Frequency Assignment in GSM Networks: Models, Heuristics, and Lower Bounds

vorgelegt von
Diplom-Mathematiker

ANDREAS EISENBLÄTTER

Von der Fakultät II - Institut für Mathematik
der Technische Universität Berlin
zur Erlangung des akademischen Grades eines

DOKTOR DER NATURWISSENSCHAFTEN

genehmigte Dissertation

Promotionsausschuß:

Vorsitzender:	Prof. Dr. Michael Scheutzow
Berichter:	Prof. Dr. Martin Grötschel
Berichter:	Prof. Dr. Günter M. Ziegler

Tag der wissenschaftlichen Aussprache: 5. Juni 2001

Berlin 2001
D 83

Abstract

Mobile cellular communcication is a key technology in today's information age. Despite the continuing improvements in equipment design, interference is and will remain a limiting factor for the use of radio communication. This Ph. D. thesis investigates how to prevent interference to the largest possible extent when assigning the available frequencies to the base stations of a GSM cellular network. The topic is addressed from two directions: first, new algorithms are presented to compute "good" frequency assignments fast; second, a novel approach, based on semidefinite programming, is employed to provide lower bounds for the amount of unavoidable interference.

The new methods proposed for automatic frequency planning are compared in terms of running times and effectiveness in computational experiments, where the planning instances are taken from practice. For most of the heuristics the running time behavior is adequate for interactive planning; at the same time, they provide reasonable assignments from a practical point of view (compared to the currently best known, but substantially slower planning methods). In fact, several of these methods are successfully applied by the German GSM network operator E-Plus.

The currently best lower bounds on the amount of unavoidable (co-channel) interference are obtained from solving semidefinite programs. These programs arise as nonpolyhedral relaxation of a minimum k-partition problem on complete graphs. The success of this approach is made plausible by revealing structural relations between the feasible set of the semidefinite program and a polytope associated with an integer linear programming formulation of the minimum k-partition problem. Comparable relations are not known to hold for any polynomial time solvable polyhedral relaxation of the minimum k-partition problem. The application described is one of the first of semidefinite programming for large industrial problems in combinatorial optimization.

Keywords: GSM, frequency planning, mimimum graph k-partition, heuristics, semidefinite programming, integer programming, polytopes.
Mathematics Subject Classification (MSC 2000): 90C27 90C35 90B18 90C22 90C57

Preface

A crucial and difficult task in operating a GSM network is to establish a good frequency plan. When the project described in this thesis started in 1995, the commercially available software tools to assist a radio engineer in this task were insufficient. Hence, many engineers kept on planning the frequency (re-)use essentially by hand. Facing a stunning growth of the GSM network installations, this habit soon hit its limits. In search for new planning algorithms the German operator E-Plus Mobilfunk GmbH & Co. KG approached Professor Dr. Martin Grötschel, head of the optimization department at the Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB). A cooperation between E-Plus and ZIB on the frequency planning problem was set up.

At that time, I applied at ZIB for a Ph. D. position, and it became my task and my challenge to develop automatic frequency planning software for the use at E-Plus. The software that was developed and several subsequent extensions are nowadays in successful use at E-Plus, integrated into the regular network planning system.

This thesis describes in detail the planning methods developed, the underlying mathematical model, its connection to the problem of finding a minimum k-partition in a graph, and how a quality guarantee for a frequency assignment can be computed by solving a large-scale semidefinite program. All of this is documented in a form accessible and informative to a mathematician as well as to a radio engineer, I hope.

I am greatly indebted to my family, my friends, and my colleagues for their continuing support in many ways. This thesis would not have been possible without them. To all of them go my sincere thanks.

In particular, I would like to mention three persons. My advisor Professor Dr. Martin Grötschel has provided a most fertile and stimulating environment at the Konrad-Zuse-Zentrum für Informationstechnik Berlin. Dr. Thomas Kürner from the E-Plus Mobilfunk GmbH & Co. KG has been my link to the radio engineering world, and he introduced me to the European Cooperative Research in Science and Technology action 259 or COST 259, for short. My understanding of the GSM radio interface, in general, and the technical aspect of frequency planning, in particular, has benefitted substantially from the numerous discussions with him and other participants of Cost 259. My colleague Dr. Christoph Helmberg has seen me by-pass his advertisements for semidefinite programming for a long time, and yet he supported me right on from the minute I decided to give it a finally successful try.

February 14, 2001 Andreas Eisenblätter

Contents

List of Figures

List of Tables

List of Algorithms

CHAPTER 1

Introduction

Frequency planning for GSM cellular radio networks is the topic of this thesis. We present results which were obtained in the context of a cooperation between the Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB) and the German GSM 1800 network operator E-Plus Mobilfunk GmbH & Co. KG. This cooperation started in September 1995, and has since then been extended several times.

Our focus was primarily on fast frequency planning heuristics for the use in the regular radio planning process at E-Plus. New planning methods were developed at ZIB and integrated into E-Plus' software environment. In 1997, our software was first used successfully in practice, and, in the meantime, it has been extended to better meet practical needs. We also studied approaches to provide quality guarantees for heuristically generated frequency plans.

GSM is a second generation digital cellular radio system. Among others, GSM provides telephony service: a mobile phone may establish a communication link with any other party reachable through a public telephone network. This is achieved by means of a radio link to some stationary antenna which is part of a large infrastructure, see Figure 1.1. Since the introduction of GSM, radio telephony has grown from a costly service used by few professionals to a mass market with penetration rates as high as 70 % in Finland and Iceland, for example. In more and more countries, the mobile cellular phone subscribers outnumber the fixed-line telephone subscriptions.

Frequency planning is a key issues in fully exploiting the radio spectrum available to GSM. It has a significant impact on the quantity as well as on the quality of the radio communication services. Roughly speaking, radio communication requires a radio signal of sufficient strength which is not suffering too severely from interference by other signals. In a cellular system like GSM, these two properties, strong signals and little interference, are in conflict. The problem of finding a "good" frequency plan is sketched in the following and described in full detail later.

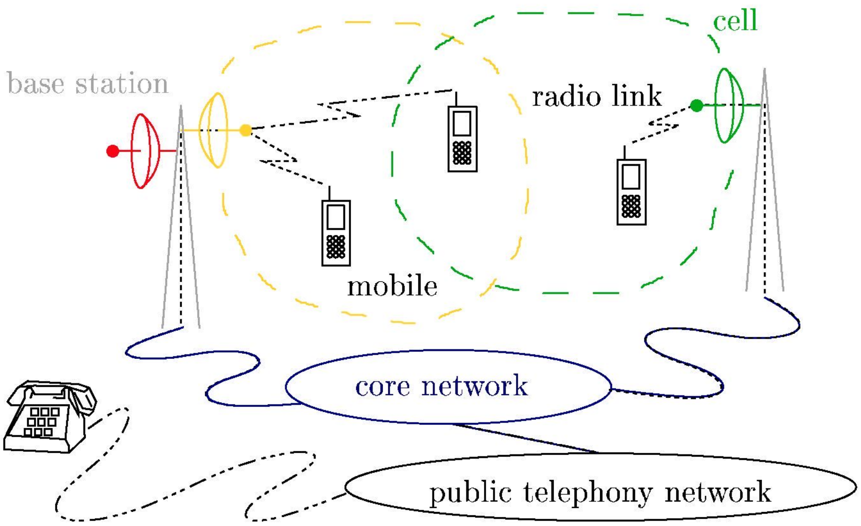


Figure 1.1: GSM in principle

Every base station operates a number of elementary transceivers, each of which uses some frequency to transmit on. A network operator has usually between 30 and 120 evenly spaced out frequencies available to satisfy the demand of several thousand transceivers. The reuse of frequencies is therefore unavoidable, but this reuse is limited by interference and by so-called separation requirements. Significant interference may occur between transceivers using the same frequency (co-channel) or directly neighboring frequencies (adjacent channels). Separation requirements are given for pairs of transceivers and impose that the assigned frequencies have a specified minimum separation in the electromagnetic spectrum. Furthermore, not every frequency is necessarily available for all transceivers. In summary, the problem to be solved is the following.

> Given are the transceivers, the set of generally available frequencies, the local unavailabilities, as well as three square matrices specifying the necessary minimum separation, the potential co-channel, and the potential adjacent channel interference values. One frequency has to be assigned to every transceiver such that the following holds. All separation requirements are met, and all assigned frequencies are locally available. The optimization goal is to find a frequency assignment resulting in the least possible interference.

We are primarily interested in minimizing the sum over the incurred co- and adjacent channel interferences here, but other goals of practical interest exist as well. Striving for "minimum interference" assignments is in a sense a luxury to be paid for with frequencies. If only few frequencies are available to a GSM operator, then the emphasis is likely on providing some acceptable frequency plan at all. But the optimization aspect gains importance when feasible assignments can be obtained "easily." E-Plus is currently in the latter position. The network contains roughly 8000 base stations, and 115 frequencies are available.

New assignments have to be computed on several occasions. Some examples are: the network is modified or expanded, the characteristics of a transceiver are changed, or significant unpredicted interference is reported and has to be resolved.

Several commercial software packages exist which allow to document the network configuration, to plan radio coverage, and to predict interference in addition to frequency planning. GSM infrastructure manufacturers develop such tools, but also independent companies such as AIRCOM International (Asset), COSIRO GmbH (Fun), Lociga Plc. (Odyssey), L&S Hochfrequenztechnik GmbH (CHIRplus), or Metapath Software International Limited (PlaNet). At the time when the cooperation with E-Plus started, however, the optimization of frequency assignments with respect to interference was often only poorly supported. This has certainly improved since then.

In the following, we deal with a broad spectrum of topics ranging from the technical background of the GSM frequency planning problem over alternative mathematical models and heuristic planning methods to quality assessments for the generated frequency plans. In addition to this introduction, the thesis comprises seven chapters and an appendix containing a compilation of mathematical notation used in the following. The content of each chapter is now briefly stated.

In *Chapter 2*, we give a survey of GSM and explain the technical conditions to be taken into account during frequency planning. We also describe how the input data is generated and stress the importance of reliable interference predictions for the success of automatic frequency planning.

In *Chapter 3*, the frequency planning problem (as sketched above) is formalized as a combinatorial minimization problem. We investigate the computational complexity of the model beyond stating its $\mathcal{NP}$-hardness, and we discuss extensions of the model as well as alternative models.

In *Chapter 4*, seven heuristic frequency planning methods are described. Depending on the point of view, five or six of them can be used (in combination) for generating frequency assignments in practice. In accordance with the objective of the cooperation with E-Plus, our focus is on fast methods rather than on more elaborate, but slower methods.

In *Chapter 5*, the previously described planning methods are compared on the basis of realistic frequency planning problems. In this comparison, we include the currently best performing method we know of as a reference. An analysis of the realistic planning scenarios is provided, and we explain how to use the described heuristics in order to obtain time savings and quality improvements in practice.

In *Chapter 6*, a lower bound on the amount of unavoidable co-channel interference is computed for each planning scenario. These bounds are obtained by solving large semidefinite programs (which are challenges to the currently existing solvers). Based on these bounds, quality guarantees are provided for the frequency assignments from the preceding chapter. Moreover, we introduce a relaxed version of our frequency planning problem. The solutions for the relaxed problem can sometimes be turned into feasible assignments for the original problem. Exploiting this connection, we point out room for further development of heuristics.

The relaxed version of frequency planning leads us to the study of the mathematical MINIMUM K-PARTITION problem and its semidefinite relaxation (which we considered so far mostly as a "black box" providing lower bounds).

In *Chapter 7*, we mostly review results on a polytope, which is obtained as the convex hull of the feasible solutions to an integer linear programming formulation of the MINIMUM K-PARTITION problem. Particular emphasis is on the hypermetric inequalities.

In *Chapter 8*, we first give an introduction to semidefinite programming and then study the semidefinite relaxation for the MINIMUM K-PARTITION problem. In particular, we describe a large class of valid inequalities for the solution set of the semidefinite relaxation (a shifted version of hypermetric inequalities), and we prove that neither the linear programming relaxation of the integer linear programming formulation nor the semidefinite programming relaxation is always stronger than the other.

CHAPTER 2

Frequency Planning in GSM

The *General System for Mobile Communications* or GSM,[1] for short, is a multi-service cellular communication system providing speech and data services. The most important service is radio telephony, but data services like short message service (SMS) and mobile Internet access building on the Wireless Application Protocol (WAP) are also rapidly gaining popularity.

GSM

In this chapter, the ground is laid for understanding the constraints and the objectives of frequency planning for a GSM network. Moreover, the frequency planning problem is informally stated. A brief sketch of GSM's history is given in Section 2.1. The four major subsystems are explained in Section 2.2, and those parts of the radio interface which are relevant to frequency planning are discussed in detail. In Section 2.3, we show how to phrase frequency planning as an optimization problem, explain the constraints to be met, discuss how the input data is generated, and report on practical aspects of frequency planning. The reader who is familiar with GSM and is primarily interested in frequency planning may skip straight to Section 2.3.

2.1 A Brief History of GSM

GSM has been designed as a pan-European cellular communications system to be operated in the 900 MHz radio frequency band. It has subsequently been extended to the 1800 MHz band in Europe. Today, there are also variants operated in the 1900 MHz band in other parts of the world. The respective systems are nowadays called GSM 900, GSM 1800, and GSM 1900. A fourth variant, called GSM 400, is under specification and will operate between 400 and 500 MHz. Table 2.1 lists the precise frequency bands for mobile station to base station (up-link) and base

[1] GSM and "General System for Mobile Communications" are trademarks of the GSM Association, Geneva, Switzerland.

station to mobile station (down-link) radio communication for all GSM variants. Apart from the frequency bands (and the thereby caused differences in the radio transmission equipment) there is little difference between the systems.

system	up-link band	down-link band
GSM 900	890–915 MHz	935–960 MHz
GSM 1800	1710–1985 MHz	1805–1880 MHz
GSM 1900	1850–1910 MHz	1930–1990 MHz
GSM 400	450.4–457.6 MHz 478.8–486.0 MHz	460.4–467.6 MHz 488.8–496.0 MHz

Table 2.1: GSM radio frequency bands

CEPT

In 1978, two bands of 25 MHz radio spectrum around 900 MHz were reserved for mobile communication in Europe. In 1982, the *Conférence Européenne des Postes et Télécommunications* (CEPT) established the Groupe Spéciale Mobile, abbreviated as GSM. The task of this group was to develop the specification of a pan-European mobile communications network. Four years later, a Permanent Nucleus of GSM was set up to coordinate the further developments, including the installation of test beds to compare alternative system and radio interface designs. By 1987, it was apparent that the new (second generation) system would be digital (as opposed to the then existing first generation analog systems) and use time division multiple access on the radio interface.

PTT

On the 7th of September 1987, thirteen European countries signed the GSM Memorandum of Understanding (MoU) which covered, for example, time-scales for the procurement and the deployment of the system, compatibility of numbering and routing plans, concerted service introductions, and harmonization of tariff principles (cf. Mouly and Pautet [1992]). From then on, many *Posts, Telegraphs, and Telephones public operating companies* (PTTs), manufactures, and research institutes collaborated in the design of an entirely digital system.

About two years later, the United Kingdom published a document calling for a mass market mobile communications system operating in the 1800 MHz frequency band. This lead to the definition of DCS-1800. DCS-1800 is now being called GSM 1800.

Around 1990, it became evident that a deployment of GSM systems within the foreseen time-scales would be impossible without issuing the specification in mutually compatible phases. GSM became an evolving standard. The majority of the Phase 1 specification was published in 1990. At that time, the Technical Specification of GSM 900 contained 130 recommendations on more than 5000 pages. These recommendations comprised the full specification of the radio interface as well as a detailed specification of infrastructure, architecture, and many intra- and intersystem interfaces. The first GSM pilot network was successfully demonstrated at the Telecom '91 fair, organized by the *International Telecommunication Union* (ITU). Later in the same year, several networks were fully operational, but type approved GSM terminals were not available, and GSM was made fun of as the acronym for the prayer "God Send Mobiles." The reason was simply that the procedures for type approval were not settled. In April 1992, an *Interim Type Approval* (ITA) was agreed on.

ITU

ITA

In the course of 1992, hand-held terminals with ITA became widely available, and by the end of 1992 GSM networks were operative in Denmark (2), Finland (2), France (1), Germany (2), Italy (1), Portugal (2), and Sweden (3). Some roaming agreements had also been signed. In the year 1993, the first million of GSM subscribers was registered, 70 parties from 48 countries had signed the MoU, and the British operator One-2-One launched the first GSM 1800 network. The world-wide success of GSM is well reflected by its growth in terms of operating networks, total number of subscribers, and the number of countries with GSM installations over the last decade, see Table 2.2, basing on figures published by GSM Association [2000]; www.emc-database.com [2000].

year	networks	subscribers	countries
1992	13	250,000	7
1993	32	1,000,000	18
1994	69	4,000,000	43
1995	117	12,000,000	69
1996	167	30,000,000	94
1997	178	73,000,000	107
1998	320	135,000,000	118
1999	355	255,000,000	130
2000[a]	376	397,000,000	142

[a] October 2000

Table 2.2: Growth of GSM

GSM soon spread beyond Europe. In 1992, the first non-European operator, Telstra from Australia, had signed the MoU. In 1994, the Federal Communications Commission (FCC) of the United States of America auctioned several licenses to operate mobile networks around 1900 MHz. No particular network type was imposed, and the first GSM 1900 network (then still called PCS 1900) was launched by American Personal Communications in November 1995.

By the end of the third quarter of the year 2000, there were 376 operating GSM networks world-wide with a total of 396.6 million subscribers. In Europe alone (including Russia), there were 141 GSM 900 and GSM 1800 networks with a total of 255.1 million subscribers.

In the meantime, the specification of GSM had been continued. GSM Phase 2 was issued in 1993. Numerous extensions were made such as an option for half-rate speech telephony, improved short message services, calling/connected line identity presentation, call waiting and call hold features, multi-party calls, and advice of charge. But data transmission kept essentially restricted to at most 9.6 kbps. Opening up this bottleneck has become a central theme in the still ongoing specifications of Phase 2+. Three major new technologies are introduced. (The transmission rates are taken from GSM Association [2000, Glossary].)

HSCSD

High Speed Circuit Switched Data (HSCSD) allows the transmission of circuit-switched data with a speed of up to 57.6 kbps. The data rate per time slot is increased to 14.4 kbps and up to four consecutive time slots may be concatenated.

GPRS

General Packet Radio Service (GPRS) introduces the option for packet-switched services into GSM. GPRS will provide data transmission speeds of up to 115 kbps to mobile users.

EDGE

Enhanced Data for GSM Evolution (EDGE) uses a new modulation scheme to allow data transmission with rates of up to 384 kbps on the basis of the GSM infrastructure.

These technologies, however, require a higher signal to noise ratio at the receiver (i. e., they can cope with less interference) than regular data transmissions in order to guarantee proper reception. This has an impact on the planning of the radio interface in general and frequency planning in particular.

UMTS

Finally, over the past years the standards for third generation cellular mobile systems (IMT-2000) have been under development. The *Universal Mobile Telecommunications System* (UMTS) is one of them, for which a first standard was issued in the beginning of the year 2000. The radio interface of UMTS is different from that of GSM. The *Code Division Multiple Access* (CDMA) scheme is used, and no frequency planning problem comparable to that of GSM has to be solved. UMTS is expected to be commercially available in Europe around the year 2002. It allows for true global roaming, and it is supposed to support a wide range of voice and data services. Depending on the user mobility and the propagation environment, different maximal data transmission rates are foreseen: 144 kbps for vehicular, 384 kbps for pedestrian, and 2 Mbps for indoor users. UMTS will be deployed parallel to GSM, and more than ten years of coexistence of GSM and UMTS are expected. In Germany, for example, the first GSM license expires at the end of the year 2009.

2.2 The General System for Mobile Communications

GSM is a multi-service cellular radio system, capable of transmitting *speech* as well as *data* and with numerous *supplementary features*. The area covered by a GSM network consists of (overlapping) cells, which are served by stationary antennas. The kind of service provided depends on the *content of the subscription*, the *capabilities of the network*, and the *capabilities of the user-held equipment*.

2.2.1 Mobile Stations

A radio link connects a mobile station to the GSM network infrastructure. A switched-on mobile station is either in idle mode or in dedicated mode. In *idle mode*, the mobile station listens to control channels, but does not have a channel of its own. In *dedicated mode*, a bidirectional channel is allocated to the mobile terminal allowing it to exchange information with and through the GSM network. A mobile terminal switches from idle into dedicated mode, for example, if the user wants to place a call. The mobile sends a corresponding request to the cell of which it monitors the control channel. Another example is the arrival of a call. In that case, however, the network is generally not aware of the cell a mobile terminal is listening to (if any) so that the mobile is "paged."

idle mode
dedicated mode

To limit the amount of paging messages, location areas are defined. A *location area* is a group of cells, and every cell belongs to exactly one location area. The identity of the location area is broadcast by each cell so that a mobile station can always find out what location area it is in. In case the mobile is moved and the location area changes, a message is sent out, and the network registers the change. This process is called *location updating*. When a call for a mobile station arrives, a paging message for that mobile station is broadcast in all cells of the location area the mobile station has last registered in. (Sometimes, this is preceded by paging the

location area
location updating

mobile station only in the cell of last active contact with the network.) If this paging fails, a paging message is broadcast in all cells of the network.

A mobile terminal may, of course, also be moved while in dedicated mode. Depending on the distance to the serving base station and the propagation conditions, the radio link can degrade below the required quality. The bidirectional channel has then to be dropped or to be maintained by another cell. Changing the serving cell in dedicated mode is called *hand-over*. During a hand-over, the network has to reroute the communication channel without the user noticing. The decisions, when to perform a hand-over and to which cell, are taken in the network infrastructure, but with the support of the mobiles. Each mobile terminal routinely monitors a list of neighboring cells, records the reception quality, and sends measurement reports the network.

hand-over

Despite the option of international roaming, a GSM telephone call usually comes to an end at national borders due to a call drop. The reasons are primarily billing issues. But (presuming frequency band compatibility) the mobile station may then log on into a foreign network in order to place and to receive calls, if the user's subscription allows international roaming and appropriate roaming agreements are made between the operators.

2.2.2 Subsystems

Next to the *mobile stations*, the three further major parts of GSM are the *base station subsystem*, the *network and switching subsystem*, and the *operation and maintenance subsystem.* A detailed description of these subsystems and their interfaces is given in the relevant standards issued by the *European Telecommunications Standards Institute* (ETSI), Sophia Antipolis, France. A more accessible source of information, however, is the book of Mouly and Pautet [1992].

ETSI

MS

SIM

A *Mobile Station* (MS) usually consists of some mobile equipment, like a hand-held mobile, and a *Subscriber Identification Module* (SIM), which is inserted into the mobile equipment. Depending on the frequency band of the network, see Table 2.1, different mobile equipment is typically required, but the same SIM can be used. Modern dual- or triple-band mobile terminals allow to communicate in two or three of those bands. The SIM carries an *International Mobile Subscriber Identify* (IMSI), personalizing the mobile equipment, and can be protected by a *Personal Identity Number* (PIN), similar to the PINs of credit cards. The SIM is the peer of the network during authentication, and it is involved in ciphering and

IMSI

PIN

deciphering transmitted messages (when encryption is applied).

BSS *BTS* *sector* *cell* *TRX* *BSC*

The *Base Station Subsystem* (BSS) comprises base transceiver stations and base station controllers. A *Base Transceiver Station* (BTS) is the peer of a mobile terminal in radio communications, both having radio transmission and reception devices, including antennas and all necessary signal processing capabilities. The site at which a BTS is installed is organized in *sectors*; one or three sectors are typical. An antenna is operated for each sector. If three sectors exist, then antennas with an opening angle of 120 degree are usually employed. If only one sector exists at a site, then an omnidirectional antenna can be used. (The details of how many sectors to choose, which antenna types, etc., depend on the practical needs, and are more complex than indicated here.) Each sector defines a *cell*. The capacity of a cell is determined by the number of elementary transmitter/receiver units, called *TRXs*, installed for the sector. As a rule of thumb, the first TRX of a sector provides capacity for 6 parallel calls, and each additional TRX for seven to eight more calls. The reduced capacity of the first and some of the additional TRXs is due to the need to transmit cell organization and protocol information. A maximum of 12 TRXs can be installed for one sector of a BTS. Every BTS is connected to one *Base Station Controller* (BSC), whereas one BSC typically handles several BTSs in parallel. A BSC is in charge of the allocation and release of radio channels as well as the management of hand-overs. All cells in a location area have to be controlled by the same BSC, but one BSC may serve more than one location area.

NSS *MSC* *core network* *GMSC* *HLR* *VLR* *AuC*

The *Network and Switching Subsystem* (NSS) manages the communication to and from GSM users. Every BSC is connected to one *Mobile service Switching Center* (MSC), and the *core network* interconnects the MSCs. Specially equipped *Gateway MSC*s (GMSCs) interface with other telephony and data networks. The *Home Location Registers* (HLRs) and *Visitors Location Registers* (VLRs) are data base systems, which contain subscriber data and facilitate mobility management. Each Gateway MSC consults its home location register if an incoming call has to be routed to a mobile terminal. The HLR is also used in the authentication of the subscribers together with the *Authentication Center* (AuC). The VLRs are associated to one or more MSCs and temporarily store information on all subscribers that were last traced in one of the BSCs attached to any of its associated MSC(s). The interworking of all components of the NSS is organized via a SS7 signaling network.

OSS

The *Operation and maintenance SubSystem* (OSS) is specified to a smaller extent than the rest of GSM. The network is run and maintained through the OSS: calls have to be billed and charged; SIMs have to be initialized; stolen or misbehaving mobile equipment is registered and possibly excluded from network service on the basis of the *Equipment Identity Register* (EIR). The network and switching subsystem, the base station subsystem, and, to some extent, also the mobile stations (via the BSS) are administered from *Operation and Management Centers* (OMC).

EIR

OMC

Three of the four subsystems are shown in Figure 2.1: Mobile Station (MS), Base Station Subsystem (BSS), and Network and Switching Subsystem (NSS). The interface between the MSCs and the BSCs is called *A interface*; the interface between the BSCs and the BTSs is called *Abis interface*; and the *Radio Interface* is between the BTSs and the MSs.

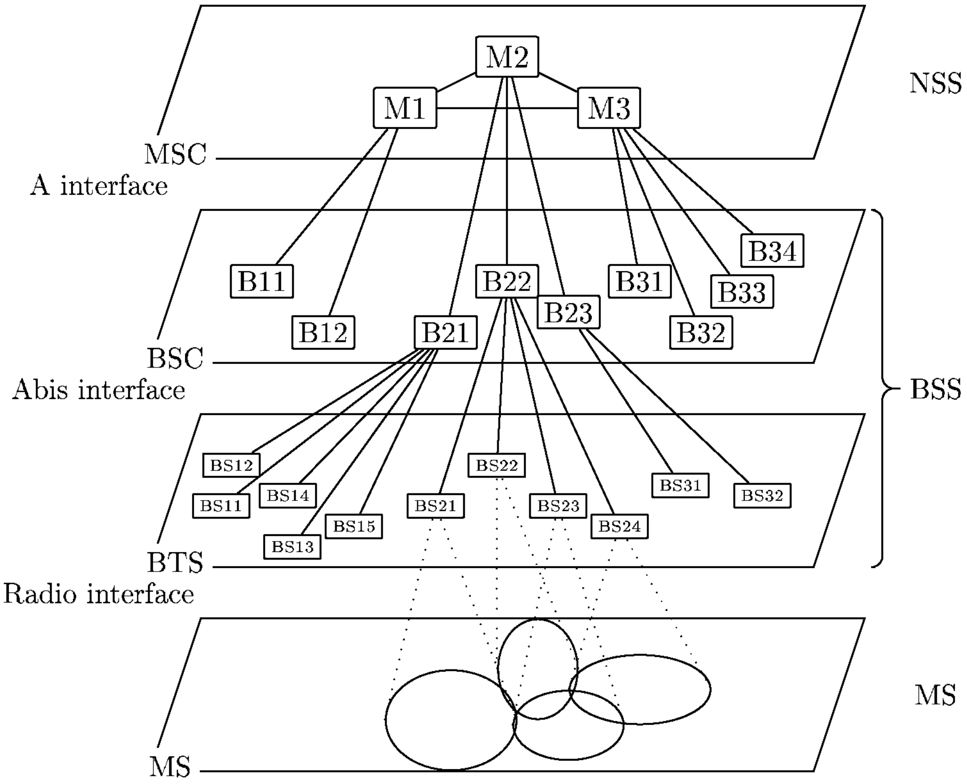


Figure 2.1: Architecture of GSM

2.2.3 Network Dimensioning

Having seen the major subsystems of GSM, a natural question is how to lay out an actual GSM network such that it provides the desired services cost-effectively. Numerous decisions have to be taken. We give a few examples with a strong appeal to combinatorial optimization:

Where to install the BTSs? How to adjust the antennas and what frequencies to use? How to connect the BTSs to the BSCs, and where to put the MSCs? How to connect the MSCs among each other and to the BSCs?

These important questions have to be answered prior to network deployment or expansion. All of them have an impact on generating revenues, because these decisions affect the cost of deploying and operating the network as well as the quality of service that can be offered.

Before focusing on frequency assignment in the chapters to come, we pick out some of these questions and explain the underlying optimization problem briefly. We give references, whenever we are aware of them.

At the core of planning a network deployment or extension is customers' demand. This demand may be observed or forecasted. In one way or another, the customers' demand for mobile telecommunications has to be made precise in a geographical distribution in terms of Erlang, a unit for measuring telecommunication demand. This distribution essentially states how large the need for mobile telecommunications is depending on the location.

Base Transceiver Station Location is the step in which radio engineers decide how many and where to erect BTSs in order to provide service for the (prospective) demand. This is a mixture of determining sites, which are preferable from an "electromagnetic" point of view (providing good coverage), and searching for sites, which are actually available. Research in this direction has been carried out, for example, in ACTS/STORMS project (supported the European Union), see Menolascino and Pizarroso [1999], as well as by Eidenbenz, Stamm, and Widmayer [1999] and Tutschku, Mathar, and Niessen [1999].

Base Transceiver Station Clustering denotes here the problem of where to place the BSCs and which BTSs to connect to them. Examples for the issues to be taken into account are the costs for renting or building spaces for operating BSCs and the running cost of attaching BTSs to BSCs by cables or point-to-point radio links. The mobility profile of customers also plays a role here, because hand-overs between cells handled by the same BSC are treated locally for the most part, whereas an inter-BSC hand-over requires a rerouting of connections in the core network also. Similar comments apply with respect to location-updating. Ferracioli and Verdone [2000] report on results in this area.

Core Network Design denotes here the planning necessary to decide where to operate MSCs, which BSCs to connect to them, and how to interconnect the MSCs among each other. The locations of MSCs are usually more dependent on "political" rather than "technical" considerations. The core network may comprise leased lines, the operator's own cable infrastructure, and point-to-point radio links. Usually, not every pair of MSCs is connected directly in the core network. Instead, routing tables are used to describe how to route traffic from one MSC to another along one or more links. The network has to be laid out (selection of connections, capacities, and routings) in such a way that a failure of a single link or a failure of a single MSCs has only a "manageable" impact on the traffic volume, which can be handled by the remaining part of the network. Such a network is called "survivable" in the literature, see, for example, Wessäly [2000] and the references therein.

Frequency Assignment or Channel Assignment or Frequency Planning are synonyms for the following problem. Once the sites for the BTSs are selected and the sector layout is decided, the number of TRXs to be operated per sector has to be fixed. This is done by means of the Erlang-B formula, taking the demand to support and the maximally tolerable blocking probability (of 2% or the like) as input. The result is a listing of the demand in TRXs per cell. Now, every TRX has to receive a channel. This demand has to be satisfied by a frequency plan.

The last problem is going to be the central topic from now on, and further details of the radio interface are discussed next.

2.2.4 Along the Radio Interface

In order to understand the various restrictions and the possible alternative objectives in frequency planning, we take a closer look at the technicalities of the GSM radio interface. Even more details can be found in the books by Mouly and Pautet [1992] and Redl, Weber, and Oliphant [1995] as well as in the relevant ETSI standards.

FDMA
TDMA

GSM uses a *Frequency Division Multiple Access* (FDMA) and *Time Division Multiple Access* (TDMA) scheme to maintain several communication links within one cell "in parallel." The available frequency band is slotted into *channels* of 200 kHz width. The time axis is organized in 8 cyclicly recurring time slots, numbered TN0, TN1, ..., TN7. A

channel

schematic frequency/time diagram is shown in Figure 2.2. The square blocks of 200 kHz by 7.5/13 ms in the frequency/time diagram are called *slots*. BTSs and MSs both transmit *bursts* of data within slots. Of the at most 147 bit per burst, no more than 114 bit are traffic data.

slot
burst

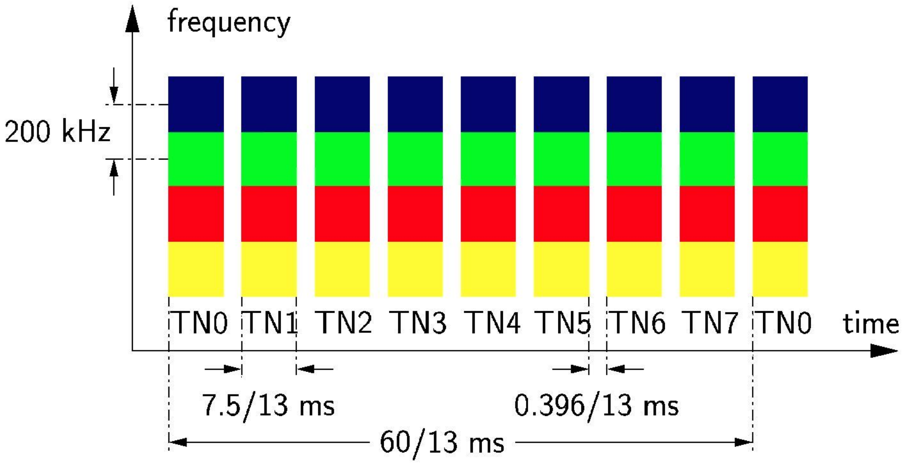


Figure 2.2: Frequency/time slot diagram

The direction from BTS to MS is the *down-link* and the reverse direction is the *up-link*, see Table 2.1. Up- and down-link channels are paired and referred to by their *absolute radio frequency channel numbers* (ARFCNs), which are defined separately within each variant of GSM. In GSM 900, for example, there are 124 (paired) channels numbered 1 through 124 and the associated frequencies are $890.0\,\text{MHz} + (200\,\text{kHz}) \cdot n$ for the up-link and $935.0\,\text{MHz} + (200\,\text{kHz}) \cdot n$ for the down-link part of the nth channel. The 374 channels in GSM 1800 are numbered from 512 up to 885, and the frequencies are $1710.0\,\text{MHz} + (200\,\text{kHz}) \cdot (n - 511)$ and $1805.0\,\text{MHz} + (200\,\text{kHz}) \cdot (n - 511)$ for the nth up- and down-link channel, respectively.

down-link
up-link
ARFCN

Recall from Section 2.2.1 that the first TRX of a sector usually offers capacity for up to six parallel (full-rate) speech connections and that additional TRXs typically offer seven to eight such connections. The first TRX has to use TN0 to broadcast cell organization information, among others. The channel used by the first TRX is therefore called *broadcast control channel* (BCCH). Additional cell management information is transmitted in one of the time slots TN2, TN4, or TN6. The remaining six slots are used for traffic. Although, the need for signaling increases with additional TRXs, this can often be handled by already installed signaling channels. Hence, some additional TRXs may transmit traffic

BCCH

data in all eight time slots. The channels used by any of the additional TRXs in a cell are called *traffic channels* (TCHs).

TCH

For full-rate speech telephony, the BTS and MS transmit a burst of encoded speech data of 114 bit in every eighth time slot. This results in a net speech rate of 13 kbps. (An option for half-rate service is specified in GSM Phase 2. Only about half the number of bits are transmitted, but due to a different encoding scheme the perceived quality is much better than half as good.)

Speech data is assembled in code words of 456 bit. If a code word is distorted at scattered rather than clustered positions, then the code allows for error detection and correction to a significant extent. Only every eighth bit of a code word is therefore transmitted in one burst, and each code word is spread over eight bursts. The applied scheme is referred to as *restructuring*, *reordering*, and *diagonal interleaving*.

Several hurdles have to be taken in order to receive a burst properly at a remote receiver. At reception, the signal has suffered from distortion in the modulator and demodulator, by the transmission medium, from noise sources, and from fading phenomena. In an urban environment, for example, the transmission medium suffers from shadowing, multipath propagation, and resulting delay spread. The noise sources comprise natural frequency radiation, human-made sources, and, most prominently, other transmitters within the GSM network itself.

SDMA

A cellular system like GSM uses by definition *Space Division Multiple Access* (SDMA) to the precious resource of radio spectrum. (In the sense that the same frequency can be reused in several cells, but not yet in the sense of reuse within the same cell, which is possible with beamforming antennas.) A cellular layout of the systems allows to support a high traffic density over large regions. The area covered by cells varies considerably. The "cell diameter" ranges from around 20 km or 35 km for Macro-cells in GSM 1800 and GSM 900, respectively, over a few hundred meters for Micro-cells to less than one hundred meters for (indoor) Pico-cells.

Between the number of channels available to a GSM operator and the number of TRXs operating in the network are often two orders of magnitude. Hence, the same frequency slot has to be used in parallel on several BTSs, and the only shielding against mutual interference comes from attenuation. Only *co-channel* and *adjacent channel interference*, i. e., signals from transmitters using the same channel or one of the two neighboring channels, have to be considered as serious intrasystem noise sources. According to the GSM specification, a burst has to be decoded properly if it is received at a signal level of at least 9 dB above noise, including intrasystem interference.

A number of measures is foreseen in GSM to counteract the generation of and the sensitivity to interference. We mention only those with a significant impact on the frequency planning problem.

Power Control is a feature of GSM that allows to dynamically adjust the transmission power to an appropriate level. A maximum emission power is specified for GSM transmitters. For hand-held mobiles this is 1 W or 2 W, depending on the GSM variant. In case less transmission power is sufficient to guarantee proper reception, the power can be reduced. Any power excess would only cause unnecessary interference and power consumption. A trade off between power control and hand-over has to be made: without the emission power being at the maximum level, a hand-over may be favorable to enter another cell, where a yet smaller power level suffices.

DTX

Discontinuous Transmission *(DTX)* is a feature of GSM that suppresses transmission if no data has to be transmitted. There is, for example, no need to transmit the (short) phases of silence within a conversation. The transmission is suspended and the receiving mobile generates a so-called *comfort noise* to make the suppression (almost) imperceptible. Triggered by a mechanism called *voice activity detection*, the transmission resumes as soon as the need arises. Figure 2.3(a) gives an illustration, where the pattern indicates the bursts. In case a channel is used as BCCH, a burst has to be transmitted in every time slot and DTX cannot be applied. (Hence, none of channels in Figure 2.3(a) is used as BCCH in the corresponding cell.)

SFH *HSN* *MAIO*

Slow Frequency Hopping *(SFH)* allows the transmission of consecutive bursts on different frequencies. Two variants exist. With *synthesized frequency hopping*, each TRX of a sector transmits successive bursts on different channels. The sequence, in which the available channels are switched, is determined by two parameters. One is the *Hopping Sequence Number* (HSN), selecting one out of 64 hopping sequences, and the other is the *Mobile Allocation Index Offset* (MAIO), which determines the starting point within the sequence. If more than one TRX is used for a sector, *baseband frequency hopping* can be applied alternatively. Each TRX uses a fixed channel, and the code words constituting a flow of communication are dispatched to changing TRXs, see Figure 2.3(b).

frequency diversity

Frequency hopping addresses two problems. The quality of a radio path is frequency dependent. *Frequency diversity* is obtained by varying the frequency, and the odds of always having a bad frequency for a particular radio link are thus reduced. This is of interest mostly to users who are moving slowly or not at all. For fast moving users, the diversity is caused by the movement. Another effect of changing the transmission frequencies is that successive bursts suffer from varying sources of interference. This phenomenon is called *interferer diversity*. The distortions of the received signals are less correlated, and this increases the probability of correcting the transmission errors. Notice, however, that in any case no hopping is applied at the broadcast control channel (BCCH) in time slot TN0.

interferer diversity

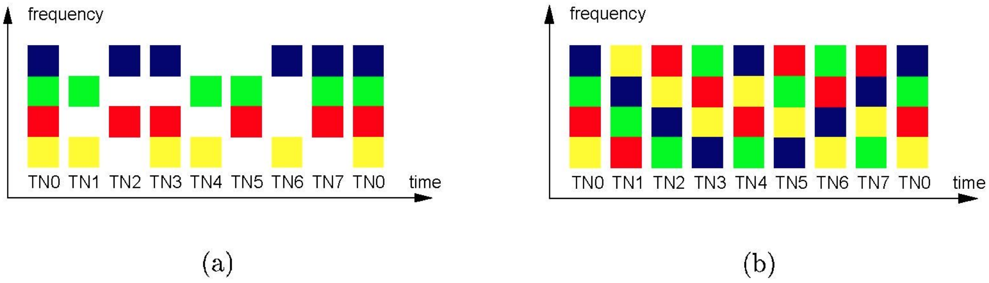

frequency
TN0 TN1 TN2 TN3 TN4 TN5 TN6 TN7 TN0
time
frequency
TN0 TN1 TN2 TN3 TN4 TN5 TN6 TN7 TN0
time
(a)
(b)

Figure 2.3: DTX and SFH

Although, it is not stated here explicitly, there are numerous parameters, which the individual GSM operator is able to change. The setting of those parameters also affects the efficiency of the radio interface.

2.3 Automatic Frequency Planning

As stated before, frequency planning is a key point for providing capacity and quality of service by fully utilizing the available radio spectrum in GSM. The automatic generation of a good frequency plan for a GSM network is a delicate task for which the three major building blocks are:

(i) a concise model

(ii) the relevant data

(iii) efficient optimization techniques

The importance of each prerequisite is explained in the following.

First, the automatic generation of a frequency assignment by a computer relies on the representation of all relevant aspects. Hence, a concise (mathematical) model of the frequency planning problem is necessary. On one hand, this model should be simple for the sake of easy handling. On the other hand, all information has to be captured which is necessary to accurately estimate a frequency plan's quality (without testing it in the real network). This is, for example, a point where the traditional model with hexagonal cell shapes fails, compare with Section 2.3.2. (The model still receives attention in the literature, however, because all relevant data is easily generated and planning based on this model is more easily accessible.) The spectrum of models currently in use is wide. It ranges from simplistic graph coloring models over graph-based models dealing with the maximization of satisfied demand or the minimization of interference to models building directly on signal predictions and looking at the probability of failed code word reception (frame erasure rate), see, for example, Koster [1999], Murphey, Pardalos, and Resende [1999], and Correia [2001, Section 4.2]. After preparing the ground in Section 2.3.1, we come back to models in Chapter 3.

Second, the concise model is futile unless the corresponding data is provided. The main difficulties here are related to data on radio signal levels. This data is needed in ample ways, for example, in order to estimate how much interference can occur between transmitters or to determine between which cells a hand-over can be supported. Details are discussed in Section 2.3.2.

Third, with a concise model and reliable data in hands, the task of producing a good frequency plan can be reduced to the problem of finding a solution to a mathematical optimization problem. Special software for this purpose is in demand. Operations Research has picked up this problem in the late 1960s and dealt with it steadily, compare Metzger [1970], Hale [1980], and Roberts [1991a]. The most progress has been made within recent years, accompanying the deployment and extension of GSM networks and often stimulated by close cooperations between research facilities and network operators or equipment manufacturers. We come back to planning algorithms in Chapter 4.

An overview on the frequency planning process in practice is given in Figure 2.4. Starting from the site data, including information on antenna locations, sectorizations, tilts, etc., as well as information on terrain, building structures, and sometimes even vegetation data, the signal propagation is predicted for all antennas. The results are used in calculation the cell areas. Linked to cell areas is the interference analysis, the hand-over planning, and the traffic estimation, each of which produces mandatory input data for the actual frequency assignment. Details on most of these items are given in the remainder of this chapter.

