.3: Delivery probability in 2D for turned and augmented sector

parameters in Figure 7.1 on page 60 ( $\phi := \alpha + \beta$ ), the density function is:

$$f(x) = \begin{cases} \pi x^2 (\tan^2 \phi - \tan^2 \alpha) & (0 \leq x \leq R \cos \phi) \\ \pi((R^2 - x^2) - x^2 \tan^2 \alpha) & (R \cos \phi < x \leq R \cos \alpha) \\ 0 & (\text{otherwise}) \end{cases}$$

The value of  $R$  is obtained by normalizing the integral:

$$\begin{aligned} \pi(\tan^2 \phi - \tan^2 \alpha) \int_0^{R \cos \phi} x^2 dx + \pi \int_{R \cos \phi}^{R \cos \alpha} ((R^2 - x^2) - x^2 \tan^2 \alpha) dx &= 1 \\ \iff \frac{2}{3} \pi R^3 (\cos \alpha - \cos \phi) &= 1 \end{aligned}$$

This equation has two complex and one real solution for  $R$ . The real solution is

$$R = \frac{1}{2} \sqrt[3]{\frac{12}{\pi(\cos \alpha - \cos \phi)}}.$$

The expected progress is calculated numerically according to Section 7.1.

For the turned recovery sector, the density function is:

$$\begin{aligned} f(x) &= \begin{cases} \pi(R^2 - x^2) - \frac{\pi}{3}x^2 & (0 \leq x \leq \frac{\sqrt{3}}{2}) \\ 0 & (\text{otherwise}) \end{cases} \\ &= \begin{cases} \pi(R^2 - \frac{4}{3}x^2) & (0 \leq x \leq \frac{\sqrt{3}}{2}) \\ 0 & (\text{otherwise}) \end{cases} \end{aligned}$$

Normalizing the integral yields  $R = \sqrt[6]{3}/\sqrt[3]{\pi}$ . The rest is straightforward.

The recursive formulas for calculating the delivery probability require the volume of the spherical sector and the augmented pieces. The values are listed in Table 7.2 on the next page.

The delivery probabilities for the different strategies are depicted in Figure 7.4 on the following page. The network density is  $d = \frac{150}{139^3}$  nodes per  $\text{m}^3$ , which corresponds again to 15 nodes per transmission volume in order to make it comparable to the 2D results. In contrast to the 2D case, the augmented sector yields much better results than the turned sector. This is because the augmented sector strategy covers half of the transmission volume, whereas turning the sector covers only  $\frac{5}{2} - \frac{5}{4}\sqrt{3} \approx 33.5\%$  of the transmission volume. In 2D, both strategies cover half of the transmission area. These results suggest that the augmented spherical sector should be used in 3D. With this forwarding volume, a high delivery probability is ensured even for large distances.



## 7 Analytical Evaluation

Table 7.2: Volume of different 3D shapes (augmented sector: volume of augmented pieces, no overlapping between these shapes)

Shape	Volume
Spherical sector	$\frac{1}{3}\pi r^3(2 - \sqrt{3})$
Augmented piece 1	$\frac{1}{3}\pi r^3(\sqrt{3} - \sqrt{2})$
Augmented piece 2	$\frac{1}{3}\pi r^3(\sqrt{2} - 1)$
Augmented piece 3	$\frac{1}{6}\pi r^3(2 - \sqrt{6} + \sqrt{2})$
Augmented piece 4	$\frac{1}{6}\pi r^3(\sqrt{6} - \sqrt{2})$

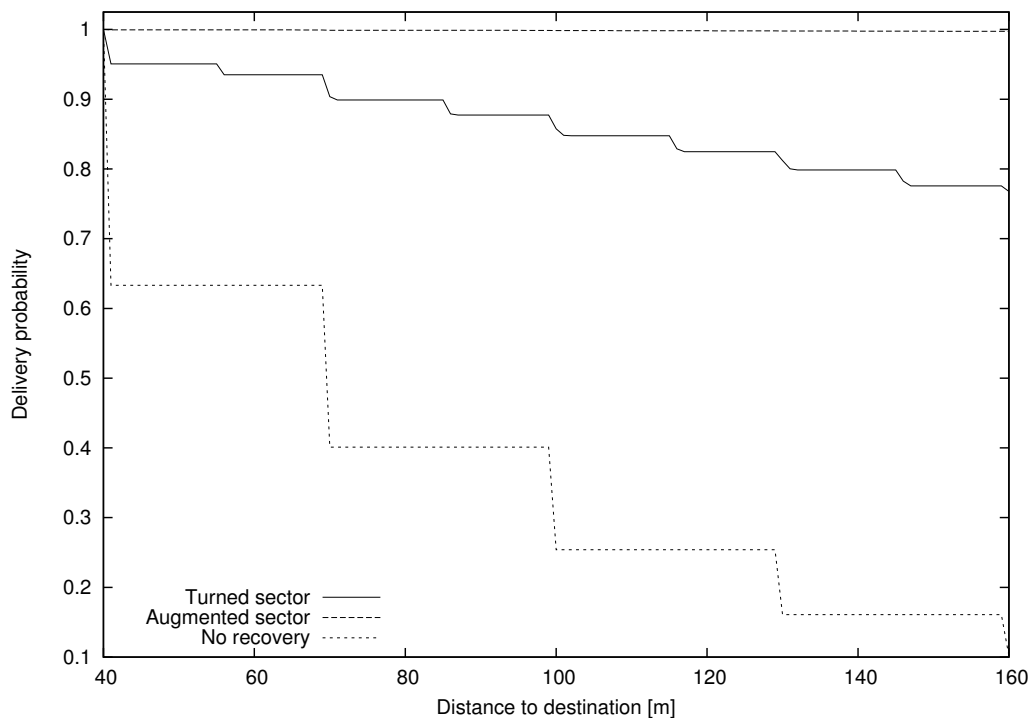


Figure 7.4: Delivery probability for spherical sector

## 7.6 Discussion

Although only an approximation, the presented formulas give valuable evidence to the performance of BGR. In concrete, they demonstrate that recovery is needed both in 2D and in 3D to ensure a high delivery rate. In 2D, the different recovery strategies both lead to a delivery ratio close to 1.0 in networks of medium node density; even in low-density networks, the delivery ratio remains high. In 3D, only the augmented spherical sector yields such outstanding results, because it covers a larger volume than the turned spherical sector.



## 8 Simulation Results

In this chapter, the performance of BGR is studied through extensive simulation. BGR has been implemented in the network simulator ns-2 [USC], which allows for realistic simulation of wireless nodes, including simulation of MAC and physical layers. IEEE 802.11 has been used as MAC protocol, because it is the only well-functioning built-in MAC protocol for wireless networks in ns-2 and is commonly used in wireless simulation studies with ns-2. It is a CSMA/CA scheme (carrier sense multiple access with collision avoidance) that exchanges RTS, CTS, Data, and ACK frames for unicast transmissions; for broadcasts, only the data is sent. The implementation performs up to three retransmissions (i. e., four tries altogether) in case of transmission failures for unicast packets.

For a comparison, the original implementation of GPSR used in [KK00] has been ported to a recent version of ns-2 (2.27). The Gabriel Graph was used for planarization. Variants of GPSR to cope with location errors and radio irregularity have also been implemented and simulated.

Each value represents the average of 20 simulation runs with different square (cubic in 3D) topologies. To make the results more comparable, the same topologies were used for all experiments at the same network density. All static topologies were connected.

Table 8.1 on the next page shows the relevant simulation parameters that are taken if not specified otherwise. The reason for taking 151 nodes is that in most scenarios, a designated sink node is placed at the center of the topology (to minimize border effects), so that 150 nodes remain, which are distributed randomly over the complete topology. Every 10 seconds, one node sends a message to the sink. Each node sends exactly one message, so 150 messages are sent altogether. A start phase of 100 seconds ensures that the neighborhood tables have been built before sending the first data message in GPSR. An end phase of 100 seconds assures that the simulation does not stop while a message is still on its way.

The network density is varied between 100 m<sup>2</sup> per node and 400 m<sup>2</sup> per node; i. e., lower values reflect a higher density. The values correspond to 2.5 to 10 nodes per 1000 m<sup>2</sup> or 11.57 to 49.27 neighbors per radio range on average. Networks with lower densities than these lead to frequent routing failures of any algorithm, as the probability of network disconnection increases [KWZ03]. The transmission range is not varied, because this is just another way of changing the number of neighbors.

The following metrics are investigated in the simulation experiments:

Table 8.1: Simulation parameters

Number of nodes	151
Transmission range $r$	40 m
Forwarding area	Reuleaux triangle
Timer function	linear
Maximum delay $m$	0.5 s
Additional delay $a$	0.05 s
Closeness	exact
Multiplicity	exactly-one
Accept-outside	false
Beacon interval (GPSR)	10 s

- The *delivery ratio* is the fraction of the number of generated messages that successfully reached the destination. In geocasting, it represents the percentage of the nodes within the geocast region that received the message.
- The *end-to-end delay* is the time between the source sending the message and the first destination node consuming the message.
- The *stretch* is the ratio between the actual and the optimal path length. The value is divided by the delivery ratio, because topologies with low delivery ratios often have low path lengths (messages on longer paths rarely reach the destination), which leads to a lower stretch value, as messages mostly take the optimal path. The incorporation of the delivery ratio annihilates this effect.
- The *packet overhead* represents the total number of packets sent per generated message on average. To route a message over  $h$  hops, at least  $h$  packets are needed. All packets sent by the routing algorithm are comprised in the packet overhead, including beacons.

## 8.1 Unit Disk Graph Model

In this section, the performance of BGR under the unit disk graph model (cf. Definition 2.2 on page 7) is evaluated. The transmission range  $r$  is fixed and there is no packet loss apart from collisions. This is the standard model implemented in ns-2. As a consequence of the fixed transmission range, all links are bidirectional. The fixed transmission range implies that in BGR, all nodes within a forwarding area/volume

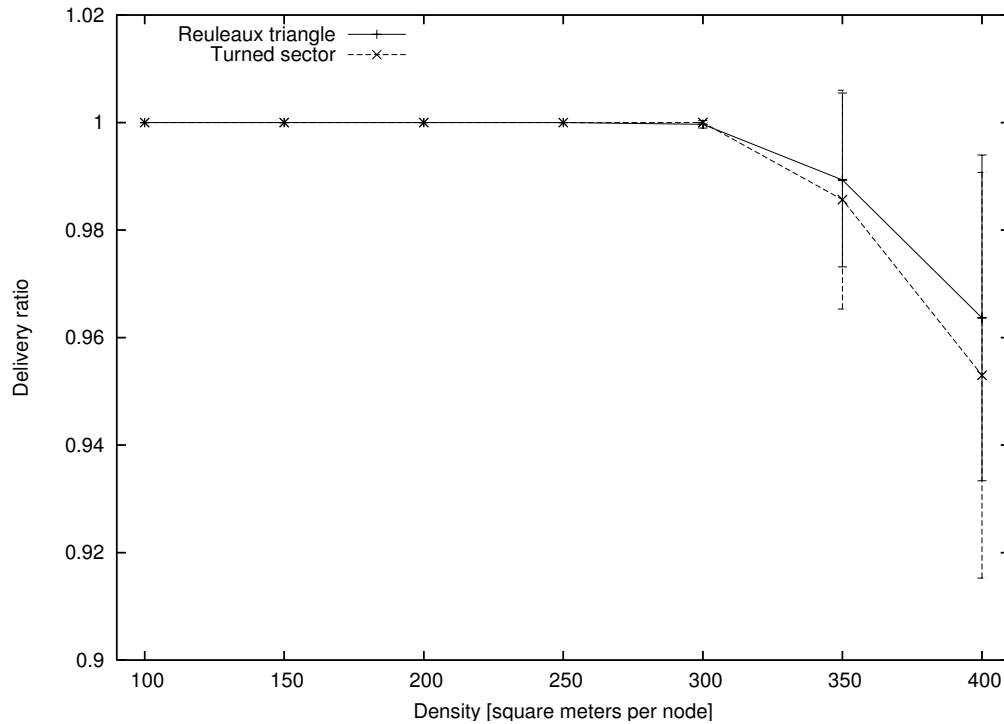


Figure 8.1: Delivery ratio for different forwarding areas

can mutually communicate with each other, with the exception of the augmented sector. This is somewhat idealistic, but leads to a better understanding of the general performance of BGR. It also makes the results comparable to performance studies of other protocols, because this model is used in almost all performance evaluations of wireless protocols; in many cases, it is even the only model that is used.

### 8.1.1 Forwarding Areas

First, the different forwarding areas of BGR are compared; these are sector, circle, and Reuleaux triangle. For the sector, two different recovery strategies are used: turning and augmenting. For comparison, the results for GPSR are also presented.

Figure 8.1 shows the average delivery ratio of BGR with Reuleaux triangle and sector (with turning as recovery strategy). The curves for other forwarding areas are omitted for readability reasons. The curve for the augmented sector is very similar to the one of the Reuleaux triangle, while the performance of the circle is between Reuleaux triangle and turned sector. The vertical bars represent confidence intervals of 95%. At medium and high network densities, the delivery ratio is equal to 100%. At low densities, where recovery is triggered frequently, the delivery ratio decreases slightly, but remains over 95%. The results are similar for all forwarding areas. The turned sector performs worse than the other forwarding areas in low-density networks

## 8 Simulation Results

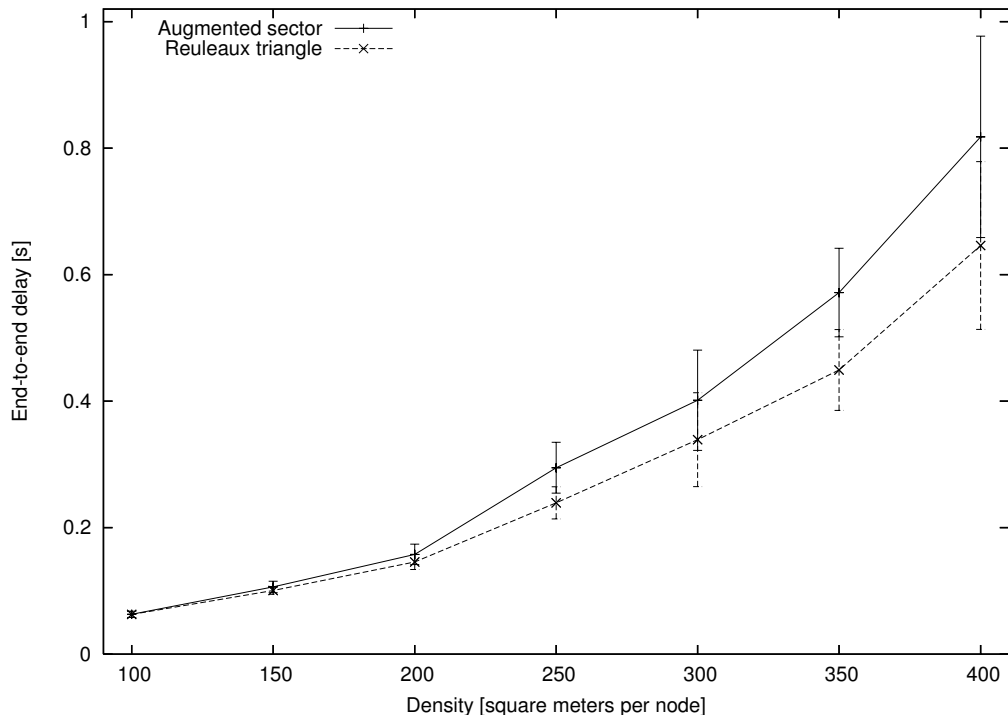


Figure 8.2: End-to-end delay

due to its small area: recovery is triggered more often than when using a larger forwarding area, and turning the sector can result in dead ends due to choosing the wrong direction. The results for GPSR are not shown, because GPSR provides a delivery ratio of 100 % under the unit disk graph model, if source and destination are connected. Only collisions with beacon packets can lead to delivery failures; this appeared just once in the simulations, which generated 21000 messages totally.

