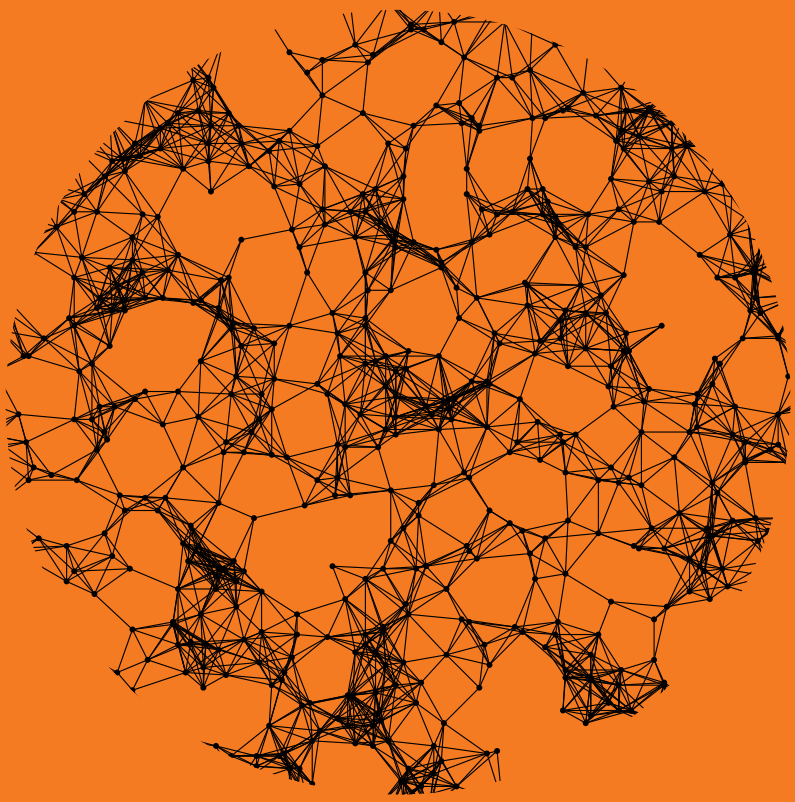
Department of Communications and Networking

Forwarding Capacity in Large Wireless Multihop Networks

A Computational Approach

Jarno Nousiainen



Aalto University

DOCTORAL DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 107/2013

Forwarding Capacity in Large Wireless Multihop Networks

A Computational Approach

Jarno Nousiainen

A doctoral dissertation completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Electrical Engineering, at a public examination held at the lecture hall S1 of the school on 28 June 2013 at 12.

Aalto University
School of Electrical Engineering
Department of Communications and Networking

Supervising professors
Professor Samuli Aalto

Thesis advisors
Professor (emer.) Jorma Virtamo
Docent Pasi Lassila

Preliminary examiners
Professor Stavros Toumpis, Athens University of Economics and Business, Greece
Doctor Petteri Mannersalo, VTT Technical Research Centre of Finland, Finland

Opponents
Professor Thomas Bonald, Télécom ParisTech, France

Aalto University publication series
DOCTORAL DISSERTATIONS 107/2013

© Jarno Nousiainen

ISBN 978-952-60-5234-2 (printed)
ISBN 978-952-60-5235-9 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-60-5235-9

Unigrafia Oy
Helsinki 2013

Finland

441 697
Printed matter

Aalto University

Abstract

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Author
Jarno Nousiainen

Name of the doctoral dissertation
Forwarding Capacity in Large Wireless Multihop Networks: A Computational Approach

Publisher School of Electrical Engineering

Unit Department of Communications and Networking

Series Aalto University publication series DOCTORAL DISSERTATIONS 107/2013

Field of research Teletraffic Theory

Manuscript submitted 4 February 2013 **Date of the defence** 28 June 2013

Permission to publish granted (date) 26 April 2013 **Language** English

☐ **Monograph** ☒ **Article dissertation (summary + original articles)**

Abstract

Wireless multihop networks are networks without any fixed infrastructure. This thesis concentrates on a network consisting of a plethora of immobile nodes communicating with each other over a shared wireless channel. The intrinsic nature of the shared wireless channel makes it difficult to efficiently avoid interference between the transmissions, and the exact capacity of such a network is in many respects an open question.

At first, we characterize the capacity problem in a massively dense wireless network where a separation of scales emerges, and the problem can be separated into two different subproblems. The two subproblems loosely correspond to routing at the global scale and forwarding at the local scale. We focus on the latter one and study the microscopic level multidirectional forwarding capacity problem that considers an infinitely large network's capability to relay information. Because of the complexity of analyzing a large random network and wireless interference, the main approach is to construct algorithms that produce numerical bounds or estimates for the forwarding capacity, and simulate them for large network realizations.

The methods used for studying the forwarding capacity are presented in two parts. The first part considers the instantaneous forwarding capacity. The instantaneous forwarding capacity can be achieved temporarily but cannot be maintained for a longer time period. It is a natural upper bound for the actual forwarding capacity and can be analyzed with more complex ways of modeling interference, such as the SINR-based models, in addition to the simple Boolean interference model.

The actual forwarding capacity with multihop traffic under the Boolean interference model is considered in the second part. In this part, the upper bound provided by the instantaneous capacity is tightened for a small number of neighbor nodes, where it is less accurate. We also provide a lower bound that shows a notable improvement compared with previous results for uncoordinated opportunistic forwarding. Finally, an estimate is found for the forwarding capacity. The dependence of the estimate on the directional distribution of the traffic is studied to determine the possible gain from interleaving traffic in different directions compared with time sharing between the directions. Eventually, it is illustrated how the results for the forwarding capacity can be used with the macroscopic level results to obtain the total capacity of a large wireless network. The thesis hence makes it possible to calculate a numerical estimate for the total capacity.

Keywords forwarding capacity, wireless multihop networks, massively dense networks, scheduling, spatial reuse

ISBN (printed) 978-952-60-5234-2 **ISBN (pdf)** 978-952-60-5235-9

ISSN-L 1799-4934 **ISSN (printed)** 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Espoo **Location of printing** Helsinki **Year** 2013

Pages 224 **urn** http://urn.fi/URN:ISBN:978-952-60-5235-9

Tekijä
Jarno Nousiainen

Väitöskirjan nimi
Suuren langattoman monihyppyisen verkon välityskapasiteetin määrittäminen

Julkaisija Sähkötekniikan korkeakoulu

Yksikkö Tietoliikenne- ja tietoverkkotekniikan laitos

Sarja Aalto University publication series DOCTORAL DISSERTATIONS 107/2013

Tutkimusala Teleliikenneteoria

Käsikirjoituksen pvm 04.02.2013 **Väitöspäivä** 28.06.2013

Julkaisuluvan myöntämispäivä 26.04.2013 **Kieli** Englanti

☐ **Monografia** ☒ **Yhdistelmäväitöskirja (yhteenveto-osa + erillisartikkelit)**

Tiivistelmä

Langattomat monihyppyverkot ovat verkkoja, joilla ei ole kiinteää rakennetta. Tämä työ tarkastelee verkkoa, joka koostuu suuresta joukosta liikkumattomia päätelaitteita, jotka kommunikoivat keskenään käyttäen yhteistä langatonta kanavaa. Tämä jaettu media tekee lähetysten välisten häiriöiden tehokkaan välttämisen vaikeaksi, ja kyseisen verkon tarkka kapasiteetti on siksi monelta osin avoin kysymys.

Aluksi työssä luonnehditaan kapasiteettiongelmaa massiivisen tiheässä langattomassa verkossa, jossa voidaan erottaa mikroskooppinen ja makroskooppinen mittakaava, ja verkon kapasiteetin maksimointi jakautuu kahdeksi erilliseksi ongelmaksi. Nämä kaksi aliongelmaa vastaavat löyhästi kokonaistason reititystä ja paikallisen tason välitystä. Tämä työ keskittyy jälkimmäiseen ja tutkii mikroskooppisen tason monisuuntaista välityskapasiteettiongelmaa, joka käsittelee äärettömän suuren verkon kykyä välittää informaatiota. Suurten satunnaisten verkkojen ja langattoman häiriön analysoinnin vaikeudesta johtuen ongelmaa on lähestytty konstruoimalla algoritmeja, jotka tuottavat numeerisia ylä- tai alarajoja tai estimaatteja välityskapasiteetille, ja simuloimalla niitä suurissa verkkorealisaatioissa.

Välityskapasiteetin tutkimisessa käytetyt menetelmät on esitetty kahdessa osassa. Ensimmäinen osa käsittelee hetkellistä välityskapasiteettia. Hetkellinen välityskapasiteetti voidaan saavuttaa tilapäisesti, mutta sitä ei voida ylläpitää pidempiä ajanjaksoja. Se on luonnollinen yläraja todelliselle välityskapasiteetille, ja sitä voidaan analysoida myös monimutkaisemmilla häiriömalleilla, kuten signaalin ja häiriön suhteeseen perustuvilla malleilla, yksinkertaisen Boolen häiriömallin lisäksi.

Toisessa osassa tarkastellaan todellista välityskapasiteettia monihyppyiselle liikenteelle, kun häiriömalliksi oletetaan Boolen malli. Tässä osassa hetkellisen kapasiteetin muodostamaa ylärajaa tiukennetaan naapurisolmujen lukumäärän ollessa pieni, jolloin se on löyhä. Lisäksi annetaan alaraja, joka on huomattava parannus verrattuna aikaisempiin tuloksiin koordinoimattomalle opportunistiselle välitykselle. Seuraavaksi työssä esitetään arvio välityskapasiteetista. Arvion riippuvuutta liikenteen suuntajakaumasta tutkitaan, jotta voidaan määrittää erisuuntaisten liikennevirtojen yhdistämisestä saatava mahdollinen hyöty verrattuna aikajakoon suuntien välillä. Lopuksi havainnollistetaan, kuinka välityskapasiteetille tuotetut tulokset voidaan yhdistää makroskooppisen tason tulosten kanssa suuren langattoman verkon kokonaiskapasiteetin määrittämiseksi. Työ siis paitsi valottaa kapasiteetin muodostumiseen vaikuttavia tekijöitä myös mahdollistaa numeerisen arvion laskemisen kokonaiskapasiteetille.

Avainsanat välityskapasiteetti, langattomat monihyppyverkot, massiivisen tiheät verkot, aikataulutus, spatiaalinen uudelleenkäyttö

ISBN (painettu) 978-952-60-5234-2 **ISBN (pdf)** 978-952-60-5235-9

ISSN-L 1799-4934 **ISSN (painettu)** 1799-4934 **ISSN (pdf)** 1799-4942

Julkaisupaikka Espoo **Painopaikka** Helsinki **Vuosi** 2013

Sivumäärä 224 **urn** http://urn.fi/URN:ISBN:978-952-60-5235-9

Preface

When my studies were at the point of choosing my major and minor subjects, I only chose a major. Later, when browsing through the offered courses, I came across teletraffic theory. After a brief meeting with Professor Jorma Virtamo, the minor was selected. I had to get an approval for the combination, as it wasn't on the recommended list. This minor event turned out to be a major juncture.

In the summer of 2007, I started working in the Networking Laboratory. This manuscript is the result of the work that started back then, first as a special assignment, and then as a master's thesis. The work has been carried out in the ABI, AWA and HEWINETS projects funded by the Finnish Funding Agency for Technology and Innovation (TEKES) and companies like Ericsson, Nokia Siemens Networks and Cassidian Systems. Since June 2009, I have had the great opportunity to be a student in the Graduate School in Electronics, Telecommunications and Automation (GETA), which has provided the majority of my funding. In addition, I'm grateful for the personal scholarships from the TES and Nokia foundation.

I wish to express my gratitude to my original supervisor, Professor Jorma Virtamo, for the enthusiasm and ideas that made this thesis possible. I'm deeply grateful for the privilege of receiving his guidance even after his retirement. I would like to thank the supervisor of this thesis, Professor Samuli Aalto, for bringing a much-needed fresh perspective to the final phases of the work. In addition, I'm thankful for my instructor, Docent Pasi Lassila, with whom I've shared an office with. It's been truly helpful to get instant feedback and suggestions with sharp insight. A special thank-you also goes to all past and present members of the teletraffic theory group. I would further like to thank Professor Patrick Thiran for hosting my visit at the EPFL and the pre-examiners of this thesis Professor Stavros Toumpis and Dr. Petteri Mannersalo for their valuable comments.

During my time here, the name of the unit has changed to the Department of Communications and Networking, even the name of the university has changed, but the good working atmosphere has remained. For that, I would like to thank all of the staff. Finally, I would like to thank my family, friends and everyone who has helped me to get my mind of the work for a while.

Espoo, May 28, 2013,

Jarno Nousiainen

Contents

List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

I Jarno Nousiainen, Jorma Virtamo and Pasi Lassila. Forwarding Capacity of an Infinite Wireless Network. In *Proceedings of the 11th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '08)*, pages 177–184, Vancouver, Canada, October 2008.

II Jarno Nousiainen and Pasi Lassila. Approximating Maximum Directed Flow in a Large Wireless Network. In *Proceedings of the IEEE International Conference on Communications (ICC '09)*, pages 1–6, Dresden, Germany, June 2009.

III Jarno Nousiainen, Jorma Virtamo and Pasi Lassila. Maximum Weight Independent Sets in an Infinite Plane. In *Proceedings of the 21st International Teletraffic Congress (ITC 21)*, pages 1–8, Paris, France, September 2009.

IV Jarno Nousiainen, Jorma Virtamo and Pasi Lassila. On the Achievable Forwarding Capacity of an Infinite Wireless Network. In *Proceedings of the 13th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '10)*, pages 151–159, Bodrum, Turkey, October 2010.

V Jarno Nousiainen, Jorma Virtamo and Pasi Lassila. Maximum Weight Independent Sets in an Infinite Plane with Uni- and Bidirectional Interference Models. *Annals of Telecommunications*, volume 66, numbers 1–2, pages 119–132, February 2011.

VI Jarno Nousiainen, Jorma Virtamo and Pasi Lassila. Optimal Transmission Modes by Simulated Annealing. In *Proceedings of the 6th ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks (PM2HW2N '11)*, pages 101–108, Miami Beach, FL, USA, October 2011.

VII Jarno Nousiainen, Pasi Lassila and Jorma Virtamo. Instantaneous Forwarding Capacity under the SINR Threshold Interference Model. In *Proceedings of the 11th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net 2012)*, pages 67–74, Ayia Napa, Cyprus, June 2012.

VIII Jarno Nousiainen, Jorma Virtamo and Pasi Lassila. Multidirectional Forwarding Capacity in a Massively Dense Wireless Network. In *Proceedings of the 24th International Teletraffic Congress (ITC 24)*, pages 1–8, Krakow, Poland, September 2012.

Author's Contribution

Publication I: "Forwarding Capacity of an Infinite Wireless Network"

The publication is a joint work between the authors. The idea for the moving window algorithm came from Prof. Jorma Virtamo. The present author has implemented the algorithm, carried out the simulations and drafted the paper.

Publication II: "Approximating Maximum Directed Flow in a Large Wireless Network"

The publication is a joint work between the authors. The idea of utilizing the clique formulation came from Prof. Jorma Virtamo. The present author has constructed the approximation algorithm and performed the simulations.

Publication III: "Maximum Weight Independent Sets in an Infinite Plane"

The publication is a joint work between the authors. The idea for the asymptotic study came from Prof. Jorma Virtamo who did the analysis of case B. The present author has done the numerical evaluations and drafted the paper with the exception of Section IV.B.

Publication IV: "On the Achievable Forwarding Capacity of an Infinite Wireless Network"

The publication is a joint work between the authors. The idea for the path scheduling algorithm came from the present author. The present author has implemented the algorithms and carried out the simulations.

Publication V: "Maximum Weight Independent Sets in an Infinite Plane with Uni- and Bidirectional Interference Models"

This publication is based on Publication III. The extensions are by the present author.

Publication VI: "Optimal Transmission Modes by Simulated Annealing"

The publication is a joint work between the authors. The present author has implemented the algorithms and carried out the simulations.

Publication VII: "Instantaneous Forwarding Capacity under the SINR Threshold Interference Model"

The publication is a joint work between the authors. The present author has done the numerical evaluations and drafted the publication.

Publication VIII: "Multidirectional Forwarding Capacity in a Massively Dense Wireless Network"

The publication is a joint work between the authors. The present author has implemented the algorithms and carried out the simulations.

List of Abbreviations

CAA	Clique approximation algorithm
CDMA	Code division multiple access
CSMA	Carrier sense multiple access
FC	Forwarding capacity
FDMA	Frequency division multiple access
GMWS	Greedy maximum weight scheduling
IFC	Instantaneous forwarding capacity
LP	Linear programming
MANET	Mobile ad hoc network
MWA	Moving window algorithm
MWIS	Maximum weight independent set
MWS	Maximum weight scheduling
PPP	Poisson point process
PSA	Path scheduling algorithm
RTS/CTS	Request to send / clear to send
SA	Simulated annealing
SINR	Signal-to-interference-and-noise ratio
WMN	Wireless mesh network
WSN	Wireless sensor network

List of Symbols

A	area
$\mathcal{A}$	network domain, $\mathcal{A} \subset \mathbb{R}^2$
$c(l)$	capacity of an edge, $c : \mathcal{L} \to \mathbb{R}^+$
$c(Q)$	capacity of a cut, $c : \mathcal{Q} \to \mathbb{R}^+$
C	nominal link capacity
$\mathcal{C}$	clique, $\mathcal{C} \subset \mathcal{V}$
$f(l)$	flow, $f : \mathcal{L} \to \mathbb{R}^+$
$f(\vartheta)$	directional distribution
g	attenuation factor, path loss function
$\mathcal{G}$	directed graph, $\mathcal{G} = (\mathcal{V}, \mathcal{L})$
I^*	multidirectional forwarding capacity (FC)
$\hat{I}$	instantaneous forwarding capacity (IFC)
I_1^*	single-directional forwarding capacity
I_∞^*	FC for isotropic traffic
$\mathcal{I}$	interference graph
l	edge/link
$\mathcal{L}$	set of edges/links
m	set of transmitting links/trasmission mode
$\mathcal{M}$	set of transmission modes
n	node density
N	number of nodes
$\mathcal{N}$	flow network, $\mathcal{N} = (\mathcal{G}, c, \mathcal{S}, \mathcal{T})$
P_0	transmission power

$\mathcal{P}$	routing system
q	queue length
Q	cut, $Q = (Q_S, Q_T)$
$\mathcal{Q}$	set of cuts
$r(l)$, r_l	receiving node of a link
$R_l(m)$	spectral efficiency
$\mathcal{S}$	set of start nodes/sources
t	time/time slot
$t(l)$, t_l	transmitting node of a link
T	temperature parameter
$\mathcal{T}$	set of terminal nodes/sinks
$u(\cdot)$	dimensionless component of FC
U	IFC excluding capacity, $U = \hat{I}/C$
v	vertex/node
$\boldsymbol{v}$	location of a node
$\mathcal{V}$	set of vertices/nodes
w_l	weight of a link
$w(f)$	value of flow $w : f \to \mathbb{R}^+$
$w(m)$	weight of a transmission mode
α	attenuation coefficient
$\boldsymbol{\alpha}$	schedule, $\boldsymbol{\alpha} = \{t_1, \ldots, t_{\lvert\mathcal{M}\rvert}\}$
$\delta(\vartheta)$	Dirac delta function
θ	SINR threshold
ϑ	angle/direction
λ	traffic demand density
Λ	total traffic load
ν	mean node degree
ξ	point process
ρ	transmission range
σ^2	thermal noise power
φ	angular flux of packets
Φ	scalar flux of packets

1. Introduction

1.1 Wireless multihop networks

Over the past few years, advances in wireless technology and decaying prices have brought large-scale wireless multihop networks closer and closer to reality. These are networks without any fixed infrastructure. Packets are relayed from source to destination in several short hops, and the network nodes have a dual role, both as terminals and as routers.

Wireless multihop networks are divided into a few main groups. Mobile ad hoc networks (MANETs) are decentralized networks that can be formed with a minimal amount of planning [RT99, MWH01, SSV12]. A self-configuring mode of operation is required as the devices are free to move. Wireless sensor networks (WSNs) consist of sensor nodes that monitor their surroundings and cooperatively forward the collected measurement data typically to a gateway node [ASSC02, TM03, YMG08]. Wireless mesh networks (WMNs) can be added to the list as a special case. They may be used to provide additional, more dynamic, connectivity to a fixed infrastructure in certain areas, but they often have a more planned configuration than a typical ad hoc network [VWM13, BHG12, GRDK13]. The predicted rapid increase of wireless devices that connect to the Internet is related to the concept of the Internet of Things (IoT). It also encompasses parts, like WSNs, that operate in a multihop fashion [AIM10].

Naturally, wireless multihop networks have attracted significant interest in the research community. However, the intrinsic nature of the shared wireless channel has made them a challenging topic. Efficient spatial reuse of the channel requires multiple links between the nodes to be activated simultaneously, but the interference caused by concurrent trans-

missions makes the scheduling difficult and prone to collisions of transmissions, preventing successful packet reception.

The capacity of the network can be defined using throughput. The throughput describes the sum of the average data rates that are delivered to all destinations in the network. It is usually measured in bits per second or its multiples like packets per time slot. As the throughput capacity depends on many things such as the distances between the sources and the destinations or delivery semantics (unicast, broadcast, etc.), it is sometimes more appropriate to study the rate together with the progress of information. Transport capacity measures the network performance in bit-meters per second, where one bit-meter means that one bit has been transported a distance of one meter toward its destination [GK00].

1.2 Contributions of the thesis

This thesis concentrates on networks consisting of a plethora of immobile, randomly located nodes communicating with each other over a shared wireless channel. The nearest real counterpart would, consequently, be a large WSN with message exchange between different parts of the network. The exact capacity of such a network is, in many respects, an open question. In this thesis, we attempt to shed light on these questions and on how the total capacity of the network is formed. Especially, we concentrate on the network's capability to relay information. It is measured using forwarding capacity that gives the density of packets multiplied by their mean velocity in their given directions (similarly to the transport capacity per unit area). In summary, the main contributions of this thesis are the following:

- We characterize the capacity problem in a massively dense wireless network and extend the knowledge on how it can be separated into two different subproblems.

- We focus on one of the subproblems, namely the microscopic level multidirectional forwarding capacity, that considers the network's capability to relay information.

- We explore the maximum weight per unit area of an independent set of links on an infinite plane, which can be used for obtaining an upper bound for the forwarding capacity.

- We construct new algorithms that produce numerical results on the upper and lower bounds of the forwarding capacity.

- We find an estimate for the forwarding capacity under our network model and study its dependence on the directional distribution of the traffic.

- We discuss how the forwarding capacity affects the total capacity of a large wireless network.

The following lists the implemented algorithms in more detail. The maximum weight independent set problem, corresponding to the instantaneous forwarding capacity, is studied using two numerical algorithms: the moving window algorithm and the simulated annealing algorithm. An asymptotic analysis is performed to examine the capacity with network densities beyond the capability of the numerical algorithms. The instantaneous forwarding capacity provides an upper bound for the actual forwarding capacity with multihop traffic. Further bounds are obtained by constructing two new numerical algorithms. The clique approximation algorithm and the path scheduling algorithm are both based on the LP formulation of the problem. The forwarding capacity is estimated using a greedy maximum weight scheduling (GMWS) algorithm. The algorithm is improved to be able to better handle traffic with different directional distributions. The simulated annealing algorithm is applied with the Boolean interference model as well as with SINR-based interference models. In the rest of the cases, the Boolean interference model is assumed.

The locations of the nodes are modeled using a homogeneous Poisson point process. In most cases, the same methodologies and algorithms can be used with other stationary and isotropic point processes as well. As a result, the algorithms produce estimates for the microscopic level forwarding capacity. The forwarding capacity describes how much traffic can be routed through a certain area of the network. Together with the information about the routing and the resulting loads, the new results on

the forwarding capacity allow us to calculate a numerical estimate for the total capacity.

1.3 Outline of the thesis

The outline of the thesis is as follows. In Chapter 2, we introduce the used models including the graph notations and models for node deployment and interference. We also review closely related models found in the literature.

In Chapter 3, we characterize the capacity problem in a massively dense wireless network and present the state-of-the-art results on the capacity of wireless multihop networks that support our work. This chapter presents the main research questions and the scope of the thesis. First, we give a background on the topic and then move on to the literature review. The main subjects of the review are the methods that in theory are able to achieve the maximum capacity and the scaling of the capacity as a function of the network size. We then continue with the very large networks and consider a situation where the capacity problem can be separated on two spatial scales. The two subproblems loosely correspond to routing at the global scale and forwarding at the local scale. At this point, we introduce the concept of forwarding capacity that is the main topic of the thesis.

Chapter 4 considers the instantaneous forwarding capacity, i.e., the capacity that can be achieved temporarily but cannot be maintained for a longer time period. The instantaneous capacity is, of course, an upper bound for the actual forwarding capacity. This simplification allows us to characterize the asymptotic behavior of the capacity when the group of other nodes that a node is able to communicate with (i.e., the neighborhood) is very large. Another advantage is that we are able to compare a variety of different interference models. We implement two numerical algorithms for studying the problem whose results are complemented by an asymptotic analysis of the instantaneous forwarding capacity. The moving window algorithm is able to find the instantaneous forwarding capacity for a network with limited height but unlimited length. Simulated annealing is a probabilistic method suitable for studying a variety of interference models.

Chapter 5 represents the results on the forwarding capacity and the techniques used to obtain them. We approach the problem from differ-

ent angles. On one hand, we tighten the upper bounds obtained from the instantaneous capacity with small neighborhoods, where they are not accurate. On the other hand, we provide lower bounds that show a notable improvement compared with the uncoordinated opportunistic forwarding. Finally, we aim for the sharpest possible estimate for the forwarding capacity and discuss the significance of the results on the overall capacity of the network. We construct two novel algorithms for assessing the capacity, one for an upper bound and another for a lower bound. The clique approximation algorithm studies the necessary constraints of an LP formulation of the problem. The path scheduling algorithm finds a densely packed set of paths that do not interfere with each other and, thus, simplify the scheduling problem. A final estimate is obtained by implementing a greedy version of the optimal maximum weight scheduling algorithm. Solving the maximum weight independent set problem repeatedly in a large-scale network is very complex, and a greedy approach is required for obtaining the results.

Chapter 6 summarizes the work. We conclude by briefly going through the problem and the used methodology, which, because of the complexity of the problem, is mainly based on simulations. We also recapitulate the role of the developed algorithms in obtaining new information about the quantity of interest: the forwarding capacity.

2. Network modeling

A mathematical model of a real-life phenomenon is always an attempt to capture the essential aspects of the system. Building a model involves a trade-off between simplicity and accuracy, and the model should be selected to match the objectives. When only little is known about the subject, the tractability of the model is a significant advantage. This chapter introduces the models used in this thesis and delineates how they relate to other models typical for analyzing wireless multihop networks.

The most obvious way to model a set of nodes where some pairs are connected by links is a graph. We start by reviewing the graph definitions used in this thesis. Next, we define the Poisson point process, which is used to model the random locations of the nodes. Finally, we study the interaction between the nodes over the shared wireless medium and the interference that is intrinsic to wireless networks.

2.1 Graphs

The most common model used with communication networks is probably a graph. A graph is an abstract representation of a set of vertices where some pairs of the vertices are connected by edges. Next, we go through the terminology used in this work.

When the nodes of the network correspond to the vertices, $\mathcal{V}$, of a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{L})$, there exists an edge $(u, v) \in \mathcal{L},\ u, v \in \mathcal{V}$ if node u is able to transmit data to node v, i.e., u and v form a link. If we denote a link by $l = (u, v)$, then $t(l)$ and $r(l)$ refer to the transmitting node, u, and the receiving node, v, respectively. In the interference graph, $\mathcal{I}$, the links of the network correspond to vertices, and two vertices are adjacent if the corresponding links interfere with each other. With the term network we

refer to a pair $(\mathcal{G}, c)$ of a digraph and a mapping $c : \mathcal{L} \to \mathbb{R}^+$. In general, $c(l)$ is called the capacity of edge l.

Let us consider a network $(\mathcal{G}, c)$ where we distinguish between two disjoint sets of special vertices: the start nodes $\mathcal{S}$ and the terminal nodes $\mathcal{T}$, or the sources and the sinks. Now, we have a structure $\mathcal{N} = (\mathcal{G}, c, \mathcal{S}, \mathcal{T})$ that is often called a flow network, and we can define a flow in the network. A mapping $f : \mathcal{L} \to \mathbb{R}^+$ is a flow if it satisfies the following conditions:

$$f(l) \in [0, c(l)] \quad \forall\, l \in \mathcal{L}, \tag{2.1}$$

$$\sum_{r(l)=v} f(l) = \sum_{t(l)=v} f(l) \quad \forall\, v \in \mathcal{V} \setminus (\mathcal{S} \cup \mathcal{T}). \tag{2.2}$$

The first, the feasibility condition, guarantees that there is a non-negative bounded flow through every edge, and the second, the flow conservation condition, means that flows are preserved (except at the sources and the sinks). The value of flow f is

$$w(f) = \sum_{t(l)\in\mathcal{S}} f(l) - \sum_{r(l)\in\mathcal{S}} f(l) = \sum_{r(l)\in\mathcal{T}} f(l) - \sum_{t(l)\in\mathcal{T}} f(l). \tag{2.3}$$

A cut Q of $\mathcal{N}$ is a partition $\mathcal{V} = Q_S + Q_T$ where the plus sign denotes the union of two disjoint sets ($\mathcal{V} = Q_S \cup Q_T$, $Q_S \cap Q_T = \emptyset$) such that $\mathcal{S} \subset Q_S$ and $\mathcal{T} \subset Q_T$. The capacity of the cut is

$$c(Q) = \sum_{t(l)\in Q_S,\, r(l)\in Q_T} c(l). \tag{2.4}$$

Let $\mathcal{T}_v$ be the set of all nodes u such that $(v, u) \in \mathcal{L}$. Similarly, let $\mathcal{R}_v$ be the set of all nodes u such that $(u, v) \in \mathcal{L}$. That is, node v can receive packets from $\mathcal{R}_v$ and transmit packets to $\mathcal{T}_v$. Together they form the neighborhood $\mathcal{U}_v = \mathcal{R}_v \cup \mathcal{T}_v$ of node v. Throughout this thesis, we study cases where $\mathcal{U}_v = \mathcal{R}_v = \mathcal{T}_v$. The number of neighbors, $|\mathcal{U}_v|$, of node v is the same as the number of edges incident to v in the simple underlying graph of the network. A simple underlying graph can be obtained from $\mathcal{G}$ by replacing all directed edges with undirected edges and by removing all loops and multiple edges. In graph theory, the number of edges incident to a node is called the node degree. We use the terms node degree and number of neighbors interchangeably. The mean degree of a node is denoted by ν.

In a wireless network, not all links can be active simultaneously because of interference, and thus, the effective link capacity $c(l)$ of a link is less than the nominal capacity C. The capacity is measured in packets

per second, unless otherwise stated. To define $c(l)$, we have to establish a schedule $\boldsymbol{\alpha}$ which tells us how the links are used. All the links that are active simultaneously have to belong to the same independent set of links to avoid collisions. A set of links is said to be independent if the corresponding vertices form an independent set (no two are adjacent) in $\mathcal{I}$. This is equivalent with forming a clique in the complement of $\mathcal{I}$. A complement of a graph has the same vertices as the original graph, and two vertices are adjacent if and only if they are not adjacent in the original graph. The independent sets of links are cliques in the complement interference graph. A clique, $\mathcal{C}$, is a set of vertices such that for every pair of vertices there is an edge connecting them.