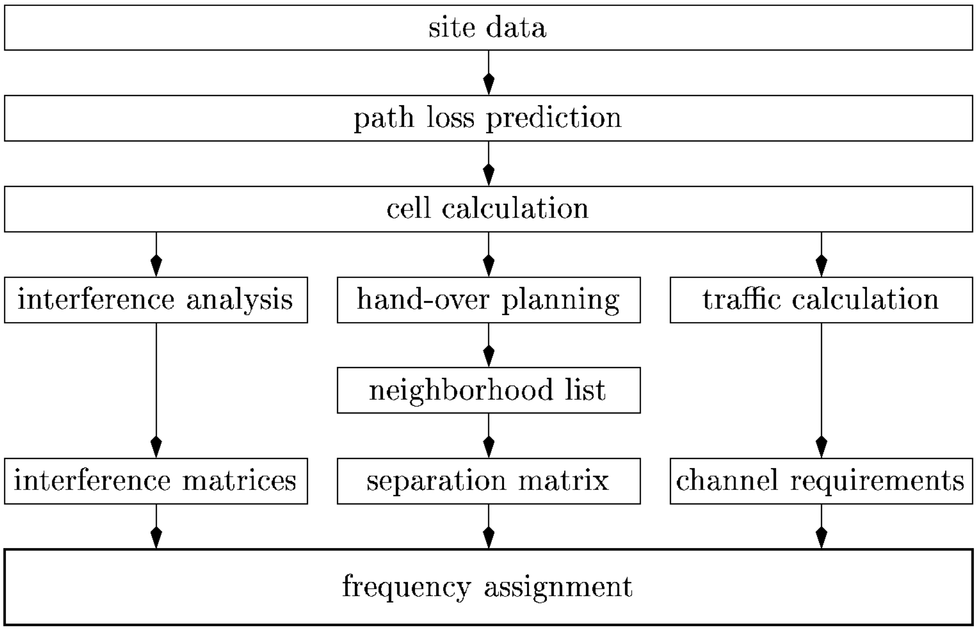


Figure 2.4: Frequency planning process

2.3.1 Objective and Constraints

Next, we explain the most important parameters to be taken into account for frequency planning. Those parameters must be present in the mathematical model. We use a small artificial but realistic example network called TINY for this purpose, see Figure 2.5.

site
sector
cell

TINY comprises three *sites*, named A, B, and C. Site A has three *sectors* with sector numbers 1, 2, and 3. Sites B and C have two sectors, numbered 1 and 2. Each sector of a site defines a *cell*. The numbers of elementary transceivers (TRXs) installed per cell are given in Table 2.3.

Cell	$A1$	$A2$	$A3$	$B1$	$B2$	$C1$	$C2$
TRXs	1	3	2	2	1	1	2

Table 2.3: Number of TRXs installed per cell

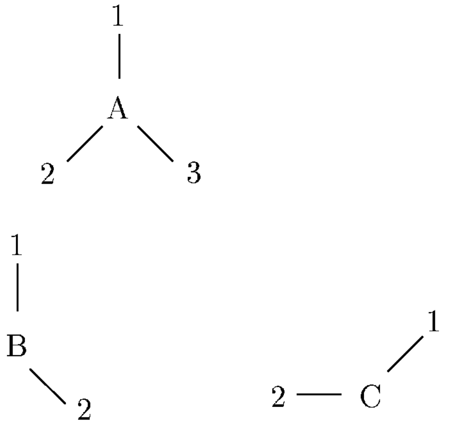


Figure 2.5: Network TINY

We assume that TINY is a GSM 1800 network and that the paired frequency bands 1750.0–1752.4 MHz and 1845.0–1847.4 MHz are available. The absolute radio frequency channel numbers (ARFCNs) of the corresponding thirteen channels are 711–723, and we call this set the *spectrum* of available channels.

spectrum

Due to technical and regulatory restrictions, some channels in the spectrum may not be available in every cell. Such channels are called *locally blocked*. Local blocking can be specified for every cell. We assume that channels 711 and 712 are blocked in cell $B2$, and that channel 719 is blocked in cell $C1$.

locally blocked

Each cell operates one *broadcast control channel* (BCCH) and possibly some dedicated *traffic channels* (TCHs). Two to three TCHs in a cell are common for urban areas today.

The difference of the ARFCNs of two channels is a measure for their proximity. Sometimes a restriction applies for a pair of TRXs on how close their channels may be. This is called a *separation* requirement, and its purpose is to ensure that the TRXs can transmit and receive properly or to support the preparation of call hand-overs between cells or to avoid strong interference. Separation requirements and locally blocked channels give rise to so-called *hard constraints*. None of them is allowed to be violated by an assignment.

separation

hard constraints

There are several sources of separation requirements. For example, if two or more TRXs are installed at the same site, *co-site separation* constraints have to be met. A co-site separation of 2 is assumed for all sites of TINY. Furthermore, if two TRXs serve the same cell, a *co-cell separation* constraint has to be met. The minimum co-cell separation is 3

co-site separation

co-cell separation

in each cell for TINY. In practice, this value may vary from cell to cell due to different technologies in use, but the values given here are typical.

→	$A1$	$A2$	$A3$	$B1$	$B2$	$C1$	$C2$
$A1$		●	●				
$A2$	●		●	●			
$A3$	●	●				●	●
$B1$		●			●		●
$B2$				●			●
$C1$			●				●
$C2$			●			●	

Table 2.4: Hand-over relation for TINY

During a hand-over, an ongoing call is passed from one cell to another. Technically speaking, the cellular phone switches from using a channel operated in the passing-on cell to a channel used by some TRX in the receiving cell. The hand-over relation is defined between all ordered pairs of cells and tells from which cell to which other cell a hand-over is possible. The hand-over relation for TINY is given in Table 2.4. A "•" at the intersection of a row and a column indicates that a call may be handed over from the cell listed in the row to the cell listed in the column.

Since the hand-over operation is a sensitive process, some separation between the channels in the two involved cells is required. Table 2.5 lists the minimum separation to support hand-over for TINY. The BCCH and all TCHs in the source cell have to be separated by at least 2 from the BCCH in the target cell. The BCCH and all TCHs in the source cell have to be separated by only 1 from the TCHs in the target cell. These values are again typical.

→	BCCH	TCH
BCCH	2	1
TCH	2	1

Table 2.5: Hand-over separation for TINY

co- and adjacent channel interference

In GSM, significant interference between transmitters may only occur if the same or adjacent channels are used. Correspondingly, we speak of *co-channel* and *adjacent channel interference*.

Interference in the up-link band may occur between mobile stations being served in different cells. Interference in the down-link band may occur between TRXs operated at different sites. Although the up-link is usually more critical in GSM than the down-link, the interference is specified for the down-link. The reason for this is the lack of appropriate ways to predict up-link interference. Already the prediction of down-link interference is intricate, see Section 2.3.2.

Interference relations do not have to be symmetric, i. e., if cell $B1$ interferes with cell $A1$, cell $A1$ does not necessarily also interfere with cell $B1$. And in case two cells interfere mutually, the ratings of the interference can be different. The ratings are normalized such that all interference values lie between 0.0 and 1.0. The co- and adjacent channel interference ratings for cell pairs in Tiny are specified in terms of affected cell area in Table 2.6. The upper number in each cell of the table refers to co-channel interference, and the lower number refers to adjacent channel interference. Blank spaces indicate that either no interference is predicted or interference is ruled out by separation requirements.

→	$A1$	$A2$	$A3$	$B1$	$B2$	$C1$	$C2$
$A1$							
$A2$				0.30 0.10	0.10 0.02		
$A3$						0.05 0.00	0.20 0.06
$B1$	0.01 0.00	0.25 0.09					0.25 0.08
$B2$							0.15 0.04
$C1$			0.01 0.00				
$C2$		0.06 0.01	0.12 0.03		0.25 0.08		

Table 2.6: Interference between cells in Tiny

The specification of interference for pairs of cells rather than for pairs of TRXs presupposes that all TRXs in a cell use the same technology, the same transmission power, and emit their signals via the same antenna. If this assumption does not hold, then a sector of a base transceiver station can be treated as the host for several "cells" within which the assumption holds. This is for example relevant if discontinuous transmission is applied, because the average interference caused by a TCH applying DTX is less than that of the BCCH, which is not allowed to apply DTX.

In case interference is very strong, it may not be possible to process calls. Interference should then be ruled out by means of separation requirements with minimum separation of one or two. A minimum separation of one excludes co-channel interference, because the involved pairs of TRXs may not use the same channel. A minimum separation of two excludes co- and adjacent channel interference. For TRXs installed at the same site, interference is generally ruled out by appropriate co-cell and co-site separation requirements. Table 2.7 displays a channel assignment for TINY, which incurs no co-channel interference and a total of 0.02 adjacent channel interference. The interference relations are also called *soft constraints* in the literature.

soft constraints

Cell	$A1$	$A2$			$A3$		$B1$		$B2$	$C1$	$C2$	
TRX	0	0	1	2	0	1	0	1	0	0	0	1
Channel	715	717	713	722	711	720	720	711	715	714	723	718

Table 2.7: Feasible assignment for TINY incurring interference

Because a frequency assignment is typically already installed in (parts of) the network when generating new plan, some of the existing assignments might have to be kept fixed. A TRX, for which the channel shall not be changed, is called *unchangeable*. Otherwise, we call it *changeable*.

(un-)changeable

All this data has to be represented adequately and in a computationally tractable fashion as a basis for automated frequency planning.

Our objective then is to find frequency plans incurring the least possible amount of overall interference, which we define as the sum over all interferences between pairs of TRXs. Although this figure reveals only a small part of the picture from a practical viewpoint, it has nevertheless proven effective in practice. We give one example for its inadequacy. Let us consider two frequency assignments incurring the same amount of overall interference. In one case, the entire interference occurs in one area, whereas in the other case the interference is scattered in small quantities. The second plan is certainly favored in practice, but the objective function does not show the difference. A few alternative optimization objectives (with other drawbacks) are discussed in Section 3.1.2.

The effects of discontinuous transmission (DTX) and slow frequency hopping (SFH) are not explicitly addressed here. How this can be done accurately during the planning process is, in fact, unclear, compare with Section 2.3.2. Common practice is to evaluate their impact by computer simulations once ordinary frequency planning has been performed. In case the outcome is not satisfactory, the planning process is repeated.

Our version of the *frequency assignment problem* is, thus, as follows:

> *Given are a list of TRXs, a range of channels, a list of locally blocked channels for each TRX, as well as the minimum separation, the co-channel interference, and the adjacent channel interference matrices.*
>
> *Assign to every TRX one channel from the spectrum which is not locally blocked such that all separation requirements are met and such that the sum over all interferences occurring between pairs of TRXs is minimized.*

We give a mathematical statement of this problem in Chapter 3. Next, we explain how the input data is generated with sufficient accuracy and in which way solving the above problem is embedded in practice.

2.3.2 Precise Data

The main difficulties concerning reliable data arise with respect to radio signal levels. Signal levels are provided through measurements in few cases only. In the other cases, the signal strength is predicted using wave propagation models. We sketch the most prominent tasks and the related problems to be tackled in preparation for algorithmic frequency planning.

Cells and Neighbors

The area, where a mobile station may get service from a particular sector of a BTS, is called *cell area*. Cell areas may overlap. The cell areas have to be estimated for at least two purposes.

cell area

One purpose concerns the provision of sufficient cell capacity. We are looking mostly at call blocking probability here, that is, the probability of not being able to get full service from the network due to lacking capacity at the radio interface. The cell capacity is provided by installing TRXs. How many TRXs are sufficient for a cell depends on the expected traffic load. More precisely, there have to be predictions (supplemented by measurements) of the peak communications traffic depending on the location. (A relevant measure for the peak traffic is the number of *busy-hour call attempts* (BHCA).) The traffic data is then related to the cell areas, resulting in a traffic estimate in Erlang per cell. Let λ_c denotes the traffic of cell c in Erlang, then the number of required communication channels m_c is determined from the well-known *Erlang-B formula*

BHCA

Erlang-B formula

$$B(\lambda_c, m_c) = \left(\sum_{k=0}^{m_c} \frac{\lambda_c^k}{k!}\right)^{-1} \frac{\lambda_c^{m_c}}{m_c!}$$

by setting m_c to the least possible value such that a blocking probability $B(\lambda_c, m_c)$ of 2%, say, is not exceeded. Then the smallest number of TRXs is chosen for cell c, which allows to support m_c simultaneous calls.

The other purpose of calculating the cell areas is to decide on the hand-over relations, that is, from which cell to which other cell a hand-over should be possible. This has to be settled in advance, because every cell broadcasts on the BCCH to which neighboring cells a hand-over is supported, and, correspondingly, hand-over separation requirements have to be observed during frequency planning. In order to hand an established communication link from one cell over to another, the mobile station has to be located in the overlap of the two cell areas.

Notice that the cell area does not only depend on the installation and configuration of the BTS and its sectors (including antenna height, tilt, transmission power, etc.) but also on the noise and interference from other BTSs. In addition to having a sufficiently strong radio signal at the receiver, this signal must also be sufficiently undistorted to be decoded correctly. This issue, however, is neglected in the following discussion of cell area prediction models.

The simplest model assigns each point to the cell with the strongest signal. The BTSs are assumed to be spaced out regularly on a grid and to have identical antenna configurations as well as identical transmission powers. The propagation conditions are taken to be isotropic. The result is a *hexagonal cell* pattern. In case the antennas radiate omnidirectionally, the BTS would be in the middle of a cell. In case a sectorization with 120 degree is used, the BTSs are located on the intersection of three cells, each of the sectors serving one of the cells, see Figure 2.6(a).

hexagonal cell

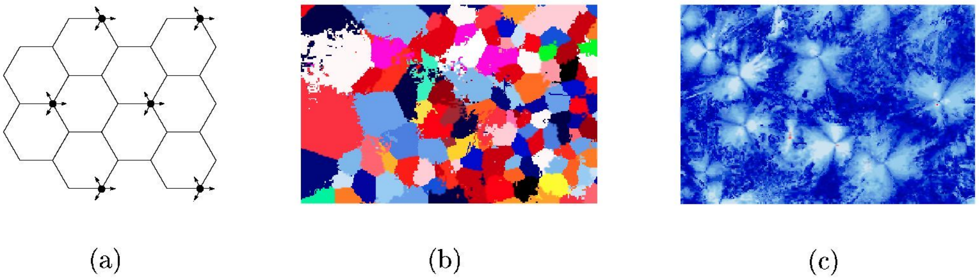
More precise cell models rely on realistic signal propagation predictions. For each sector, an attenuation diagram for the emitted radio signal is computed. For the following discussion, we assume that for each grid point of a regular mesh the signal strength of the surrounding base stations is known. Each of the grid points is a representative of its surrounding. Typical mesh sizes are 5 × 5 m (metropolitan), 50 × 50 m (urban), and 200 × 200 m (suburban & rural). Up to which distance base stations have to be considered is a matter of experience. In a GSM 1800 network, this distance can be in the order of up to 50 km.

best server

The *best server model* is commonly used today. Each grid point is assigned to the cell with the antenna providing the strongest signal. This results in a partition of the service region into cell areas without overlap, see Figure 2.6(b).

assignment probability

In the *assignment probability model*, the probability is estimated that a mobile station, located at a given grid point, is served by a given cell.



(a) (b) (c)

Images (b) and (c) are kindly provided by E-Plus Mobilfunk GmbH & Co. KG.

Figure 2.6: Cell models: hexagonal, best server, assignment probability

Every cell providing a signal of sufficient quality (see the discussion below) is considered as a potential server, and the probability of serving is computed by simulating the hand-over behavior of moving mobile stations. This model gives a better indication of the cell area than the best server model. So far, however, it is hardly used in the context of frequency planning. Typical applications are related to location-dependent tariffs like "local calls" within city borders or fixed network tariffs at home and its close surroundings. In Figure 2.6(c), the probability of being served by the cell with the strongest signal is color coded. The lighter the color gets, the higher is the probability of being served by one particular cell.

Interference Predictions

Several ratings of interference are conceivable. Area-based and traffic-based ratings are most often used in practice. The occurrence of interference is either measured or predicted. A purely distance-driven estimation of interference, as it is sometimes used in Operations Research literature, is unacceptable. There are, for example, drastic differences with respect to signal propagation between a flat rural environment and a metropolitan environment with narrow street canyons and irregular building structures, see, e. g., Kürner, Cichon, and Wiesbeck [1993] or Damosso and Correia [1999].

The standard procedure nowadays is to aggregate the grid-based signal predictions into interference predictions at a cell-to-cell level. For an area-based rating, this is typically done using the best server model, see above. Signals from cells are neglected if they are more than t dB below the strongest signal. All other signals are considered as potential interference. The way, in which area-based interference is accounted for, is depicted schematically in Figure 2.7. Two cells, A and B, are shown together with their cell areas. The cell areas are assumed to be determined according to the best server model. We focus on interference in cell A caused by cell B. The shaded portion of the cell A indicates the area, where cell B has a signal level of at most t dB less than cell A itself. The "interference" of cell B in cell A is taken as the number of shaded (distorted) pixels in cell A relative to the number of all pixels in the area of cell A. The same procedure, but with a different threshold value t, is used to determine adjacent channel interference. The converse direction is treated identical.

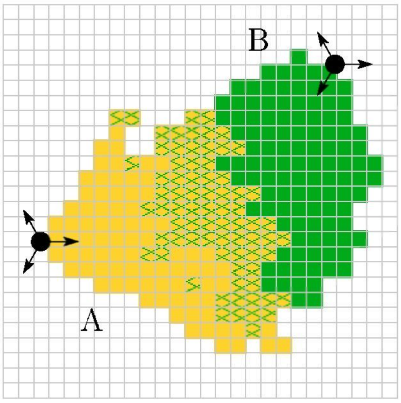


Figure 2.7: Area-based interference prediction

The GSM specifications state that a signal has to be decoded properly by a receiver if it is 9 dB above noise and interfering signals (and of sufficient strength). As a consequence, the value $t = 9$ is often used as threshold in practice. An investigation carried out by Eisenblätter, Kürner, and Fauß [1999] reveals, however, that a threshold value of 15 dB or even 20 dB often results in frequency plans, where interference is more evenly distributed and at a lower overall level. No satisfactory explanation for this observation is known so far.

Clearly, the accuracy of the interference predictions is a cornerstone for automated frequency planning. An analysis of how accurate interference predictions affect the quality of a resulting frequency plan is given by Eisenblätter, Kürner, and Fauß [1998], see also Correia [2001, Section 4.2.7]. Three interference predictions are computed for the same planning region on the basis of the best server model and using three different signal propagation prediction models.

free space model

- In the *free space* model, the propagation conditions of free space are assumed, but a decay factor of 1.5 rather than 1 is used. The increase of the factor from 1 to 1.5 (or the like) is taken as an empirical value between the decay factor when only the direct ray is taken into account (resulting in a decay factor of 1) and the decay factor observed in a two ray model, see, e. g., Kürner and Fauß [1994]. In the two-ray model, the interaction between the direct ray and a reflected ray results in a decay factor of 2 for distances larger than a specific threshold.

race model

- The Modified Okumura-Hata *race* predictor bases on an 1800 MHz extension of the basic path loss equation as described in Damosso and Correia [1999]. Land use information is used by means of empirical correction factors for each land use class. Terrain variations are taken into account by using an effective antenna height. Topographical obstacles are treated as knife-edges, that is, infinitely long, straight "razor blades," for which a closed, simple formula for the diffraction is known.

eplus model

- The *eplus* propagation prediction model, see Kürner, Fauß, and Wäsch [1996], is the most sophisticated approach used in the comparison. The model consists of a combination of several propagation models like COST 231-Walfisch-Ikegami, Maciel-Xia-Bertoni, and Okumura-Hata. It is developed for GSM 1800 and calibrated with numerous measurements in the network of E-Plus.

Ranking these wave propagation prediction models has its difficulties. The crucial question is how to compare assignments computed on the basis of different predictions without implementing the assignments into the live network and performing measurements. In the approach taken by Eisenblätter *et al.* [1998], each assignment's interference is determined according to all three interference predictions. The findings are as follows.

- The assignments computed on the basis of the predictions from the eplus model have relatively little interference according to all three predictions.
- The assignments computed on the basis of the predictions from the free space model have decent interference ratings according to the race predictions, but are mediocre according to the eplus predictions.
- For the assignments computed on the basis of the race predictions the worst picture is obtained. They are mediocre to bad according to the two other predictions.

In view of this, the eplus model is ranked above the free space model, which, in turn, is ranked above the race model (in this particular context). In total, varying the signal propagation predictor shows a larger impact on the frequency assignment quality than choosing among the different frequency planning heuristics considered by Eisenblätter *et al.* [1998], which are similar to those described in Chapter 4 and Section 5.1.2.

Effects of DTX and SFH

The GSM features of discontinuous transmission (DTX) and slow frequency hopping (SFH) both address the problem of interference either by reducing interference itself (DTX) or by reducing the impact of interference (SFH).

Neither of these features is explicitly addressed within our frequency assignment model, but it is possible to incorporate their effects into the interference ratings. Nielsen and Wigard [2000] and Majewski, Hallmann, and Volke [2000] propose different ways to do so, both being validated using GSM simulators. Nielsen and Wigard [2000] introduce two parameters called *hopping gain* and *load gain* by whose product the interference rating is scaled. The setting of the parameters depends on the load of each cell, a voice activity factor, and the number of channels to hop on, among others. Majewski *et al.* [2000] introduce *pre factors* and *post factors* in order to scale interference, but do not provide the full details.

hopping gain
load gain

pre factors
post factors

Björklund, Värbrand, and Yuan [2000] optimize the hopping sequence for each cell. This sequence is determined by the hopping sequence number (HSN) and sequence starting point (MAIO).

2.3.3 Practical Aspects

Since the introduction of GSM, operators have steadily increased their network's coverage and capacity. This typically involves installing additional TRXs and providing them with channels. Hence, the frequency plan has to be adjusted. The same holds if the transmission characteristics of a BTS change.

Installing a new frequency plan is not as simple as it may sound at first. In Germany, for example, the operator has to submit the frequency plan to a governmental regulation office and ask for approval. This approval is given if the frequency plan adheres to the bilateral agreements on channel use along national borders and if no interference with other radio systems operating in the same frequency band is expected. The restrictions from both sources are recorded as locally blocked channels.

The former are known in advance, but the latter are often only revealed through rejection. The turn around time for such an approval is in the order of a few weeks.

Changes in the frequency plan take effect at the BTSs. In "old times," the channels had to be adjusted manually at the combiners. Nowadays, remotely tunable combiners may be used. These allow to change the channels through the OSS, but this convenience comes at the expense of less effective combiners. In principle, a TCH can be changed while the cell is in operation as long as the corresponding TRX is not in use. Changing a BCCH, however, requires to shut down the cell completely for a couple of minutes. Therefore, changes in the frequency usage are mostly performed at night times.

Another problem is that the effects of the changes are not easily assessed. Extensive, time-consuming quality measurements campaigns could be performed, but much rather a "sit and wait" strategy is adopted: measurements by the Operation and Maintenance Center (OMC) of the rate of quality-driven hand-overs and an increase in customer complaints substitute the explicit quality assessment. Common to both alternatives is that they require users which are getting service or unsuccessfully try to get service. This happens to a sufficient extent only at the next day.

The way in which frequency planning is done differs from operator to operator. Some operators divide their service area among regional offices, which act more or less independently. For example, E-Plus operates five regional offices. Between the regional offices, the channel use along the regional border is settled through agreements similar to bilateral agreements for national borders. Obviously, regional borders (these are the ones an operator may choose) should be in areas with little telecommunications traffic, where planning is simple even with additional restrictions.

Even if operators are confident in the overall reliability of the frequency planning process, they try to change the BCCH assignment rather seldom. Recall in this context that solving the combinatorial optimization problem of finding a good frequency plan is merely one important step in this process. Other, equally important ones, are maintaining up to date and sufficiently detailed data about terrain and buildings as well as generating accurate interference predictions.

Some GSM operators split their available spectrum into two separate parts, one for BCCHs, the other for TCHs. This is called *band split*. The reasoning behind performing a band split is to be able to plan the TCHs (almost) independently from the BCCHs. Table 2.8 shows an assignment for TINY, which is compatible with splitting the spectrum of 711–723 into a BCCH-band of 711–716 and a TCH-band of 717–723.

band split

Cell	$A1$	$A2$	$A3$	$B1$	$B2$	$C1$	$C2$	$A2$		$A3$	$B1$	$C2$
TRX	0	0	0	0	0	0	0	1	2	1	1	1
Channel	711	716	713	711	713	711	715	719	723	721	721	718

Table 2.8: Assignment respecting a band split

Planning BCCHs from a separate, often relatively large band is one way to protect these channels against interference. Moreover, only TCHs perform DTX, which leads to load-dependent interference among the corresponding TRXs. Again, by using separate bands, the BCCHs are shielded against this load-dependency. The only conflicts may arise where the two bands meet and adjacent channel interference from TCHs extends into the BCCH-band. Since network expansion is mostly capacity enhancement provided by additional TRXs, the capacity increase is typically achieved at the expense of additional interference in the TCH-band.

While extensions and minor changes of a frequency plan are performed regularly, major changes or even replanning the entire service area are treated with great precaution. Some operators are willing to replan about once a year, others use even larger time intervals.

The planning proceeds in steps regardless of whether a small or a large change is envisaged. TRXs eligible to changes are selected, and one or a few frequency plans are computed. These plans are analyzed thoroughly according to various criteria. Our objective function of minimizing the overall interference is just a coarse approximation of that. If none of the plans is considered good enough, the radio engineer may change technical characteristics of BTSs, such as the direction of the sectors, in order to decrease interference potential. New frequency plans are computed. This process is iterated until a decent frequency plan is identified. During the iterations, fast heuristics for frequency planning are favored because of their short running times. If a final plan is to be determined, the use of more time consuming, elaborate methods is acceptable.

Fast planning methods are presented in Chapter 4, and one selected example of a more time consuming method is described in Section 5.1.2. Notice that the heuristic planning methods discussed here and elsewhere typically address the problem of assigning channels to several hundreds of TRXs at once. In case of a minor network expansion, the situation is different: only a few TRXs have to be assigned, up to a hundred, say, while taking restrictions from the presently installed assignment into account. In this case, branch-and-cut methods can often find the optimal assignment in reasonable time. Using standard tricks of Integer Programming the integer linear program (3.6) can be solved effectively, see the work of Koster [1999] and that of Jaumard, Marcotte, and Meyer [1999].

CHAPTER 3

Mathematical Models

In the following, we translate the informal statement of the GSM frequency planning problem from Section 2.3.1 into a mathematical model. For convenience, we restate the problem:

> *Given are a list of TRXs, a range of channels, a list of locally blocked channels for each TRX, as well as the minimum separation, the co-channel interference, and the adjacent channel interference matrices.*
>
> *Assign to every TRX one channel from the spectrum which is not locally blocked such that all separation requirements are met and such that the sum over all interferences occurring between pairs of TRXs is minimized.*

Our mathematical model is presented in Section 3.1. In the context of GSM frequency planning, similar models are used by Duque-Antón and Kunz [1990]; Duque-Antón, Kunz, and Rüber [1993]; Carlsson and Grindal [1993]; Plehn [1994]. During the late nineties, this model became popular among researchers as well as practitioners, see Koster [1999, Section 2.6] and Correia [2001, Section 4.2.5], for example. A few competing models are addressed in Section 3.1.2. The computational complexity of solving our model of the frequency assignment problem is studied in Section 3.2. It turns out that finding a feasible solution is $\mathcal{NP}$-complete, and even if that were simple, finding (close to) optimal solutions would remain $\mathcal{NP}$-hard. Finally, two reformulations of the frequency assignment problem as integer linear programs are given in Section 3.3. The mathematical notions used in the following are explained in Appendix A.

3.1 The Model FAP

The objective of minimizing the overall interference is blind to the "direction" of interference, that is, whether the use of a channel in a cell causes interference somewhere or whether the channel itself suffers from interference. The mathematical formulation of our frequency planning problem is therefore undirected, and we simply add the interference ratings given for the two directions into a single value.

carrier
spectrum
blocked channel
available

Let (V, E) be an undirected graph. The vertices of the graph are also called *carriers* and represent the TRXs. The *spectrum* C is a finite interval in $\mathbb{Z}_+$, the set of nonnegative integers, representing the range of channels. For every carrier $v \in V$, a set $B_v \subsetneq C$ of *blocked channels* is specified. The channels in $C \setminus B_v$ are called *available* at carrier v. B_v may be empty.

separation
co-channel
adjacent channel

Three functions, $d\colon E \rightarrow \mathbb{Z}_+$, $c^{co}\colon E \rightarrow [0,2]_{\mathbb{Q}}$, and $c^{ad}\colon E \rightarrow [0,2]_{\mathbb{Q}}$, are specified on the edge set. For an edge $vw \in E$, $d(vw)$ gives the *separation* necessary between channels assigned to v and w. $c^{co}(vw)$ and $c^{ad}(vw)$ denote the *co-channel* and *adjacent channel interference*, respectively, which may occur between v and w. (Both functions map into the interval $[0,2]_{\mathbb{Q}}$ rather than into $[0,1]_{\mathbb{Q}}$ because of the symmetrization mentioned above. This may be remedied by scaling if desired.)

carrier network
assignment
feasible

We refer to the 7-tuple $N = (V, E, C, \{B_v\}_{v\in V}, d, c^{co}, c^{ad})$ as *carrier network* or *network*, for short. A *frequency assignment* or simply an *assignment* for N is a function $y\colon V \rightarrow C$. An assignment is *feasible* if every carrier $v \in V$ is assigned an available channel and all separation requirements are met, that is, if

$$\begin{aligned} y(v) \quad &\in C \setminus B_v && \forall v \in V, && (3.1)\\ |y(v) - y(w)| \quad &\geq d(vw) && \forall vw \in E. && (3.2) \end{aligned}$$

list coloring

Feasible assignments are a generalization of list colorings and are related to T-colorings of graphs in the following way. For a *list coloring* problem, a graph and lists of colors for every vertex are given. The task is to find a vertex coloring for the graph such that every vertex receives a color from its list and such that no two adjacent vertices receive the same color, compare Erdős, Rubin, and Taylor [1979]. Since an available channel has to be picked for every carrier, feasible assignments are list colorings.

T-coloring

T-colorings are introduced by Hale [1980]. Given an undirected graph $G = (V, E)$ and nonempty finite sets $T(vw)$ of positive integers for all edges $vw \in E$, a *T-coloring* of G is a labeling f of the vertices of G with nonnegative integers such that $|f(v) - f(w)| \not\in T(vw)$ for all $vw \in E$.

list T-coloring

A frequency assignment has to meet list coloring as well as T-coloring constraints in order to be feasible. Such *list T-coloring* are first studied by Tesman [1993].

The definition of a carrier network is illustrated using the scenario Tiny from Section 2.3.1. The vertex set is $V = \{A1_0, A2_0, A2_1, A2_2, A3_0, A3_1, B1_0, B1_1, B2_0, C1_0, C2_0, C2_1\}$. The edge set can be identified from Table 3.1 as all pairs of vertices where at least one of the functions d, c^{co}, or c^{ad} is nonzero.

Table 3.1 shows the minimum required separation, the co- and adjacent channel interference in full detail. The lower left-hand part displays the nonzero separation requirements, whereas the upper right-hand part lists the co-channel interference on top of the adjacent channel interferences. The symbol "∞" is used, where interference cannot arise due to separation requirements.

	$A1_0$	$A2_0$	$A2_1$	$A2_2$	$A3_0$	$A3_1$	$B1_0$	$B1_1$	$B2_0$	$C1_0$	$C2_0$	$C2_1$
$A1_0$		∞ ∞	∞ ∞	∞ ∞	∞ ∞	∞ ∞						
$A2_0$	2		∞ ∞	∞ ∞	∞ ∞	∞ ∞	∞ ∞	∞ 0.19	0.10 0.02			
$A2_1$	2	3		∞ ∞	∞ ∞	∞ ∞	∞ ∞	∞ 0.19	0.10 0.02			
$A2_2$	2	3	3		∞ ∞	∞ ∞	∞ ∞	∞ 0.19	0.10 0.02			
$A3_0$	2	2	2	2		∞ ∞				∞ ∞	∞ ∞	∞ ∞
$A3_1$	2	2	2	2	3					∞ ∞	∞ ∞	∞ 0.09
$B1_0$		2	2	2				∞ ∞	∞ ∞		∞ ∞	∞ 0.08
$B1_1$		1	1	1			3		∞ ∞		∞ ∞	∞ 0.08
$B2_0$							2	2			∞ ∞	∞ 0.12
$C1_0$					2	2					∞ ∞	∞ ∞
$C2_0$					2	2	2	2	2	2		∞ ∞
$C2_1$					2	1	1	1	1	2	3	

Table 3.1: Separation and interference for Tiny

The spectrum C is the set $\{711, \ldots, 723\}$. The local blockings are listed in Table 3.2.

	$A1_0$	$A2_0$	$A2_1$	$A2_2$	$A3_0$	$A3_1$	$B1_0$	$B1_1$	$B2_0$	$C1_0$	$C2_0$	$C2_1$
$B.$	—	—	—	—	—	—	—	—	711, 712	719	—	—

Table 3.2: Local blockings for Tiny

Four assignments for the network are shown in Table 3.3. The assignment y^1, which is the same as that in Table 2.7, is feasible and incurs no co-channel interference, but a total of 0.02 adjacent channel interference. The assignment y^2 is also feasible and incurs no interference at all. The assignments y^3 and y^4 are both infeasible. The local blocking of channel 719 for $B2_0$ is not obeyed by y^3, whereas the required separation of 2 between the channels for the carriers $A2_3$ and $A3_2$ is not achieved in y^4.

	$A1_0$	$A2_0$	$A2_1$	$A2_2$	$A3_0$	$A3_1$	$B1_0$	$B1_1$	$B2_0$	$C1_0$	$C2_0$	$C2_1$
y^1	715	717	713	722	711	720	720	711	715	714	723	718
y^2	713	728	722	725	711	720	711	720	713	713	715	718
y^3	713	730	724	727	711	715	714	717	712	713	719	722
y^4	712	711	717	715	713	716	719	720	713	711	722	723

Table 3.3: Assignments for Tiny

Our objective is to determine a feasible assignment that minimizes the sum of co- and adjacent channel interferences.

Definition 3.1. *Given a carrier network N, we call the optimization problem*

$$\min_{y \text{ feasible}} \sum_{\substack{vw\in E:\\ y(v)=y(w)}} c^{co}(vw) \quad + \sum_{\substack{vw\in E:\\ |y(v)-y(w)|=1}} c^{ad}(vw) \tag{FAP}$$

the frequency assignment problem *FAP*.

Focusing on interference minimization is justified if feasible assignments can be produced sufficiently well. This is the case for (most of) the planning scenarios we are interested in, but not necessarily in general, as is shown in Section 3.2. The example of planning a capacity extension for an already congested area underlines that even in practice-relevant situations this assumption will not always be satisfied. Two questions naturally arise in such a situation: Is no feasible extension found because none exists? And if so, how many of the established assignments have to be changed the least (and which are the ones) in order to obtain a feasible plan for the extended network?

Both questions are linked to interesting lines of research. The first question has close connections to the "min-span problem," compare with Section 3.1.2, and the second one is related to the "minimum blocking problem." The latter problem addresses the minimization of the call blocking probability per cell, i. e., the maximization of the portion of the specified demand per cell that is satisfied. Neither of those questions is pursued here, because, as already mentioned, feasibility is almost always easily obtained for the test instances at our disposal, see Section 5.1.1.

We point the interested reader to the surveys given by Koster [1999] and Murphey *et al.* [1999].

Our objective function asks for minimizing the sum of all interferences. One consequence is that we will exchange a number of small interferences for one big interference between two carriers as long as the total interference is reduced. Not much effort is required to come up with an example where this is inappropriate. Such examples, however, do not seem to be typical for GSM frequency planning practice, and the objective of minimizing the total interference is widely accepted in practice. In addition, our model does allow to counteract undesired exchanges in at least two ways. One method is explained as "tightening the separation" in detail in Section 4.1.2. Roughly speaking, high interferences between pairs of carriers are ruled out by introducing extra separation requirements. The other method, we have in mind, transforms the specified interference ratings by applying a monotonously increasing function. This changes the trade off between many small interferences and one large interference in favor of the many small ones.

Another comment on our model is the following. Each vertex of the carrier network corresponds to an individual TRX in the GSM network. As an alternative, one might identify "equivalent" TRXs per cell and represent those by one vertex. Two TRXs of a sector would be considered equivalent if they shared the same planning requirements. The number of channels to assign to a vertex would depend on the vertex, and within the set of channels assigned to a vertex the minimum co-cell separation would have to be met. Both alternatives are equivalent in the mathematical sense, of course. But the latter may be more appealing if synthesized frequency hopping is applied and more channels can be assigned to a cell than there are TRXs. In our model, this is mimicked by introducing extra carriers for those cells.

3.1.1 Variants

There are some straightforward generalizations of FAP. The first one, we present, addresses the question of finding an interference-minimal assignment extending a given partial assignment. In the second variant, an assignment is given and the objective is to find a cost-minimal assignment, when, in addition to interference, cost for changing the channel of a carrier is accounted for. The third variant generalizes the first and the second one. An assignment is given together with a set of carriers, which are not to be changed. Furthermore, a cost function on the changeable carriers is given, and if a carrier's channel is changed, then the corresponding cost is accounted for in the objective function. The second and the the third variant are both examples for *bi-criteria optimization* problems. The general arguments about the trade off between the two competing optimization goals apply here, too.

bi-criteria optimization

partial assignment extension

FAP$_p$ A *partial assignment* $p\colon P \to C$ with $P \subseteq V$ is given. The carriers in P are already assigned. A frequency assignment y *extends* p if $y(v)$ equals $p(v)$ for all carriers $v \in P$. Our first variant is the optimization problem

$$\min_{\substack{y \text{ feasible} \\ y \text{ extends } p}} \sum_{y(v)=y(w)} c^{co}(vw) \quad + \sum_{|y(v)-y(w)|=1} c^{ad}(vw). \qquad (\mathsf{FAP}_p)$$

If there is no partial assignment specified, this problem is just the ordinary frequency assignment problem FAP. Conversely, this can also be expressed as an ordinary FAP by setting $B_v = C \setminus \{p(v)\}$ for all carriers in P.

reassignment penalty

FAP$_r$ A frequency assignment $y_{pre}\colon V \to C$ is supplied, and a *penalty* has to be paid if any of the channels is changed. The individual reassignment penalties are specified by a mapping $r\colon V \to \mathbb{Q}_+$. The penalty for changing the channel of a carrier is independent of the amount by which the new and the old channel differ, because important is merely whether the assignment is changed. The objective is to minimize the total cost of a frequency assignment, where the savings in interference are traded off against the spending for reassigning carriers. Our second variant is the optimization problem

$$\min_{y \text{ feasible}} \sum_{y(v)=y(w)} c^{co}(vw) + \sum_{|y(v)-y(w)|=1} c^{ad}(vw) + \sum_{y(v)\neq y_{pre}(v)} r(v). \qquad (\mathsf{FAP}_r)$$

The ordinary frequency assignment problem FAP is obtained in case nothing is charged for reassigning a carrier.

FAP$_{pr}$ Given are a frequency assignment $y_{pre}\colon V \to C$ and a set $P \subseteq V$ of carriers, which are supposed not to be altered. Moreover, penalties for reassigning each carrier v are specified by a function $r\colon V \to \mathbb{Q}_+$. (Because the carriers in P cannot be changed, the values $r(v)$ for $v \in P$ are irrelevant. These values are given merely for notational convenience.) In order to give a mathematical formulation, consider

> the partial assignment $p\colon P \to C$ obtained from y_{pre} by letting $p = y_{pre}|_P$, i. e., p maps every carrier v contained in P to the channel $p(v)$ and is undefined on all carriers not in P. Our third variant is the optimization problem
>
> $$\min_{\substack{y \text{ feasible} \\ y \text{ extends } p}} \sum_{y(v)=y(w)} c^{co}(vw) + \sum_{|y(v)-y(w)|=1} c^{ad}(vw) + \sum_{y(v)\neq y_{pre}(v)} r(v). \tag{FAP_{pr}}$$
>
> A similar effect can be achieved by using the model FAP_r and setting $r(v)$ for all $v \in P$ to an arbitrarily high value (exceeding all other penalties by far).

By specializing FAP_{pr}, the two first variants and FAP itself can be obtained: in case $r(v) = 0$ for all penalties, FAP_{pr} reduces to FAP_p; in case $P = \emptyset$, FAP_{pr} simplifies to FAP_r; and in case both previous restrictions hold, then FAP_{pr} turns into FAP.

Conversely, an equivalent instance of FAP exists for every instance of FAP_{pr} for which the maximum penalty for reassigning is bounded by 2. The penalties for reassigning in FAP_{pr} are accounted for as co-channel interference with additional carriers in FAP. The bound of 2 comes from the definition of a carrier network, in particular, from the maximum admissible amount of co-channel interference.

Let $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$, $P \subseteq V$, $r\colon V \to [0,2]_{\mathbb{Q}}$, and $y_{pre}\colon V \to C$ be an instance of FAP_{pr}. We define a FAP instance $\tilde{N}$ as follows. An extra carrier is introduced for every $f \in C$. The edges in $\tilde{N}$ are all edges from N plus all vf for $v \in V$, $f \in C$ with $r(v) \neq 0$ and $y_{pre}(v) \neq f$. The spectrum is unchanged and so are the locally blocked channels for all $v \in V$. We set $B_f = C \setminus \{f\}$ for the new carriers $f \in C$. The edge labelings d, c^{co}, and c^{ad} are extended to the new edges with zeros for $\tilde{d}$ and $\tilde{c}^{ad}$ and with $\tilde{c}^{co}(vf) = r(v)$. (The transformation can be generalized to the case where $r(v) > 2$ for some $v \in V$. More than $|C|$ additional carriers become necessary, which may lead to carrier networks of significantly increased size.)

Notice that if changing the channel allocation of a TRX is penalized too heavily in comparison to the interference it would incur without a change, then the character of the optimization problem may change. In this context, optimal solutions for problem with up to one thousand TRXs are reported by Aardal, Hurkens, Lenstra, and Tiourine [1996] on test instances which do not arise from GSM frequency assignment. Nevertheless, these findings are likely to hold for GSM instances, too.

3.1.2 Alternative Models

The scientific study of frequency planning started in the late sixties. The presentation of Metzger [1970] is often seen as the starting point. Ten years later, Hale [1980] published a classification of frequency planning problems and their applications. To our best knowledge, these early works do not address problems that are equivalent to frequency planning for GSM. This changed with the deployment of GSM networks.

Several different problem types are subsumed under the general terms frequency planning, frequency assignment, and channel allocation. In our following discussion of alternative models, we focus solely on models which we consider relevant for GSM frequency planning. More comprehensive treatments of frequency planning in general are given by Koster [1999] and by Murphey *et al.* [1999]. Notice also that we only address *Fixed Channel Allocation* (FCA), where the channel demand per cell is fixed. Two other types of problems with varying demands are briefly explained and references are given.

FCA

DCA

Dynamic Channel Allocation (DCA) considers varying traffic profiles and, consequently, also varying channel demands per cell. Channels have to be assigned on request. GSM does not support dynamic channel allocation. The planning is done statically for the busy hour, i. e., with the peak traffic in view although the traffic load is clearly changing over time. Studies on the potential impact of using DCA for GSM networks are described by Kennedy, Vries, and Koorevaar [1998], for example. More general studies of DCA are performed by Malesińska [1997], Grace, Burr, and Tozer [1998], and Grace [1999].

HCA

A mixture of FCA and DCA is called *Hybrid Channel Allocation* (HCA). Some basic demand is covered by solving an FCA problem, and additional demand is handled dynamically in the sense of DCA. Hybrid channel allocation is studied by Malesińska [1997], for example. HCA is also not supported by GSM.

We now turn back to FCA and alternative models for FAP. The actual quality of a frequency plan is at best predictable with very time-consuming simulations using a GSM link-level simulator. In practice, the quality will typically be only observed once a frequency plan is installed. The model underlying the planning process should express the anticipated quality reasonably well and be pessimistic in doing so. It would be unrealistic to expect that the model is completely accurate. Consequently, there is a range of different and yet reasonable models focusing on different quality aspects of a frequency plan.

min-span model

In the *min-span model*, frequency planning is handled as a generalized graph coloring problem (see Tesman [1993]). Each vertex has a set of available "colors" and the edges are labeled with minimum separation requirements. The result is a mixture of list and T-coloring as explained in Section 3.1. The objective is to find a coloring of the vertices, which satisfies the list as well as the T-coloring constraints and which uses "colors" from an as narrow range as possible. The *span* is the difference between the largest and the smallest "color" used.

span

Interference information is not directly taken into account within this model. Instead, when setting up the underlying graph and fixing the minimum required separation along each edge, a distinction is made between "acceptable" and "unacceptable" interference. This is typically done by means of a threshold value. For example, co-channel interference above the threshold is ruled out by introducing a separation requirement of at least 1. Interference below the threshold value is neglected altogether. The min-span model is not directly applied in GSM frequency planning, but occurs as the core problem in minimizing the maximal interference. This is the next approach described below.