The average end-to-end delay for augmented sector and Reuleaux triangle is depicted in Figure 8.2. The curves for the other forwarding areas are similar to the curve for the augmented sector. The delay increases at lower network densities because recovery mode is triggered more often, and before a forwarding node enters recovery mode, it must wait for  $m + a = 0.55$  s. The shortest delay is achieved with the Reuleaux triangle, which apparently demonstrates the best combination of greedy and recovery mode. The end-to-end delay of GPSR is below 20 ms, which is an implication of its unicast scheme that does not use timers.

In Figure 8.3 on the next page, the stretch of the paths is depicted. The fact that the value is divided by the delivery ratio results in lower values for GPSR in sparse networks, since GPSR has a delivery ratio of 100 % under the unit disk graph model. In the comparison of the forwarding areas in BGR, the augmented sector performs best, the circle worst; the other forwarding areas are not depicted in the figure. Using the

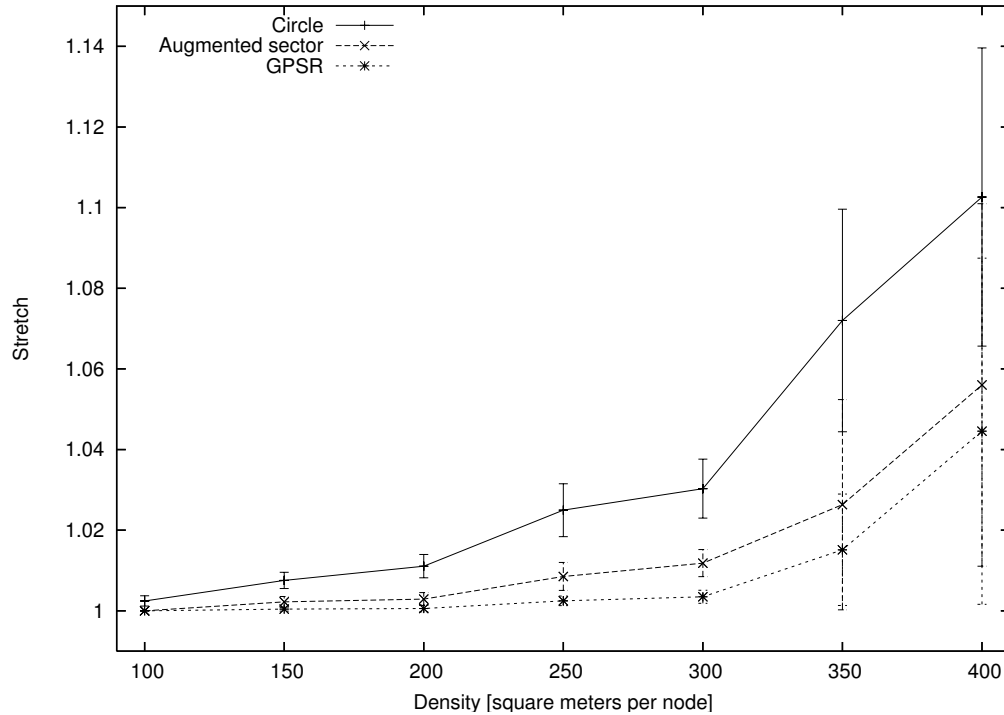


Figure 8.3: Stretch

circle, BGR takes longer paths than using the other forwarding areas, obviously because the circle contains fewer nodes with large progress (see condition 1 on page 19).

The packet overhead, which is the total number of packets sent per generated message, is similar for all forwarding areas. It is between 2.7 and 5.7, depending on the network density. At lower densities, the packet overhead is slightly larger, because recovery is triggered more often. In GPSR, the total number of packets depends heavily on the beacon interval. The number of packets for routing a message from source to destination is similar in BGR and GPSR; if only greedy mode is used and no abnormal situations like ASF occur,  $h$  packets are needed to route a message via  $h$  hops in GPSR, while BGR generates  $h + 1$  packets due to the extra CANCEL packet. Thus, unless using very high data communication rates, the beacon interval decisively influences the communication overhead of GPSR compared to BGR. The packet overhead of GPSR is between 171.8 and 173.1 in the simulations.

### 8.1.2 Timer Functions

In this section, the influence of different contention timer functions on the performance of BGR is examined. In Section 5.2, five timer functions have been proposed: square root, logarithmic, linear, exponential, and quadratic. The simulation experiments disclosed that the influence on the delivery ratio and the stretch is marginal.



## 8 Simulation Results

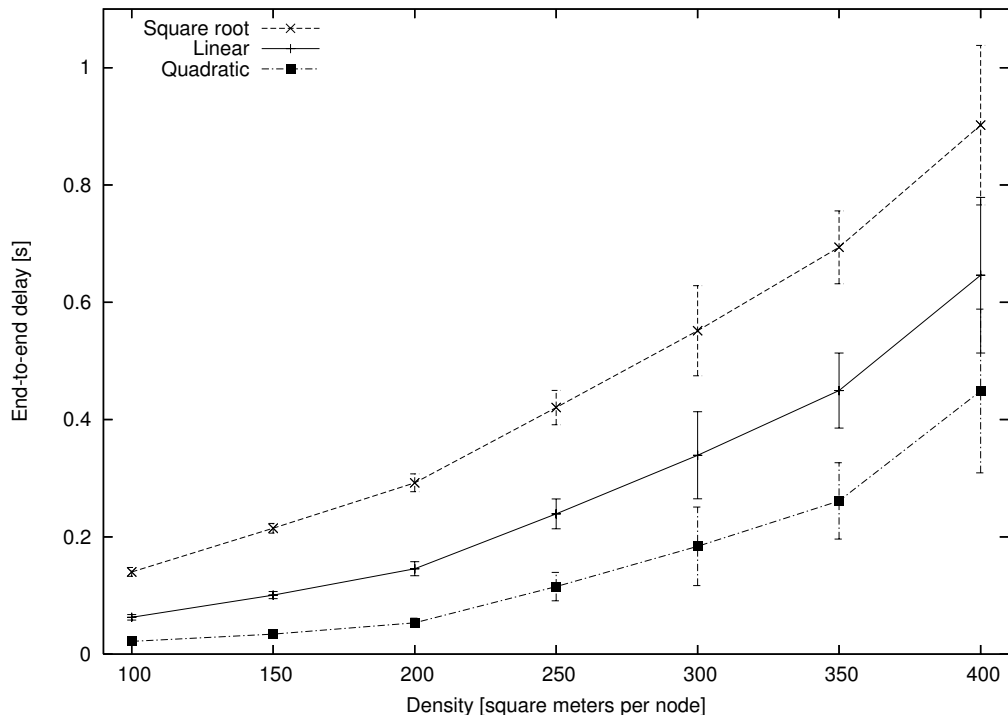


Figure 8.4: End-to-end delay for different timer functions

The end-to-end delay, however, is significantly different. It is shown in Figure 8.4 for three timer functions. As expected, the timer functions that compute smaller values result in shorter delays. The curves for the logarithmic and exponential functions are not shown; they run close to the curve for the linear function.

The packet overhead for different timer functions is depicted in Figure 8.5 on the facing page. For the timer functions not shown here, the curves are very similar to the curve for the linear function. Using the quadratic timer function, more packets are sent because of more simultaneous forwardings caused by similar timer values.

As a result, it can be said that the quadratic function offers faster message delivery at a slightly higher packet overhead. If fast delivery is not crucial, the linear function, which is the easiest one to compute, can be used. The other functions do not provide further benefit.

### 8.1.3 Recovery

Comparing BGR with and without recovery shows to what extent recovery improves the delivery ratio. The result is depicted in Figure 8.6 on page 78, using the Reuleaux triangle as forwarding area. It is evident that in dense networks recovery is not needed because the forwarding areas are never empty. At medium node densities, recovery is able to yield 100 % message delivery, whereas BGR without recovery frequently fails

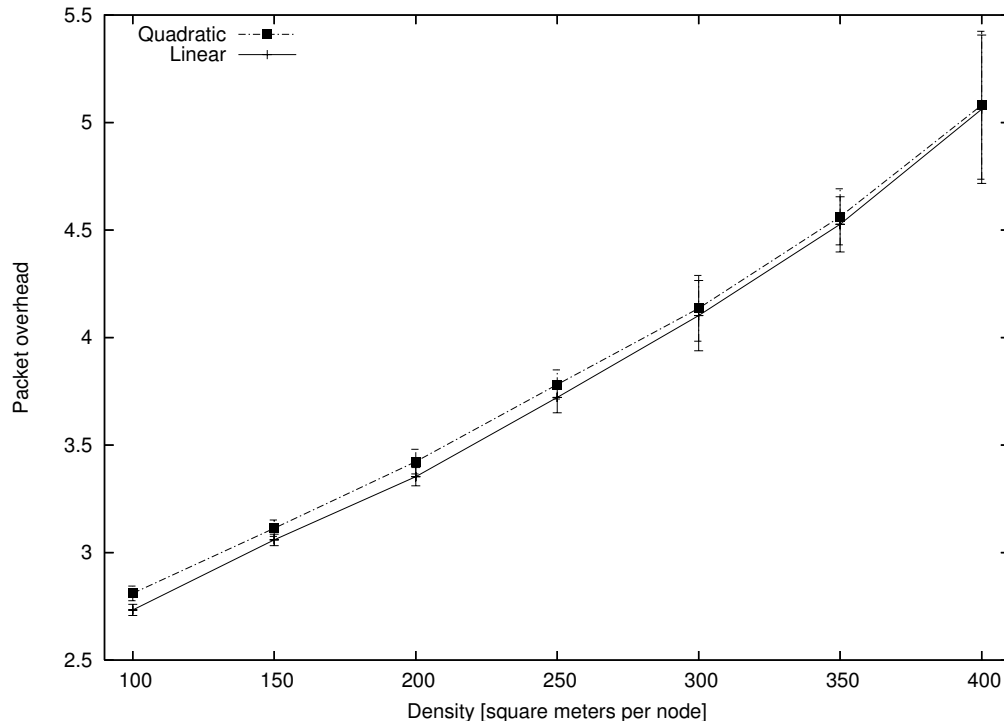


Figure 8.5: Number of packets per generated message for different timer functions

to deliver messages. At very low node densities, the delivery ratio drops under 80 % without recovery, while the delivery ratio of BGR with recovery remains over 95 %.

When using the sector as forwarding area, it is interesting to compare the two different recovery schemes. Figure 8.7 on the following page shows the results for augmented and turned sector. At medium and high node densities, both strategies yield 100 % message delivery. In low-density networks, the augmented sector leads to a slightly higher delivery ratio. Obviously, the risk of routing messages in directions with dead ends is higher when using the turned sector. The higher delivery ratio of the augmented sector comes with the cost of a higher packet overhead due to packet duplication, as illustrated in Figure 8.8 on page 79. In most situations, however, this small overhead is negligible.

An important question is if the recovery strategy of BGR is capable of routing messages around large radio obstacles. To test this, a high-density network of 100 m<sup>2</sup> per node with a large circular hole of diameter 80 m in the center has been simulated. The sink has been placed in the corner of the topology. The Reuleaux triangle served as forwarding area. All messages were successfully delivered, which demonstrates the robustness of the recovery scheme.

## 8 Simulation Results

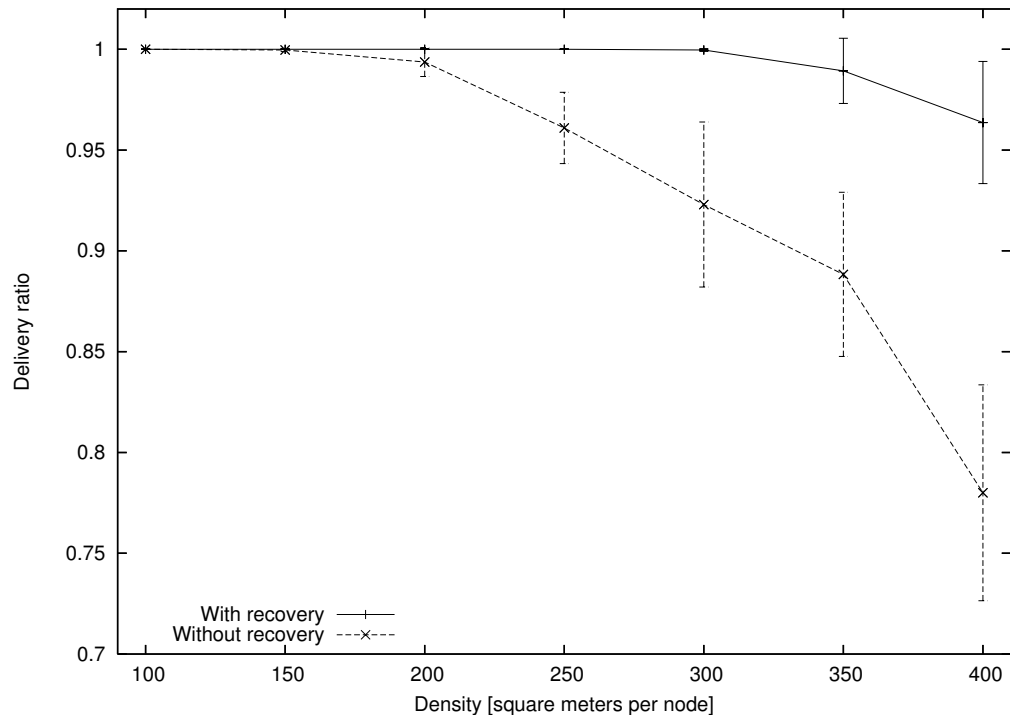


Figure 8.6: Delivery ratio with and without recovery (Reuleaux triangle)