We call the sets of links that are used for transmitting simultaneously *transmission modes* (these are independent sets unless otherwise mentioned) and denote the set of transmission modes with $\mathcal{M} = \{m_1, \ldots, m_M\}$. A transmission mode is called maximal if the corresponding independent set in the interference graph of the network is maximal (i.e., not a subset of any other independent set). The schedule $\boldsymbol{\alpha} = \{t_1, \ldots, t_M\}$ assigns each transmission mode m_i with the proportion of time t_i that it is used. Now the effective capacity of link l is

$$c(l) = C \sum_{i=1}^{M} t_i \mathbb{1}_{\{l \in m_i\}}, \tag{2.5}$$

that is, the nominal capacity multiplied by the time share the link is active. Often the time shares are multiples of some t_0. We assume (mainly for simplicity of the presentation) that the time is slotted, and the length of a time slot, t_0, is the time required to send one packet.

The term random geometric graph refers to a graph where the nodes have some random geographic locations, and two nodes are connected if the distance between them does not exceed a threshold called transmission range. We denote the location of node v by the position vector $\boldsymbol{v}$. The following sections present models for locating the nodes and defining the transmission range and the rules of interference.

2.2 Node deployment

The geographical locations of the network devices are an important factor affecting the performance of a wireless multihop network. If the locations of the nodes can be chosen arbitrarily, a reasonable selection is usually a regular grid of some sort. The possibility to choose the node pattern gives

a possibility to calculate exact values for different performance metrics, but we deem it outside the scope of this thesis.

In a more typical scenario, we do not have full control on the locations of the nodes and have to treat them as random. In random networks, the nodes are randomly distributed, and the node locations are described by a point process [IPSS08] in time and (two- or three-dimensional Euclidean) space. We operate in two dimensions, i.e., on plane $\mathbb{R}^2$, and assume that the nodes do not move. A point process can be seen as a collection of points, $\xi = \{\boldsymbol{x}_i\} \subset \mathbb{R}^2$, or a counting measure, where $\xi(\mathcal{A})$ gives the number of points in $\mathcal{A} \subset \mathbb{R}^2$. A set of very general assumptions leads to the following popular point process for the node locations.

2.2.1 Poisson point process

Assume that we know that there are, on average, n nodes per unit area, but we do not know anything about their locations. If we have no information about the locations of the nodes, we should, a priori, assume that they are completely random, that is, uniformly distributed. Now we have points uniformly distributed over the whole plane with density n. A couple of interesting observations can be made:

1. The number of points in a bounded set $\mathcal{A}$ has a Poisson distribution with mean $n|\mathcal{A}|$.

2. The number of points in disjoint sets are independent.

These two properties characterize the Homogeneous Poisson point process.

A (general) Poisson point process (PPP) can be defined as follows. Given a locally finite non-null measure μ, a Poisson point process ξ of intensity measure μ is given by its distribution

$$\mathbb{P}(\xi(\mathcal{A}_1) = N_1, \ldots, \xi(\mathcal{A}_k) = N_k) = \prod_{i=1}^{k} e^{-\mu(\mathcal{A}_i)} \frac{\mu(\mathcal{A}_i)^{N_i}}{N_i!},$$

for every integer $k \geq 1$ and all disjoint bounded sets $\mathcal{A}_i$, $i = 1, \ldots, k$. If $\mu(\mathcal{A}) = n|\mathcal{A}|$, i.e., μ is a multiple of a Lebesgue measure, ξ is the above-mentioned homogeneous PPP with intensity n. The definition is also equal to the following two properties similar to the homogeneous equivalents

1. $\xi(\mathcal{A}) \sim \text{Poisson}(\mu(\mathcal{A}))$ for bounded set $\mathcal{A}$.

2. $\xi(\mathcal{A}_i)$ are independent for disjoint bounded sets $\mathcal{A}_i$.

Generating realizations

Later in this work, we use the homogeneous PPP to model the random locations of the nodes and the following ways to generate realizations. From the previous it is quite clear that a finite realization of the process can be generated by uniformly distributing a Poisson distributed number of points on an area. Another possibility in a rectangular network is evident after noting that the x-coordinates of the nodes are events of a one-dimensional Poisson process in a fixed interval. Consequently, the distance between consecutive nodes has an exponential distribution with mean $1/nH$, where H is the height of the network. Hence, the difference in two successive x-coordinates is exponentially distributed, and the y-coordinate of each point is uniformly distributed.

2.2.2 Other models

We limit our studies to the PPP but present here a couple of alternatives that could also be used and appear often in conjunction with wireless networks. Hard-core models are a class of point processes whose points are separated at least by some given distance $d > 0$. The points can be seen as the centers of hard disks with radius $d/2$. For the PPP there exists no $d > 0$ such that the hard-core property for d is satisfied. Hard-core processes generate relatively regular point patterns with smaller spatial variability compared with the PPP.

The Matérn hard-core process of type I [Mat86] can be constructed from an underlying PPP by retaining all points whose distance to their nearest neighbor is greater than a given d. This means that if two points of the PPP are close to each other, they are both removed.

Assume that the points of a PPP are marked with uniform random numbers distributed over [0,1]. A Matérn hard-core process of type II [Mat86] can be constructed from the PPP by retaining the points with the smallest mark within the hard-core distance d. In more detail, a marked point process can be represented as a collection of pairs $\tilde{\xi} = \{(\boldsymbol{x}_i, m_i)\}_i$, where $\xi = \{\boldsymbol{x}_i\}$ is the set of points and $\{m_i\}$ the set of marks. Using an independently marked PPP $\tilde{\xi} = \{(\boldsymbol{x}_i, U_i)\}_i$ with marks $\{U_i\}$ uniformly distributed

on $[0, 1]$, we have

$$\xi_{\mathrm{MHC}} = \{\boldsymbol{x}_i \in \xi \,|\, U_i < U_j \,\forall\, \boldsymbol{x}_j \in B_{\boldsymbol{x}_i}(d) \setminus \{\boldsymbol{x}_i\}\},$$

where $B_{\boldsymbol{x}_i}(d)$ is a disk of radius d centered at $\boldsymbol{x}_i$. A point is removed if there exists another point in the underlying marked PPP within the hard-core distance with a smaller mark.

Cluster point processes have higher spatial variability compared with the PPP as the points interact by attracting other points. A Matérn cluster process [Mat86] can be constructed in the following fashion. The cluster heads form a PPP with a given intensity n_0, and the actual points are within a given distance $d > 0$ from these, as inside these discs PPPs with intensity n_1 are generated.

Furthermore, it is always possible to combine hard-core and clustered models appropriately or define more general point processes. It is possible, for example, to define soft-core models where points repel each other, but there is no hard limit for minimal distance. For more information on point processes, see, e.g., [IPSS08].

2.3 Interference models

A wireless sensor network may consist of hundreds of terminals spread over a large area, all using the same channel. Hence, spatial reuse is a key feature of wireless multihop networks that determines how the scarce resource, the channel, can be utilized. The significance of the interference model selection cannot be overstated.

More formally, to apply graph theory, we need a definition for when two nodes form a link and when and how two links interfere. In the following, we assume that the nodes are homogeneous, and they have only one radio. This assumption means that a node can participate in only one transmission at a time (either as a receiver or as a transmitter), and that all the links are symmetrical ($(u, v) \in \mathcal{L} \Leftrightarrow (v, u) \in \mathcal{L}$).

2.3.1 Boolean interference model

Under the (unidirectional) Boolean interference model, there exists a link $l = (u, v) \in \mathcal{L}$, $u, v \in \mathcal{V}$ if the Euclidean distance between the nodes, $|\boldsymbol{u} - \boldsymbol{v}|$, is less than the common transmission range ρ. A node is able to receive a transmission if it is inside the transmission radius of only one active

transmitter. A transmission on link l is, thus, successful if

$$|\boldsymbol{r}_l - \boldsymbol{t}_l| \leq \rho \quad \wedge \quad |\boldsymbol{r}_l - \boldsymbol{t}_{l'}| > \rho \quad \forall\, l' \in m \setminus \{l\}, \tag{2.6}$$

where $\boldsymbol{t}_l$ is the location of the transmitting node of link $l \in \mathcal{L}$, $\boldsymbol{r}_l$ is the location of the receiving node, and m is the set of transmitting links.

2.3.2 SINR threshold model

We assume that all the nodes transmit with constant power P_0 and the attenuation factor between the transmitter of link l and the receiver of link l' follows the simple power law

$$g(l, l') = \left(\frac{|\boldsymbol{t}_l - \boldsymbol{r}_{l'}|}{\rho_0} \right)^{-\alpha}, \tag{2.7}$$

where ρ_0 is a freely choosable reference distance, and α is a given attenuation coefficient.

Under the SINR threshold model, there exists a link $l = (u, v) \in \mathcal{L},\ u, v \in \mathcal{V}$ if the signal-to-interference-and-noise ratio (SINR) is greater than or equal to a given threshold, θ. The SINR at $\boldsymbol{r}_l$ for link l in transmission mode m is

$$\mathrm{SINR}(\boldsymbol{r}_l, m) = \frac{P_0 \cdot g(l, l)}{\sum\limits_{l' \in m \setminus \{l\}} P_0 \cdot g(l', l) + \sigma^2}, \tag{2.8}$$

where σ^2 is the thermal noise power. The spectral efficiency of links with

$$\mathrm{SINR}(\boldsymbol{r}_l, m) \geq \theta \tag{2.9}$$

is assumed constant irrespective of the SINR,

$$R_l(m) = R(\theta) = \log_2(1 + \theta), \tag{2.10}$$

i.e., the spectral efficiency at the threshold according to Shannon's formula. The spectral efficiency of other links is assumed to be zero. The interference model is also referred to as the physical model [GK00].

As the attenuation coefficient α and the threshold θ (and possibly also the thermal noise power, σ^2) are given constants, we can define a length unit

$$\rho(P_0) = \rho_0 \sqrt[\alpha]{P_0/(\theta\sigma^2)}, \tag{2.11}$$

that is, the maximum distance at which a reception is possible if there are no competing transmissions (zero interference).

Under the SINR threshold model, the interaction between the links can no longer be described using an interference graph. As the interference

is additive, a pairwise comparison between two links is not enough to exclusively determine if the links can be activated at the same time. The term independent set of links can still be used for a set of links that can be successfully activated simultaneously.

The difference between the SINR threshold model and the Boolean interference model in an example scenario with three active transmitters is illustrated in Figure 2.1. The areas where a reception is possible from each of the transmitters are drawn in the figure. Under the simple Boolean interference model, a node is able to receive a transmission if it is inside the (fixed) transmission radius of only one active node. The main difference between the SINR interference model and the Boolean interference model is that, although the border of the reception area approaches the one of the Boolean interference model when α grows, the SINR interference model is more realistic in always allowing a reception near the transmitter.

2.3.3 Shannon model

In this model, we still assume that all the nodes transmit with constant power P_0, and that the attenuation factor follows power law (2.7). Now, any set of links that do not share an endpoint can be activated simultaneously. The SINR is calculated using (2.8), but the spectral efficiency of link l in transmission mode m, is calculated according to Shannon's formula

$$R_l(m) = \log_2(1 + \mathrm{SINR}(\boldsymbol{r}_l, m)). \tag{2.12}$$

As the attenuation coefficient, α, and (possibly) the thermal noise power, σ^2, are given constants, we can define a length unit

$$\rho(P_0) = \rho_0 \sqrt[\alpha]{P_0/\sigma^2}, \tag{2.13}$$

that is, the distance at which the received signal power equals the noise power.

In contrast to the previous two interference models, under the Shannon model, the capacity of a link is not constant. A node can transmit to any other node in the network, but the spectral efficiency of the link depends on the SINR. Similarly as with the SINR threshold model, the additive interference prevents the use of an interference graph as an analyzing tool. The fact that the interference model cannot be properly modeled using a graph makes it difficult to examine.

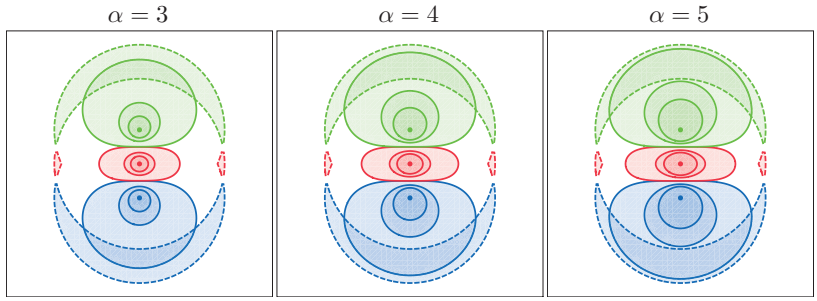


Figure 2.1. The areas where a reception is possible for three transmitters (dots) under the Boolean interference model (dashed lines) and the SINR threshold model with threshold 1 (outermost solid lines). The inner solid lines are contours for SINR values 7 and 31.

2.3.4 Other interference models

We begin our review of other models with an interference model as simple as possible. Under the primary interference model, a transmission over a link is successful if none of the links sharing a node with the transmitting link is active during the transmission. In wireless systems this basically means that there are multiple channels available (e.g., FDMA or CDMA), but each node still has only one transceiver. The interference model is also known as the node exclusive interference model.

Secondary interference models take into account the interference coming from other transmissions, as is necessary when the communicating nodes share a single channel. The Boolean interference model and the SINR based interference models represent the two extremes of secondary interference models. The Boolean interference model is one of the simplest models while additive interference models attempt to capture the complex interaction between the nodes.

The next step from the Boolean interference model towards more complex models is to separate the interference range from the transmission range. In practice, the interference range is typically larger than the transmission range. Under this interference model a transmission is successful if

$$|\boldsymbol{r}_l - \boldsymbol{t}_l| \leq \rho \quad \wedge \quad |\boldsymbol{r}_l - \boldsymbol{t}_{l'}| > (1+\Delta)\rho \quad \forall\, l' \in m \setminus \{l\},$$

where m is the set of transmitting links and $\Delta > 0$.

A step further is to make the interference range dependent on the length of the transmitting link. The shorter the link, the closer the interferers

are allowed to be, and we have:

$$|\boldsymbol{r}_l - \boldsymbol{t}_l| \leq \rho \quad \wedge \quad |\boldsymbol{r}_l - \boldsymbol{t}_{l'}| > (1+\Delta)|\boldsymbol{r}_l - \boldsymbol{t}_l| \quad \forall\, l' \in m \setminus \{l\}.$$

This interference model is referred to as the protocol model [GK00]. The protocol model can be written as

$$P_0 \left(\frac{|\boldsymbol{r}_l - \boldsymbol{t}_l|}{\rho_0} \right)^{-\alpha} \geq P_0 \left(\frac{\rho}{\rho_0} \right)^{-\alpha} \quad \wedge$$
$$\frac{P_0(|\boldsymbol{r}_l - \boldsymbol{t}_l|/\rho_0)^{-\alpha}}{P_0(|\boldsymbol{r}_l - \boldsymbol{t}_{l'}|/\rho_0)^{-\alpha}} > (1+\Delta)^{\alpha} \quad \forall\, l' \in m \setminus \{l\}.$$

Using (2.7) this can be simplified to be

$$P_0 \cdot g(l,l) \geq \theta_r \quad \wedge \quad \frac{P_0 \cdot g(l,l)}{P_0 \cdot g(l',l)} > \theta_c \quad \forall\, l' \in m \setminus \{l\},$$

where $\theta_r = P_0(\rho/\rho_0)^{-\alpha}$ and $\theta_c = (1+\Delta)^{\alpha}$. Hence, if all the nodes use the same transmission power and the path loss function is g (2.7), the protocol model requires that the received signal is above a certain threshold, and the signal-to-interference ratio is above another threshold for each interferer individually.

The final steps towards the SINR threshold model are thus:

- Additive interference from all the interfering transmitters that requires that the signal-to-interference ratio is above the threshold for the sum interference.

- Thermal noise power that sets the level for the required received signal in the absence of interference and combines the two constraints (required received signal power and signal-to-interference ratio) into one expression (2.9).

Between the SINR threshold model with a fixed spectral efficiency and the Shannon model a with continuous spectral efficiency lie the graded SINR interference models (see, e.g., [SMR+09]), where different SINRs yield different spectral efficiencies according to multiple theresholds. Further information on different interference models can be found in [Car10] and [IRK09], or from an experimental point of view in [MJD08].

2.3.5 Additional aspects

Power control

In this thesis, we touch the topic of power control under the Boolean interference model. We assume that the nodes can simply use an *adjustable*

transmission radius just large enough to reach the receiving node and thus minimize the interference.

Bidirectional interference

We also simply model the effect of RTS/CTS handshaking in 802.11 networks, during which the transmitter and receiver both need to be able to hear each other, using a bidirectional interference model. Under the *bidirectional Boolean interference model*, a successful transmission requires that both the sender and the receiver are outside the range of competing transmissions:

$$|\boldsymbol{r}_l - \boldsymbol{t}_l| \leq \rho \quad \wedge \quad |\boldsymbol{r}_l - \boldsymbol{t}_{l'}| > \rho \quad \wedge \quad |\boldsymbol{t}_l - \boldsymbol{t}_{l'}| > \rho \quad \forall\, l' \in m \setminus \{l\}. \quad (2.14)$$

3. Network capacity

In this chapter, we get to the main topic of the thesis, that is, the forwarding capacity of a large wireless multihop network. The capacity of a wireless network is fundamentally affected by interference, which determines whether two nodes are able to form a link in the first place and, further, if two such links can be activated simultaneously. When the tricky behavior of the shared wireless channel is combined with irregular placement of the nodes, it is certain that the question at hand is not an easy one, especially so if we are dealing with multihop traffic. With that in mind, we move from classical, and finite, flow networks towards large-scale networks.

Our goal is to introduce the concept of forwarding capacity, the main topic of this thesis. As the size of the network grows towards infinity, the effect of local bottlenecks, caused by the individual locations of the randomly placed nodes, begins to average out. This simplifies the problem. The network can be seen as a medium with a certain capability to carry information either in a single direction or in multiple directions simultaneously. The forwarding capacity describes this characteristic. After having defined the forwarding capacity, the scope of the thesis is explained in detail.

Along the way we cover the methodology that is later used to simulate relay traffic on an infinite plane to determine the sought-after forwarding capacity. We also see how our quantity of interest, the forwarding capacity, is connected to the overall capacity of a large wireless network and review the state-of-the-art research on the related areas. But first, we briefly cover the topic of connectivity, as without connectivity there is no capacity.

3.1 Connectivity

We begin with the most fundamental question of wireless networking, namely, the one of connectivity. Connectivity creates the basis for positive capacity in the first place.

From an information theoretic point of view, a wireless network is always fully connected, and the achievable rate depends on the SINR (2.8). This is the case only as long as we use the Shannon model (see Section 2.3.3) for interference. As soon as we set a minimum value for the SINR (and there is some thermal noise), we have a transmission range, ρ, that sets a limit for the transmission distance. If the node locations follow a homogeneous PPP with density n, the probability that a node is isolated is $e^{-\pi n \rho^2}$. To keep the network asymptotically connected, the expected node degree should grow faster than the logarithm of the number of nodes [GK98]. For results on asymptotic connectivity, see also [Bol85, WY04, XK04, Far05].

Nodes can communicate through some multihop route if and only if they belong to the same connected component. The best one can usually hope for is that a node can communicate with an infinite number of nodes. The infinite wireless multihop network is said to be connected if this is the case, that is, if the network percolates [Gil61].

3.1.1 Percolation

The long range connectivity of the network is related to the existence of an infinite connected cluster — the so-called giant component. Let $\mathcal{G} = (\mathcal{V}, \mathcal{L})$ be an (undirected) graph and $K(v)$ be the set of vertices that are accessible from $v \in \mathcal{V}$. We denote the probability that an arbitrary node belongs to a cluster of infinite size, $|K(v_0)| = \infty$, by $\eta(\boldsymbol{p})$, where $\boldsymbol{p}$ is a parameter vector, and call it the percolation probability. The percolation threshold is the critical surface for the parameters of the problem, $\boldsymbol{p}$, such that the percolation probability first becomes strictly positive. For example, if we have just a single parameter, p, this becomes $p_c = \sup\{p \mid \eta(p) = 0\}$. If $\eta(\boldsymbol{p}) > 0$, it now follows from Kolmogorov's zero-one law that there almost surely is some infinite cluster.

The following theorem ([DTH02], proofs in [MR96]) states that there exists a finite, positive value n_c for the node density in a network model where the nodes are located according to a 2-dimensional Poisson process

with intensity n and have a fixed transmission radius of ρ, under which the percolation probability is zero and above which it is strictly positive:

Consider a Poisson Boolean model $\mathcal{B}(n, \rho)$ in $\mathbb{R}^2$. There exists a critical density $n_c > 0$ such that when $n < n_c$, all clusters are bounded a.s., and when $n > n_c$, there exists a unique unbounded cluster a.s.

Instead of n, the transmission range ρ can also be varied since the models $\mathcal{B}(\gamma n, \rho)$ and $\mathcal{B}(n, \rho\sqrt{\gamma})$, where γ is a constant, are associated with identical graphs.

The exact value of the critical density is not known. Some analytical bounds have been found, see, e.g., [Gil61, PPT89, MR96], in addition to numerical bounds [BBW05] and estimates [QTZ00, QZ07]. For example [QZ07] gives the estimate $\phi_c = 0.6763475 \pm 0.0000006$ for the critical volume fraction[1], $\phi = 1 - e^{-\pi n(\rho/2)^2}$, from which we get $\nu = \pi n \rho^2 = -4\ln(1-\phi)$ for the mean node degree (mean number of neighbors) and $\nu_c \approx 4.512$ for the percolation threshold.

The SINR threshold model leads to a similar connectivity graph as the Boolean models, but because of the additive interference, the links may only be available (and the above critical value holds) if a single node transmits at a time. In case of multiple simultaneous transmissions, as is the usual case, a signal to interference ratio graph should be used to analyze percolation [DT04, DBT05, DFM$^+$06, PMS09].

The critical value for the mean number of neighbors is noteworthy when we later present the results for the forwarding capacity, since the true performance below the percolation threshold would always be zero. The infinite cluster guaranteeing the long range connectivity exists almost surely only in the super-critical phase ($n > n_c$).

In what follows, we assume that the flow networks are connected (at least the sources and the sinks). When studying infinite networks, we operate above the percolation threshold. In this case, some of the nodes are not connected to any other node, and there are clusters that are separate from the giant component. We call this kind of network the transport network. It allows the transportation of information around the network, but to access the network or to deliver the information, the nodes might need a larger transmission radius. This can be realized in a separate access phase that does not hinder the operation of the transport network. We return to the question in Section 3.4.

[1]Here the transmission range, ρ, is replaced with a disk of radius $\rho/2$, and communication corresponds to overlapping disks.

3.2 Network capacity problem

To begin the consideration of determining the capacity of a network, we introduce the most common form of a network capacity problem. It is well known that for a given finite network, the capacity maximization can be expressed as a linear programming (LP) problem. The maximum flow problem is a classic problem in graph theory and combinatorial optimization with a variety of applications. It considers finding a feasible flow through a flow network $\mathcal{N} = (\mathcal{G}, c, \mathcal{S}, \mathcal{T})$ that is maximal. The definitions for graphs are presented in Section 2.1. A flow f is maximal if the value of the flow, $w(f)$, is greater than or equal to $w(f')$ for all flows f' on $\mathcal{N}$. Next, we formulate the maximum flow problem, and then we discuss the computational aspects of the problem.

3.2.1 Maximum flow problem

The problem is to maximize the value of the flow (2.3) subject to the flow constraints (2.1-2.2) and a constraint that says that the sum of the time shares allocated to the transmission modes cannot exceed one. Hence, the decision variables are the flow, f, and the schedule, $\boldsymbol{\alpha} = \{t_1, \ldots, t_M\}$, that specifies the capacities. The complete LP formulation can be expressed as

$$\begin{aligned} &\max_{f,\boldsymbol{\alpha}} \; w(f) \quad \text{s.t.} \qquad (3.1)\\ &0 \le f(l) \le C \sum_{i=1}^{M} t_i \mathbb{1}_{\{l \in m_i\}} \quad \forall\, l \in \mathcal{L},\\ &\sum_{r(l)=v} f(l) - \sum_{t(l)=v} f(l) = 0 \quad \forall\, v \in \mathcal{V} \setminus (\mathcal{S} \cup \mathcal{T}),\\ &\sum_{i=1}^{M} t_i \le 1,\; t_i \ge 0, \quad i = 1, \ldots, M. \end{aligned}$$

The previous constraints are the necessary and sufficient conditions for the flow optimality.

A fundamental result in flow theory considers the duality of this problem. It turns out that the minimum capacity of a cut (2.4) has a significant effect on the capacity of the network. A cut $Q = (Q_S, Q_T) \in \mathcal{Q}$ is called a minimum cut if $c(Q) \le c(Q')$ for all cuts Q' of the network. The following max-flow min-cut theorem [FF56] states that:

The maximum value of a flow on a flow network $\mathcal{N}$ equals the capacity of a minimum cut in $\mathcal{N}$.

This basically means that the bottlenecks of the network dictate the amount of traffic the network can carry.

With a fixed schedule the maximal flow in the wireless flow network equals the capacity of the minimal cut, but to find the overall maximum value for the flow, we also have to optimize the schedule. The value of the optimal flow ensues from the problem

$$\max_{\boldsymbol{\alpha}} \min_{Q \in \mathcal{Q}} c(Q, \boldsymbol{\alpha}), \tag{3.2}$$

where $c(Q, \boldsymbol{\alpha})$ is the capacity of cut Q with schedule $\boldsymbol{\alpha}$, and $\mathcal{Q}$ is the set of all cuts.

Our formulation for the problem allows multiple sources and multiple sinks, and so far we have not distinguished among the flow units. That is, we are not interested in which source generates flow to which sinks. This formulation is called a single-commodity flow problem, and it is used later when we study relay traffic whose exact origin and destination are irrelevant. There is also a class of network flow problems called multi-commodity flow problems in which it is necessary to distinguish among the flows in the network. In this case, we have a set of pairs of vertices where each pair defines a commodity and contains a source node and a sink node. Each commodity must satisfy flow conservation (2.2) at each vertex other than its own source and sink. Also, the sum of flows routed through an edge should not exceed its capacity, cf. (2.1). One can also allow a commodity to have more than one source and/or sink. With the term *traffic class*, we refer to a commodity that might have multiple sources and sinks.

The inherent computational complexity of the LP problem can be eased by relaxing some of the necessary constraints resulting in upper/lower bounds. Methods utilizing this idea are considered next.

Column generation

If only a subset of the transmission modes is used, we get a lower bound. In column generation [LD05], the idea is to include only transmission modes that are part of the optimal solution. A new transmission mode is added at each iteration of the algorithm, but only if it improves the result. The transmission mode added to the master problem is solved from another optimization problem called the sub-problem. In [KWE08], column generation is utilized for determining the minimum-length schedule that satisfies certain traffic demands.

Clique formulation

From the constraint that says that the sum of the time shares of the transmission modes is less than or equal to one, we get that the total capacity of a set of links all interfering with each other never exceeds the nominal capacity C. This follows simply from the fact that two interfering links cannot be used at the same time. A constraint that limits the capacity of a maximal clique of the interference graph to C is, thus, a necessary condition for a feasible flow. By replacing the constraint considering the time shares with constraints for the clique capacities, we get an upper bound for the maximum value of a flow in the network. Since the flow through a link is bounded by the capacity, these constraints can further be replaced by similar ones considering the flow. When the flow satisfies these, the capacities can always be chosen to match the flow. Thus, the only decision variable is the flow, and we have

$$\begin{aligned} \max_f \; & w(f) \quad \text{s.t.} \\ & f(l) \geq 0 \quad \forall\, l \in \mathcal{L}, \\ \sum_{r(l)=v} & f(l) - \sum_{t(l)=v} f(l) = 0 \quad \forall\, v \in \mathcal{V} \setminus (\mathcal{S} \cup \mathcal{T}), \\ & \sum_{l \in \mathcal{C}_i} f(l) \leq C \quad \forall \text{ cliques } \mathcal{C}_i \text{ of the } \mathcal{I}. \end{aligned} \tag{3.3}$$

The problem of finding a maximum clique is NP-hard [GJ79]. It is computationally equivalent to finding a maximum independent set through the concept of complement graph. But while the number of transmission modes increases rapidly, when the diameter of the network grows compared to the interference range, the growth in the number of cliques is much more modest. A clique of interfering links is always local; a transmission mode, on the other hand, may contain links from all over the network. This sometimes makes the clique formulation tempting although it only gives an upper bound.

Limited set of cuts

Also the max-flow min-cut theorem can be used for obtaining upper bounds. Suppose we now have a wireless network, and we study it under a fixed schedule α. Let $\mathcal{N}$ be the resulting flow network. The maximum value of a flow on $\mathcal{N}$ is equal to the minimum capacity of a cut in $\mathcal{N}$. The number of cuts in the network is $2^{|\mathcal{V}|-|\mathcal{S}|-|\mathcal{T}|}$, which makes finding the minimal cut an overwhelming task even for relatively small values of $|\mathcal{V}|$. To find the

overall maximum flow, we would still have to maximize the capacity of the minimal cut with respect to the schedule.

We can get an upper bound for the performance by limiting our examinations to a smaller set of cuts $\mathcal{Q}'$, because the minimum of a subset is always greater or equal to the original minimum. This gives us the constraint

$$\max_{\boldsymbol{\alpha}} \min_{Q \in \mathcal{Q}} c(Q, \boldsymbol{\alpha}) \leq \max_{\boldsymbol{\alpha}} \min_{Q \in \mathcal{Q}' \subset \mathcal{Q}} c(Q, \boldsymbol{\alpha}) \tag{3.4}$$

for the maximum value of the flow. Another upper bound can be obtained by switching the order of minimization and maximization. Since the capacity of a cut with the optimal schedule, $c(Q, \boldsymbol{\alpha}^*)$, is always less or equal to the maximum capacity of the cut, $\max_{\boldsymbol{\alpha}} c(Q, \boldsymbol{\alpha})$, we have

$$\max_{\boldsymbol{\alpha}} \min_{Q \in \mathcal{Q}} c(Q, \boldsymbol{\alpha}) \leq \min_{Q \in \mathcal{Q}} \max_{\boldsymbol{\alpha}} c(Q, \boldsymbol{\alpha}). \tag{3.5}$$

The maximum capacity of a cut actually equals the size of the maximum independent set of links crossing the cut multiplied by C, because the maximum can be achieved by selecting as many independent links as possible and using them the whole time. This is always more than the capacity of the cut with the optimal schedule (for the whole network) since the links cannot be used continuously in order for the flow network to be connected under that schedule (unless the link connects a source and a sink directly). Thus, the value of the maximum flow is limited by the size of the smallest maximum independent set of links crossing a cut in $\mathcal{Q}$.

3.3 Achieving maximum capacity

Although the maximum capacity of a large wireless network is not known, there are methods that are able to achieve this capacity. These methods to be discussed shortly can even be distributed. However, they tend to be too slow to be simulated in a large network. This is unfortunate because if it were possible to run the algorithms in a large enough network, they would answer our question and reveal the sought-after forwarding capacity.