A detailed discussion of the min-span model and an extensive survey of related mathematical results and algorithmic developments is given by Murphey *et al.* [1999], see also Koster [1999, Section 2.4] and FAP web [2000]. Numerous authors have addressed the problem of computing lower bounds on the required span. We mention only a few important contributions: Gamst [1986] provides a bound basing on cliques in the underlying graph; Raychaudhuri [1994], Roberts [1991b], Smith and Hurley [1997] obtain bounds from solving the Traveling Salesman Problem (see Section 6.4.2) on subgraphs; and Janssen and Kilakos [1996] as well as Janssen and Kilakos [1999] derive bounds from polyhedral studies.

In GSM frequency planning practice, the *min max interference* approach has been popular until recently. At first, similar input data is generated as required for the specification of our carrier network. In a second step, the interference predictions are the basis for increasing separation requirements in order to prevent the occurrence of severe interference. As indicated in the context of the min-span model, a tentative threshold value is chosen to separate "acceptable" from "unacceptable" interference. "Unacceptable" interference is transformed into additional separation requirements, and "acceptable" interference is simply ignored.

min max interference

The result is a min-span problem with local blockings recorded at the vertices and with minimum required separation recorded at the edges. An attempt is made to solve this min-span problem. Two outcomes are possible. In one case, a feasible solution is generated. This corresponds to a frequency assignment obeying all imposed conditions. In the other case, no solution is found. The tentative threshold value is increased, which results in less “unacceptable” interference and, correspondingly, in fewer additional separation requirements. The planning process is started over with the new input.

The final threshold value is the result of repeatedly increasing or decreasing the value with the goal of finding an assignment that barely fits into the available spectrum. In practice, the threshold value is driven up beyond desirable limits by capacity-related interference problems in metropolitan areas. As a consequence, interference not exceeding this threshold becomes generally invisible to the planning process. Although avoidable outside of the critical areas by careful planning, many interference situations are not resolved because they have become imperceptible. Alternatively, the planning radio engineer may choose (potentially many) location-dependent threshold values. Either way, this planning style proved unattractive in practice.

min max local interference

Minimizing the maximal interference experienced by a TRX can be done on the basis of our carrier network representation as well, but the use of a directed version is also conceivable. The difference to our frequency planning problem is primarily in the definition of the objective function. The goal here is to keep the maximal interference experienced by a TRX as low as possible. To that end, the impact of the interfering TRXs is recorded for every TRX and the maximum is determined. This maximum is to be minimized. Most prominently, this approach is pursued by Fischetti, Lepschy, Minerva, Jacur, and Toto [2000], who also give satisfactory computational results for realistic planning problems with several hundreds of TRXs.

3.2 Computational Complexity

The results presented in this section are the justification for our focus in Chapter 4. Resorting to heuristic methods for solving FAP would not be (easily) justified if reasonable approximations of optimal frequency assignments were computable in polynomial time. We show in Proposition 3.11 that this is not the case.

3.2.1 Preliminaries

Throughout the following discussion, we assume basic familiarity with the concepts of computational complexity. Several textbooks contain introductions, see, for example, Garey and Johnson [1979], Bovet and

Crescenzi [1994], Papadimitriou [1994] as well as Cormen, Leiserson, and Rivest [1990, Chapters 36, 37]. With respect to the complexity of approximation, we recommend the book of Ausiello, Crescenzi, Gambosi, Kann, Marchetti-Spaccamela, and Protasi [1999], which is also our primary reference here. Nevertheless, we recall the basic definitions, which are required in the following, from the literature.

We start with the complexity of an algorithm. (See any of the above mentioned books for a discussion of the subtleties of defining "algorithm.") For an algorithm A, let $\hat{t}_A(x)$ denote the number of steps executed by A on input x. The *worst case running time* of A is defined as $t_A(n) = \max\{\hat{t}_A(x) \mid x : |x| \leq n\}$. The size $|x|$ of an instance x depends on the encoding scheme. We assume here that a "reasonable" compact binary encoding of the instances is used, see the discussion in Garey and Johnson [1979, Chapter 2] or Grötschel, Lovász, and Schrijver [1988, Section 1.3], for example. Algorithm A has a *running time complexity* $\mathcal{O}(g(n))$ if $t_A(n)$ is in $\mathcal{O}(g(n))$, see Appendix A for the meaning of the $\mathcal{O}(\cdot)$-notation. In accordance with this definition, we say that A runs in *polynomial time* if $t_A(n)$ is in $\mathcal{O}(p(n))$ for some polynomial p. Likewise, A runs in *exponential time* if $t_A(n)$ is in $\mathcal{O}(2^{p(n)})$ for some polynomial p. Similar definitions for the *space complexity* of an algorithm exist.

worst case running time

running time complexity

polynomial time

exponential time

space complexity

Next, we deal with decision and optimization problems. We address decision problems first. Formally, we define a *decision problem* P as a tuple (I, SOL), where I is the set of instances and $SOL\colon I \to \{0,1\}$ associates with every instance $x \in I$ either zero or one. The problem P is identified with the *language* $L_P = \{x \in I \mid 1 = SOL(x)\}$. Answering the question whether $x \in L_P$ for any given x is called to *recognize* the language L_P or to *solve* the problem P. The two most prominent complexity classes concerning decision problems are defined as follows.

decision problem

language

recognize

solve

Definition 3.2. *A decision problem P is solved by a* deterministic algorithm *A if the algorithms halts for every instance $x \in I_P$ and returns* Yes *if and only if $x \in L_P$. The class $\mathcal{P}$ consists of all problems solvable in polynomial time by some deterministic algorithm.*

$\mathcal{P}$

In addition to deterministic algorithms, nondeterministic algorithms are considered. One way of thinking about a *nondeterministic algorithm* is that the algorithm nondeterministically chooses one out of at most two possible instructions for execution at each step.

nondeterministic algorithm

Definition 3.3. *A decision problem P is solved by a nondeterministic algorithm A if, for any instance $x \in I_P$, A halts and $x \in L_P$ if at least one possible sequence of instructions causes the algorithm to return* Yes.

$\mathcal{NP}$

The class $\mathcal{NP}$ consists of all problems solvable in polynomial time by some nondeterministic algorithm.

Clearly, $\mathcal{P} \subseteq \mathcal{NP}$. One of the challenging open problems in computational complexity theory is to settle the question whether $\mathcal{P} \overset{?}{=} \mathcal{NP}$. Despite continuous and serious efforts over the past three decades, this problem is still open. It is, however, commonly believed that $\mathcal{P} \subsetneq \mathcal{NP}$.

reducible

Given two decision problems P_1 and P_2, P_1 is said to be (polynomial time m-)*reducible* to P_2 if a polynomial time algorithm A exists which maps instances $x \in I_{P_1}$ into instances $A(x) \in I_{P_2}$ satisfying $x \in L_{P_1} \iff A(x) \in L_{P_2}$. On the basis of this reducibility, the notions of $\mathcal{NP}$-completeness and $\mathcal{NP}$-hardness are defined.

$\mathcal{NP}$-complete

Definition 3.4. *A decision problem P is called $\mathcal{NP}$-hard if every problem in $\mathcal{NP}$ is reducible to P. The problem is $\mathcal{NP}$-*complete *if, additionally, P itself is in $\mathcal{NP}$.*

If $\mathcal{P} \subsetneq \mathcal{NP}$, then no $\mathcal{NP}$-complete problem can in general be solved deterministically in polynomial time; in Section 3.2.3, we show that deciding whether there is a feasible assignment for a carrier network is $\mathcal{NP}$-complete.

We now turn to the complexity of optimization problems and approximation. Our focus is on minimization problems, and all our definitions are specialized to this case. Definitions applying simultaneously for minimization and maximization problems can be found in Ausiello *et al.* [1999, Chapters 1, 3, and 8].

minimization problem
feasible solutions
measure function
value

Formally, a *minimization problem* P is characterized by a triple (I_P, SOL_P, m_P), where I_P is the set of instances of P; SOL_P is a function that assigns to every instance $x \in I_P$ a set $SOL_P(x)$ of *feasible solutions* for x; and m_P is the *measure function* that assigns to every pair (x, y) with $x \in I_P$ and $y \in SOL_P(x)$ a positive integer, called the *value* of the feasible solution y.

optimal solution

underlying language

Given some $x \in I_P$, the objective is to find an *optimal solution* $y^* \in SOL_P(x)$, satisfying $m_P(x, y^*) = \min_{y \in SOL_P(x)} m_P(x, y)$. We denote the value of an optimal solution by $m_P^*(x)$. A decision problem is associated to every optimization problem in a natural manner: let $P_D = (I_P \times \mathbb{Z}_+, SOL)$, where $SOL(x, K) = 1$ if $m_P^*(x) \leq K$ and $SOL(x, K) = 0$ otherwise. The language L_{P_D} is called the *underlying language* of P.

P_{FAP}

According to the above definition, FAP itself does not qualify as a minimization problem, because the objective function may take every value in $\mathbb{Q}_+$. By adding the following measure function, this is remedied. Given an instance x, let digits(x) denote the number of decimal digits in the smallest interference value (other than zero) in x. P_{FAP} is defined by

- I_{FAP}, which is the set of all carrier networks;
- SOL_{FAP}, which associates with each carrier network its set of feasible assignments;
- m_{FAP}, which computes digits(x) and the total interference of an assignment, scales the result by the factor of $10^{\text{digits}(x)}$ (in order to make it integral) and adds one (in order make it strictly positive).

With respect to optimization problems, $\mathcal{NPO}$ plays a role similar to that of $\mathcal{NP}$ among decision problems.

$\mathcal{NPO}$

Definition 3.5. *A minimization problem $P = (I_P, SOL_P, m_P)$ belongs to the class $\mathcal{NPO}$ if the following three conditions hold:*

(i) the set I_P of instances is recognizable in polynomial time;

(ii) a polynomial q exists such that $|y| \leq q(|x|)$ for all $y \in SOL_P(x)$ and such that it is decidable in polynomial time whether $y \in SOL_P(x)$ for every y with $|y| \leq q(|x|)$;

(iii) the measure function m_P is computable in polynomial time.

The problem P_{FAP} is in $\mathcal{NPO}$. First, it is recognizable in polynomial time whether a string encodes a carrier network. Second, the encoding of a feasible assignment does not take more space than the encoding of a carrier network (assuming that all carriers are listed individually and that all channels in the spectrum are listed as well); and it is recognizable in polynomial time whether an assignment is feasible. Third, digits(x) and the total interference of an assignment are computable in polynomial time and so is the above measure function.

$\mathcal{NP}$-hard

An optimization problem $P \in \mathcal{NPO}$ is *$\mathcal{NP}$-hard* if the language underlying P is $\mathcal{NP}$-complete. (The precise definition of $\mathcal{NP}$-hardness for optimization problems is more involved, compare, e. g., Ausiello *et al.* [1999, Definition 1.19]. Our definition is merely an immediate consequence of the more general definition.) Sometimes, the $\mathcal{NP}$-hardness of a minimization problem P is only due to instances involving large numbers. If this is not the case, the problem P is said to be strongly $\mathcal{NP}$-hard. P_{FAP} is strongly $\mathcal{NP}$-hard as we show in Section 3.2.3.

strongly $\mathcal{NP}$-hard

Definition 3.6. *Consider a problem $P \in \mathcal{NPO}$, and let* $\max(x)$ *denote the value of the largest number occurring in the instance x. For a polynomial p, $P^{\max,p}$ is the restriction of P containing only the instances x with* $\max(x) \leq p(|x|)$. *The problem P is called* strongly $\mathcal{NP}$-hard *if $P^{\max,p}$ is $\mathcal{NP}$-hard for some polynomial p.*

performance ratio

Finally, we address the complexity of approximating optimal solutions for minimization problems. Given a minimization problem P, the *performance ratio* of a solution $y \in SOL_P(x)$ with respect to instance $x \in I_P$ is defined as

$$R(x,y) = \frac{m_P(x,y)}{m_P^*(x)}.$$

Recall in this context that the measure function takes values in the positive integers even though it often seems convenient to allow zero or negative values, too. But this would clash with the previous definition.

r-approximate

Definition 3.7. *Given an optimization problem P, an algorithm A for solving P, and a function $r\colon \mathbb{Z}_+ \to]1,\infty[$, then A is an* r-approximate algorithm *for P if for every instance $x \in I_P$ with $SOL_P(x) \neq \emptyset$ the performance ratio of the approximate (feasible) solution $A(x)$ is bounded by $r(|x|)$, i. e., $R(x, A(x)) \leq r(|x|)$.*

In case the function r maps all arguments to some constant $c \in \mathbb{Q}_+$, we also speak of a c-approximate algorithm. The class $\mathcal{APX}$ contains of all minimization problems in $\mathcal{NPO}$ that are approximable with a constant performance guarantee in polynomial time. We show in Section 3.2.3 that P_{FAP} is not in $\mathcal{APX}$.

$\mathcal{APX}$

Definition 3.8. *A minimization problem $P \in \mathcal{NPO}$ is in $\mathcal{APX}$ if, for some constant $c \geq 1$, a c-approximate polynomial time algorithm for P exists.*

3.2.2 Classical Problems related to FAP

The variant P_{FAP} of FAP is in $\mathcal{NPO}$. P_{FAP} is closely related to several optimization problems studied in the literature. We quote some examples from the list of $\mathcal{NPO}$-problems of Ausiello *et al.* [1999, Appendix B]. The labels correspond to those given in the reference.

GT 5 MINIMUM GRAPH COLORING

GT 22 MAXIMUM CLIQUE

GT 27 Minimum Edge-Deletion Subgraph with Property Π

GT 32 Minimum Edge Deletion k-Partition

GT 33 Maximum k-Colorable Subgraph

ND 14 Maximum Cut

ND 17 Maximum k-Cut

ND 55 Minimum k-Clustering Sum

MS 16 Maximum Frequency Allocation

We pick out the Minimum Edge Deletion k-Partition problem and show in which way P_{FAP} generalizes this problem. We then quote results on the complexity of Minimum Edge Deletion k-Partition from the literature and derive lower bounds on the complexity of P_{FAP}.

Definition 3.9. *An instance of the* Minimum Edge Deletion k-Partition *problem consists of an undirected graph $G = (V, E)$, a weighting $w\colon E \to \mathbb{Z}_+$ of the edges, and a positive integer $k \leq |V|$. The objective is to find a partition of V into at most k disjoint sets $V_1, \ldots, V_p$ such that*

$$\sum_{l=1}^{p} \sum_{ij \in E: i,j \in V_l} w_{ij}$$

is minimized. The associated measure function evaluates the objective function and adds 1.

Proposition 3.10. *The following statements hold with respect to the computational complexity of* Minimum Edge Deletion k-Partition*:*

(i) The problem is strongly $\mathcal{NP}$-hard.

(ii) Unless $\mathcal{P} = \mathcal{NP}$, the problem is not in $\mathcal{APX}$. (Sahni and Gonzalez [1976])

(iii) For $k \geq 3$ an approximation within $\mathcal{O}(|E|)$ is $\mathcal{NP}$-hard, even when restricting the instances to graphs with $|E| = \Omega(|V|^{2-\varepsilon})$ for a fixed ε, $0 < \varepsilon < 1$. (Kann, Khanna, Lagergren, and Panconesi [1997])

(iv) Unless $\mathcal{P} = \mathcal{NP}$, no polynomial time algorithm can achieve a better performance ratio than 1.058 in the case of $k = 2$. (Håstad [1997])

(v) *In case of $k = 2$, a polynomial time algorithm with a performance guarantee of* $\log|V|$ *is known. (Garg, Vazirani, and Yannakakis [1996])*

(vi) *In case of $k = 3$, a polynomial time algorithm with a performance guarantee of $\varepsilon|V|^2$ for any $\varepsilon > 0$ is known. (Kann et al. [1997])*

Although the first fact listed in Proposition 3.10 is certainly known, we are not aware of a reference. The following simple proof is given for the sake of completeness.

Proof of Proposition 3.10 (i). Let P denote the minimization problem MINIMUM EDGE DELETION K-PARTITION. We show that the restricted version $P^{\max,1}$ is $\mathcal{NP}$-hard by reducing the $\mathcal{NP}$-complete decision problem GRAPH K-COLORABILITY, see Garey and Johnson [1979, GT 4], to the language underlying $P^{\max,1}$.

We associate to every instance (G, k) of GRAPH K-COLORABILITY an instance $x_{G,k}$ of MINIMUM EDGE DELETION K-PARTITION by simply labeling every edge in G with a weight of 1. Clearly, G is k-colorable if and only if the optimal solution to the associated instance $x_{G,k}$ has a value of 0. Hence, the language underlying $P^{\max,1}$ is $\mathcal{NP}$-hard. □

3.2.3 Complexity of FAP

We now transfer the negative results concerning MINIMUM EDGE DELETION K-PARTITION to P_{FAP}. To every instance of the former problem, we associate an instance of P_{FAP} as follows:

$$(V, E), w, k \quad \mapsto \quad (V, E, \{1, \dots, k\}, (\emptyset)_{v \in V}, 0, c^{co}, 0),$$

where $c^{co}(ij) = 10^{-\operatorname{digits}(w)}\, w(ij)$ and $\operatorname{digits}(w)$ denotes the number of digits of $\max\{w(ij) \mid ij \in E\}$ (in a representation to the basis 10). The corresponding carrier network can be computed in polynomial time and, in particular, its encoding length is polynomially bounded in the encoding length of the MINIMUM EDGE DELETION K-PARTITION instance. The k-partitions of the graph (V, E) and the frequency assignments for the carrier network are in one-to-one correspondence. By definition, the measure functions produce identical values for all partitions of V with respect to both problems. Hence, the hardness results from Proposition 3.10 translate directly to P_{FAP}. Together with the already established $\mathcal{NP}$-hardness of finding any feasible assignment for a carrier network, we obtain the following list of results.

Proposition 3.11.

(i) *Deciding whether a feasible assignment exists for a carrier network is $\mathcal{NP}$-complete.*

(ii) $P_{\mathsf{FAP}} \in \mathcal{NPO}$.

(iii) P_{FAP} *is strongly* $\mathcal{NP}$*-hard.*

(iv) *Unless* $\mathcal{P} = \mathcal{NP}$, P_{FAP} *is not in* $\mathcal{APX}$.

(v) *Unless* $\mathcal{P} = \mathcal{NP}$, *an approximation of* P_{FAP} *within* $\mathcal{O}(|E|)$ *is impossible in polynomial time for* $k \geq 3$.

Unlike in the case of Minimum Edge Deletion k-Partition, positive results on the approximation of P_{FAP} cannot be proven due to Proposition 3.11 (i) unless $\mathcal{P} = \mathcal{NP}$. We read the above negative results on the approximation of P_{FAP} in the following way: even if feasible solutions were producable in polynomial time (while still assuming $\mathcal{P} \neq \mathcal{NP}$), finding approximately optimal solutions would be hard nevertheless.

As a final remark concerning the approximation complexity, we add that the result stated in Proposition 3.11 (v) does not depend on the particular measure function associated with P_{FAP}. The result of Kann *et al.* [1997] also transfers directly to FAP, because their proof only involves instances with edge weights of 1.

3.3 Alternative Formulations

In the remainder of this chapter, two integer linear programming formulations of FAP are presented. Solving either of the associated integer linear programs (ILPs) for a given carrier network to optimality is equivalent to solving FAP.

The modeling of the nonlinear separation constraints $|y(v) - y(w)| \geq d(vw)$ poses a problem in linear formulations. The two models cope with them differently. Our first formulation, called "stable set model," is classical. The variables are only allowed to take the values zero or one. The second formulation, called "orientation model," uses binary as well as integer variables. For the ease of notation, let us define

$$E^d = \big\{vw \in E \mid d(vw) > 0\big\}, \tag{3.3}$$

$$E^{co} = \big\{vw \in E \mid d(vw) = 0, c^{co}(vw) > 0\big\}, \tag{3.4}$$

$$E^{ad} = \big\{vw \in E \mid d(vw) \leq 1, c^{ad}(vw) > 0\big\}, \tag{3.5}$$

for a given carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$.

3.3.1 Stable Set Model

Binary variables y_v^f are used to select a channel per carrier. The co- and adjacent channel interference indicator variables z_{vw}^{co} and z_{vw}^{ad}, respectively, are needed for accounting the total amount of interference in the objective function. The following ILP is equivalent to FAP:

$$\min \sum_{vw \in E^{co}} c_{vw}^{co} z_{vw}^{co} + \sum_{vw \in E^{ad}} c_{vw}^{ad} z_{vw}^{ad}$$

s. t.

$$\begin{aligned}
&\sum_{f \in C \setminus B_v} y_v^f &&= 1 && \forall v \in V && \text{(3.6a)}\\
&y_v^f + y_w^g &&\leq 1 && \forall vw \in E^d, |f-g| < d_{vw} && \text{(3.6b)}\\
&y_v^f + y_w^f - z_{vw}^{co} &&\leq 1 && \forall vw \in E^{co}, f \in C \setminus (B_v \cup B_w) && \text{(3.6c)}\\
&y_v^f + y_w^{f-1} - z_{vw}^{ad} &&\leq 1 && \forall vw \in E^{ad}, f \in C \setminus B_v, f-1 \in C \setminus B_w && \text{(3.6d)}\\
&y_v^f &&\in \{0,1\} && \forall v \in V, f \in C \setminus B_v && \text{(3.6e)}\\
&z_{vw}^{co} &&\in \{0,1\} && \forall vw \in E^{co} && \text{(3.6f)}\\
&z_{vw}^{ad} &&\in \{0,1\} && \forall vw \in E^{ad} && \text{(3.6g)}
\end{aligned}$$

We first explain the constraints and then the objective function. The constraints (3.6a) model that precisely one available channel has to be assigned to every carrier. In case channel f is assigned to carrier v, then y_v^f takes the value 1. Otherwise, the value is 0. The inequalities (3.6b) enforce that the selection of the available channels also satisfies all separation constraints. Every 0/1-assignment of the variables y_v^f satisfying the constraints (3.6a) and (3.6b) corresponds to a feasible frequency assignment for N, see constraints (3.1) and (3.2) on page 34.

We now turn to the interference accounting. A binary variable z_{vw}^{co} is used for every edge $vw \in E^{co}$ to indicate co-channel interference, i. e., $z_{vw}^{co} = 1$ has to hold if $vw \in E^{co}$ and v and w receive the same channel. This implication is implemented by the constraints (3.6c). Likewise, a binary variable z_{vw}^{ad} is used for every potential adjacent channel interference. In case $vw \in E^{ad}$ and v and w have adjacent channels, $z_{vw}^{ad} = 1$ has to hold. This is achieved by the corresponding constraints (3.6d). Exchanging v and w and replacing f by $f+1$ in (3.6d), yields the constraint $y_v^f + y_w^{f+1} - z_{vw}^{ad} \leq 1$.

Notice that the interference accounting variables are not fully controlled by the given constraints in the sense that z_{vw}^{co}, for example, may take the value 1 where a value of 0 would be sufficient. This never happens in an optimal solution because we are minimizing and all objective function coefficients are assumed to be strictly positive.

We call the formulation (3.6) a *stable set model* for the following reason: if no interference were allowed, then the set of carriers receiving the same channel in a feasible assignment would be stable sets in the underlying graph $G = (V, E)$. (A subset of the vertices in G is called *stable* or *independent* if no two vertices are adjacent.) Hence, a feasible assignment not incurring any interference partitions the vertices of $G = (V, E)$ into stable sets. The Ph. D. thesis of Borndörfer [1998] contains an in-depth treatment of set partition problems and the related problems of set covering and set packing. We also refer to the Ph. D. thesis of Schulz [1996, Chapter 4], where an extension of the set packing problem is described. This so-called *transitive packing* puts our interference accounting constraints (3.6c) and (3.6d) into a general framework.

stable
independent

transitive packing

We come back to the stable set model in Section 6.2.2.

3.3.2 Orientation Model

The orientation model, as presented here, is only correct if the input data satisfies additional restrictions. The first type of restrictions is that $c^{co}_{vw} > c^{ad}_{vw}$ has to be met for every edge $vw \in E$ with $d_{vw} = 0$. This is uncritical from a practical point of view and can be seen as a merely technical restriction. The second type of restrictions concerns locally blocked channels. Those are not handled in the model, and we assume that none exist. This restriction is drastic, but we explain how to bypass it later on. The model is introduced and discussed thoroughly by Borndörfer, Eisenblätter, Grötschel, and Martin [1998b].

As the main difference to the stable set model no binary variables y^f_v with $v \in V$ and $f \in C$ are used. Instead, an integral variable y_v with domain C is introduced for every carrier $v \in V$. The value of y_v indicates the channel to assign. As before, binary variables z^{co}_{vw} and z^{ad}_{vw} are used for accounting interference.

The intuition behind the model is to consider FAP as two nested problems. An acyclic orientation A of the edges E in the graph underlying the carrier network is determined in the outer part. Such an orientation A of the edges induces a partial order $<_A$ on the carrier set: we declare $v <_A w$ if a directed path starting at v and ending at w exists in (V, A). (Checking that $<_A$ is indeed a partial order is straight forward.) The inner part of the nested problem is to find a feasible frequency assignment

outer part

inner part

compatible

$y\colon V \to C$ with minimum interference, which is *compatible* with the partial order $<_A$ in the following sense:

$$y(v) < y(w) \qquad \Longleftrightarrow \qquad v <_A w$$

Two variables, $o_{(v,w)}$ and $o_{(w,v)}$, are introduced for every edge, one for each possible direction. The *orientation model* reads as follows for the carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$.

$$\min_{o} \min_{y, z^{co}, z^{ad}} \sum_{vw \in E^{co}} c^{co}_{vw}\, z^{co}_{vw} + \sum_{vw \in E^{ad}} c^{ad}_{vw}\, z^{ad}_{vw}$$

$$\begin{array}{llllr}
\text{s.t.} & & & & \\
o_{(v,w)} + o_{(w,v)} & = & 1 & \forall vw \in E & \text{(3.7a)} \\
\begin{bmatrix} -y_v + y_w \\ +y_v - y_w \end{bmatrix} + \quad + & \geq & d_{vw} \begin{bmatrix} o_{(v,w)} \\ o_{(w,v)} \end{bmatrix} - \begin{bmatrix} o_{(w,v)} \\ o_{(v,w)} \end{bmatrix} M & \forall vw \in E^d & \text{(3.7b)} \\
\begin{bmatrix} -y_v + y_w \\ +y_v - y_w \end{bmatrix} + z^{co}_{vw} + & \geq & 1 \begin{bmatrix} o_{(v,w)} \\ o_{(w,v)} \end{bmatrix} - \begin{bmatrix} o_{(w,v)} \\ o_{(v,w)} \end{bmatrix} M & \forall vw \in E^{co} & \text{(3.7c)} \\
\begin{bmatrix} -y_v + y_w \\ +y_v - y_w \end{bmatrix} + \quad + z^{ad}_{vw} & \geq & 2 \begin{bmatrix} o_{(v,w)} \\ o_{(w,v)} \end{bmatrix} - \begin{bmatrix} o_{(w,v)} \\ o_{(v,w)} \end{bmatrix} M & \forall vw \in E^{ad} \setminus E^{co} & \\
& & & & \text{(3.7d)} \\
\begin{bmatrix} -y_v + y_w \\ +y_v - y_w \end{bmatrix} + 2z^{co}_{vw} + z^{ad}_{vw} & \geq & 2 \begin{bmatrix} o_{(v,w)} \\ o_{(w,v)} \end{bmatrix} - \begin{bmatrix} o_{(w,v)} \\ o_{(v,w)} \end{bmatrix} M & \forall vw \in E^{ad} \cap E^{co} & \\
& & & & \text{(3.7e)} \\
y_v & \in & C & \forall v \in V & \text{(3.7f)} \\
z^{co}_{vw} & \in & \{0,1\} & \forall vw \in E^{co} & \text{(3.7g)} \\
z^{ad}_{vw} & \in & \{0,1\} & \forall vw \in E^{ad} & \text{(3.7h)} \\
o_{(v,w)}, o_{(w,v)} & \in & \{0,1\} & \forall vw \in E & \text{(3.7i)}
\end{array}$$

We set $M = C_{max} = \max\{f \mid f \in C\}$. The different parts in the ILP formulation are explained in the following. The objective function is the same as before.

By means of the constraints (3.7a), a direction is selected for every edge in the carrier network. Depending on the direction, one of the "paired" constraints in (3.7b), (3.7c), and (3.7d) is vacuously true, because the value of the left-hand side cannot be less than $-C_{max} = -M$.

Concerning (3.7b), if the edge $vw \in E^d$ is oriented from v to w, i.e., $o_{(v,w)} = 1$ and $o_{(w,v)} = 0$, then the channel y_w has to be at least as large as y_v. The separation constraint $|y_v - y_w| \geq d_{vw}$ simplifies to $y_w - y_v \geq d_{vw}$.

This is expressed in the first constraint of (3.7b). Conversely, if the edge vw is oriented the other way around, the separation constraint reduces to $y_v - y_w \geq d_{vw}$, as imposed in the second part of (3.7b).

Concerning (3.7c), if the edge $vw \in E^d$ is oriented from v to w, then $y_w \geq y_v$ has to hold. This is enforced by the first constraint of (3.7c). Co-channel interference arises in the case of $y_w = y_v$, and the same constraint drives the variable z^{co}_{vw} to 1. The reverse case is analogous.

Accounting adjacent channel interference is more subtle. The cases $vw \in E^{ad} \setminus E^{co}$ and $vw \in E^{ad} \cap E^{co}$ are distinguished. First, we consider two carriers $v, w \in V$ with $d_{vw} = 1$ and $c^{ad}_{vw} > 0$, i. e., $vw \in E^{ad} \setminus E^{co}$. Examining the two cases of either $o_{(v,w)} = 1$ and $o_{(w,v)} = 0$ or $o_{(v,w)} = 0$ and $o_{(w,v)} = 1$, we observe that the pair of constraints of (3.7d) drive z^{ad}_{vw} to 1 if $|y_v - y_w| = 1$. Second, we consider two carriers $v, w \in V$ with $d_{vw} = 0$ and $c^{co}_{vw}, c^{ad}_{vw} > 0$, i. e., $vw \in E^{ad} \cap E^{co}$. We pick the orientation expressed by $o_{(v,w)} = 1$ in order to discuss the effects of the paired constraints (3.7e). Assuming $o_{(v,w)} = 1$ and $o_{(w,v)} = 0$, the second part is vacuously true, and the first part reduces to

$$-y_v + y_w + 2z^{co}_{vw} + z^{ad}_{vw} \geq 2.$$

Consequently, $z^{co}_{vw} = 1$ has to hold in case of $y_v = y_w$. If $y_w - y_v = 1$, however, both $z^{co}_{vw} = 1$ and $z^{ad}_{vw} = 1$ would satisfy the constraint. At this point, the additional assumption of $c^{co}_{vw} > c^{ad}_{vw}$ steps in to guarantee that $z^{ad}_{vw} = 1$ holds in an optimal solution.

The integrality constraints need no further explanation, but recall that $B_v = \emptyset$ is assumed for all carriers.

Although we do not give all the details here, it should be clear that solving the orientation model to optimality is equivalent to solving FAP, if the additional restrictions mentioned above hold.

Finally, we indicate how the restriction of $B_v = \emptyset$ for all carriers $v \in V$ can be by-passed. We do not formalize how to include locally blocked channels in the orientation model, because it is simpler to express in words. The problem with the local blockings is that they may "puncture" the otherwise contiguous domain C of the y-variables. Let $B = \bigcup_{v \in V} B_v$ denote the union of all locally blocked channels. (By definition, $B \subseteq C$.) We introduce an artificial carrier b for each $b \in B$ and restrict its available set of channels to the singleton $\{b\}$. Additionally, for each carrier v with $b \in B_v$, an edge vb is inserted, and the edge labelings d, c^{co}, and c^{ad} are extended by setting $d(vb) = 1$, $c^{co}(vb) = 0$, $c^{ad}(vb) = 0$. The orientation model for this extended carrier network fixes the values of the variables y_b to $b \in C$. A closer inspection of the extended model reveals that some new constraints may be superfluous, but the details are technical and we skip the corresponding discussion.

The "inner part" of the orientation model, which deals with finding an assignment of minimal interference among all feasible assignments compatible with a given orientation, forms the basis of the improvement heuristic Mcf. This is discussed in Section 4.3.4.

Chapter 4

Fast Heuristic Methods

Before we proceed, let us recapitulate our points up to now. The problem of generating "good" frequency plans for GSM networks was explained in detail, see Section 2.3, and the notion of a carrier network, see Section 3.1, was introduced to represent the essential characteristics of a GSM frequency planning problem: $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$ denotes a generic carrier network, the carriers in the set V represent the TRXs, the edges in the set E the relation between carriers, the channels in the set C the available spectrum, and the sets $\{B_v\}_{v \in V}$ the locally blocked channels; furthermore, d, c^{co}, and c^{ad} represent the required minimum separation, the expected co- and the expected adjacent channel interference between pairs of carriers, respectively.

We argued that the sum over all co- and adjacent channel interferences between carriers is an adequate measure for the quality of a frequency assignment. This quantity is minimized in the mathematical optimization problem FAP, see Section 3.1. Furthermore, we pointed out that the generation of reliable input data for setting up the carrier network is intricate. This holds in particular for interference data. But we also stated that the generation of input data can be mastered with a sufficient accuracy today, see Section 2.3.2.

We are now at the point to address the optimization problem FAP computationally. Recall from Section 3.2.3 that solving FAP is $\mathcal{NP}$-hard and that finding solutions which are guaranteed to be close to optimal is also $\mathcal{NP}$-hard. Hence, according to present understanding, both tasks are unlikely to be algorithmically solvable in a running time which is polynomially bounded in the size of the input. This is our justification for the development of fast methods which neither necessarily produce close to optimal nor even feasible assignments (provided some exist).

Our focus is on frequency planning heuristics, capable of dealing with carrier networks of around 2000 carriers in a quarter of an hour on a modern PC or workstation. Such methods are well-suited for GSM frequency planning in practice with particular emphasis on the intermediate steps in the planning cycle, see Section 2.3.3. Several of these methods are now in use at the GSM operator E-Plus Mobilfunk GmbH & Co. KG.

More elaborate but also more time-consuming methods are capable of producing better results than our fast heuristics. This type of methods targets primarily the generation of the final frequency plan, where running times in the order of one or two days are acceptable. We come back to this issue in Section 5.6.

The algorithmic solution of frequency planning problems is not a new topic, of course. Well *over a hundred articles and reports* propose and discuss algorithmic techniques for frequency planning in general or in the specialization to GSM. Recent *surveys* are given by Jaumard *et al.* [1999], Koster [1999], and Murphey *et al.* [1999].

surveys

The published algorithms can be distinguished according to whether they guarantee to produce optimal solutions or approximately optimal solutions or neither of both. None of the enumeration or cutting-plane algorithms, however, which may produce provably optimal solutions in the general case are suited for frequency planning in practice, because their running times grow exponentially with the size of the carrier network. And for the algorithms proven to produce (approximately) optimal assignments for special cases in polynomial running time, the relevant cases are not known to appear in practice. Hence, these two types are not of interest to us. Among the algorithms without quality guarantee, there are proposals for procedures building on the meta-heuristics Simulated Annealing, Tabu Search, Genetic Algorithms, Neural Networks, etc. Such procedures typically have much higher running times than the ones we are aiming at. But also fast methods are proposed, which mostly build on ideas published in connection with computing graph colorings.

Despite our focus on fast methods, we also performed extensive experiments with Simulated Annealing and Tabu Search, see Schneider [1997] for an early report on these experiments. A general introduction to local search methods as well as to Genetic Algorithms and Neural Networks is given in Aarts and Lenstra [1997]. The implementations of those methods typically have running times in the order of several hours up to one or two days on large carrier networks. Our own results are in the range of what is published by Beckmann and Killat [1999]. Further implementations of Simulated Annealing or Tabu Search for GSM frequency planning are discussed by Duque-Antón *et al.* [1993], Castelino, Hurley, and Stephens [1996], Hurley, Smith, and Thiel [1997], Hao, Dorne, and Galinier [1998], Smith, Allen, Hurley, and Watkins [1998], and Correia [2001, Section 4.2.5]. This list is incomplete, but provides starting points for a further exploration of the literature.

The chapter is organized as follows. In Section 4.1, we describe techniques for preprocessing the carrier network prior to running frequency planning procedures. Then, we turn to two types of algorithms, which are generally known for efficiently "solving" combinatorial optimization problems: greedy construction heuristics and simple improvement heuristics. We describe methods of these kinds in Sections 4.2 and 4.3, respectively. Computational results for our methods are reported in Chapter 5.

For the most part, this chapter describes joint work with Ralf Borndörfer, Martin Grötschel, and Alexander Martin as well as with the students Daniel Haberland and Margherita Hebermehl. The MCF method is described together with the orientation model (presented in Section 3.3.2) by Borndörfer *et al.* [1998b]. The other heuristics of Sections 4.2 and 4.3, except for K-OPT and VDS, are also contained in Borndörfer, Eisenblätter, Grötschel, and Martin [1998a].

4.1 Preprocessing

Prior to calling some planning method, a carrier network may be preprocessed in order to simplify the frequency planning problem. Techniques to identify possible simplifications are, for example, studied in the field of constraint satisfaction programming (CSP). We are not aware, however, of any specific study for our version of the frequency planning problem.

We consider two types of modifications here. In the case which we call *structure-preserving*, the modified carrier network allows feasible assignments if and only if the original carrier network allows them. Moreover, for each optimal assignment for the modified network, an optimal assignment for the original network can be generated in polynomial time. In the other case, we call it *heuristic*, the modifications may change the feasibility status, and optimal assignments to both networks are not necessarily related. We are interested in this type of modifications, nevertheless, because the frequency assignments produced by some of our methods are often better when such a preprocessing is applied.

structure-preserving

heuristic

We present structure-preserving modifications in Section 4.1.1 and heuristic modifications in Section 4.1.2.

4.1.1 Eliminating Channels and Carriers

We specify a few situations in which the carrier network may be reduced structure-preservingly by either dropping available channels for a carrier or by dropping a carrier altogether.

Dropping Channels

As a motivating example, we look at a carrier network, where some of the carriers are effectively fixed by having only one channel available. Let v_0 denote such a carrier with $C \setminus B_{v_0} = \{f_0\}$. Moreover, let w be a carrier with $d(v_0w) > 0$. Then the channels $f_0-(d(v_0w)-1), \ldots, f_0+(d(v_0w)-1)$ may be added to B_w without changing the set of feasible solutions.

This can be generalized to situations, where v_0 has a few channels available and for a fixed $g \in C \setminus B_w$ the relation $|f - g| < d(v_0w)$ holds for all channels $f \in C \setminus B_{v_0}$. Then the channel g may be added to B_w. The idea can be pursued further and extended to larger sets of carriers than merely two carriers. The computational burden of recognizing such a situation, however, becomes significantly harder with each additional carrier considered.

None of the above cases turns out to be relevant for our test instances introduced in Section 5.1.1. In planning practice, however, the first case is relevant and therefore part of our preprocessing. The other cases are not addressed.

Dropping Carriers

Clearly, we may drop every carrier for which only one available channel is left. This is preferably done after dropping channels. Another situation, in which carriers may be dropped without harm, is described next. We call the *generalized degree* $gd(v)$ of a carrier v the quantity

generalized degree

$$gd(v) = \sum_{vw \in E} \big(2 \max\{d(vw), h(vw)\} - 1\big)$$

with

$$h(vw) = \begin{cases} 2, & \text{if } c^{ad}(vw) > 0, \\ 1, & \text{if } c^{co}(vw) > 0, c^{ad}(vw) = 0, \\ 0, & \text{otherwise.} \end{cases}$$

For a fixed carrier w, the summand "$2 \max\{d(vw), h(vw)\} - 1$" equals the maximal number of channels which may become burdened with separation constraints or interference for carrier v if w gets assigned. Hence, the above sum gives an upper bound on the effect of assigning channels to all adjacent carriers of v. The actual effect, however, may be much smaller.

If the generalized degree of a carrier is less than the number of available channels, then the carrier may be dropped for the following reason.

Whatever the assignments to the adjacent carriers are, there is always at least one channel available which can be assigned without causing a separation constraint violation or interference.

As reported in Section 5.1.1, this form of preprocessing often allows to drop a few carriers. If one carrier is dropped, the generalized degree of its adjacent carriers reduces. Consequently, the technique can be applied repeatedly until no more carrier is dropped. A feasible assignment for the reduced carrier network is extended in the reverse dropping order to a feasible assignment of the original network and without introducing extra interference. Carriers may also be dropped if the generalized degree exceeds the number of available channels by some small factor. In that case, however, the modification is heuristic.

dominates

We also look for possibilities to reduce the size of a carrier network by amalgamating carriers, i. e., by treating them exactly the same way. Let v and w be two nonadjacent carriers. We say that v *dominates* w if $B_w \subseteq B_v$ and if the edge vx is in E for all $wx \in E$ satisfying $d(vx) \geq \max\{d(wx), h(wx)\}$. The function h is the same as defined above. Then, a feasible assignment for a carrier network without w can be extended to include w by simply assigning the channel of v to w as well. Although appealing in principle, domination plays no role for any of our planning instances.

4.1.2 Tightening the separation

The following modification is a heuristic in the sense defined above. Let v and w be adjacent carriers, then $d(vw)$ is the minimum necessary separation between the channels assigned to v and w. In case $d(vw) \geq 1$, the same channel must not be given to both carriers. This rules out co-channel interference between v and w. Similarly, in case $d(vw) \geq 2$, no adjacent channel interference between v and w occurs in a feasible assignment.

threshold

One approach to control interference beyond minimizing its overall sum is to exclude assignments causing large interference between individual carrier pairs. To this end, we introduce a *threshold* t. The threshold is used to impose a sufficiently large separation between carriers which may otherwise cause interference exceeding t:

$$d^t(vw) = \begin{cases} \max\{1, d(vw)\}, & \text{if } c^{co}(vw) > t \text{ and } c^{ad}(vw) < t \\ \max\{2, d(vw)\}, & \text{if } c^{ad}(vw) > t \\ d(vw), & \text{otherwise} \end{cases}$$

The carrier network $N^t = (V, E, C, \{B_v\}_{v\in V}, d^t, c^{co}, c^{ad})$ is obtained from N by *tightening the separation with t*. A feasible assignment for N^t may still incur interference, but none exceeding the threshold t between a carrier pair. Because an assignment causing high interference between one pair may save considerably between others, no optimal assignment for the original problem may be feasible for the modified one.

tightening

Despite this fact, tightening the separation works well in conjunction with some of our heuristics. By applying heuristics to N^t for different threshold values, solutions of varying quality are usually obtained. A suitable threshold value t may be determined by some search routine. One example for such a search routine is described in Section 5.2.2.

4.2 Greedy Methods

Greedy methods compute a frequency assignment from scratch, step-wise extending an initially empty assignment to a complete assignment. In the course of the construction, partial frequency assignments occur. A *partial frequency assignment* is a mapping $y\colon A \to C$ that is defined on a subset A of the carrier set V. In case $A = V$, a partial assignment is just an ordinary frequency assignment.

partial assignment

We have performed experiments with several greedy methods and describe three prototypical ones in the following. Among them is the adaption DSATUR WITH COSTS of the well-known graph coloring heuristic DSATUR. This is our most successful greedy method, and it is used in frequency planning practice at E-Plus Mobilfunk GmbH & Co. KG.

4.2.1 T-Coloring

We recall the definition of a T-coloring. Given an undirected graph $G = (V, E)$ and nonempty finite sets $T(vw)$ of nonnegative integers for all $vw \in E$. A T-coloring of G is a labeling f of the vertices of G with nonnegative integers such that $|f(v) - f(w)| \notin T(vw)$ for all edges $vw \in E$. Since their introduction by Hale [1980], T-colorings of graphs and methods to produce them have been studied by several authors. A survey is given by Murphey *et al.* [1999]. Here, we are interested in the blend of T-coloring and list coloring introduced by Tesman [1993], where each vertex has a list of colors available.

Our T-COLORING heuristic, designed and implemented by Haberland [1996], is a modification of a T-coloring procedure proposed by Costa [1993]. The underlying idea is, however, already used in the graph

Algorithm 1 T-Coloring

```
Input: carrier network without interference information:
        $V, E, C, \{B_v\}_{v \in V}, d$
Output: a feasible assignment $y$ or a resignation message
  {Initialization}
  for all $v \in V$ do
    $satdeg[v] := |B_v|$   {saturation degree: unavailable channels}
    $spadeg[v] := \sum_{vw \in E} d(vw)$   {spacing degree: $\sum_{vw \in E: w \text{ unassigned}} d(vw)$}
  end for
  {Assigning}
  $U := V$   {contains unassigned carriers}
  while $U \neq \emptyset$ do
    pick $u$ s. t. $satdeg[u] = \max_{v \in U}\{satdeg[v] \mid spadeg[v] = \max_{w \in U} spadeg[w]\}$
                                        {ties are broken arbitrarily}
    $U := U \setminus \{u\}$
    let $y(u)$ be the available channel for $u$ of least index
    if no such available channel exists then
      resign   {$y$ is merely a partial assignment}
    end if
    for all $v \in U$ with $uv \in E$ do
      update $satdeg[v], spadeg[v]$
    end for
  end while
  {$y$ holds a feasible assignment}
```

coloring heuristic Dsatur by Brélaz [1979]. T-Coloring is our only "min-span" method, compare with Section 3.1.2. That is, T-Coloring does not try to minimize the overall interference but focuses solely on computing a feasible assignment using channels from a spectrum which is as narrow as possible. In conjunction with the preprocessing technique of tightening the separation, see Section 4.1.2, T-Coloring may be employed to search for assignments for which the maximal incurred interference between carrier pairs is minimal.