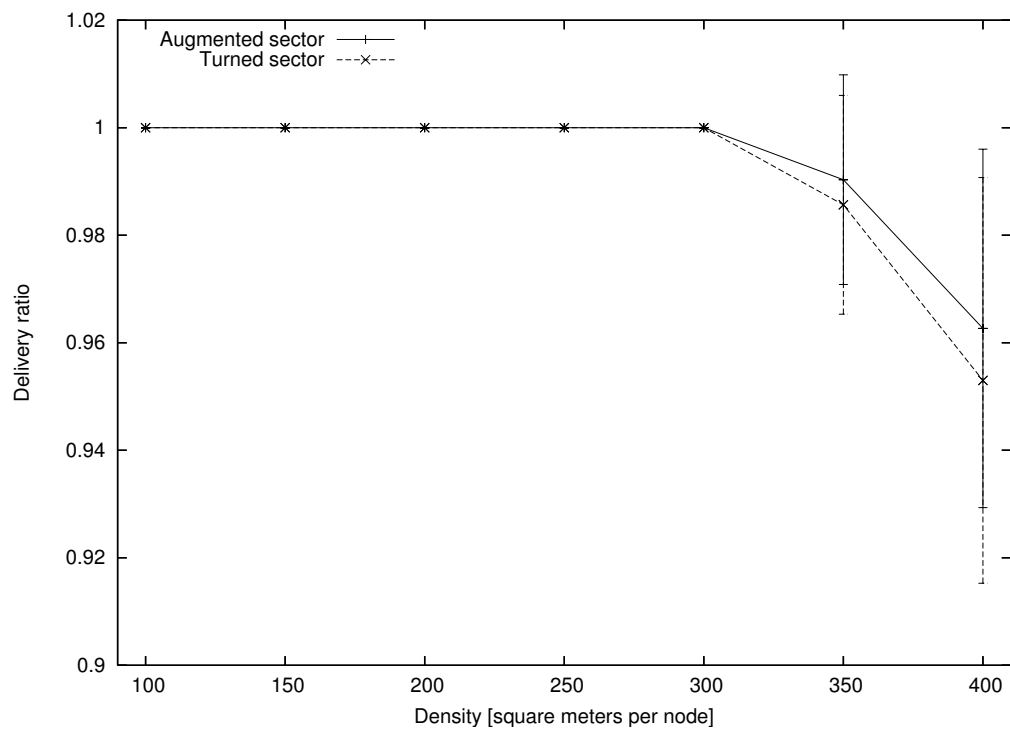


Figure 8.7: Delivery ratio using augmented and turned sector

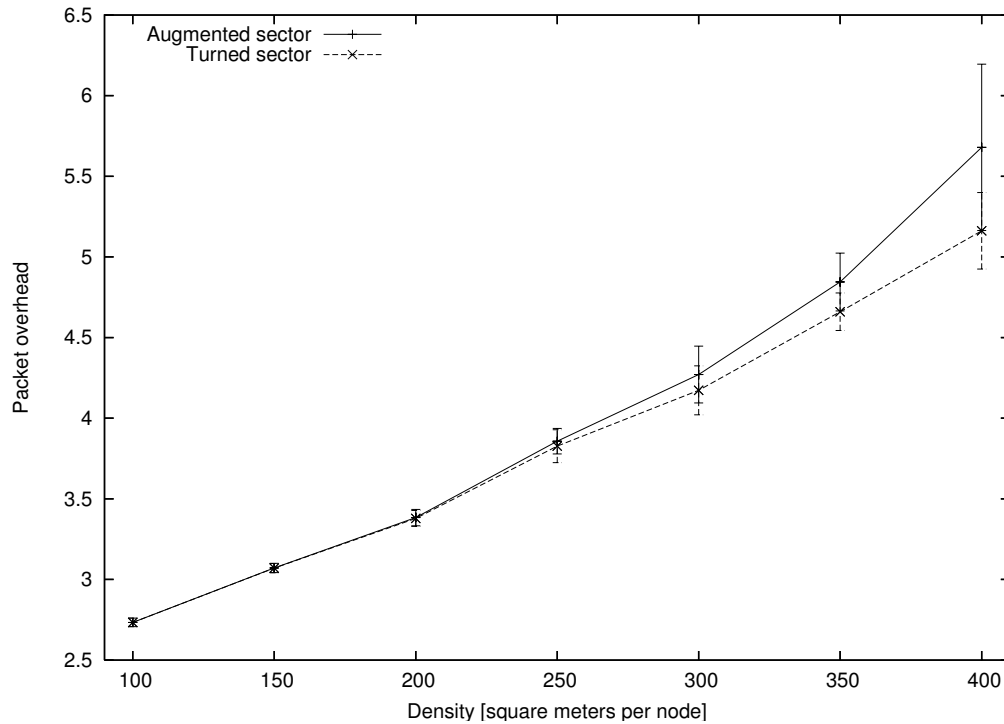


Figure 8.8: Packet overhead using augmented and turned sector

#### 8.1.4 ASF

Avoidance of Simultaneous Forwarding (ASF) is a novel strategy to prevent nodes from stopping forwarding the message in situations where the contention timers of two or more nodes expire almost simultaneously. Without ASF, the message would be dropped at all participating nodes, because each of them assumes that another node has forwarded the message. Figure 8.9 on the following page demonstrates that this strategy leads to a higher delivery ratio, because simultaneous forwarding situations do indeed occur sometimes. A delivery ratio of 100 % at medium node densities is only achievable with ASF turned on. ASF does not have noticeable influence on other metrics (packet overhead, stretch, and end-to-end delay), because it changes the behavior of BGR only in the rare cases of simultaneous forwardings.

#### 8.1.5 Geocasting

The geocasting scheme of BGR makes use of contention timers so that not all nodes within the geocast region broadcast the message. The alternative is flooding, where all nodes that receive the message broadcast it, if they have not broadcast it before. Neither flooding nor geocasting guarantees that all nodes within the geocast region receive the message, but flooding is likely to reach more nodes, because more nodes broadcast the message. This is confirmed by the simulation results. Figure 8.10 shows

## 8 Simulation Results

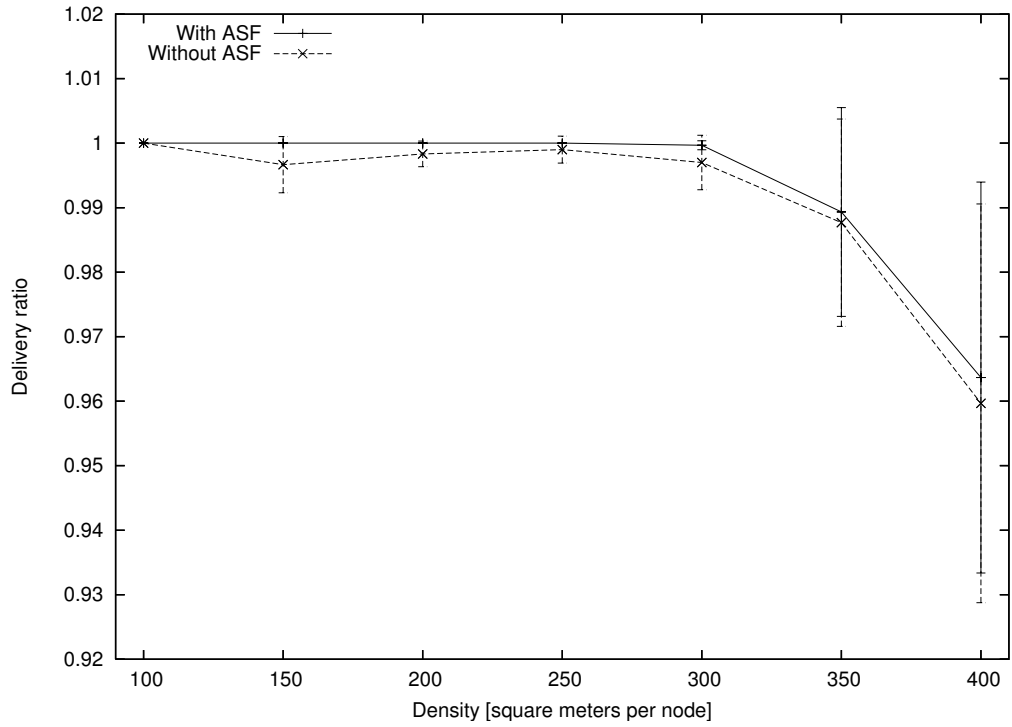


Figure 8.9: Delivery ratio with and without ASF

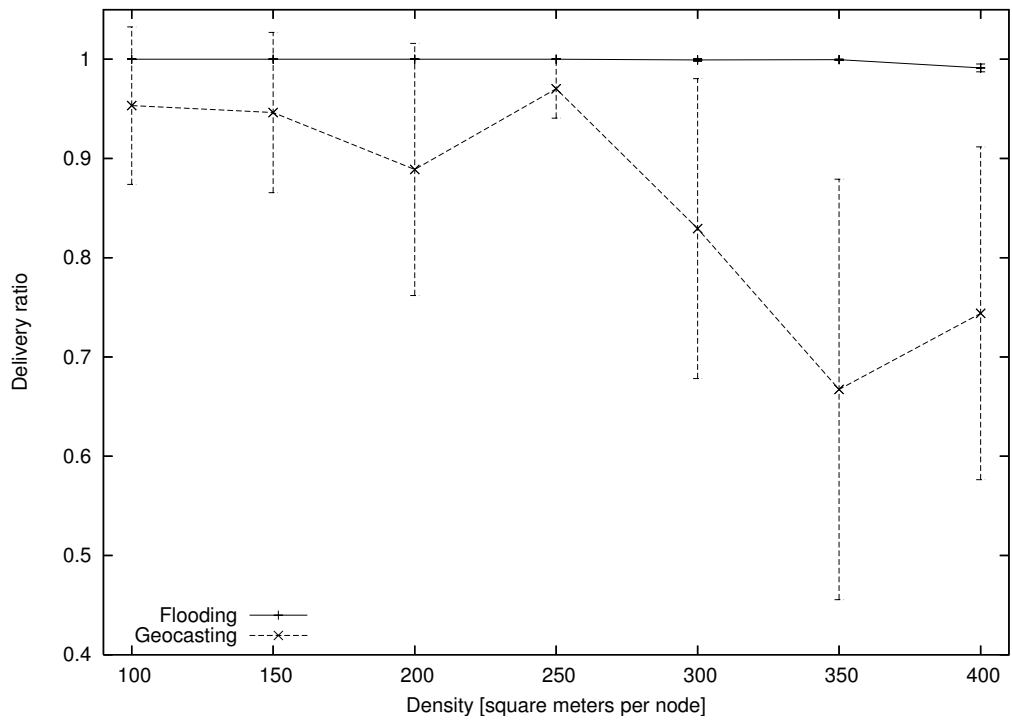


Figure 8.10: Delivery ratio (percentage of nodes within geocast region that received the message)

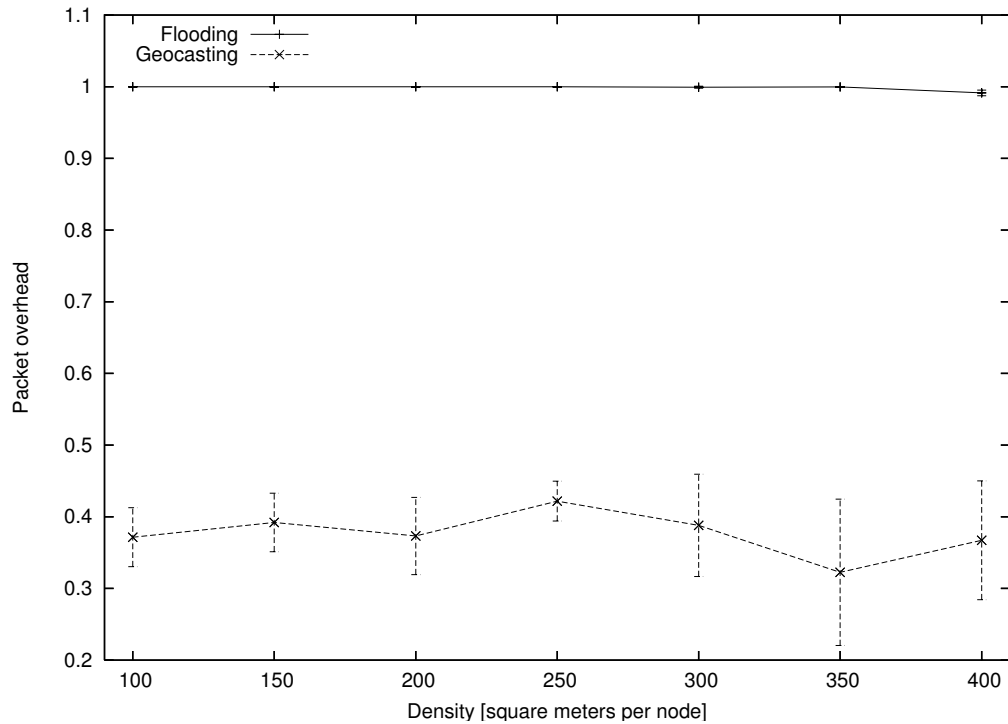


Figure 8.11: Packet overhead (number of packets divided by number of nodes)

the delivery ratio of flooding and geocasting. The entire network topology was configured as geocast region; the source node, which generated one message per simulation run, was placed in the corner. The results indicate that flooding leads to nearly 100 % delivery. The geocasting scheme yields significantly lower delivery ratios, even at high network densities. The large confidence intervals suggest that the performance is heavily dependent on the actual network topologies.

The packet overhead of flooding and the geocasting scheme is depicted in Figure 8.11. It is calculated as number of packets needed to spread the message in the geocast region divided by the number of nodes. As expected, flooding leads to a value of 1.0 (or little below in sparse networks, corresponding to the delivery ratio), because each node sends exactly one packet. Geocasting reduces the packet overhead drastically; it is between 0.3 and 0.4 at all network densities.

The conclusion of this evaluation is that the geocasting scheme of BGR is appropriate when delivery to all nodes within the geocast region is not essential, but communication overhead is crucial. Otherwise, ordinary flooding should be performed within the geocast region.

## 8 Simulation Results

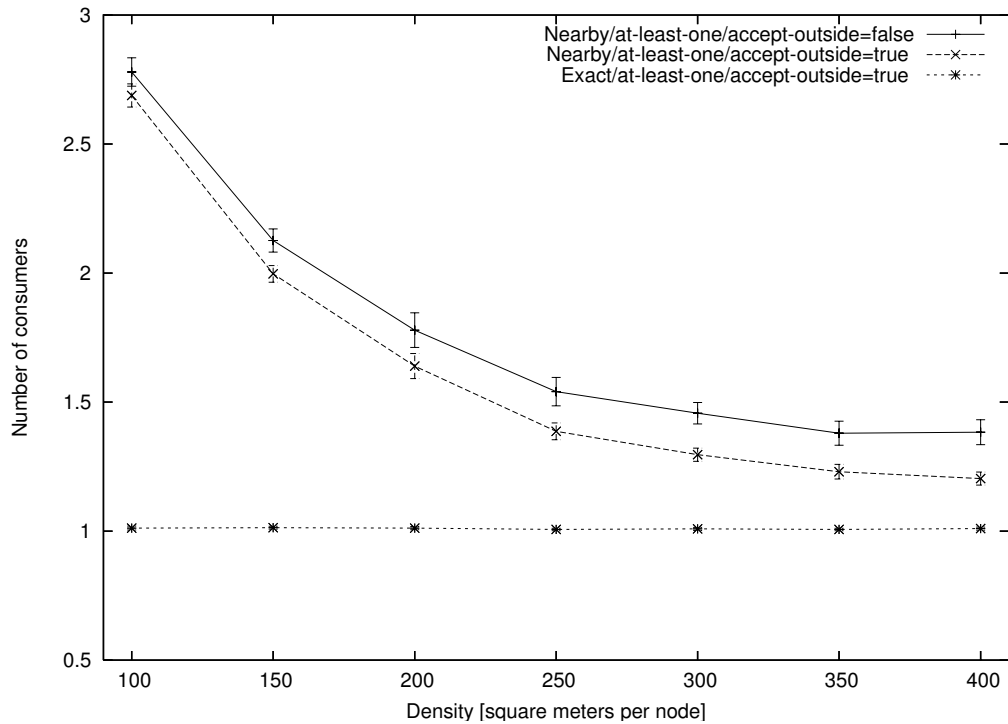


Figure 8.12: Average number of consumers for *at-least-one* semantics

### 8.1.6 Delivery Semantics

The delivery semantics introduced in this thesis are novel, so the evaluation exhibited here represents the first results in the literature within this field. For the following experiments, the topologies were divided into 64 squares of equal size. The source was placed in the center of the topology and generated 64 messages; the destination locations were the centers of the squares.

When the *multiplicity* parameter is *at-least-one*, there can be more than one node that consume a message. The average number of consuming nodes at different network densities is shown in Figure 8.12. Using *exact* semantics, the number of consuming nodes is close to one in all experiments. (*Exact* semantics is only reasonable in combination with *accept-outside=true* in this simulation setup, because there are no nodes having exact destination locations.) Using *nearby* semantics (here with a limit of tolerance of 10 m), the value is about 2.75 for dense networks and 1.25 for sparse networks. The value is slightly lower when *accept-outside* is *true*, because more messages were delivered in this case (when no nodes with fitting limit of tolerance were found), but only one node consumed the message, namely the node where the message got stuck.