This section covers methods that are able to achieve the maximum capacity in a wireless multihop network. They are divided into two groups: maximum weight scheduling (MWS) algorithms and distributed random access algorithms. In Chapter 5, we use a greedy version of MWS to estimate the capacity. The distributed random access methods are related to some of the topics of Chapter 4.

3.3.1 Maximum weight scheduling

In [TE92], the authors present a maximum throughput policy that stabilizes the network for all arrival rates of multiclass traffic for which it is stabilizable. The fundamentals of the used network model are as follows. New packets appear at the nodes of the network with given rates and join the queue. The packets are divided into classes based on their set of destination nodes. Once the packets reach any of their destination nodes they leave the network. As the network is also stable for the arrival rates that realize the maximum capacity, the policy achieves maximum capacity.

The algorithm has three stages. In time slot t, the first stage is to calculate a weight w_l^t for each link $l \in \mathcal{L}$ as follows,

$$w_l^t = \max_j \left(q_j^{t-1}(t(l)) - q_j^{t-1}(r(l)) \right), \tag{3.6}$$

where j is the class index and $q_j^t(u)$ is the queue length (number of packets) of class j at node u at time t. In the second stage, a maximum weight transmission mode is selected

$$m^*(t) = \arg\max_{m \in \mathcal{M}} \sum_{l \in m} w_l^t. \tag{3.7}$$

Finally, in the third and last stage, if we index the links with $i = 1, \ldots, |\mathcal{L}|$, and denote by j^* the class for which $w_l^t = q_{j^*}^{t-1}(t(l)) - q_{j^*}^{t-1}(r(l))$, we get the information about the activated links in binary form, $E(t)$, at time slot t as follows,

$$E_{ij}(t) = \begin{cases} 1, & \text{if } l_i \in m^*(t),\ j = j^*, \text{ and } q_j^{t-1}(t(l_i)) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{3.8}$$

In each time slot the policy finds the transmission mode that is of the maximum weight. The weight of each link is the maximum (over the classes) difference in the queue lengths between the transmitting and receiving end of the link multiplied by the capacity of the link. A link in the maximum weight transmission mode is activated if the transmitter has enough packets to send.

Unfortunately, finding the maximum weight transmission mode is NP-complete and requires global information. Thus, such algorithms are often too difficult to implement when the problem is large. Randomized versions of MWS [Tas98] provide a linear complexity implementation of the algorithm.

A randomized maximum weight scheduling algorithm can be constructed simply as follows [Tas98]. A transmission mode candidate is selected from

a certain distribution, p, at random. If the weight of the candidate is higher than the weight of the current transmission mode, it is scheduled at the next time slot. Otherwise the current transmission mode is scheduled to be used again. As long as the probability of selecting the maximum weight transmission mode is positive in distribution p, the randomized iterative algorithm achieves maximum throughput.

The first distributed scheduling framework that guarantees maximum throughput [MSZ06] is an implementation of this algorithm that generates feasible schedules and compares and merges them in a distributed manner. Despite the fact that the algorithm may sometimes switch to a worse transmission mode, it has been shown to achieve the optimal throughput.

The algorithm [MSZ06] is presented for single hop traffic under the primary interference model, but it is extendable to more general models. Although, transmission modes can be compared and merged in a distributed manner by collecting information from a few hops away, the algorithm requires a notable amount of information exchange for each scheduling decision.

Under some interference models, it is possible to guarantee a fraction of the throughput region using greedy maximum weight scheduling, see [CKLS08]. For example, for the primary interference model this fraction is $1/2$, but for the Boolean interference model no such fraction exists.

3.3.2 Distributed random access

Since the introduction of the classical Aloha protocol [Abr70] and its slotted version [Rob75], random access algorithms have been a recurring topic in wireless network research because of their simplicity and distributed nature. The fact that each user decides independently when to attempt to access the channel may cause a lot of collisions and thus deteriorate the capacity of the system. A class of random access algorithms tries to avoid this using carrier sense multiple access (CSMA) [KT75]. The users willing to transmit must first listen to the channel, and if the channel is not busy, they can choose to transmit.

The performance of such systems has in many respects remained an open question. An important result was the characterization of the capacity region of a multi-access network with given access probabilities [BMP08]. The result is exact when the number of users grows to infinity but also very accurate in the case of small systems.

In [MEO11], the authors analyze asynchronous CSMA policies for scheduling packet transmissions in wireless multihop networks under the primary interference model. They present a fixed point approximation for given routes and arrival rates that is asymptotically accurate for large networks with small sensing delay. This means that the maximum throughput can be achieved under the assumption that the network traffic consists of many small flows. Simulations also show that the approximation is accurate for moderately sized networks.

A recent paper [RSS09] proposes a random access CSMA algorithm in which the access probability of a node is a function of its own queue length and the estimate of the maximum queue length in the network. The maximum queue length in the network is a global variable but an estimation of it can be maintained if each node broadcasts exactly one number to all of its neighbors in every time slot.

The algorithm is throughput optimal assuming that collisions are eliminated. This requires that the feedback from the carrier sensing mechanism telling whether the link can be activated in the current transmission mode is instantaneous.

The paper considers single hop flows, but the authors strongly believe the algorithm to be expandable to the multihop setting. The authors also conjecture that, for the algorithm to be throughput optimal, it is enough if each node chooses its access probability as a slowly varying function of its own queue length. By ignoring the estimate of the maximum queue length, the algorithm would be totally distributed.

A similar distributed adaptive CSMA algorithm was proposed in [JW10]. Instead of instantaneous queue lengths, it uses empirical arrival and service rates. The algorithm is totally distributed and achieves throughput optimality assuming the idealized CSMA (no collisions). The paper combines the CSMA scheduling algorithm with congestion control to satisfy a given fairness criterion among competing (multihop) flows. The effect of collisions is also discussed.

Both of the algorithms can be seen as continuous-time reversible Markov chains that operate on the space of transmission modes with adaptive transition probabilities. The algorithms adjust the aggressiveness (based on a slowly varying function of instantaneous queue lengths [RSS09] or empirical arrival and service rates [JW10]) at which inactive links attempt a transmission when they sense the channel free. The process re-

sembles simulated annealing (see Section 4.4.2), which can be used, for example, to find the maximum weight transmission mode needed in MWS.

Through changing the CSMA parameters, the algorithms adapt to the current traffic load of the system. At the same time, they also change the stationary distribution of the above mentioned Markov chain. The important observation is that, if the adaptation is slow enough, the Markov chain remains always close to its stationary distribution, and the algorithms effectively simulate MWS.

3.4 Scaling laws

So far we have covered topics that are usually presented in connection with networks of very limited size. Now, we start moving towards the kind of networks that we are mostly interested in this thesis, the very large ones. When large networks are studied, an arising question of interest is how the per-node capacity of a wireless multihop network scales as the number of nodes, N, increases. The results in this area are known as scaling laws. The fact that, as the network grows, the nodes have to, in addition to their own traffic, relay an increasing amount of other nodes' traffic means limitations to the capacity available for originating traffic.

There are two basic ways to approach a very large network. One can either keep the network area constant and let the node density tend to infinity or keep the node density constant and let the network area tend to infinity. These are called a dense network and an extended network, respectively. In the limit where the size of the network approaches infinity, we have what is called a massively dense network and an infinite network.

In the seminal paper [GK00], the authors show that randomly located nodes in a finite domain can achieve the throughput of $\Theta(C/\sqrt{N \log N})$ bits per second per node for a randomly chosen destination under the protocol model.[2] Later, it was shown that $\Theta(C/\sqrt{N})$ can also be achieved [FDTT07].

The main difference between the two works is that [GK00] assumes that the nodes use a common transmission range. As all of the nodes act as sources and participate in the communication, a common transmission range means that it has to be increased as a function of the number of

[2] $f(n) = \Theta(g(n))$ as $n \to \infty$ denotes that f is bounded both above and below by g asymptotically, i.e., $f(n) = O(g(n))$ and $g(n) = O(f(n))$.

nodes to keep the network connected with high probability. This results in worse spatial reuse as less links can be activated simultaneously. However, since the network traffic is dominated by the relay traffic, it makes sense to use a larger transmission range for access and delivery phases and a shorter one for transport phase to maximize the spatial reuse. Similar results have been obtained for interference models with fixed transmission radius as well as the SINR-based ones.

In the previous works, it has been assumed that all the nodes participate in the communication. If this is not the case, and only one source-destination pair is active at any given time (all remaining nodes acting only as possible relays), the per-node throughput remains constant [DFT06]. Also node mobility can be utilized to achieve constant scaling [GT02]. The cost of this improvement is a substantial delay. In general, there is a trade-off between throughput and delay [GMPS04, TG04].

3.5 Large network scenario and separation of scales

We continue with extremely large networks and at the very limit where the number of nodes in the network is infinite. We look at the network from two different perspectives: the perspective of the whole network, from which a single node is meaningless, and the local perspective, from which the events elsewhere in the network bear no significance. These two viewpoints represent the separation of scales.

If we keep the network area constant and let the node density tend to infinity, in the limit, we have a massively dense network. Keeping the node density constant and letting the network area tend to infinity results in an infinite network. They can be viewed as the two sides of the same coin. At the global scale, an infinite number of nodes in a closed domain, $\mathcal{A}$, with infinitesimal transmission ranges and paths that are smooth geometric curves allow a continuous representation of the network — so-called continuum approximation [Jac04, KS04, CTM09]. From the local perspective, the nodes see a network of separate nodes that continues ad infinitum and are concerned about forwarding the relay traffic that traverses through them.

At this limit, the network capacity problem separates into two scales [HV09]:

1. Macroscopic level routing tries to find routes enabling to carry as much traffic as possible through the network without exceeding the microscopic level capacity constraint.

2. Microscopic level forwarding aims at coordinating the transmissions so that the packets are relayed hop-by-hop as efficiently as possible and spatial reuse is maximized.

Hence, the forwarding problem at the microscopic level sets an upper bound for the amount of traffic that the routing problem at the macroscopic level is allowed to direct to an area of the network.

For example, let us consider a network with uniform traffic demand, where we want to maximize the common per-node capacity. Assume that packets traversing different directions are handled via time sharing. This means that, in each time slot, packets are transmitted to one direction, and the direction changes between time slots so that all the traffic can be forwarded. The microscopic level capacity constraint is determined by the used media access control protocol. The macroscopic level routing system determines the local traffic load at each part of the network. As different directions are handled via time sharing (the microscopic level capacity constraint does not depend on the directional distribution of the traffic), optimal routing balances the load evenly in the network. The common per-node capacity can be increased until the maximum local traffic load reaches the capacity constraint. To maximize the capacity, we have to maximize the microscopic level capacity constraint (forwarding capacity) and minimize the maximum local load.

The separation of scales can be seen as a version of the combined maximum flow and scheduling problem. The microscopic level forwarding task includes scheduling the transmissions and determines the effective capacity of the links. The macroscopic level routing problem attempts to utilize this capacity while taking into account the flow conservation. In the same way as the effective capacity of a link sets an upper bound for the possible flow, the forwarding capacity sets an upper bound for the local load. The difference is that, while the max flow problem works with individual links, the network constitutes a continuous medium on the macroscopic level.

In the following sections, we consider in more detail the two problems. In particular, the microscopic level one is the main topic of this thesis.

3.5.1 Macroscopic level problem

The interest at the macroscopic level is in the end-to-end paths that are smooth continuous curves. As the nodes form a continuum, the traffic demand is defined as the density $\lambda(\boldsymbol{x}_1, \boldsymbol{x}_2)$, where $\lambda(\boldsymbol{x}_1, \boldsymbol{x}_2) \cdot \mathrm{d}A^2$ is the rate of flow of packets from a differential area element $\mathrm{d}A$ at $\boldsymbol{x}_1$ to a differential area element $\mathrm{d}A$ at $\boldsymbol{x}_2$. We call λ the traffic demand density [pkts/s/m^4]. The traffic demand is satisfied by carrying the packets along the paths of a routing system $\mathcal{P}$. This set of paths contains at least one path for every pair of a source and a destination. The local traffic load that is formed as a result is described using the following definitions. Angular flux of packets in direction ϑ, denoted by $\varphi(\vartheta)$, is equal to the rate [pkts/s/m/rad] at which packets flow in the angle interval $(\vartheta, \vartheta + \mathrm{d}\vartheta)$ across a small line segment of the length $\mathrm{d}s$ perpendicular to direction ϑ divided by $\mathrm{d}s \cdot \mathrm{d}\vartheta$ in the limit when $\mathrm{d}s \to 0$ and $\mathrm{d}\vartheta \to 0$. We write $\varphi(\vartheta) = \Phi \cdot f(\vartheta)$, where Φ is the scalar flux $\Phi = \int_0^{2\pi} \varphi(\vartheta)\mathrm{d}\vartheta$ and $f(\vartheta)$ is the directional distribution $\int_0^{2\pi} f(\vartheta)\mathrm{d}\vartheta = 1$. The directional distribution of traffic $f(\vartheta)$ represents the fraction of traffic in a given direction ϑ. The local traffic load Φ gives the total offered traffic intensity in [pkts/s/m] summed over all the angles.

As mentioned, on the macroscopic scale the problem is the following: given a network area $\mathcal{A}$ and the traffic matrix, find a routing system $\mathcal{P}$, i.e., a set of paths (smooth curves), such that at every point $\boldsymbol{x}$, the local microscopic scale capacity constraint is satisfied. With I^* denoting the microscopic level variable, the multidirectional forwarding capacity, the constraint reads,

$$\Phi(\boldsymbol{x}; \mathcal{P}) \leq I^*[f(\vartheta, \boldsymbol{x}; \mathcal{P})] \qquad \forall \boldsymbol{x} \in \mathcal{A}, \tag{3.9}$$

where the scalar flux Φ and the directional distribution $f(\vartheta)$ are functions of $\boldsymbol{x}$ as determined by the routing system $\mathcal{P}$. In particular, the network capacity problem is to find a routing system $\mathcal{P}$ such that the above condition is satisfied with the maximal possible scalar multiplier of a given form of the traffic matrix. To be explicit, this leads to the following modified load balancing problem

$$\max_{\mathcal{P}} \min_{\boldsymbol{x} \in \mathcal{A}} I^*[f(\vartheta, \boldsymbol{x}; \mathcal{P})] / \Phi(\boldsymbol{x}; \mathcal{P}). \tag{3.10}$$

When the above maxmin problem is solved with a unit traffic matrix with the total traffic of 1 pkts/s, then the maxmin value gives the network capacity. While solving the maxmin problem is outside the scope of the

thesis, we will return to the question of the impact of the microscopic level constraint on the macroscopic level problem later.

3.5.2 Microscopic level problem

From the local perspective, only the direction of each packet is relevant, and at the microscopic level the nodes are concerned about forwarding a given packet to this direction that is defined by the chosen routing. Doing this as efficiently as possible is referred to as the microscopic level forwarding problem, and it is the main focus of this thesis. At the microscopic level, the network appears to be infinite. We assume that node locations are distributed according to a stationary and isotropic point process that is also ergodic [Hae12] (such as the homogeneous PPP used throughout the thesis). In this "locally infinite" network, the traffic is solely relay traffic and the directional distribution of the traffic appears the same everywhere in the network. A special emphasis is given to the single-directional traffic as it gives a guaranteed lower bound for the problem. In what follows, we go through the bases for the above problem formulation and other characteristics of the microscopic level.

The local directional distribution of the traffic, which is determined by the routing system $\mathcal{P}$, is the only parameter coming to the microscopic level problem from the macroscopic level. The task at the microscopic level is to find a coordinated forwarding scheme that handles traffic with this directional distribution as efficiently as possible. The capability of the microscopic level to forward traffic going in different directions in the seemingly infinite network sets an upper bound for the allowed macroscopic level load. This microscopic level characteristic, the capability to forward traffic with given directional distribution in an infinite network, is called the multidirectional forwarding capacity.

The *multidirectional forwarding capacity* I^* is defined as the maximum sustainable mean density of progress [pkts/m/s], i.e., the density of packets multiplied by their mean velocity in their respective directions.[3] It depends on the directional distribution, $f(\vartheta)$, of the traffic, and we denote it by $I^*[f(\vartheta)]$, using square brackets to emphasize the functional dependence on $f(\vartheta)$. As the point process is ergodic, it is possible to obtain a statistically meaningful estimate for the forwarding capacity by analyzing an appropriately large sample of one network realization, as if we

[3] Information could also be measured, e.g., in bits.

were analyzing one part of a very large network. It is equivalently possible to study multiple network realizations.

The multidirectional forwarding capacity, when studied as a function of the node degree, remains zero until the network percolates. Currently, not much more is known about $I^*[f(\vartheta)]$, but on general grounds one can state that for any $f(\vartheta)$

$$I_1^* \leq I^*[f(\vartheta)] \leq I_\infty^*, \tag{3.11}$$

where I_∞^* is the limit for the case where the directional distribution is uniform, $f(\vartheta) = 1/2\pi$, and I_1^* is the limit with traffic in a single direction with $f(\vartheta) = \delta(\vartheta)$, i.e., the Dirac delta function.

The first inequality (3.11) is the following sufficiency condition: If the traffic flow of intensity I_1^* can be sustained in a single direction, then a traffic load Φ with an arbitrary directional distribution $f(\vartheta)$ satisfying $\Phi \leq I_1^*$ can be handled by a simple time-sharing, by allocating the traffic in the direction increment $(\vartheta, \vartheta + \mathrm{d}\vartheta)$ the time share $f(\vartheta)\mathrm{d}\vartheta$. While the stated constraint is sufficient, it is not a necessary condition. Namely, when the traffic consists of a mixture of flows in different directions, it is, in general, possible to carry more traffic by properly interleaving the use of links for flows in different directions in the same time slot. This is not generally true if the assumption about the isotropy of the point process is dropped.

The second inequality (3.11) is rather obvious after this. The multidirectional forwarding capacity for isotropic traffic is at least as great as that for any other traffic pattern, as the uniform distribution can be constructed as a linear superposition of rotated (by $\mathrm{d}\vartheta$) copies of any other distribution and correspondingly served in time-sharing manner. That is to say that the interleaving advantage is greatest for isotropic traffic. It should be noted that all the quantities depend also on the network parameters, suppressed here for clarity. The next subsection outlines how the number of parameters is controlled using dimensional analysis.

Dimensional analysis

In the microscopic level problem, the number of parameters needed to describe the problem can be reduced by dimensional analysis [Buc14]. The maximum sustainable density of flow (obtained with optimal global coordination of the transmissions) depends on the physical parameters at hand: density of nodes n [1/m^2], (maximum) transmission range ρ [m],

and nominal capacity[4] of a link C [1/s]. For a given directional distribution $f(\vartheta)$, the multidirectional forwarding capacity, $I^*[f(\vartheta)]$, can be expressed as any combination of the parameters having the dimension 1/m/s multiplied by a function of all the independent dimensionless parameters that can be formed. A combination of parameters of dimension 1/m/s is provided by $C\sqrt{n}$, and there is only one dimensionless parameter, namely the mean degree of a node $\nu = \pi n \rho^2$ (the constant π is unimportant as it can be absorbed in the definition). Thus,

$$I^*[f(\vartheta)](C, n, \rho) = C\sqrt{n}\, u(\nu; f), \tag{3.12}$$

where u is an unknown dimensionless function to be determined.

The separation of scales exhibits the same $C/\sqrt{N}$ behavior as mentioned in Section 3.4. Given the network domain of area A, the traffic demand density, and the routing system $\mathcal{P}$, the scalar flux Φ at $\boldsymbol{x}$ is just a constant. Thus the scaling law for the network capacity per node follows from the $C\sqrt{n}$ factor of I^* (multiply by A to get the progress/time in the network, then divide by the mean progress needed to reach the destination node ($\sim \sqrt{A}$), and finally divide per node $N = nA$).

3.5.3 Justifying the separation

We now discuss when the separation of scales is valid. From the global point of view, the microscopic level problems (in different parts of the network) are not separate but connected. In a massively dense network, however, this connection is "small". As mentioned, from the local perspective, the events elsewhere in the network are meaningless, and the directional distribution of traffic appears to be the same all around (uniform, i.e., independent of location, but not necessarily balanced). There is still a mismatch between the directional distributions in different parts of the network at the macroscopic level. The section at hand illustrates how this seeming defect can be understood.

The true directional distribution of the traffic is different in different parts of the network, but it is enough to study the local directional distributions (one at a time). Consider an N-node network in a unit square with routing system $\mathcal{P}$. We divide the network into subsquares of size $1/\lfloor N^{1/4} \rfloor \times 1/\lfloor N^{1/4} \rfloor$. Essentially, we now have $\sqrt{N}$ subsquares with the average of $\sqrt{N}$ nodes per square. As the number of nodes in the network,

[4] Under SINR-based interference models, this can be interpreted as bandwidth [Hz].

N, grows to infinity, so does the number of nodes in each of the $\sqrt{N}$ subsquares. At the same time, the number of subsquares tends to infinity and the area of a subsquare to zero. As the paths are smooth geometric curves, the directional distribution of the traffic (angular flux) in each subsquare becomes more and more homogeneous. The adjacent subsquares also begin to statistically resemble each other. This means that the number of additional time slots needed to schedule the links near the borders reduces and becomes negligible.

3.6 Scope of the thesis

The focus of this thesis is on the microscopic forwarding capacity problem. We begin by studying the instantaneous forwarding capacity, $\hat{I}$, in Chapter 4. We try to find the transmission mode that would maximize the number of transmissions per unit area multiplied by the distance the packets are moved in their respective directions. The results are upper bounds for the forwarding capacity with multihop traffic since the same performance cannot be sustained over all time slots. Even the single time slot version is a difficult problem of stochastic geometry that does not allow an analytical solution. As we are studying only one transmission mode, we are able to examine the asymptotic behavior of the quantity though. Also more complex SINR-based interference models are considered. The studied cases correspond to a single-directional case and a case where the directional distribution of traffic is positive for all directions.

In Chapter 5, we study the continuous case and the (actual) forwarding capacity. As mentioned, the nodes see the network from the local perspective as separate nodes that continue ad infinitum and are concerned about forwarding the relay traffic that traverses through them. The traffic is solely relay traffic and no originating or terminating traffic exists in the problem. This can be understood as follows. As the number of nodes in the network tends to infinity the per-node capacity goes to zero, as described in Section 3.4. The relay traffic is, thus, dominating in the network. All the originating and terminating traffic can be scheduled in separate time slots whose number is negligible compared to the amount of time slots needed to carry the relay traffic similarly as in [FDTT07].

In practice, this means that, in the simulations, we have to generate the relay traffic somehow. We do this by placing artificial sources and sinks on different sides of the network. For example, when the sources

are placed on the left-hand side of the network and the sinks on the right-hand side, we have a single-directional traffic flow along the x-axis. The packets are not aimed to a specific sink, but a small deviation from the optimal direction is allowed or even necessary to avoid local bottlenecks. The studied networks are large but finite. By studying the forwarding capacity as a function of the size of the network, we can extrapolate to infinity.

Much emphasis is given to the single-directional forwarding capacity problem (i.e., finding I_1^*) because, as discussed, it represents a guaranteed minimum capacity for any directional distribution. Regardless of the macroscopic level routing and the resulting distribution of directions, time sharing can be used for handling the directions separately, one per time slot. The results for the single-directional forwarding capacity are also compatible with the existing results for the macroscopic level problem, i.e., the ones studying the simple load balancing problem [HV06, HV07b, HV07a, HV08, HV09]. Recall that assuming time sharing between directions, the macroscopic level problem is simply that of load balancing without the sense of direction. Combining the results for the forwarding capacity with the macroscopic level routing problem is considered in Section 5.6.

4. Instantaneous forwarding capacity

By studying the maximum amount of traffic that can be forwarded in a single time slot, the *instantaneous forwarding capacity* (IFC), it is possible to get a natural upper bound for the capacity of continuous flows. In this chapter, we concentrate on the microscopic level transport network of randomly distributed nodes in an infinite plane (see Section 3.5.2). The upper bound is obtained from the maximum weight independent set of links, i.e., a single transmission mode of the network that has the maximum weight per unit area. As a single transmission mode does not constitute a connected flow network, the same performance cannot be achieved with continuous traffic.

When the transport network is sparse, i.e., when the mean neighborhood size is small, the upper bound is loose. Short range connections can be formed by all the nodes that have at least one neighbor, but the actual forwarding capacity requiring long range connectivity becomes nonzero only at the percolation threshold (see Section 3.1.1). While the upper bound is poor when ν is small, it gets better when the neighborhood size grows. In most cases, it is computationally too challenging to obtain results from dense transport networks in the continuous case. The instantaneous capacity offers a possibility to study just these networks, as when there are a lot of links to choose from, it is possible to use a very good transmission mode in every time slot.

In what follows, we introduce the IFC in more detail. Then we study how it behaves asymptotically, i.e., when the network is either very sparse or very dense. The node locations are assumed to follow a homogeneous PPP. Two numerical simulation algorithms are presented next for studying the IFC between these extremes. In the last part of the chapter, we summarize the findings.

4.1 Instantaneous forwarding capacity problem

We try to find the maximum weight independent set (MWIS) of links in a network large enough to represent an infinite plane. With an appropriate weight, it gives us the instantaneous forwarding capacity, that is, the maximum amount of traffic that can be forwarded in a single time slot.

In the single-directional case, the density of progress is maximized when a link is weighted by the length of the projection of the link in the given direction, e.g., on the x-axis. We assume that the direction (in the single-directional case) is always along the x-axis and refer to this case as x-progress. The length of the projection tells us how much the traffic is moved in the x-direction or the amount of x-progress. The instantaneous capacity of the network is maximized by activating the transmission mode that maximizes the x-progress of the links per unit area.

Alternatively, if we take the length of the link as its weight, we get another upper bound. The solution of the MWIS problem now bounds the multidirectional forwarding capacity with any directional distribution $f(\theta)$, since the progress of a packet in one time slot in its intended direction, whatever it is, cannot exceed the length of the link. In particular, it gives an upper bound for the forwarding capacity I^*_∞ with uniform directional distribution, which we argued to be the greatest one, cf. (3.11). The case weighted by the x-progress gives an upper bound (a tighter one) only for the forwarding capacity I^*_1 corresponding to single-directional traffic or multidirectional traffic handled via time sharing.

In addition to these two cases, we study an unweighted case (weight equal to one). This allows us, for example, to compare the fraction of transmitting nodes of the weighted cases to the maximum number of transmitting nodes. The three cases are referred to as:

A. Unweighted,

B. Weighted by the x-progress of the links,

C. Weighted by the length of the links.

The IFC is denoted by $\hat{I}$, but we also use the notation $U = \hat{I}/C$ as the nominal link capacity C is unimportant. Note that under the SINR-based interference models, C can be interpreted as the bandwidth. Also taking

into account the SINR-based interference models, we measure information in bits instead of packets, as used earlier. Hence, the capacity of a transmission mode is measured either in bits per second (link weight one) or bit-meters per second (weighted by the x-progress or the length of the link). The unit of $\hat{I}$ is either 1/s/m^2 or 1/s/m, respectively. The link weights are denoted by subindices ι, x and l in the context of $\hat{I}$ (and its dimensionless counterpart u introduced next), while subindex $*$ refers to any of the weights. Superindices, on the other hand, are later used to refer different models, e.g., interference. Depending on the definition of the link weight, $\hat{I}(\nu)$ can be expressed using dimensional analysis (cf. Section 3.5.2) as

$$\hat{I}_\iota(\nu) = C\, n\, u_\iota(\nu(n,\rho)), \tag{4.1}$$

$$\hat{I}_x(\nu) = C\sqrt{n}\, u_x(\nu(n,\rho)), \text{ or} \tag{4.2}$$

$$\hat{I}_l(\nu) = C\sqrt{n}\, u_l(\nu(n,\rho)), \tag{4.3}$$

where $u_\iota(\nu)$, $u_x(\nu)$, and $u_l(\nu)$ are dimensionless functions of the independent dimensionless parameter ν. Note that functions $u_*(\nu)$ are different for different interference models (denoted by superindices) and their parameters.

4.1.1 Studying the IFC

The instantaneous forwarding capacity, and case B in particular, has a connection with the minimum cut problem and the upper bounds that can be obtained with a limited set of cuts (3.4) and by switching the order of optimization (3.5). The single-directional forwarding capacity, I_1^*, can be interpreted as the number of packets crossing a unit length of line perpendicular to the direction of the flow in a unit time. Thus, it is reasonable to consider cuts that correspond to a straight line in the vertical direction when the traffic is flowing in the horizontal direction. If the limited set of cuts consists solely of a single cut, the task equals finding the size of the maximum independent set of links crossing the given line. Considering just one cut gives a relatively loose upper bound for the single directional forwarding capacity (see Publication I). After the maximization of the capacity is done, the cut does not represent an average cut (and much less a minimum cut), since the same kind of performance cannot be achieved with a cut that suffers from the interference caused by the links of this cut.

As the set of cuts grows, the bound gets tighter. With multiple cuts, the task becomes maximizing the smallest of the capacities of the cuts. When

the plane is filled with (infinitely long) straight vertical cuts separated by a distance d, the situation is symmetric, and by maximizing the sum of the capacities, we maximize the minimum. As the distance between the cuts goes to zero, the task of maximizing the number of times a link crosses a cut (i.e., the corresponding line) becomes maximizing the sum of the progresses of the links, that is, finding the maximum weight independent set. The contribution of a link, i.e., the number of times it crosses a cut, is proportional to its progress, and in order to get the optimal value for the forwarding capacity, we need to find the independent set of links with maximal total progress.

For the computational task of finding numerical values for the instantaneous forwarding capacity, we have implemented two simulation algorithms. The moving window algorithm (MWA) is able to find the maximum weight independent set of links in a strip of limited height but unlimited length. By studying the results as a function of the height of the strip, we are able to extrapolate to the infinite plane. The method is suitable for interference models with fixed transmission/interference range such as the Boolean interference model. The algorithm is presented and analyzed in Publications I, III, and V.

Simulated annealing (SA) is a probabilistic method for solving difficult optimization problems. It is statistically guaranteed to find an optimal solution. However, in a finite time, it is only possible to find an approximate solution. In addition, the method is time consuming. The main merit of the method is that it permits the transition from a graph based interference model to a more realistic modeling of the underlying wireless medium. Publication VI covers the Boolean interference model and the Shannon model while Publication VII considers the topic under the SINR threshold model.

Before introducing the two implementation methods and the results, we study what happens to the instantaneous forwarding capacity when the mean neighborhood size is either small or large (as presented in Publications III, V, VI, and VII).

4.2 Asymptotic characterization

In order to gain a better understanding of how the dimensionless functions $u_*(\nu)$, given by (4.1-4.3), behave, it is useful to consider them analytically under the different interference models. The asymptotic analysis

studies the performance in small densities, $\nu \ll 1$, and in high densities, $\nu \gg 1$. The obtained theoretical limits provide useful insight and will be compared to simulations in Section 4.5.

4.2.1 Boolean interference model asymptotics

In this section we consider the asymptotic behavior of the dimensionless functions $u_*(\nu)$ of the Boolean interference model when the mean node degree approaches zero or infinity. The analysis when ν approaches infinity is rudimentary but believed to capture the essential dependency. We also consider bidirectional interference (see Section 2.3.5) where both the transmitter and the receiver have to be outside the range of interfering transmissions. The regular model where only the receiver has to be free is referred to as unidirectional interference.

Asymptotics in the limit $\nu \to 0$

When the mean degree of a node approaches zero, the same consideration is valid for both uni- (2.6) and bidirectional (2.14) interference as well as both fixed and adjustable transmission radius (cf. Section 2.3.5). Basically, it is more about being able to form links than it is about the links interfering with each other.