Algorithm 1 gives a sketch of the procedure. After the initialization, the carriers are assigned in the while-loop. The carrier to assign next is determined by means of the saturation and spacing degrees. For a formal definition of both quantities see Algorithm 1. Roughly speaking, the *saturation degree* keeps track of how many channels from the spectrum are no longer available for each of the remaining unassigned carriers. The

saturation degree

spacing degree

spacing degree represents how much impact assigning all of a carrier's still unassigned neighbors would have on its own assignability. If this impact is larger than that of its neighbors, the carrier should be handled before its neighbors. For similar reasons, carriers with a high saturation degree should be assigned as soon as possible.

T-Coloring is implemented using binary heaps (see Cormen *et al.* [1990, Chapter 7]) for bookkeeping of which carrier to assign next. The running time obtained this way is in $\mathcal{O}(|C||E| + |E|\log|V|)$, and the space requirement is in $\mathcal{O}(|C||V| + |E|)$. Computational results for the T-Coloring heuristic are given in Sections 5.2.1, 5.4, and 5.5.

4.2.2 Dsatur With Costs

The Dsatur with Costs heuristic is another modification of the graph coloring heuristic Dsatur proposed by Brélaz [1979], and again we incorporate ideas of Costa [1993]. The goal of Dsatur with Costs is to produce a feasible assignment of least possible total interference. Repeatedly, that carrier is assigned next, which seems to be hardest to deal with. The measure for "hardest to deal with" generalizes the saturation and spacing degrees introduced with the T-Coloring heuristic. Each carrier is assigned the channel presently incurring the least additional interference. An outline of the procedure is given as Algorithm 2.

A matrix *cost* is used to record the cost (= interference + separation violation penalty) of the carrier/channel combinations. The rows of *cost* are indexed by the carriers, and the columns are indexed by the channels. All entries corresponding to unavailable carrier/channel combinations are invalidated during the initialization. The matrix *cost* is updated throughout the process by adding update matrices, which reflect the effect of the current step. The generic update matrix $\Delta^{(v,f)}$ is defined component-wise by

$$\Delta^{(v,f)}[w][g] = \begin{cases} M, & \text{if } vw \in E, g \in C \setminus B_w, |f-g| < d(vw), \\ c^{co}(vw), & \text{if } vw \in E, d(vw) = 0, f = g, g \in C \setminus B_w, \\ c^{ad}(vw), & \text{if } vw \in E, d(vw) \leq 1, |f-g| = 1, g \in C \setminus B_w, \\ 0, & \text{otherwise.} \end{cases}$$

M is a suitably chosen constant. The still unassigned carriers are maintained in a heap H. A carrier's heap key is defined by

$$\mathit{key}(v) = |B_v|\, M + \sum_{f \in C \setminus B_v} h(\mathit{cost}[v][f]) \quad \text{with} \quad h(c) = \begin{cases} M, & \text{if } c \geq M, \\ c, & \text{otherwise.} \end{cases}$$

Algorithm 2 Dsatur with Costs

```
Input: carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$
Output: an assignment $y$, possibly infeasible
  {Initialization}
  for all $v \in V$ do
    $cost[v][f] := \begin{cases} 0, & \text{if } f \in C \setminus B_v \quad \{\text{initialize}\} \\ \infty, & \text{otherwise} \quad \{\text{invalidate}\} \end{cases}$
    insert $v$ into the heap $H$ with $key(v)$
  end for
  {Assigning}
  while $H \neq \emptyset$ do
    extract carrier $v$ with maximum $key(v)$ from heap $H$
    $y(v) := f$, where $f$ is available and of least value in the row $cost[v]$
    update $cost$ by adding $\Delta(v, f)$
    update $key(v)$ for all $v \in H$
  end while
  {$y$ holds the resulting assignment}
```

While the heap is not empty, a carrier v with maximum value $key(v)$ is extracted and assigned its least costly available channel f. This channel may induce separation violations, but then all other available channels do as well (assuming that M is chosen large enough).

A Fibonacci heap (see Cormen *et al.* [1990, Chapter 21]) is used in our implementation to determine the next carrier. The minimum-cost channel for a carrier v is determined simply by a search in the matrix row $cost[v]$. Notice that, once a carrier is assigned, the corresponding row in $cost$ is no longer needed and may become outdated without harm. This fact is exploited in our implementation and in the analysis of the amortized running time (see Cormen *et al.* [1990, Chapter 18]). Assuming that the graph underlying the carrier network is connected and, hence, $|V| \in \mathcal{O}(|E|)$, the running time of Dsatur with Costs is in $\mathcal{O}(|C||E| + |V| \log|V|)$, and the space requirement is in $\mathcal{O}(|V||C| + |E|)$.

starting point

The choice of the first carrier to assign has a considerable impact on the quality of the resulting assignment. No sufficiently general rule, however, is known to determine the carrier to start with. As a rather time consuming countermeasure, all carriers may be tried in turn, and the best resulting assignment is picked in the end. The following compromise between the two extreme of choosing only one or all carriers as starting points proves effective. Some small subset of the carriers, 5% say, is chosen at random, each of those is used as starting point, and the best resulting assignment is returned in the end. The computational behavior of DSATUR WITH COSTS is documented in Sections 5.2.2 and 5.4, see also Section 5.5.

4.2.3 Dual Greedy

The DUAL GREEDY heuristic constructs an assignment by means of eliminating alternatives. Repeatedly, an option for assigning some channel to a carrier is eliminated until only a single channel remains for every carrier. Our interest in this type of method is due to the results obtained by Jünger, Martin, Reinelt, and Weismantel [1994] in VLSI design.

The DUAL GREEDY is greedy in the following sense: the exclusion of a carrier/channel combination is based on a local perception of what seems to be an unfavorable combination. We use a set $A \subseteq V \times C$ to keep track of the remaining eligible combinations. Initially, A contains all carriers paired with all their available channels. A basic version of the procedure is shown as Algorithm 3.

Hebermehl [1996] investigates this basic version and several variants. The principal distinction among the studied variants lies in the definition of "unfavorable." The following example, where y is the already established partial assignment and A is the set of remaining eligible combinations, is taken from Hebermehl [1996]:

$$\begin{aligned}\text{unfavorable}(y, A; (w,g)) = & \sum_{\substack{(v,g)\in A:\\ vw\in E, d(vw)=0}} c^{co}(vw) \;+ \sum_{\substack{(v,g\pm 1)\in A:\\ vw\in E, d(vw)\leq 1}} c^{ad}(vw) \\ & + \sum_{\substack{(v,f)\in A: vw\in E,\\ |f-g|<d(vw)}} M_I \;+\; |\text{conflict}(y;(w,g))|\, W\end{aligned}$$

The set conflict$(y; (w,g))$ contains the already assigned carriers in y which are adjacent to w and which satisfy one of the three conditions $g = y(v), c^{co}(vw) > 0$ or $|g - y(v)| = 1, c^{ad}(vw) > 0$ or $|g - y(v)| < d(vw)$. M_I and W are two parameters.

This definition of unfavorable$(\cdot,\cdot,\cdot)$ is not the best performing among those investigated, but it is easily explained. On assigning channel f to carrier v, the parameter W is used to penalize all still eligible combinations that would result in interference or a separation violation if one of them were picked for assignment. Among the still unassigned carriers, potential co- and adjacent channel interference is directly accounted for, and potential separation constraint violations are penalized with the parameter M_I. High values for W should lead to little interference and few separation violations—if any. M_I weighs separation violations against interference. High values for M_I put emphasis on obtaining a feasible assignment.

Algorithm 3 Dual Greedy

```
Input: carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$
Output: an assignment $y$, possibly infeasible
  {Initialization}
  $A := \{(v, f) \in V \times C \mid f \in C \setminus B_v\}$   {eligible combinations}
  {Assigning/Eliminating}
  while $A \neq \emptyset$ do
    {Assigning}
    for all $(v, f) \in V \times C$ s. t. $|\{g \in C \mid (v, g) \in A\}| = 1$ do
      set $y(v) := f$ and remove $(v, f)$ from $A$
    end for
    {Eliminating}
    if $A \neq \emptyset$ then
      delete $(w, g)$ with highest value unfavorable$(y, A; (w, g))$ from $A$
    end if
  end while
  {$y$ holds the resulting assignment}
```

The success of the Dual Greedy procedure hinges on the definition of unfavorable$(\cdot, \cdot, \cdot)$. Reasonable rules for a given carrier network could be identified in extensive experiments, but those rules are rather problem dependent.

Fibonacci heaps (see Cormen *et al.* [1990, Chapter 21]) are used to keep track of unfavorable eligible carrier/channel combinations. Using such heaps, the Dual Greedy heuristic has a running time in $\mathcal{O}(|C|^2|V| \log(|C||V|) + |C|^2|E|)$ and requires $\mathcal{O}(|C||V| + |E|)$ space. In order to decrease the practical running time, the method for increasing the key of a heap element (in the Leda 3.6.1 [1998] implementation) is tuned. The amortized running time of this operation is still $\mathcal{O}(\log n)$, but practical time savings of roughly 25% are achieved. (See Cormen *et al.* [1990, Chapter 18] for an introduction to the concept of amortized analysis.) Nevertheless, the computational study of Hebermehl [1996] suggests that this method is generally inferior in terms of assignment quality as well as in terms of running time in comparison to (most) other methods presented here. We therefore exclude Dual Greedy from our comparison in Chapter 5.

4.3 Improvement Methods

improvement heuristic

An *improvement heuristic* takes a (partial) assignment as input and tries to improve it. Neither the assignment to be improved nor the assignments obtained are required to be feasible.

In the following, we generically refer to N as the carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$. Moreover, for a (partial) assignment y for N, the set E^y is defined as $E^y = \big\{vw \in E \mid y(v), y(w) \text{ are defined}\big\}$. The "cost" of the (partial) assignment is

$$\operatorname{cost}(y) \;=\; \operatorname{interf}(y) + \operatorname{infeas}(y),$$

where

$$\operatorname{interf}(y) \;=\; \sum_{\substack{vw \in E^y,\\ y(v) = y(w)}} c^{co}(vw) \;+\; \sum_{\substack{vw \in E^y,\\ |y(v) - y(w)| = 1}} c^{ad}(vw)$$

and

$$\begin{aligned}\operatorname{infeas}(y) \;&=\; M_I \;\big|\big\{vw \in E^y \mid |y(v) - y(w)| < d(vw)\big\}\big| \\ &\quad+\; M \;\big|\big\{v \in V \mid y(v) \text{ undefined}\big\}\big|.\end{aligned}$$

M_I and M are parameters. The definition of $\operatorname{cost}(y)$ does not penalize the use of locally unavailable channels, because our methods never assign an unavailable channel. The cost of a carrier/channel combination (v, f), $f \in C \setminus B_v$, with respect to an assignment y is defined as

$$\operatorname{cost}(y; (v, f)) = \operatorname{interf}(y; (v, f)) + \operatorname{infeas}(y; (v, f)),$$

where

$$\operatorname{interf}(y; (v, f)) = \frac{1}{2} \sum_{\substack{vw \in E:\\ y(w) \text{ defined},\\ f = y(w)}} c^{co}(vw) \;+\; \frac{1}{2} \sum_{\substack{vw \in E:\\ y(w) \text{ defined},\\ |f - y(w)| = 1}} c^{ad}(vw)$$

and

$$\operatorname{infeas}(y; (v, f)) = \frac{M_I}{2} \;\big|\big\{vw \in E \mid y(w) \text{ defined}, |f - y(w)| < d(vw)\big\}\big|.$$

Together with

$$\operatorname{cost}(y; (v, -)) = M$$

as the cost for not assigning carrier v, we obtain that cost(y) is equal to the sum over the costs for all combinations $(v, y(v))$ if v is assigned or $(v, -)$ if v is not assigned. We set $M = 2$, and we let M_I be large enough to penalize separation violations more heavily than interference under "usual circumstances."

We explain four improvement heuristics in the following, namely, Iterated 1-Opt, Vds, k-Opt, and Mcf. The first three heuristics rely on classical local improvement steps and have already been applied numerous times in other contexts. The last method does not easily fit into the framework of local improvement and its motivation is given by the orientation model (3.7) described in Section 3.3.2.

4.3.1 Iterated 1-OPT

Our simplest improvement method, Iterated 1-Opt, repeatedly applies an *1-opt step*, where the assignment of one carrier is changed to the current best channel. The 1-opt steps themselves are organized in passes. Within a *pass*, each carrier is considered once, according to a given order. This order is determined at the beginning of a pass by ordering the carriers decreasingly according to their cost. One or more passes may be performed up to the point where no further improvement is achieved. Algorithm 4 is a schematic formulation of the procedure.

1-opt step

pass

Algorithm 4 Iterated 1-Opt

```
Input: carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$,
       (partial) assignment $y_0$
Output: an assignment $y$, possibly infeasible
  {Initialization}
  $y := y_0$
  for all $v \in V$ do
    sort carriers into decreasing order $O$ according to cost$(y; (v, y(v)))$
                      {unassigned carriers should be at the beginning of $O$}
  end for
  {Pass}
  for all carriers $v \in V$, according to order $O$, do
    pick $f$ s. t. cost$(y; (v, f)) = \min_{g \in C \setminus B_v}$ cost$(y; (v, g))$
    $y(v) := f$
    update the order $O$ to conform with the initial sorting criterion
  end for
  {$y$ holds the resulting assignment}
```

ITERATED 1-OPT is a classical local search method. The states in the search space are the assignments, the neighborhood relation is defined through the 1-opt step, and the merit of moving to a neighbor is the difference of the costs. A comprehensive survey on local search methods is given by Aarts and Lenstra [1997], for example.

local minimum

Unless the current assignment constitutes a *local minimum* with respect to the neighborhood relation and the cost structure, consecutive passes result in repeated improvements. We iterate while improvements are obtained, and computational experiments indicate no need for some tailing-off control in practice.

Fibonacci heaps (see Cormen *et al.* [1990, Chapter 21]) are used to determine which carrier to consider next and what channel to assign to that carrier. We observe that the running time of a single pass is in $\mathcal{O}(|C||E|\log|C| + |V|\log|V|)$ and that the required space is in $\mathcal{O}(|C||V| + |E|)$. In theory, the number of improving passes is not polynomially bounded in the size of a carrier graph, because that number may depend exponentially on the values of c^{co} and c^{ad}. Computational results for the ITERATED 1-OPT are reported in Sections 5.3.1 and 5.4.

4.3.2 Variable Depth Search

The VDS heuristic is an implementation of the concept of variable depth neighborhood search as introduced by Lin and Kernighan [1973], see also Aarts and Lenstra [1997, Chapter 1]. The carriers are sorted into some order O, and the assignment of each carrier is changed one by one according to this order. The best alternative channel to the presently assigned one is tentatively selected. Once all carriers have been considered, the sequence of tentative changes is scanned from start to end and the cost of every intermediate assignment is recorded. Finally, the tentative changes are committed up to the point, where the first assignment of least cost is obtained. All further tentative changes are rejected.

pass

Like in the case of ITERATED 1-OPT, we call the processing of one order a *pass*. Passes are performed as long as improvements are achieved. A formal description of VDS is given as Algorithm 5.

VDS is a local search method. Given an assignment, another assignment is its neighbor if this assignment can be obtained by the improvement process described above for some order of the vertices. Hence, the neighborhood of a given assignment consists of all assignments that can be produced by executing the tentative assigning and the committing part of Algorithm 5 for some order O. (In that respect, our method

Algorithm 5 VDS

```
Input: carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$,
       assignment $y_0$
Output: an assignment $y$, possibly infeasible
  {Initialization}
  $\hat{y} := y_0$
  $c^* := \text{cost}(y_0)$
  for all carriers do
    sort carriers into decreasing order $O$ according to $\text{cost}(\hat{y}; (v, \hat{y}(v)))$
  end for
  {Tentative changes}
  for all carriers $v \in V$, according to the order $O$, do
    pick $f$ s. t. $\text{cost}(\hat{y}; (v, f)) = \min_{g \in C \setminus (B_v \cup \{\hat{y}(v)\})} \text{cost}(\hat{y}; (v, g))$
    $\hat{y}(v) := f$
    if $\text{cost}(\hat{y}) < c^*$ then
      $c^* := \text{cost}(\hat{y})$
      $v^* := v$
    end if
  end for
  {Commit changes}
  $y := y_0$
  for all carriers $v \in V$ up to $v^*$ according to the order $O$ do
    $y(v) := \hat{y}(v)$
  end for
  if $\text{cost}(y) < \text{cost}(y_0)$ then
    $y_0 := y$
    goto Initialization    {iterate in case of improvement}
  end if
  {$y$ holds the resulting assignment}
```

is rather limited because only a single neighbor is determined.) If the cost of the original assignment is already minimal among the sequence of obtained assignments, the search is trapped and aborted.

The number of improving executions of the core part of VDS is not polynomially bounded in the size of a carrier graph in general. The complexity analysis is therefore only given for a single pass. The running time of VDS is in $\mathcal{O}(|E||C| \log|C| + |V| \log|V|)$ and the required space is in $\mathcal{O}(|C||V| + |E|)$. Computational results for the VDS heuristic are reported in Sections 5.3.2 and 5.4, see also Section 5.5.

4.3.3 k-Opt

An extended version of the Iterated 1-Opt procedure may choose more than just one carrier at the time and optimize their channels simultaneously with respect to each other as well as with respect to the remaining fixed assignments. In the k-Opt heuristic, k carriers are picked out each time. The number of possible assignments on k carriers is $|C|^k$, in principle. This quantity grows exponentially with k, but the amount of feasible combinations is usually much smaller in practice. The reason for this reduction is typically not due to genuine locally blocked channels. Instead, it stems from the restrictions imposed by the carriers which are not to be changed. Nevertheless, the possibilities are still too numerous to simply rely on enumeration. We use a branch-and-cut algorithm instead. (The principle of a branch-and-cut algorithm is briefly explained in Section 7.4. Comprehensive treatments are given by Jünger, Reinelt, and Thienel [1995b] and Thienel [1995], for example.)

The stable set model, see Section 3.3.1, forms the basis for the procedure. Let K denote the set of carriers to optimize over, and let $A_v \subseteq C \backslash B_v$ denote the set of channels assignable to $v \in K$ without causing separation violations with any carrier in $V \backslash K$. This restricted version of the corresponding stable set formulation (3.6) is stated here for convenience. Recall the definitions of the sets E^d, E^{co}, and E^{ad} from (3.3), (3.4), and (3.5), respectively.

$$\min \sum_{\substack{vw \in E^{co}:\\ v \in K}} c^{co}_{vw} z^{co}_{vw} + \sum_{\substack{vw \in E^{ad}:\\ v \in K}} c^{ad}_{vw} z^{ad}_{vw}$$

s. t.

$$\sum_{f \in A_v} y^f_v = 1 \quad \forall v \in K \tag{4.1a}$$

$$y^f_v + y^g_w \le 1 \quad \forall vw \in E^d, v, w \in K, |f - g| < d_{vw} \tag{4.1b}$$

$$y^f_v + y^f_w - z^{co}_{vw} \le 1 \quad \forall vw \in E^{co}, v, w \in K, f \in A_v \cap A_w \tag{4.1c}$$

$$y^f_v + y^{f-1}_w - z^{ad}_{vw} \le 1 \quad \forall vw \in E^{ad}, v, w \in K, f \in A_v, f - 1 \in A_w \tag{4.1d}$$

$$y^f_v - z^{co}_{vw} \le 1 \quad \forall vw \in E^{co}, v \in K, w \notin K, y(w) = f \in A_v \tag{4.1e}$$

$$y^f_v - z^{ad}_{vw} \le 1 \quad \forall vw \in E^{ad}, v \in K, w \notin K, y(w) \pm 1 = f \in A_v \tag{4.1f}$$

$$y^f_v \in \{0,1\} \quad \forall v \in K, f \in A_v \tag{4.1g}$$

$$z^{co}_{vw} \in \{0,1\} \quad \forall vw \in E^{co} : v \in K \tag{4.1h}$$

$$z^{ad}_{vw} \in \{0,1\} \quad \forall vw \in E^{ad} : v \in K \tag{4.1i}$$

Our implementation builds on the ABACUS framework developed by Thienel [1995]. The objective here is not to design an effective branch-and-cut algorithm to solve FAP in general. Our own computational experiments as well as the experiences of other research groups, compare Aardal, Hipolito, van Hoesel, Jansen, Roos, and Terlaky [1995] and Jaumard, Marcotte, and Meyer [1998], indicate that even rather small carrier networks with 25 carriers and 20 channels, say, are hardly solvable to optimality. The LP relaxation is highly degenerate.

Due to storage capacity and running time considerations, we do not start with the full description of the program (4.1). Initially, only the constraints listed under (4.1a), (4.1c), (4.1d), (4.1e), and (4.1f) are contained in the LP relaxation. In addition to that, the relaxations of (4.1g), (4.1h), and (4.1i) to arbitrary values between 0 and 1 are also imposed. The constraints (4.1b), reflecting the minimum separation requirements, are separated at need. They are not considered individually, however, but in the stronger, aggregate form of the well-known clique constraints. More precisely, given a fractional solution of the LP relaxation, a graph (W, F) is constructed from the nonzero y_v^f-variables, where

$$W = \big\{(v, f) \in K \times C \mid f \in A_v,\ y_v^f > 0\big\}$$

and

$$\begin{aligned} F = \big\{\{(v, f), (w, g)\} \mid {} & (v, f), (w, g) \in W, \\ & \text{“}y_v^f + y_w^g \leq 1\text{” is listed in (4.1b)}\big\}. \end{aligned}$$

The vertices of the graph are weighted with the y_v^f-values. We associate to every subset of the vertex set the total weight of its vertices. By means of an algorithm similar to that proposed by Carraghan and Pardalos [1990], we compute a maximum weight clique Q in the graph (W, F). If the weight of Q exceeds 1, then the valid inequality

$$\sum_{(v,f) \in Q} y_v^f \leq 1$$

is violated, and this constraint is added to the LP. Notice that the problem of finding maximum weighted clique in a graph is $\mathcal{NP}$-hard, compare Ausiello *et al.* [1999, Appendix B, GT 22]. In our application, however, such a clique is usually found sufficiently fast.

We also make use of ABACUS's ability to remove slack constraints from the LP in order to keep the LP small. The branching is performed on z^{co}- and z^{ad}-variables only. Among those, a variable of value closest to 0.5 is selected. There are, of course, many ways to improve our coarse implementation. Two examples are the use of refined branching schemes and the separation of other valid inequalities (such as a blend of the feasibility constraints (4.1b) with the interference-accounting constraints (4.1c) and (4.1d) in the form of transitive cliques, see Schulz [1996, Chapter 4]).

Computational results for the K-OPT heuristic are contained in Sections 5.3.3 and 5.4, see also 5.5.

4.3.4 Min-Cost Flow

The MCF method is originally proposed as the inner part of a two-level heuristic to solve the frequency assignment problem. The two levels correspond to the "outer" and the "inner" optimization goal in the orientation model explained in Section 3.3.2. Our presentation follows along the lines of the more detailed exposition given by Borndörfer *et al.* [1998b]. We treat only the most basic case here. The corresponding restrictions are:

restrictions

(i) no channel from the spectrum is blocked locally;

(ii) for all carrier pairs with no required separation the co-channel interference is at least twice as much as the adjacent channel interference.

We assume that the carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$ satisfies the above restrictions and that y_0 is an associated feasible assignment. A directed version of the graph (V, E) underlying the carrier network is defined: let each edge point from the vertex with the smaller channel to the vertex with the larger channel; if the channels of both vertices are the same, either orientation is fine as long as the resulting orientation does not contain a (directed) cycle. Such an orientation can always be computed in polynomial time. Let (V, A) denote the directed graph obtained in this way. Notice that y_0 is compatible with A in the sense of Section 3.3.2. The MCF method finds the best feasible assignment y which is also compatible with A.

We set up a linear program which is closely related to the dual of the LP relaxation of (3.7). The primal program is modified slightly before forming its dual. The changes are the following. We add constraints under (3.7b) for all remaining edges with $d(vw) = 0$, and we drop the upper bound constraints on the z^{co}_{vw}- and z^{ad}_{vw}-variables. Both types of modifications do not affect the optimal value of (3.7), but will be convenient in the following.

Notice that for a fixed orientation of the edges, each of the constraint pairs (3.7b), (3.7c), and (3.7d) reduces to one constraint. Furthermore, as a consequence of the above restrictions, the constraints (3.7e) disappear from the orientation model formulation (3.7). The linear program we are interested in is the dual to the LP relaxation of the resulting ILP.

The dual variables x^d_{vw}, x^{co}_{vw}, and x^{ad}_{vw} are associated to the constraints (3.7b), (3.7c), and (3.7d), respectively. The dual variables l_v and u_v are associated with the upper and lower bound constraints which are implicitly given by (3.7f). With $\check{C} = \min\{f \mid f \in C\}$ and $\hat{C} = \max\{f \mid f \in C\}$ the dual program now reads as follows:

$$\max_{\substack{x^d, x^{co}, x^{ad},\\ u, l}} \sum_{vw \in E} d_{vw} x^d_{vw} + \sum_{vw \in E^{co}} x^{co}_{vw} + \sum_{vw \in E^{ad} \setminus E^{co}} x^{ad}_{vw} + \sum_{v \in V} (\check{C}\, l_v - \hat{C}\, u_v)$$

s. t.

$$\begin{array}{lll}
- \sum\limits_{\substack{vw \in E:\\ (v,w) \in A}} x^d_{vw} + \sum\limits_{\substack{vw \in E:\\ (w,v) \in A}} x^d_{vw} & & \\
- \sum\limits_{\substack{vw \in E^{co}:\\ (v,w) \in A}} x^{co}_{vw} + \sum\limits_{\substack{vw \in E^{co}:\\ (w,v) \in A}} x^{co}_{vw} & & \\
- \sum\limits_{\substack{vw \in E^{ad} \setminus E^{co}:\\ (v,w) \in A}} x^{ad}_{vw} + \sum\limits_{\substack{vw \in E^{ad} \setminus E^{co}:\\ (w,v) \in A}} x^{ad}_{vw} & & \\
- \, u_v \qquad + l_v & \leq 0 & \forall v \in V \qquad (4.2a) \\
x^{co}_{vw} & \leq c^{co}_{vw} & \forall vw \in E^{co} \qquad (4.2b) \\
x^{ad}_{vw} & \leq c^{ad}_{vw} & \forall vw \in E^{ad} \setminus E^{co} \qquad (4.2c) \\
x^d_{vw} & \geq 0 & \forall vw \in E \\
x^{co}_{vw} & \geq 0 & \forall vw \in E^{co} \\
x^{ad}_{vw} & \geq 0 & \forall vw \in E^{ad} \setminus E^{co} \\
l_v, u_v & \geq 0 & \forall v \in V
\end{array}$$

The program (4.2) may be solved by computing a min-cost flow (see Ahuja, Magnanti, and Orlin [1992], for example) on an auxiliary graph. First, observe that the inequality constraints (4.2a) may be turned into equations, because any slack can be eliminated by increasing l_v with positive or no effect on the objective value (due to $\check{C} \geq 0$). Further necessary transformations are:

- the objective function of (4.2) is multiplied by -1 and the max operator replaced by the min operator;
- the constraints (4.2a) (with equality sign) are multiplied by -1;
- a new vertex s is added, and the variables l_v and u_v are replaced by x_{vs} and x_{sv}.

The constraints (4.2a) read now as

$$\begin{aligned}\sum_{\substack{vw\in E:\\(v,w)\in A}} x^d_{vw} - \sum_{\substack{vw\in E:\\(w,v)\in A}} x^d_{vw} + \sum_{\substack{vw\in E^{co}:\\(v,w)\in A}} x^{co}_{vw} - \sum_{\substack{vw\in E^{co}:\\(w,v)\in A}} x^{co}_{vw}& \\ + \sum_{\substack{vw\in E^{ad}\setminus E^{co}:\\(v,w)\in A}} x^{ad}_{vw} - \sum_{\substack{vw\in E^{ad}\setminus E^{co}:\\(w,v)\in A}} x^{ad}_{vw} + x_{sv} - x_{vs} &= 0 \qquad \forall v \in V.\end{aligned} \tag{4.3}$$

By means of those transformations, the problem (4.2) turns into the problem of computing a min-cost flow in a directed graph with parallel edges. The edges corresponding to x^d_{vw}, x_{vs}, and x_{sv} are uncapacitated, the edges corresponding to x^{co}_{vw} and x^{ad}_{vw} have limited capacity. There is no sink or source. Instead, we are looking for a circulation, meeting the flow conservation constraints (4.3). Numerous methods for solving such a problem are described in the literature, see Ahuja *et al.* [1992, Chapters 9–11], for example. Given an optimal circulation, we may construct integral node potentials π_v (a dual solution) with $\pi_s = 0$. Setting $y(v) = \pi_v$ for all $v \in V$, we obtain the desired frequency assignment. A more detailed discussion on this connection is given by Borndörfer *et al.* [1998b].

Algorithm 6 gives a sketch of the employed procedure, where we also show how to handle the case in which the above restriction (ii) is not met. Mcf reduces the adjacent channel interference of the corresponding edges temporarily in order to meet this constraint. That way, Mcf turns into a heuristic with respect to its own optimization goal. The flow along each arc in the associated directed graph incurs integral cost. From general min-cost flow theory, we may conclude that the dual solution is also integral. In fact, the dual solution induces a feasible frequency assignment.

Restriction (i) may also be relaxed in the following sense. Let v be a carrier where some channel is blocked locally, that is, $B_v \neq \emptyset$. The local exceptions puncture the set of otherwise contiguous channels in C. We may think of the set $C \setminus B_v$ as the union of maximal intervals of window

active window

contiguous channels. We call each such interval a *window*, and, given a feasible assignment y_0, the window $I_v^{y_0}$ with $y_0(v) \in I_v^{y_0}$ is called the *active window* for v. If local blockings are present, the objective function of (4.2) can be changed by replacing $\check{C}$ and $\hat{C}$ by $\check{I}_v^{y_0} = \min\{f \mid f \in I_v^{y_0}\}$ and $\hat{I}_v^{y_0} = \max\{f \mid f \in I_v^{y_0}\}$, respectively. The effect of this change is that each node receives a channel from the active window.

The auxiliary directed graph is easily constructed in $\mathcal{O}(|E|)$ time. The min-cost flow problem is solved by means of a Network Simplex Method implementation, see Löbel [1997]. The space requirement of this algorithm is in $\mathcal{O}(|E|)$, but its worst-case running time is exponential in the input size. Although there are strongly polynomial min-cost flow algorithms (see Ahuja *et al.* [1992, Chapter 10]), we choose this implementation of the Network Simplex Algorithm for its typically competitive running time in practice. Computational results for the Mcf heuristic are reported in Sections 5.4.

Algorithm 6 Mcf

```
Input: carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$,
       feasible assignment $y_0$
Output: a feasible assignment $y$
  {Initialization: orient edges of $G = (V, E)$}
  for all edges $vw \in E$ with $y_0(v) \neq y_0(w)$ do
    $O := O \cup \begin{cases} \{(v,w)\}, & \text{if } y_0(v) < y_0(w) \\ \{(w,v)\}, & \text{if } y_0(v) > y_0(w) \end{cases}$
  end for
  for all edges $vw \in E$ with $y_0(v) = y_0(w)$ do
    $O := O \cup \{(v,w)\}$ or $O := O \cup \{(w,v)\}$
                                {ensure that the final orientation is acyclic}
  end for
  {Construction of an auxiliary directed graph $D = (W, A)$}
  $W := V$
  for all $(v,w) \in O$ without $d_{vw} = 0,\ c^{co}_{vw} > 0,\ c^{ad}_{vw} > 0$ do
    if $d_{vw} = 0,\ c^{co}_{vw} > 0$ then
      $A := A \cup \{(v,w)\}$ with capacity $c^{co}_{vw}$ and cost $-1$
    else if $d_{vw} = 1,\ c^{ad}_{vw} > 0$ then
      $A := A \cup \{(v,w)\}$ with capacity $c^{ad}_{vw}$ and cost $-1$
    end if
    $A := A \cup \{(v,w)\}$ with capacity $\infty$ and cost $-d_{vw}$
  end for
  for all $(v,w) \in O$ with $d_{vw} = 0,\ c^{co}_{vw} > 0,\ c^{ad}_{vw} > 0$ do
    $W := W \cup \{s_{vw}\}$
    $A := A \cup \{(v, s_{vw})\}$ with cap. $\max\{c^{co}_{vw} - c^{ad}_{vw}, \frac{1}{2}c^{co}_{vw}\}$ and cost $-1$
    $A := A \cup \{(v, s_{vw})\}$ with cap. $\infty$ and cost $0$
    $A := A \cup \{(s_{vw}, w)\}$ with cap. $\min\{c^{ad}_{vw}, \frac{1}{2}c^{co}_{vw}\}$ and cost $-1$
    $A := A \cup \{(s_{vw}, w)\}$ with cap. $\infty$ and cost $0$
  end for
  $W := W \cup \{s\}$
  for all $v \in V$ do
    $A := A \cup \{(s, w)\}$ with capacity $\infty$ and cost $-\check{I}^{y_0}_v$
    $A := A \cup \{(w, s)\}$ with capacity $\infty$ and cost $+\hat{I}^{y_0}_v$
  end for
  {Solve min-cost flow problem}
  solve resulting min-cost flow problem on $D = (W, A)$
  let $y(v) := \pi(v)$, $v \in V$, for an optimal, integral dual solution $\pi$
  {$y$ holds the resulting feasible assignment}
```

Chapter 5

Computational Studies

In this chapter, we report on computational experiments performed with the frequency planning heuristics described in the previous chapter. Results are given for eleven realistic benchmark scenarios.

We introduce these scenarios in Section 5.1. The individual performance of the heuristics and their parameter interdependence is analyzed in Sections 5.2 and 5.3. The concerted acting of the heuristics is studied in Section 5.4, where we also select our favorite combination of methods together with the relevant parameter settings. We focus on three out of the eleven scenarios for these extensive studies. In Section 5.5, our favorite combinations of heuristics are applied to all benchmark scenarios and a detailed analysis of the resulting frequency plans is given. Finally, in Section 5.6, we compare our swiftly generated plans with those computed by the elaborate planning heuristic procedure of Hellebrandt and Heller [2000], explained in Section 5.1.2. We give recommendations for the use of the various heuristics in practice.

The following technical information on the computer system environment should allow to estimate how the running times of the heuristics provided here translate to other system environments. The computations are performed on an IBM ThinkPad 600X with an Intel Pentium III processor, operating at 650 MHz clock speed, and equipped with 576 MB of system memory. The operating system is GNU/Linux in the SuSE 6.4 distribution, kernel version 2.2.14.

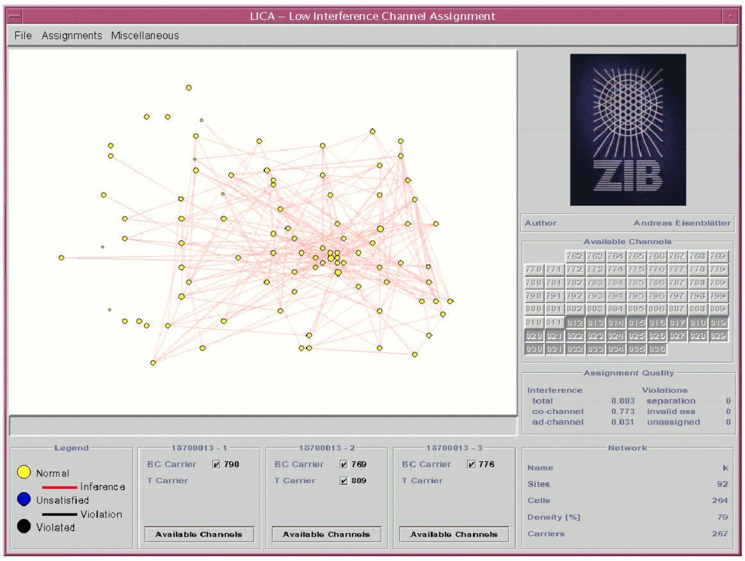
P III, 650 MHz
GNU/Linux

The heuristics are implemented in the programming language C++, using data structures for graphs and priority queues from the Library of Efficient Data structures and Algorithms (LEDA), version 3.6.1, see Mehlhorn and Näher [1999]; LEDA 3.6.1 [1998]. The compilations are performed by means of GNU g++, version 2.95.2, with `-mcpu=i686 -O6` as optimization flags. The min-cost flow problem arising within the MCF heuristic is solved using the Network Simplex Method implementation of Löbel [1997], version 1.0. ABACUS, version 2.2, is used to implement the K-OPT heuristic, and all LPs are solved using CPLEX, version 6.5.

C++
LEDA 3.6.1
ABACUS 2.2
CPLEX 6.5

The reported running times of the heuristics do not include the initialization of the test framework, the reading of the data, or the setup of the carrier network.

A graphical user interface (GUI) allows to call the various methods and to visualize the interference incurred by a frequency plan. This user interface was implemented using the programming language Java for a demonstration at the CeBIT 99 fair. Figure 5.1 shows the display panel of the program. Within the display panel, geographical and other information on the planning instance K, see Section 5.1.1, is listed. In addition, information on the current frequency plan is given in numbers (in the lower right panel) as well as pictorially (in the upper left panel). Every node in this panel denotes a site, and edges between two sites represent interference among TRXs of the two sites.



Author Andreas Eisenblätter

Available Channels

		762	763	764	765	766	767	768	769
770	771	772	773	774	775	776	777	778	779
780	781	782	783	784	785	786	787	788	789
790	791	792	793	794	795	796	797	798	799
800	801	802	803	804	805	806	807	808	809
810	811	812	813	814	815	816	817	818	819
820	821	822	823	824	825	826	827	828	829
830	831	832	833	834	835	836			

Assignment Quality

Interference		Violations	
total	0.803	separation	0
co-channel	0.773	invalid ass	0
ad-channel	0.031	unassigned	0

18700013 - 1
BC Carrier ☑ 790
T Carrier

18700013 - 2
BC Carrier ☑ 769
T Carrier ☑ 809

18700013 - 3
BC Carrier ☑ 776
T Carrier

Network

Name	k
Sites	92
Cells	264
Density [%]	79
Carriers	267

Figure 5.1: GUI for automatic frequency planning

5.1 Benchmarks

A fair comparison of computational experiments is often hard to achieve. The running times of the same method, for example, depend heavily on the computing environment as well as on the actual implementation of the method. This can hardly be remedied. Another source of uncertainty, however, can be remedied by means of using publicly available and established input data. Within the Cost 259 action, the subgroup on "Standard Scenarios for Frequency Planning" established a collection of realistic GSM frequency planning scenarios in order to allow a sound comparison of different planning methods. Several of these test scenarios are used here, plus one additional scenario. All our scenarios are available via the Internet from the FAP web [2000]. They are introduced in Section 5.1.1, together with an analysis of their characteristics.

In Section 5.1.2, we describe the frequency planning heuristic proposed by Hellebrandt and Heller [2000]. This heuristic is presently the most competitive one for our scenarios, and we use its results as reference in the comparison of our own, faster methods.

5.1.1 Test Instances

Our benchmark instances are introduced in the following. For notational convenience, we use other labels than the original ones, but provide the original names in the descriptions. The brief descriptions of the planning scenarios are mostly taken from Eisenblätter and Kürner [2000]. Each scenario gives rise to a carrier network.

Data for a GSM 1800 network with 92 active sites, 264 cells, and an average of 1.01 TRXs per cell. Fifty contiguous channels form the spectrum. (Provided by E-Plus Mobilfunk GmbH & Co. KG.) K

Data for a GSM 1800 network with 649 active sites and 1886 cells. The parameter t scales the traffic demand. The available spectrum consists of 75 contiguous channels. (Provided by E-Plus Mobilfunk GmbH & Co. KG as "bradford_nt-d-eplus.") B[d]

The basic traffic load is drawn at random according to a distribution observed empirically by Gotzner, Gamst, and Rathgeber [1997]. This traffic is then scaled with the factor d equal to 0, 1, 2, 4, and 10 prior to applying the Erlang-B formula in order to obtain the required number of TRXs per cell. The resulting average numbers of TRXs per cell are 1.00, 1.05, 1.17, 1.47, and 2.20, respectively. The different traffic demands may be seen as the evolution of a network over time. The interference predictions base on signal propagation predictions according to the "eplus" model, see Section 2.3.2. The two alternative prediction models "free space" and "race", also mentioned in Section 2.3.2, are not considered here, because in most of the cases a frequency plan without any interference is easily obtainable, see Eisenblätter and Kürner [2000].

SIE1 Data for a GSM 900 network with 179 active sites, 506 cells, and an average of 1.84 TRXs per cell. The available spectrum consists of two contiguous blocks containing 20 and 23 channels, respectively. (Provided by Siemens AG as "siemens1.")

SIE2 Data for a GSM 900 network with 86 active sites, 254 cells, and an average of 3.85 TRXs per cell. The available spectrum consists of two contiguous blocks containing 4 and 72 channels, respectively. (Provided by Siemens AG as "siemens2.")

SIE3 Data for a GSM 1800 network with 366 active sites, 894 cells, and an average of 1.82 TRXs per cell. The available spectrum comprises 55 contiguous channels. (Provided by Siemens AG as "siemens3.")

SIE4 Data for a GSM 900 network with 276 active sites, 760 cells, and an average of 3.66 TRXs per cell. The available spectrum consists of 39 contiguous channels. (Provided by Siemens AG as "siemens4.")

SW Data for a GSM 900 network in a city with many locally blocked channels. On average, 2.09 TRXs are installed per cell. There are 148 cells with 1 to 4 TRXs and 707 neighbor relations. In general, 52 channels in two contiguous blocks of sizes 3 and 49 are available, but 136 cells have local restrictions. Only 15 channels are available in the worst case; the median of available channels per cell is 29. (Other figures concerning the availability of channels are provided with the scenario. We give the results of our own computations.) Together with the scenario, a partial assignment is supplied which is supposed to be extended. The restrictions imposed by the unchangeable TRXs are already taken into account in the above figures. (Provided by Swisscom Ltd. as "Swisscom.")

Further details concerning the underlying GSM network are listed in Table 5.1; namely: the *number of sites* in the planning area; the *number of cells* (no site hosts more than three cells); the *spectrum* size or, in the presence of globally blocked channels, the sizes of the contiguous portions in the spectrum; the *average number of TRXs per cell*; and the *maximum number of TRXs per cell.* Notice that in each of the scenarios with globally blocked channels, the resulting gap in the spectrum exceeds the maximal required separation. Thus, there is no direct coupling between distinct contiguous portions of the spectrum.

Three further figures are given for each scenario in Table 5.1. These are easiest explained in terms of the carrier network which is obtained

	# sites	# cells	avg. TRXs/cell	max. TRXs/cell	spectrum size	min. # channels	avg. # channels	avg. # adj. cells	max. # adj. cells	diameter
K	92	264	1.01	2	50	50	50.00	149.30	236	3
B[d]	649	1886	Table 5.2		75	75	75.00	256.38	779	5
Sie1	179	506	1.84	4	20, 23	43	43.00	41.76	104	7
Sie2	86	254	3.85	6	4, 72	76	76.00	121.98	225	3
Sie3	366	894	1.82	3	55	12	51.25	75.76	267	10
Sie4	276	760	3.66	5	39	39	39.00	70.31	184	5
Sw	87	148	2.09	4	3, 49	15	29.39	11.43	44	10

Table 5.1: Scenario characteristics

if each cell operates only one TRX. Then, every carrier in the network corresponds to a cell, and the graph underlying this network reflects the relations among the cells. For this graph, the average degree *(average number of adjacent cells)*, the maximum degree *(maximum number of adjacent cells)*, and the *diameter* (of the largest connected component) are listed. See Appendix A for definitions of these terms.