The influence of *accept-outside* on the delivery ratio is depicted in Figure 8.13 on the next page. When it is *true*, the delivery ratio is very close to one, except for low

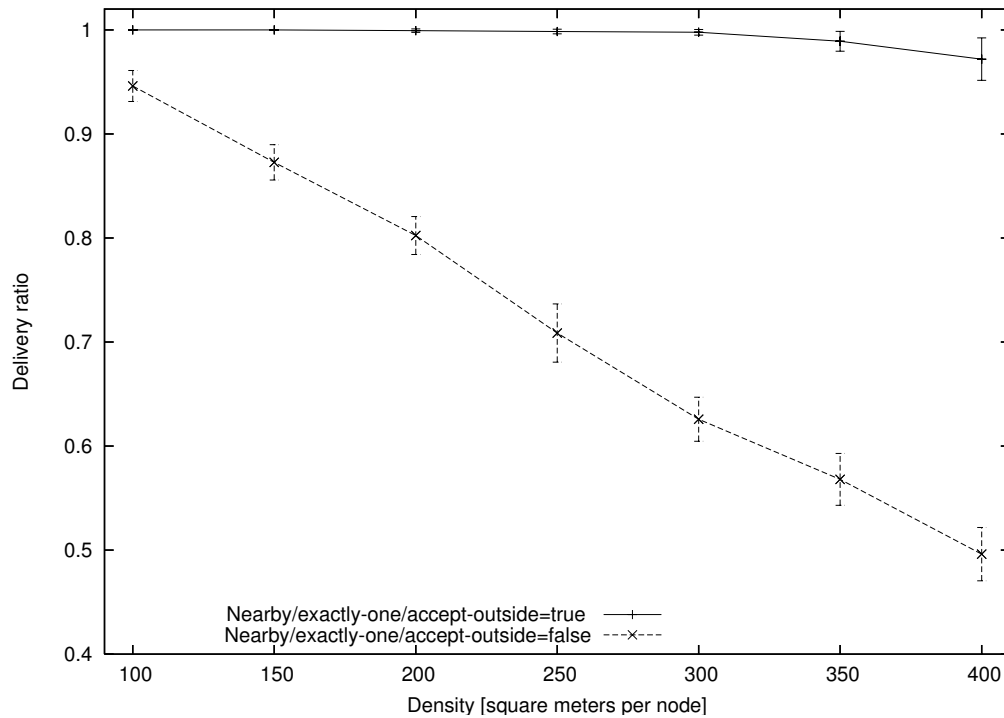


Figure 8.13: Influence of *accept-outside* on delivery ratio

network densities, where messages sometimes get stuck far away from the destination location (note that a message is only consumed by a node that is within the transmission range from the destination location). When *accept-outside* is set to *false*, the delivery ratio decreases at lower densities, because the probability that no nodes with matching limit of tolerance can be found decreases.

Figure 8.14 on the following page shows the results of an investigation of the *nearby* semantics. The average distance between destination location and location of consumer is depicted. Not surprisingly, the value grows at lower network densities. When *accept-outside* is *true*, the value is significantly higher. At very low densities, the value is even larger than the limit of tolerance (10 m).

The results prove that the implementation of the delivery semantics in BGR works as expected and produces reasonable results.

### 8.1.7 Multi-flow Traffic

It is important that a routing algorithm shows good performance under a heavy load. The evaluation of multi-flow scenarios provides evidence of the suitability of BGR in the presence of high-volume traffic on many different paths. In the simulations, the network has been divided equally, but randomly, into sending and receiving nodes. Each sending node has a corresponding receiving node. At a constant rate, each send-



## 8 Simulation Results

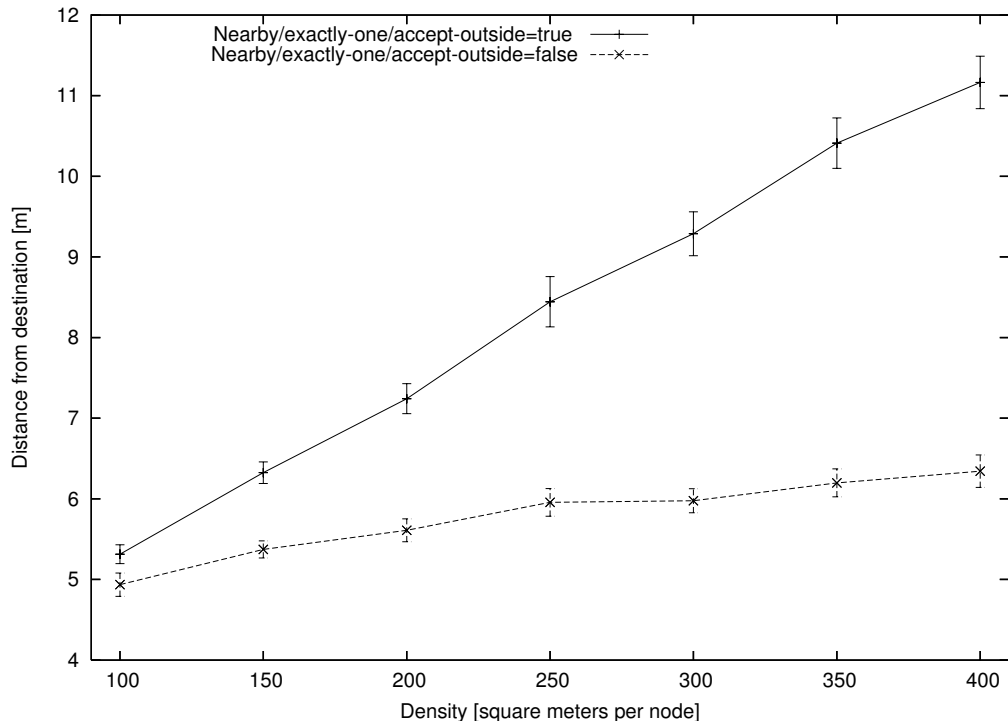


Figure 8.14: Average distance between destination location and location of consumer

ing node sends a new message to its receiving node simultaneously. This is repeated ten times. To incorporate recovery mode in the evaluation, low-density networks of  $400\text{ m}^2$  per node were simulated. The experiments were run for BGR and GPSR to compare these algorithms. Figure 8.15 on the next page shows that BGR performs much better than GPSR at high message rates. The unicast scheme of GPSR leads to many collisions. The advantage of BGR is that no node is selected a priori, so when the optimal node does not receive the packet due to a collision, there is a chance that another node receives the packet and forwards it. At high message rates, the stretch is larger, but Figure 8.16 on the facing page illustrates that in BGR, the stretch does not increase significantly. The high increase of the stretch in GPSR comes from the low delivery ratio, which is comprised in the stretch value.

### 8.1.8 Mobile Nodes

The advantage of not storing topology information in BGR arises perspicuously when the nodes are mobile. In beacon-based algorithms, node movements result in outdated neighborhood tables, thus the beacon interval has to be adjusted according to the node speed. Shorter beacon intervals, however, result in significantly more communication overhead.

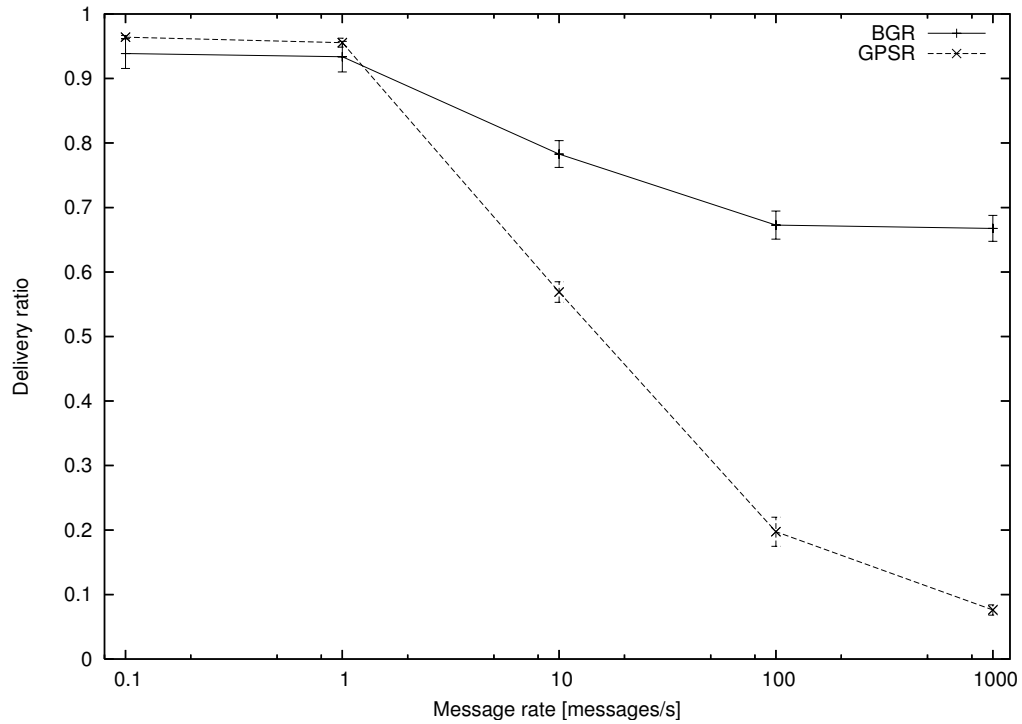


Figure 8.15: Delivery ratio at multi-flow traffic (logarithmically scaled x-axis)

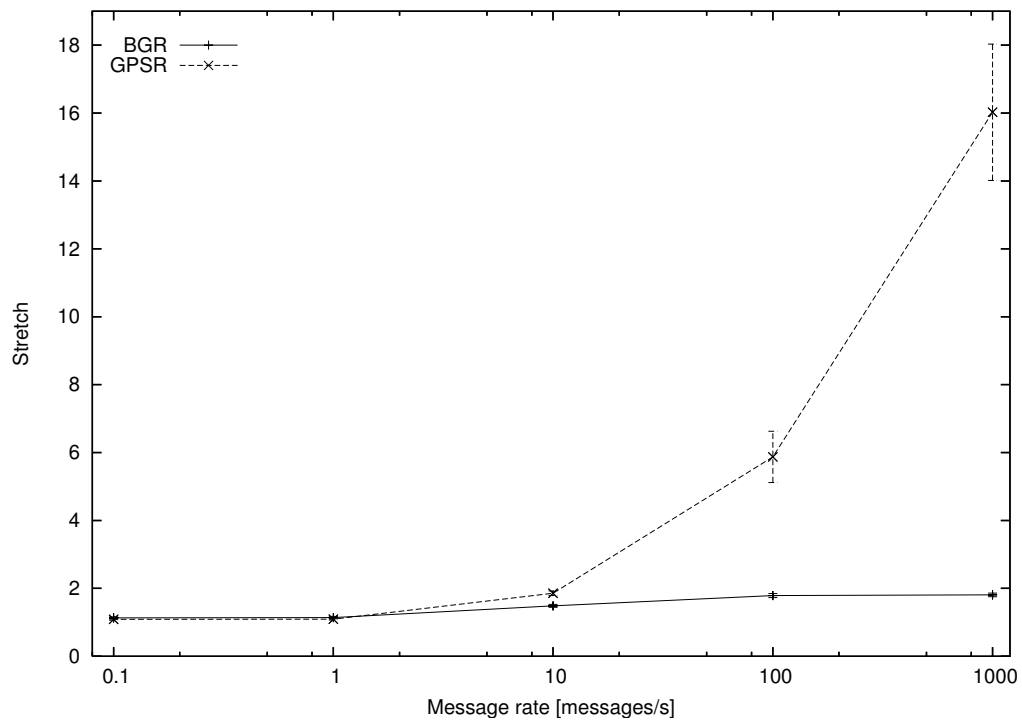


Figure 8.16: Stretch at multi-flow traffic (logarithmically scaled x-axis)

## 8 Simulation Results

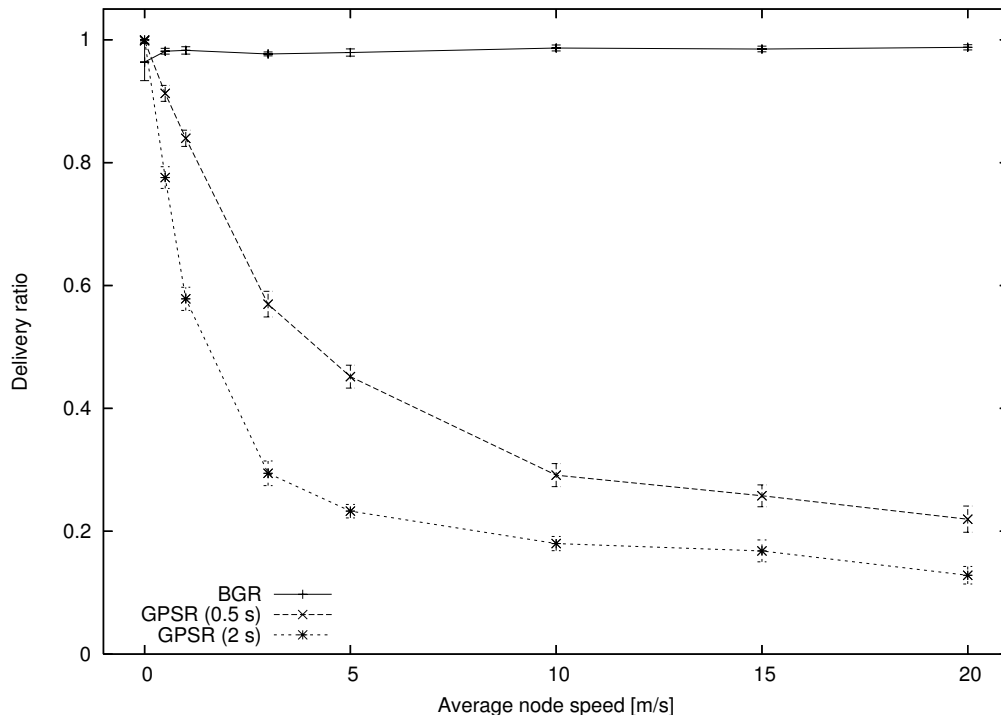


Figure 8.17: Performance of BGR and GPSR (with different beacon intervals) in case of mobile nodes

To investigate the influence of mobility on the performance of BGR and GPSR, simulations were conducted using the *random walk with reflection* model presented in [LBV05]. The random waypoint model frequently used in simulation studies for ad hoc networks has not been used, because it produces unreliable results [YLN03]. The travel time has been set to  $20\text{ s} \pm 10\text{ s}$ , pause time is zero. Simulation runs for GPSR were performed using beacon intervals 0.5 s and 2 s. Low-density networks ( $400\text{ m}^2$  per node) were simulated. The average node speed has been varied between 0 and 20 m/s, speed delta (maximum deviation from average speed) is 50 %. The sink is stationary at the center of the topology, because GPSR does not support mobile destinations, and in BGR the results depend on the delivery semantics.

Figure 8.17 shows that node movements do not have negative impact on the delivery ratio of BGR. Quite the contrary, the delivery ratio even increases at higher node speeds. GPSR, on the other hand, has massive problems even at low node speeds. This is a clear advantage of the broadcast forwarding of BGR. The unicast scheme of GPSR is subject to fail in case of mobility. Especially the usage of greedy mode is harmful in conjunction with movements, since the node with largest packet progress toward the destination is likely to move out of transmission range soon.