In the unweighted case (A), a general upper bound for the function U_ι, the maximum number of links per unit area, is given by

$$U_\iota(n, \rho) \leq \frac{1}{2}\, n(1 - e^{-\nu}).$$

The reasoning with the above inequality is that there are on the average n nodes per unit area, and that one obviously gets an upper bound for $U_\iota(n, \rho)$ if each node can freely choose the neighbor to form a link with, without any restrictions imposed by other links. The factor $1/2$ accounts for the fact that it takes two nodes to form a link. The parenthetical expression is the probability that a node has a neighbor. Written in terms of $u_\iota(\nu)$ the upper bound takes the form

$$u_\iota(\nu) \leq \frac{1}{2}(1 - e^{-\nu}).$$

It is also obvious that asymptotically when $\nu \to 0$ the upper bound becomes tight, since in the rare cases when a node has a neighbor within its transmission radius, they can indeed form a link with a high probability without any other link interfering. In this asymptotic regime the probability $(1 - \exp\{-\nu\}) \approx \nu$, and we have

$$u_\iota(\nu) \sim \frac{1}{2}\,\nu, \text{ when } \nu \to 0. \tag{4.4}$$

For the case weighted by the x-progress (B), the same reasoning otherwise applies but instead of the probability of having a neighbor, we have the mean x-progress of the link to the furthest neighbor. Hence, the general upper bound for the function U_x becomes

$$U_x(n,\rho) \leq \frac{1}{2}\, n\rho\, X(\nu),$$

where $X(\nu)$ is the mean distance (in units of ρ) from a randomly chosen node to its most distant neighbor node in the x-direction, i.e., absolute value of the x-distance (if there is none, the distance is taken to be zero). For $u_x(\nu)$, we have

$$u_x(\nu) \leq \frac{1}{2}\sqrt{\frac{\nu}{\pi}}\, X(\nu).$$

When ν is small, $X(\nu) \approx 4\nu/(3\pi)$, where $4/(3\pi)$ is the mean x-distance to a neighbor, and ν is the approximate probability of having a neighbor. Thus, we have

$$u_x(\nu) \sim \frac{2}{3}\left(\frac{\nu}{\pi}\right)^{3/2}, \text{ when } \nu \to 0. \tag{4.5}$$

The third case (C) is similar to the second case, but we have to replace the x-distance between the nodes by the actual distance, $L(\nu)$. Hence, $L(\nu) \approx 2\nu/3$, and

$$u_l(\nu) \sim \frac{1}{3\sqrt{\pi}}\, \nu^{3/2}, \text{ when } \nu \to 0. \tag{4.6}$$

Asymptotics in the limit $\nu \to \infty$

We now turn our attention to how $U_*(n,\rho)$ behaves for large n when ρ is considered to be fixed and present a plausible reasoning for the asymptotics. The analysis is different for the unidirectional and bidirectional interference.

Unidirectional interference leads to the starting observation that if the end points of a link can be arbitrarily placed on a continuous plane, then the most efficient way of packing links is to form vertical columns. The claim is most obvious in the case with x-progress (B), where we assume single-directional traffic from left to right. There has to be a distance larger than ρ between two consecutive links, as illustrated in Figure 4.1, but the vertical distance between the links can be small. In fact, the Boolean interference model (unrealistically) sets no limit on how densely the links can be vertically packed: two parallel links of maximal length ρ, however close, never interfere with each other. This suggests that for a very high n, when there are nodes almost everywhere, the maximum weight independent sets consist of links forming vertical columns.

In cases A and C, that are undirected, the packing can be done even more efficiently by changing the direction of every other column as shown in Figure 4.2. This way a small distance ε is enough between the columns as the endpoints near each other are all either transmitters or receivers.

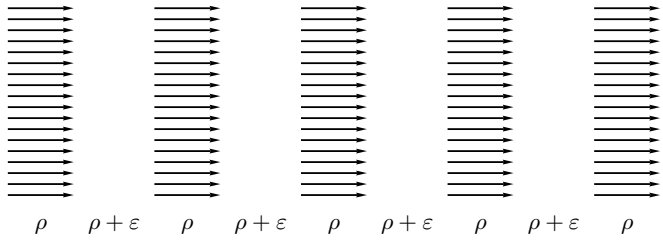


Figure 4.1. On a continuous plane links can be efficiently stacked in vertical columns (case B).

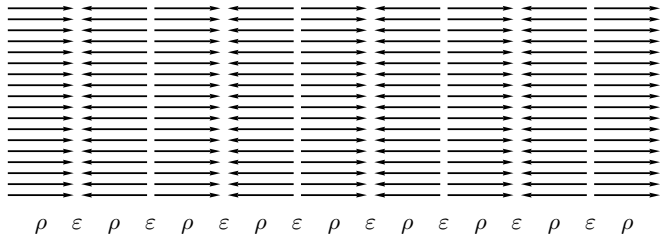


Figure 4.2. In cases A and C, links can be packed even tighter since ε margin is enough between columns transmitting in alternating directions.

The next step is to estimate the expected vertical distance between the links. Based on the above observation we consider a naive model where, starting from a horizontal link of length ρ, the end points of the next link above are determined independently by proceeding in the vertical direction in the shown areas of Figure 4.3 until next node (from the Poisson process) is found.

The width x of the area A (gray in Fig. 4.3) between the vertical line and the circle is for small heights y approximately parabolic, $x \approx y^2/(2\rho)$. Denote the coordinates (random variables) of the node by (X, Y). Since $A \sim \mathrm{Exp}(n)$ and $A \approx Y^3/(6\rho)$, we have the complementary cumulative distribution function of Y,

$$\mathbb{P}\{Y > y\} = \mathbb{P}\{A > y^3/(6\rho)\} = e^{-ny^3/(6\rho)},$$

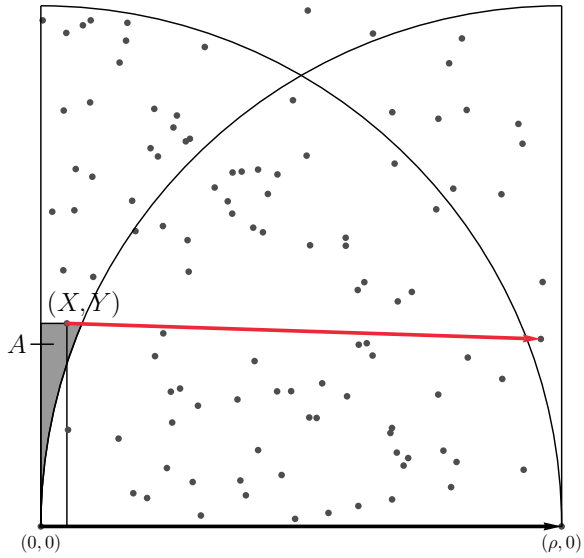


Figure 4.3. The simplified model for estimating the vertical distance between the stacked links.

i.e., $\mathbb{P}\{Y^k > y\} = \mathbb{P}\{Y > y^{1/k}\} = e^{-ny^{3/k}/(6\rho)}$, from which

$$\begin{aligned}\mathbb{E}[Y^k] &= \int_0^\infty e^{-ny^{3/k}/(6\rho)}\mathrm{d}y \\ &= \left(\frac{6\rho}{n}\right)^{k/3}\frac{k}{3}\int_0^\infty s^{k/3-1}e^{-s}\mathrm{d}s \\ &= \left(\frac{6\rho}{n}\right)^{k/3}\frac{k}{3}\,\Gamma\left(\frac{k}{3}\right) = \left(\frac{6\rho}{n}\right)^{k/3}\Gamma\left(1+\frac{k}{3}\right),\end{aligned}$$

where, in the second step, a new variable $s = ny^{3/k}/(6\rho)$ has been introduced. In particular, we have

$$\mathbb{E}[Y] = \Gamma\left(\frac{4}{3}\right)\left(\frac{6\rho}{n}\right)^{1/3}, \quad \mathbb{E}[Y^k] = \frac{\Gamma(1+\frac{k}{3})}{\Gamma(\frac{4}{3})^k}\,\mathbb{E}[Y]^k,$$

whence the variance is

$$\mathbf{Var}(Y) = \left(\frac{\Gamma(\frac{5}{3})}{\Gamma(\frac{4}{3})^2} - 1\right)\,\mathbb{E}[Y]^2 \approx 0.132\mathbb{E}[Y]^2.$$

The distribution of X is determined by that of Y, $X \sim \mathrm{Uniform}(0,\, Y^2/(2\rho))$, from which

$$\mathbb{E}[X] = \mathbb{E}[\mathbb{E}[X|Y]] = \mathbb{E}[\frac{Y^2}{4\rho}] = \frac{\Gamma(\frac{5}{3})}{4\rho\Gamma(\frac{4}{3})^2}\,\mathbb{E}[Y]^2$$

and

$$\begin{aligned}\mathbf{Var}(X) &= \mathbb{E}[\mathbb{E}[X^2|Y]] - \mathbb{E}[X]^2 = \mathbb{E}[\frac{1}{3}(\frac{Y^2}{2\rho})] - \mathbb{E}[\frac{Y^2}{4\rho}]^2 \\ &= \frac{\frac{1}{3}\Gamma(\frac{7}{3}) - \frac{1}{4}\Gamma(\frac{5}{3})^2}{4\Gamma(\frac{4}{3})^4\rho^2}\,\mathbb{E}[Y]^4 \approx \frac{0.0759}{\rho^2}\,\mathbb{E}[Y]^4.\end{aligned}$$

Now, consider the random walk $\mathbf{X}_k = \sum_{i=1}^{k}(X_i, Y_i)$, $k = 1, 2, \ldots$. When $n \to \infty$ this random walk tends to a deterministic motion along the vertical line with constant rate. This is because both $\mathbb{E}[X]$ and $\mathrm{Var}(Y)$ go to zero quadratically in $\mathbb{E}[Y]$. Thus, over a finite interval y, which takes on the average $k = y/\mathbb{E}[Y]$ steps, the expected total displacement in the x-direction is $k\mathbb{E}[X] \sim y\mathbb{E}[Y]$, which goes to zero with $\mathbb{E}[Y]$ as $n \to \infty$. Similarly, the total variance of the displacement in the y-direction after k steps is $k\mathrm{Var}(Y) \sim y\mathbb{E}[Y]$ and goes to zero as $n \to \infty$ (the total variance of the x-displacement goes to zero even faster as the one step variance $\mathrm{Var}(X) \sim \mathbb{E}[Y]^4$).

The fact that the independent random walks of both the end points tend to constant deterministic motion along the vertical lines, in the hindsight justifies considering each step starting from a vertical link of maximal length ρ; the wiggle and contraction of the added links tend to zero.

Finally, we are able to calculate the asymptotic behavior in the three cases starting from the unweighted one (A). From the above it follows that $\mathbb{E}[Y]$ defines the vertical packing distance. As there is one vertical link in every rectangle of height $\mathbb{E}[Y]$ and width $(1+\varepsilon)\rho$, cf. Figure 4.2, the reward per unit area is asymptotically $U_\iota(n, \rho) \approx 1/(\rho\,\mathbb{E}[Y])$,

$$\begin{aligned} U_\iota(n, \rho) &\approx \frac{1}{\Gamma(\frac{4}{3})} \left(\frac{n}{6\rho^4}\right)^{1/3}, \\ u_\iota(\nu) &\approx \frac{1}{\Gamma(\frac{4}{3})} \left(\frac{\sqrt{6}}{\pi}\,\nu\right)^{-2/3}. \end{aligned} \tag{4.7}$$

Similarly in the weighted case (B), as there is one vertical link of length ρ in every rectangle of height $\mathbb{E}[Y]$ and width $(2+\varepsilon)\rho$, the weight per unit area is asymptotically $U_x(n, \rho) \approx 1/(2\mathbb{E}[Y])$,

$$\begin{aligned} U_x(n, \rho) &\approx \frac{1}{2\,\Gamma(\frac{4}{3})} \left(\frac{n}{6\rho}\right)^{1/3}, \\ u_x(\nu) &\approx \frac{1}{2\,\Gamma(\frac{4}{3})} \left(\frac{36}{\pi}\,\nu\right)^{-1/6}. \end{aligned} \tag{4.8}$$

Finally in the third case (C), $U_l(n, \rho) = 2\,U_x(n, \rho)$, and $u_l(\nu) = 2\,u_x(\nu)$ as the number of links compared to the packing in Figure 4.1 can be doubled. Hence,

$$\begin{aligned} U_l(n, \rho) &\approx \frac{1}{\Gamma(\frac{4}{3})} \left(\frac{n}{6\rho}\right)^{1/3}, \\ u_l(\nu) &\approx \frac{1}{\Gamma(\frac{4}{3})} \left(\frac{36}{\pi}\,\nu\right)^{-1/6}. \end{aligned} \tag{4.9}$$

For more realistic interference models, one can conjecture that the asymptotic tail of $u_*(\nu)$ comes down more rapidly than for the unidirectional interference model due to the fact that this model unrealistically allows multiple transmissions just outside the interference range of a receiving node.

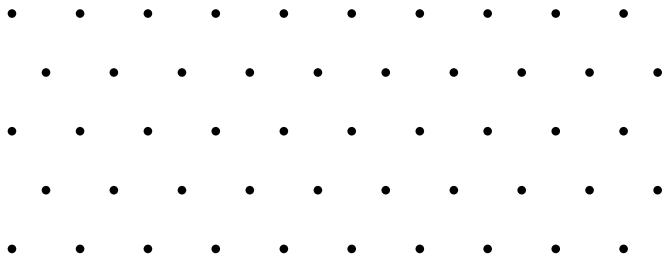


Figure 4.4. Short links forming a triangular lattice maximize the number of transmissions (case A).

Bidirectional interference prevents the links from packing very close to each other. In the unweighted case (A), short links are preferred because then the interference ranges of the origin and destination overlap, and the exclusion area of the link is minimized. In the limit, the length of the links tends to zero, and the link constellation approaches equilateral triangular lattice with one transmission range between the links, see Figure 4.4. Hence,

$$\begin{aligned} U_\iota(n,\rho) &\approx \frac{2}{\sqrt{3}\rho^2}, \\ u_\iota(\nu) &\approx \frac{2\pi}{\sqrt{3}}\,\nu^{-1}. \end{aligned} \tag{4.10}$$

In the weighted cases (B) and (C), the end points of the links form the same lattice as depicted in Figure 4.5. Hence, we have the reward ρ per area of $\sqrt{3}\rho^2$, and

$$\begin{aligned} U_x(n,\rho) = U_l(n,\rho) &\approx \frac{1}{\sqrt{3}\rho}, \\ u_x(\nu) = u_l(\nu) &\approx \sqrt{\frac{\pi}{3}}\,\nu^{-1/2}. \end{aligned} \tag{4.11}$$

The presented asymptotic behavior of $u_*(\nu)$ presumably gives everywhere an upper bound of the true curve. We return to the comparison with the numerical values later in Section 4.5. The above asymptotics for $\nu \to \infty$ apply when the transmission radius is fixed. If adjusting the transmission radius is allowed, the neighborhood size is calculated using the maximum value of the transmission radius. Though the number of potential neighbors increases with a greater ρ, it is always possible to use the

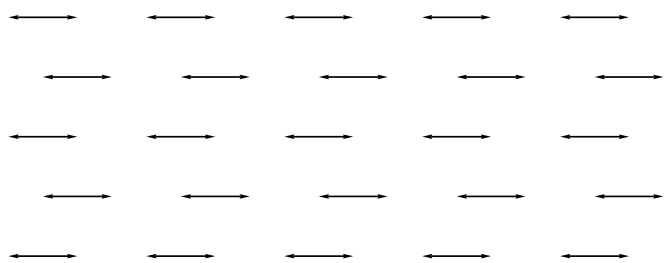


Figure 4.5. When the end points of the links form a triangular lattice (with bidirectional interference in cases B and C), the number of transmitting nodes at the minimum distance from a node is maximized.

previous link configuration unless a better one becomes available. Thus in these cases, $u_*(\nu)$ approaches some limit.

4.2.2 SINR threshold model asymptotics

Now, we move to the SINR-based interference models. Under these interference models, the analysis is more difficult because of the additive interference that has no fixed range. The dimensionless functions corresponding to the first of the models, the SINR threshold model, are denoted by $u_*^{\alpha,\theta}(\nu)$, and they are different for different values of α and θ.

Let us rewrite (2.8) as follows,

$$\mathrm{SINR}(\boldsymbol{r}_l, m) = \frac{\bar{g}(l,l)\,\nu^{\alpha/2}}{\sum\limits_{l' \in m \setminus \{l\}} \bar{g}(l',l)\,\nu^{\alpha/2} + 1}, \tag{4.12}$$

where $\bar{g}$ is the dimensionless function

$$\bar{g}(l',l) = \left(\sqrt{\pi n}\,|\boldsymbol{t}_{l'} - \boldsymbol{r}_l|\right)^{-\alpha}. \tag{4.13}$$

An important observation is that, because of insertion of the factor $\sqrt{\pi n}$, for any realization of the spatial Poisson process the function $\bar{g}(l,l)$ is independent of the scale. That is, if all the distances are stretched or contracted by some factor, the value of $\bar{g}(l,l')$ remains unchanged for any pair of links $\{l,\, l'\}$. Thus, the dependence on the density is fully incorporated in the factors $\nu^{\alpha/2}$ in (4.12). Since this factor controls the ratio of the two terms in the denominator, the low and high density limits may equivalently be called the *noise-limited* and *interference-limited* cases, respectively.

Curves $u_*^{\alpha,\theta}(\nu)$ defined in (4.1) through (4.3) are increasing functions of ν. This stems from the fact that $\mathrm{SINR}(\boldsymbol{r}_l, m)$ of (4.12) is an increasing function of ν. The total capacity of any transmission mode is constant,

but the maximizing mode, and thus the maximum capacity, may change as ν increases and makes more transmission modes feasible.

Let us now consider the interference-limited case $\nu \gg 1$. In this case, the one in the denominator of (4.12) may be neglected, whence the factors $\nu^{\alpha/2}$ cancel out. Therefore, provided that ν is large enough, the problem becomes completely scale-free, independent of ν. No matter how the scale is stretched or contracted, it is always the same mode of active links that realizes the optimum.

Next we turn our attention to the noise-limited case $\nu \ll 1$. Now, the interference term in the denominator of (4.12) may be neglected, whence the $\mathrm{SINR}(\boldsymbol{r}_l, m)$ reduces to the numerator of the expression on the right hand side. Without interference, reception is possible anywhere in the transmission region as it is under the Boolean interference model in the absence of an interfering transmitter, and the two interference models coincide if the spectral efficiency is assumed to be equal. In the noise-limited case, a higher SINR threshold θ directly leads to better spectral efficiency (2.10) as there is no interference and the system is studied as a function of ν.

Under a reasonable assumption that $\theta > 1$, reception is only possible from the closest transmitting node. Now, strong attenuation (large α) increases the capacity independent of ν. As α tends to infinity, the interference is dominated by the interfering transmitter closest to the receiver of the link (or the noise if there are no interfering transmitters within the distance of ρ) that is still farther than the transmitter of the link. Hence, the SINR (4.12) tends to infinity as $\alpha \to \infty$. The effect is stronger when ν is large, and there are more competing transmissions.

4.2.3 Shannon model asymptotics

As with the threshold model, the curves $u_*^\alpha(\nu)$ are increasing functions of ν since $\mathrm{SINR}(\boldsymbol{r}_l, m)$ of (4.12) is an increasing function of ν. Now, also the total weight of transmission mode m,

$$w(m) = \sum_{l \in m} w_l R_l(m), \tag{4.14}$$

where w_l is the weight of link l,

$$w_l = \begin{cases} 1, & \text{unweighted} \\ |[\boldsymbol{t}_l]_1 - [\boldsymbol{r}_l]_1|/\rho_0, & x\text{-progress} \\ |\boldsymbol{t}_l - \boldsymbol{r}_l|/\rho_0, & \text{length}, \end{cases} \tag{4.15}$$

is an increasing function of ν. It follows that this property is inherited also by the optimum, i.e., the maximum over $m \in \mathcal{M}$, even though the mode that realizes the maximum may change as ν increases. Another view to the result is provided by the observation that an increase in ν may be realized by decreasing the noise power σ^2, everything else being held unaltered. It is obvious that this leads to an increase in the capacity.

Again in the interference-limited case $\nu \gg 1$, the one in the denominator of (4.12) may be neglected, whence the factors $\nu^{\alpha/2}$ cancel out. Therefore, provided that ν is large enough, the problem becomes completely scale-free, independent of ν. No matter how the scale is stretched or contracted, it is always the same mode of active links that realizes the optimum.

When also α is large, the capacity of a transmission mode (that is proportional to the sum of the weighted spectral efficiencies) is dominated by the links with high SINR. Thus, (2.12) may be approximated by

$$\log_2 \Big(\bar{g}(l,l) / \sum_{l' \in m \setminus \{l\}} \bar{g}(l',l) \Big).$$

Furthermore, the interference is dominated by the closest interfering transmitter, and the capacity of the transmission mode is proportional to

$$\alpha \sum_{l \in m} w_l \big(\log_2 |\boldsymbol{r}_l - \boldsymbol{t}_{l'}| - \log_2 |\boldsymbol{r}_l - \boldsymbol{t}_l| \big),$$

where $\boldsymbol{t}_{l'}$ is the nearest interfering transmitter. In all, $u_*^\alpha(\nu)$ is approximately a linear function of $\alpha \gg 1$.

Finally, we turn our attention to the noise-limited case $\nu \ll 1$. As before, the interference term in the denominator of (4.12) may now be neglected, whence the $\mathrm{SINR}(\boldsymbol{r}_l, m)$ reduces to the numerator of the expression on the right hand side. Moreover, because of the assumption $\nu \ll 1$, it is small. Then, the logarithm function in (2.12) may be approximated by $\log_2 e \cdot \bar{g}(l,l)\, \nu^{\alpha/2}$, and $\nu^{\alpha/2}$ may be taken out of the sum of the link capacities (see (4.14)) as a common factor, *everything else being independent of* ν. We can again reason that the optimum is always obtained by a given mode, independent of ν, provided that this is small enough. Furthermore, as only the sum $\sum_{l \in m} w_l \ \bar{g}(l,l)$ depends on the transmission mode, one can conclude that in the noise-limited case every node should be an end point of a link, i.e., the fraction of transmitting nodes should approach 50 %, when $\nu \to 0$. The value of the function $u_*^\alpha(\nu)$, however, does not tend to constant as $\nu \to 0$ but to a constant multiplied by the common scale factor $\nu^{\alpha/2}$.

4.3 Moving window algorithm

In this section, we derive an algorithm similar to Retrospective optimization introduced in a study of reservation systems [Vir92]. The moving window algorithm (MWA) is able to find the maximum weight independent set of links in a strip of limited height but unlimited length. By studying the results as a function of the height of the strip, we are able to extrapolate to the infinite plane. The algorithm can be used with interference models that have a finite transmission/interference range such as the Boolean interference model studied here. The algorithm is presented using an illustrative example. A pseudocode of a reference implementation can be found from Publication V.

The algorithm considers a small portion of the network at a time, a rectangular window that moves, and regarding the strip that the moving window covers during a simulation, the result is exact. The algorithm uses a binary tree to enumerate all the possible link combinations in the window area to find the maximum size or weight of an independent set of links per unit area so far conditioned on the choice of the combination of conflict-free links. The length of the simulation is not limited, and the covered strip can be of any desired length. We repeat the simulation for windows of different height to extrapolate the value of the maximum weight per unit area for an infinitely large network.

Because the height of the window in practice is limited, the top and the bottom of the strip can be connected to diminish the border effect and represent an infinite dimension, see Figure 4.6. The perimeter of the formed cylinder needs to be large enough for the results to be meaningful. The other direction can be handled by moving the window along the cylindrical network. The width of the window (i.e., the length of the cylindrical window) needs to be large enough for the window to contain all the links that can possibly interfere with the links that are going to enter the window in the future (that is, 3ρ which is the maximum length of two links and a $\rho + \varepsilon$ margin). The possible combinations of these links are maintained in the binary tree. The links that have already left the window do not affect the possible on/off-state of the links entering the window and can thus be removed by a procedure explained next.

A rooted binary tree represents all the possible link combinations in the window area. Every edge of the tree describes whether the link corresponding to that level is active or not, and the value assigned to each

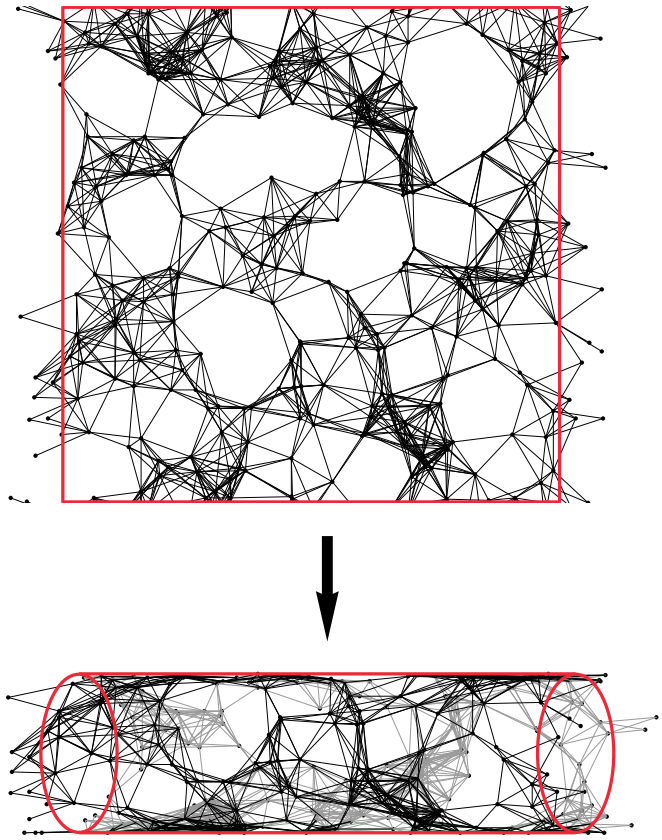


Figure 4.6. The top and the bottom of the window are connected together to diminish the border effect. The formed cylinder is moved in the direction of its axis.

leaf shows the maximum size or weight of the independent set thus far (starting from the initial position of the window) conditioned on the combination of active links in the window represented by the leaf. This is illustrated in Figure 4.7. The figure represents an example of a situation where the simulation of the unweighted case (A) (the values in the vertices represent the size of the independent set of links) with unidirectional interference has just started, and the first four nodes have entered the window making it possible to form six links. The maximum size of an independent set of links is two corresponding to transmission mode[1] $\{uv, xw\}$ or $\{vu, wx\}$.

When the window of Figure 4.7 is being moved to the right, the first event is the node u leaving the window. Since the entering and exiting links are independent, we can combine the on- and off-branches corresponding to a link whose endpoint has been dropped out of the window

[1]Here, a shorter notation uv is used for link (u, v).

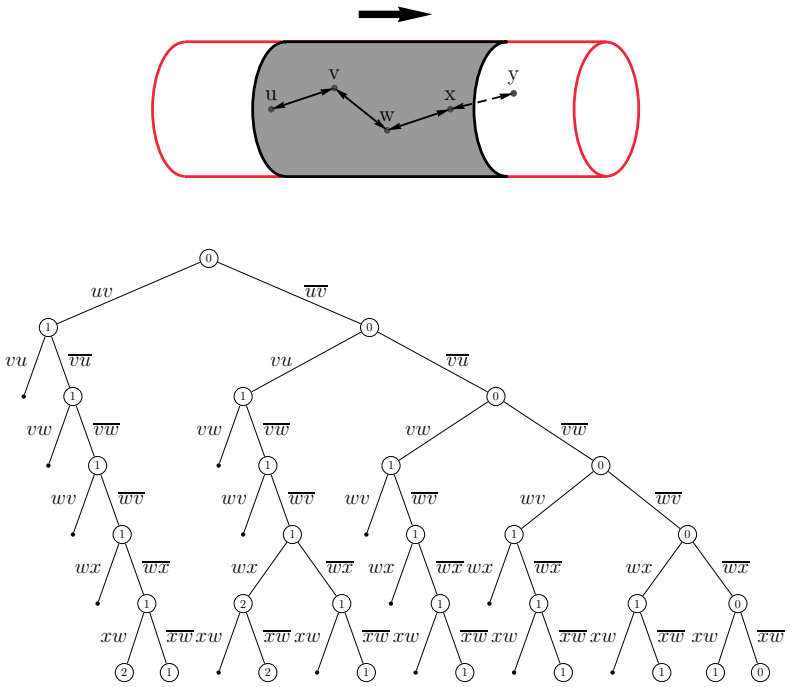


Figure 4.7. A window containing 6 links and the corresponding binary tree with 6 levels representing the links (in alphabetical order) in the window.

and choose the greater values for the new tree. That is, we compare leaves[2] that only differ in the dropped link and choose the maximum of those to be the value of the same node in the new tree where the level corresponding to the dropped link has been eliminated in this way. For example, when the first link to leave the window, uv, is being eliminated from the tree, the leaf with value 2 corresponding to the transmission mode $\{uv, xw\}$ (first from the left in Figure 4.7) is compared to the leaf with value 1 corresponding to the transmission mode $\{xw\}$ (second from the right), and the value of leaf $\{xw\}$ in the new tree in Figure 4.8 (second from the right) is thus 2. Also the link vu has to be removed from the tree when the node u leaves the window.

The next event, when moving the window, happens when node y enters and makes two new links possible. These new links are then added to the binary tree, after which the shape of the tree is the same as in Figure 4.7, but the value assigned to each leaf, except for those with vw or wv, is one higher since in these cases it is possible to use either the link uv or vu that

[2]Besides the leaves also the other vertices can be updated, but they hold no significance to the final result since the leaves cover all the possible link combinations.

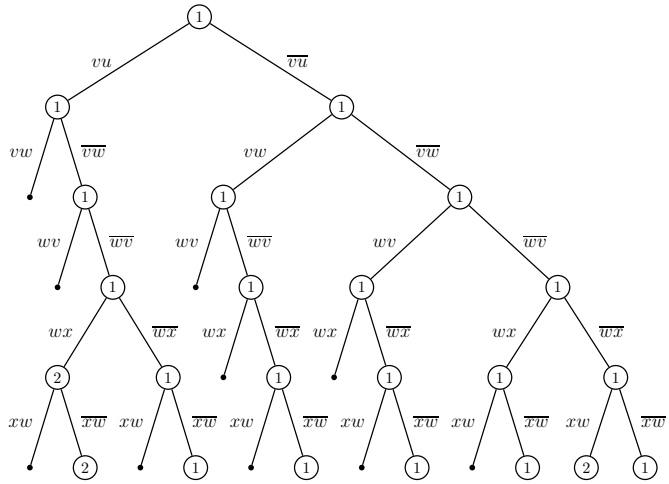


Figure 4.8. The binary tree of Figure 4.7 after the first link to leave the window uv has been removed. In addition to the leaves, also the other vertices have been updated.

have already exited the window. At this point, it is not explicitly visible which dropped links can be activated. Thus, the maximum value in the tree is the maximum size of the independent set of links so far given the set of active links in the window area. In this way, we can generate the network realizations on the fly and progressively find the maximum size or weight of the independent set of links.