We point out a few peculiarities of those graphs. First, for reasons which are not entirely clear, the graph is not connected for all scenarios. Sometimes, a few cells form small subgraphs, which are isolated from the rest. (This phenomenon might be due to shielded indoor cells.) The sizes of the small connected components are as follows: 2×1, 1×2, and 2×3 for B[1], 1×34 for Sie3, and 1×1 for Sw. All those small components are cliques with the exception of Sie3, where the largest clique contains only 12 of the 34 vertices.

d	# TRXs	
	avg.	max.
0	1.00	1
1	1.05	3
2	1.17	5
4	1.47	9
10	2.20	12

Table 5.2: Properties of B[d] depending on the traffic factor d

Second, looking at the column showing the average number of adjacent cells, we see that cells are adjacent to surprisingly many other cells on average. In scenario K, for example, each cell is adjacent to more than half of all cells. (We believe that the comparatively high numbers for the scenarios provided by E-Plus Mobilfunk GmbH & Co. KG are due to the use of a 20 dB threshold in distinguishing between interference affected and unaffected pixels, see Section 2.3.2 for further explanations.)

Finally, the last column shows that the cells are all rather "nearby" in almost all scenarios. Recall that the diameter in a connected graph is the maximum length among all shortest paths between pairs of vertices. Thus, a diameter of 2, for example, implies that for every nonadjacent pair of cells, there is one cell being adjacent to both of them. If the graph has more than one connected component, we give the diameter of the largest component. The diameter of the small connected component of Sie3 is 5. The diameter of all other small components is either 0 or 1.

After having looked at the scenarios, we turn to the carrier networks derived from the scenarios. The focus is on the undirected graphs underlying the scenarios and on a few characteristics of the edge labelings d, c^{co}, and c^{ad}. The results of our analysis are given in Table 5.3, which is organized in five blocks of columns. In the first column, the label of the associated scenario is given. The next block contains information on the graph, namely, its number of vertices, its edge density (that is, number of edges relative to the maximum possible number of $\frac{|V|(|V|-1)}{2}$), the average degree of its vertices as well as the maximum degree, and the size of a maximum clique. The third block addresses the minimum separation requirements as specified by the labeling d. The total number of edges with nonzero separation requirements as well as the breakdown of the total according to the required separation is displayed. The forth and fifth block provide information on the co- and adjacent channel interference labelings c^{co} and c^{ad}, respectively. In each block, we list the total number of interference relations, their average and maximal interference values as well as the sum over all interference values.

There are again some noteworthy facts. First, the average degree of a carrier is significantly higher than the number of available channels in all the carrier networks except for Sw. This indicates that the channel assignment for different cells has to be carefully tuned in order to produce good frequency plans.

Second, the size of the maximum cliques in most carrier networks is larger than the number of available channels. For those carrier networks every feasible assignment must incur interference. We investigate this further in Chapter 6.

	$\vert V\vert$	density [%]	avg. degree	max. degree	max. clique	$\vert\{e\in E : d(e)\neq 0\}\vert$	$\vert\{e\in E : d(e)=1\}\vert$	$\vert\{e\in E : d(e)=2\}\vert$	$\vert\{e\in E : d(e)=3\}\vert$	$\vert\{e\in E : c^{co}(e)\neq 0\}\vert$	$\text{avg}_{\{e\in E : c^{co}(e)>0\}} c^{co}(e)$	$\max_{\{e\in E\}} c^{co}(e)$	$\sum_{e\in E} c^{co}(e)$	$\vert\{e\in E : c^{ad}(e)\neq 0\}\vert$	$\text{avg}_{\{e\in E : c^{ad}(e)>0\}} c^{ad}(e)$	$\max_{\{e\in E\}} c^{ad}(e)$	$\sum_{e\in E} c^{ad}(e)$
K	267	56.57	151.0	238	69	1053[a]	4	1046	0	19111	0.15	1.9	2857	996	0.03	0.8	29
B[0]	1886	13.59	256.4	779	81	7288	0	7288	0	234479	0.09	1.8	20539	4263	0.02	0.8	79
B[1]	1971	13.46	265.3	805	84	7996	203	7707	86	253441	0.09	1.8	22266	4825	0.02	0.8	108
B[2]	2214	13.50	299.0	916	93	10284	1085	8852	347	320684	0.09	1.8	28069	6871	0.03	0.8	212
B[4]	2775	13.44	373.0	1133	120	16663	3620	11981	1062	500805	0.09	1.8	43376	12524	0.04	0.8	505
B[10]	4145	13.41	555.9	1704	174	38234	13798	20653	3783	1113850	0.09	1.8	96559	33548	0.05	0.8	1684
Sie1	930	9.03	84.0	209	52	6039	0	5603	436	33002	0.07	1.7	2452	9911	0.02	0.6	198
Sie2	977	49.17	480.4	877	182	17761	0	16259	1502	216912	0.02	0.5	4530	25615	0.00	0.0	94
Sie3	1623	9.18	149.1	519	78	23093	0	22360	733	97861	0.03	1.1	2609	15069	0.01	0.3	130
Sie4	2785	10.50	292.3	752	100	27964	0	23933	4031	379052	0.03	0.7	10747	26445	0.01	0.0	134
Sw	310	8.29	25.7	94	21	3984	2075	1721	188	0	0.00	0.0	0	2075	0.29	1.0	611

[a] Three constraints require a separation of 4.

Table 5.3: Characteristics of carrier networks

Third, recall from Section 3.1 that the interference labelings are obtained by summing up the two directed interference ratings between pairs of TRXs. Those directed ratings are normalized and take values between 0 and 1. The sum is thus bounded by 2. In several cases the maximal co- or adjacent channel interference is, in fact, close to 2. The occurrence of such heavy interference is considered completely unacceptable in practice. In Section 5.5, we see that such heavy interference can typically be avoided by "good" frequency assignments. This happens either by coincidence, indirectly due to the optimization goal, or is imposed by "tightening the separation" as explained in Section 4.1.2.

Fourth, we see that no co-channel interference is reported for Sw. When generating the scenario, the radio planer apparently decided to rule out this type of interference entirely by adding the necessary separation constraints.

planar graph

Fifth, one might have guessed that the graphs underlying carrier networks are not much denser than planar graphs. (A *planar graph* can be drawn in the Euclidean plane in such a manner that edges are represented by piece-wise straight lines and no edges cross.) Planar graphs are of interest because many generally $\mathcal{NP}$-hard problems on graphs are solvable in polynomial time if restricted to planar graphs. The following relation between the number of vertices and edges in a planar graph is well-known and follows from Euler's formula.

Proposition 5.1. *A planar graph $G = (V, E)$ satisfies $|E| \leq 3\,|V| - 6$.*

For the carrier network K, for example, the expression yields $3 \cdot 267 - 6 = 795$ as opposed to 27388 edges. Even the number of edges with separation constraints is roughly 1.3 times this value. Consequently, the graph underlying the carrier network cannot be planar. The large maximum cliques in the carrier networks as well as the small diameters of the connected components also indicate substantially tighter couplings among large sets of carriers than one might have suspected.

tree decomposition

The degree of interdependencies clearly affects the possibilities for decomposing the optimization problem. Ideally, we would like to be able to independently solve small subproblems to optimality and assemble an optimal solution for the whole problem from the solutions to the subproblems. The stronger the dependencies between the subproblems we choose, the harder is the assembling. In the Ph. D. thesis of Koster [1999], such an approach is described using a *tree decomposition* of a graph, see, for example, Bodlaender [1993] for an introduction to tree decomposition. We do not give details here, but merely state that this approach is successfully applied to graphs with a treewidth of up to 11, say, and that the running time grows exponentially in the treewidth. Because the size of maximum clique minus one is a lower bound for the treewidth, this approach is likely to fail on all our scenarios.

In an heuristic decomposition approach, the subproblems need not to be solved optimally, and the reassembling might introduce controlled degradation. But even in this relaxed sense, no convincing decomposition method has been proposed so far. We explain one of the obstacles to overcome.

maximum clique

We define a labeling $mc\colon V \to \mathbb{Z}_+$ of the carriers in a network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$ by setting $mc(v)$ to the size of the largest clique in which v is contained. (On our instances, a maximum clique can be computed in reasonable time by means of a branch-and-bound algorithm.) We then consider the graphs $G^i = G[\{v \in V \mid mc(v) \geq i\}]$, i. e., the subgraphs of $G = (V, E)$ induced by the set of all carriers which are contained in a clique of size at least i. The graphs G^i are nontrivial for i between 1 and the size of a maximum clique in G.

Figures 5.2, 5.3, and 5.4 show the sizes of the connected components of G^i for the carrier networks of the instances K, B[1], and Sie1, respectively. Obviously, large cliques are no isolated phenomena in these carrier networks. On the contrary, a major portion of all carriers is contained in cliques which are larger than the number of available channels. Similar results can also be observed for the other scenarios. Leaving the small connected components of the carrier networks aside, we also see that only

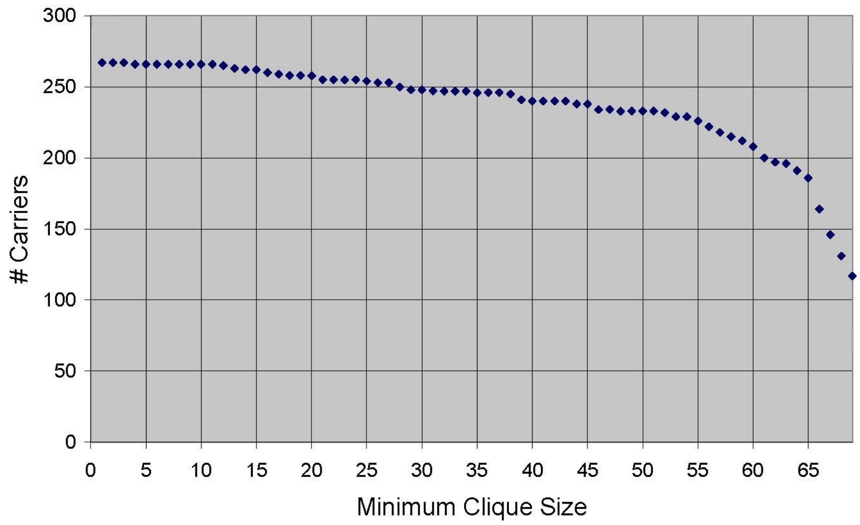


Figure 5.2: Unions of cliques larger than 50 in K

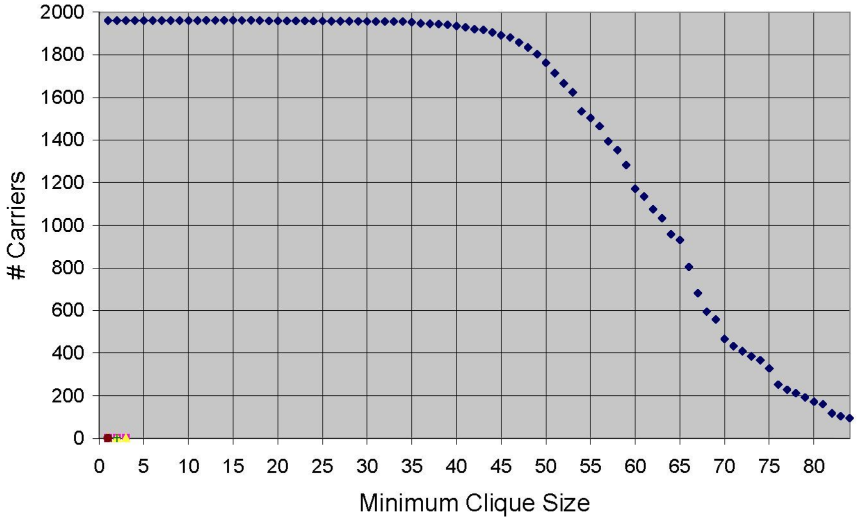


Figure 5.3: Unions of cliques larger than 75 in B[1]

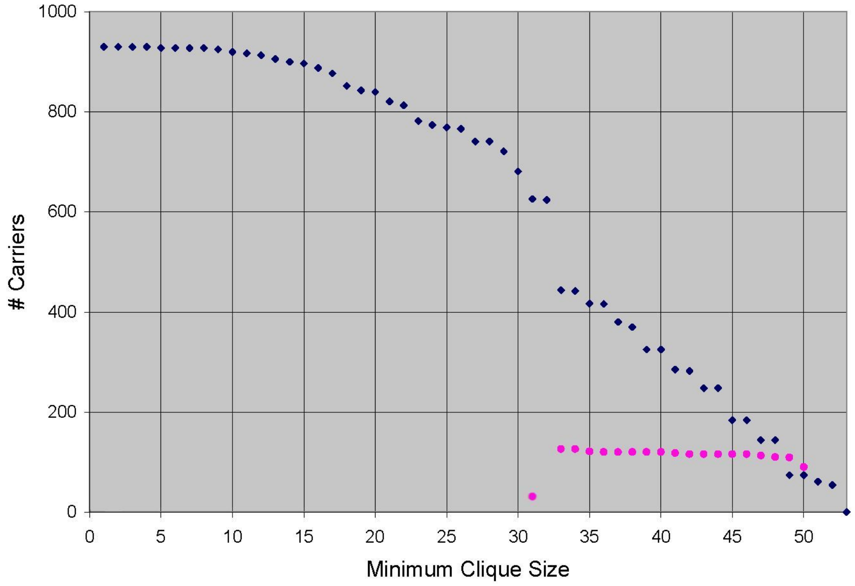


Figure 5.4: Unions of cliques larger than 20 + 23 in Sie1

in case of Sie1 the subgraphs G^i decompose. (For $i = 32$, the cliques observed for 31 disappear, and from $i = 33$ up to 50 a new component splits off from the major chunk.) Good frequency plans thus have to resolve separation conflicts and interference among large sets of strongly interdependent carriers.

preprocessing

In this context, we also report on the effects of the preprocessing techniques proposed in Section 4.1. Table 5.4 documents that these techniques have only a rather limited impact. A few carriers can often be safely excluded from frequency planning due to a small generalized degree. Most dominated carriers, however, are in fact isolated and thus trivially dominated by every other carrier. The scenario Sw may be considered as a minor exception, because the reduced 85 carriers constitute a significant portion of all carriers. Nonetheless, Sw is the scenario for which almost no feasible assignment can be produced using our heuristics from Chapter 4, compare with Section 5.5.

	K	B[0]	B[1]	B[2]	B[4]	B[10]	Sie1	Sie2	Sie3	Sie4	Sw
reduction	15	27	26	22	19	21	60	6	96	10	83
domination	0	2	2	1	0	0	0	1	1	0	5

Table 5.4: Effects of preprocessing

With a number of benchmark scenarios in stock, we now turn to a "benchmark" heuristic to compare our own heuristics with.

5.1.2 Threshold Accepting

We use the heuristic proposed by Hellebrandt and Heller [2000] to compare our computational results with. The implementation of Hans Heller, Siemens AG, Germany, shows presently the best performance in comparison with several methods on the Cost 259 scenarios, see the study of Eisenblätter and Kürner [2000]. The referenced frequency plans are also available via the Internet from the FAP web [2000]. Because the method is not yet published in open literature, we give an outline of this method, drawing freely on our presentation in Correia [2001, Section 4.2.5].

The method of Hellebrandt and Heller [2000] makes use of the Threshold Accepting paradigm. Threshold Accepting is proposed by Dueck and Scheuer [1990] as a variant of Simulated Annealing with a deterministic acceptance criterion: a proposed change of the solution, also called a *move*, is accepted if an improvement is achieved or the deterioration is below a threshold. The value of this threshold declines with the progress of the algorithm. The basic scheme is augmented here by alternating be-

move

tween random changes and local optimization. A sketch of the algorithm is given as Algorithm 7.

Algorithm 7 THRESHOLD ACCEPTING

```
Input: carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$,
       assignment $y_0$, time bound $T_{tot}$
Output: an assignment $y$, possibly infeasible
  {Initialization}
  initialize threshold
  while stopping criterion not met do
    for time $T$ do
      select random move $M$
      if cost of $M$ $<$ threshold then
        execute $M$
      end if
      select random cell and optimize it
    end for
    decrease threshold
  end while
```

The cost of a move is the difference between the cost of the assignments after and before the move. For the most part, Threshold Accepting inherits from Simulated Annealing the variety of popular implementations for each general step. The following choices are made by Hellebrandt and Heller [2000].

Start Solution: The random moves described below are not specially suited for finding a feasible assignment if many hard constraints have to be taken into account. In such a case, a feasible assignment should be provided at the start. Otherwise, any assignment is fine.

Initial Threshold: The threshold value controls how much deviation is allowed from the cost of the current solution. A binary search is performed to identify a threshold, where 80–90% of the proposed random moves are accepted.

Stopping Criterion: The termination is triggered when the acceptance rate for random moves sinks below 5%.

Inner Loop/Threshold Reduction: The running time of the procedure depends on the initial threshold, the factor by which the threshold is reduced, the length of the inner loop, and the stopping criterion. The length of the inner loop is limited by a time bound of 10 sec, for example. Taking this and a maximum desired running time into account, a factor for reducing the threshold at the end of each outer loop is computed. Running times in the range of 10 minutes to several hours on a modern PC are used for our benchmark scenarios.

Random Moves: First, a TRX is chosen at random. Then, an available channel is chosen for this TRX that does not cause any separation constraints. This constitutes the random change. Both choices are done according to a uniform distribution. In case no channel is available for the TRX, the choice of the TRX is repeated.

Cell Optimization: The optimization of a cell is done by Dynamic Programming. This is the topic of the following paragraphs.

The essential feature of this variant of the Threshold Accepting algorithm is the cell optimization step. Cell optimization is a counterweight to the perturbation and deterioration of random moves. The special structure of the GSM frequency assignment problem allows a complete and yet efficient optimization of one cell, provided that the assignment in all other cells is kept fixed. The reason for this is that co-cell separation is usually at least 3 and, therefore, no interference can arise among TRXs within the same cell.

The optimization of a cell is first explained under the provision that a broadcast control channel (BCCH) and a traffic channel (TCH) show no difference with respect to frequency planning. How this restriction can be removed is discussed later. Two observations can be made under this provision. Consider a feasible assignment and a cell with two or more TRXs.

(i) The channels assigned to the cell's TRXs may be redistributed among the TRXs without changing the feasibility or the cost of the assignment. (Recall that the TRXs in a cell are ordered.) Hence, it suffices to consider assignments, where the channels assigned to a cell's TRXs are in increasing order.

(ii) The current costs of the channels and the currently forbidden channels have to be computed only once, because they are the same for all TRXs in the cell.

Finding an optimal assignment for the cell can thus be reduced to the following problem. Identify an increasing list of channels such that its lengths matches the number of TRXs in the cell, such that successive channels are at least the required co-cell separation apart, and such that the list incurs minimal cost. This optimization problem is efficiently solvable using Dynamic Programming with Memoization (see Cormen *et al.* [1990, Chapter 16]), where a top-down computation strategy is used and the solutions to subproblems are stored for later look-ups.

We now come back to the assumptions made above. These assumptions are trivially met by splitting up each cell into as many "virtual cells" of TRXs with identical needs as necessary. The cell optimization is then performed merely within the virtual cells.

In Section 5.6, we give computational results for the Threshold Accepting heuristic and compare them with the ones obtained from our own methods. Our methods are analyzed in the following three sections, and we start out by looking at the construction heuristics.

5.2 Analysis of Greedy Heuristics

In this section, the behavior of the two heuristics T-Coloring and Dsatur with Costs, described in Section 4.2, is analyzed on the three carrier networks K, B[1], and Sie1. We observe that tightening the separation has a significant impact. As explained in Section 4.2.3, the Dual Greedy heuristic is not considered here, because neither the quality of its frequency assignments nor its running time efficiency have met our expectations.

5.2.1 T-Coloring

In order to study the effect of tightening the threshold on the assignments produced by T-Coloring, compare with Section 4.2.1, we allow the use of an unbounded number of channels. Recall, however, that the T-Coloring heuristic does not assign a previously unused channel unless all of those already in use are infeasible. In that case, the smallest possible new channel is taken. Thus, inspecting the assignment after termination reveals which spectrum would have been sufficient for finding a feasible assignment. The separation is tightened with thresholds values between 0.0 and 1.0 in steps of 0.01. The consumed spectrum size and the incurred total interference are shown in the Figures 5.5, 5.6, and 5.7. Roughly speaking, the interference increases and the required spectrum size decreases with increasing threshold values. A minor surprise may be the fact that the interference increases more evenly than the required spectrum size decreases.

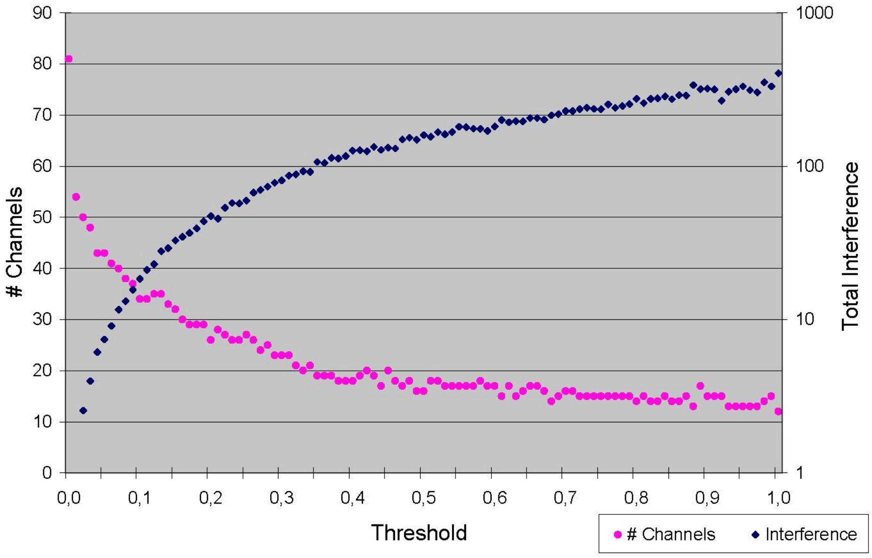


Figure 5.5: T-Coloring on instance K

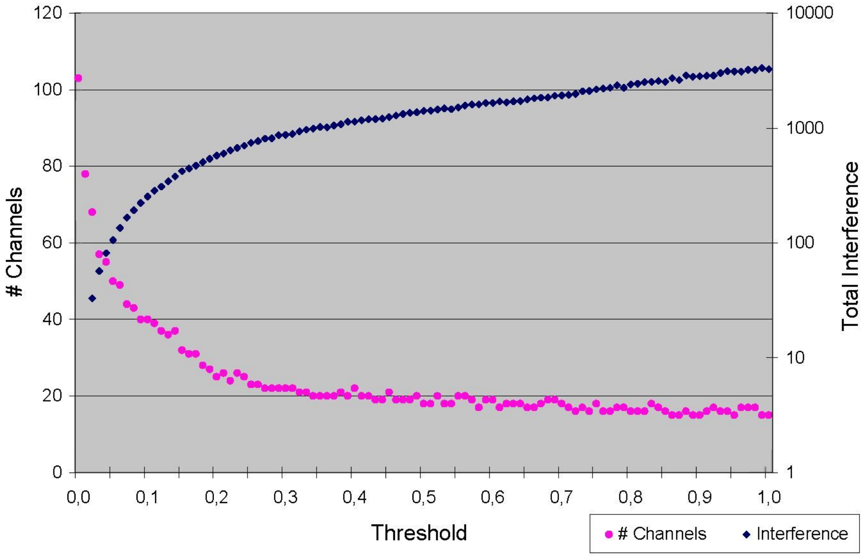


Figure 5.6: T-Coloring on instance B[1]

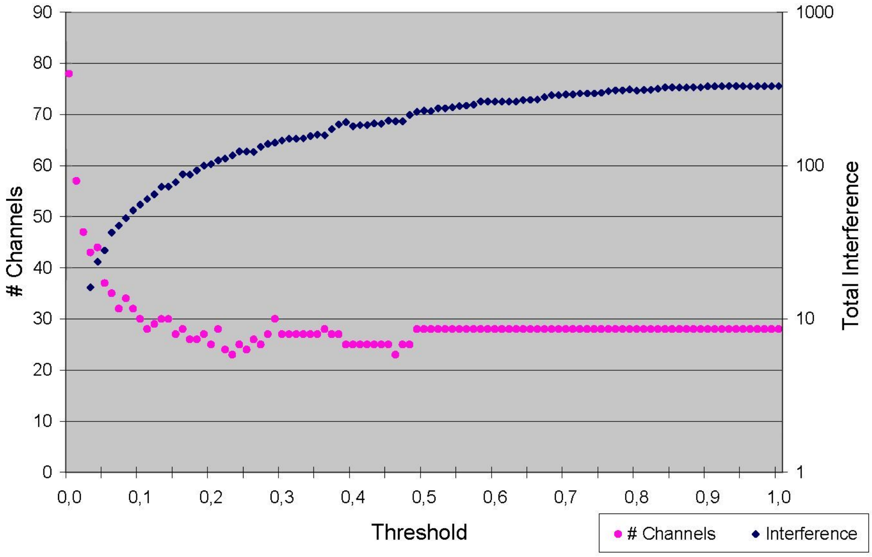


Figure 5.7: T-Coloring on instance Sie1

5.2.2 DSATUR with Costs

For the Dsatur with Costs heuristic two parameters are relevant, compare with Section 4.2.2. These parameters are the threshold value for tightening the separation and the amount of carriers selected randomly as starting points. We consider the threshold values $t = 0.01$, 0.025, 0.05, 0.075, 0.10, 0.15, 0.20, 0.25, 0.50, 0.75, and 1.00. Moreover, 1, 2, 3, 5, 10, 25, 50, and 100% of the carriers are used as starting points in turn. For all possible combinations of these two parameters frequency plans are computed for the carrier networks K, B[1], and Sie1.

The running times of this algorithm do not depend significantly on the threshold value and scale linearly with the percentage value. Table 5.5 shows the running times in seconds of Dsatur with Costs for 1% of the carriers as starting points and using a threshold value $t = 0.05$.

	K	B[1]	Sie1
time	0.10	17.85	1.04

Table 5.5: Running times in seconds of Dsatur with Costs

The total interference incurred by the resulting frequency plans are given in Figures 5.8, 5.9, and 5.10. We make three observations:

- The results depend significantly on the threshold value for tightening the separation. The best results are typically obtained for small threshold values at which already a substantial portion of the produced assignments is feasible.

- There is little dependence on the percentage of starting points. The improvements from taking more than 5% of the carriers as starting points are often negligible and do not justify the additional computational effort.

- In all cases, the threshold value yielding the best result if all carriers serve as starting points (100%) also gives the best result among the solutions obtained for 1%.

Certainly, similar observations cannot be expected to be made for all possible carrier networks. Nevertheless, based on the above observations, which are in accordance with our general computational experience with the Dsatur with Costs heuristic for practice-relevant carrier networks, we make the following assumption: the interference decreases with the value of the threshold as long as sufficiently many feasible assignments are found, and it increases again if the threshold value is lowered beyond that point. This assumption is exploited in a binary search for an appropriate threshold value. During this search, only 1% of the carriers serve as starting points. Once a good threshold is found, higher percentage values, like 5%, are used for computing an assignment.

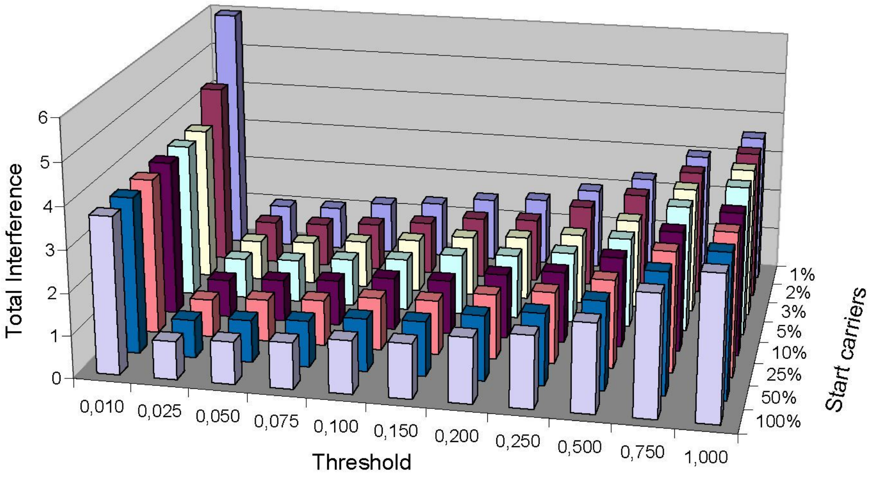


Figure 5.8: Dsatur with Costs on instance K

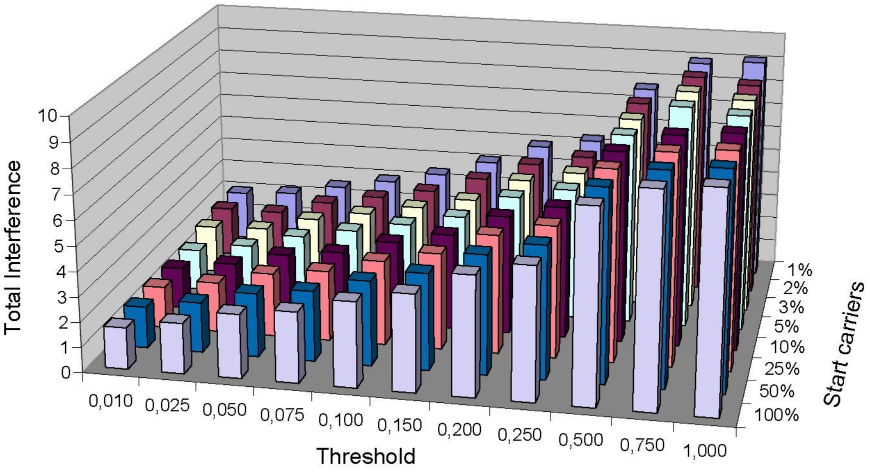


Figure 5.9: Dsatur with Costs on instance B[1]

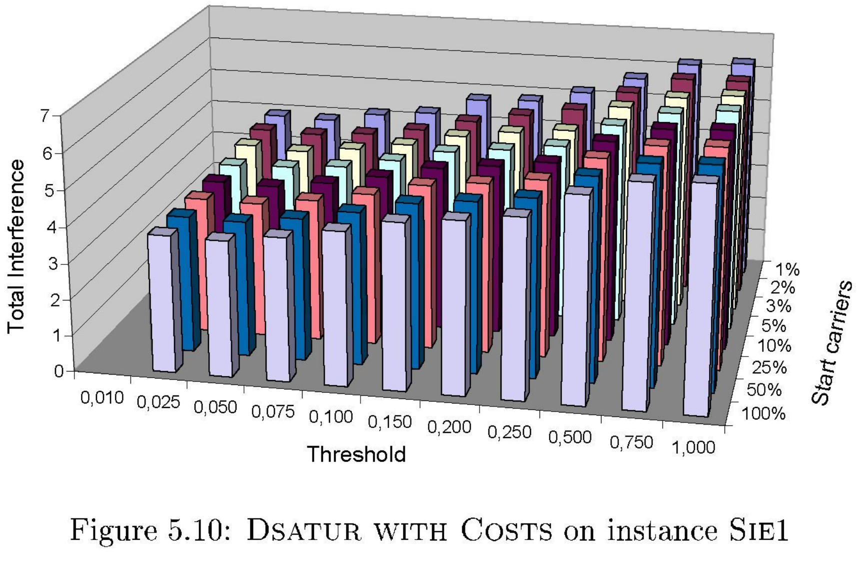


Figure 5.10: Dsatur with Costs on instance Sie1

A few details concerning the search are important. It is advantageous to randomly select the starting points once and for all in the beginning. Our search proceeds in two phases. First a coarse approximation t_1 of a good threshold value is determined. Then, the search is repeated between $t_1/2$ and $2\,t_1$ for the final threshold value t_2. The desired accuracy is 1% of t_1. Table 5.6 lists the outcomes of applying this procedure in case the initial threshold value t_0 is set to 1.0. In this particular setting, Dsatur with Costs is executed 9 times in the first as well as in the second phase.

	K	B[1]	SiE1
total interference	0.93	1.86	3.58
time [s]	17.89	322.47	46.42
final threshold	0.0270	0.0139	0.0330

Table 5.6: Dsatur with Costs including threshold search

5.3 Analysis of Improvement Heuristics

First, we investigate the performance of the two improvements heuristics Iterated 1-Opt and Vds when started from randomly generated assignments.

One hundred random assignments for each of the carrier networks K, B[1], and SiE1 are generated. None of these assignments is feasible. Table 5.7 lists the average number of separation violations and the average interference. These assignments are taken as starting points for applying both heuristics. The threshold t for tightening the separation is set to the values 0.010, 0.025, 0,050, 0.075, 0.100, 0.150, 0.200, 0.250, 0.500, 0.750, and 1.000. If no tightening is applied, we write $t = 2.000$.

For the instances K, B[1], and SiE1, the Tables 5.8, 5.9, and 5.10, respectively, show the number of feasible assignments obtained; the best,

	100 random assignments					
	separation violations			total interference		
	min.	avg.	max.	min.	avg.	max.
K	47	62.5	82	43.63	59.43	72.74
B[1]	280	320.4	366	270.14	300.73	322.12
SiE1	374	425.2	488	56.10	65.60	76.58

Table 5.7: Evaluation of 100 random assignments

the average as well as the worst total interference among the obtained feasible assignment; and the average number of separation constraint violations among the infeasible assignments. The Figures 5.11 up to 5.16 are generated from the total interference of the feasible assignments and give a qualitative impression of the best, the 25th, 50th, 75th and the 100th assignments.

t	Iterated 1-Opt					Vds				
	feasible	total interference			avg. viol.	feasible	total interference			avg. viol.
		min.	avg.	max.			min.	avg.	max.	
0.010	0				7.9	0				5.0
0.025	0				4.7	26	0.77	2.44	5.50	2.0
0.050	6	5.16	5.16	5.16	2.3	81	0.90	1.37	3.63	1.1
0.075	18	2.47	3.36	4.49	2.0	94	0.99	1.28	2.25	1.0
0.100	44	1.70	3.43	5.06	1.6	100	0.91	1.27	2.42	
0.150	77	1.74	2.63	5.18	1.0	100	0.90	1.29	1.61	
0.200	90	1.88	2.51	4.11	1.0	100	1.03	1.35	1.73	
0.250	97	1.65	2.30	3.52	1.0	100	0.98	1.30	1.63	
0.500	100	1.79	2.16	2.75		100	0.98	1.31	1.70	
0.750	100	1.69	2.22	3.06		100	1.02	1.35	1.75	
1.000	100	1.86	2.26	2.75		100	1.02	1.33	1.63	
2.000	100	1.76	2.22	2.75		100	0.99	1.37	1.77	

Table 5.8: Evaluation of 100 random assignments for instance K improved by Iterated 1-Opt and Vds.

5.3.1 Iterated 1-Opt

Let us now focus on Iterated 1-Opt and examine the Tables 5.8, 5.9 as well as the Figures 5.11, 5.12, and 5.13.

A significant spread between the best and the worst result can typically be observed for each threshold value. For small threshold values, where feasible solutions are barely obtained, the few feasible assignments are often mediocre to bad with respect to their total interference. Once the threshold value is large enough to yield feasible assignments for most of the random assignments, the results hardly depend on the actual value of the threshold.

The Table 5.11 lists the average number of passes performed and the average running times in seconds of the Iterated 1-Opt heuristic on the set of 100 random assignments. We observe that more passes are

t	Iterated 1-Opt					Vds				
	fea-	total interference			avg.	fea-	total interference			avg.
	sible	min.	avg.	max.	viol.	sible	min.	avg.	max.	viol.
0.010	0				39.8	0				17.6
0.025	0				13.6	0				4.0
0.050	0				2.9	47	3.50	4.52	5.75	1.2
0.075	50	6.01	7.01	8.77	1.5	98	3.81	4.49	5.29	1.0
0.100	93	5.95	6.66	7.39	1.0	100	3.87	4.61	5.37	
0.150	100	5.89	6.62	7.44		100	4.04	4.74	5.38	
0.200	100	5.80	6.63	7.27		100	4.21	4.60	5.38	
0.250	100	5.82	6.56	7.42		100	4.08	4.64	5.39	
0.500	100	5.91	6.60	7.25		100	3.95	4.66	5.21	
0.750	100	6.01	6.52	7.11		100	4.19	4.67	5.25	
1.000	100	5.87	6.57	7.32		100	4.16	4.63	5.57	
2.000	100	6.02	6.60	7.15		100	4.07	4.59	5.53	

Table 5.9: Evaluation of 100 random assignments for instance B[1] improved by Iterated 1-Opt and Vds.

t	Iterated 1-Opt					Vds				
	fea-	total interference			avg.	fea-	total interference			avg.
	sible	min.	avg.	max.	viol.	sible	min.	avg.	max.	viol.
0.010	0				58.4	0				36.1
0.025	0				22.9	0				9.0
0.050	0				6.7	10	4.38	4.98	6.46	2.1
0.075	11	5.82	6.61	7.25	2.1	61	4.38	4.90	5.83	1.1
0.100	54	5.72	6.28	6.93	1.1	92	4.46	4.97	5.61	1.0
0.150	93	5.38	6.17	6.62	1.1	100	4.41	5.01	5.57	
0.200	98	5.56	6.18	6.89	1.0	100	4.52	5.06	5.75	
0.250	99	5.70	6.27	6.83	1.0	100	4.58	5.02	5.49	
0.500	100	5.76	6.21	7.03		100	4.59	5.03	5.55	
0.750	100	5.38	6.14	6.80		100	4.30	4.98	5.76	
1.000	100	5.53	6.12	6.59		100	4.57	5.02	5.59	
2.000	100	5.56	6.16	6.68		100	4.61	5.04	5.89	

Table 5.10: Evaluation of 100 random assignments for instance Sie1 improved by Iterated 1-Opt and Vds.

performed as the number of carriers increases from instance K over SIE1 to B[1]. There is, however, little dependence on the threshold t.

t	Iterated 1-Opt					
	K		B[1]		SIE1	
	passes	time	passes	time	passes	time
0.010	7.22	0.69	12.43	14.86	8.46	2.18
0.025	6.60	0.65	10.15	12.88	7.07	1.92
0.050	6.39	0.64	9.62	12.41	7.07	1.77
0.075	6.27	0.64	9.27	12.15	7.21	1.78
0.100	6.14	0.63	9.13	12.27	6.93	1.75
0.150	6.32	0.64	9.25	12.39	7.34	1.80
0.200	6.24	0.63	9.26	12.29	6.89	1.76
0.250	6.35	0.64	9.07	12.25	6.96	1.72
0.500	5.92	0.61	9.21	12.35	7.18	1.76
0.750	5.83	0.62	9.01	12.19	7.05	1.74
1.000	5.76	0.61	9.41	12.31	6.83	1.74
2.000	6.12	0.61	9.15	12.30	6.92	1.81

Table 5.11: Averaged figures over the number of passes and the running times in seconds for Iterated 1-Opt on the 100 random assignments.

In conclusion, we recommend not to tighten the separation during the application of the Iterated 1-Opt heuristic.

5.3.2 Variable Depth Search

Now, we focus on the performance of the Vds heuristic on the same sets of 100 random assignments as before for each of the carrier networks K, B[1], and SIE1. The Tables 5.8, 5.9, and 5.10 are once more of interest. Moreover, the Figures 5.14, 5.15, and 5.16 are relevant.

Again, we observe a significant spread between the best and the worst result among the feasible assignments for a specific threshold value. Unlike in the case of the Iterated 1-Opt heuristic, however, in two out of the three cases the best assignment is found for the smallest threshold value for which a feasible assignment is produced at all. In the one remaining case, the result is second best. The results for K clearly advocate the tightening of the separation with a small threshold value, but the results from the other two instances are less conclusive.