In the original implementation of GPSR, a node is removed from the neighborhood table when three successive beacons from this node have not been received. In case of

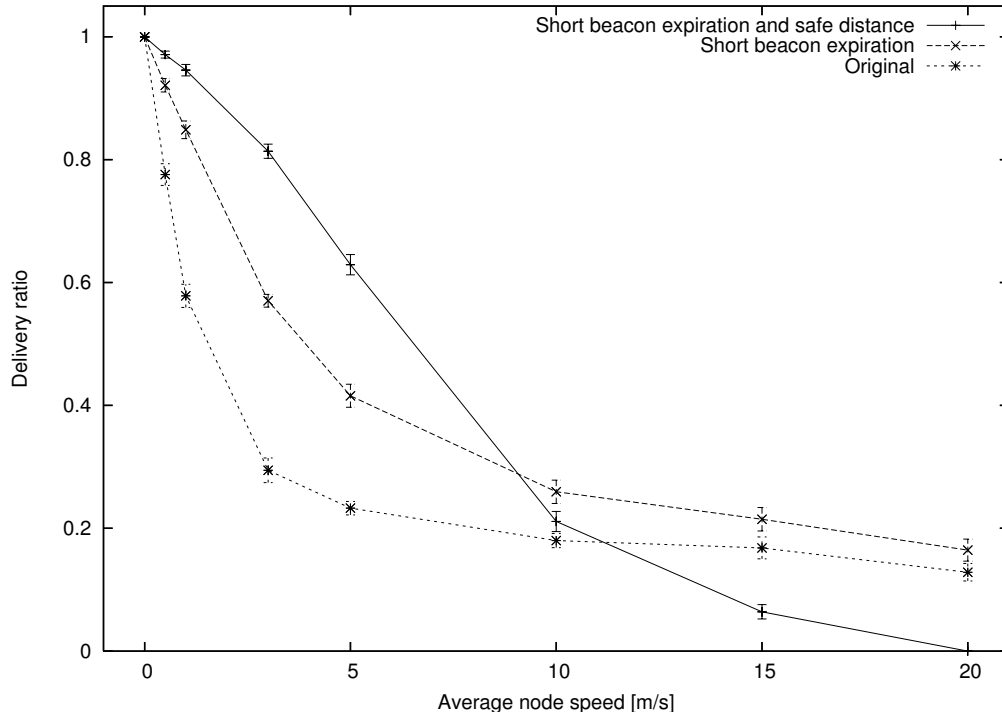


Figure 8.18: Modifications of GPSR for mobility (beacon interval 2 s)

mobility, however, this limit is too high, because unicasts are frequently sent to nodes which have already moved out of transmission range. The delivery rate of GPSR can be improved for mobile networks by setting this limit to only one beacon, so that a node is removed from the neighborhood table immediately, when a beacon from this node has not been received. The key issue is that there are more beacons which are missing due to mobility than due to transmission failures or collisions. Figure 8.18 confirms that this modification (short beacon expiration) leads indeed to a higher delivery ratio.

Another issue is that sending to a node which has been close to the boundary of the transmission area when broadcasting its beacon is subject to fail, because those nodes are likely to have moved out of range since then. Sending to these nodes can be prevented by including only those nodes into the neighbor list whose distance to the transmission area boundary is at least  $b \cdot v$ , where  $b$  is the beacon interval and  $v$  is the average node speed. Figure 8.18 illustrates that this improves the delivery ratio again, but only under moderate node speed. At high speeds, the safe distance (maximum distance between forwarder and candidate) is too short, which results in too few forwarding candidates. At a speed of 20 m/s, the safe distance is zero under the simulation parameters in use. BGR, however, still performs much better even at lower node speeds. Other values for the safe distance have also been investigated, but appeared to perform worse than the proposed threshold.

## 8 Simulation Results

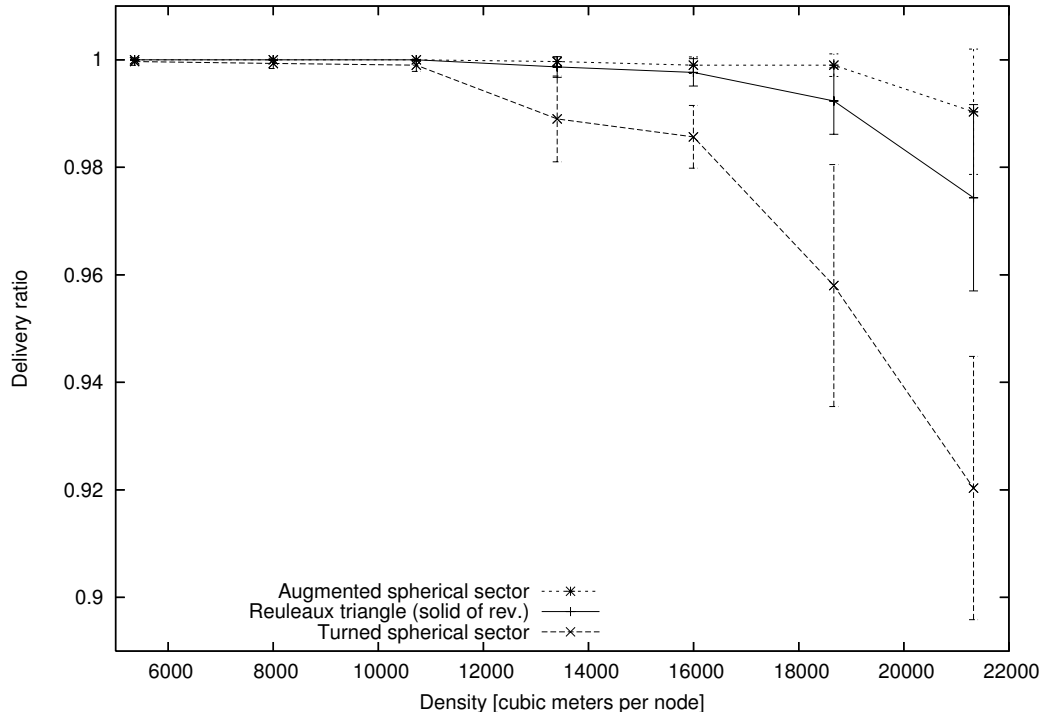


Figure 8.19: Delivery ratio of BGR in 3D topologies

In BGR, the high delivery ratio comes at the cost of a small (approximately linear) increase of communication due to duplications when candidate nodes move away from each other and both forward the message. This is tolerable, since communication overhead is still very low.

### 8.1.9 Three-dimensional Topologies

For the evaluation of BGR in three-dimensional networks, topologies with the same average number of neighbors as in the two-dimensional topologies were created. Thus, the node density varies now in the range between 5362 m<sup>3</sup> per node and 21327 m<sup>3</sup> per node.

Figure 8.19 shows the delivery ratio of BGR with different forwarding volumes. The curve for the sphere has been omitted; it runs between the curves of the Reuleaux triangle and the turned spherical sector. The results indicate that the coverage of the forwarding volumes (including recovery mode) has a strong influence on the delivery ratio. The augmented spherical sector, which covers the entire half-sphere, has the highest delivery ratio; it is close to one, except for very sparse networks. The turned spherical sector, on the other hand, covers the smallest volume and therefore has the smallest delivery ratio.

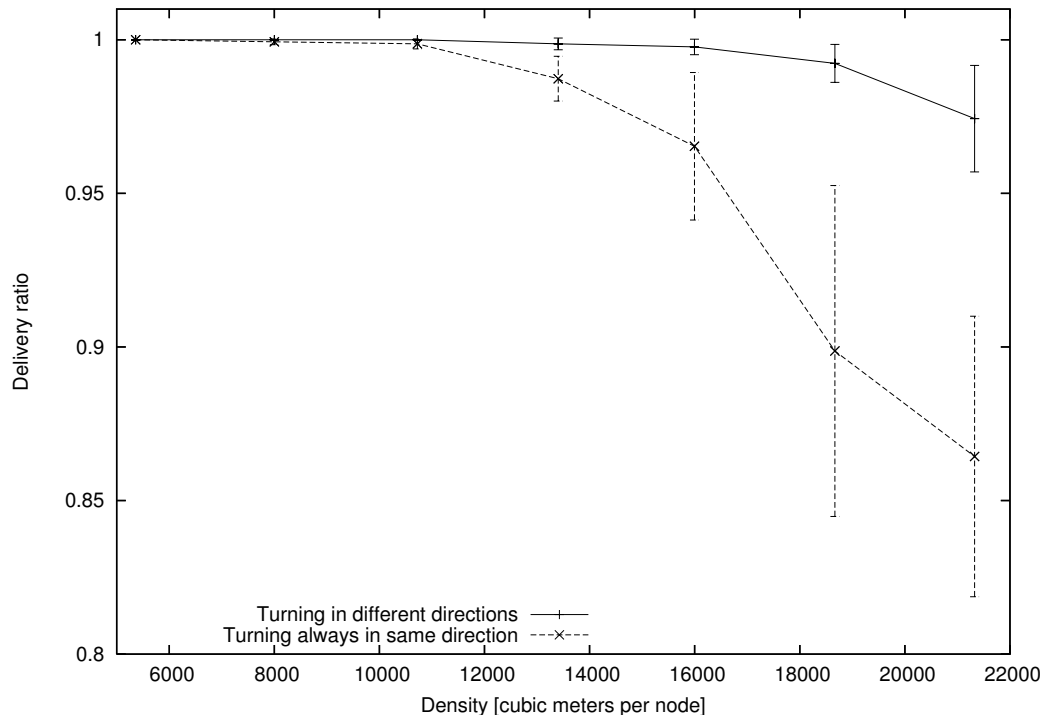


Figure 8.20: Delivery ratio in 3D using different turning strategies

Another observation is that for low-density networks, the delivery ratio is higher than for 2D networks with equal average number of neighbors (cf. Figure 8.1 on page 73). This is in conflict with the results discussed in Section 5.4, which states that the delivery ratio is lower in 3D topologies. The solution for this seeming contradiction is that the turning strategy of BGR in recovery mode (turn the forwarding volume upward first when the node is located close to the ground) prevents many routing failures. Figure 8.20 supports this explanation: It shows the delivery ratio of BGR with the solid of revolution of the Reuleaux triangle as forwarding volume with and without the strategy to incorporate the z coordinate in the decision for the turning direction. It is obvious that the strategy of turning the forwarding volume in different directions performs much better. This result suggests that applying the same strategy at all network boundaries (in 2D and 3D) would lead to even higher delivery rates.

## 8.2 Location Errors

BGR has been designed to tolerate location errors, which has been discussed in Chapter 6. The width of the forwarding area/volume is adjusted dependent on the estimated standard deviation of the location error. Some fixes for GPSR in presence of location errors have also been proposed. In short, these fixes are:

## 8 Simulation Results

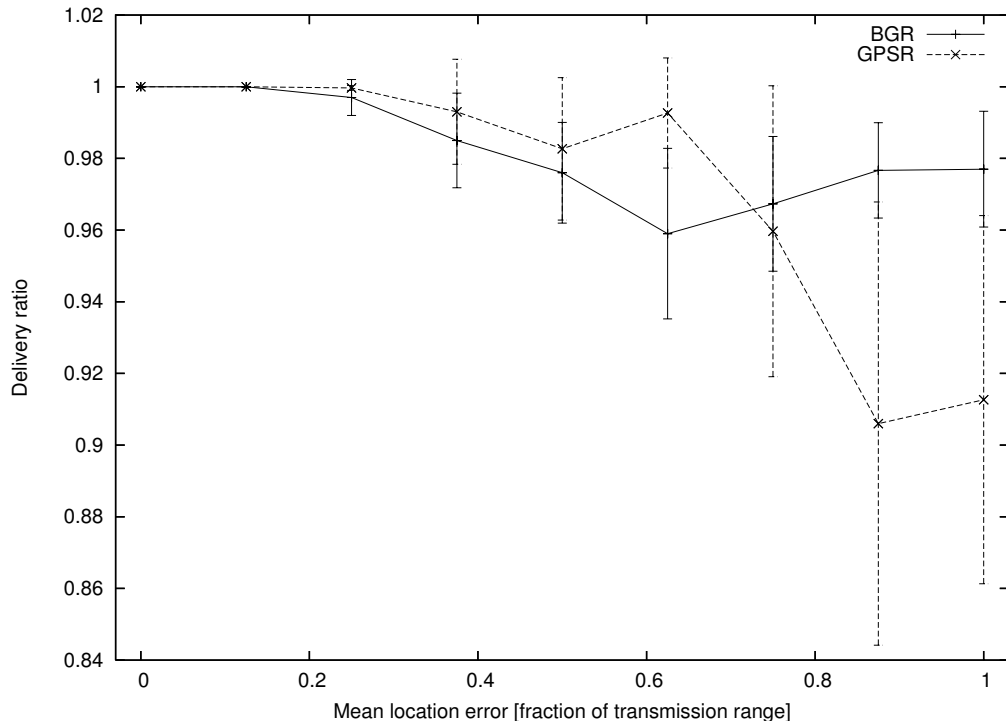


Figure 8.21: Delivery ratio at location errors

1. incorporation of two-hop neighborhood information,
2. link removal only if node distance is below threshold,
3. enlargement of Gabriel circle, and
4. combination of the two previous fixes.

Figure 8.21 shows the performance of BGR and original GPSR in high-density networks ( $100 \text{ m}^2$  per node). The mean location error (parameter  $\sigma$  of the Gaussian distribution) is varied between zero and the transmission range. Errors up to the order of magnitude of the transmission range can occur in localization schemes [LP04]. At low location errors, GPSR performs slightly better than BGR; obviously, the recovery strategy of BGR routes messages in the wrong direction from time to time under location errors. At high location errors, however, BGR performs significantly better than GPSR. At location errors of  $0.75 r$  and higher, GPSR frequently switches to face routing, which fails due to incorrect planarization. Almost all messages that GPSR failed to deliver were dropped after having looped around the perimeter.

The proposed fixes for GPSR were intended to solve these problems. The performance of the fixes one and four is depicted in Figure 8.22 on the facing page. The curves for the other fixes have been omitted for readability reasons; they are similar to the curve for original GPSR. Fix 1 makes the performance even worse, while

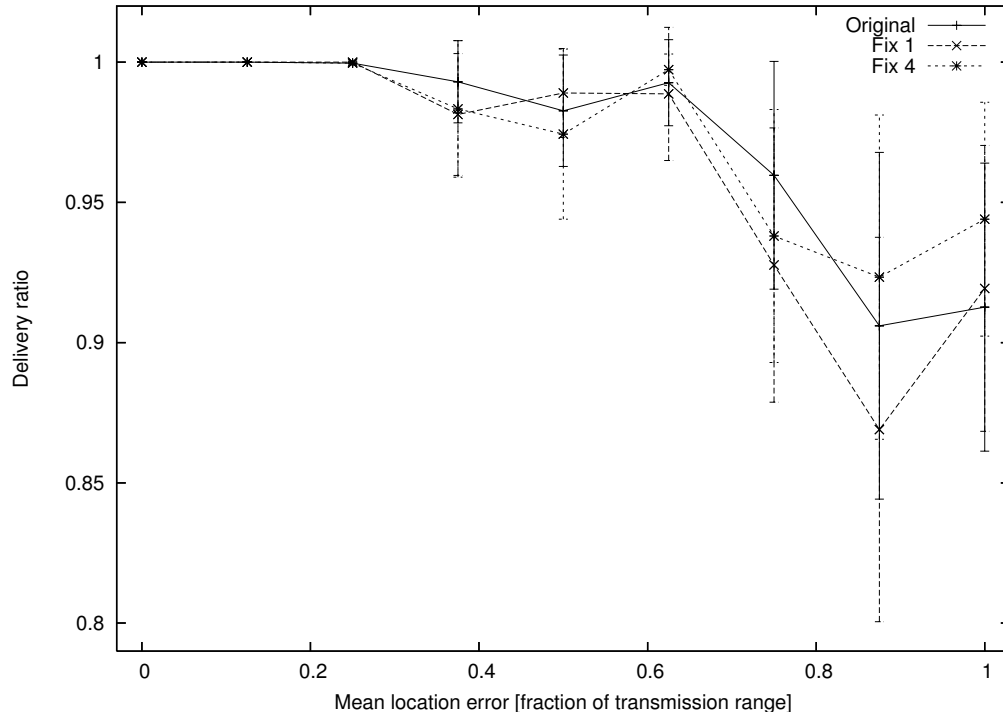


Figure 8.22: Delivery ratio at location errors for GPSR and fixed variants

fix 4 improves the delivery ratio only at very high location errors. However, BGR still performs better under high location errors.