We do not maintain information about the links belonging to the maximum weight set, although, this information could be extracted from the algorithm with the cost of used memory. To further minimize the memory requirements the links are removed from the window as soon as they stop interfering with links that are going to enter the window in the future. When a link does not interfere with future links anymore, the information whether the link belongs to the maximum weight independent set is no longer required in the calculations, but the link can be removed from the binary tree maintaining the on/off status of the relevant links. In this way the size of the binary tree, which is the bottleneck limiting the usefulness of the algorithm, can be kept as small as possible. The algorithm limits in no way the length of the simulation in the direction in which the window moves, and when the execution is continued, the result converges (as the process is ergodic) without bias towards the true value. The network realization can be generated on the fly as the window moves, and there is no need to store any information about the network that has left the window. When the simulation is ended, the maximum size or weight

of the independent set of links is the maximum of the values assigned to the leaves of the binary tree. The simulation is repeated to produce confidence intervals for the value. In the other direction, we have to rely on extrapolation and estimate the maximum value for an infinitely wide cylinder, as discussed in the next section.

4.3.1 Extrapolation

This section concentrates on extrapolating the maximum weight per unit area for the infinite plane from the measurements considering only strips of the network with limited height.

The simulations with the MWA produce values $u_*(\nu, p)$, where p (in units of ρ) is the perimeter of the cylinder. For a given ν, a range of p are needed to extrapolate $u_*(\nu)$ to an infinitely wide cylinder. Figure 4.9 represents $u_\iota(p)$ of case A with unidirectional interference and fixed ρ for different values of ν. As seen from the figure, the narrowest cylinders do not give a reliable estimate for larger values of ν. The exact number of values of p required for the extrapolation depends heavily on the case studied as discussed next related to the x-progress case.

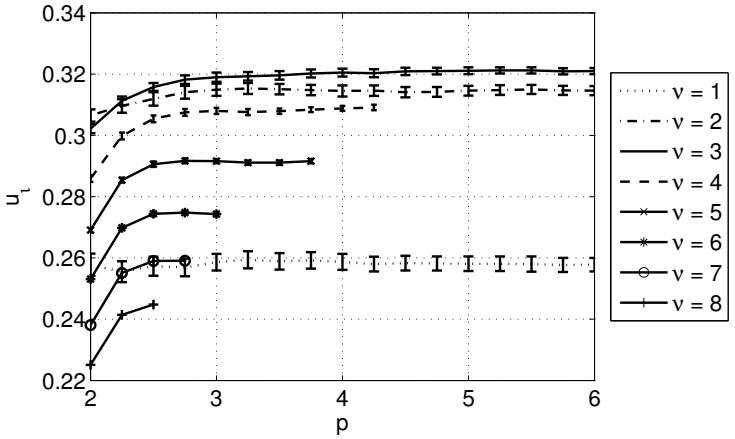


Figure 4.9. Function $u_\iota(p)$ with unidirectional interference and fixed ρ for different values of ν, and the 95 % confidence intervals.

The second unidirectional case (B) with x-progress differs from the other unidirectional cases as it is the only directed case. The working principle of the algorithm does not depend on the direction of the traffic, i.e., the direction in which the progress of the maximal independent set is calculated, but it has to be fixed. We have two extremes: the direction is parallel with the direction in which the cylinder moves (along the cylin-

der) or the progress is calculated perpendicular to the movement of the cylinder (around the cylinder). In the latter case $u_x(p)$ depends heavily on the number of link columns that we are able to fit around the cylinder. The maxima appear when the perimeter is approximately a multiple of 2ρ, meaning that we are able to fit full-length links and the margins $\rho + \varepsilon$ between them. When the direction of the progress is turned by a right angle, we get more stable results as the vertical distance between the links in a column is more stochastic. This effect is illustrated in Figure 4.10. The observation supports the assumption made in Section 4.2 about the most efficient way of packing the links, that is, to form vertical columns.

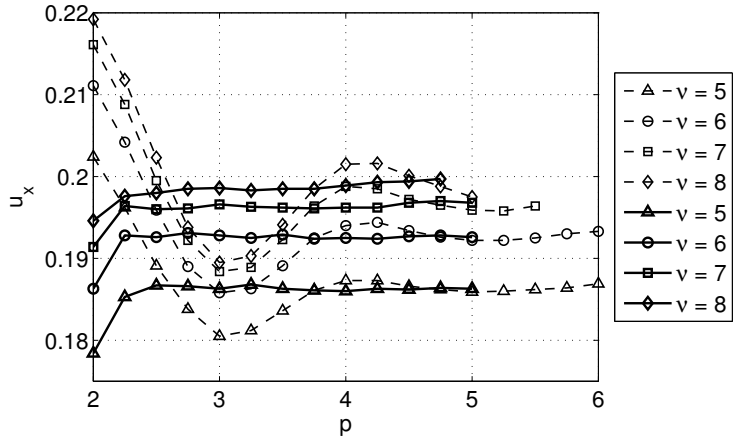


Figure 4.10. Function $u_x(p)$ with unidirectional interference and fixed ρ for different values of ν when direction of progress is around the cylinder (dashed lines) and along the cylinder (solid lines).

With bidirectional interference and fixed transmission radius, the active links start to form patterns when the network gets denser (recall Fig. 4.5). The difference to the towers formed in the unidirectional case is that the distance between the links is more deterministic, and $u_*(\nu, p)$ is thus more sensitive to the perimeter of the cylinder. The undulation is the heaviest in the weighted cases and with large values of ν, and the interesting parameter region of case B is thus the most difficult to reach. When the transmission range is adjustable, $u_*(\nu, p)$ is more stable as the effective number of neighbors is smaller.

4.3.2 Computational complexity

For large window sizes and dense networks, the running time of the algorithm is dominated by the time required to update the binary tree. Let W be the set of nodes in the window and L the set of links between the

nodes in the window. In the worst case, the nodes communicate in pairs, i.e., the links do not interfere with each other (except for the two links between the same two nodes in the undirected cases), and the size of the binary tree is of the order of $O(2^{|L|})$ (i.e., $O(\sqrt{2}^{|W|})$) in case B and $O(\sqrt{3}^{|L|})$ (i.e., $O(\sqrt{3}^{|W|})$) in cases A and C. As in our case the underlying network is random, there is usually heavy interference between the links. If all the links in the window interfere with each other, the size of the tree behaves as $O(|L|^2)$, i.e., $O(|W|^4)$.

4.3.3 Half-space considerations

Here we study the maximum weight problem by considering the Poisson process in a half-space configuration. We show that the maximum weight over the whole plane can be related to a local additive contribution from a single node added on the border of the half space. The result also allows an alternative way of justifying the MWA algorithm.

Consider the problem of the maximum weight of an independent set of links per unit area in a half-space configuration, see Figure 4.11. The border introduces a boundary effect but far from the boundary, inside the body of the Poisson process, the expected total weight per unit area is given by $U_*(n, \rho)$.

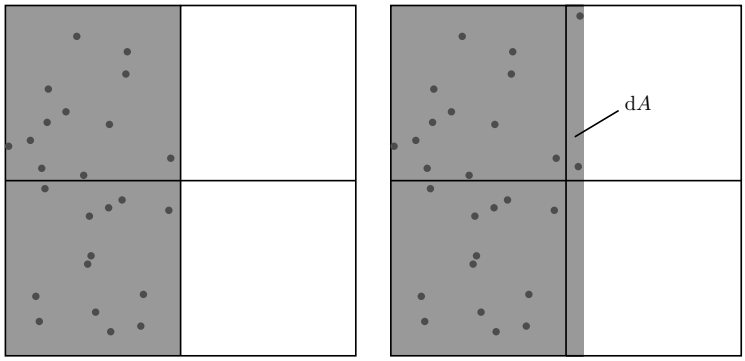


Figure 4.11. Poisson process in the half space and an incremental shift of the boundary.

Now, consider moving the boundary incrementally to the right, so that a new area $\mathrm{d}A$ is covered. The increase in the total weight of the maximal independent set can be evaluated in two different ways: a) one can think that the slice $\mathrm{d}A$ has been added in the body, pushing the boundary to the right; then the added weight is $U_*(n, \rho)\,\mathrm{d}A$, b) one can think that the slice has been added to the right boundary introducing new nodes, as shown in Figure 4.11; In the limit $\mathrm{d}A \to 0$, the added nodes are far apart and

the increase is the number of nodes, $n\,\mathrm{d}A$, multiplied by the contribution from a single added node at the boundary, as illustrated in Figure 4.11. Equating these two yields the average weight per node

$$\frac{1}{n} U_*(n, \rho) = D,$$

where D is the expected total increase of weight due to a single node added at the border. This is in itself an interesting result as it relates mean value over the whole infinite plane to a quantity that has a local character.

To study D, we have to compare two cases. In the first case, there is a node on the border of the network, and in the second there is not. Now, the window starts from a point where one side (right) of the window corresponds to the border of the network and moves away (left) from the border. The top and the bottom of the window are again connected to diminish the border effect and form a cylinder. The effect of the node is the difference in the maximum value in the binary tree in these two simulations. The problem of this formulation is that one simulation produces a single sample instead of some kind of mean, and this causes a large variance.

From the algorithm point of view, it makes no difference whether the additional node is the first node of the simulation or the last. We can draw multiple samples from a single simulation by assuming that every node entering the window is the node on the border of the network. Not even the differences need to be calculated separately in this case — the sum of the differences is simply the total weight of the independent set (and is given by the maximum leaf value in the tree). Hence, we have returned to the original algorithm.

4.4 Simulated annealing

In order to determine the IFC of a large network (that represents the entire plane) also for the SINR-based interference models, we find the transmission mode that maximizes the capacity per unit area using simulated annealing (SA). Simulated annealing is a probabilistic method for solving difficult optimization problems applicable for finding the maximum of a target function that may have multiple local maxima. It is based on [MRR$^+$53], and was later formulated as a more general optimization technique in [KGV83]. The strength of SA is in its ability to cover a wide variety of objective functions and constraints.

The idea of the method comes from the physical process of annealing, where a material cooled slowly enough approaches the ground state of the system, i.e., the state with minimum energy (maximum of the negative energy). As the name indicates, SA tries to simulate this kind of a process. In this method, the current solution is randomly moved to a "neighboring" solution with a probability that depends on the height of the ascent/descent and a parameter called *temperature*. By allowing the algorithm to move to a worse solution, it is possible to avoid being stuck at local optima. When the temperature parameter is properly modified during the optimization, the algorithm also eventually reaches the optimal solution.

The SA algorithm uses the following elements in its operation (physical analogues in parentheses):

1. A finite set of possible states.

2. A real-valued target function (energy) that defines the set of optimal states.

3. A set of neighboring states for each state and the rule for randomly choosing the next state from the neighbors of the current state.

4. A cooling schedule (temperature) that "freezes" the probability distribution of the states to the set of optimal states over time.

Also required are the initial state and the termination condition. In our case, the set of transmission modes forms the state space of the system. The order of the states is determined by the weight of the transmission mode $w(m)$, and the neighboring states are those transmission modes that differ only by one link.

Let us consider a Markov chain with the state space composed of the different transmission modes and with the steady-state distribution

$$\pi(m) \sim \exp\{w(m)/T\}, \tag{4.16}$$

where T is a constant. When T is small, the probability mass is concentrated to the maximum we are interested in,

$$m^* = \arg\max_{m \in \mathcal{M}} w(m). \tag{4.17}$$

The time needed for the Markov chain to reach the steady-state with small T can, depending on the heights of the local maxima, be inordinate. The idea of simulated annealing is to avoid this problem by slowly decreasing the temperature T. Even then, the cooling schedule has to be slow enough for the system to avoid being quenched in a local extremum [Haj88].

4.4.1 Simulated annealing algorithm

A Markov chain M_τ with state space $\mathcal{M}$ is formed, and the steady-state probability of a transmission mode m is chosen to be

$$\pi(m) = \frac{e^{w(m)/T}}{\sum_{m' \in \mathcal{M}} e^{w(m')/T}}, \tag{4.18}$$

where T, i.e., the temperature, is a positive parameter and $w(m)$ is the weight of transmission mode m given by (4.14). For the Boolean interference model $R_l(m) = 1$. For the other interference models the spectral efficiency is given by (2.10) and (2.12). As mentioned, we study three different link weights

$$w_l = \begin{cases} 1, & \text{unweighted} \\ |[\boldsymbol{t}_l]_1 - [\boldsymbol{r}_l]_1|/\rho_0, & x\text{-progress} \\ |\boldsymbol{t}_l - \boldsymbol{r}_l|/\rho_0, & \text{length}. \end{cases}$$

When T is small, the probability of the transmission mode with the highest capacity is close to one.

To assure the required steady-state distribution, the transition probabilities, $p(m', m)$, are chosen so that π shows detailed balance, i.e.,

$$p(m', m)\pi(m') = p(m, m')\pi(m).$$

This is achieved using a proposal distribution $q(m', m)$ along with an acceptance/rejection procedure. The proposal distribution $q(m', m)$ gives the probability that transmission mode m is the candidate to be selected as the next transmission mode when the current transmission mode is m'. When a proposal m with a higher capacity is accepted with probability 1, a transition to a lower capacity, $w(m) < w(m')$, is accepted with probability, r, that can be solved from the detailed balance equation,

$$\begin{aligned} & p(m', m)\pi(m') = p(m, m')\pi(m) \\ \Leftrightarrow\ & r \cdot q(m', m)\pi(m') = 1 \cdot q(m, m')\pi(m) \\ \Leftrightarrow\ & r = \frac{q(m, m')\pi(m)}{q(m', m)\pi(m')}, \end{aligned}$$

and is rejected otherwise. Note that r is always defined since a transition from m' to m can only occur if both $q(m', m)$ and $\pi(m')$ are nonzero.

When the proposal m is obtained by randomly choosing a link, $l \in \mathcal{L}$, and adding it to m' if l is feasible and does not belong to m' and removing it from m' if it does, we have $q(m', m) = q(m, m') = 1/|\mathcal{L}|$, and

$$r = e^{-(w(m')-w(m))/T}. \tag{4.19}$$

This equation is referred to as the Metropolis (acceptance) criterion.

Markov chain M_τ with the known steady-state distribution is simulated to find transmission modes with near optimal capacity. As the temperature T is decreased, the samples come from a process that more and more heavily favors modes with a large capacity.

4.4.2 Implementation aspects

In this section, we discuss implementation aspects of the simulated annealing algorithm and some considerations that need to be taken into account in setting up the simulations.

General parameters

A key parameter of the simulated annealing algorithm is the temperature of the system. When the temperature parameter is properly modified during the optimization, the algorithm also eventually reaches the optimal solution. One way to properly modify the temperature is to use logarithmic cooling, $T(t) = \gamma/\log(1 + t)$, where t is the time and γ is a constant that needs to be large enough [Haj88].

In order to avoid being stuck at local optima, the initial temperature, that depends on γ, needs to be large enough. On the other hand, in order to find the true optimum, the simulation time has to be long. That is, the temperature at the end of the simulation needs to be very small. Using the logarithmic cooling schedule this might take a long time (depending on γ).

A linear cooling schedule, $T(t) = \gamma/t$, is faster than the logarithmic cooling schedule and is not guaranteed to "freeze" the process to the optimal state. However, using a linear cooling schedule it is easier to choose an initial temperature that is large enough for the process to explore the search space and still have a low end temperature that "freezes" the process to a good pseudo-optimal solution. This is a desired property especially when

the simulation time is limited, and the discovery of the optimum cannot be guaranteed (even using logarithmic cooling).

Figure 4.12 illustrates the performance of the two cooling schedules, logarithmic and linear, as a function of the final temperature when the simulation time is fixed to one million steps. From the figure one can see that, though both schedules give nearly the same performance with correctly chosen γ, that fixes the end temperature, the linear schedule is much less sensitive to the selection of the parameter. The larger error bars of the logarithmic schedule indicate that some simulation runs have been unable to escape local maxima and have thus resulted in much worse performance.

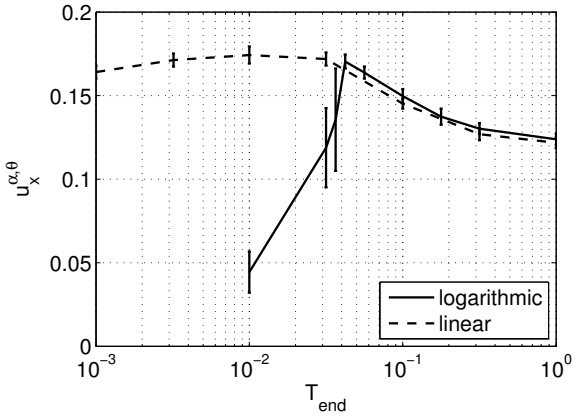
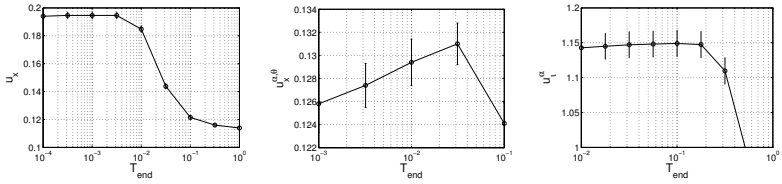


Figure 4.12. The curves $u_x^{\alpha,\theta}(T_{\text{end}})$ for the logarithmic and linear cooling schedules in a one million step simulation with parameters $\alpha = 3$, $\theta = 7$, $\nu = 6$.

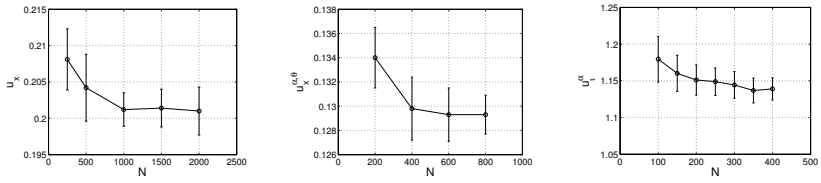
As a conclusion, it is to be noted that when the simulation time is limited, the simulated annealing algorithm cannot be guaranteed to find the optimal solution. Since both the initial and the end temperature are significant to the end result, the parameter selection is easier with a linear cooling schedule. We use a fixed simulation length and linear cooling. Hence, the temperatures of the system form a harmonic sequence. The effect of the end temperature is illustrated in Figure 4.13 for the different interference models.

For the results to be generalizable to a plane, the simulated network needs to be large enough. Under the Boolean interference model, the required size is larger than using SINR-based interference as can be seen from Figure 4.14 that illustrates the capacity as a function of the expected network size. This is due to the fact that the active links generate more deterministic transmission patterns under the unrealistic Boolean inter-



(a) Boolean interference **(b)** SINR threshold **(c)** Shannon

Figure 4.13. Example effects of tuning the temperature at the end of the simulation.



(a) Boolean interference **(b)** SINR threshold **(c)** Shannon

Figure 4.14. The effect of the expected network size (number of nodes) on the network capacity in the simulations.

ference model. To eliminate border effects, we identify the top and the bottom of the square and the circular edges of the formed cylinder to form a torus. Additionally, in the SINR-based model, we allow an interfering signal to travel around the torus for a given number of rounds.

Next, we go through the characteristics of the different interference models.

Boolean interference model

Under the Boolean interference model, the following version of the method is more efficient. If we study a Markov process $\{M(t) \in \mathcal{M} \,|\, t \geq 0\}$, $M(0) = \emptyset$, where free links, $l \in F(t) = F(M(t))$, become active with rate $\lambda_l = \lambda \exp\{w_l/T\}$, and active links, $l \in M(t)$, deactivate themselves with constant rate $\mu_l = \lambda$, the steady-state probability of transmission mode m is (4.18). The embedded discrete-time jump chain M_τ has the activation probability of a free link l equal to

$$p_l = \frac{e^{w_l/T}}{\sum_{l' \in F} e^{w_{l'}/T} + |M|}. \tag{4.20}$$

The deactivation probability of an active link l is $\bar{p}_l = p_l / \exp\{w_l/T\}$.

By simulating this Markov chain when T goes to zero, we are able to go through "good" states in the course of the simulation without any rejected jumps. The drawback comes from maintaining the set of free links, but this cost is relatively small since the interference is local. If the network is operated under a MAC protocol where the activation rate of free links and

the deactivation rate of active links are the ones mentioned above, then the (fixed) parameter T determines the fairness of the network. Small T leads to a high capacity, but only the best links are able to catch the channel, while larger T leads to more random behavior. The SA algorithm takes the network to a highly unfair territory and finally "freezes" it to the optimal transmission mode.

SINR-based interference models

We study square $B = [0, \sqrt{A}] \times [0, \sqrt{A}]$ of a plane with nodes placed according to a PPP of density n. To estimate the amount of interference that is not covered by considering the nodes of B, we calculate the fraction of interference coming from outside a circle that can be fitted into B compared to the interference coming from outside a circle with a radius that equals the mean distance of a node to its nearest neighbor $1/2\sqrt{n}$:

$$\frac{\int_{\sqrt{A}/2}^{\infty} z\, n\, r^{-\alpha}\, 2\pi r\, dr}{\int_{1/2\sqrt{n}}^{\infty} z\, n\, r^{-\alpha}\, 2\pi r\, dr} = \frac{1}{\sqrt{nA}} = \frac{1}{\sqrt{N}}, \tag{4.21}$$

where N is the total number of nodes in the simulation area and z is the fraction of transmitting nodes.

For example, if $N = 100$, less than 10 % of the interference is caused by nodes outside the square. Such a residual interference may approximately be taken into account by letting an interfering signal to travel around the torus for a given number of rounds. Essentially, this is the same as adding copies of B around it in a network that has not been wrapped up to a torus (our implementation has 201×201 squares). This way, we are able to accurately take into account most of the interference coming from outside B.

Under the Shannon model, any two currently inactive nodes are eligible to form a link. Thus, the set of transmission modes easily becomes too large to manage. To keep the simulation times feasible in the final simulations, we limit the number of eligible receivers to, on average, 10 % of the closest nodes by setting the probability of long links to zero in the proposal distribution. As the interference model heavily favors short links, this significantly eases the computational burden. This is definitely true in the interference-limited case, where long links, because of the interference, are doomed to do more harm than they contribute to the total capacity. In the noise-limited end of the curve, this may cause a small anomaly to the transmission modes.

In the noise-limited case, the fraction of transmitting nodes should approach 50 % as the role of interference is negligible, and it is always profitable to form a link between any pair of free nodes. However, the capacity of the longest active links is very small, and thus the temperature parameter has also to be very small for the steady-state probabilities of two transmission modes, one with a long link active and one without, to differ significantly. Temperatures where this happens are no longer suitable for the simulation as a whole. Hence, the SA algorithm is able to find a good estimate for the capacity of a transmission mode (by practically ignoring the longest links) but is unable to find the optimal transmission mode. The phenomenon is illustrated in Figure 4.15 where the fraction of transmitting nodes is shown as a function of ν for the case $\alpha = 3$. In the greedy approach, links are activated in the order of their length, starting from the shortest one, and the next feasible link is always left active if it improves the total capacity. The greedy approach shows that it is possible to activate a higher fraction of nodes than what the SA algorithm suggests without decreasing the capacity — in fact, the greedy approach gives a marginally (small compared with the confidence intervals) better capacity than SA. For larger values of ν, the SA algorithm outperforms the greedy one.

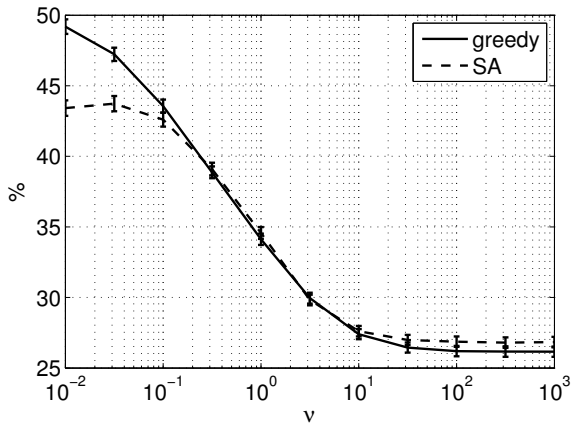


Figure 4.15. The fraction of transmitting nodes in the chosen transmission mode under the Shannon model when $\alpha = 3$ for a greedy approach and simulated annealing.

4.5 Numerical results for IFC

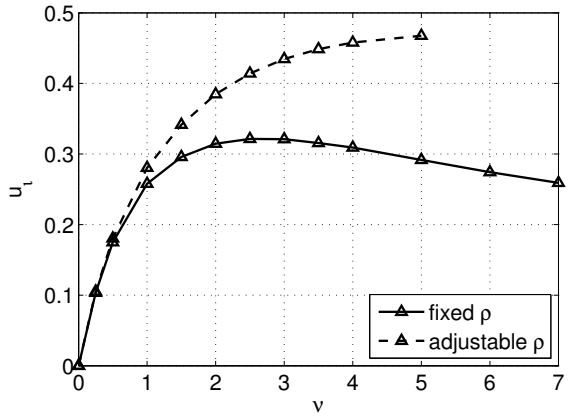
This section presents the numerical results for the instantaneous forwarding capacity obtained using the two algorithms of this chapter. We begin with the moving window algorithm and continue with simulated annealing.

4.5.1 Results of MWA

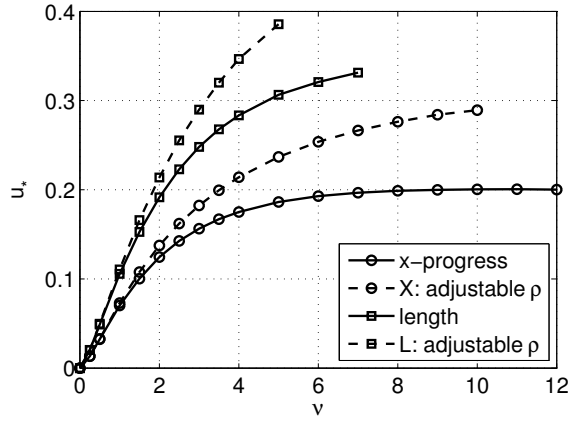
In this section, we present the numerical results obtained by the moving window algorithm and the extrapolation techniques of Section 4.3 for the three cases: unweighted (A), weighted by x-progress (B), and weighted by length (C) under both unidirectional and bidirectional Boolean interference model. In addition to fixed transmission radius ρ, we consider transmission radii freely adjustable up to this maximum value. In this case, the parameter ν is defined to correspond to the mean number of neighbors within the maximum range. The nodes are distributed according to a PPP.

Unidirectional interference
We begin with the unidirectional interference model. Figure 4.16a shows $u_\iota(\nu)$ for the unweighted case (A) with both fixed and adjustable transmission radius. With a fixed transmission radius the maximum occurs at $\nu^* = 2.7$ and equals 0.32. The curve with power control is an increasing one as all configurations that are feasible with a given maximum radius are also possible with a greater maximum, and upper bounded by the theoretical maximum of $1/2$, i.e., one link per two nodes, it tends to a limit when $\nu \to \infty$. As can be seen, the limit is relatively close to the theoretical maximum, implying that the maximum gain (adjustable transmission radius when $\nu \to \infty$ compared with fixed radius at ν^*) from a freely adjustable transmission radius in this setting is approximately 50 % (30 % when both at ν^*).

Figure 4.16b presents $u_*(\nu)$ for the weighted cases (B and C) with a fixed transmission radius and with the possibility to reduce the transmission power to the minimum required. Even though the computational complexity grows with the number of links, it is possible to simulate x-progress with a fixed transmission radius up to the optimal size of the neighborhood. The maximum occurs at $\nu^* \approx 10$ and equals 0.20. In the case of a length-weighted set (C), the number of links in the window is



(a) unweighted case (A)



(b) weighted cases (B) and (C)

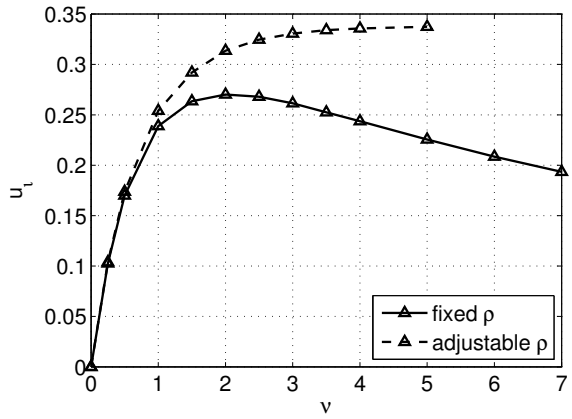
Figure 4.16. Function $u_*(\nu)$ for unidirectional interference with and without power control. The values for case A with fixed ρ are those extrapolated from Figure 4.9.

doubled compared to the second case, since we have to consider both directions separately. Thus, we are not able to find the optimal neighborhood size. As with the unweighted case, the curves corresponding to cases with power control do not have a maximum but are increasing functions of ν tending to a limit when $\nu \to \infty$. Again, the maximum gain from an adjustable transmission radius seems to be close to 50 % in case B.

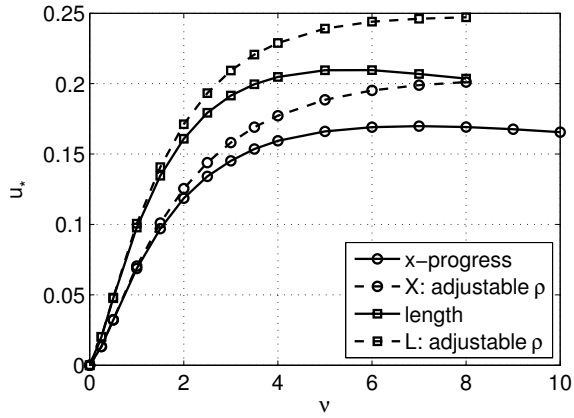
Bidirectional interference

As bidirectional interference (see Section 2.3.5) adds a new constraint for a succesful transmission, the values of $u_*(\nu)$ are less than the corresponding values with unidirectional interference. The unweighted case (A) with the bidirectional interference model is presented in Figure 4.17a. The

maximum with fixed transmission radius equals 0.27 and is achieved with $\nu^* = 2$. The maximum gain from a freely adjustable transmission radius is approximately 25 % (15 % when compared at ν^*).



(a) unweighted case (A)



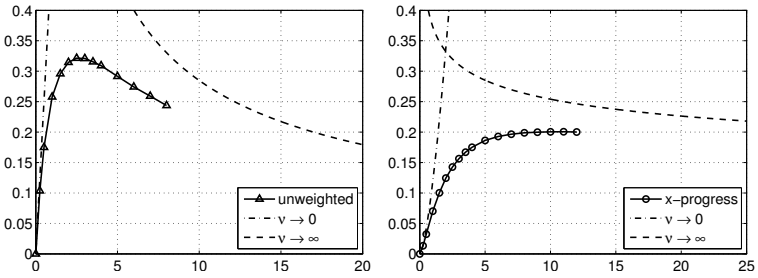
(b) weighted cases (B) and (C)

Figure 4.17. Function $u_*(\nu)$ for bidirectional interference with and without power control.

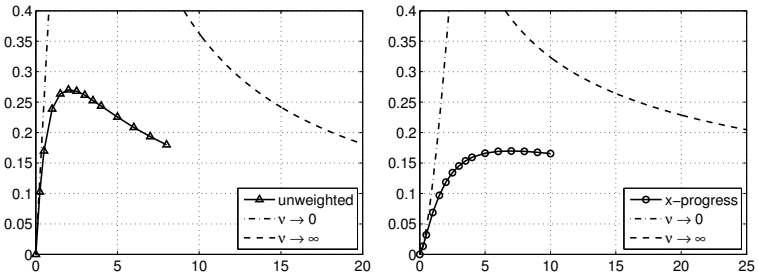
Figure 4.17b shows how $u_x(\nu)$ and $u_l(\nu)$ behave with bidirectional interference. The maxima are $u_x(7) = 0.17$ and $u_l(5.5) = 0.21$. The corresponding approximate gains from an adjustable transmission radius are 20 % and 15 % when calculated at ν^*.