Table 5.12 displays the average number of passes performed and the average running times for the Vds heuristic observed on the set of 100

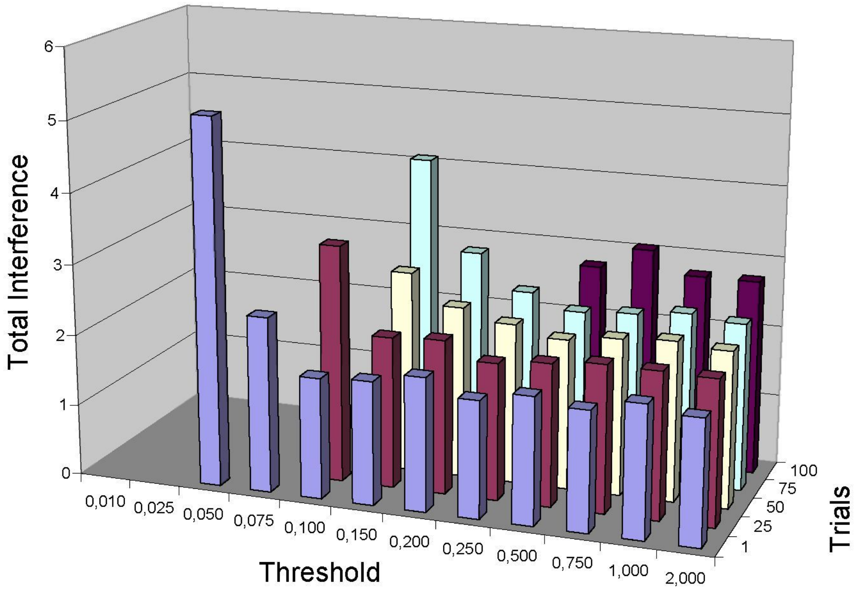


Figure 5.11: ITERATED 1-OPT on 100 random assignments for K

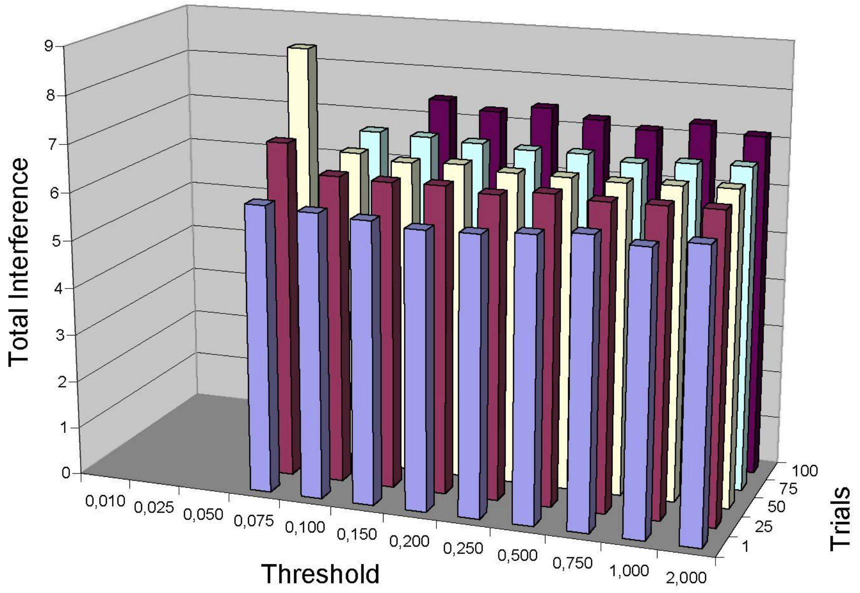


Figure 5.12: ITERATED 1-OPT on 100 random assignments for B[1]

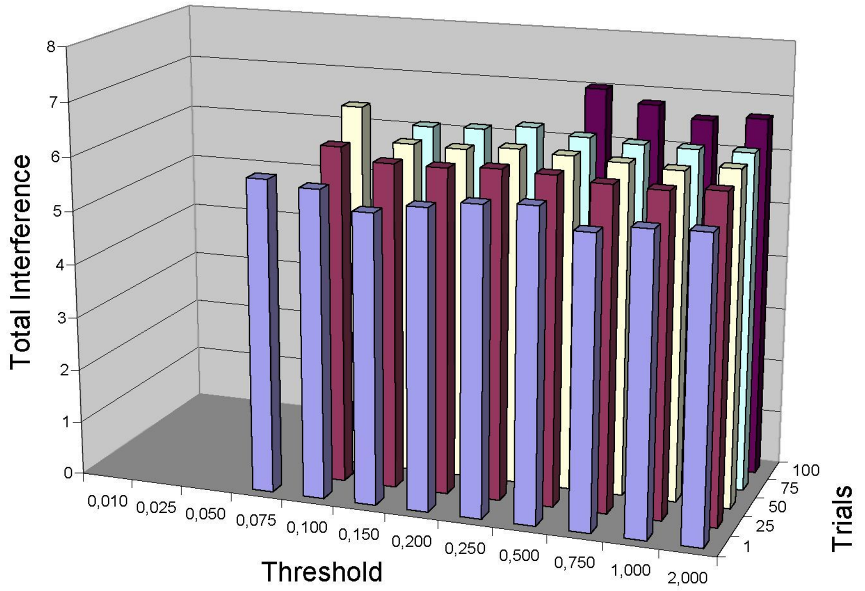


Figure 5.13: Iterated 1-Opt on 100 random assignments for Sie1

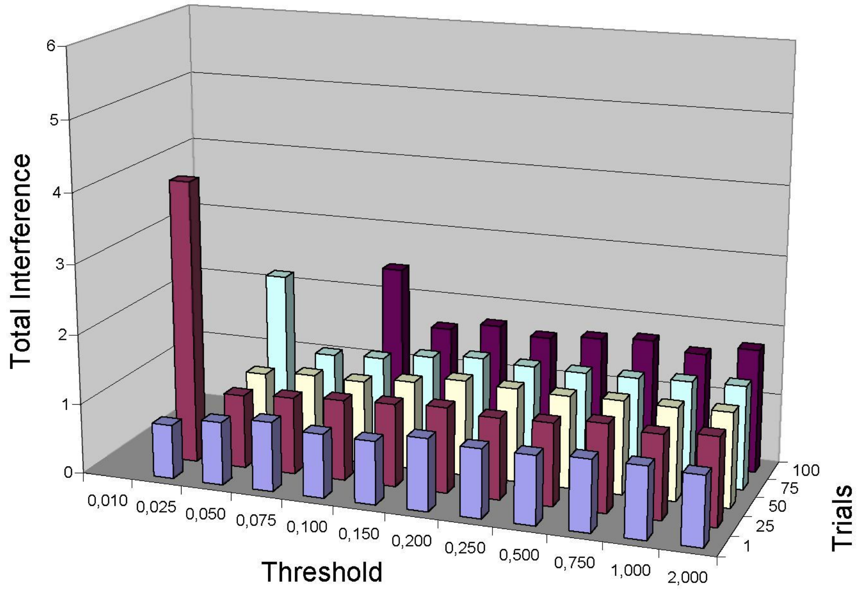


Figure 5.14: Vds on 100 random assignments for instance K

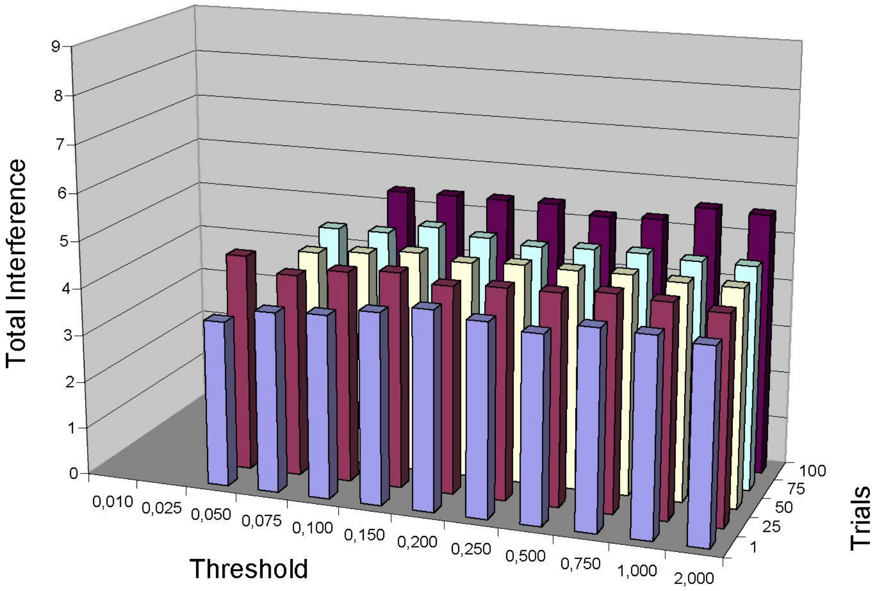


Figure 5.15: Vds on 100 random assignments for instance B[1]

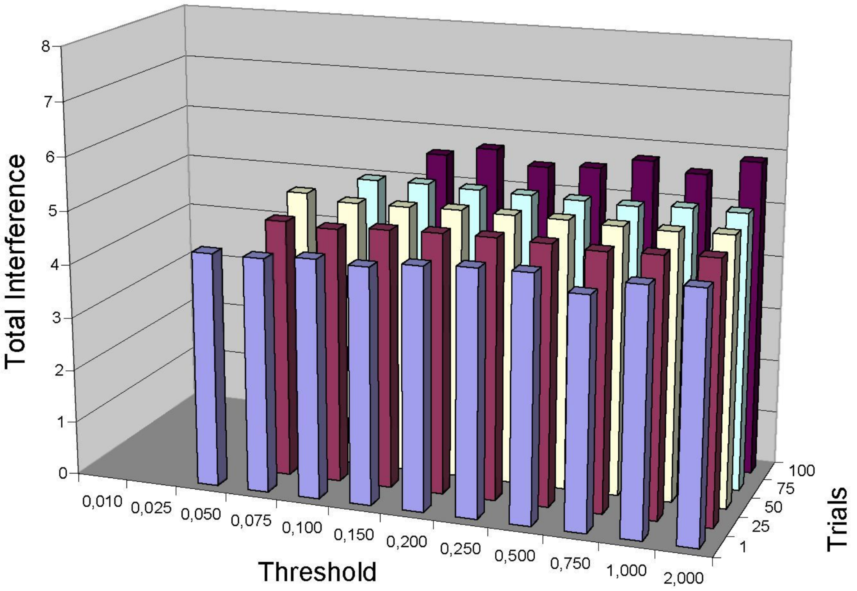


Figure 5.16: Vds on 100 random assignments for instance Sie1

t	Vds					
	K		B[1]		Sie1	
	passes	time	passes	time	passes	time
0.010	18.91	5.16	50.24	235.05	22.63	16.35
0.025	20.58	5.63	30.24	142.16	20.30	14.85
0.050	15.82	4.32	27.70	127.01	13.96	9.87
0.075	14.06	3.89	24.57	113.50	11.27	8.04
0.100	13.45	3.66	21.77	99.55	11.35	8.06
0.150	12.67	3.45	20.87	97.45	11.26	8.02
0.200	12.45	3.36	21.90	101.17	11.12	7.93
0.250	12.38	3.36	22.28	102.34	10.65	7.54
0.500	11.42	3.10	21.81	101.07	11.25	7.76
0.750	11.40	3.10	21.97	102.26	10.78	7.41
1.000	11.72	3.18	21.74	100.07	11.12	7.62
2.000	11.46	3.04	22.28	103.13	10.84	7.74

Table 5.12: Averaged figures over the number of passes and the running times in seconds for Vds on the 100 random assignments.

random assignments for the instances K, B[1], Sie1. There is a significant dependency of the number of passes on the value of the threshold t. In comparison to not tightening the separation ($t = 2.000$) the number of passes increases by a factor of 2.4, for example, for the smallest threshold ($t = 0.010$) in case of B[1].

In conclusion, we favor not to tighten the separation. In our opinion, the sometimes better alternative of choosing small threshold values has a too negative effect on the running times.

5.3.3 k-Opt

The tractability of k-opt steps strongly depends on the restrictions which are imposed by the carriers that remain unchanged on those carriers to be optimized. If k becomes too large, we are no longer able to solve the corresponding optimization problem in reasonable time. Moreover, it turns out that experiments like those for Iterated 1-Opt and Vds fail. Starting from an infeasible assignment, which most of the randomly generated assignments are, has a strong negative effect on the bounding part and blows up the branch-and-bound tree.

Therefore, we perform different experiments here to compare 1-opt and Vds-type steps on one hand, and k-opt steps on the other hand. Starting with 100 random assignments, we run Iterated 1-Opt and Vds on each of them and apply k-Opt to the best assignments obtained.

Instead of selecting carriers completely at random, we select the carriers by sites. This is reasonable because the co-site and, in particular, the co-cell separation constraints impose strong mutual restrictions on their assignment. The optimization is again organized in passes. All sites are arranged in a random order in the beginning, and in the course of a pass the first unprocessed sites are iteratively selected until either at least k many carriers or the carriers from s many sites are selected.

We consider three cases. First, one site is optimized at a time for $s = 1$ and no bound on k. In the two remaining cases, s is unbounded and as many sites as necessary are selected to obtain k carriers in total. We let k either be the number of available channels in the scenario or one and a half times that quantity. In case $s = 1$, we stop after completing one pass without improvement. In the other two cases, the optimization is halted once two passes without improvement are completed.

With the above parameter choices the optimization gets sometimes "stuck" for some selection of carriers and either runs for hours or exhausts the available memory. We avoid such "dead-ends" by restricting the permitted branch-and-bound tree to at most 8 levels. This number is determined experimentally as a compromise between completing most computations and aborting when continuing a computation seems futile.

The results are shown in Table 5.13. The interference incurred by the best plans from Iterated 1-Opt and Vds is contrasted with the result obtained from k-Opt. For the different runs of the k-Opt heuristic we list the number of passes, the average number k of carriers optimized over, the average number s of considered sites, and the number of times the growth of the branch-and-bound tree triggered termination. No running times are listed, because these computations are performed on a different computer system. Translated into running times on the PC used otherwise, they range from some minutes to one or two days.

Obviously, the local optimization performed by k-Opt is capable of improving on the results from Iterated 1-Opt and Vds in all cases except for one, namely, Vds applied to B[1]. Notice that in three cases, two for K and one for B[1], k-Opt computes a better result when started from the inferior assignment produced by Iterated 1-Opt. To some extent unexpected, however, are the merely modest improvements if we consider the large amount of carriers optimized over simultaneously. We come back to this issue in Section 5.5.

		interference		k-Opt			
		orig.	k-Opt	passes	avg. k	avg. s	aborts
K	1-Opt	1.76	1.74	3	2.90	1.00	0
			0.71	30	44.50	15.33	2
			0.69	21	66.75	23.00	7
	Vds	0.99	0.98	2	2.90	1.00	0
			0.74	27	44.50	15.33	0
			0.70	19	66.75	23.00	0
B[1]	1-Opt	5.98	5.96	4	3.04	1.00	0
			3.24	78	75.73	24.94	2
			3.09	81	109.50	36.06	3
	Vds	4.05	4.05	1	3.04	1.00	0
			3.48	48	75.75	24.94	1
			2.93	96	109.50	36.06	1
Sie1	1-Opt	5.46	5.24	8	5.20	1.00	0
			4.15	29	44.29	8.52	40
			3.70	55	64.58	12.43	169
	Vds	4.64	4.55	3	5.20	1.00	0
			3.79	53	44.29	8.52	61
			3.58	39	64.65	12.44	141

Table 5.13: k-Opt versus Iterated 1-Opt and Vds

5.3.4 Min-Cost Flow

The Mcf method, as described in Section 4.3.4, does not succeed in turning any of the infeasible 100 random assignments into a feasible assignment. This is due to limitations of the possible changes. A more interesting application of Mcf is in combination with the other improvement methods, see Section 5.4.

5.3.5 Comparisons

The k-Opt heuristic is already compared with Iterated 1-Opt and Vds in Section 5.3.3. There is nothing to add here. In order to compare Iterated 1-Opt and Vds, we reexamine the Tables 5.8, 5.9, and 5.10.

First of all, Vds is more successful in producing feasible assignments than Iterated 1-Opt. With the threshold values as chosen, we observe that Vds is essentially one value earlier capable of producing feasible assignments, which also show competitive interference values. Moreover, if the results of the two heuristics are compared per threshold value, then the worst result of VDS is in most cases better than the best result of ITERATED 1-OPT. The average of the VDS results is always better than the best of ITERATED 1-OPT, see also Figures 5.11 up to 5.16.

On average, VDS performs between 2 and 3 times as many passes as ITERATED 1-OPT, and each pass takes about three times as long. The resulting effect on the total running time for the sets of 100 assignments is not uniform. Taking averages again, VDS is about 5 to 10 times slower than ITERATED 1-OPT.

In conclusion, VDS is our method of choice in practice, because K-OPT is practically not an alternative and VDS always outperforms ITERATED 1-OPT in our experiments. In addition, we consider the increase in running time to be tolerable. Reviewing our previous discussion about which threshold value to use for VDS, we tend not to tighten the separation for its potential negative effect on the running time.

5.4 Combinations of Heuristics

In this section, we study the concerted acting of our heuristics. Numerous combinations of the greedy construction and the improvement heuristics are possible. We identify favorable combinations of these methods together with the relevant parameter settings.

Given two frequency plans of different quality, the application of (the same) improvement heuristics can possibly lead to plans for which the quality ranking is reversed. As we observe in extensive comparisons, this is seldom the case for the heuristics considered here. Therefore, we use for each heuristic the parameters setting which is favored in Section 5.3, that is, the setting we obtained from looking at each of the heuristic separate from the others.

We compare combinations of heuristics on the instances K, B[1], and SIE1 in Table 5.14. For each instance, three assignments are produced using a greedy starting heuristic:

- Plain T-COLORING is called without tightening the separation. This usually results in an assignment, where only few of the available channels are used. T-COLORING

- The acronym TS T-COLORING stands for calling T-COLORING with the separation being tightened with the least possible threshold value yielding an assignment that still fits into the available spectrum. This threshold is computed in the same fashion as for TS T-COLORING

Dsatur with Costs, see Section 5.2.2. The threshold's initial value is 0.5.

Ts Dc5

- We denote by Ts Dc5 the following application of Dsatur with Costs. The value of the threshold t for tightening the separation is determined as explained in Section 5.2.2. The threshold's initial value is 0.5. During the threshold search 1% of the carriers are randomly selected as starting points, whereas 5% are selected at the final threshold.

1-Opt

(Mcf 1-Opt)+

(Mcf Vds)+

In the subsequent improvement phase, the separation is never tightened. Each of the previously constructed assignments is alternatively improved by calling either Mcf, Iterated 1-Opt (1-Opt, for brevity) or Vds. In the latter two cases, we try to obtain further improvements by alternately calling Mcf and 1-Opt, denote by "(Mcf 1-Opt)+," or Mcf and Vds, denoted by "(Mcf Vds)+," respectively, until this fails.

	K		B[1]		SiE1	
	intf.	time	intf.	time	intf.	time
T-Coloring	558.19	0.03	4766.92	0.51	351.16	0.10
oMcf	49.21	0.26	732.89	12.56	296.88	2.14
o1-Opt	1.84	0.83	6.69	13.37	6.14	2.09
oo(Mcf1-Opt)+	1.84	1.11	5.92	60.80	5.94	7.21
oVds	1.39	3.04	4.56	125.02	5.05	8.25
oo(McfVds)+	1.39	3.61	4.55	156.48	5.04	12.62
Ts T-Coloring	2.72	0.95	20.50	15.92	15.59	2.36
oMcf	2.48	1.13	19.71	44.28	14.82	3.73
o1-Opt	1.46	1.27	4.32	24.16	5.16	3.78
oo(Mcf1-Opt)+	1.46	1.52	4.08	62.23	5.16	6.84
oVds	1.25	3.13	3.54	122.68	4.95	8.96
oo(McfVds)+	1.25	3.73	3.54	131.81	4.87	15.76
Ts Dc5	0.93	20.70	1.77	364.70	3.60	53.90
oMcf	0.93	20.85	1.77	367.59	3.59	55.00
o1-Opt	0.93	20.84	1.68	368.67	3.54	54.50
oo(Mcf1-Opt)+	0.93	21.40	1.67	388.63	3.54	55.80
oVds	0.86	23.76	1.57	412.34	3.44	59.70
oo(McfVds)+	0.86	28.01	1.56	461.06	3.44	61.90

Table 5.14: Computational results for combinations of heuristics

Table 5.14 displays the incurred total interference of the resulting frequency plans and the required running times in seconds. A "$\circ$" in front of the name of an improvement heuristic indicates that the assignment obtained from the last preceeding construction heuristic is improved. A "$\circ\circ$" indicates that the result of the directly preceeding heuristic is used as starting point. The *running times in seconds are totals in all cases.* A few observations can be made:

- The start heuristics show a clear ranking with respect to their results: Ts Dc5 is best, Ts T-Coloring second, and T-Coloring third. The order reverses when considering running times.
- The ranking of the assignments produced by different start heuristics is in no case changed when the same combination of improvement heuristics is applied. Leaving Mcf aside, this is also true without requiring the use of the same combination of heuristics.
- The Mcf heuristic has no arguable use on its own. Even the simple Iterated 1-Opt heuristics beats Mcf in all cases.
- The Mcf heuristic is of some use in combination with Iterated 1-Opt, but hardly ever in combination with Vds.
- Vds alone is the preferable improvement heuristic to be applied after a start heuristic.
- The total running times required by T-Coloring $\circ$Vds and Ts T-Coloring $\circ$Vds are about the same in all cases.
- T-Coloring or Ts T-Coloring in combination with Vds produce worse assignments than Ts Dc5 alone, but spend only about a fifth of the time required by Ts Dc5.

In summary, our combination of choice is Ts Dc$\circ$Vds. That this selection is not best in all cases, however, is documented in Table 5.15. If a feasible assignment is not easily obtained, then T-Coloring $\circ$Vds and Ts T-Coloring $\circ$Vds are attractive alternatives.

5.5 Selected Results for all Benchmark Scenarios

The total amount of interference incurred by a frequency assignment is used as cost function during the optimization. As mentioned in Section 2.3.1, this figure reveals only a small part of the picture from a practical point of view. In the following, we give a more detailed account of a frequency plan's properties than previously. Nevertheless, we stress again that the ultimate benchmark for a plan's quality is its performance in the network. Not even network simulations give a fully reliable prediction of a frequency plan's impact. Since such simulations are not an option here, we have to resort to even simpler means of analyzing a frequency plan.

interference plot

Interference plots are commonly used in network planning practice. An *interference plot* depicts the likely occurrence of interference on the basis of the signal level predictions at pixel-level. Two variants are common, both of which implicitly associate each pixel to the sector providing the strongest signal (best server model). In the one case, the difference in dB between the serving sector's signal and the second strongest signal at the same frequency is color-coded, whereas in the other case the difference between the serving sector's signal and the superposition of all interfering signals is color-coded. Figure 5.17 gives examples of the former plot, depicting interference in same region before and after optimization.

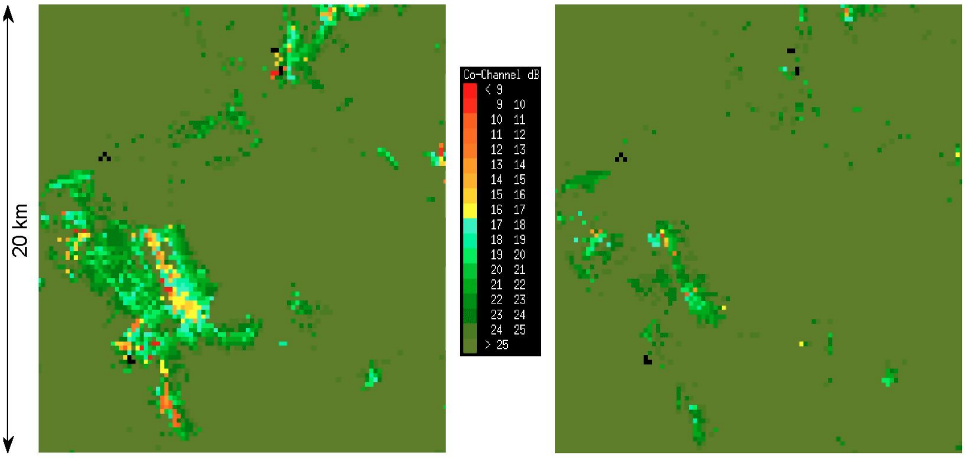


Figure 5.17: Interference plots: improvements from optimization

line plot

A cruder visualization of interference, called *line plot* here, requires only coordinates for the carriers (sites) in the Euclidean plane in addition to the carrier network. Figure 5.18 gives two examples of line plots. Two sites are connected by a colored line if the frequency assignment results in interference among TRXs from the two sites. If the line is drawn in a pale color, then the interference is small. With increasing interference, the color of the line turns into black.

A line plot still provides qualitative information on the interference and its geographical distribution. For several of the scenarios, however,

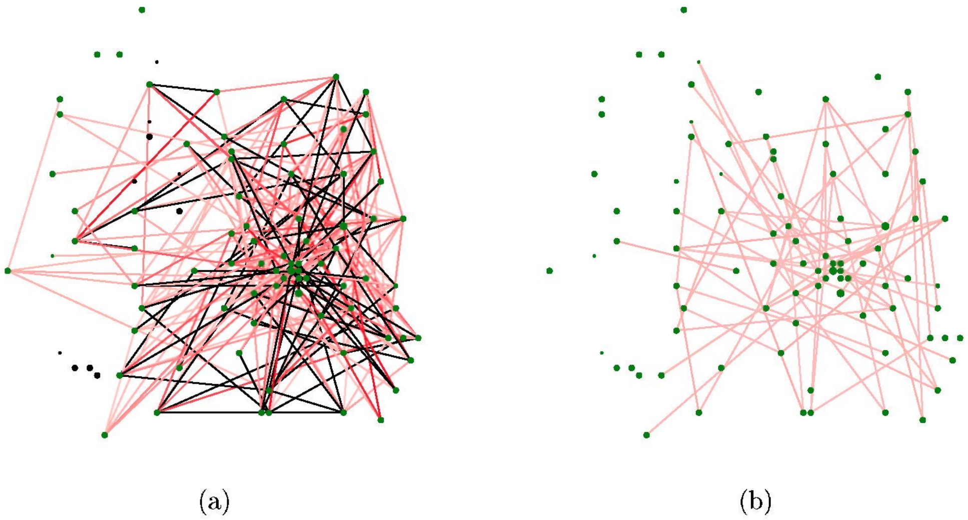


Figure 5.18: Line plots: interference reduction from optimization by 96 %

we lack the required site coordinates. Solely on the basis of the carrier network the following characteristics of a frequency assignment can be determined, compare with Eisenblätter and Kürner [2000].

- The numbers of *separation violations*, *invalid channel assignments*, and *unassigned carriers* are reported.

- The *total interference* is the sum over all co- and adjacent channel interference occurring between carriers.

- The *co- and adjacent channel interference* is given in terms of the maximum, average (among the occurrences), and standard deviation of interference of each type.

- The *interference at carriers* is given in terms of the maximum, average (among the occurrences), and standard deviation. The interference is summed up from the perspective of a carrier, regardless if it is co- and adjacent channel interference from other carriers.

- The *histogram of interference* displays how many times the interference between two carriers exceeds the value of 0.01, 0.02, 0.03, 0.04, 0.05, 0.10, 0.15, 0.20, and 0.50, respectively.

Four assignments are generated for each of the eleven benchmark instances by means of:

T-Col ∘ Vds

a. T-Coloring followed by Vds;

Ts T-Col ∘ Vds

b. T-Coloring with automatic tightening of the separation followed by Vds;

Ts Dc5 ∘ Vds

c. Dsatur with Costs with 5 % of the carriers randomly selected as starting points, automatic tightening of the separation, and followed by Vds;

Threshold Accepting

a. Threshold Accepting (the assignments are kindly provided by Hans Heller, Siemens AG, Germany).

These forty-four assignments in total are analyzed according to our above criteria, and the results are displayed in Table 5.15. The fields which would otherwise contain a zero are left blank for the sake of better legibility. We first observe a few points concerning the feasibility of the assignments.

- T-Coloring fails to generate a (feasible) assignment for Sie3 Sie4, and Sw, even without the separation being tightened. In this case, no assignment is generated by T-Coloring, and we take the situation with all carriers unassigned as starting point for Vds. Hence, the assignments obtained for T-Coloring ∘ Vds and Ts T-Coloring ∘ Vds are the same. For these three scenarios the resulting assignment is infeasible.

- Ts Dc5 ∘ Vds also fails to generate a feasible assignment for instances Sie3, Sie4, and Sw.

Next, we focus our attention to the results of Ts Dc5 ∘ Vds and Threshold Accepting on the instances, where Ts Dc5 ∘ Vds produces a feasible assignment.

- In all cases, the assignment from the combination Ts Dc5 ∘ Vds incurs more interference in total than that from Threshold Accepting. The difference ranges between 29 % and 90 %. The average difference is 60 %.

- The values for the maximum co- and adjacent channel interference and the maximum interference affecting a carrier are about equal. In the case of instance B[4], the Threshold Accepting result is even noticeably worse in that respect.

		infeasible			interference																		
						co-channel			adjacent channel			at carriers			larger than								
		s.	i.	u.	total	max.	avg.	std.	max.	avg.	std.	max.	avg.	std.	0.01	0.02	0.03	0.04	0.05	0.10	0.15	0.20	0.50
K	a				1.38	0.05	0.01	0.01	0.12	0.01	0.02	0.16	0.01	0.02	41	14	6	5	1	1			
	b				1.25	0.02		0.01	0.02	0.01	0.01	0.05	0.01	0.01	40	2							
	c				0.82	0.03		0.01	0.01			0.05	0.01	0.01	21	5							
	d				0.45	0.02						0.03			6	1							
B[0]	a				3.13	0.13	0.01	0.01	0.02	0.01	0.01	0.15	0.01	0.01	89	32	11	6	4	1			
	b				2.39	0.02			0.01			0.03	0.01	0.01	37	1							
	c				0.94	0.01			0.01			0.02			4								
	d				0.57	0.02			0.01			0.02			6								
B[1]	a				4.56	0.07	0.01	0.01	0.03	0.01	0.01	0.10	0.01	0.01	136	48	18	8	6				
	b				3.54	0.02			0.01			0.05	0.01	0.01	95	1							
	c				1.63	0.01			0.01			0.03			14								
	d				0.86	0.02			0.01			0.03			12								
B[2]	a				10.99	0.25	0.01	0.01	0.03	0.01	0.01	0.29	0.02	0.02	326	122	63	38	15	2	1	1	
	b				9.03	0.06	0.01	0.01	0.02	0.01	0.01	0.09	0.01	0.01	317	53	3	1	1				
	c				5.73	0.02			0.02			0.06	0.01	0.01	144	18							
	d				3.17	0.04			0.01			0.05	0.01	0.01	68	11	2						
B[4]	a				36.39	0.24	0.01	0.02	0.05	0.01	0.01	0.34	0.03	0.04	1048	480	260	159	98	29	6	2	
	b				35.60	0.09	0.01	0.01	0.04	0.01	0.01	0.20	0.03	0.03	1208	569	291	93	30				
	c				27.42	0.08	0.01	0.01	0.03	0.01	0.01	0.17	0.02	0.02	899	354	125	46	5				
	d				17.73	0.16	0.01	0.01	0.01			0.20	0.02	0.02	492	182	77	41	19	2	1		
B[10]	a				225.22	0.39	0.02	0.03	0.10	0.01	0.02	1.05	0.11	0.11	5211	3222	2135	1530	1148	372	137	64	
	b				225.33	0.37	0.02	0.03	0.13	0.01	0.02	0.94	0.11	0.10	5523	3420	2309	1644	1209	317	57	16	
	c				201.69	0.50	0.02	0.02	0.10	0.01	0.01	0.92	0.10	0.09	5395	3286	2101	1426	964	161	17	7	1
	d				146.20	0.34	0.01	0.02	0.08	0.01	0.01	0.70	0.07	0.08	3855	2140	1303	857	588	131	43	24	
Sie1	a				5.04	0.08	0.01	0.01	0.06	0.01	0.01	0.12	0.02	0.02	144	53	24	8	4				
	b				4.95	0.03	0.01	0.01	0.05	0.01	0.01	0.10	0.02	0.02	156	48	2	1	1				
	c				3.36	0.03	0.01		0.03	0.01	0.01	0.08	0.01	0.01	95	15	4						
	d				2.30	0.03			0.03			0.05	0.01	0.01	43	7	2						
Sie2	a				20.93	0.18	0.01	0.01	0.04			0.28	0.04	0.04	571	206	96	43	28	7	2		
	b				20.20	0.10	0.01	0.01	0.02			0.22	0.04	0.03	563	191	81	13	1				
	c				19.10	0.05	0.01	0.01	0.02			0.16	0.04	0.03	546	151	46	5					
	d				14.75	0.06	0.01	0.01	0.02			0.17	0.03	0.03	368	91	34	13	8				
Sie3	a	1			10.25	0.09	0.01	0.01	0.09	0.01	0.01	0.20	0.02	0.02	270	95	39	15	7				
	b	1			10.25	0.09	0.01	0.01	0.09	0.01	0.01	0.20	0.02	0.02	270	95	39	15	7				
	c	1			10.43	0.14	0.01	0.01	0.10	0.01	0.01	0.30	0.02	0.03	285	99	42	20	11	1			
	d				5.26	0.04			0.03			0.07	0.01	0.01	87	10	3						
Sie4	a	17			123.09	0.34	0.01	0.03	0.04	0.01		0.67	0.09	0.09	2775	1452	955	706	520	149	60	23	
	b	17			123.09	0.34	0.01	0.03	0.04	0.01		0.67	0.09	0.09	2775	1452	955	706	520	149	60	23	
	c	1			118.22	0.53	0.01	0.02	0.05			0.61	0.09	0.08	2690	1421	918	661	481	129	33	20	1
	d				80.97	0.17	0.01	0.01	0.05			0.36	0.06	0.05	2143	933	502	328	197	13	1		
Sw	a	14			8.59				0.44	0.25	0.06	0.90	0.37	0.20	35	35	35	35	35	35	35	27	
	b	14			8.59				0.44	0.25	0.06	0.90	0.37	0.20	35	35	35	35	35	35	35	27	
	c	13			5.98				0.37	0.21	0.05	0.82	0.36	0.21	29	29	29	29	29	29	29	10	
	d				26.75				0.96	0.33	0.18	3.35	0.78	0.63	80	80	80	80	80	80	80	59	12

a. T-Coloring ∘ Vds b. Ts T-Coloring ∘ Vds c. Ts Dc5 ∘ Vds d. Threshold Accepting

Table 5.15: Assignments for all benchmark scenarios

The second point can be explained as a side effect of automatically tightening the separation in Ts Dc5. It is, nevertheless, fair to say that the Threshold Accepting heuristic is the clear winner among the methods compared here in terms of the overall incurred interference and it is competitive with respect to the local distribution of interference. None of our improvement heuristics (except for k-Opt) improves the Threshold Accepting results. (We also apply the k-Opt heuristic to the Threshold Accepting assignments for the instances K, B[1], and SiE1 without any success. The parameter k is set to 1.5 times the number of available channels, the parameter s is left unbounded. We stop the optimization after two unsuccessful passes, compare with Section 5.3.3.)

We should keep in mind, however, that one of our primary design goals has been to device methods with a small overall running time. Although the precise running times of Threshold Accepting are not known to us, it takes about one order of magnitude longer than Ts Dc5 ∘ Vds (Heller [2000]). Taking this into account, Ts Dc5 ∘ Vds presents itself as a fast planning method which does not compromise too much on the optimization goal. In case Ts Dc5 ∘ Vds is still too slow, then Ts T-Coloring∘Vds or, even faster, Ts T-Coloring∘Iterated 1-Opt are alternatives. Again, some degradation of the solution has to be accepted.

We come back to the issue of how good a frequency plan is in the next chapter.

5.6 Conclusions and Challenges

A steady expansion of GSM networks can still today be observed in terms of coverage as well as capacity. Every change of the network in this respect calls for a change of the frequency assignment. Up to this point, we focused on how to use mathematical optimization techniques in order to provide frequency plans for GSM cellular networks incurring little interference between radio signals.

Planning the use of frequencies is one central tasks in engineering the radio interface of a GSM network and a cornerstone for providing the desired grade of service as well as the desired quality of service. One of the limiting factors is interference; and it has been our objective to design algorithmic methods for quickly generating frequency plans that incur as little interference as possible.

Typically, a network operator tries to keep the assignment essentially fixed for a considerable time, performing only minor changes, which are forced by modifications of the network. But every now and then, once a year, say, a new assignment is generated for (major portions of) the network. Then, all previous changes are fully taken into account and the foreseen future changes are prepared for. This is a highly complex task with literally tens or even hundreds of thousands constraints to consider. Clearly, there is a need for algorithmic optimization procedures to support the radio planner in this task. From our point of view three distinct planning tasks can be identified in practice.

- In what we call the *relaxed planning* situation, the planner wants to produce a new frequency assignment for a large region in a network and is in the fortunate position to have ample frequencies available. The objective is then to minimize interference and thus to deliver the radio service at high quality. *relaxed planning*
- In what we call the *congested planning* situation, again a new plan for large portions of a network is to be produced, but this time the number of available frequencies hardly allows to provide the desired grade of service (at the least accepted level of quality). *congested planning*
- In the *adaption planning* case, the planner is interested in adapting locally the frequency assignment to changes in the network. *adaption planning*

According to our experience each of these situations calls for different algorithmic planning methods. Our focus here is clearly on the "relaxed" planning situation. Nevertheless, we consider the K-Opt heuristic discussed in Section 4.3.3 as a prime choice for adapting locally to changes. Surveys on research directed more towards the "congested" case are given by Koster [1999], Murphey *et al.* [1999], and Jaumard *et al.* [1999], for example. The book of Nielsen and Wigard [2000] is also of interest in that respect because it deals with the use of GSM features like slow frequency hopping (SFH) in order to tune the radio interface.

In the course of the previous chapters, we gave a thorough introduction to the GSM frequency planning problem and discussed several algorithmic methods solving this problem heuristically. In more detail, we explained the technical background of frequency planning and picked an adequate mathematical model of the planning task. The resulting optimization problem is $\mathcal{NP}$-hard, and solving this optimization problem to optimality is not practicable (from today's point of view).

We therefore designed a number of planning algorithms with small theoretical and practical running times. The latter property, in particular, renders them attractive for use in an interactive planning process, where frequency plans are generated for alternative tentative changes in the network. We analyzed their computational behavior on eleven realistic planning scenarios. These scenarios have been made publicly available at the FAP web [2000] through the Cost 259 action, see Eisenblätter and Kürner [2000] or Correia [2001, Section 4.2.7].

Our findings can be summarized as follows. The self-set goal of swiftly generating low-interference frequency plans is achieved reasonably well on all scenarios except for one, where no feasible assignment is obtained. This exception, however, represents a hard "congested" planning situation. Among our methods, we identified a combination of two heuristic, namely, Dsatur with Costs and Vds, as a reasonable compromise between planning effectiveness and running time efficiency. In comparison to the best alternative method we are aware of our assignment incur at most twice the amount of interference on the studied scenarios.

Notice, however, that this alternative planning heuristic Threshold Accepting proposed by Hellebrandt and Heller [2000] is a randomized local search procedure similar to Simulated Annealing, and it often requires at least one order of magnitude higher running times than our heuristics. Despite the superiority of that method in terms of the overall incurred interference, our favorite combination produces assignments which are competitive in terms of local interference. Such local properties are of equal practical interest. Therefore, we consider our methods a reasonable choice during the interactive planning process. More elaborate planning methods, like Threshold Accepting, may be used preferably for computing the final production plan in a batch process.

We mention two more points concerning the heuristics before turning to their practical merits. First, although we applied k-opt steps for values of k as large as 1.5 times the number of available frequencies, the assignments provided by Heller, Siemens AG, were not improved in our experiments (see Section 5.5 for details). This is surprising because Threshold Accepting relies on a mix of a comparatively weak local optimization step and small random perturbations of declining deterioration. It is not obvious to us why a much more powerful local optimization fails to improve the three resulting assignments for the scenarios K, B[1], and Sie1—in particular, since we show in the next chapter that at least the assignment for K is far from optimal.

The second point concerns the question whether some sort of decomposition might be applied to a planning problem. The goal would be to solve the resulting parts separately and to combine the partial solutions to an assignment for the whole problem in the end. Our experiments in this direction failed. The discussion concerning the high connectivity observed in the planning instances (see Section 5.1.1) is certainly one item in explaining this failure, but we lack a better understanding.

After having dwelt on the mathematical optimization problem and discussing the pros and cons of trading off running time versus solution quality, it is time to turn back to practice. To cut a long story short: *our software is used successfully in practice at E-Plus, and better frequency assignments have been obtained more much quickly than through the previous planning process.* In one example, a region containing 2118 cells with 1–3 TRXs per cell and 75 available frequencies was planned using DSATUR WITH COSTS followed by ITERATED 1-OPT. The assignment was installed into the network. After performing fewer changes of cells in reaction to unforeseen heavy interference than usual (a regular online optimization process), the down-link quality hand-over request rate had dropped around 20 % in comparison to the previously operational frequency plan.

Finally, GSM networks will certainly remain in operation throughout this decade. Most European networks face major capacity extensions in order to support the foreseen increase in data services. Hence, the presently satisfying planning methods may reach their limits at some future point in time. We close by listing four directions for further research in that respect.

- Improvements in the planning methods are still possible. We give reasons for our point of view in the next chapter, where we obtain a much better assignment for the benchmark scenario K than those presented up to now. Among others, however, the unsuccessful attempts to improve the assignments provided by Heller, Siemens AG, using the local optimization K-OPT indicate that the local optimization steps considered so far are not powerful enough to deal with the complex interdependencies among large amounts of TRXs.

- A more practical point is the design and tuning of software which automatically handles all three types of the above mentioned planning situations. Such a software should also suggest when to perform a major replanning.

- Instead of aggregating all pixel-based signal predictions into interference relations at cell-level, all the available data should be fed into a more complex optimization model. This would allow for a more accurate analysis of a frequency plan during the optimization process. Moreover, the effects of GSM features like discontinuous transmission (DTX) or slow frequency hopping (SFH) could better be taken into account. (We are aware that some work in this direction is already done within companies and at the University of Cardiff, but further studies are certainly of use.)

migration path

- Among the several ways to enlarge the scope of the frequency plan optimization, we want to highlight the following. Given a network and the installed frequency plan, find an optimized assignment and a *migration path* from the given assignment to the ultimate new one. Here, we understand a migration path as a sequence of changes to be performed, one after the other, such that all the intermediate stages constitute feasible frequency plans and the changes comply with restrictions related to the available maintenance personnel, for example.

Chapter 6

Quality of Frequency Plans

In the previous two chapters we explained and analyzed several heuristic methods to generate frequency plans for GSM networks. A significant spread is observed in how well (combinations of) these methods achieve the goal of finding feasible assignments incurring minimum interference. So far, however, it remains open whether the best results are actually good or merely the relative bests among mediocre ones. In order to remedy this uncertainty we would like to prove statements like the following: this assignment, for the given planning scenario, incurs at most twice the amount of interference which is unavoidable. Such a statement is called a *quality guarantee*. Our aim in this chapter is to provide such guarantees for the assignments of Section 5.5.

quality guarantee

Let us put our quest for lower bounds and quality guarantees into perspective. We show in Section 3.2 that unless $\mathcal{P} = \mathcal{NP}$ no polynomial time algorithm is capable of computing frequency assignments that are close to optimal (or merely feasible) in all cases. As a consequence, we do not analyze algorithms here. We analyze the algorithms' input, namely, the carrier networks. We want to prove that a certain amount of interference is unavoidable due to the network layout. Hence, the bounds have to be computed on a per-instance basis. For reasons to become clear, our bounds take co-channel but no adjacent channel interference into account.

We provide the first significant lower bounds for the objective of minimizing overall interference on realistic frequency planning scenarios. Large gaps between the amount of interference incurred by the heuristically generated frequency plans and the lower bounds still exist. We give reasons that these gaps are not to be blamed entirely on the weakness of the lower bounds. Instead, the gaps also indicate room for improvement on the side of the frequency planning methods.

Up to now, our bounds have no direct practical impact. The situation may improve if information from the lower bound computation were to be exploited in the frequency planning process. With the approach pursued here, however, it is not yet clear how this can be done effectively.

The chapter is organized as follows. In Section 6.1, we introduce the "relaxed frequency planning problem." Optimal solutions to this problem yield lower bounds on the unavoidable interference in the original problem. In Section 6.2, the tight connection between the relaxed planning problem and the well-known MINIMUM K-PARTITION problem is shown. Moreover, two possible approaches for (approximately) solving a MINIMUM K-PARTITION problem are described. In Section 6.3, lower bounds are provided by means of one of the approaches, namely, by solving large semidefinite programs. We use these bounds to estimate the quality of the frequency assignments from Section 5.5. In Section 6.4, we revisit frequency planning heuristics. Our goal is to turn feasible frequency plans for the relaxed frequency planning problem into feasible plans for the ordinary planning problem. In some cases it is possible to produce significantly better frequency plans than before. This supports our opinion that, from the point of view of mathematical optimization, the frequency planning problem is not yet fully mastered; this also sheds a new light on the lower bounds, which now appear stronger than before.

Including this chapter, our focus is primarily on frequency assignment. We change our point of view for the last two chapters, where we deal with the mathematics behind our computations of unavoidable interference, that is, the MINIMUM K-PARTITION and its semidefinite relaxation.

6.1 Relaxed Frequency Planning

We consider a "relaxed" version of our frequency planning problem, where the feasibility constraints are weakened and the objective function is simplified. Although the complexity status of both problems is formally the same, the resulting problem is more accessible than the original frequency assignment problem. The simplifications are the following.

Dropping adjacent channel interference: The major portion of the total interference incurred by a frequency plan is often co-channel interference, see Table 5.15 in Chapter 5. Good lower bounds for our scenarios are, thus, hardly obtained without fully exploiting co-channel interference, whereas adjacent channel interference often plays a minor role. (An exception is the scenario SW, where only adjacent channel interference is specified. Our approach does not allow to derive a nontrivial lower bound for the scenario SW, but as explained in Section 5.1.1, this scenario is atypical.)

Dropping local blockings: Only few local blockings are present in our scenarios. Thus, assuming $B_v = \emptyset$ for all $v \in V$ leads to minor changes in the scenarios. We pointed out, however, that in a real-life planning numerous local blockings may be present. One source of blockings are the agreements between GSM network operators on the use of available spectrum in border regions, e. g., along national borders. Another source is the necessity to seamlessly integrate the frequency plan for a rearranged portion of the network with the assignment for the remaining part of the network. (The frequency plan for an entire network is hardly ever changed at once.)

Cutting down required separation: A certainly debatable step is to relax the separation requirements by bounding the maximal required separation in d by 1. But this is essential for arriving at a "simpler" problem.

All three simplifications together yield a problem, which is almost a MINIMUM EDGE DELETION K-PARTITION problem, see Sections 3.2. Computational methods may now be applied that are not suited for solving the original frequency planning problem, see Sections 6.2 and 6.3.

Formally, we write the relaxed problem as an ordinary frequency planning problem on a simplified carrier graph. Given a carrier network $N = (V, E, C, \{B_v\}_{v \in V}, d, c^{co}, c^{ad})$, the associated *simplified carrier network* is defined as the 5-tuple $N_s = (V, E, C, \underline{d}, c^{co})$, where $\underline{d}\colon E \to \{0, 1\}$ with $vw \mapsto \min\{d(vw), 1\}$. The sets B_v of locally blocked channels as well as the adjacent channel interference c^{ad} disappear, and the required minimum separations in d are cut off at 1.

simplified carrier network

As before, a *frequency assignment* or just an *assignment* for N_s is a function $y\colon V \to C$ that assigns a channel to every carrier. An assignment is *feasible* if $|y(v) - y(w)| \geq \underline{d}(vw)$ for all $vw \in E$.

assignment

feasible

Definition 6.1. *An instance of the* relaxed frequency planning problem *consists of a simplified carrier network N_s, and the objective is to solve the optimization problem*

relaxed FAP

$$\min_{y \text{ feasible}} \sum_{\substack{vw \in E\colon \\ y(v) = y(w)}} c^{co}(vw). \tag{RFAP}$$

Solving the relaxed frequency assignment problem clearly yields a lower bound on the optimal solution for the ordinary assignment problem, but not vice versa. Finding an optimal solution for the relaxed frequency planning problem is not known to be simpler than for the original problem. Both problems are $\mathcal{NP}$-hard. (The considerations of Section 3.2 for the ordinary frequency planning problem transfer directly to the relaxed problem.) The following observation, however, points out one major difference between the original problem and its relaxed version.

Observation 6.2. *For a simplified carrier network and an associated frequency assignment y the following holds. Let f_1 and f_2 be two channels from the spectrum, and let the assignment $\tilde{y}$ be obtained from y by changing all carriers with channel f_1 to f_2 and all carriers with channel f_2 to f_1. Then, $\tilde{y}$ is feasible if and only if y is feasible. Furthermore, $\tilde{y}$ and y incur the same amount of interference.*

Consequently, all assignments differing only in the permutation of channels may be considered equivalent. If all channels are in use, then there are $|C|!$ many assignments which are equivalent in that sense.

6.2 Minimum k-Partition

Relaxed frequency planning is done on a simplified carrier network $N_s = (V, E, C, \underline{d}, c^{co})$, where no adjacent channel interference or locally blocked channels are specified and minimum separation is at most 1. Basically, a partition $V_1, \ldots, V_p$ of V into at most $k = |C|$ disjoint sets of carriers (using distinct frequencies) has to be determined such that no two vertices v, w with $\underline{d}(vw) = 1$ are in the same set and such that the sum over the edge weights in the induced subgraphs $G[V_l]$, $1 \leq l \leq k$, is minimized. This problem is almost the same as finding a minimum k-partition of the vertex set of the graph $G = (V, E)$, which underlies the simplified carrier network. The edge weights for G are derived from $\underline{d}$ and c^{co}.