The poor performance of the first fix is another indicator that the main problem of location errors in face routing is not disconnection due to incorrect edge removal (as claimed in [SHG04]), but intersection of links, as discussed in Section 6.4.

For BGR, the conclusion of these simulation studies is that the performance is acceptable even at high location errors.

### 8.3 Irregular Radio Model

Since the unit disk graph model is not appropriate for evaluation of realistic ad hoc network scenarios, simulation studies using an irregular radio model are indispensable. In particular, the following phenomena are missing in the unit disk graph model, but do arise in real deployments (see Sections 2.2 and 3.1.4):

- varying transmission ranges,
- anisotropic signal propagation,
- transient communication failures,



## 8 Simulation Results

- unidirectional links.

A realistic radio model that contains all of these issues has been developed in [ZK04]. The model was constructed using techniques from communication theory; the suggested values result from measurements on MICA2 motes. The packet reception rate at a given node distance  $d$  is

$$PRR(d) = \left( 1 - \frac{1}{2} \exp\left(-\frac{\gamma(d)}{2} \frac{1}{0.64}\right) \right)^{8f}.$$

In this function,  $f$  is the frame length in bytes, and  $\gamma(d)$  is the signal-to-noise ratio. It is defined as

$$\gamma(d) = P_t - PL(d) - P_n \quad (\text{in dB}),$$

where  $P_t$  is the transmission power,  $P_n$  is the noise floor, and  $PL(d)$  is the path loss, for which the log-normal shadowing model is used:

$$PL(d) = PL(d_0) + 10n \log_{10} \left( \frac{d}{d_0} \right) + X_\sigma.$$

Here,  $d_0$  is a reference distance,  $n$  the path loss exponent, and  $X_\sigma$  a Gaussian-distributed random variable with mean zero and standard deviation  $\sigma$ , which models shadowing effects.

For evaluation of BGR and GPSR, this radio model has been implemented in ns-2. The parameters were adopted from [ZK04]; the power settings were adjusted such that the average transmission range is about 40 m in order to compare the results with the unit disk graph model. Table 8.2 lists the chosen parameters. Figure 8.23 on the facing page shows the resulting packet reception rate as a function of the node distance.

Table 8.2: Radio parameters

Parameter	Value
Frame length $f$	50 bytes
Transmission power $P_t$	0 dBm
Noise floor $P_n$	-105 dBm
Reference distance $d_0$	1 m
Path loss $PL(d_0)$	48 dB
Path loss exponent $n$	3
Shadowing deviation $\sigma$	3 dB

### 8.3 Irregular Radio Model

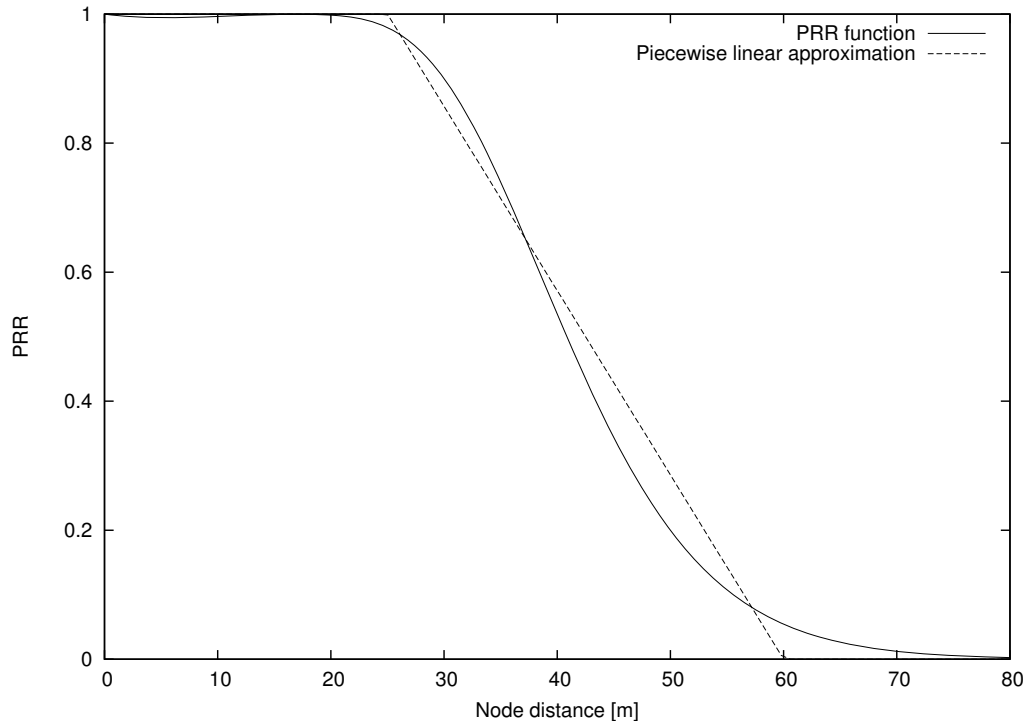


Figure 8.23: Packet reception rate in irregular radio model

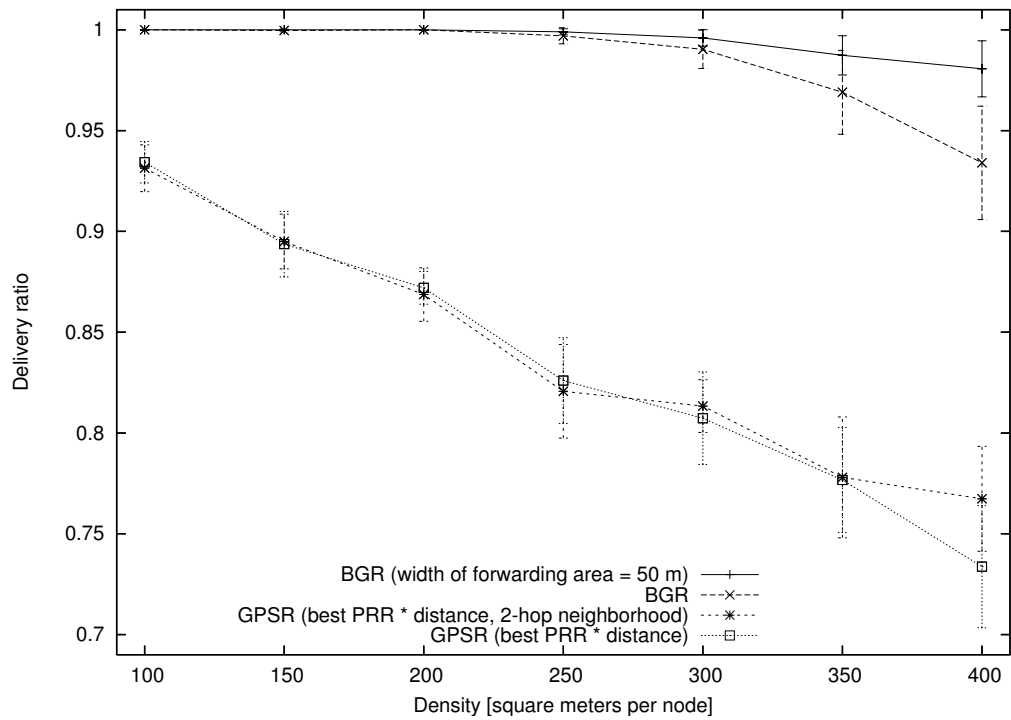


Figure 8.24: Delivery ratio using irregular radio model (BGR with Reuleaux triangle of different width)

## 8 Simulation Results

Because the original GPSR algorithm was designed for unit disk graphs, an improvement of GPSR for irregular radio models has been simulated. Several improvements have been suggested recently. CLDP [KGKS05a] and LCR [KGKS06] improve face routing by removing crossing links. Our experiments, however, revealed that face routing is performed very rarely in the applied radio model; in most cases, unicast sending in greedy mode resulted in delivery failures. Thus, improvements of face routing would only have a marginal effect on the total delivery rate. Instead, an improvement of greedy forwarding has been implemented. It has been proposed by Seada et al. [SZHK04]; instead of choosing the neighbor closest to the destination, the neighbor with highest product of PRR and distance progress toward destination is selected. PRR is an estimation of the packet reception rate of the neighbor in question. In our implementation, it is a piecewise linear approximation of the PRR function, also depicted in Figure 8.23 on the previous page. The function is

$$PRR_{\text{approx}}(d) = \begin{cases} 1 & (d < 25) \\ 0 & (d > 60) \\ \frac{60-d}{35} & (\text{otherwise}) \end{cases}$$

The delivery ratio of BGR and GPSR is depicted in Figure 8.24 on the preceding page. At high node densities, the delivery ratio is close to 1.0 for BGR, which demonstrates the robustness of BGR against radio irregularity. Even at low densities, the delivery ratio is not much behind the results from the unit disk graph model. It can also be seen that an enlargement of the forwarding area improves the delivery ratio even more, with 98 % delivery when using a Reuleaux triangle of width 50 m even in very sparse networks.

In contrast to that, GPSR performs significantly worse despite of the use of the new metric. This is mainly due to the unicast transmission scheme, which leads to failures in case of unidirectional links and variances in transmission range. Recall that the MAC layer already implements up to three retransmissions for unicast packets. To improve the performance of GPSR, the variant with two-hop neighborhood information, which has already been used in the simulations of location errors, has also been simulated. A forwarding node sends the message only to a neighbor that is known to have received beacons from the forwarder recently. This strategy raises the probability that the transmission succeeds. Figure 8.24 on the previous page, however, reveals that there is only little improvement on the delivery ratio in low-density networks and almost no improvement in dense networks. The results indicate that GPSR is not suitable as a routing protocol for networks with radio irregularity, while BGR performs very well, especially when enlarging the forwarding area.

### 8.4 Combining Location Errors and Irregular Radio Model

The most realistic results are achieved when simulations include both location errors and an irregular radio model. Such simulation studies are still rare nowadays; simu-

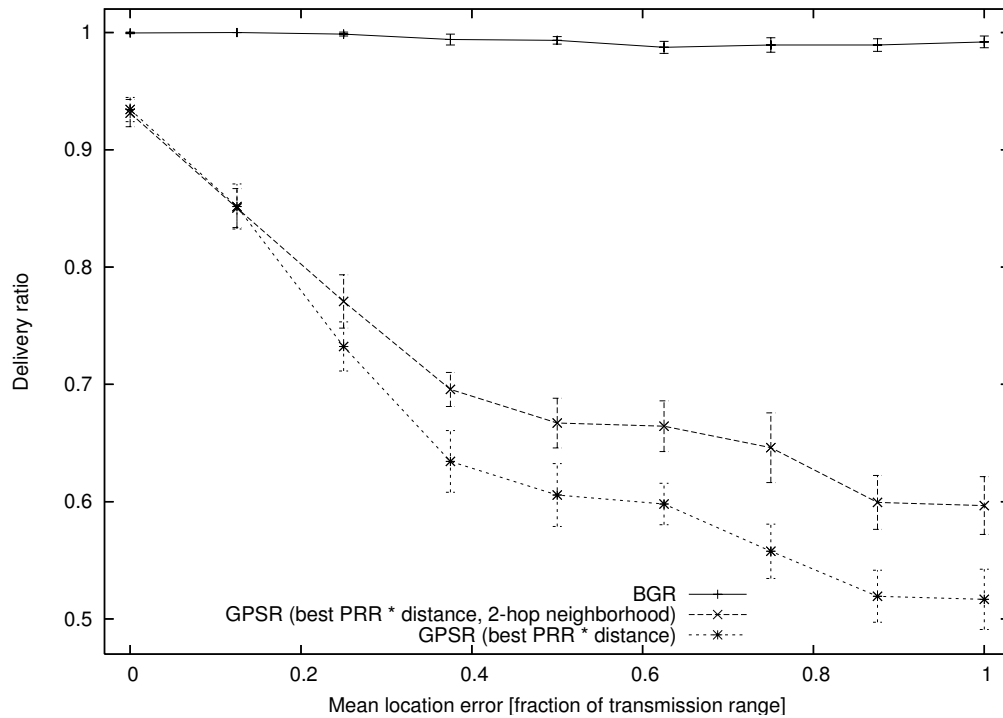


Figure 8.25: Delivery ratio with location errors and irregular radio model

lation experiments that claim to be realistic mostly include only one of them, while the majority of simulations are still run under the unit disk graph model. Studies with the aim of being realistic are usually done on real hardware; these studies, however, suffer from unpredictable influences, are difficult to repeat and hard to generalize. Conducting simulation experiments with realistic settings produces much more systematic results. For BGR and GPSR, this has been done by combining location errors and irregular radio model from the previous sections.

The results are depicted in Figure 8.25. Like in Section 8.2, high-density networks of  $100 \text{ m}^2$  per node have been simulated. The superiority of BGR is clearly evident. The delivery ratio of BGR is very close to 1.0 even for high location errors, despite of using normal-sized forwarding areas (width is equal to estimation of transmission range). Enlarging the forwarding areas to 50 m leads to even higher delivery rates.

The performance of GPSR, on the other hand, is poor, despite of applying the improvements presented in Section 8.3. These results demonstrate that GPSR as a representative of beacon-based routing is not an acceptable routing algorithm for realistic networks, while BGR succeeds even at high location errors and radio irregularity.

### 8.5 Discussion

In the following, the results of the simulation studies are summarized and discussed.

To ensure a delivery ratio close to one, BGR requires a sufficiently high network density of about 16 neighbors per node on average. At a transmission range of 40 m, this is equivalent to a maximum of about  $300 \text{ m}^2$  per node in 2D, or  $16000 \text{ m}^3$  per node in 3D.

The Reuleaux triangle appeared to be the best choice for the forwarding area in 2D, because it offers a high delivery ratio, a low end-to-end delay, and a small path stretch. The augmented sector results in similar delivery rates and even lower stretch, but higher end-to-end delay, because recovery is triggered more often. Additionally, the packet overhead is slightly higher when using the augmented sector, which is due to packet duplication. In three-dimensional topologies, the augmented spherical sector should be used, because it is the only forwarding volume that leads to a delivery ratio of almost one. This is because the other forwarding volumes leave large gaps in recovery mode.

The geocasting scheme of BGR has a very low packet overhead (below 40%), but the delivery ratio is considerably below one. If this is an issue, ordinary flooding should be performed within the geocast region.

Multi-flow traffic does not lead to a significant performance decrease of BGR, neither in terms of delivery ratio nor in terms of stretch. In contrast to GPSR, which fails almost completely at high message rates, the performance of BGR is still acceptable.

In mobile networks, the performance of BGR is excellent. The delivery ratio remains close to one even at very high node speeds. The delivery ratio of GPSR decreases even at low node speeds, although modifications were implemented in GPSR to select only neighbors that are close to the forwarder.

Under realistic settings (location errors and radio irregularity), BGR proved to have a high delivery ratio even at location errors equal to the transmission range. Enlarging the forwarding areas yet leads to a further improvement. GPSR, on the other hand, shows a poor performance, although applying several modifications to cope with radio irregularity. Especially the  $\text{PRR} \times \text{distance}$  metric frequently cited in the literature did not help to come close to the performance of BGR.