Discussion

For reference, the most important numerical results from all cases have been gathered in Table 4.1. Since the curves are flat near the maximum, it is harder to determine the optimum neighborhood size than the maximum value itself (given with the accuracy of ± 0.001). The values of ν^*



(a) unidirectional $u_\iota(\nu)$ **(b)** unidirectional $u_x(\nu)$



(c) bidirectional $u_\iota(\nu)$ **(d)** bidirectional $u_x(\nu)$

Figure 4.18. Numerically evaluated curve for the functions $u_*(\nu)$ along with the low- and high-ν asymptotic curves.

Table 4.1. Numerical results

	unidirectional			bidirectional		
	fixed		adjustable	fixed		adjustable
	ν^*	$u(\nu^*)$	$u(\infty)$	ν^*	$u(\nu^*)$	$u(\infty)$
A	2.7	0.322	> 0.46	2.1	0.270	> 0.33
B	10	0.200	> 0.29	7.0	0.170	> 0.20
C	> 7	> 0.33	> 0.38	5.5	0.210	> 0.24

are results of parabolic fits near the maximum and given with the accuracy of two significant figures. The lower bounds for the cases with an adjustable transmission radius (and for unidirectional case C with a fixed transmission radius) are based on the obtained maximum values.

As mentioned, the size of the binary tree places limitations on the feasible simulation parameters (ν and p). Since the process is stochastic, the number of links in the window may temporarily grow very large, and the size of the tree may exceed the available memory. Thus, we are only able to simulate cylinders wide enough until a certain value of ν in each case. An adjustable transmission radius is always computationally more com-

plex than a fixed one since it increases the number of conflict-free link combinations and the size of the tree.

In the unidirectional case weighted by the x-progress (B) and in the bidirectional cases, we do not have to consider both links between two nodes. In case B, we never activate a link with a negative weight, and with bidirectional interference, both the links are equal and only one of them can be active. In these cases, it is possible to run simulations with higher values of ν, but the number of links in the window stays continuously on a high level, and the simulations are slow. The results are presented as far as it has been possible to proceed in a reasonable time using a computer with at most 16 GB memory.

Finally, we compare the numerical results with the asymptotic results of Section 4.2.1. Figure 4.18a presents these for $u_\iota(\nu)$ and Figure 4.18b for $u_x(\nu)$ with unidirectional interference. Figures 4.18c and 4.18d present the corresponding curves with bidirectional interference. As can be seen from these figures, even the rudimentary analysis seems to yield a plausible asymptotic behavior for large ν. However, the figures also show that asymptotes alone do not characterize the curves accurately in the most interesting parameter area.

4.5.2 Results of SA

This section presents the results that have been obtained using simulated annealing. The results for the Boolean interference model are compared with earlier results obtained by the MWA in the two studied cases: weighted by x-progress and weighted by length. The results for the SINR threshold model are presented for different combinations of α and θ. The results for the Shannon model are presented last. The nodes are distributed according to a PPP.

The SA algorithm produces as a result the maximum weight of a transmission mode $w(m)$ (4.14) for a finite network realization. When the reference distance is chosen suitably, $\rho_0 = 1/\sqrt{n}$, the dimensionless functions can be calculated simply as $u_*(\nu) = w(m)/N$, where N is the expected number of nodes in the network.

Boolean interference model

Figure 4.19 represents $u_x(\nu)$ and $u_l(\nu)$ compared with the corresponding values from the MWA. As can be seen from the figure, the results are consistent and verify the operation of the SA method. In addition, SA is

able to produce results for larger neighborhood sizes than MWA that is limited by the number of links inside the window (memory requirements determined by the size of a decision tree grows exponentially with the number of links).

The data points of Figure 4.19 are averages over ten network realizations with the average of 1000 nodes. The 95 % confidence intervals are hardly visible. The temperature at the end of the 5-million-step simulation was selected to be $1/10^{2.5} \approx 0.003$ (see Fig. 4.13). The computer time required for a single annealing run with the used parameters varies from 10 minutes (with small ν in the case of the second weight model (B), i.e., the length of the x-projection of the link) to 5 hours (with large ν in the case of the third weight model (C), i.e., the undirected length of the link).

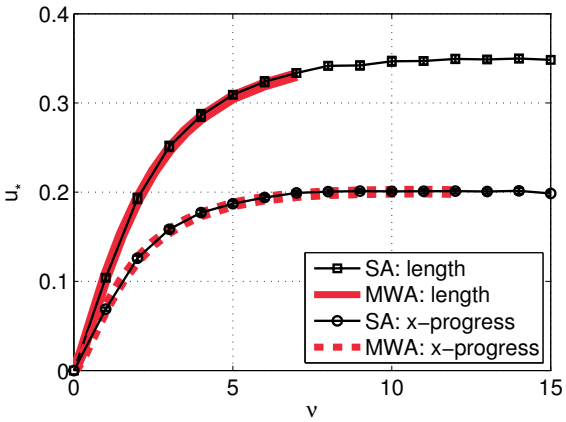


Figure 4.19. Curves $u_x(\nu)$ and $u_l(\nu)$ under the Boolean interference model. The results for SA coincide with the earlier ones and are obtained for a larger range of ν.

SINR threshold model

Figure 4.20 shows the curve $u_x^{\alpha,\theta}(\nu)$ for different combinations of α and θ. The results are averages over 5 network realizations with the mean network size of 800 nodes. The 95 % confidence intervals are shown as error bars. The end value of the temperature parameter T was chosen to be 0.01 after the 20-million-step simulation. As predicted in Section 4.2.2, the curves are increasing functions of ν. The figure also matches with the deductions for α and θ.

Threshold θ has a notable significance when ν is small while α has almost no importance. When ν is small, interference has very little role in constituting the capacity as the problem revolves around being able to

form the links. Thus, the attenuation coefficient α is less important than θ that directly affects the spectral efficiency of the links that can be formed.

When the mean number of neighbors, ν, grows, the effect of α increases as the interfering signals and their attenuation become more important. In this case, higher α naturally leads to higher capacities. The effect of θ with larger ν is not straightforward, but the SINR threshold value that maximizes the capacity depends on the neighborhood size. Figure 4.21 depicts the curve $u_x^{\alpha,\theta}(\theta)$ for different values of ν when α is equal to three. From the figure, it can be seen that when ν is small a higher threshold leads to better performance, but the optimal θ becomes smaller as ν grows.

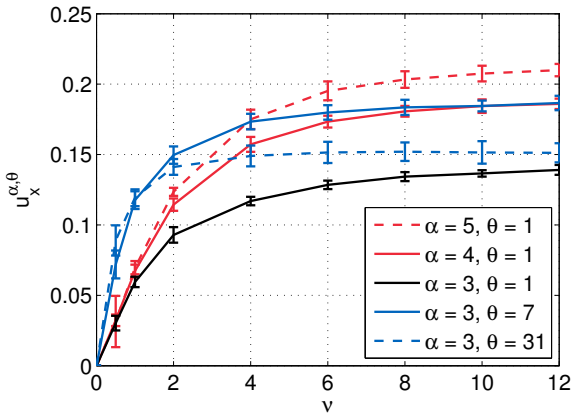


Figure 4.20. The results for the SINR threshold model weighted by the x-progress.

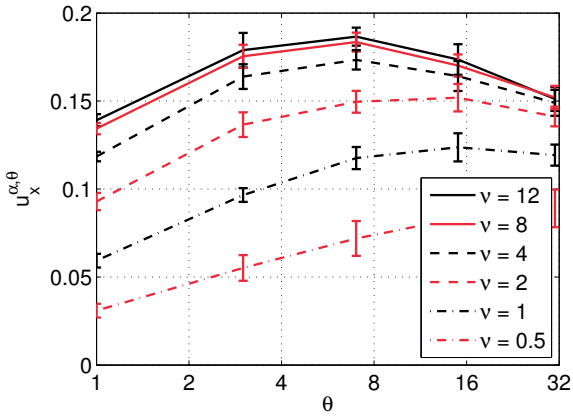


Figure 4.21. The curve $u_x^{\alpha,\theta}(\theta)$ for the SINR threshold model weighted by the x-progress when $\alpha = 3$ and ν varies.

Figure 4.22 illustrates the percentage of transmitting nodes in the optimal transmission mode as a function of ν. When parameter α grows, the interference attenuates faster, and more links can be activated. Thresh-

old θ has an opposite effect. When the threshold grows, a higher SINR value is required for a successful transmission, and a smaller fraction of the links can be activated. The curves start leveling off when ν grows indicating that the asymptotically optimal transmission mode starts to be established already with relatively small values of ν.

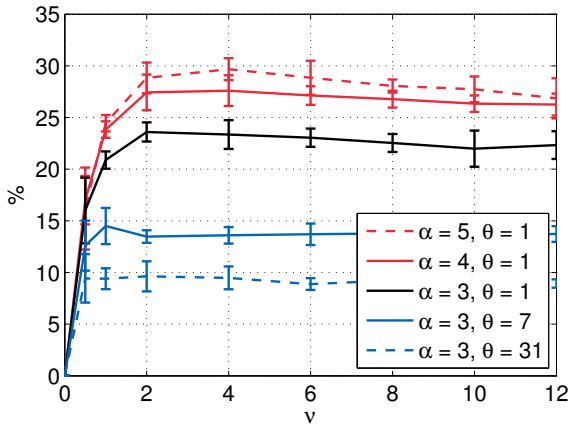


Figure 4.22. The fraction of transmitting nodes under the SINR threshold model weighted by the x-progress.

Shannon model

In Figure 4.23, the simulation result for the function $u_i^\alpha(\nu)$ are presented for different values of α. This function captures all the information about the maximal capacity per node in the considered model as a function of ν. Figure 4.24 illustrates the corresponding percentage of transmitting nodes in the optimal transmission mode as a function of ν. Both curves exhibit a transition around the value $\nu \approx 1$.

The results are averages over 100 network realizations with the average of 250 nodes. The 95 % confidence intervals are shown as error bars. The temperature at the end of the 5-million-step simulation was selected to be 0.1 (see Fig. 4.13b). The computer time required for a single annealing run with the used parameters was at most one hour.

The deduction of Section 4.2.3 that it is always the same mode of active links that realizes the optimum in the interference-limited case is reflected in Figures 4.23 and 4.24 in that both curves level off for large ν, becoming independent of it. The difference between the end points of the curves is almost constant showing that the interference-limit capacity is approximately linear with respect to α.

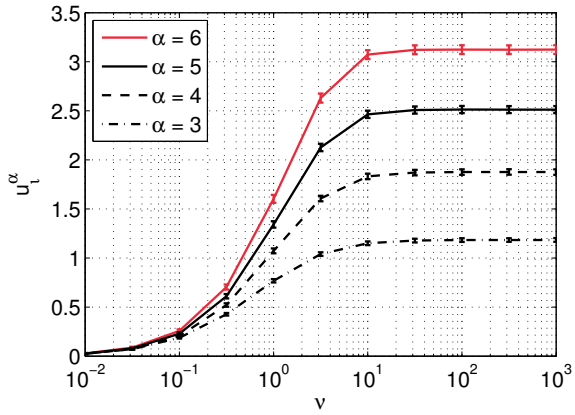


Figure 4.23. The results for the unweighted Shannon model, i.e., the per-node spectral efficiency.

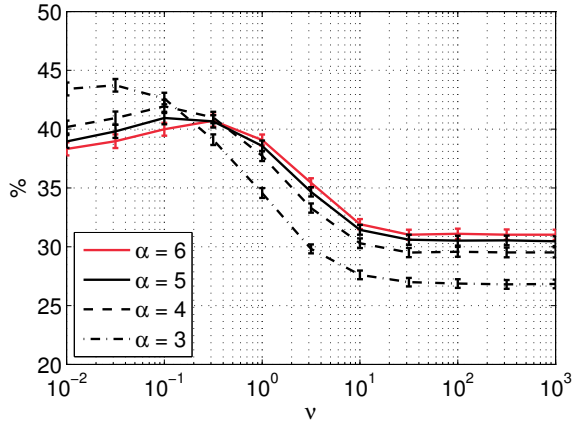


Figure 4.24. The fraction of transmitting nodes under the unweighted Shannon model.

In the noise-limited case, the fraction of transmitting nodes should approach 50 %. The value in Figure 4.24, however, is still less than 45 %. This is mainly due to the fact that the links left out have a negligible weight and are thus easily turned off by the algorithm even in small temperatures, and partly due to the discussed artefact in the simulation.

Figure 4.23 verifies that $u_i^\alpha(\nu)$ is an increasing function of ν. Though also the total capacity of the optimal transmission mode m is increasing function of ν, the mode that realizes the maximum changes as ν increases. This can be seen from Figure 4.24 around the value $\nu \approx 1$. That the fraction of active nodes goes down when ν increases through the transition region is quite understandable. For small ν the interference plays no or only a minor role and it is advantageous to activate as many links as possible, favoring the good ones, i.e., the short ones. When ν increases and

interference becomes appreciable, the trade-off changes and it becomes advantageous to switch off some links in favor of the capacity of some others.

Figures 4.25 and 4.26 show function $u_x^\alpha(\nu)$ for different values of α and the corresponding fraction of transmitting nodes in the optimal transmission mode. The data points of the figures are averages over 100 network realizations with the average of 250 nodes. The 95 % confidence intervals are shown as error bars. Though the interval for ν is shorter, the asymptotic characteristics start to be similar to the unweighted case. The curves are an upper bound for curves $u_x^{\alpha,\theta}(\nu)$ with the same α and any threshold θ. They are also an upper bound for graded SINR threshold model that allows the use of multiple thresholds to better utilize the potential of the links, and illustrates the gain that can be achieved with perfect adaptive coding and modulation.

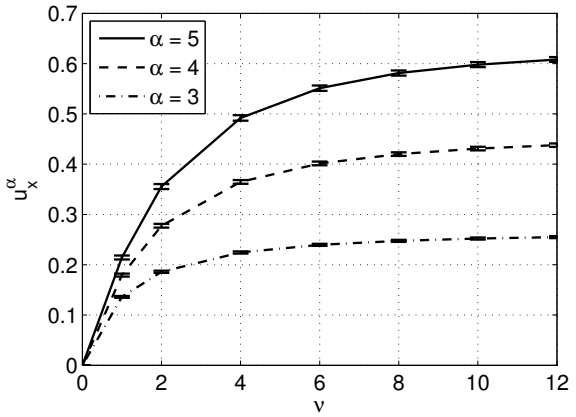


Figure 4.25. The results for the Shannon model weighted by the x-progress of the links.

4.6 Summary of Chapter 4

In this chapter, we obtained an upper bound for the forwarding capacity by studying the maximum amount of traffic that can be forwarded in a single time slot. Under the Boolean interference model, the problem is essentially a maximum weight independent set problem. The SINR-based interference model cannot be described using a simple interference graph, but the problem is still to find the asymptotic weight of the maximum weight independent set of links per unit area when the network domain approaches an infinite plane. With an appropriate weight, the maximum weight independent set of links gives us the IFC, i.e., the above

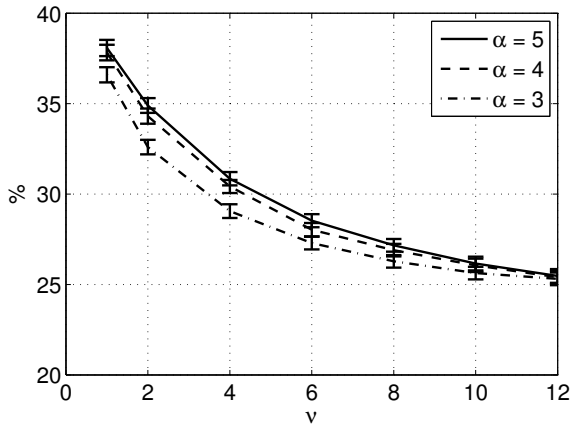


Figure 4.26. The fraction of transmitting nodes under the Shannon model weighted by the x-progress of the links.

mentioned amount of traffic that can be forwarded in a single time slot. The best transmission mode cannot be used all the time with multihop traffic, and its performance is thus an upper bound for the performance with continuous traffic. In the single-directional case, referred to as the x-progress, the density of progress is maximized when a link is weighted by the length of the projection of the link on the x-axis. Respectively, in the multidirectional case, where we have traffic in every direction, the links were weighted by their length. The case weighted by the length of the links provides an upper bound for all the cases with continuous traffic.

In addition to the previous two cases, we studied an unweighted case. This is a problem of stochastic geometry that relates to the question of the largest possible number of simultaneous successful transmissions, i.e., the spatial reuse in wireless multihop networks. In addition to this, the maximum weight independent set problem appears frequently in the context of wireless networks. For example, the maximum weight scheduling algorithm in Section 3.3.1 requires solving a MWIS problem, where the weight of a link is determined by a queue length difference in every time slot. The sub-problem of the column generation algorithm also involves solving a MWIS problem (see Section 3.2.1).

To find numerical values for the IFC, we implemented two simulation algorithms. The moving window algorithm is suitable for the Boolean interference model that has a fixed transmission and interference range. It is able to find the maximum weight independent set of links in a network whose height is limited, but the other dimension can be made arbitrarily large. By studying the results as a function of the network height,

we were able to extrapolate to the infinite plane. The second algorithm, simulated annealing, allowed us to study more complex SINR-based interference models. Because of its probabilistic nature, SA is only able to find approximate solutions. Comparisons with the MWA showed that the results match, and the algorithm is practicable for obtaining numerical values for the IFC. The numerical results were compared to the asymptotic behavior of the IFC.

Later on, we will see how the obtained upper bound compares against the results for the actual forwarding capacity. It is easy to see that the upper bound will be loose when the network is sparse, as a single transmission mode can have a substantial weight before the network even becomes connected. When the network gets denser, the upper bound is bound to get better. As the number of transmission modes grows, a good mode can be selected for every time slot. Dense networks are also the area where bounds are needed as simulating continuous flows becomes computationally challenging.

5. Forwarding capacity

In the previous chapter, we studied how much information can be relayed in a single time slot. Now, we study the same question with multihop traffic, i.e., we attempt to find the microscopic level forwarding capacity (FC) of the network, as defined in Section 3.5.2. The forwarding capacity describes the network's capability to relay information. It is defined as the maximum sustainable mean density of progress, i.e., the density of packets multiplied by their mean velocity in their respective directions. The term single-directional forwarding capacity is used in a situation where all the traffic is flowing in the same direction, in contrast to the multidirectional forwarding capacity that depends on the directional distribution of the traffic.

Finding a coordinated scheme or schedule that forwards the traffic efficiently is referred to as the microscopic level problem in the separation of scales (see Section 3.5). How efficiently the forwarding is done limits the amount of traffic that can be routed through the network. The macroscopic level traffic load cannot exceed the FC at any point of the network. If time sharing between the directions is used, then the single-directional forwarding capacity sets the upper bound. In general, the traffic load is upper bounded by the multidirectional forwarding capacity for the resulting directional distribution. Significant gains compared with time sharing can be achieved by exploiting multidirectional forwarding.

The contents of this chapter are briefly the following. We present three numerical algorithms to study the maximum achievable forwarding capacity. The interference is modeled using the Boolean interference model, and the node locations are assumed to follow a homogeneous PPP. Algorithms based on the LP formulation of the problem are used to obtain bounds. Finally, a more accurate estimate is obtained using GMWS. We conclude the chapter with a discussion on how the results on the micro-

scopic level forwarding capacity can be combined with the macroscopic level routing problem. We also consider the implications of multidirectional forwarding on the solution of the routing problem. The following section further elucidates the forwarding capacity problem and the contents of this chapter.

5.1 Forwarding capacity problem

The multidirectional forwarding capacity describes the capability to forward traffic with a given directional distribution in an infinite network. This sets an upper bound for the allowed macroscopic level load. The local traffic load at any point of the network should not exceed the available capacity. Thus, the task at the microscopic level is to find a coordinated forwarding scheme that is as efficient as possible. The multidirectional forwarding capacity is defined as the maximum sustainable mean density of progress, i.e., the density of packets multiplied by their mean velocity in their respective directions. It depends on the directional distribution of the traffic.

Regardless of the directional distribution, it is always possible to use time sharing between the directions. In this case, traffic is transmitted to a single direction at a time, and the direction is changed between time slots to serve the specific directional distribution. As this can be done with an arbitrary directional distribution, the single-directional forwarding capacity, I_1^* of (3.11), is of special interest.

Significantly higher capacities can be achieved by exploiting the fact that traffic flowing in different directions can be interleaved in the microscopic level scheduling. Our objective is to evaluate the gain from the multidirectional forwarding compared with only single-directional forwarding in certain special cases, namely balanced bidirectional and balanced four-directional cases. The corresponding forwarding capacity is denoted by I_2^* or I_4^*, respectively. We use the word balanced to describe a traffic pattern where the traffic streams are equal in the different directions.

We study the forwarding capacity with the three mentioned traffic patterns. Throughout this chapter we only consider the Boolean interference model and possibly its variant with an adjustable transmission range. The methodology is based on the results presented in Sections 3.2 and 3.3. We specifically apply two LP-based formulations and a greedy version of maximum weight scheduling. This results in a certain setup for the max-

imal flow problem, where the sources and sinks lie at the opposite edges of a large but finite square-shaped network and traffic is relayed through the network. Under our network assumptions, the forwarding capacity depends only on the mean number of neighbors, ν. The achieved forwarding capacity is evaluated as a function of this, and also of the network size to allow extrapolation of the results to an infinite network.

The first LP formulation uses the clique constraints to address the single-directional forwarding capacity. A heuristic polynomial time algorithm is derived for approximating the complete set of constraints resulting from identifying all maximal cliques. The approximation still yields an upper bound for the total achievable single-directional forwarding capacity. The results significantly tighten the previous single-slot results on the upper bounds of the single-directional forwarding capacity with small neighborhood sizes. The clique approximation algorithm is presented and analyzed in Publication II.

The second LP-based approach is to construct an algorithm that searches for a densely packed set of pairwise non-interfering paths. Within these paths, scheduling of the links is trivial, as under the used Boolean interference model and fixed transmission radius, every third link can be active simultaneously in the single-directional case. If there is simultaneous traffic in both directions (balanced bidirectional case), every second link can be activated. The path algorithm is computationally efficient and can be applied to very large network realizations. As it results in an actual schedule, it gives a lower bound for the forwarding capacity. The algorithm is presented and analyzed in Publication IV.

Finally, we apply greedy maximum weight scheduling where we simulate a large network in discrete time and the decision about the resource allocation in each time slot is based on the current queue-length-based weight of each link. It is well known that maximum weight scheduling is able to achieve the capacity limit. This means that if it were possible to realize the scheme ideally, the simulations would result in the actual value for the sought-after forwarding capacity. However, the ideal application of maximum weight scheduling entails solving the maximum weight independent set problem in every time slot, which is infeasible for the network sizes we consider. Hence, we use a greedy algorithm which can be realized more efficiently but, being suboptimal, results in a lower bound for the forwarding capacity. The algorithm is presented in Publication IV that covers the single-directional case both when the transmission radius

is fixed and in the case where it can be reduced (by power control) so as to just reach the destination to minimize the interference. In Publication VIII, we present the multidirectional case for balanced bidirectional and balanced four-directional traffic.

5.2 LP formulations

The linear programming formulation (3.1) of the maximum capacity problem is computationally complex and, in practice, infeasible for large network sizes. By using only a subset of the necessary constraints, it is however possible to obtain upper/lower bounds. In this section, we present two algorithms. The first, clique approximation algorithm, produces an upper bound for the max flow and the corresponding single-directional forwarding capacity. The second, path scheduling algorithm, gives a lower bound. It also gives a separate higher lower bound for balanced bidirectional traffic.

5.2.1 Clique approximation algorithm

Before presenting the algorithm for generating a set of clique constraints, we briefly recapitulate the main ideas behind the clique formulation of the max-flow problem (3.3).

Each link in the network corresponds to a vertex in the interference graph, and two vertices are adjacent if the corresponding links interfere with each other. Vertices in a clique are all pairwise adjacent, and hence only one of the corresponding links can be used at a time. This means that a constraint that limits the capacity of a maximal clique in the interference graph to C is a necessary condition for a feasible flow. By replacing the constraint considering the time shares of the transmission modes with constraints for the clique capacities, we get an upper bound for the maximum value of a flow in the network. These constraints can further be replaced by similar ones considering the flow. If the flow satisfies these, the capacities could always be chosen to match the flow, and the only decision variable is the flow.

Although the method only gives an upper bound, the local nature of cliques makes it usable with large networks. Maximal transmission modes consist of links from all around the network while the links of a clique are

close to each other. Hence, the number of transmission modes can grow much faster than the number of cliques as the network is made bigger.

The problem of finding a maximum clique is NP-hard [GJ79]. It is computationally equivalent to finding a maximum independent set through the concept of complement graph. This means that though the clique formulation of the LP problem is simpler, it is still not usually possible to generate the constraints explicitly for a given large network in a feasible time. Next, we propose a *clique approximation algorithm* (CAA) for reducing the size of the problem.

Algorithm

If the number of constraints (cliques found) in the problem is reduced, the obtained result is still an upper bound but a looser one. We try to limit the size of the problem, and the time required for creating it by using the following algorithm to find a subset of maximal cliques in the interference graph. For the same reason, to limit the size of the problem, we focus on the single-directional forwarding capacity. This means that all the traffic is going to the same direction. As the microscopic level transport network appears to be infinite, the situation is symmetric for an isotropic point process such as the PPP, and the choice of the direction is arbitrary. We assume traffic from left to right. In the simulations we generate this relay traffic by placing artificial sources on the left side of the network and sinks on the right side (see Section 5.4.1).

The algorithm works as follows. For every forward link in the network, i.e., a link that has positive progress in the direction of the packet flow, we create one maximal clique as explained next. In addition to the original link, all the links with the receiving node same as the transmitting node of the original link and all the links with the transmitting node same as the receiving node of the original link are added to the clique (they always interfere with each other since the transmitting node of the original link hears the transmissions from the receiving end). This way we rule out the possibility of any of three consecutive links to transmit at the same time. After this other links are added to the clique in some order until the clique is maximal. The algorithm is also presented in Table 5.1, and its complexity is of the order of $O(|\mathcal{L}|^3)$

Example

Let us consider the following network of 11 nodes and 24 links presented in Figure 5.1. Nodes 1 and 2 serve as sources while 10 and 11 are sinks.

Table 5.1. Clique approximation algorithm

```
0.  i := 0
1.  forall l ∈ L do
2.      if l is a forward link do
3.          i := i + 1
4.          C_i := {l}
5.          forall a ∈ L \ C_i do
6.              if t(a) = r(l) ∨ r(a) = t(l) do
7.                  C_i := C_i ∪ {a}
8.              end if
9.          end for
10.         forall a ∈ L \ C_i do
11.             if forall ã ∈ C_i : {a, ã} ∈ L(I) do
12.                 C_i := C_i ∪ {a}
13.             end if
14.         end for
15.     end if
16. end for
```

The interference graph resulting from the Boolean interference model is depicted in the left subfigure of Figure 5.2. Note that only links with positive progress (receiving node closer to the sinks) are drawn to make the figure more readable. The nominal link capacity is normalized to be equal to one ($C = 1$).

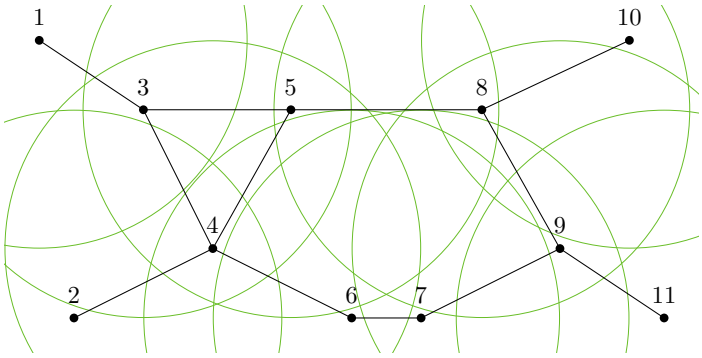


Figure 5.1. The example network of 11 nodes.

A schedule resulting in a maximum flow in the network is represented in Table 5.2. The value of the flow is 7/12. This consists of the capacity of the upper path, 1/3, and the capacity of the lower path which is 1/4. The clique

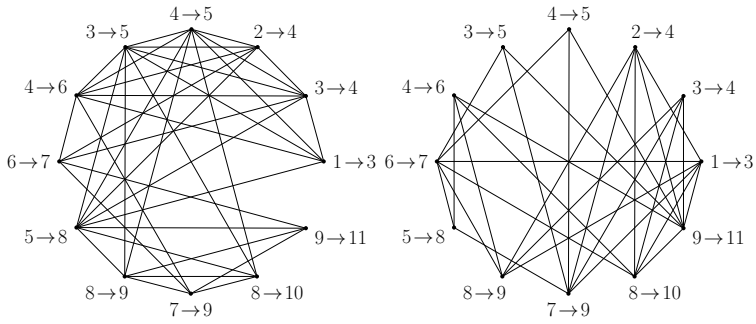


Figure 5.2. The iterference graph and its complement when only forward links are drawn.

Table 5.2. Schedule

#	transmission mode				timeshare
1.	$1 \to 3$	$2 \to 4$	$8 \to 10$	$9 \to 11$	$1/4$
2.	$3 \to 5$	$7 \to 9$			$1/4$
3.	$4 \to 6$	$5 \to 8$			$1/4$
4.	$1 \to 3$	$6 \to 7$	$8 \to 10$		$1/12$
5.	$3 \to 5$	$6 \to 7$			$1/12$
6.	$6 \to 7$	$5 \to 8$			$1/12$

formulation (3.3) gives a higher optimum of 2/3. This means that the optimality gap due to the relaxation is 1/12. The same result is obtained using only the 12 cliques (10 different) produced by the algorithm of Table 5.1. The total number of maximal cliques is 27, and there exists a total of 74 (maximal) transmission modes.

The results for the original problem with transmission modes and for the clique formulation of the problem start to differ when the mean node degree is large enough so that the “paths”, used to transfer packets from one side of the network to another, begin to interfere with each other. This can be seen in the example where the upper and lower paths are clearly visible. The clique approximation algorithm is still able to produce the same result with the full clique formulation as the number of interfering paths is only two. When the number of interfering paths grows, the algorithm is only able to describe the interference caused by a single clique in the interference graph per forward link. If the interfering paths do not share a common clique then some of the pairwise dependencies might be missed.

5.2.2 Path scheduling algorithm

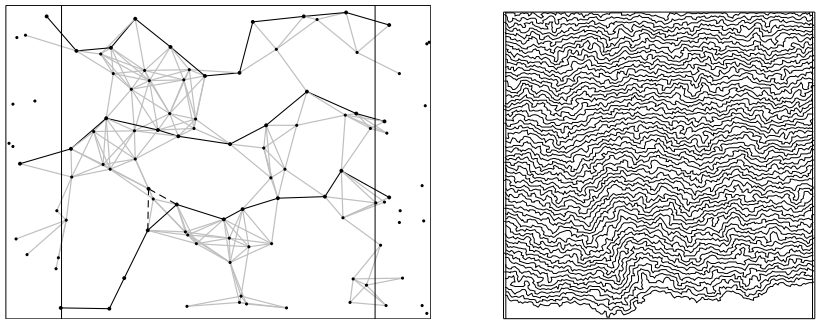
As the clique formulation gives an upper bound for the feasible flow, it does not result in an actual schedule (as such schedule usually does not exist). Now, we approach the problem from the opposite direction and try to find a schedule that is easy to construct yet reasonably efficient. In this section, we construct an algorithm for finding a densely packed set of paths whose links do not interfere with the links of other paths and that connect the sources and the sinks on the opposite sides of the network. Again, we study the microscopic level relay traffic that has been generated to the relay network by artificial sources on different sides of the network. We utilize this idea to obtain a lower bound for the forwarding capacity. The used methods bear similarities to those of geographic routing [Sto02].