Let $E_{co} = \{vw \in E : c^{co}_{vw} > 0\}$ and $E_d = \{vw \in E : d_{vw} > 0\}$. We may assume that the intersection of E_{co} and E_d is empty and that their union is E. Furthermore, we set $M = \sum_{vw \in E_{co}} c^{co}_{vw} + 1$. The edge weight function $c\colon E \to \mathbb{R}_+$ is defined as $c(vw) = c^{co}_{vw}$ in case of $vw \in E_{co}$ and $c(vw) = M$ in case of $vw \in E_d$.

Any solution to this MINIMUM K-PARTITION problem induces a frequency assignment for N_s. The same channel is assigned to all vertices in one block, and distinct channels are given to distinct blocks. If an optimal solution to the k-partition problem has a value less than M, then the induced assignment is feasible for N_s. The solution value is the interference incurred by a corresponding assignment—no matter how we choose to distribute the frequencies among the blocks. If, on the other hand, the optimal solution value is at least M, then there is no feasible assignment for N_s.

k-partite graph

Another way of looking at the problem is to find a weight-minimum set of edges whose removal results in a k-partite graph. A graph is *k-partite*, if its vertex set can be partitioned into at most k independent sets, i. e., all edges have their endpoints in distinct sets. A graph is obviously k-colorable if and only if it is k-partite. A graph without edges is k-partite, and a graph containing a clique of size $k+1$ is not. A natural question to ask is: what is the maximum number of edges that a k-partite graph on n vertices may have? The well-known answer is $\binom{k}{2}\left\lfloor\frac{n}{k}\right\rfloor^2+r(k-1)\left\lfloor\frac{n}{k}\right\rfloor+\binom{r}{2}$, where r denotes the remainder of the integer division of n/k. Hence, k-partite graphs can be fairly dense. The complete graph K_{267}, for example, has $\binom{267}{2} = 35511$ edges, a 50-partite graph on 267 vertices can have as many as 34926 edges, which is only $r\binom{\lceil n/k\rceil}{2} + (k-r)\binom{\lfloor n/k\rfloor}{2} = 585$ less than in the complete graph.

Formally, the minimum graph k-partition problem or Minimum k-Partition problem can be stated as follows:

Minimum k-Partition

Definition 6.3. *An instance of the* Minimum k-Partition problem *consists of an undirected graph $G = (V, E)$, a weighting $c\colon E \to \mathbb{Q}$ of the edges, and a positive integer k. The objective is to find a partition of V into at most k disjoint sets $V_1, \ldots, V_p$ such that the value*

$$\sum_{l=1}^{p} \sum_{vw\in E(G[V_l])} c(vw)$$

is minimized.

The Minimum k-Partition problem is a generalization of the Minimum Edge Deletion k-Partition problem, see Definition 3.9, to general rational edge weights. Therefore, the computational complexity of the former is at least as hard as that of the latter.

The Minimum k-Partition problem is explicitly studied by Chopra and Rao [1993], but more is known from related problems, e. g., by means of the following equivalence: for every partition, an edge has either its both endpoints within the same block or within distinct blocks; hence, the problem of finding a minimum k-partition is equivalent to finding a k-cut, where the sum over the weight of all edges with their endpoints in distinct sets is maximized. The Maximum k-Cut problem has received more attention in the literature than the Minimum k-Partition problem, see, e. g., Deza, Grötschel, and Laurent [1991, 1992] and Chopra and Rao [1995]. Results on the approximation of the MAXIMUM K-CUT problem are obtained by Karger, Motwani, and Sudan [1994] as well as by Frieze and Jerrum [1997]. Those results are, however, of little help here. The optimal cut value is underestimated so that the value of the MINIMUM K-PARTITION is overestimated, and no lower bound is supplied that way. Assuming that all edge weights are nonnegative, Goldschmidt and Hochbaum [1994] show how a *maximum* partition of the graph into k nonempty components can be computed in $\mathcal{O}(n^{k^2/2-3k/2+4}\ T(n,m))$ time, where $T(n,m)$ is the time required to compute a minimum (s,t)-cut. Due to the sign-constraint, their result does not apply here either.

MAXIMUM CUT

Notice also the connection to the MAXIMUM CUT problem: an edge-weighted graph is given, and the task is to find a partition of the vertex set into two sets (one possibly empty) such that the sum over the weights of all edges with their endpoints in different sets is maximized. The literature on the MAXIMUM CUT problem is extensive, see the survey article by Poljak and Tuza [1995], the book by Deza and Laurent [1997], and the references contained therein. The MAXIMUM CUT problem is equivalent to the MINIMUM 2-PARTITION problem.

6.2.1 Interference is not essentially metric

MINIMUM K-CLUSTERING SUM

The MINIMUM K-PARTITION problem gets simpler if the edge weights are nonnegative and fulfill the triangle inequalities, i. e., $c_{vx} + c_{xw} \geq c_{vw}$ for all triangles in the graph. The problem is then also called MINIMUM K-CLUSTERING SUM, see Ausiello *et al.* [1999, Appendix B, ND 55], and is proven to be approximable within a factor of two in polynomial time by Buttmann-Beck and Hassin [1998]. We show, however, that the triangle inequalities are far from being fulfilled by our data sets.

A systematic reason is due to the mapping of separation constraints to large weights. Consider the placement of transmitters and their pairwise interference as depicted in Figure 6.1. The rectangle in the center represents an obstacle. Assume the availability of only two adjacent channels and that the channels for transmitters a and b need to be at least one apart. If all triangle inequalities are to be met, then the triangle between a, b, and c bounds the value of M to at most 0.2. Setting M to 0.2 or less, however, implies that the minimum-weight solution assigns a, b, and d the same channel. The resulting assignment is infeasible. Hence, separation conditions can, in general, not be represented adequately by an appropriately chosen weight without violating the triangle inequality. This first obstacle stems solely from representing separation constraints by weights.

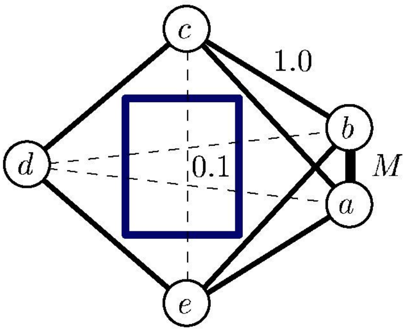


Figure 6.1: Turning separation to interference clashes with $\triangle$ inequalities

A second obstacle is based on interference entries alone. Interference predictions are derived by means of signal propagation predictions for an outdoor environment so that the edge weights are not arbitrary. Maybe this implies that the triangle inequalities are (almost) fulfilled. This is not the case, as we show now.

We drop all edges carrying a separation constraint from the graph and check whether the triangle inequalities are met among the remaining edges. To this end, we solve the following linear program derived from the graph $G = (V, E^{co})$ with the edge labeling c:

$$
\begin{aligned}
&\min \sum_{vw \in E^{co}} r_{vw} \qquad \text{s.t.} \\
&(c_{vx} - r_{vx}) + (c_{xw} - r_{xw}) - (c_{vw} - r_{vw}) \geq 0 \quad \forall\, vx, xw, vw \in E^{co} \qquad (6.1)\\
&0 \leq r_{vw} \leq c_{vw} \quad \forall\, vw \in E^{co}
\end{aligned}
$$

The optimal value to this linear program gives the total sum of how much the weight of individual edges have to be decreased in order to obtain weights that meet all triangle inequalities. Clearly, if the optimal value is zero, then all triangle inequalities are met without change. Notice that the alternative of increasing the edge weights is not available to us, because we want to use the results as lower bounds.

Table 6.1 shows how severely the triangle inequalities are violated by the interference predictions for the realistic scenarios K, B[1], and SiE1 from Chapter 5. We generate (6.1) on the basis of the entire carrier network (*all*), the carrier network induced by all vertices in a clique larger than the set of available channels (*union*), and the carrier network induced by a maximum clique (*clique*). In order to obtain LPs of reasonable sizes, the separation is tightened with 0.05, see Section 4.1.2, and all edges with co-channel interference less than 0.001 are dropped. This reduces the number of potentially violated triangle inequalities in the carrier networks. The first two columns of Table 6.1 indicate the instance and the type of the (sub-)network. Then, the number of violated triangle inequalities, the maximum violation, and the average violation are shown. Next, the number of variables and constraints in the generated LPs are reported. Finally, the optimal value of the LP is given. The optimal solution could not be computed for B[1]/all due to a lack of computer memory (Cplex requests more than 4 GB of memory).

		# viol.	max.	avg.	LP(6.1)		
		$\triangle$	violation		# vars	# cons	opt.
K	all	32290	0.0476	0.0132	7477	137394	89.02
	union	30179	0.0476	0.0132	6744	129000	82.04
	clique	649	0.0447	0.0133	500	3033	5.34
B[1]	all	1343380	0.0480	0.0131	143758	5826507	—
	union	54995	0.0475	0.0131	9825	254466	138.65
	clique	811	0.0439	0.0119	764	5112	6.18
Sie1	all	81683	0.0466	0.0140	21709	323649	202.72
	union	28686	0.0466	0.0140	8278	118833	88.86
	clique	540	0.0406	0.0111	475	3153	3.07

Table 6.1: Violation of the $\triangle$ inequality by interference predictions

The reductions of the edge weights that are necessary to fulfill all triangle inequalities are, in fact, orders of magnitude larger than the total interference incurred by the feasible frequency plans analyzed in Table 5.15. Hence, is seems futile to consider the (polynomial time) 2-approximation algorithm for the Minimum k-Clustering Sum problem by Buttmann-Beck and Hassin [1998] as a reasonable option for computing strong lower bounds on the unavoidable interference. We therefore turn back to the general Minimum k-Partition problem.

6.2.2 An ILP formulation and a SDP relaxation

Two formulations of the Minimum k-Partition problem on a complete graph K_n with $n \geq k \geq 2$ are given next. (The graph $G = (V, E)$ with edge weights c is completed to $K_{|V|}$, and the edge weighting is extended to all new edges by assigning a weight of zero.) These two formulations have relaxations of quite different kind.

The first formulation is a plain integer linear program (ILP), see (6.2), with an LP relaxation obtained by dropping all integrality constraints. A binary variable z_{vw} is associated to every edge vw of the graph. The value of z_{vw} equals 1 if and only if both endpoints are in the same partite set. The constraints (6.2a) require the setting of the variables to be consistent, that is, transitive. For example, if z_{vx} and z_{xw} indicate that v, x, and w are in the same partite set (by transitivity), then the setting of z_{vw} has to reflect that as well. The constraints (6.2b) impose that at least two from a set of $k+1$ vertices have to be in the same partite set. Together with the constraints (6.2a) this implies that there are at most k partite sets. We deal with the ILP (6.2) extensively in the next chapter.

$$
\begin{aligned}
&\min \sum_{v,w \in V} c_{vw}\, z_{vw} && && \\
&\text{s.t.} && && \\
&z_{vx} + z_{xw} - z_{vw} && \leq 1 && \forall\, v, x, w \in V && (6.2a)\\
&\sum_{v,w \in Q} z_{vw} && \geq 1 && \forall\, Q \subseteq V \text{ with } |Q| = k+1 && (6.2b)\\
&z_{vw} && \in [0,1] && && \\
&z_{vw} && \text{integer} && &&
\end{aligned}
$$

The second formulation can be seen as a semidefinite program with "integrality" constraints. This formulation builds on the following two facts: there are k unit vectors in $\mathbb{R}^n$ with mutual scalar products of $\frac{-1}{k-1}$, and this value of the scalar products is least possible. Consider, for example, a simplex with k vertices in $\mathbb{R}^n$, centered at the origin and scaled such that all vertices have a Euclidean distance of 1 to the origin. The vectors pointing at the vertices of this simplex have the desired property. Formally, the following can be proved.

Lemma 6.4. *For all integers n and k satisfying $2 \leq k \leq n+1$ the following holds:*

(i) There exist k unit vectors $\bar{u}_1, \ldots, \bar{u}_k \in \mathbb{R}^n$ such that $\langle \bar{u}_i, \bar{u}_j \rangle = \frac{-1}{k-1}$ for all $i \neq j$.

(ii) Any k unit vectors $u_1, \ldots, u_k \in \mathbb{R}^n$ satisfy:

- $\sum_{i<j} \langle u_i, u_j \rangle \geq -\frac{k}{2}$;
- *if $\langle u_i, u_j \rangle \leq \delta$ for all $i \neq j$, then $\delta \geq \frac{-1}{k-1}$.*

Hence, the scalar products among the unit vectors $\bar{u}_1, \ldots, \bar{u}_k \in \mathbb{R}^n$ are indeed least possible. Although we believe that Lemma 6.4 is folklore, we only know references for the first and the last claim, see, e. g., Karger, Motwani, and Sudan [1998] or Frieze and Jerrum [1997]. We give a complete proof of Lemma 6.4 here for the sake of completeness.

Proof. Ad (i): Considering the case of $k = n + 1$ suffices to prove the existence of k unit vectors $\bar{u}_1, \ldots, \bar{u}_k \in \mathbb{R}^n$ such that $\langle \bar{u}_i, \bar{u}_j \rangle = \frac{-1}{k-1}$ for $i \neq j$. Since the one-dimensional case is trivial, we focus on the cases $n \geq 2$. Let t_i, $1 \leq i \leq n+1$, be the $(n+1)$-dimensional vector with all entries equal to $-\sqrt{\frac{1}{n(n+1)}}$ except for the ith one, which is $\sqrt{\frac{n}{n+1}}$. Every t_i is a unit vector in $\mathbb{R}^{n+1}$, and $\langle t_i, t_j \rangle = \frac{n-1}{n(n+1)} - \frac{2}{n+1} = \frac{-1}{n}$ for $i \neq j$.

If the vectors t_i were in $\mathbb{R}^n$ instead of $\mathbb{R}^{n+1}$, then the claim would be proved; indeed, the subspace spanned by those vectors is at most n-dimensional, because $\langle t_i, \begin{bmatrix} 1 & \ldots & 1 \end{bmatrix}^T \rangle = -n\sqrt{\frac{1}{n(n+1)}} + \sqrt{\frac{n}{n+1}} = 0$ for all i. Hence, we may rotate the coordinate system in such a way that the vector $\begin{bmatrix} 1 & \ldots & 1 \end{bmatrix}^T$ turns into a multiple of the vector $\begin{bmatrix} 0 & \ldots & 0 & 1 \end{bmatrix}^T$. Then the last coordinate of each vector t_i is zero according to the new coordinate system, and we obtain the vector $\bar{u}_i$ from t_i by truncation. Clearly, each $\bar{u}_i$ is a unit vector in $\mathbb{R}^n$, and we have $\langle \bar{u}_i, \bar{u}_j \rangle = -\frac{1}{n}$ for all $i \neq j$.

Ad (ii): For proving the first claim, we fix n and use induction on k. For $k = 2$, the scalar product of any unit vector $u \in \mathbb{R}^n$ and its negative $-u$ is $-1 = \frac{-1}{2-1}$; this is least possible. We now consider $k \geq 3$. By induction hypothesis,

$$\sum_{\substack{i<j \\ i,j \neq l}} \langle u_i, u_j \rangle \geq -\frac{k-1}{2} \qquad \forall\, l = 1, \ldots, k.$$

Summing up all these inequalities yields

$$-k\,\frac{k-1}{2} \leq \sum_{l=1}^{k} \sum_{\substack{i<j \\ i,j \neq l}} \langle u_i, u_j \rangle = (k-1) \sum_{i<j} \langle u_i, u_j \rangle,$$

and the claim follows.

Finally, we turn to the second claim of (ii). From the fact that the norm of a vector is nonnegative and the assumption, we conclude $0 \leq \langle u_1 + \cdots + u_k, u_1 + \cdots + u_k \rangle = \sum_i \langle u_i, u_i \rangle + \sum_{i \neq j} \langle u_i, u_j \rangle \leq k + k\,(k-1)\,\delta$. This implies $\delta \geq \frac{-1}{k-1}$, as desired. □

According to Lemma 6.4 we may fix a set $U = \{u_1, \ldots, u_k\} \subseteq \mathbb{R}^n$ of unit vectors with $\langle u_i, u_j \rangle = \frac{-1}{k-1}$ for $i \neq j$ (and $\langle u_i, u_i \rangle = 1$ for $i = 1, \ldots, k$). These k vectors are used as labels (or representations) for the k partite sets. The Minimum k-Partition problem can then be phrased as follows. We search for an assignment $\phi \colon V \mapsto U$ that minimizes the expression

$$\sum_{v,w \in V(K_n)} c_{vw} \frac{(k-1)\,\langle \phi(v), \phi(w) \rangle + 1}{k}. \tag{6.3}$$

Notice that the quotient in the summands evaluates to either 1 or 0, depending on whether the same vector or distinct vectors are assigned to the respective two vertices.

If we assemble the scalar products $\langle \phi(v), \phi(w) \rangle$ into a square matrix X, being indexed row- and column-wise by V, then the matrix X has the following properties: all entries on the principal diagonal are ones, all off-diagonal elements are either $\frac{-1}{k-1}$ or 1, and X is positive semidefinite.

Notably, every matrix X satisfying the above properties defines a k-partition of V in the same way as ϕ does. This can be seen as follows. Since X is a positive semidefinite matrix, there exists a matrix C such that $X = C^T C$. We claim that C contains at most k distinct column vectors. For the sake of a contradiction, let us assume that $c_1, \ldots, c_{k+1}$ are $k+1$ distinct column vectors from C. Then $\langle c_i, c_j \rangle = \frac{-1}{k-1}$ for all $i \neq j$ and $\langle c_i, c_i \rangle = 1$ for all i. According to Lemma 6.4, the $k+1$ unit vectors may only have a mutual scalar product as low as $\frac{-1}{(k+1)-1} = \frac{-1}{k} > \frac{-1}{k-1}$, a contradiction. Therefore, the columns of C may indeed serve as the vectors assigned by ϕ, representing the partite sets.

The combinatorial problem to minimize (6.3) may be relaxed to a semidefinite program (SDP). First, the explicit reference to the set U is dropped, and the problem is rewritten as follows:

$$\begin{aligned}
&\min \sum_{vw \in E(K_n)} c_{vw} \frac{(k-1)\,X_{vw} + 1}{k} \\
&\text{s.t.} \\
&X_{vv} = 1 && \forall\, v \in V && \text{(6.4a)} \\
&X_{vw} \in \{\frac{-1}{k-1}, 1\} && \forall\, v, w \in V && \text{(6.4b)} \\
&X \succeq 0
\end{aligned}$$

Then, we replace the constraints (6.4b) by $X_{vw} \geq \frac{-1}{k-1}$. Notice that $X_{vw} \leq 1$ is enforced implicitly by X being positive semidefinite and

$X_{vv} = 1$. The SDP relaxation of the problem is the following:

$$\begin{aligned}
&\min \sum_{vw \in E(K_n)} c_{vw} \frac{(k-1)\,X_{vw}+1}{k} && \\
&\text{s. t.} && \\
&X_{vv} = 1 && \forall\, v \in V && \text{(6.5a)}\\
&X_{vw} \geq \frac{-1}{k-1} && \forall\, v, w \in V && \text{(6.5b)}\\
&X \succeq 0 &&
\end{aligned}$$

Lovász [1979] introduces this type of relaxation of a combinatorial optimization problem to compute the Shannon capacity of a graph, and Goemans and Williamson [1995] use it in an approximation algorithm for the Maximum Cut problem. Although published some years apart, Karger *et al.* [1994, 1998] as well as Frieze and Jerrum [1997] used around the same time and independently the semidefinite relaxation (6.5) in combination with randomized rounding to obtain a polynomial time approximation algorithm for the Maximum k-Cut problem.

An introduction to semidefinite programming and an analysis of the semidefinite program (6.5) are provided in Chapter 8. Our interest is here merely in the fact that the semidefinite relaxation of the Minimum k-Partition problem associated to our test instances are (approximately) solvable on today's PCs.

6.3 Numerical Bounds and Quality Assessments

In this section, we provide nontrivial lower bounds on the amount of unavoidable co-channel interference. We consider all our planning scenarios with the natural exception of Sw, because no co-channel interference is specified for Sw. The lower bounds are obtained through solving the semidefinite relaxation (6.5) associated to the simplified carrier graphs. In fact, we solve a slightly modified semidefinite program, where the constraint $X_{vw} \geq \frac{-1}{k-1}$ is enforced at equality for all edges with $\underline{d}(vw) = 1$ in the simplified carrier graph. The previously suggested construction, using very high weights for these edges, is therefore not necessary.

We (mostly) apply dual solution methods to solve the SDPs. A dual method is not guaranteed to find the value of the optimal solution, but it computes a lower bound on the optimal value and terminates if no further (significant) improvement is foreseen. Primal-dual methods for solving SDPs exist as well. Such a method computes primal as well as dual feasible solutions and terminates if a provably optimal solution is found or the value of the primal solution is sufficiently close to the lower bound. Hence, by design, a primal-dual method produces more reliable information on the optimal value than a dual method.

To our best knowledge, however, no presently available primal-dual SDP solver can handle the sizes of our problems. For example, the software package SeDuMi of Sturm [1998] requires more than 800 MB of memory to solve a problem on merely 100 vertices. Hence, we use two implementations of dual methods, namely, Bmz by Burer, Monteiro, and Zhang [1999] and Sb by Helmberg [2000]. In both cases, the running times for solving our SDPs range from several minutes up to days or even weeks.

SeDuMi

Bmz

Sb

Due to the dual character of the employed SDP solvers, a few irregularities can be observed in their computational behavior. The best lower bound, for example, may not be obtained for the SDP derived from the entire (simplified) carrier network, but rather for one which is derived from an induced subnetwork. A thorough comparison of these and other SDP solvers is currently performed as part of the seventh DIMACS implementation challenge, see Johnson, Pataki, and Alizadeh [2000] and Mittelmann [2000, Semidefinite/SQL Programming].

We give an account of our computational results in Table 6.2. In addition to the bound obtained for the entire carrier network (*all*), we report results for a subnetwork induced by a maximum clique (*clique*) and for the subnetwork induced by the union of all cliques larger than the number of available channels (*union*), not counting blocked channels. A field is left blank if no reasonable result has been produced, either

	clique	union	all
K	0.0206	0.1735	**0.1836**
B[0]	0.0013	**0.0096**	
B[1]	0.0052	**0.0297**	
B[2]	0.0213	**0.4747**	0.1097
B[4]	0.2893		**4.0342**
B[10]	2.7035		**54.0989**
Sie1	0.0165	0.1242	**0.1280**
Sie2	1.3378	6.8300	**6.9463**
Sie3	0.0444	**0.4132**	0.4103
Sie4	0.4598	21.4610	**27.6320**

Table 6.2: Lower bounds on unavoidable interference

because the solver stops prematurely with a negative objective function value or because this value is still negative after running for at least one day. The best lower bound for each scenario is typeset in bold face. Almost all entries are produced using Bmz, due to its superior running time behavior in comparison to Sb. The exceptions are as follows. The bound for K/clique is computed with SeDuMi. The other two bounds for K are best for Sb. The same holds for B[4]/all and B[10]/all.

Table 6.3 lists quality guarantees for the assignments from Table 5.15 on the basis of the best bound for each scenario except for Sw. In order to provide information on the split between co- and adjacent channel interference in the assignments, we give the respective values for the assignments from Threshold Accepting. The gap $\frac{I_y-L}{L}$ between the total interference I_y incurred by a assignment y produced heuristically and the lower bound L on the unavoidable co-channel interference is reported in percent. If no feasible assignment is generated by a heuristic for some scenario, then the corresponding cell contains a "—."

These are the first significant lower bounds on the amount of overall unavoidable interference for realistic GSM frequency planning scenarios. (Some forerunners of these bounds, obtained from an even weaker relaxation, are described by Eisenblätter [1998].) We learn, for example, that the best of the assignments listed for K, B[10], Sie2, and Sie4 incur no more than three times the amount of provably unavoidable interfer-

	lower	best assignment		gap [%]			
	bound	co-ch.	ad.-ch.	d	c	b	a
K	**0.1836**	0.43	0.02	145	347	668	657
B[0]	**0.0096**	0.55	0.02	5838	9692	25942	32504
B[1]	**0.0297**	0.84	0.02	2796	15254	12762	5388
B[2]	**0.4747**	3.10	0.07	568	1107	—	2215
B[4]	**4.0342**	17.29	0.44	339	580	—	800
B[10]	**54.0989**	142.09	4.11	170	273	339	316
Sie1	**0.1280**	1.06	1.24	1697	2525	3658	3845
Sie2	**6.9463**	12.57	2.18	112	175	195	201
Sie3	**0.4132**	3.64	1.62	1173	—	1911	1911
Sie4	**27.6320**	71.09	9.87	193	—	279	279

a. T-Coloring ∘ Vds; b. Ts T-Coloring∘ Vds; c. Ts Dc5 ∘ Vds;
d. Threshold Accepting

Table 6.3: Quality guarantees for selected frequency assignments

ence. Yet, it is fair to say that our quality guarantees are not particularly strong. The gap between the upper and the lower bound is considerable in all cases. Nevertheless, the gaps are small enough so that the differences in quality between simple (and fast) methods like DSATUR WITH COSTS and the more intricate method THRESHOLD ACCEPTING are not diminished.

One might conjecture that the large gaps between upper and lower bounds are to be blamed primarily on the lower bounds. After all, we are merely approximately solving a relaxation of a relaxation of the original problem. In the next section, however, we present a frequency plan for the scenario K that is much better than the best one reported in Table 5.15. This shows that the heuristically generated plans are not generally as good as one might have hoped.

6.4 Relaxed and Ordinary Frequency Planning

We would like to turn a feasible frequency assignment for a simplified carrier network into a feasible one for the associated ordinary carrier network. The only change we allow is to relabel the channels in the assignments. Thus, among the $|C|!$ many equivalent assignments for the simplified carrier network we look for one which is feasible for the original problem (and incurs as little adjacent channel interference as possible). How to find such a permutation is the topic of this section.

The absence of locally blocked channels, of separation constraints larger than one, and of adjacent channel interference simplifies the tasks of a planning heuristic. Running the same heuristic methods as before, we often obtain frequency plans with less co-channel interference than in the original setting, and we try to take advantage of this as follows. First, we heuristically produce a frequency plan for the relaxed problem; then, provided the plan is feasible, we try to turn this plan into a feasible one for the original problem by relabeling the channels. If we succeed, the new plan will have the same amount of co-channel interference as the one for the relaxed case. The amount of additional adjacent channel interference should be as small as possible. Although this procedure is certainly of limited applicability, we obtain the best known frequency plan for scenario K in this way. This assignment incurs 18 % less interference than the previously best one.

6.4.1 Feasible Permutations

We start out by formally introducing the notion of a feasible permutation.

feasible permutation

Definition 6.5. *Given a carrier network N and a feasible assignment y for the associated simplified carrier network N_s, we call $\pi\colon C \to C$ a* feasible permutation *of the channels if the assignment $\pi \circ y$ is feasible for N.*

Testing whether a feasible permutation exists is $\mathcal{NP}$-hard in general. (The $\mathcal{NP}$-complete problem of checking for a Hamiltonian Path in an undirected graph, compare with Garey and Johnson [1979, GT39], can be reduced to it.) In the following, we describe a method for finding a feasible permutation. Despite the fact that the running time of the method is not polynomially bounded in the input size in general, the method is merely a heuristic. Its results are, however, optimal if no channel is locally blocked (in particular, the spectrum has to be contiguous).

The major building block of the proposed method is to find a Hamiltonian path of minimum weight satisfying one extra condition. Recall that a path in a graph is called *Hamiltonian* if it contains every vertex. The extra condition rules out "shortcuts," which are defined below.

Hamiltonian path

Given a carrier network $N = (V, E, C, \{B_v\}_{v\in V}, d, c^{co}, c^{ad})$ and a feasible assignment y for the simplified network $N_s = (V, E, C, \underline{d}, c^{co})$, we construct a complete graph $K_{|C|}$. The channels in the spectrum C are the vertices. The weight w_{ij} of edge ij is the maximum of 1 and the separations required among all carriers assigned channel i and all carriers assigned channel j, i. e., $w\colon E(K_{|V|}) \to \mathbb{Z}_+, cc' \mapsto w_{cc'} = \max\{1, d(vx) \mid vx \in E : y(v) = c, y(x) = c'\}$. We call this edge-weighted graph the *separation graph* associated to carrier network N and assignment y.

separation graph

Every Hamiltonian path $p = v_0, \ldots, v_{|C|-1}$ in the separation graph defines a permutation $\pi_p\colon C \to C$. The permutation π_p is obtained from the order in which the vertices occur in the path $p = \pi_p(c_1), \ldots, \pi_p(c_{|C|})$. Clearly, the following holds.

Observation 6.6. *The Hamiltonian paths in the separation graph and the permutations on C are in one-to-one correspondence.*

We call a path $p = v_0, \ldots, v_l$ in the separation graph a *shortcutting path* or simply a *shortcut* if $\sum_{i=1}^{l} w_{v_{i-1}v_i} < w_{v_0v_l}$. A path $p = a, b, c$ with three vertices, for example, is a shortcut if and only if the triangle inequality $w_{ab} + w_{bc} \geq w_{ac}$ is violated. We say that a path *contains a shortcut* if some of its consecutive vertices form a shortcutting path.

shortcut

Figure 6.2 gives an illustration of a separation graph on the vertex set $\{a, b, c, d, e\}$. The edge weights are written next to the edges; that is, *ab*, *ad*, *bc*, *bd*, and *de* require a separation of 1; *ae*, *ce* a separation of 2; and *ac* a separation of 3. The triangle inequality is violated once, namely, $w_{ab} + w_{bc} < w_{ac}$, and the paths *abc* and *cba* are shortcuts. The Hamiltonian path *abcde* of weight 4 contains a shortcut, whereas the Hamiltonian paths *bcdea* of weight 5 does not. The Hamiltonian path *abedc* contains no shortcut and its weight of 4 is minimal. The associated permutation π_{abedc} is defined by $\pi_{abedc}(a) = a$, $\pi_{abedc}(b) = b$, $\pi_{abedc}(c) = e$, $\pi_{abedc}(d) = d$, and $\pi_{abedc}(e) = c$.

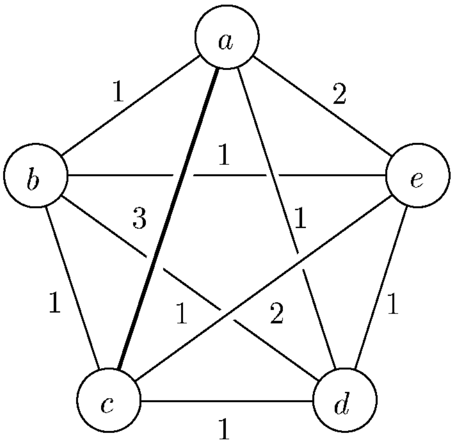


Figure 6.2: Separation graph with shortcuts

Although, shortcuts in a separation graph may contain more than two edges, their length is bounded by the largest edge weight in the separation graph, and thus by the largest required separation in the underlying carrier network.

Observation 6.7. *Given a shortcut* $v_0, \ldots, v_l$ *in a separation graph, the number of its vertices is bounded by* $l \leq w_{v_0 v_l}$. *Hence, every shortcut contains at most* $\max\{w_{ij} \mid i, j \in C\}$ *many vertices.*

The first part follows from $w_{ij} \geq 1$ for all $i, j \in C$ and the latter is a consequence of the former. For the carrier graphs introduced in Section 5.1.1, the largest required separation is three in all cases except for K, where it is four. Four is also the largest value for which we know a technical reason in the underlying GSM network.

Observation 6.8. *If a Hamiltonian path p in a separation graph has*

- *weight* $\sum_{i=1}^{|C|-1} w_{v_{i-1} v_i} \geq |C|$
- *or contains a shortcut,*

then π_p *is not feasible.*

Hence, only Hamiltonian paths with all edges having weight 1 may give rise to feasible permutations. The converse, however, holds only if no channel is locally unavailable.

In addition to finding some feasible permutation, we also want to keep the adjacent channel interference under control. This issue is addressed as follows. We set $l_{ij} = w_{ij} + \frac{ad_{ij}}{|C| \max\{ad_{ij} \,|\, i,j \in C\}}$ for every edge in the separation graph, where

$$ad \colon E \to \mathbb{R}_+, \quad ij \mapsto ad_{ij} = \sum_{\substack{v:y(v)=i \\ w:y(w)=j}} c^{ad}(vw).$$

Then $w_{ij} \leq l_{ij} \leq w_{ij} + |C|^{-1}$, and every Hamiltonian path of least weight with respect to l is of least weight with respect to w. With our preceding discussion in mind, the following is easy to see.

Proposition 6.9. *Given are a carrier network N without blocked channels, that is, $B_v = \emptyset$ for all $v \in V$, and a feasible assignment y for the corresponding simplified carrier network. Moreover, let p be a Hamiltonian path of least weight with respect to l in the separation graph associated to N and y. Then the following holds.*

If p has weight $|C| - 1$ with respect to w and does not contain a shortcut, then $\pi_p \circ y$ is a feasible solution for the carrier network N. Furthermore, $\pi_p \circ y$ incurs the least amount of adjacent channel interference among all feasible assignments $\pi \circ y$.

Next, we explain how such a Hamiltonian path can be computed heuristically by solving a modified Traveling Salesman Problem.

6.4.2 Tours without shortcuts

Finding a Hamiltonian path of minimum weight in a graph is traditionally done by solving a Traveling Salesman Problem. An instance of the Traveling Salesman Problem (TSP) consists of a complete graph together with edge weights. The task is to find a minimum weight *tour* (or cycle) containing every vertex. The weight of an edge is usually called its length in the context of a TSP, and a tour of minimum weight is called a *shortest tour*. We stick to this tradition. The TSP and many of its variations receive considerable attention in the literature. Jünger, Reinelt, and Rinaldi [1995a], for example, give a survey on the TSP, and Applegate, Bixby, Chvátal, and Cook [1998] report on recent progress.

TSP
tour
shortest tour

Our restricted Hamiltonian path problem is transformed into a restricted TSP as follows. First, we add one additional vertex σ to the separation graph, which is made adjacent to every other vertex. (This is the first step of the classical transformation of a Hamiltonian path problem to a TSP.) We call the result the *augmented separation graph*. The edge weighting l of the separation graph is extended to the augmented graph by letting $l_{\sigma v} = \max\{l_{ij} \mid i, j \in C\}$ for every edge incident to σ.

augmented separation graph

The notion of a shortcutting path is transfered to the augmented separation graph. A path $p = v_0, \ldots, v_l$ is called a *shortcut* if $\sum_{i=1}^{l} \lfloor l_{v_{i-1}v_i} \rfloor < \lfloor l_{v_0 v_l} \rfloor$, where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x. The slight change in the definition of a shortcut has the advantage that a path is a shortcut in the augmented separation graph (with respect to l) if and only if it is a shortcut in the separation graph (with respect to w). No shortcut contains σ. A Hamiltonian paths (of minimum weight) in the separation graph gives rise to a (shortest) tour in the augmented graph by connecting both endpoints of the path to σ. Conversely, every (shortest) tour in the augmented graph gives rise to two Hamiltonian path (of minimum weight) by chopping off σ and reading the remaining path in both possible directions. The same direct correspondence holds under the condition that no shortcuts may be contained.

shortcut

The ILP (6.6) is the classical integer linear programming formulation of the TSP, extended by the constraints (6.6c), called *shortcut constraints* here. A binary variable x_{ij} is used for every edge ij in the augmented separation graph, and an edge ij is in the tour if and only if x_{ij} is 1. Without the shortcut constraints, every optimal solution corresponds to a shortest tour. With the shortcut constraints included, each optimal solution corresponds to a tour that does not successively contain a shortcut and is shortest among all those.

shortcut constraints

$$
\begin{array}{llll}
\min\limits_{x} & \sum\limits_{ij \in E} l_{ij}\, x_{ij} & & \\
\text{s.t.} & & & \\
\sum\limits_{j \in V : ij \in E} x_{ij} & = 2 & \forall\, i \in V & \text{(6.6a)} \\
\sum\limits_{\substack{i \in S, j \in V \setminus S \\ ij \in E}} x_{ij} & \geq 2 & \forall\, \emptyset \subsetneq S \subsetneq V & \text{(6.6b)} \\
\sum\limits_{ij \in p} x_{ij} & \leq |p| - 1 & \forall\, \text{shortcuts } p & \text{(6.6c)} \\
x_{ij} & \in \{0, 1\} & \forall\, ij \in E &
\end{array}
$$

The use of a TSP for finding a feasible permutation is inspired by the TSP bound for the minimum span of particular T-coloring problems, see Section 3.1.2. This connection is first observed by Raychaudhuri [1985] and used by Roberts [1991b]; Janssen and Kilakos [1996, 1999]. Allen, Smith, and Hurley [1999] extend this idea and augment their TSP formulation by introducing "excess"-variables in order to rule out conflicts with shortcuts. Our idea is similar to their's, although in a slightly different context and with a different goal.

6.4.3 Computational Results

Any branch-and-cut method for the ordinary TSP can, in principle, be modified in order to solve the ILP (6.6). Recall from the previous subsection that in our case none of the separation graphs contains a shortcut of length four or more. In this case, the number of shortcuts is polynomially bounded in the problem size, and all shortcuts can be enumerated in polynomial time. We do not pursue this further here, because this is not in the center of our interest. Instead, we use the state-of-the-art solver for the ordinary TSP with some additional processing. The CONCORDE program, developed by Applegate, Bixby, Chvátal, and Cook [1997], is employed to solve the basic TSPs.

We are often lucky and find a shortest tour not containing a shortcut. But there are, of course, cases where the shortest tour does contain a shortcut. We then resort to the following crude heuristic. A shortcutting path is determined, and the length of one edge is increased in such a way that this shortcut is eliminated and no other is introduced. (There are several possible variations of this principle.) The TSP is then solved afresh, and the process is repeated until a tour without shortcut is obtained. If any such tour exists, then one will ultimately be found due to the way in which we increase the lengths.

Computational results are given for the scenarios K and B[1] in Table 6.4. The scenario SIE1 is omitted here, because its spectrum is not contiguous and this does not fit with our assumptions. A feasible solution is generated with the THRESHOLD ACCEPTING heuristic for the associated simplified carrier networks (*all*), for the subnetworks induced by a maximum *clique*, and for the subnetworks induced by the *unions* of cliques larger than the size of the spectrum. Every available channels is used at least once in each assignment. The table displays the co-channel interference of the assignments with respect to the simplified carrier networks. The adjacent channel interference and the number of separation violations with respect to the associated carrier networks are also given.

		interference		sep.
		co-ch.	adj.-ch.	viols.
K	all	0.37	1.38	38
	union	0.37	1.33	36
	clique	0.02	0.76	6
B[1]	all	0.59	2.23	212
	union	0.11	0.21	21
	clique	0.01	0.27	6

Table 6.4: Analysis of assignments for simplified carrier networks

Table 6.5 gives details concerning the separation graphs constructed from each of these assignments. Ordered by columns, we first list the number of vertices in the (induced) simplified carrier network. For the separation graph, we then list the number of vertices, a histogram of the edge weights, the number of violated triangle inequalities, and the minimal number of edges of weight 1 incident to a vertex (minimal 1-degree). Recall that every violated triangle inequality gives rise to a shortcut of length two. In the case of B[1]/all and the given assignment, some vertex in the associated separation graph has no incident edge of weight 1. A feasible permutation cannot exist.

				weight				viol.	min.
		$\|V\|$	$\|C\|$	1	2	3	4	$\triangle$	1-deg.
K	all	267	50	476	746	0	3	78	8
	union	233	50	511	711	0	3	92	7
	clique	69	50	1101	123	0	1	36	36
B[1]	all	1971	75	157	2532	86	0	24	0
	union	252	75	2098	669	8	0	373	38
	clique	84	75	2554	218	3	0	198	54

Table 6.5: Analysis of separation graphs

In all other cases, a feasible permutation is obtained by applying the TSP-based heuristic. Notably, the resulting assignments incur no or only very little additional adjacent channel interference. This is achieved by using the augmented separation graph when determining the best permutation. All permutations are optimal with respect to the amount of adjacent channel interference incurred. If additional interference is incurred, then the corresponding permutation is obtained without changing any edge weight in the augmented separation graph. The results are given in Table 6.6. The table also displays the gap between the permuted assignments and the lower bounds listed in Table 6.2. Attempts to further improve the assignments using VDS fail: only in one case a negligible improvement is achieved.

		interference		sep.	gap
		co-ch.	adj.-ch.	viols.	[%]
K	all	0.37	0.0009	0	102
	union	0.37	0.0009	0	114
	clique	0.02	0.0000	0	0
B[1]	all	—	—	—	—
	union	0.11	0.0000	0	270
	clique	0.01	0.0000	0	92

Table 6.6: Analysis of permuted assignments for carrier networks

As stated before, the assignment obtained for the instance K/all is significantly better than the previously reported ones. The total interference is reduced from 0.46 or more to a value of 0.37. Correspondingly, the gap is reduced from 151 % or more to 102 %.

6.5 Conclusions

In the preceding chapters, we have dealt with models and heuristics for frequency planning in GSM networks. Here, we considered the issue of proving that, for a given carrier network, a certain amount of interference is unavoidable by any feasible frequency plan. This allows to compare the interference incurred by a frequency plan with the amount of provably unavoidable interference. In the ideal case, where both values are equal, the plan is proven to be optimal (in terms of the mathematical optimization problem FAP). But also in the more likely case, where the values do not coincide, knowing how much interference is unavoidable can be very helpful. We may use this information to estimate the quality of a frequency assignment, or use it as a common reference point when comparing results from several heuristics.

Unfortunately, it is still unknown how to compute strong lower bounds on the interference in general. We proposed an approach to bound the unavoidable co-channel interference from below. In this context, we introduced the "relaxed frequency planning problem" and explained its relation to the MINIMUM K-PARTITION problem. Drawing on the semidefinite relaxation of the MINIMUM K-PARTITION problem and using state-of-the-art SDP solver, we computed the first significant lower bounds for the frequency planning problem FAP. In the best case, we show that a frequency assignment incurs merely twice the amount of provably unavoidable interference. In the worst case, however, the factor is almost 60. This situation is not satisfactory and deserves further investigations

A drawback of our approach is that solving the large semidefinite programs is presently quite challenging and may take days or even weeks of running time. Moreover, it is not yet clear how information acquired through the solution of the semidefinite program can be effectively exploited to generate better frequency plans. This also deserves additional attention.

We pointed out that for our scenarios many triangles in the carrier graph violate the triangle inequality significantly. Alternative approaches to compute strong lower bounds on the unavoidable interference may fail if they rely on the triangle inequalities being (almost) fulfilled.

Finally, frequency planning methods for GSM networks have been developed for almost a decade now, and several of the recent methods show good performance in practice. One might conclude that frequency planning, at least from a practical point of view, can be considered as "solved," see also our comments in Section 5.6. In this respect, however, it is irritating that we were able to provide a significantly better solution to a realistic planning problem than known before. The employed method, i. e., first solving the relaxed frequency planning problem heuristically and then trying to fix all separation violations by relabeling the channels, is certainly of limited applicability. Its success, however, documents that improvements on the presently used techniques are possible. This will be of interest for GSM network planners if, at some future point in time, frequency planning again becomes a limiting factor. From the combinatorial optimization point of view, the frequency assignment problem FAP, even on the rather restricted set of realistic planning data, is clearly not yet fully mastered.

Chapter 7

Partition Polytopes

The focus of the remaining two chapters is no longer directly on frequency assignment. Instead, we pursue the problem of finding a k-partition of the vertex set in a complete graph such that the edge weights in the induced subgraphs are minimal (Minimum k-Partition). We are led to this problem by its close relation to the relaxed frequency planning problem, see Section 6.2.

In this chapter, we mostly survey results from the literature concerning the polytope $\mathcal{P}_{\leq k}(K_n)$, which is defined by the convex combination of all feasible solutions to the ILP formulation (6.2) of the Minimum k-Partition problem. This polytope is full-dimensional in the space spanned by the edge variables. A particular emphasis is on the *hypermetric inequalities.* Moreover, we address the complexity of solving the separation problem for several classes of (facet-defining) inequalities.

In the next chapter, we turn to the semidefinite relaxation (6.5) of the Minimum k-Partition problem. The set of the relaxation's feasible solutions is studied and related to the polytope $\mathcal{P}_{\leq k}(K_n)$. This is done mostly on the basis of the hypermetric inequalities.