## 9 Conclusion

### 9.1 Summary

The main goal of this thesis was the design of a robust, beacon-less geographic routing algorithm for ad hoc networks that operates with as little communication overhead as possible and performs well in realistic networks. With Blind Geographic Routing (BGR), this goal has been achieved. BGR is also the first beacon-less routing algorithm that has been designed to operate in three-dimensional networks.

BGR is a beacon-less algorithm, i. e., messages are broadcast without explicit knowledge about the neighborhood. This is beneficial in different ways:

- No communication is needed for maintaining neighborhood information,
- communication failures are reduced, because no unicast communication is performed,
- node mobility is supported better, because no neighborhood tables are used, which are subject to become outdated.

The main disadvantage of BGR is that a high node density is needed to prevent routing failures. In high-density networks, however, a delivery ratio of 100 % is achieved. Another issue is that a small amount of memory is needed to store the IDs of the last  $v$  forwarded messages to prevent routing loops. However, this does not cause a serious memory overhead, if  $v$  is chosen small, which is legitimate unless network traffic is very high. Since packets have to be stored while a timer is running, this causes additional memory overhead that depends on traffic volume and number of traffic flows. The delay induced by the timers makes BGR less suitable for highly delay-sensitive applications. However, a speedup at the cost of higher collision probability is possible by reducing the maximum delay  $m$  or using the exponential or quadratic timer function.

BGR makes use of forwarding areas (volumes in 3D) to perform a timer-based contention for the next hop. The width of the forwarding areas/volumes guarantees that all nodes located within it can mutually communicate with each other (in the unit disk/ball graph model). This prevents multi-path forwarding, which would come at additional communication costs. Theoretical consideration and simulation studies revealed that not only the size of the forwarding areas/volumes is crucial, but also the number of contained nodes close to the forwarder and close to the destination. The Reuleaux triangle appeared to be the best choice.

## 9 Conclusion

The recovery strategies of BGR are simple and robust. Turning the forwarding area/volume and re-broadcasting the message can be done without large communication overhead for detecting neighbors or strategies that fail in realistic networks, such as face routing. Another strategy that improves the delivery rate at the cost of additional communication overhead is augmentation of the sector. Because the width of the augmented sector is larger than the transmission range, multi-path flows can occur. The higher delivery rate compared to the turning strategy results from the lower probability to route messages in the wrong direction. In the 3D version of BGR, an additional advantage of the augmented sector is that it is the only forwarding volume that covers the entire half-sphere after all recovery steps.

ASF (Avoidance of Simultaneous Forwarding) is another novel strategy introduced in BGR. It solves problems that arise when two or more nodes forward the message almost simultaneously. Without ASF, all nodes that receive the first and the second message would cancel their timers upon receiving the second message, because they assume that this was the forwarding of the next hop. This would lead to a routing failure. With ASF, the hop count stored in the message is evaluated so that the timers are not canceled in this situation.

The concept of delivery semantics for geographic routing has been introduced in this thesis. The parameters *closeness*, *multiplicity*, and *accept-outside* have been proposed. The *closeness* defines how close a node must be located to the destination position in order to declare itself as destination and consume the message. The *multiplicity* specifies how many nodes may serve as destination nodes. In case of *exactly-one*, additional overhead is necessary in order to determine the consumer. Finally, the parameter *accept-outside* defines whether a node may consume the message when no node can be found that meets the *closeness* semantics, but the destination is located within transmission range of the current node. BGR is the first routing algorithm that supports delivery semantics.

A low-communication variant of geocasting has also been implemented in BGR. It is based on intelligent flooding using the same timer function as for forwarding. Nodes that notice that another node floods the message do not broadcast it. This scheme operates with very little communication, but delivery to all nodes within the geocast region is rarely achieved. If this is crucial, ordinary flooding should be performed.

BGR has been designed to tolerate location errors. The width of the forwarding areas/volumes is adjusted according to the estimation of the standard deviation of the location error. The width is set to the expected value for the measured distance between two nodes whose real distance is equal to the transmission range. This value can be calculated numerically and stored in the nodes as a constant. Additionally, several novel improvements for GPSR in case of location errors have been proposed and evaluated. Although at small location errors, GPSR performs slightly better than BGR, the latter is clearly superior even to the improved GPSR variants at high location errors.

An analytical model for calculating the delivery probability of BGR has been developed in the thesis. An approximation of the delivery probability under the unit disk graph model (unit ball graph model in 3D) has been calculated dependent on the source-destination distance, the transmission range, and the network density. The model supports the (spherical) sector as forwarding area/volume with turning or augmentation as recovery strategy.

Extensive simulation studies with ns-2 approve that BGR is indeed robust against communication failures, location errors, radio irregularity, and node mobility. Although delivery is not guaranteed (which is impossible in 3D for local algorithms anyway), performance is very high, while communication overhead is minimal. The simulations showed that BGR is superior to GPSR in realistic scenarios, even when applying several improvements of GPSR for mobility, radio irregularity, and location errors. This confirms that BGR is a highly suitable routing algorithm.

## 9.2 Future Perspectives

Although BGR is a complete and ready-to-use algorithm, there is potential for future improvements. In the following, some directions are presented.

An open issue is the implementation of the delivery semantics *nearest* introduced in Section 4.1. It requires large communication overhead, because the entire vicinity of the destination location must be searched for the nearest node.

Another challenge is to develop better recovery strategies for the 3D version of BGR. From the current strategies, only the augmented spherical sector covers the entire half-sphere, while the other strategies leave large gaps; nodes within these gaps are not used for forwarding. Recovery strategies that do not leave gaps between the forwarding volumes are an open topic.

The geocasting scheme could also be optimized. It is likely to be possible to improve the delivery ratio within the geocast region without using ordinary flooding. Different timer functions have to be investigated, or other flooding strategies might be taken into consideration.

The analytical framework presented in the thesis leaves still room for improvements. The calculated delivery probability is merely an approximation, because the expected progress is used in the formula instead of the progress distribution, and the calculated progress is based on the projection to the source-destination line. Formulas for computing more accurate probability values are a challenging task to find.

An implementation of BGR on real sensor node or MANET hardware is missing until now. A prototype implementation on the ScatterWeb platform [FUB] of parts of the algorithm does exist, but a full implementation and extensive test runs are still outstanding.





## References

- [AFO06a] Alaa Eddien Abdallah, Thomas Fevens, and Jaroslav Opatrny. Hybrid Position-Based 3D Routing Algorithms with Partial Flooding. In *Proceedings of the 19th Canadian Conference on Electrical and Computer Engineering (CCECE)*, pages 227–230, Ottawa, Canada, May 2006.
- [AFO06b] Alaa Eddien Abdallah, Thomas Fevens, and Jaroslav Opatrny. Randomized 3D Position-based Routing Algorithms for Ad-hoc Networks. In *Proceedings of the 3rd Annual International Conference on Mobile and Ubiquitous Systems (MobiQuitous)*, pages 1–8, San Jose, California, USA, July 2006.
- [AFO07] Alaa Eddien Abdallah, Thomas Fevens, and Jaroslav Opatrny. Power-Aware 3D Position-based Routing Algorithms for Ad Hoc Networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 3130–3135, Glasgow, Scotland, June 2007.
- [AH06] S. M. Nazrul Alam and Zygmunt Haas. Coverage and Connectivity in Three-Dimensional Networks. In *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 346–357, Los Angeles, California, USA, September 2006.
- [AKK04] Jamal N. Al-Karaki and Ahmed E. Kamal. Routing Techniques in Wireless Sensor Networks: A Survey. *IEEE Wireless Communications*, 11(6):6–28, December 2004.
- [APM05] Ian F. Akyildiz, Dario Pompili, and Tommaso Melodia. Underwater acoustic sensor networks: research challenges. *Elsevier Ad Hoc Networks Journal*, 3(3):257–279, May 2005.
- [AY05] Kemal Akkaya and Mohamed Younis. A survey on routing protocols for wireless sensor networks. *Elsevier Ad Hoc Networks*, 3(3):325–349, May 2005.
- [BHSS03] Brian Blum, Tian He, Sang Son, and John Stankovic. IGF: A state-free robust communication protocol for wireless sensor networks. Technical Report CS-2003-11, Department of Computer Science, University of Virginia, 2003.

## References

- [BM04] Sanjit Biswas and Robert Morris. Opportunistic Routing in Multi-Hop Wireless Networks. *ACM SIGCOMM Computer Communication Review*, 34(1):69–74, January 2004.
- [BMSU99] Prosenjit Bose, Pat Morin, Ivan Stojmenović, and Jorge Urrutia. Routing with Guaranteed Delivery in ad hoc Wireless Networks. In *Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 48–55, Seattle, Washington, USA, August 1999.
- [CACM03] Douglas S. J. De Couto, Daniel Aguayo, Benjamin A. Chambers, and Robert Morris. Performance of Multihop Wireless Networks: Shortest Path is Not Enough. *ACM SIGCOMM Computer Communication Review*, 33(1):83–88, January 2003.
- [CDV05a] Dazhi Chen, Jing Deng, and Pramod Varshney. A State-Free Data Delivery Protocol for Multihop Wireless Sensor Networks. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1818–1823, New Orleans, Louisiana, USA, March 2005.
- [CDV05b] Dazhi Chen, Jing Deng, and Pramod Varshney. On the Forwarding Area of Contention-Based Geographic Forwarding for Ad Hoc and Sensor Networks. In *Proceedings of the Second Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, pages 130–141, Santa Clara, California, USA, September 2005.
- [CK03] Chee-Yee Chong and Srikanta P. Kumar. Sensor Networks: Evolution, Opportunities, and Challenges. *Proceedings of the IEEE*, 91(8):1247–1256, August 2003.
- [CMN<sup>+</sup>05] Paolo Casari, Alessia Marcucci, Michele Nati, Chiara Petrioli, and Michele Zorzi. A Detailed Simulation Study of Geographic Random Forwarding (GeRaF) in Wireless Sensor Networks. In *Proceedings of the IEEE Military Communications Conference (MILCOM)*, volume 1, pages 59–68, Atlantic City, New Jersey, USA, October 2005.
- [CV07] Dazhi Chen and Pramod Varshney. A Survey of Void Handling Techniques for Geographic Routing in Wireless Networks. *IEEE Communications Surveys and Tutorials*, 9(1):50–67, 2007.
- [DKN08] Stephane Durocher, David Kirkpatrick, and Lata Narayanan. On Routing with Guaranteed Delivery in Three-Dimensional Ad Hoc Wireless

- Networks. In *Proceedings of the 9th International Conference on Distributed Computing and Networking (ICDCN)*, pages 546–557, Kolkata, India, January 2008.
- [FGL<sup>+</sup>05] Dario Ferrara, Laura Galluccio, Alessandro Leonardi, Giacomo Morabito, and Sergio Palazzo. MACRO: An Integrated MAC/Routing Protocol for Geographic Forwarding in Wireless Sensor Networks. In *Proceedings of the 24th Annual IEEE Conference on Computer Communications (INFOCOM)*, volume 3, pages 1770–1781, Miami, Florida, USA, March 2005.
- [Fin87] Gregory Finn. Routing and Addressing Problems in Large Metropolitan-Scale Internetworks. Technical Report ISI/RR-87-180, Information Sciences Institute, University of Southern California, March 1987.
- [FJ94] Sally Floyd and Van Jacobson. The Synchronization of Periodic Routing Messages. *IEEE/ACM Transactions on Networking (TON)*, 2(2):122–136, April 1994.
- [FUB] Computer Systems and Telematics Group at Freie Universität Berlin. ScatterWeb. <http://scatterweb.mi.fu-berlin.de>. Visited 2008-09-01.
- [FW08] Roland Flury and Roger Wattenhofer. Randomized 3D Geographic Routing. In *Proceedings of the 27th Annual IEEE Conference on Computer Communications (INFOCOM)*, Phoenix, Arizona, USA, April 2008.
- [FWMH03] Holger Füßler, Jörg Widmer, Martin Mauve, and Hannes Hartenstein. A Novel Forwarding Paradigm for Position-Based Routing (with Implicit Addressing). In *IEEE Computer Communications Workshop (CCW)*, pages 194–200, Dana Point, California, USA, 2003.
- [GEW<sup>+</sup>02] Deepak Ganesan, Deborah Estrin, Alec Woo, David Culler, Bhaskar Krishnamachari, and Stephen Wicker. Complex Behavior at Scale: An Experimental Study of Low-Power Wireless Sensor Networks. Technical Report CSD-TR 02-0013, UCLA Computer Science Department, February 2002.
- [GGH<sup>+</sup>01] Jie Gao, Leonidas J. Guibas, John Hershberger, Li Zhang, and An Zhu. Geometric Spanner for Routing in Mobile Networks. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 45–55, Long Beach, California, USA, October 2001.

## References

- [GS69] K. Ruben Gabriel and Robert R. Sokal. A new statistical approach to geographic variation analysis. *Systematic Zoology*, 18(3):259–278, September 1969.
- [GS03] Silvia Giordano and Ivan Stojmenović. *Ad Hoc Wireless Networking*, chapter Position Based Routing Algorithms For Ad Hoc Networks: A Taxonomy, pages 103–136. Kluwer Academic Publishers, November 2003.
- [HB03] Marc Heissenbüttel and Torsten Braun. BLR: A Beacon-Less Routing Algorithm for Mobile Ad-Hoc Networks. Technical Report IAM-03-001, University of Bern, Switzerland, March 2003.
- [HBBW04] Marc Heissenbüttel, Torsten Braun, Thomas Bernoulli, and Markus Wälchli. BLR: Beacon-Less Routing Algorithm for Mobile Ad-Hoc Networks. *Elsevier's Computer Communications Journal*, 27(11):1076–1086, July 2004.
- [HHB<sup>+</sup>03] Tian He, Chengdu Huang, Brian Blum, John Stankovic, and Tarek Abdelzaher. Range-Free Localization Schemes for Large Scale Sensor Networks. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 81–95, San Diego, California, USA, September 2003.
- [HLR03] Qingfeng Huang, Chenyang Lu, and Gruia-Catalin Roman. Spatiotemporal Multicast in Sensor Networks. In *Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 205–217, Los Angeles, California, USA, November 2003.
- [HPS02] Zygmunt J. Haas, Marc R. Pearlman, and Prince Samar. The Zone Routing Protocol (ZRP) for Ad Hoc Networks. IETF MANET Internet Draft, July 2002.
- [HSS06] Y. Thomas Hou, Yi Shi, and Hanif D. Sherali. Optimal Base Station Selection for Anycast Routing in Wireless Sensor Networks. *IEEE Transactions on Vehicular Technology*, 55(3), May 2006.
- [HTL04] Chi-Fu Huang, Yu-Chee Tseng, and Li-Chu Lo. The Coverage Problem in Three-Dimensional Wireless Sensor Networks. In *Proceedings of the Global Telecommunications Conference (GLOBECOM)*, pages 3182–3186, Dallas, Texas, USA, December 2004.
- [HYW<sup>+</sup>06] John Heidemann, Wei Ye, Jack Wills, Affan Syed, and Yuan Li. Research Challenges and Applications for Underwater Sensor Networking.