By identifying non-interfering paths, scheduling under the Boolean interference model becomes trivial. When the paths are chosen in the way that, in addition to the previous and following link, a transmission only interferes the reception of the link preceding the previous one, every third link in a path can be activated if all the active links point towards the sink. Thus, it is possible to schedule a (single-directional) flow of $C/3$ for a single path. This can be done with only three transmission modes. Path scheduling also clearly illustrates the benefit of multidirectional forwarding. With bidirectional traffic, every other link on a path may be activated, as the direction of the active links alternates. Using four transmission modes, it is possible to schedule a flow of $C/2$ for a single path. With respect to the classical max flow problem, this means that we fix the schedule to consist of the three or four transmission modes. As a result the capacity of the links that are not between the nodes of a path is zero. The operation of the *path scheduling algorithm* (PSA) is presented next.

Algorithm

The transport network with the average of N nodes resides in a unit square that has a strip of sources added to the left side and another strip of sinks added to the right side. The width of the added strips is ρ since it is not possible to form links with the nodes of the network from farther away, see Figure 5.3a.

The algorithm starts from the uppermost source (towards right) and then chooses the leftmost neighbor to be the next hop. The leftmost neighbor is the neighbor that forms the largest angle ($[-\pi, \pi]$) with respect to the current direction of the path. Note, that the neighbors on the right



(a) Paths in a small network **(b)** Paths in a large network

Figure 5.3. Examples of the operation of the path search algorithm.

have negative angles. The leftmost neighbor is chosen with one exception: the path is not allowed to cross itself, and such neighbors are ignored. The selection of the next hop is illustrated in Figure 5.4. In the figure, we have a path traversing through nodes 1 and 2 to node 3 that has four neighbors (A, B, C, and D) in addition to the used node 2. Continuation to node A forms the largest angle with respect to the current direction of the path (marked with an arrow), but would cause the path to cross itself. Instead, node B is chosen because of the second largest angle. Note that the angle created with node D is negative.

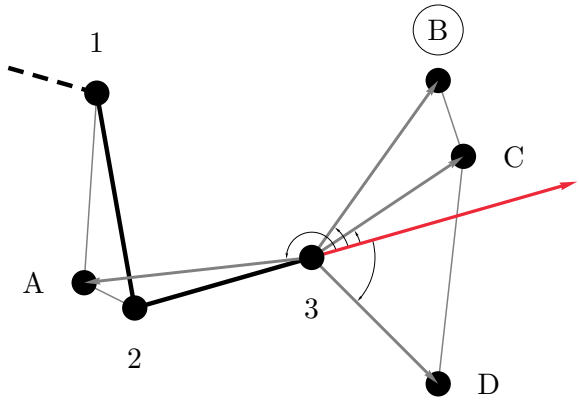


Figure 5.4. Next-hop selection using the path search algorithm.

The algorithm continues the path until it reaches an exception. The handling of the exceptions is done in the following way:

- If the path comes to a dead end (no available neighbors), the algorithm returns to the previous node.

- If the algorithm returns to the source, no path from this source can be found. The algorithm continues from the next source.

- If the algorithm reaches another source, then no path from this source and the nodes along the way can be found and they can be ignored later. The algorithm continues from the next source.

- If the algorithm reaches a sink, we have found a path and continue the algorithm from the next source. The nodes belonging to the path and their neighbors are ignored later.

The search is continued until all the sources have been gone through.

The paths that the algorithm finds do not interfere with each other, but it is still possible that the links of the path interfere with other links of the same path in a way that it is not possible to schedule a $C/3$ flow to the path. Nevertheless, it is always possible to pick a subset of the nodes from the original path to form the interference-free path. This can be done by starting from the sink and always choosing the neighbor that has the smallest index in the path to be the previous node in the interference-free path. Now, the formed links do not interfere with each other since the supposedly interfering node would transmit directly to the receiver instead of interfering it. The phenomenon can be seen in the lowest path of Figure 5.3a, which illustrates the set of paths for a small network, where the algorithm advances using the dashed lines, but the packets need to be transmitted along the solid line.

Network size

It is easier to connect the sources to the sinks through a small network. For example, in a finite network there is always a positive probability for a non-zero flow, while in an infinite network a positive flow becomes possible only above the percolation threshold ($\nu \approx 4.5$). Thus, the simulations should be done in a network with a sufficient number of nodes for the results to be meaningful when considering the infinite network. When the simulations are conducted in a finite network, there are two border effects causing an error to the quantity of interest, the density of the paths in the vertical direction, and hence to the achievable forwarding capacity.

The horizontal effect depends on the length (in hops) of the paths. If the distance between the sources and the sinks is only a few hops, the

existence of the paths is more probable. When the number of nodes in the network, N, is increased, the capacity goes down. The vertical effect is caused by the fact that the top and the bottom of the network confine the paths to a limited area. Thus, the behavior of the paths is different near the borders, and the interesting quantity, the number of paths per height unit, differs from that in the middle of the network. This can be seen in Figure 5.3b. When the size of the network is increased, the relative area that is wasted due to the artificial limitations decreases, and the capacity goes up.

As can be seen from Figure 5.5 (solid lines), horizontal effect, dominating with small networks, dies out faster than the vertical effect. Thus, the capacity first goes down rapidly and then slowly starts to increase. The combined effect of the two is that $u(N)$ approaches the asymptotic value from below. It is also possible to calculate the number of paths originating from the middle part of the network, e.g., by adding some extra space on top of and below the network residing in the unit square. In this case, the capacity approaches the asymptotic value from above as can be seen from Figure 5.5 (dashed lines).

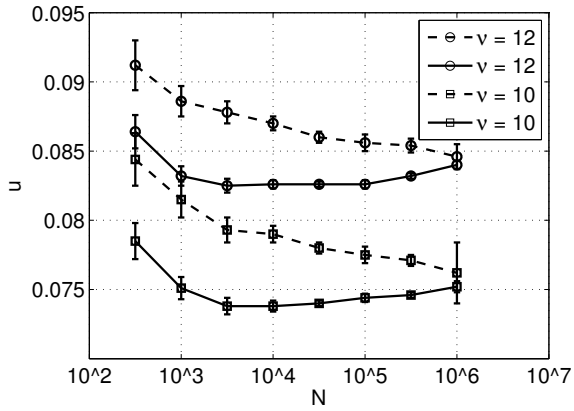


Figure 5.5. The effect of network size on u using the path search algorithm (single-directional traffic).

5.3 Greedy maximum weight scheduling

In this section, we describe a method for resolving the problems related to finding a feasible way to schedule the transmissions efficiently. It is based on the maximum throughput policy presented in Section 3.3.1. While being suboptimal, greedy maximum weight scheduling offers a practical ap-

proach to multiclass traffic from the network simulation point of view. The different traffic classes can be used to represent traffic flows in different directions. This allows us to study the gain from the multidirectional forwarding compared with only single-directional forwarding in certain special cases. The covered cases include two opposite directions with equal flows, $f(\vartheta) = (\delta(\vartheta) + \delta(\vartheta - \pi))/2$, and the four cardinal directions with balanced flows, $f(\vartheta) = (\delta(\vartheta) + \delta(\vartheta - \pi/2) + \delta(\vartheta - \pi) + \delta(\vartheta - 3\pi/2))/4$. We will find that the corresponding limits I_2^* and I_4^* indeed can be considerably greater than I_1^* depending on the network parameters. We also modify the basic greedy maximum weight scheduling algorithm to be more suitable for traffic flowing in the four cardinal directions.

5.3.1 Original algorithm

The maximum throughput policy that stabilizes the network for all arrival rates of multiclass traffic for which it is stabilizable [TE92] was presented in Section 3.3.1. In our setting, the customer classes of the algorithm correspond to the traffic flows in different directions. In the simulations, the classes are discrete. Generally, the traffic in the direction increment $(\vartheta, \vartheta + \mathrm{d}\vartheta)$ is equivalent to a class. The original algorithm also allows multiple link capacities, but here all the links are assumed to have the same nominal capacity.

The algorithm has three stages. In time slot t, the first stage is to calculate a weight w_l^t for each link $l \in \mathcal{L}$ as using (3.6). In the second stage, a maximum weight transmission mode is selected (3.7). Finally, in the third and last stage, we get the information about the activated links in binary form, $E(t)$, at time slot t from (3.8).

In each time slot the policy finds the transmission mode that is of the maximum weight. Since all the links have the same nominal capacity, the weight of each link is simply the maximum (over the classes) difference in the queue lengths between the transmitting and receiving end of the link. A link in the maximum weight transmission mode is always activated if the transmitter has a packet to send.

In [TE92], the packets arrive to the source nodes at certain rates. At the beginning, the number of packets in the network is small, and a notable number of links may have the same weight. As there is no way to differentiate between transmission modes of maximum weight, the scheduling decision is more random. This leads to suboptimal performance, and the queue lengths increase. As the queues grow, some of the ties are resolved,

and eventually the scheduling policy stabilizes the system whenever possible. This might take a long time.

We use a fixed queue length, q_0, at the sources. This, in a way, sets the network directly at the stability limit. It also fixes the set of possible link weights and deteriorates the performance if the level is too low. The choice of q_0 is eventually a compromise. Long queues reduce the probability that the weights of two links are equal but require a longer initial transient in the simulations.

5.3.2 Basic greedy algorithm

If it were possible to run the maximum weight algorithm properly, we could find out the true forwarding capacity. Unfortunately, finding the maximum weight transmission mode (3.7) is NP-complete. Because of the limitations in the computing capacity and the large network size, it is necessary to simplify the problem. Therefore, we choose the links of the transmission mode in a greedy fashion.

In the basic greedy algorithm, the used transmission mode is chosen in each time slot in the following fashion. The links are sorted in the descending order according to their weigth (3.6). The heaviest link is added to the transmission mode, and all the links interfering with it are removed from the list of possible links. The next heaviest link (that does not interfere with first one) is then added to the transmission mode. Again, the links interfering with the added link are removed from the list. The procedure is continued until there are no links left to add to the transmission mode, i.e., the mode is maximal.

The basic greedy algorithm works well in all the cases when ν is small. When ν is very large, however, better results are obtained in the case of traffic in four cardinal directions by making time sharing, with equal shares, between the two pairs of opposite streams and applying the greedy algorithm for one pair in turn. This behavior can be understood as follows. According to Section 4.2.1, if links can be arbitrarily placed on a continuous plane, then the most efficient way of packing them for carrying traffic in the horizontal direction is to form vertical columns of horizontal links. The Boolean interference model sets no limit on how close two parallel links of maximal length, ρ, can be. When the direction of the columns alternates, the endpoints close to each other are all either transmitters or receivers and a small distance ε is enough between the columns.

The above suggests that for a high ν, when there are nodes almost everywhere, a good strategy is to try to form vertical columns. Since the transport network is very dense, the same kind of transmission mode can be used in every time slot. The configuration is just rotated and/or moved by ε resulting in a flow where the same packet is moved once in every ρ/ε time slots allocated for this pair of opposite directions. Thus, packets traversing in two opposite directions are transmitted in a single time slot, and different directions are handled via time sharing.

In order to have a single algorithm that covers all values of ν in the case of equal traffic streams in the four cardinal directions, we developed an improved algorithm described in detail in the next subsection.

5.3.3 Improved greedy algorithm

In the basic greedy algorithm, as described, the links are ordered in a list according to their weights. The links are chosen from this list starting from the link with greatest weight and skipping links that are in conflict with some of the already chosen links.

In the improved algorithm we introduce two parameters, one discrete, $k = 1, 2$, and one continuous, $\beta \in [0, 1]$. For given values of k and β, the algorithm works as follows. First, the greatest weight of all the links, $w_{\max}$, is found. Then, priority is given to the pair k of opposite streams, $k = 1$ corresponds to the left-right pair, and $k = 2$ corresponds to the up-down pair. Weights (3.6) are calculated taking into account only the two customer classes in the preferred pair of directions. Links are ordered according to these weights and conflict-free links are chosen as in the basic greedy algorithm from this list, starting from the link with the greatest weight, as long as the weight exceeds the value $\beta \cdot w_{\max}$. After this point, the weights (3.6) are calculated for all the remaining links taking into account all four customer classes and links are chosen as in the basic greedy algorithm.

For each setting of the parameters (k, β) we get a different algorithm yielding a different set of selected links with associated total weight. In principle, we could make an optimization to find the parameters that give the highest total weight in each time slot, to become as close as possible to the max weight scheduling. However, making such an optimization in every time slot for the continuous parameter β would be too time consuming. Therefore, in the algorithm we use a fixed value of β, the same in all time slots. This value is, however, optimized externally to give maximal

flow for a given ν. Optimization over the two values of k is done explicitly in each time slot, i.e., both values are tried, and the one that gives the higher total weight is selected.

Note that when $\beta = 1$ the algorithm reduces to the basic greedy algorithm. One can also see that when $\beta = 0$ the result cannot be worse than time sharing between the two pairs of opposite directions. Hence, with the optimized k and β, we always get results that are at least as good as those of the two simpler algorithms.

5.4 Numerical results for FC

This section presents the numerical results for the forwarding capacity obtained using the algorithms of this chapter. We begin with the two LP-based algorithms and continue with greedy maximum weight scheduling.

5.4.1 Results of CAA

At the microscopic level the network appears as an infinite one. The idea is to approximate this network with a finite, but large, one where the relay traffic flows, e.g., from left to right. A unit square with the average of N nodes distributed according to a spatial Poisson process represents the relay network, and a strip of sources is laid on the left side of the square and a similar strip of sinks on the right. One transmission range ρ is a sufficient width for the strips since no longer links can be established. Finally, the top and the bottom of the network are connected to reduce harmful border effects. An example network with $\nu = 10$ and $N = 1000$ is depicted in Fig. 5.6. After solving the optimization problem (3.3) with the reduced set of clique constraints from the algorithm of Table 5.1, we get an upper bound for the maximum flow from left to right (note that this is a single-directional flow). Since the network resides in a unit square, we have $u = w(f)/\sqrt{N}$.

To properly simulate the operation of an infinite network, N needs to be as large as possible. Due to the computational limitations, it is not possible to solve the problem for sufficiently large networks when the mean degree of a node, ν, is large. Because it is easier to establish a flow through shorter paths, the price to pay for a smaller network is that we get a looser upper bound. The border effect caused by the left and the right sides of the network, where the sources and sinks are located, is illustrated in Fig.

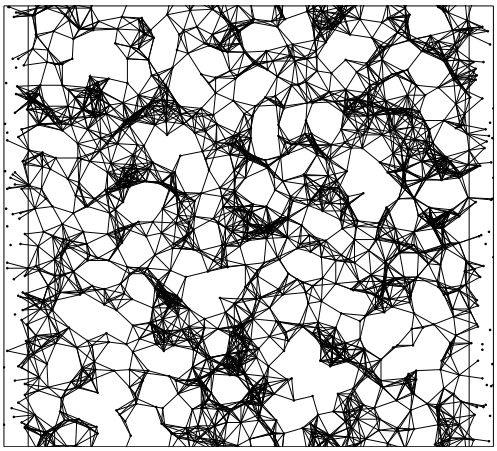


Figure 5.6. Transport network with $\nu = 10$, $N = 1000$.

5.7, which shows u as a function of the system size N for $\nu = 7$. The figure also shows the rate at which the network starts to resemble an infinite network.

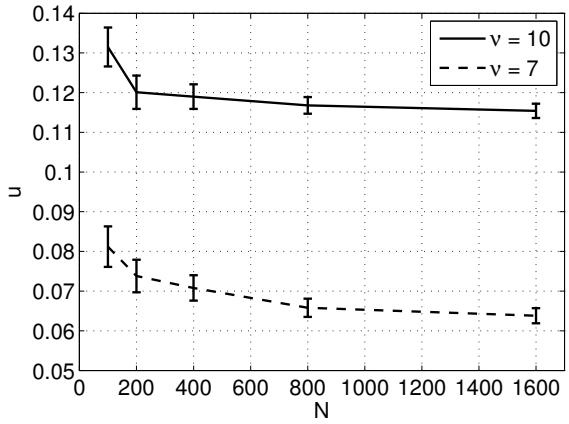


Figure 5.7. The effect of the network size on u of I_1^*, when using CAA.

Fig. 5.8 represents u as a function of ν obtained using the CAA with $N = 500$ as well as the 95% confidence intervals. The figure also depicts the maximum capacity achievable in one time slot (see Section 4.5) and a result from an actual forwarding method, namely the opportunistic forwarding method [ALV06, Nou07], that is a variant of ExOR [BM05]. As can be seen from the figure, the upper bound provided by the LP approximation is much tighter than the IFC, the one-slot maximum, when the mean degree of a node is small. On the other hand, when ν increases even the relaxed LP problem becomes too complex, and the results of Chapter 4 remain as the only available upper bounds.

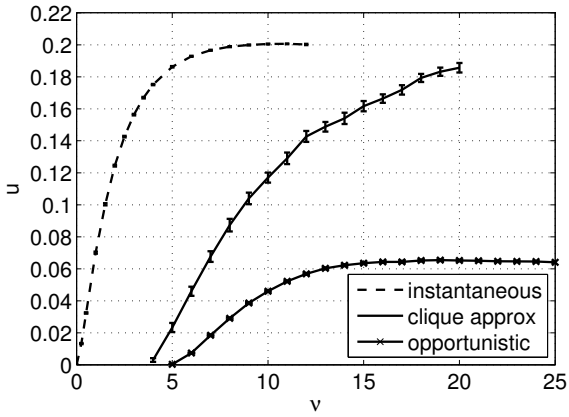


Figure 5.8. The dimensionless function $u(\nu)$ of I_1^* with 95 % confidense intervals.

We finally comment on the relationship between the IFC and the achievable flow for large ν. The maximum total progress in one time slot gives an upper bound for the total progress achievable with feasible flows. This is intuitive since in a continuous flow the best possible progress cannot be achieved in every time slot. When the mean degree of a node is relatively small, the upper bound should be loose. For example, just before the network becomes connected the feasible flow is still zero, but there are plenty of links that can contribute to the one slot maximum that is quite high. When the mean number of neighbors increases, the one-slot approximation becomes better since the difference to the bottleneck-slot limiting the feasible flows decreases. This is due to the idea that when the network is very dense, it is always possible to use a relatively good transmission mode.

5.4.2 Results of PSA

The simulation setup for the PSA is as follows: The nodes are distributed to a rectangular network area with width $1 + 2\rho$ and a varying height according to a PPP with density N. The nodes within one transmission radius from the left side of the network are assigned the role of a source and the ones within ρ from the right side of the network are sinks. The relay network is a unit square with the average of N nodes in the middle of the network. When N is large enough the results meaningfully represent the infinite network that appears when a massively dense network is viewed from the perspective of a single node.

The path search algorithm gives, as a result, the number of independent paths connecting the sources to the sinks. With the used network height of 3, the paths beginning from a source with the y-coordinate between 1 and 2 were used as an estimate for the vertical path density $\bar{n}_{\text{paths}}$. From this, we can calculate the dimensionless component of I_1^*, $u = \bar{n}_{\text{paths}}/(3\sqrt{N})$.

Figure 5.9 presents u as a function of ν near the optimum size of the neighborhood (see also Fig. 5.10 for wider range) with fixed transmission radius. The maximum of $u = 0.09$ and occurs at $\nu = 18$. The results have been obtained with $N = 10^5$, and are averages over 100 network realizations. The errorbars show the 95 % confidence intervals.

The advantage of the algorithm is that it is computationally efficient, and the simulations can be done with a very large network. This is important when the network appears heterogeneous from the point of view of a single node, i.e., ν is small. Thus, the results of path scheduling for the smallest values of ν are very accurate. The paths connecting the sources to the sinks occur so rarely that routing is only possible along these paths, and different paths do not interfere with each other.

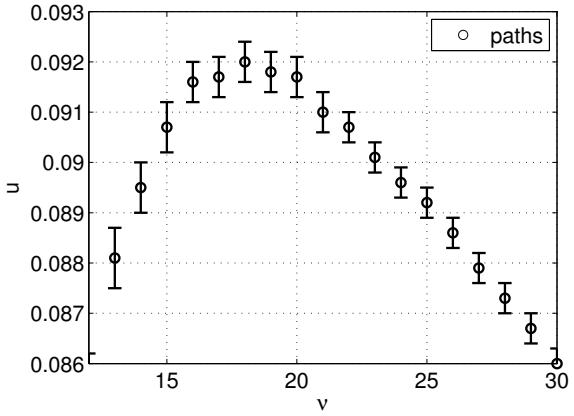


Figure 5.9. The dimensionless component u of I_1^* as a function of the mean neighborhood size using path search algorithm.

5.4.3 Results of single-directional GMWS

The simulation setup with the GMWS differs from the one of the path scheduling in the way that the top and the bottom of the network of unit height are connected together to form a cylinder. This is done to reduce the border effects in the vertical direction. The greedy maximum weight scheduling algorithm gives as a result the number of packets per slot arriving to the sinks, and u can be obtained by dividing it by $\sqrt{N}$. Alterna-

tively, it is possible to monitor the number of packets leaving the sources or the progress of the packets in the horizontal direction. When these three quantities are approximately the same, the simulation can be considered to have reached steady state.

Figure 5.10 presents the results obtained from the greedy maximum weight scheduling compared to other bounds for the forwarding capacity with fixed transmission radius. The results have been obtained with $q_0 = 100$ and $N = 1000$. They are averages over 10 network realizations and the errorbars show the 95 % confidence intervals. The values of q_0 and N are a practical compromise between the accuracy and the necessity to keep the simulation times reasonable. The maximum, $u = 0.13$, occurs at $\nu = 18$. Since the studied methods are not particularly good at coordinating the transmission when the mean degree of a node is large, the true optimum size for ν is likely to be higher.

When ν is very small, the difference between the results of greedy maximum weight scheduling and path scheduling is due to the network size. The one third flow that the scheduling is able to achieve along a single path is maximal so the results do not suffer from the greedy heuristic with the smallest neighborhoods. The smaller network used with greedy maximum weight scheduling leads to too high capacity as there is more likely to be better connectivity through a small network.

When ν is larger, the effect of the network size is less relevant since the network is more homogeneous from the scheduling point of view. That is to say, there are always multiple possible links to choose from, and no clear bottlenecks appear as the network is made larger. The capacities are achievable since the meaningful effects, the greedy scheduling and the queue length at sources, both move the result downwards. Greedy scheduling is not able to coordinate the transmissions efficiently enough when the number of interfering links is large, and, in addition to the weight, one should also consider how the links interfere with each other. Also, the queue length at sources should be made larger when the number of links grows to allow the weights of the links to be separable.

Power control

In addition to a fixed transmission radius, we consider greedy maximum weight scheduling in a case where the transmission radius can be reduced from the maximum value ρ to the length of link l. The idea is that the transmitting nodes use a radius just large enough to reach the receiving

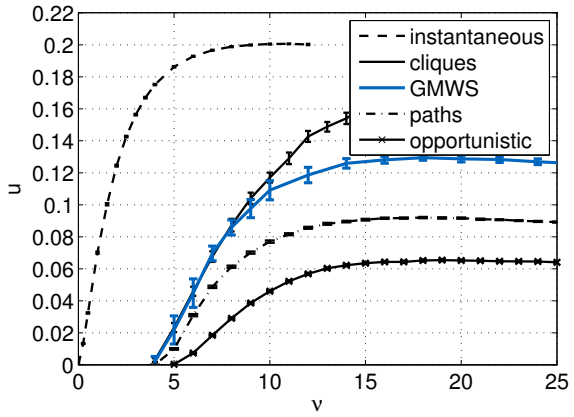


Figure 5.10. Comparison of the results for single-directional traffic. GMWS gives an estimate, while the two topmost curves are upper and the two undermost curves are lower bounds.

node and thus minimize the interference. In this case, the parameter ν corresponds to the mean number of neighbors with the maximum radius.

Figure 5.11 shows $u(\nu)$ for greedy maximum weight scheduling with both fixed and adjustable transmission radius. With power control, it is possible to activate more links simultaneously, and the maximum gain from a freely adjustable transmission radius is a little over 50 %.

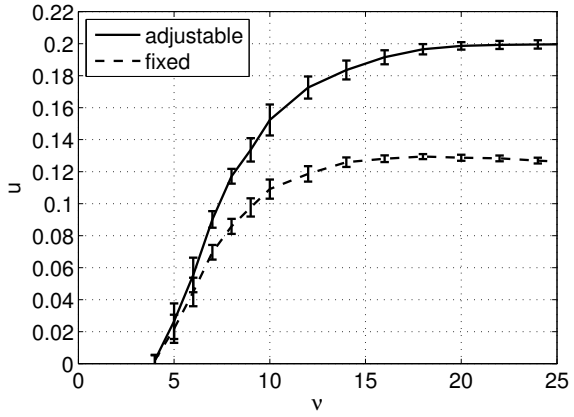


Figure 5.11. Function $u(\nu)$ for greedy maximum weight scheduling with and without power control.

5.4.4 Results of multidirectional GMWS

This section contains the numerical results for the studied two- and four directional balanced cases, while the discussion on their significance on

more general directional distributions as well as on the macroscopic level problem is in Sections 5.5 and 5.6.

Improved greedy algorithm

The performance of the improved greedy algorithm as a function of the priority threshold parameter β with $q_0 = 100$ is presented in Figure 5.12. The figure illustrates how the optimal value of β becomes smaller as the transport network density, ν, increases. The data points are averages over 10 network realizations and the error bars show the 95 % confidence intervals. The maximum values of each curve, corresponding to the optimization over β, form the capacity curve of the improved algorithm. This curve is represented in Figure 5.13 along with the results from the basic greedy algorithm with two and four directions. The figure shows clearly how the improved algorithm is able to outperform the basic greedy algorithm with four directions when ν is large, achieving the level of the two-directional one, but is still able to utilize the multidirectional gain from all of the four directions when ν is smaller.

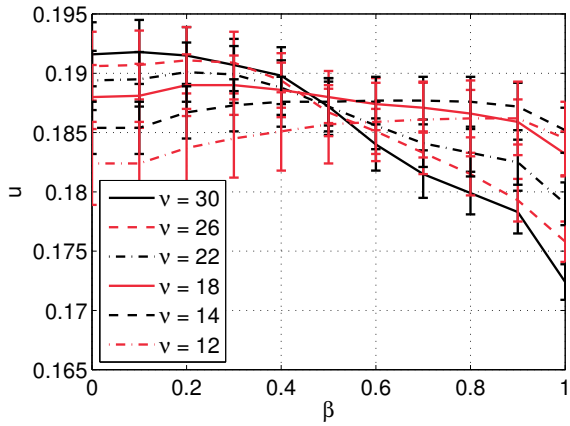


Figure 5.12. The results from the improved greedy algorithm as a function of the threshold β.

Comparison with the single-directional capacity

Figure 5.14 presents the dimensionless components, $u(\nu)$ in (3.12), of I_1^*, I_2^*, and I_4^* obtained using $N = 1000$ and $q_0 = 100$ from the basic greedy maximum weight scheduling algorithm with one and two directions and the improved greedy maximum weight scheduling algorithm with four directions. The results are averages over 10 network realizations and the error bars show the 95 % confidence intervals. The values of q_0 and N are

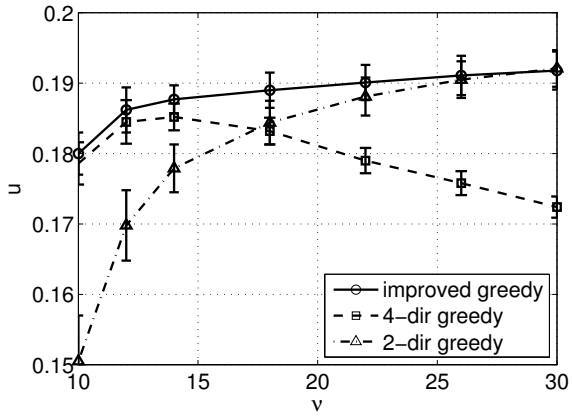


Figure 5.13. The improved algorithm for traffic in four directions compared with the basic greedy algorithm with two and four directions.

a practical compromise, based on Figure 5.7, between the accuracy and the necessity to keep the simulation times reasonable.

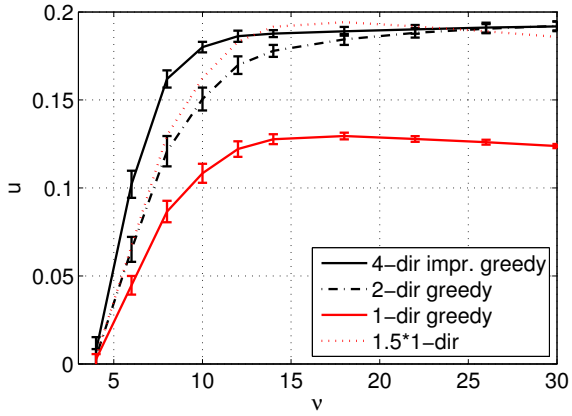


Figure 5.14. Multidirectional forwarding capacity for traffic in one, two and four directions.

As can be seen from the figure, the multidirectional gain is indeed considerable, ranging from a factor of over 2 in a sparse transport network to around 1.5 in a dense transport network. After the initial drop the multidirectional gain is increasing. This is further illustrated in Figure 5.15. Traffic in two opposite directions is enough to generate the gain in a dense transport network, as discussed in Section 4.2.1, but having the four-directional distribution is beneficial in a sparse transport network. In general, the possibility to use more directions can only improve the result as it is always possible to regress back to using only a subset of the directions in a time-shared manner. It is safe to presume that the four cardinal

directions with equal flows should already be a good approximation for the isotropic traffic.

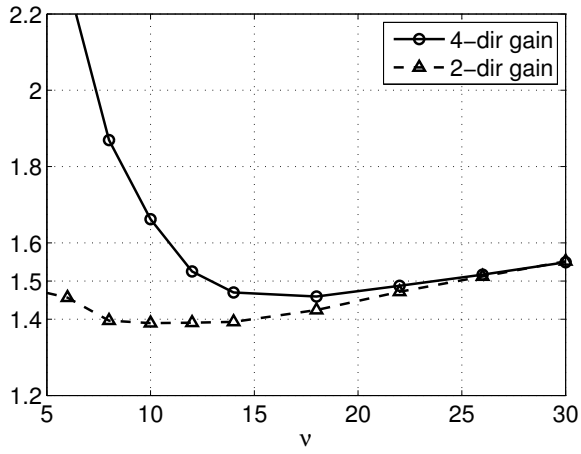


Figure 5.15. The gain from multidirectional forwarding compared with single-directional forwarding.

The forwarding capacity with two opposite directions is close to 1.5 times the single-directional when the network is sparse. This can be understood as follows. When the transport network is operated close to the percolation threshold ($\nu \approx 4.5$), the number of paths connecting different sides of the network is small. The small number of paths also implies that they are far from each other and the interference between two paths is negligible. When the schedule is chosen in the way that the links of the path only interfere with the previous and following link of the path, it is possible to use every third link of the path simultaneously with a single direction. This idea has been utilized in Section 5.2.2 to construct a lower bound for the forwarding capacity also when the mean neighborhood size, ν, is larger. In the multidirectional case, it is possible to activate more than every third link. With two opposite directions, every other link can be active when the directions of the transmissions alternate. A single-directional schedule consisting of three transmission modes leads to capacity $C/3$ for a single path while a two-directional schedule consisting of four transmission modes leads to capacity $C/2$ for a path. Hence, the two-directional case gives 1.5 times the single-directional capacity when the transport network is sparse.

5.4.5 On the accuracy of GMWS

For a very small ν, the error in the simulations is mainly due to the network size, N. The $C/3$ flow that the one-directional scheduling is able to achieve along a single path is maximal. Also the two-directional greedy scheme can schedule close to the maximum flow of one half along a path. Thus, the results do not suffer from the greedy heuristic with the smallest neighborhoods. The used network size of 1000 nodes leads to a slightly too high capacity as there is more likely to be better connectivity through a small network.

When ν is large, the effect of N is less relevant since the network is more homogeneous from the scheduling point of view. That is to say, there are always multiple possible links to choose from, and no clear bottlenecks appear as the network is made larger. The capacities are achievable since the greedy scheduling moves the result downwards. Greedy scheduling is not able to coordinate the transmissions efficiently enough when the number of interfering links is large, and, in addition to the weight, one should also consider how the links interfere with each other.

This is manifested in Figure 5.13 in that the greedy scheduling performs worse with four directions than with two when the network is dense. As concluded in Section 4.2.1, the optimal configuration for a dense network uses only two opposite directions at a time. The two-directional greedy scheduling automatically selects transmission modes that have this characteristic, and it is thus able to outperform the four-directional greedy method that cannot achieve the same spatial reuse. The improved greedy algorithm is able to rectify this for four-directional traffic, but it is still unable to coordinate the transmissions further, e.g., to form columns.

5.5 General directional distribution

Now, we consider what can be said about $I^*[f(\vartheta)]$ with a general directional distribution $f(\vartheta)$. Assume that the multidirectional forwarding capacities are known for some set of directional distributions, denoted by $\{g_1(\vartheta), g_2(\vartheta), \ldots\}$, and let these capacities be $\{J_1^*, J_2^*, \ldots\}$, respectively. We can always express the distribution $f(\vartheta)$ in the form

$$f(\vartheta) = \sum_i a_i g_i(\vartheta) + b\, h(\vartheta), \tag{5.1}$$

where the a_i and b are non-negative constants, and the remainder term $h(\vartheta) \geq 0$ for all ϑ. Note that also $h(\vartheta)$ represents a distribution (with the integral over the angle equaling one, and hence, $\sum_i a_i + b = 1$).

Given $f(\vartheta)$, we try to determine the maximal scalar flux Φ, i.e., the constant multiplier in front of $f(\vartheta)$, such that the traffic can be sustained. Each of the components of the sum in (5.1) can be handled in $\Phi \cdot a_i/J_i^*$ fraction of time. The remainder requires a fraction smaller than or equal to $\Phi \cdot b/I_1^*$, where I_1^* is the single-directional forwarding capacity. The total traffic can be sustained using time sharing between the components if the sum of the time shares is at most 1. Then, for this $f(\vartheta)$ we have the following lower bound for the multidirectional forwarding capacity

$$I^*[f(\vartheta)] \geq \Big(\sum_i a_i/J_i^* + b/I_1^* \Big)^{-1}.$$

This is a sure lower bound since the handling of the remainder term is upper bounded (by only using time-shared single-directional forwarding) and also since time-sharing among the components i is not necessarily optimal.

In our case, the directional distributions for which the multidirectional forwarding capacity is known are the single-, bi- and four-directional balanced traffic patterns. These can be utilized to forward non-balanced four-directional traffic by first separating the four-directional balanced traffic pattern, in which case the remaining traffic equals zero in at least one direction. In the other orthogonal direction, the balanced bidirectional traffic can again be extracted. This only leaves two single-directional orthogonal flows that can be handled using time sharing with single-directional forwarding. This yields the lower bound. By rotating this pattern over all angles $(0, \pi/2)$, a lower bound is obtained for $I^*[f(\vartheta)]$ for any directional distribution. Explicitly, we have

$$I^*[f(\vartheta)] \geq \left(\frac{K_1 - K_2}{I_1^*} + \frac{K_2 - K_4}{I_2^*} + \frac{K_4}{I_4^*} \right)^{-1},$$

where

$$\begin{cases} K_1 = \int_0^{2\pi} f(\vartheta)\, d\vartheta = 1, \\ K_2 = 2 \int_0^{\pi} \min\{f(\vartheta), f(\vartheta+\pi)\}\, d\vartheta, \\ K_4 = 4 \int_0^{\pi/2} \min\{f(\vartheta), f(\vartheta+\tfrac{\pi}{2}), f(\vartheta+\pi), f(\vartheta+\tfrac{3\pi}{2})\}\, d\vartheta. \end{cases}$$

In the next section, we conclude this chapter by examining how the results for the microscopic level forwarding problem can be combined with the macroscopic level problem.

5.6 Combining with the macroscopic level

Finally, we make a remark on the impact of multidirectional forwarding in the macroscopic level routing problem with a uniform traffic matrix. Under the assumption that only single directional forwarding is used at the microscopic level, the macroscopic-level routing problem is to determine the routes so that the maximum local load is minimized, i.e., a problem of load balancing, see [HV09, PRK$^+$07]. This causes the routes to be long so that traffic is pushed away from the center towards the edges, in order to avoid congesting the center of the area. Now, the multidirectional forwarding capacity increases the capacity compared with single directional forwarding, especially when the traffic is nearly isotropic. Because the traffic is naturally more isotropic at the center of the area (e.g., a disk) than at the border, there is less need to push the traffic away from the central area, and the optimal paths under multidirectional forwarding will be straighter. As long as the paths in $\mathcal{P}$ are bidirectional, i.e., the traffic between two locations uses the same route in both directions, the traffic is also bidirectional. This means that a considerable gain compared with single-directional traffic can always be achieved with uniform traffic matrix. We illustrate this with a network on a unit disk. The macroscopic level routes are not optimized, instead we simply use sortest path routing.

As a numerical example, we consider a circular disk with area A and uniform traffic demand:

$$\mathcal{A} = \left\{ \boldsymbol{x} \in \mathbb{R}^2 \mid |\boldsymbol{x}| < \sqrt{A/\pi} \right\}, \quad \lambda(\boldsymbol{x}_1, \boldsymbol{x}_2) = \Lambda/A^2.$$

The angular flux at $\boldsymbol{x}$ with shortest path routes is [HV09]:

$$\varphi_{\mathrm{SP}}(\boldsymbol{x}, \vartheta) = \frac{\Lambda}{2A^2} a_1 a_2 (a_1 + a_2),$$

where $a_1 = a_1(\boldsymbol{x}, \vartheta)$ denotes the distance to the boundary from $\boldsymbol{x}$ in direction ϑ, and a_2 in the opposite direction, $a_2(\boldsymbol{x}, \vartheta) = a_1(\boldsymbol{x}, \vartheta + \pi)$.

The following figures illustrate the quantities when the radius of the disk has been normalized to one ($A = \pi$). Figure 5.16 represents the angular flux at five points on the x-axis with shortest path routes in units of Λ. The flux gets smaller and less uniform as we move away from the origin. Figure 5.17 presents the corresponding local traffic load, i.e., the scalar flux Φ_{SP}, in units of Λ as a function of the distance from the origin. The load is further divided into a uniform component and a non-uniform component based on the directional distribution of the traffic.

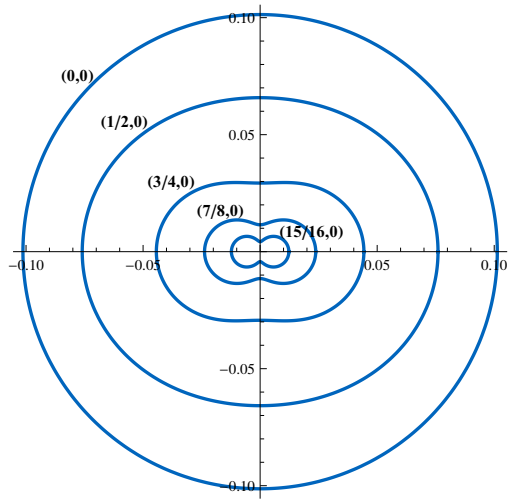


Figure 5.16. Angular flux, $\varphi(\vartheta)$, at five points on the x-axis with shortest path routes ($\Lambda = 1, A = \pi^2$).

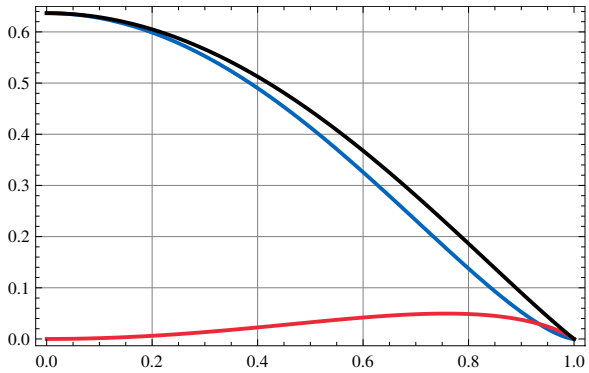


Figure 5.17. Traffic load ($\Lambda = 1, A = \pi^2$) as a function of the distance from the origin on a unit disk with uniform traffic demand and shortest path routing. The total load (highest) can be divided into a uniform component (in the middle) and a non-uniform component (lowest) based on the directional distribution.

For the scalar flux to be feasible, it is required that $\Phi(\boldsymbol{x}) \leq I^*$, for all $\boldsymbol{x} \in \mathcal{A}$, where the forwarding capacity is for the used forwarding method. The scalar flux has its maximum at the origin, $\Phi_{\mathrm{SP}}(\mathbf{0}) = (2/\sqrt{\pi A}) \cdot \Lambda$. As all the traffic can be forwarded using either single- or bidirectional forwarding, the origin is the bottleneck for both single- and bidirectional forwarding. Hence, for the total traffic load, we have

$$\Lambda \leq \frac{\sqrt{\pi}}{2} \cdot C \sqrt{nA}\, u(\nu).$$

Let us assume that we are studying a network area of 1 km^2 with 10 nodes per square meter (10^7 nodes total). Let us further assume that the forwarding capacity is similar to Figure 5.14, we have $C = 1$ packet per second for the nominal link capacity, and the transmission range is

0.5 meters. This results in an average of little under eight neighbors per node. Thus, we have $u_1 \approx 0.08$ for the single-directional forwarding capacity, and we can transmit $\Lambda \approx 230$ packets per second. As the traffic is uniform, and the shortest path routing is bidirectional, i.e., the path from $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$ is the same as the path from $\boldsymbol{x}_2$ to $\boldsymbol{x}_1$, we can also directly apply bidirectional forwarding. Since $u_2 \approx 0.12$ for the bidirectional forwarding capacity, we have $\Lambda \approx 330$ packets per second. Based on the decomposition of Figure 5.17 (the non-uniform component is small), the origin appears to be the bottleneck also for four-directional traffic. For the four-directional case $u_4 \approx 0.16$, and $\Lambda \approx 440$ packets per second.

5.7 Summary of Chapter 5

In this chapter, we studied the capability of an infinite network to relay multihop traffic with a given directional distribution. The corresponding quantity was named the forwarding capacity and defined as the maximum sustainable mean density of progress, i.e., the density of packets multiplied by their mean velocity in their respective directions. We studied the forwarding capacity as a function of the neighborhood size with three traffic patterns and assuming the Boolean interference model. The term single-directional forwarding capacity was used to describe a situation where all the traffic was flowing into the same direction. Regardless of the true directional distribution, it is always possible to achieve this capacity by using time sharing between the directions. Because of this, most effort was spent on the single-directional forwarding capacity, which was the baseline of the analysis. Notably higher capacities were achieved by exploiting the fact that traffic flowing in different directions can be interleaved in the microscopic level scheduling. We evaluated the gain from multidirectional forwarding compared with single-directional forwarding in the cases with balanced bidirectional and balanced four-directional traffic.

The forwarding capacity constitutes half of the network capacity problem at the limit where the number of nodes in the network is infinite, and the operation of the network is studied on two levels using the separation of scales. At the macroscopic level, the selected routing defines the fraction of traffic that traverses through a certain area of the massively dense network. The directional distribution of this “fraction”, on the other hand, gives the setup for the microscopic level problem. At the

microscopic level, the problem is to find a coordinated scheme or schedule that forwards the traffic efficiently. While the density of progress obtained with optimal scheduling is referred to as the forwarding capacity, which is a functional of the directional distribution of the traffic, the *achieved* forwarding capacity of the selected forwarding method sets an upper bound for the maximal macroscopic level load. The local traffic load at a certain point of the network must not exceed the available microscopic level forwarding capacity.

In the simulations, the infinite network was approximated by a large finite network where the relay traffic was generated by adding strips of sources and sinks to opposite edges of the network. The forwarding capacity was evaluated as a function of the network size to determine when the network starts to behave like an infinite one.

To find numerical bounds for the forwarding capacity, we constructed two algorithms based on the LP formulation of the problem. The first algorithm (CAA) gave an upper bound for the single-directional case by utilizing a subset of the necessary clique constraints. The second algorithm (PSA) identified a densely packed set of pairwise non-interfering paths connecting the opposite sides of the network. Within these paths, finding the optimal schedule, which gives a lower bound for the original problem, was simple. As it was also possible to schedule traffic in both directions of a path simultaneously (balanced bidirectional case), in addition to just a single direction, the approach gave the first value for the multidirectional gain, which in this case was 1.5. Eventually, we applied a greedy implementation of the maximum weight scheduling algorithm to obtain more accurate estimates and to evaluate the multidirectional gain in the selected cases. For balanced bidirectional traffic, the gain with GMWS was roughly 1.5. With balanced four-directional traffic, a greater gain was achieved with a small ν, after which it becomes favorable to use just two directions simultaneously. After an initial drop, the multidirectional gain increased, and the factor should ideally approach the value ($= 2$) obtained from the asymptotic analysis of the IFC.

The forwarding capacity remains zero until the network percolates. This happens around the neighborhood size of 4.5 (see Section 3.1.1 and the references thereof). The methods used here are not meant for studying the percolation threshold, but they complement the result. Near the percolation threshold, the combination of the used methods gives an accurate approximation, especially for the single-directional forwarding capacity. No-

tably, an earlier result for the locally coordinated ExOR protocol [BM05] was improved a by factor of two. When the mean number of neighbors grows, the estimate becomes less precise, and there remains a gap between the upper bound, obtained from the IFC, and the lower bound, obtained from the GMWS. It is obvious, however, that the greedy scheduler cannot be efficient with large neighborhood sizes where coordination is most relevant, and the IFC becomes more reachable.

We have also studied how the multidirectional traffic patterns can be used as components when approximating the forwarding capacity for an arbitrary directional distribution of the traffic. Already the bidirectional case offers a notable improvement in forwarding capacity compared with the single-directional case. It is also always possible to use bidirectional forwarding when the traffic demand is uniform, by selecting bidirectional routes at the macroscopic level.

6. Summary

The thesis considers the capacity of a large-scale wireless multihop network. We have presented a separation of scales for the capacity problem and analyzed the microscopic level forwarding capacity problem that is one of the two resulting subproblems. When the size of a wireless multihop network approaches infinity, two spatial scales can be identified. At the global scale, the network appears as a continuous medium consisting of an infinite number of nodes in a closed domain. From the local perspective the network appears to be a collection of randomly located nodes that continue ad infinitum.

In the capacity maximization problem of the infinite network, both of the viewpoints constitute their own separate subproblem. Locally, at the microscopic level, the shared wireless channel has to be used as efficiently as possible so that maximum amount of traffic can be forwarded. Globally, at the macroscopic level, the traffic load has to be spread across the network by an appropriate routing scheme so that no single area is congested. How efficiently the traffic is forwarded locally at the microscopic level sets an upper bound for the amount of traffic that can be routed to the area at the macroscopic level. On the other hand, the directional distribution of traffic that results from the macroscopic level routing problem affects the microscopic level forwarding capacity. The exact microscopic level problem is finding the maximum sustainable density of progress with a given directional distribution, which is independent of location, in an infinite plane.

The separation of scales has already been presented in [HV06]. However, the previous work on the topic treats the macroscopic level problem as a simple load balancing problem. This requires that the microscopic level forwarding capacity does not depend on the directional distribution of the traffic. For the commonly used interference models, this implies

that traffic flowing in different directions is handled via time sharing. This is not optimal, and better performance can be obtained by scheduling packets traversing in different directions in the same time slot. The capacity obtained via time sharing is thus a lower bound. Naturally, it is also possible to exploit the multidirectional gain on the microscopic level with simple, suboptimal, load balancing at the macroscopic level.

We formulate the separation of scales and take into account the effect of the directional distribution. We then describe the problems that need to be solved to obtain the network capacity. This results in a modified load balancing problem (3.10) at the macroscopic level. At the microscopic level, the problem is to find the forwarding capacity for a given directional distribution of traffic. Together these two problems can be used to resolve the open question of the capacity of a large-scale wireless multihop network.

Although the capacity of a wireless network has remained an open question, some aspects of it have been previously resolved. The scaling of the capacity is known in many situations. We reviewed the most relevant literature on the scaling laws for this thesis in Section 3.4. What is also known is that the maximum capacity can be achieved using seemingly simple algorithms. Unfortunately, the simplicity typically means that the time scale is infeasible, and it would take very long to attempt to reveal the exact value of the capacity. The related work was reviewed in Section 3.3.

Our own work in this thesis was threefold. In the first part, as already discussed above, we revised the separation of scales to take into account the possibility to utilize multidirectional traffic. The main contribution of the thesis was then related to the microscopic level problem of forwarding capacity. We began this by studying the IFC.

The IFC describes how much traffic, at most, can be forwarded in a single time slot. Multiple transmission modes are required to create a continuous flow in the network. Hence, performance of the best transmission mode, describing the instantaneous capacity, gives an upper bound for the performance with continuous traffic. By concentrating on a single transmission mode, we were able to analyze the asymptotic properties of the quantity. With different formulations for the problem, we obtained upper bounds for the case with single-directional traffic as well as for the multidirectional case where we have traffic in every direction.

An estimate of the IFC was obtained under various interference models. We studied the uni- and bidirectional Boolean interference model, both with fixed and adjustable transmission radii. We also studied the SINR-based interference models: the SINR threshold model and the Shannon model. Two algorithms were implemented for studying the optimal transmission mode. The moving window algorithm was suitable for the different versions of the Boolean interference model. The simulated annealing algorithm allowed us to also study the SINR-based interference models.

After studying the IFC, we turned to the actual forwarding capacity with multihop traffic. We presented several results on the forwarding capacity and the techniques used to obtain them. First, we tightened the single-directional upper bound obtained from the IFC with small neighborhoods, where it is not accurate, by using a clique formulation of the maximum flow problem. Then, we provided a lower bound for the forwarding capacity using path scheduling. The same method also offered the first view on what kind of gains can be achieved using multidirectional forwarding as the same path can be used either for single-directional traffic or bidirectional traffic. GMWS was then used to obtain an estimate for the forwarding capacity with different directional distributions. Finally, we discussed the significance of the results on the overall capacity of the network.

This thesis still only offers partial results on the capacity of a large-scale wireless network. There remain numerous research topics that can be studied further. Different network models constitute their own problems as, for example, the interference model can have a profound impact on the performance of the network. The effect of the multidirectional forwarding on the macroscopic level problem is still widely unknown. No work yet exists that properly considers the modified load balancing problem.

Bibliography

[Abr70] N. Abramson. THE ALOHA SYSTEM: another alternative for computer communications. In *AFIPS*, pages 281–285, 1970.

[AIM10] L. Atzori, A. Iera, and G. Morabito. The Internet of Things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.

[ALV06] O. Apilo, P. Lassila, and J. Virtamo. Performance of local forwarding methods for geographic routing in large ad hoc networks. In *MedHoc-Net*, pages 1–8, 2006.

[ASSC02] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114, 2002.

[BBW05] P. Balister, B. Bollobás, and M. Walters. Continuum percolation with steps in the square or the disc. *Random Struct. Alg.*, 26(4):392–403, 2005.

[BHG12] D. Benyamina, A. Hafid, and M. Gendreau. Wireless mesh networks design — a survey. *IEEE Communications Surveys & Tutorials*, 14(2):299–310, 2012.

[BM05] S. Biswas and R. Morris. ExOR: opportunistic multi-hop routing for wireless networks. *SIGCOMM Computer Communications Review*, 35(4):133–144, 2005.

[BMP08] C. Bordenave, D. McDonald, and A. Proutiere. Performance of random medium access control, an asymptotic approach. In *ACM SIGMETRICS*, pages 1–12, 2008.

[Bol85] B. Bollobás. *Random graphs*. Academic Press, 1985.

[Buc14] E. Buckingham. On physically similar systems; illustrations of the use of dimensional equations. *Phys. Rev.*, 4(4):345–376, 1914.

[Car10] P. Cardieri. Modeling interference in wireless ad hoc networks. *IEEE Communications Surveys & Tutorials*, 12(4):551–572, 2010.

[CKLS08] P. Chaporkar, K. Kar, X. Luo, and S. Sarkar. Throughput and fairness guarantees through maximal scheduling in wireless networks. *IEEE Transactions on Information Theory*, 54(2):572–594, 2008.

[CTM09] R. Catanuto, S. Toumpis, and G. Morabito. On asymptotically optimal routing in large wireless networks and geometrical optics analogy. *Computer Networks*, 53(11):1939–1955, 2009.

[DBT05] O. Dousse, F. Baccelli, and P. Thiran. Impact of interferences on connectivity in ad hoc networks. *IEEE/ACM Transactions on Networking*, 13(2):425–436, 2005.

[DFM+06] O. Dousse, M. Franceschetti, N. Macris, R. Meester, and P. Thiran. Percolation in the signal to interference ratio graph. *J. Appl. Probab.*, 43(2):552–562, 2006.

[DFT06] O. Dousse, M. Franceschetti, and P. Thiran. On the throughput scaling of wireless relay networks. *IEEE Transactions on Information Theory*, 52(6):2756–2761, 2006.

[DT04] O. Dousse and P. Thiran. Connectivity vs capacity in dense ad hoc networks. In *IEEE INFOCOM*, pages 476–486, 2004.

[DTH02] O. Dousse, P. Thiran, and M. Hasler. Connectivity in ad-hoc and hybrid networks. In *IEEE INFOCOM*, pages 1079–1088, 2002.

[Far05] A. Faragó. New analytical results on ad hoc network connectivity. In *CCN*, pages 126–131, 2005.

[FDTT07] M. Franceschetti, O. Dousse, D.N.C. Tse, and P. Thiran. Closing the gap in the capacity of wireless networks via percolation theory. *IEEE Transactions on Information Theory*, 53(3):1009–1018, 2007.

[FF56] L.R. Ford, Jr. and D.R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.

[Gil61] E.N. Gilbert. Random plane networks. *Journal of the Society for Industrial and Applied Mathematics*, 9(4):533–543, 1961.

[GJ79] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[GK98] P. Gupta and P.R. Kumar. Critical power for asymptotic connectivity. In *IEEE CDC*, volume 1, pages 1106–1110, 1998.

[GK00] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, 2000.

[GMPS04] A.El. Gamal, J. Mammen, B. Prabhakar, and D. Shah. Throughput-delay trade-off in wireless networks. In *IEEE INFOCOM*, pages 464–475, 2004.

[GRDK13] V. Gabale, B. Raman, P. Dutta, and S. Kalyanraman. A classification framework for scheduling algorithms in wireless mesh networks. *IEEE Communications Surveys & Tutorials*, pages 1–24, 2013. preprint.

[GT02] M. Grossglauser and D. Tse. Mobility increases the capacity of ad hoc wireless networks. *IEEE/ACM Transactions on Networking*, 10(4):477–486, 2002.

[Hae12] M. Haenggi. *Stochastic Geometry for Wireless Networks*. Cambridge University Press, 2012.

[Haj88] B. Hajek. Cooling schedules for optimal annealing. *Mathematics of Operations Research*, 13(2):311–329, 1988.

[HV06] E. Hyytiä and J. Virtamo. On load balancing in a dense wireless multihop network. In *NGI*, pages 72–79, 2006.

[HV07a] E. Hyytiä and J. Virtamo. On optimality of single-path routes in massively dense wireless multi-hop networks. In *ACM MSWiM*, pages 28–35, 2007.

[HV07b] E. Hyytiä and J. Virtamo. On traffic load distribution and load balancing in dense wireless multihop networks. *EURASIP Journal on Wireless Communications and Networking*, 2007.

[HV08] E. Hyytiä and J. Virtamo. Near-optimal load balancing in dense wireless multi-hop networks. In *NGI*, pages 181–188, 2008.

[HV09] E. Hyytiä and J. Virtamo. On the optimality of field-line routing in massively dense wireless multi-hop networks. *Performance Evaluation*, 66(3-5):158–172, 2009.

[IPSS08] J. Illian, A. Penttinen, H. Stoyan, and D. Stoyan. *Statistical Analysis and Modelling of Spatial Point Patterns*. John Wiley & Sons, 2008.

[IRK09] A. Iyer, C. Rosenberg, and A. Karnik. What is the right model for wireless channel interference? *IEEE Transactions on Wireless Communications*, 8(5):2662–2671, 2009.

[Jac04] P. Jacquet. Geometry of information propagation in massively dense ad hoc networks. In *ACM MobiHoc*, pages 157–162, 2004.

[JW10] L. Jiang and J. Walrand. A distributed CSMA algorithm for throughput and utility maximization in wireless networks. *IEEE/ACM Transactions on Networking*, 18(3):960–972, 2010.

[KGV83] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[KS04] M. Kalantari and M. Shayman. Routing in wireless ad hoc networks by analogy to electrostatic theory. In *IEEE ICC*, pages 4028–4033, 2004.

[KT75] L. Kleinrock and F. Tobagi. Packet switching in radio channels: Part I—carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE Transactions on Communications*, 23(12):1400–1416, 1975.

[KWE08] S. Kompella, J.E. Wieselthier, and A. Ephremides. Revisiting the optimal scheduling problem. In *CISS*, pages 492–497, 2008.

[LD05] M.E. Lübbecke and J. Desrosiers. Selected topics in column generation. *Operations Research*, 53(6):1007–1023, 2005.

[Mat86] B. Matérn. *Spatial Variation*. Lecture notes in statistics 36. Springer-Verlag, 1986.

[MEO11] P. Marbach, A. Eryilmaz, and A. Ozdaglar. Asynchronous CSMA policies in multihop wireless networks with primary interference constraints. *IEEE Transactions on Information Theory*, 57(6):3644–3676, 2011.

[MJD08] R. Maheshwari, S. Jain, and S.R. Das. A measurement study of interference modeling and scheduling in low-power wireless networks. In *ACM SenSys*, pages 141–154, 2008.

[MR96] R. Meester and R. Roy. *Continuum Percolation*. Cambridge University Press, 1996.

[MRR+53] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[MSZ06] E. Modiano, D. Shah, and G. Zussman. Maximizing throughput in wireless networks via gossiping. In *ACM SIGMETRICS*, pages 27–38, 2006.

[MWH01] M. Mauve, J. Widmer, and H. Hartenstein. A survey on position-based routing in mobile ad hoc networks. *IEEE Network*, 15(6):30–39, 2001.

[Nou07] J. Nousiainen. Local forwarding methods in large ad hoc networks. Technical report, Helsinki University of Technology, 2007.

[PMS09] G.S. Paschos, P. Mannersalo, and S. Stanczak. Extending the percolation threshold using power control. In *IEEE WCNC*, pages 1–6, 2009.

[PPT89] T.K. Philips, S.S. Panwar, and A.N. Tantawi. Connectivity properties of a packet radio network model. *IEEE Transactions on Information Theory*, 35(5):1044–1047, 1989.

[PRK+07] L. Popa, A. Rostami, R.M. Karp, C. Papadimitriou, and I. Stoica. Balancing traffic load in wireless networks with curveball routing. In *ACM MobiHoc*, pages 170–179, 2007.

[QTZ00] J. Quintanilla, S. Torquato, and R.M. Ziff. Efficient measurement of the percolation threshold for fully penetrable discs. *Journal of Physics A*, 33(42):L399–L407, 2000.

[QZ07] J. Quintanilla and R.M. Ziff. Asymmetry in the percolation thresholds of fully penetrable disks with two different radii. *Phys. Rev. E*, 76:051115, 2007.

[Rob75] L.G. Roberts. ALOHA packet system with and without slots and capture. *SIGCOMM Comput. Commun. Rev.*, 5(2):28–42, 1975.

[RSS09] S. Rajagopalan, D. Shah, and J. Shin. Network adiabatic theorem: an efficient randomized protocol for contention resolution. In *ACM SIGMETRICS*, pages 133–144, 2009.

[RT99] E.M. Royer and C. Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, 6(2):46–55, 1999.

[SMR+09] P. Santi, R. Maheshwari, G. Resta, S. Das, and D.M. Blough. Wireless link scheduling under a graded SINR interference model. In *ACM FOWANC*, pages 3–12, 2009.

[SSV12] D. Sivakumar, B. Suseela, and R. Varadharajan. A survey of routing algorithms for MANET. In *ICAESM*, pages 625–640, 2012.

[Sto02] I. Stojmenovic. Position-based routing in ad hoc networks. *IEEE Communications Magazine*, 40(7):128–134, 2002.

[Tas98] L. Tassiulas. Linear complexity algorithms for maximum througput in radio networks and input queued switches. In *IEEE INFOCOM*, pages 533–539, 1998.

[TE92] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, 1992.

[TG04] S. Toumpis and A.J. Goldsmith. Large wireless networks under fading, mobility, and delay constraints. In *IEEE INFOCOM*, pages 609–619, 2004.

[TM03] M. Tubaishat and S. Madria. Sensor networks: an overview. *IEEE Potentials*, 22(2):20–23, 2003.

[Vir92] J.T. Virtamo. A model of reservation systems. *IEEE Transactions on Communications*, 40(1):109–118, 1992.

[VWM13] S. Vural, D. Wei, and K. Moessner. Survey of experimental evaluation studies for wireless mesh network deployments in urban areas towards ubiquitous internet. *IEEE Communications Surveys & Tutorials*, pages 1–17, 2013. preprint.

[WY04] Peng-Jun Wan and Chih-Wei Yi. Asymptotic critical transmission radius and critical neighbor number for k-connectivity in wireless ad hoc networks. In *ACM MobiHoc*, pages 1–8, 2004.

[XK04] F. Xue and P.R. Kumar. The number of neighbors needed for connectivity of wireless networks. *Wireless Networks*, 10(2):169–181, 2004.

[YMG08] J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. *Comput. Netw.*, 52(12):2292–2330, 2008.

Aalto-DD 107/2013

ISBN 978-952-60-5234-2
ISBN 978-952-60-5235-9 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934
ISSN 1799-4942 (pdf)

Aalto University
School of Electrical Engineering
Department of Communications and Networking
www.aalto.fi

BUSINESS +
ECONOMY

ART +
DESIGN +
ARCHITECTURE

SCIENCE +
TECHNOLOGY

CROSSOVER

DOCTORAL
DISSERTATIONS