- In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pages 228–235, Las Vegas, Nevada, USA, April 2006.
- [IDL99] Chalermek Intanagonwiwat and Dante De Lucia. The Sink-based Any-cast Routing Protocol for Ad Hoc Wireless Sensor Networks. Technical Report 99-698, Computer Science Department, University of Southern California, 1999.
- [IT07] Hiroki Ishizuka and Yoshito Tobe. SenriGan: A Sensed-Point-Directed Geographic Routing for Sensor Networks. In *Proceedings of the Fourth International Conference on Networked Sensing Systems (INSS)*, pages 57–60, Braunschweig, Germany, June 2007.
- [JM96] David B. Johnson and David A. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. In T. Imielinski and H. Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.
- [JT87] John Jubin and Janet D. Tornow. The DARPA Packet Radio Network Protocols. *Proceedings of the IEEE*, 75(1):21–32, January 1987.
- [KFO05] George Kao, Thomas Fevens, and Jaroslav Opatrny. Position-Based Routing on 3-D Geometric Graphs in Mobile Ad Hoc Networks. In *Proceedings of the 17th Canadian Conference on Computational Geometry (CCCG)*, pages 88–91, Windsor, Ontario, Canada, August 2005.
- [KGKS05a] Young-Jin Kim, Ramesh Govindan, Brad Karp, and Scott Shenker. Geographic Routing Made Practical. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation*, Boston, Massachusetts, USA, May 2005.
- [KGKS05b] Young-Jin Kim, Ramesh Govindan, Brad Karp, and Scott Shenker. On the Pitfalls of Geographic Face Routing. In *Proceedings of the 2005 Joint Workshop on Foundations of Mobile Computing*, pages 34–43, Cologne, Germany, September 2005.
- [KGKS06] Young-Jin Kim, Ramesh Govindan, Brad Karp, and Scott Shenker. Lazy Cross-Link Removal for Geographic Routing. In *Proceedings of the 4th International Conference On Embedded Networked Sensor Systems (SenSys)*, pages 112–124, Boulder, Colorado, USA, November 2006.
- [KK00] Brad Karp and Hsiang-tsung Kung. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 243–254, Boston, Massachusetts, USA, August 2000.

## References

- [KKP99] Joseph M. Kahn, Randy H. Katz, and Kristofer S. J. Pister. Next Century Challenges: Mobile Networking for “Smart Dust”. In *Proceedings of the 5th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 271–278, Seattle, Washington, USA, August 1999.
- [KLH04] Yongjin Kim, Jae-Joon Lee, and Ahmed Helmy. Modeling and Analyzing the Impact of Location Inconsistencies on Geographic Routing in Wireless Networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8(1):48–60, January 2004.
- [KS06] Sungoh Kwon and Ness B. Shroff. Geographic Routing in the Presence of Location Errors. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 50(15):2902–2917, October 2006.
- [KSU99] Evangelos Kranakis, Harvinder Singh, and Jorge Urrutia. Compass Routing on Geometric Networks. In *Proceedings of the 11th Canadian Conference on Computational Geometry*, pages 51–54, Vancouver, Canada, August 1999.
- [KV00] Young-Bae Ko and Nitin H. Vaidya. Location-Aided Routing (LAR) in mobile ad hoc networks. *Wireless Networks*, 6(4):307–321, July 2000.
- [KV02] Young-Bae Ko and Nitin H. Vaidya. Flooding-Based Geocasting Protocols for Mobile Ad Hoc Networks. *Mobile Networks and Applications*, 7(6):471–480, December 2002.
- [KWZ03] Fabian Kuhn, Roger Wattenhofer, and Aaron Zollinger. Worst-Case Optimal and Average-Case Efficient Geometric Ad-Hoc Routing. In *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 267–278, Annapolis, Maryland, USA, June 2003.
- [KWZZ03] Fabian Kuhn, Roger Wattenhofer, Yan Zhang, and Aaron Zollinger. Geometric Ad-Hoc Routing: Of Theory and Practice. In *Proceedings of the 22nd ACM Symposium on Principles of Distributed Computing (PODC)*, pages 63–72, Boston, Massachusetts, USA, July 2003.
- [LBV05] Jean-Yves Le Boudec and Milan Vojnović. Perfect Simulation and Stationarity of a Class of Mobility Models. In *Proceedings of the 24th Annual IEEE Conference on Computer Communications (INFOCOM)*, volume 4, pages 2743–2754, Miami, Florida, USA, March 2005.

- [LJDC<sup>+</sup>00] Jinyang Li, John Jannotti, Douglas De Couto, David Karger, and Robert Morris. A Scalable Location Service for Geographic Ad Hoc Routing. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 120–130, Boston, Massachusetts, USA, August 2000.
- [LK03] Changling Liu and Jörg Kaiser. A Survey of Mobile Ad Hoc network Routing Protocols. Technical Report 2003-08, Department of Computer Structures, University of Ulm, Germany, 2003.
- [LLM06] Ben Leong, Barbara Liskov, and Robert Morris. Geographic Routing without Planarization. In *Proceedings of the 3rd Symposium on Networked Systems Design and Implementation*, pages 339–352, San Jose, California, USA, May 2006.
- [LP04] Loukas Lazos and Radha Poovendran. SeRLoc: Secure Range-Independent Localization for Wireless Sensor Networks. In *Proceedings of the 2004 ACM Workshop on Wireless Security*, pages 21–30, Philadelphia, Pennsylvania, USA, October 2004.
- [Mai04] Christian Maihöfer. A Survey of Geocast Routing Protocols. *IEEE Communications Surveys and Tutorials*, 6(2):32–42, 2004.
- [MGLA02] Marc Mosko and J. J. Garcia-Luna-Aceves. A self-correcting neighbor protocol for mobile ad-hoc wireless networks. In *Proceedings of the 11th International Conference on Computer Communications and Networks*, pages 556–560, Miami, Florida, USA, October 2002.
- [MMF<sup>+</sup>07] Mateusz Malinowski, Matthew Moskwa, Mark Feldmeier, Mathew Laibowitz, and Joseph A. Paradiso. CargoNet: A Low-Cost Micro-Power Sensor Node Exploiting Quasi-Passive Wakeup for Adaptive Asynchronous Monitoring of Exceptional Events. In *Proceedings of the 5th International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 145–159, Sydney, Australia, November 2007.
- [NI97] Julio C. Navas and Tomasz Imielinski. GeoCast - Geographic Addressing and Routing. In *Proceedings of the 3rd ACM/IEEE International Conference on Mobile Computing and Networking*, pages 66–76, September 1997.
- [PB94] Charles E. Perkins and Pravin Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In *Proceedings of the Conference on Communications Architectures, Protocols and Applications*, pages 234–244, London, United Kingdom, September 1994.



## References

- [PM05] Dario Pompili and Tommaso Melodia. Three-Dimensional Routing in Underwater Acoustic Sensor Networks. In *Proceedings of the ACM Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks*, pages 214–221, Montreal, Quebec, Canada, October 2005.
- [PPKS06] Sameera Poduri, Sundeep Pattem, Bhaskar Krishnamachari, and Gaurav S. Sukhatme. Sensor Network Configuration and the Curse of Dimensionality. In *Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets)*, Cambridge, Massachusetts, USA, May 2006.
- [PR99] Charles E. Perkins and Elizabeth M. Royer. Ad-hoc On-Demand Distance Vector Routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, New Orleans, Louisiana, USA, February 1999.
- [Rav04] Vlady Ravelomanana. Extremal Properties of Three-Dimensional Sensor Networks with Applications. *IEEE Transactions on Mobile Computing*, 3(3):246–257, July 2004.
- [RKS<sup>+</sup>03] Sylvia Ratnasamy, Brad Karp, Scott Shenker, Deborah Estrin, Ramesh Govindan, Li Yin, and Fang Yu. Data-Centric Storage in Sensor networks with GHT, A Geographic Hash Table. *Mobile Networks and Applications*, 8(4):427–442, August 2003.
- [SHG04] Karim Seada, Ahmed Helmy, and Ramesh Govindan. On the Effect of Localization Errors on Geographic Face Routing in Sensor Networks. In *Proceedings of the Third International Symposium on Information Processing in Sensor Networks*, pages 71–80, Berkeley, California, USA, April 2004.
- [SL01] Ivan Stojmenović and Xu Lin. Loop-Free Hybrid Single-Path/Flooding Routing Algorithms with Guaranteed Delivery for Wireless Networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(10):1023–1032, October 2001.
- [SMP02] Sasha Slijepcevic, Seapahn Megerian, and Miodrag Potkonjak. Location Errors in Wireless Embedded Sensor Networks: Sources, Models, and Effects on Applications. *ACM SIGMOBILE Mobile Computing and Communications Review*, 6(3):67–78, July 2002.
- [SMPR07] Juan A. Sanchez, Rafael Marin-Perez, and Pedro M. Ruiz. BOSS: Beacon-less On Demand Strategy for Geographic Routing in Wireless Sensor Networks. In *Proceedings of the 4th IEEE Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, Pisa, Italy, October 2007.

- [Sto04] Ivan Stojmenović. Geocasting with Guaranteed Delivery in Sensor Networks. *IEEE Wireless Communications*, 11(6):29–37, December 2004.
- [STS02] Curt Schurgers, Vlasios Tsiatsis, and Mani B. Srivastava. STEM: Topology Management for Energy Efficient Sensor Networks. In *Proceedings of the IEEE Aerospace Conference*, volume 3, pages 1099–1108, March 2002.
- [SWR05] Rahul Shah, Adam Wolisz, and Jan Rabaey. On the performance of geographical routing in the presence of localization errors. In *Proceedings of IEEE International Conference on Communications (ICC)*, pages 2979–2985, Seoul, Korea, May 2005.
- [SWW05] Rahul Shah, Sven Wiethölter, and Adam Wolisz. When Does Opportunistic Routing Make Sense? In *Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 350–356, Kauai, Hawaii, USA, March 2005.
- [SZHK04] Karim Seada, Marco Zuniga, Ahmed Helmy, and Bhaskar Krishnamachari. Energy-Efficient Forwarding Strategies for Geographic Routing in Lossy Wireless Sensor Networks. In *Proceedings of the ACM 2nd Conference on Embedded Networked Sensor Systems (SenSys)*, pages 108–121, Baltimore, Maryland, USA, November 2004.
- [TI] Texas Instruments. Low Cost, Low-Power 2.4 GHz RF Transceiver Designed for Low-Power Wireless Apps in 2.4 GHz ISM Band - CC2500 - TI Product Folder. <http://focus.ti.com/docs/prod/folders/print/cc2500.html>. Visited 2008-09-01.
- [Tou80] Godfried T. Toussaint. The Relative Neighborhood Graph of a Finite Planar Set. *Pattern Recognition*, 12(4):261–268, 1980.
- [TRV<sup>+</sup>05] Volker Turau, Christian Renner, Marcus Venzke, Sebastian Waschik, Christoph Weyer, and Matthias Witt. The Heathland Experiment: Results And Experiences. In *Proceedings of the Workshop on Real-World Wireless Sensor Networks (REALWSN)*, Stockholm, Sweden, June 2005.
- [TTS05] Niwat Thepvilojanapong, Yoshito Tobe, and Kaoru Sezaki. HAR: Hierarchy-Based Anycast Routing Protocol for Wireless Sensor Networks. In *Proceedings of the IEEE/IPSJ International Symposium on Applications and the Internet (SAINT)*, pages 204–212, Trento, Italy, February 2005.

## References

- [TWW06] Volker Turau, Matthias Witt, and Marcus Venzke. Field Trials with Wireless Sensor Networks: Issues and Remedies. In *Proceedings of the Second International Conference on Wireless and Mobile Communications (ICWMC)*, Bucharest, Romania, July 2006.
- [TWW05] Volker Turau, Christoph Weyer, and Matthias Witt. Ein robustes Datenmonitoring-Verfahren für Sensornetzwerke. *it – Information Technology*, 47(2):63–69, April 2005.
- [TWW06] Volker Turau, Matthias Witt, and Christoph Weyer. Analysis of a Real Multi-hop Sensor Network Deployment: The Heathland Experiment. In *Proceedings of the Third International Conference on Networked Sensing Systems (INSS)*, Chicago, Illinois, USA, June 2006.
- [USC] Information Sciences Institute (ISI) of the University of Southern California (USC). The Network Simulator - ns-2. <http://www.isi.edu/nsnam/ns/>. Visited 2008-09-01.
- [Wey06] Christoph Weyer. A Survey of Clustering Algorithms for Wireless Sensor Networks. Project Work, Hamburg University of Technology, Germany, April 2006.
- [WSBC04] Kamin Whitehouse, Cory Sharp, Eric Brewer, and David Culler. Hood: A Neighborhood Abstraction for Sensor Networks. In *Proceedings of the 2nd International Conference On Mobile Systems, Applications And Services*, pages 99–110, Boston, Massachusetts, USA, June 2004.
- [WT05] Matthias Witt and Volker Turau. BGR: Blind Geographic Routing for Sensor Networks. In *Proceedings of the Third International Workshop on Intelligent Solutions in Embedded Systems (WISES)*, pages 51–61, Hamburg, Germany, May 2005.
- [WT06a] Matthias Witt and Volker Turau. Delivery Semantics for Geographic Routing. In *Pedro José Marrón (ed.): 5. GI/ITG KuVS Fachgespräch Drahtlose Sensornetze. Technical Report 2006/07*, pages 87–91. University of Stuttgart, Germany, July 2006.
- [WT06b] Matthias Witt and Volker Turau. The Impact of Location Errors on Geographic Routing in Sensor Networks. In *Proceedings of the Second International Conference on Wireless and Mobile Communications (ICWMC)*, Bucharest, Romania, July 2006.
- [WT07] Matthias Witt and Volker Turau. Geographic Routing in 3D. In *6. GI/ITG KuVS Fachgespräch Drahtlose Sensornetze. Technical Report, ISSN 0935-3232*, pages 75–78. RWTH Aachen, Germany, July 2007.

- [WWT05] Matthias Witt, Christoph Weyer, and Volker Turau. Monitoring Energy Consumption In Wireless Sensor Networks. In *Kay Römer (ed.): 4. GI/ITG KuVS Fachgespräch Drahtlose Sensornetze. Technical Report TR481*, pages 77–80. ETH Zürich, Switzerland, March 2005.
- [XHE01] Ya Xu, John Heidemann, and Deborah Estrin. Geography-informed Energy Conservation for Ad Hoc Routing. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 70–84, Rome, Italy, July 2001.
- [YGE01] Yan Yu, Ramesh Govindan, and Deborah Estrin. Geographical and Energy Aware Routing: a recursive data dissemination protocol for wireless sensor networks. Technical Report UCLA/CSD-TR-01-0023, UCLA Computer Science Department, May 2001.
- [YLN03] Jungkeun Yoon, Mingyan Liu, and Brian Noble. Random Waypoint Considered Harmful. In *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, pages 1312–1321, San Fransisco, California, USA, April 2003.
- [ZHKS04] Gang Zhou, Tian He, Sudha Krishnamurthy, and John Stankovic. Impact of Radio Irregularity on Wireless Sensor Networks. In *Proceedings of the 2nd International Conference On Mobile Systems, Applications And Services*, pages 125–138, Boston, Massachusetts, USA, June 2004.
- [ZK04] Marco Zuniga and Bhaskar Krishnamachari. Analyzing the Transitional Region in Low Power Wireless Links. In *Proceedings of the First IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, pages 517–526, Santa Clara, California, USA, October 2004.
- [Zor04] Michele Zorzi. A new contention-based MAC protocol for geographic forwarding in ad hoc and sensor networks. In *Proceedings of the IEEE International Conference on Communications (ICC)*, volume 6, pages 3481–3485, Paris, France, June 2004.



# Curriculum Vitae

## Personal Data

Name	Matthias Witt
Date of Birth	November 30, 1975
Place of Birth	Reinbek, Germany
Nationality	German

## Education

2004–2008	Research and teaching assistant at the Hamburg University of Technology, Institute of Telematics
1996–2004	Studies in Computer Science (Informatik) at the University of Hamburg with diploma degree (Diplom-Informatiker), final grade: very good
1995–1996	Civilian service (Zivildienst)
1986–1995	Gymnasium, final degree: Abitur, final grade: very good (1.4)
1982–1986	Primary school