The chapter is organized as follows. In Section 7.1, two binary linear programming formulations of the Minimum k-Partition problem are compared. We explain why the formulation (6.2) is favored. In Section 7.2, the polytope obtained for $k = n$ is studied. Several classes of facet-defining inequalities from the literature are reviewed. In Section 7.3, we consider the polytope $\mathcal{P}_{\leq k}(K_n)$ for $k \leq n$ with a strong emphasis on the hypermetric inequalities. In Section 7.4, we briefly discuss the issue of developing a branch-and-cut algorithm for the Minimum k-Partition problem on the basis of the classes of valid inequalities presented in the preceding two sections.

Appendix A contains a compilation of mathematical notions, which are used but not introduced here.

Usually, we assume $k \geq 3$ or even $k \geq 4$ in the following. Clearly, if $k = 1$, then only one "partition" exists; in the case of $k = 2$, the Minimum

2-PARTITION problem is equivalent to the well-known MAXIMUM CUT problem, see the survey article by Poljak and Tuza [1995] or the book by Deza and Laurent [1997] and the references contained therein.

7.1 Binary Linear Programs

Let $G = (V, E)$ be a graph with at least three vertices, $w\colon E \to \mathbb{Q}$ be a weighting of the edges, and $2 \le k \le |V|$ integer. Two binary linear programming formulations of the MINIMUM K-PARTITION problem are considered in the literature.

The first formulation (7.1) given below is the same as (6.2). It is restated here for convenience. One binary variable is used for every edge of the graph, which has to be complete. Thus, $\binom{|V|}{2}$ many variables occur. The intended meaning is that $z_{ij} = 1$ if and only if the vertices i and j are in the same partite set of the partition. The number of triangle inequalities (7.1a) is $3\binom{|V|}{3}$, and there are $\binom{|V|}{k+1}$ many clique inequalities (7.1b). The value of the expression $\binom{|V|}{k+1}$ grows roughly as fast as $|V|^k$ as long as $2k \le |V|$. Hence, the number of constraints is not bounded by a polynomial in $|V|$ and $\log k$. Deza *et al.* [1991, 1992] as well as Chopra and Rao [1995] consider this formulation.

$$
\begin{array}{lll r}
\min \sum\limits_{i,j \in V} w_{ij}\, z_{ij} & & & \\
\text{s. t.} & & & \\
z_{ih} + z_{hj} - z_{ij} & \le 1 & \forall\, h, i, j \in V & (7.1\text{a}) \\
\sum\limits_{i,j \in Q} z_{ij} & \ge 1 & \forall\, Q \subseteq V \text{ with } |Q| = k+1 & (7.1\text{b}) \\
z_{ij} & \in [0,1] & & \\
z_{ij} & \ \text{integer} & &
\end{array}
$$

For the second formulation (7.2) the graph does not have to be complete. In addition to the edge variables, k binary variables y_i^l, $l = 1, \ldots, k$, are introduced for every vertex i with the obvious meaning of $y_i^l = 1$ if and only if the vertex i is in lth set of the partition. Hence, the number of variables is $k|V| + |E|$. There are $|V|$ many constraints of type (7.2a), and $3k\,|E|$ constraints of type (7.2b). This formulation is considered by

Chopra and Rao [1993], for example.

$$
\begin{aligned}
&\min \sum_{ij \in E} w_{ij}\, z_{ij} \\
&\text{s. t.} \\
&\sum_{l=1}^{k} y_i^l = 1 \qquad \forall i \in V && \text{(7.2a)} \\
&-y_i^l - y_j^l + z_{ij} \ge -1 \qquad \forall ij \in E, \forall l \in \{1,\dots,k\} \\
&-y_i^l + y_j^l - z_{ij} \ge -1 \qquad \forall ij \in E, \forall l \in \{1,\dots,k\} && \text{(7.2b)} \\
&+y_i^l - y_j^l - z_{ij} \ge -1 \qquad \forall ij \in E, \forall l \in \{1,\dots,k\} \\
&y_i^l, z_{ij} \in [0,1] \\
&y_i^l, z_{ij} \text{ integer}
\end{aligned}
$$

For a sparse graph, the formulation (7.2) may have significantly fewer variables than (7.1) after the graph is completed with edges of weight zero. Despite this fact, the formulation (7.2) has a major drawback. In case the vertex set of a graph is partitioned into k (nonempty) sets, then there is a unique variable setting corresponding to this partition in (7.1). But there are $k!$ many corresponding settings in (7.2). This is because the introduction of the y variables goes along with the necessity of labeling the classes of the partition. Although such a labeling is mandatory in an ILP formulation of the frequency planning problem, compare with (3.6) in Section 3.3.1, it introduces unnecessary and unwelcome degrees of freedom here.

We are not aware that either of these formulations has been used successfully for solving Minimum k-Partition problems with nonnegative weights on dense graphs with several hundred vertices.

No formulation using edge variables alone is known for incomplete general graphs. Taking, for example, simply the first formulation and applying the triangle-based constraints (7.1a) and the clique-based constraints (7.1b) merely to induced subgraphs does not work. One reason is that a consistent setting in the vector z cannot be ensured for incomplete graphs by considering triangles alone. Instead, restrictions on all chordless induced cycles apply. Let C be a chordless induced cycle in G and let $\overline{ij}$ be any edge in $E(C)$, then

$$\sum_{\substack{ij \in E(C) \\ ij \neq \overline{ij}}} z_{ij} - z_{\overline{ij}} \le |E(C)| - 2$$

has to be satisfied. The number of chordless cycles in a graph can, in general, not be bounded by a polynomial in the number of vertices. Another, more intricate reason is the following: imposing that every induced clique of size $(k+1)$ is partitioned into at most k classes does not guarantee that the entire vertex set is consistently partitioned into at most k classes. Instead, the following has to be imposed:

> A vector $z \in \{0,1\}^E$ is infeasible unless the graph obtained from G by contracting all $e \in E$ with $z_e = 1$ is k-partite.

One way to impose this condition is the introduction of vertex variables, as done in (7.2).

The focus in the remainder of this chapter is on the first formulation and its associated polytope, i. e., the set of all convex combinations of feasible solutions. From now on the underlying graph is assumed to be a complete graph K_n with $n \geq 3$. We denote the convex hull of all integral points satisfying the conditions given in (7.1) by

$\mathcal{P}_{\leq k}(K_n)$

$$\mathcal{P}_{\leq k}(K_n) = \operatorname{conv}\Big(\Big\{z \in \{0,1\}^{E(K_n)} \mid z_{hi} + z_{ij} - z_{hj} \leq 1 \quad \forall\, h,i,j \in V; \\ \sum_{i,j \in Q} z_{ij} \geq 1 \quad \forall\, Q \subseteq V, |Q| = k+1\Big\}\Big).$$

Every partition of the vertex set of K_n into at most k many sets is also a partition with at most $k+1$ many sets. Thus,

$$\mathcal{P}_{\leq 2}(K_n) \subsetneq \cdots \subsetneq \mathcal{P}_{\leq k}(K_n) \subsetneq \mathcal{P}_{\leq k+1}(K_n) \subsetneq \cdots \subsetneq \mathcal{P}_{\leq n}(K_n),$$

and each inclusion is proper. Every inequality valid for $\mathcal{P}_{\leq n}(K_n)$ is also valid for $\mathcal{P}_{\leq k}(K_n)$ for every $2 \leq k \leq n$. The boundary cases $k = 2$ and $k = n$, i. e., at most two classes and no restriction on the number of classes, have already been studied extensively in the literature. The shorthand notation $\mathcal{P}(K_n)$ for $\mathcal{P}_{\leq n}(K_n)$ is used in the following.

$\mathcal{P}(K_n)$

Observation 7.1. *$0 \in \mathcal{P}(K_n)$, but $0 \notin \mathcal{P}_{\leq k}(K_n)$ for every $2 \leq k < n$.*

Hence, no matter how good the knowledge of $\mathcal{P}(K_n)$ in terms of valid and facet-defining inequalities is, without taking the clique inequalities (7.1b) into account, the optimal value of (7.1) will always be nonpositive. In fact, if all weights are nonnegative, then the zero-vector is always an optimal solution when minimizing over $\mathcal{P}(K_n)$.

7.2 The Polytope $\mathcal{P}(K_n)$

Our survey on properties of $\mathcal{P}(K_n)$ does not aim at completeness. The Ph. D. thesis of Rutten [1998] contains a more comprehensive compilation (which, however, omits the results of Deza *et al.* [1991]).

Proposition 7.2 (Grötschel and Wakabayashi [1990]). *The polytope $\mathcal{P}(K_n)$ has dimension $\binom{|V|}{2}$.*

The clique inequalities (7.1b) are void for $\mathcal{P}(K_n)$. The remaining constraints in the binary linear programming formulation are the triangle inequalities (7.1a), the bounds on the variables, and the integrality conditions. The bound constraints are called trivial inequalities.

trivial inequalities

Proposition 7.3 (Grötschel and Wakabayashi [1990]). *With respect to the polytope $\mathcal{P}(K_n)$, $n \geq 3$,*

- *every nonnegativity constraint $z_{ij} \geq 0$ defines a facet,*
- *every triangle inequality* (7.1a) *defines a facet,*
- *no upper bound constraint $z_{ij} \leq 1$ defines a facet.*

Some properties are shared by all nontrivial facet-defining inequalities for $\mathcal{P}(K_n)$. One of the three properties listed next concerns the support graph of an inequality. Given some inequality $a^T z \geq a_0$, the *support graph* of $a^T z \geq a_0$, or just a, is the subgraph of K_n induced by all edges ij with $a_{ij} \neq 0$.

support graph

Proposition 7.4 (Grötschel and Wakabayashi [1990]). *Let $a^T z \leq a_0$ be a nontrivial inequality defining a facet of $\mathcal{P}(K_n)$, then*

- *$a_0 > 0$,*
- *a has positive and negative entries,*
- *the support graph of $a^T z \leq a_0$ is 2-connected.*

Corollary 7.5. *Under the same assumptions as in Proposition 7.4, the subgraph H^+ of K_n induced by $E_a^+ = \{ij \in E(K_n) \mid a_{ij} > 0\}$ is a connected, spanning subgraph of the support graph.*

Proof. Let $[S, T]$ be any cut in the 2-connected support graph H induced by E_a. There exists a vector $\overline{z} \in \{0,1\}^{E(K_n)}$ satisfying $a^T \overline{z} = a_0 > 0$ for which restriction to the cut $[S, T]$ is not identical to zero. (Otherwise, the facet defined by $a^T \overline{z} \geq a_0$ would be contained in all the trivial facets defined by $z_{st} \geq 0$ with $st \in [S,T]$, which is a contradiction.) The vector $\underline{z}$, obtained from $\overline{z}$ by setting

$$\underline{z}_{ij} = \begin{cases} 0, & \text{if } ij \in [S,T], \\ \overline{z}_{ij}, & \text{otherwise,} \end{cases}$$

is also the characteristic vector of a partition and thus satisfies $a^T\underline{z} \leq a_0$. Consequently, $a^T\overline{z} - a^T\underline{z} = \sum_{st\in[S,T]} a_{st}\overline{z}_{st} \geq 0$, and at least one of the a_{st}'s, $st \in [S,T]$, has to be positive.

The claim now follows since this holds for every cut in H. □

Notice, however, that H^+ is not always 2-connected. The 2-chorded path inequalities, see (7.4), form a counterexample. They can even have arbitrarily large support.

Recall that we want to partition the vertex set of a complete graph by means of deciding for each edge whether or not its two endpoints are in the same partite set. The setting of the edge variables has to be transitive, i. e., the subgraphs induced by selected edges have to be complete. In that respect, Proposition 7.4 and Corollary 7.5 can be read as follows: all facet-defining inequalities for $\mathcal{P}(K_n)$ are concerned with imposing the additional (fractional) selection of edges with negative coefficient once some set of edges with positive coefficient has already been (fractionally) selected. No inequality may, however, impose the selection of an edge to begin with, because the origin is contained in $\mathcal{P}(K_n)$.

7.2.1 2-chorded Inequalities

Two fairly general classes of valid inequalities are known for the polytope $\mathcal{P}(K_n)$. These are treated in the two subsequent subsections. Here, we list a few other classes, which have in common that the support graph of the inequality has 2-chords. Given a graph $G = (V,E)$, we call an edge ij a *2-chord* if there exists some $h \in V$ such that $ih, jh \in E$. A *2-chorded cycle* is a cycle with all 2-chords added. A 2-chorded cycle with q vertices along the cycle is denoted by C_q^2.

2-chord
2-chorded cycle
C_q^2

Proposition 7.6 (Grötschel and Wakabayashi [1990]). *Let C_q^2 be a 2-chorded cycle of length $q \geq 5$ in K_n. Let C be the edges of the cycle, and let $\bar{C}$ be the set of 2-chords, then the* 2-chorded cycle inequality

2-chorded cycle inequality

$$z(C) - z(\bar{C}) \leq \left\lfloor \frac{q}{2} \right\rfloor \qquad (7.3)$$

is valid for $\mathcal{P}(K_n)$. The 2-chorded cycle inequality defines a facet of $\mathcal{P}(K_n)$ if and only if $q \geq 5$ is odd.

Figure 7.1 depicts a 2-chorded cycle inequality on a cycle with 7 vertices. Solid lines indicate a coefficient of +1 in the corresponding inequality, whereas broken lines have a coefficient of −1.

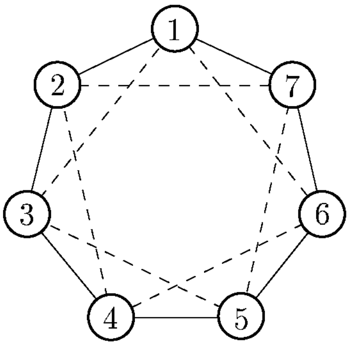


Figure 7.1: Support graph of a 2-chorded cycle inequality on 7 vertices

A *2-chorded path* is a path with all 2-chords added. We denote the 2-chorded path on q vertices by P_q^2. *2-chorded path P_q^2*

Proposition 7.7 (Grötschel and Wakabayashi [1990]). *Let P_q^2 be a 2-chorded path of length $q \geq 2$ together with a simplicial vertex $h \notin V(P_q^2)$. Let P be the edges of the path, $\bar{P}$ be the set of 2-chords, R be the set of edges from the simplicial vertex to every other vertex of P_q^2, starting with the second path vertex, and let $\bar{R}$ be the set of edges from the simplicial vertex h to every other vertex of P_q^2, starting with the first vertex in the path. Then the* 2-chorded path inequality *2-chorded path inequality*

$$z(P \cup R) - z(\bar{P} \cup \bar{R}) \leq \left\lfloor \frac{q+1}{2} \right\rfloor \tag{7.4}$$

is valid for $\mathcal{P}(K_n)$. The 2-chorded path inequality defines a facet of $\mathcal{P}(K_n)$ if and only if q is even.

In Figure 7.2, a 2-chorded path inequality on a path of length 3 is shown. As before, solid lines indicate a coefficient of +1 in the corresponding inequality, and broken lines have a coefficient of −1.

Consider a graph consisting of a cycle and one additional vertex, which is adjacent to all vertices on the cycle. This is called a *wheel* and denote by W_q, where q is the number of vertices along the cycle. We call every edge incident to a vertex on the cycle and to the simplicial vertex a *spoke*. A *2-chorded wheel* is a 2-chorded cycle wheel with an additional simplicial vertex. *wheel W_q* *spoke* *2-chorded wheel*

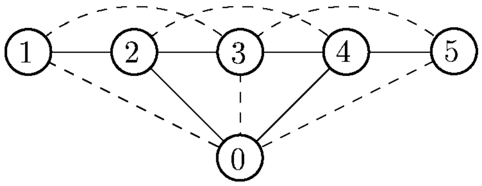


Figure 7.2: Support graph of a 2-chorded path inequality

2-chorded even wheel inequality

Proposition 7.8 (Grötschel and Wakabayashi [1990]). *Let C_q^2 be 2-chorded cycle of even length $q \geq 8$ in K_n and let $h \notin V(C_q^2)$. We denote by C the edges of the cycle, by $\bar{C}$ be the set of 2-chords, by R the set of edges from the simplicial vertex h to the every other vertex of C_q^2, and by $\bar{R}$ the set of remaining spokes. The* 2-chorded even wheel inequality

$$z(C \cup R) - z(\bar{C} \cup \bar{R}) \leq \frac{q}{2} \tag{7.5}$$

is valid for $\mathcal{P}(K_n)$ and defines a facet of $\mathcal{P}(K_n)$.

A 2-chorded even wheel inequality for a wheel with 8 vertices on the rim is depicted in Figure 7.2. Again, solid lines indicate a coefficient of +1 in the corresponding inequality and broken lines a coefficient of -1.

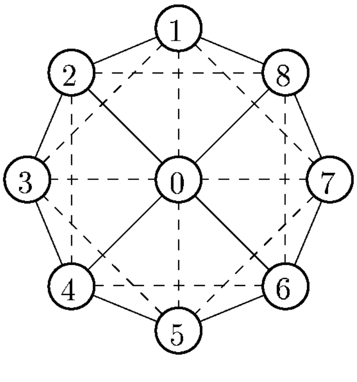


Figure 7.3: Support graph of a 2-chorded even wheel inequality

Generalizations of the previous three types of inequalities are given by Rutten [1998].

7.2.2 Clique-web Inequalities and Special Cases

The next two types of inequalities are specializations of the rather general clique-web inequalities (except for boundary cases), which are described at the end of this section.

Proposition 7.9 (Chopra and Rao [1993]). *Let W_q be a wheel in K_n with $q \geq 3$. We denote by C the edges along the rim, and by R the spokes of the wheel. The q-*wheel inequality

q-wheel inequality

$$z(R) - z(C) \leq \left\lfloor \frac{q}{2} \right\rfloor \tag{7.6}$$

is valid for $\mathcal{P}(K_n)$ and defines a facet if q is odd.

The support graph of a 5-wheel inequality is depicted in Figure 7.4. Solid lines indicate a coefficient of +1 in the corresponding inequality, and broken lines indicate a coefficient of −1.

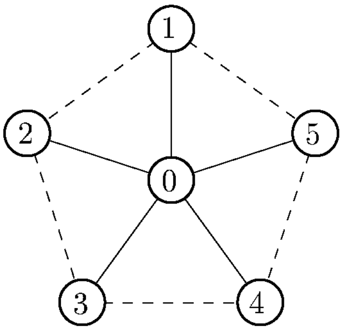


Figure 7.4: Support graphs of a 5-wheel inequality

A graph consisting of a cycle and two additional vertices, which are adjacent to each other and to all vertices on the cycle, is called a *bicycle* in the following. We denote a bicycle with q vertices along the cycle by BW_q.

bicycle BW_q

Proposition 7.10 (Chopra and Rao [1993]). *Let BW_q be a bicycle in K_n with $q \geq 3$. Let s_1 and s_2 be the two vertices not on the cycle, E_1 the spokes incident to s_1, and E_2 the spokes incident to s_2. The q-*bicycle inequality

q-bicycle inequality

$$z(E_1) - z(E_2 \cup \{s_1 s_2\}) \leq 2 \left\lfloor \frac{q}{2} \right\rfloor \tag{7.7}$$

is valid for $\mathcal{P}(K_n)$ and defines a facet if q is odd.

Figure 7.5 shows the support graph of a bicycle inequality with 5 vertices along the cycle. As before, a solid line stands for a coefficient of +1, and a broken line stands for a coefficient of −1 in the inequality.

Extending the notion of a 2-chord in a graph $G = (V, E)$, we call an edge ij an *l-chord*, $l \geq 2$, if there exists a path of length l in G with i and j as endpoints. We say that a cycle of length p is augmented with all l-chords, $2 \leq l \leq \frac{p}{2}$, if the endpoints of each path of length l on the cycle are connected by an edge.

l-chord

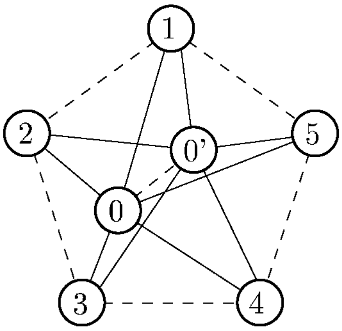


Figure 7.5: Support graph of a 5-bicycle inequality

antiweb AW_p^r

web W_p^r

Definition 7.11 (Deza and Laurent [1992b]). *Given two nonnegative integers p, r satisfying $p \geq 2r+1$, an* antiweb AW_p^r *is a graph on a vertex set $\{v_1, \ldots, v_p\}$. If $r = 0$, then there are no edges. In case $r \geq 1$, AW_p^r is a spanning cycle augmented by all l-chords with $l = 2, \ldots, r$. The* web W_p^r *is the graph complement of the antiweb AW_p^r.*

The web W_p^0, for example, is a complete graph on p vertices; W_{2r+1}^r is a graph containing $2r+1$ isolated vertices; and W_{2r+3}^r is a cycle on $2r+3$ vertices. A AW_7^2 antiweb and a W_7^2 web are shown in Figure 7.6.

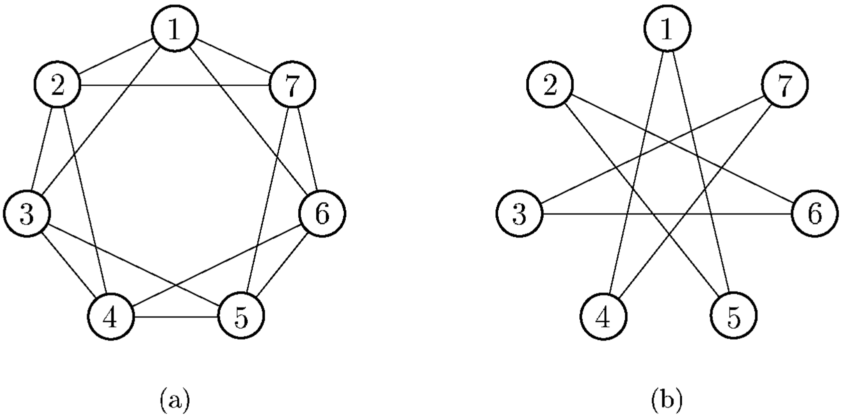


Figure 7.6: AW_7^2 is depicted in (a) and W_7^2 in (b)

clique-web inequality

Proposition 7.12 (Deza *et al.* [1991]). *Let $W_p^r = (W, F)$ be a web in K_n with $p \geq 1$ and $r \geq 0$ and let U be a set of vertices with $U \subseteq V(K_n) \setminus W$, $|U| = q \geq 1$ such that $p - q \geq 2r + 1$. Then the* clique-web inequality

$$z(E(U, W)) - z(W_p^r) - z(E(U)) \leq q(r+1) \tag{7.8}$$

is valid for $\mathcal{P}(K_n)$. The clique-web inequality defines a facet of $\mathcal{P}(K_n)$ in case of $p - q > 2r + 1$ or $p - q = 2r + 1$ and $q \geq 2$.

Setting $p = q$, $r = \frac{q-3}{2}$, and $|U| = 1$ in the clique-web inequality yields a q-wheel inequality. Keeping the same setting for p, q, and r, but considering $|U| = 2$, we arrive at the q-bicycle inequality. Furthermore, for $U = S$ and $T = W^0_{|T|}$ the 2-partition inequality, see 7.9 below, is obtained. The support graph of a clique-web inequality on a AW^2_7 antiweb and a set U of size one is depicted in Figure 7.7. As before, solid lines indicate a coefficient of $+1$ and broken lines a coefficient of -1 in the corresponding inequality.

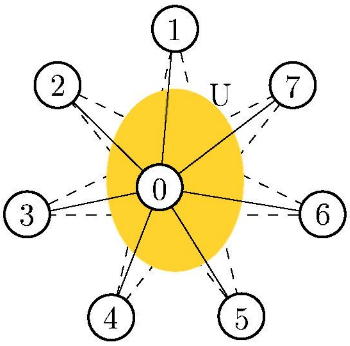


Figure 7.7: Support graph of clique-web inequality on AW^2_7 with $|U| = 1$

7.2.3 Partition and Claw Inequalities

The hypermetric inequalities are introduced by Deza and Laurent [1992a] for the Maximum Cut polytope, i. e., for the case of $k = 2$. They are generalized to $2 \leq k \leq n$ by Chopra and Rao [1995]. The right-hand side of a hypermetric inequality depends on k in a nontrivial way. We therefore defer the discussion of the hypermetric inequality to the next section, which is devoted to $\mathcal{P}_{\leq k}(K_n)$. Only two special cases are presented here.

2-partition inequality

Proposition 7.13 (Grötschel and Wakabayashi [1990]). *Let Q be a subset of the vertices in K_n of size at least 3, and let S, T be nonempty disjoint subsets of Q with $|S| \leq |T|$, then the* 2-partition inequality

$$z([S,T]) - z(E(S)) - z(E(T)) \leq |S| \tag{7.9}$$

is valid for $\mathcal{P}(K_n)$. The 2-partition inequality defines a facet of $\mathcal{P}(K_n)$ if and only if $|S| \neq |T|$.

The 2-partition inequalities generalize the triangle inequalities (7.1a). Figure 7.8 shows a triangle inequality and a (2,3)-partition inequality. Other generalizations of the 2-partition inequalities are the *general 2-partition inequalities*, described by Rutten [1998], as well as the clique-web inequality (7.8), both of which are not themselves hypermetric inequalities.

general 2-partition inequality

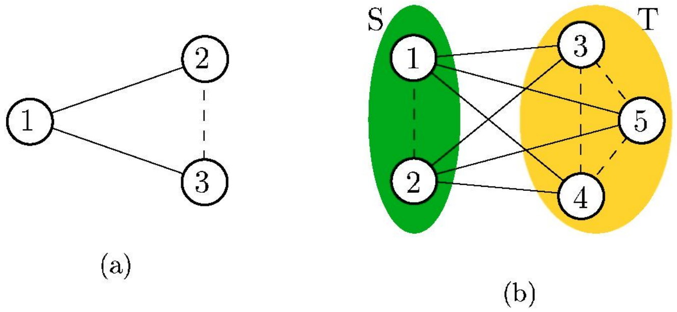


Figure 7.8: Support graphs of two 2-partition inequalities

claw inequality

Proposition 7.14 (Oosten, Rutten, and Spieksma [1995]). *Let $c \geq 1$ be integer and fix a vertex $s \in K_n$ and a vertex set $T \subseteq V(K_n) \backslash \{s\}$, then the* claw inequality

$$c \sum_{t \in T} z_{st} - \sum_{vw \in E(T)} z_{vw} \leq \binom{c+1}{2} \tag{7.10}$$

is valid for $\mathcal{P}(K_n)$. In case $c \geq 2$, a facet is defined if and only if $|T| \geq c+2$.

Notice, for $c = 1$, the claw inequality (7.10) is a special case of the 2-partition inequality (7.9).

7.3 The Polytope $\mathcal{P}_{\leq k}(K_n)$

We now drop the restriction $k = n$ and look at cases with $4 \leq k \leq n$. Sometimes $k = 2$ or $k = 3$ is also considered. By restricting the number of classes in the partition, the dimension of the polytope does not drop.

Proposition 7.15 (Barahona and Mahjoub [1986]). *The polytope $\mathcal{P}_{\leq k}(K_n)$ has dimension $\binom{n}{2}$ for every $2 \leq k \leq n$.*

The origin, however, is no longer contained in $\mathcal{P}_{\leq k}(K_n)$ if $k < n$. The valid inequalities for $\mathcal{P}_{\leq k}(K_n)$ which are violated by the origin have large support, namely:

Proposition 7.16. *Let $a^T z \leq a_0$ be a valid inequality for $\mathcal{P}_{\leq k}(K_n)$ and H its support graph. If H is k-partite, then $a_0 \leq 0$.*

Proof. Assume $H = (V_a, E_a)$ is k-partite, and let $V_1, \dots V_k$ be a partition of V_a into at most k many independent sets. Let $\bar{z}$ be the characteristic vector of the partition $(V(K_n)\setminus V_a) \cup V_1, V_2, \dots, V_k$. Then $a^T \bar{z} = 0$ because $\bar{z}_{vw} = 0$ for all $vw \in E_a$. Hence, $a_0 \leq 0$ has to hold in order for $a^T z \leq a_0$ to be valid. □

Corollary 7.17. *If $a^T z \leq a_0$ with $a_0 > 0$ is valid for $\mathcal{P}_{\leq k}(K_n)$, then the support set $E_a = \{ij \in E \mid a_{ij} \neq 0\}$ of a is of size at least $\binom{k+1}{2}$.*

Proof. The claim follows directly from Proposition 7.16, because every graph which is not k-partite has at least $\frac{(k+1)k}{2}$ many edges, see West [1996, p. 177], for example. □

A number of results in the literature give sufficient conditions on how a facet-defining inequality for the polytope $\mathcal{P}(K_m)$ can be extended onto the additional variables associated with $\mathcal{P}(K_n)$, $m < n$, such that the extended or "lifted" inequality is facet-defining for $\mathcal{P}(K_n)$. The simplest such result states that all new variables may receive the coefficient zero in the extended inequality. This is called *zero-lifting*.

zero-lifting

Proposition 7.18 (Deza and Laurent [1992a]; Chopra and Rao [1995]). *Let $a^T z \leq a_0$ be a facet-defining inequality for $\mathcal{P}_{\leq k}(K_m)$, $k \geq 2$, $m \geq 3$. Then, for every $n > m$, the inequality $\bar{a}^T z \leq a_0$ defines a facet of $\mathcal{P}_{\leq k}(K_n)$, where $\bar{a}_e = 0$ if $e \notin E(K_m)$ and $\bar{a}_e = a_e$ otherwise.*

Recall that all inequalities valid for $\mathcal{P}(K_n)$ are also valid for $\mathcal{P}_{\leq k}(K_n)$ for all $2 \leq k \leq n$. In fact, several of the inequalities from Section 7.2 remain facet-defining when turning from $\mathcal{P}(K_n)$ to $\mathcal{P}_{\leq k}(K_n)$. Sometimes restrictions on the relations between k and the inequality parameters apply. The following section contains a survey over several such results. Notice that none of the inequalities dealt with so far is violated by the origin (except for the clique inequalities (7.1b)). This is different for many cases of the hypermetric inequality, which is discussed in the next but one subsection.

7.3.1 Inequalities with right-hand side independent of k

Several of the facet-defining inequalities for $\mathcal{P}(K_n)$ are also facet-defining for $\mathcal{P}_{\leq k}(K_n)$.

Proposition 7.19 (Chopra and Rao [1993]). *For $3 \leq k \leq n$, the following inequalities define facets of $\mathcal{P}_{\leq k}(K_n)$:*

- *the triangle inequalities* (7.1a)
- *the q-wheel inequalities* (7.6) *if and only if $q \geq 3$ is odd*
- *the q-bicycle wheel inequalities* (7.7) *if and only if $q \geq 3$ is odd*

The clique-web inequalities quite often also remain facet-defining.

Proposition 7.20 (Deza *et al.* [1992]; Chopra and Rao [1995]). *For $k \geq 2$ and integers p, $q \geq 1$, $r \geq 0$ with $n = p+q$ and $p-q \geq 2r+1$, the following assertions hold with respect to $\mathcal{P}_{\leq k}(K_n)$:*

(i) For $k \geq 3$ and $p-q = kr+1$, the clique-web inequality (7.8) *defines a facet if $p \geq 2kr$ and $1 \leq r \leq k-2$.*

(ii) For $p-q = 2r+1$, the clique-web inequality (7.8) *defines a facet if $q \geq 2$.*

(iii) For $r \geq 1$ and $p-q \geq 2r+2$, the clique-web inequality (7.8) *defines a facet in case $\lceil (p-q)/(r+1) \rceil + 2 \leq k \leq n$, but does not define a facet in case $2 \leq k \leq \lceil (p-q)/(r+1) \rceil - 1$.*

(iv) If $p-q = k(r+1)$ with $r \geq 1$, then the clique-web inequality (7.8) *is not facet-inducing.*

(v) For $r = 0$ and $p-q \geq 2$, the clique-web inequality (7.8) *defines a facet if $k \geq p-q+2$, but it does not define a facet if $k \leq p-q-1$.*

antiweb inequality

The above results are cited from (i) Chopra and Rao [1995], Thm. 5.1 (dealing with the special case of a hypermetric inequality (7.12) called *antiweb inequality*); (ii) Deza and Laurent [1992b]; (iii) Deza *et al.* [1992], Thm. 1.17 (ii, iv); (iv) Deza *et al.* [1992], Thm. 1.17 (v); and (v) Deza *et al.* [1992], Thm. 1.20.

7.3.2 Hypermetric Inequalities and Special Cases

The hypermetric inequalities are introduced for the case $k = 2$ by Deza and Laurent [1992a] and generalized to $k \geq 2$ by Chopra and Rao [1995]. Their right-hand sides involve a peculiar function, depending on two integral parameters η and k, $\eta \geq 0$, $k \geq 1$:

$f_{hm}(\eta, k)$

$$\begin{aligned} f_{hm}(\eta, k) \;=\; & \binom{\eta \bmod k}{2} \left\lceil \frac{\eta}{k} \right\rceil^2 + \binom{k - \eta \bmod k}{2} \left\lfloor \frac{\eta}{k} \right\rfloor^2 \\ & + (\eta \bmod k)(k - \eta \bmod k) \left\lceil \frac{\eta}{k} \right\rceil \left\lfloor \frac{\eta}{k} \right\rfloor \end{aligned} \tag{7.11}$$

As usual, $\binom{a}{b} = 0$ in case $a < b$. If $\eta \leq k$, for example, then straightforward calculations show:

$$f_{hm}(\eta, k) = \binom{\eta}{2} \qquad (\eta \leq k)$$

Equivalently, $f_{hm}(\eta, k) = \max\left\{\sum_{1 \leq i < j \leq k} x_i\, x_j \mid \sum_{i=1}^{k} x_i = \eta, x_i \in \mathbb{Z}_+\right\}$ can be defined. This definition makes the connection to k-partitioning more explicit, but is inconvenient in several of our computations. The following facts are, however, obvious from this characterization.

Observation 7.21. *The function $f_{hm}(\eta, k)$ increases with k and strongly increases with η.*

The function $f_{hm}(\cdot, \cdot)$ appears as some kind of "correction term" in the right-hand side of the hypermetric inequality defined next.

Proposition 7.22 (Chopra and Rao [1995]). *Given $k \geq 2$ and a complete graph K_n and vertex weights $b_v \in \mathbb{Z}$ with $\eta = \sum_{v \in V(K_n)} b_v \geq 0$. The* hypermetric inequality

hypermetric inequality

$$\sum_{vw \in E(K_n)} b_v b_w\, z_{vw} \geq \sum_{vw \in E(K_n)} b_v b_w - f_{hm}(\eta, k) \tag{7.12}$$

is valid for $\mathcal{P}_{\leq k}(K_n)$.

The condition "$|\{v \in V \mid b_v > 0\}| \geq k$" given by Chopra and Rao [1995] in their Lemma 2.1 concerning the validity of the hypermetric inequality is not necessary, and it is, in fact, not used in their proof. The hypermetric inequality generalizes a number of previously known inequalities. The claw inequality (7.10) is one example. (The claw inequality was introduced later than the hypermetric inequality, but the previously known cases in which the hypermetric inequality defines a facet do not include that of the claw inequality.) Other examples are the clique inequality (7.1b) and the following general clique inequality.

general clique inequality

Proposition 7.23 (Chopra and Rao [1993]). *Consider a clique Q with $q = |Q| > k$ vertices in K_n. Then the* general clique inequality

$$\sum_{ij \in E(Q)} z_{ij} \geq \binom{\lfloor \frac{q}{k} \rfloor}{2}(k - q \bmod k) + \binom{\lceil \frac{q}{k} \rceil}{2}(q \bmod k) \tag{7.13}$$

is valid for $\mathcal{P}_{\leq k}(K_n)$ and facet defining if and only if q is not a multiple of k.

The general clique inequality (7.13) is obtained from the hypermetric inequality (7.12) by setting the weights of all vertices in Q to 1, and to 0 otherwise. A straightforward calculation shows that the right-hand sides of both inequalities are indeed the same. Moreover, the right-hand side of (7.13) is positive if and only if $q > k$. Hence, each of the general clique inequalities (7.13) separates the origin from $\mathcal{P}_{\leq k}(K_n)$.

There are other cases in which the right-hand side of the hypermetric inequality (7.12) is positive.

Proposition 7.24. *Given an integer $k \geq 2$ and a complete graph K_n, $k \leq n$, with vertex weights $b_v = \pm 1$ for all $v \in V(K_n)$ such that $S = \{v \in V(K_n) \mid b_v = -1\}$ and $T = \{v \in V(K_n) \mid b_v = 1\}$ satisfy $1 \leq |S| < |T|$. Then the hypermetric inequality* (7.12) *reads as follows:*

$$z(E(S)) + z(E(T)) - z([S,T]) \geq \binom{|S|}{2} + \binom{|T|}{2} - |S|\,|T| - f_{hm}(|T| - |S|, k) \tag{7.14}$$

A facet of $\mathcal{P}_{\leq k}(K_n)$ is defined if $(|T| - |S|) \bmod k \not\equiv 0$.

The inequality (7.14) *is valid for the origin if $|T| - 2|S| < k$. It is violated by the origin if $|T| - 2|S| \geq k + 1$. This bound is tight for $1 \leq |S| \leq k$ and can be relaxed for $|S| > k$.*

Theorem 2.1 of Chopra and Rao [1995] states that in case $|S| \geq 2$, $|T| \geq k$, $|T| \geq |S|$, and $(|T| - |S|) \bmod k \not\equiv 0$ the inequality (7.14) defines a facet. Proposition 7.24 relaxes the first two conditions so that the 2-partition inequality for $\mathcal{P}(K_n)$ and the triangle inequality for $\mathcal{P}_{\leq k}(K_n)$ are now covered, too. In the part of the proof concerned with facets, we apply the same technique as Chopra and Rao [1995], but a weaker version of their Lemma 2.2 suffices. Their full Lemma 2.2 is stated next, but only the case with the third set M_3 being empty is used later.

Lemma 7.25 (Chopra and Rao [1995]). *Let $a^T z \geq a_0$ be a valid inequality with respect to $\mathcal{P}_{\leq k}(K_n)$. Moreover, let $M_1, M_2, M_3 \subseteq V(K_n)$*

be three pairwise disjoint sets. Let $\mu_1 = (V_1, \ldots, V_r)$, $r \leq k$, be a partition of $V(K_n)$ such that $M_1, M_2 \subseteq V_1$ and $M_3 \subseteq V_2$ and such that the incidence vector $z(\mu_1)$ satisfies $a^T z(\mu_1) = a_0$ at equality. Three variations of the partition μ_1 are defined:

$$\begin{aligned}
\mu_2 &= (V_1 \setminus M_2, & V_2 \cup M_2, && V_3, \ldots, V_r)\\
\mu_3 &= (V_1 \cup M_3 \setminus M_1, & V_2 \cup M_1 \setminus M_3, && V_3, \ldots, V_r)\\
\mu_3 &= (V_1 \cup M_3 \setminus (M_1 \cup M_2), & V_2 \cup (M_1 \cup M_2) \setminus M_3, && V_3, \ldots, V_r)
\end{aligned}$$

If the incidence vectors $z(\mu_i)$, $i = 2, 3, 4$, also satisfy $a^T z(\mu_i) = a_0$, then

$$\sum_{u \in M_1, v \in M_2} a_{uv} = \sum_{v \in M_2, w \in M_3} a_{vw}.$$

In case $M_3 = \emptyset$, then

$$\sum_{u \in M_1, v \in M_2} a_{uv} = 0.$$

We now turn to the proof of Proposition 7.24.

Proof of Proposition 7.24. Consider any valid and facet-defining inequality $a^T z \geq a_0$ for $\mathcal{P}_{\leq k}(K_n)$ such that

$$\left\{ z \in \mathcal{P}_{\leq k}(K_n) \mid z \text{ satisfies (7.14) at equality} \right\} \subseteq \left\{ z \in \mathbb{R}^{\binom{n}{2}} \mid a^T z = a_0 \right\}$$

An incidence vector z of a partition attains equality in (7.14) if and only if z corresponds to a partition $V_1, \ldots, V_k$ of $V(K_n)$ with $b(V_i) \in \{\lfloor (|T| - |S|)/k \rfloor, \lceil (|T| - |S|)/k \rceil\}$ for all $i = 1, \ldots, k$ (see, e. g., the proof of Lemma 2.1 given by Chopra and Rao [1995]). Since we presuppose that $(|T| - |S|) \bmod k \not\equiv 0$, we may assume that $b(V_1) = \lceil (|T| - |S|)/k \rceil$ and $b(V_2) = \lfloor (|T| - |S|)/k \rfloor$ holds.

We first show that all entries in a corresponding to edges with both endpoints in T have the same value γ and that the entries corresponding to edges in the cut $[S, T]$ are $-\gamma$. In order to prove this, we may also assume without loss of generality that V_1 contains at least one vertex $s \in S$ and at least two vertices $t_1, t_2 \in T$. Let $M_1 = \{s, t_1\}$, $M_2 = \{t_2\}$, and $M_3 = \emptyset$. It is straightforward but technical to check that the incidence vectors of all three partitions as constructed in Lemma 7.25 satisfy (7.14) at equality and therefore $a^T z = a_0$ as well. We omit the details. Applying Lemma 7.25, we obtain $-a_{st_1} = a_{t_1 t_2}$. The roles of t_1 and t_2 can be played by any pair of distinct vertices in T. This allows to derive $a_{t_1 t_2} = \gamma$ for all $t_1, t_2 \in T$. Likewise, $a_{st} = -\gamma$ can be obtained for all $s \in S$ and $t \in T$.

In case S contains more than one element, it remains to show that $a_{s_1 s_2} = \gamma$ for all $s_1, s_2 \in S$. We consider the same partition as above, but exchange V_1 and V_2 so that now $b(V_1) = \lfloor(|T| - |S|)/k\rfloor$ and $b(V_2) = \lceil(|T| - |S|)/k\rceil$. This time, we may assume without loss of generality that V_1 contains two vertices $s_1, s_2 \in S$ and one vertex $t \in T$. With $M_1 = \{s_1, t\}$, $M_2 = \{s_2\}$, and $M_3 = \emptyset$ the incidence vectors of the partitions as constructed in Lemma 7.25 satisfy (7.14) at equality. As before, Lemma 7.25 allows us to conclude that $a_{s_1 s_2} = -a_{s_1 t}$ and, further on, $a_{s_1 s_2} = \gamma$ for all $s_1, s_2 \in S$.

Together with $\{z \in \mathcal{P}_{\leq k}(K_n) \mid z \text{ satisfies (7.14) at equality}\} \neq \emptyset$, this shows that $a^T z \geq a_0$ is a multiple of the hypermetric inequality (7.14). The scaling factor has to be positive, because both inequalities are valid for $\mathcal{P}_{\leq k}(K_n)$. Therefore, the hypermetric inequality (7.14) does indeed define a facet under the specified conditions.

Finally, the claims concerning the right-hand side of the inequality are immediate consequences of Proposition 7.26 below. □

Our next result addresses the right-hand side of the hypermetric inequality (7.12). We want to identify conditions under which the right-hand side is strictly positive. Instead of bounding the right-hand side from below, we only manage to bound it from above. For those cases, in which we show that the upper bound is attained, we can draw conclusions.

Proposition 7.26. *Given are an integer $k \geq 2$ and a complete graph K_n, $n \geq k$, together with integral vertex weights b_v, $v \in V(K_n)$, satisfying $\sum_{v \in V(K_n)} b_v \geq 0$. Let $\tau = \sum_{v: b_v > 0} b_v$ and $\sigma = -\sum_{v: b_v < 0} b_v$.*

(i) The following inequality is valid:

$$\sum_{vw \in E(K_n)} b_v b_w \leq \binom{\tau}{2} + \binom{\sigma}{2} - \tau\,\sigma$$

Equality holds in the inequality if and only if all positive weights differ by at most one and all negative weights differ by at most one.

(ii) The right-hand side of the hypermetric inequality (7.12) corresponding to the given vertex weights is bounded from above:

$$\sum_{vw \in E(K_n)} b_v b_w - f_{hm}(\tau - \sigma, k)$$

$$\leq k \binom{\lfloor \frac{\tau - \sigma}{k} \rfloor}{2} + \left\lfloor \frac{\tau - \sigma}{k} \right\rfloor (\tau - \sigma) \bmod k - \sigma$$

Equality holds under the same conditions as in (i).

(iii) Moreover, if

$$\tau - 2\sigma \geq \begin{cases} k+1, & \textit{for } 0 \leq \sigma \leq k, \\ k, & \textit{for } \sigma > k, \end{cases}$$

then

$$k\binom{\lfloor\frac{\tau-\sigma}{k}\rfloor}{2} + \left\lfloor\frac{\tau-\sigma}{k}\right\rfloor (\tau-\sigma) \bmod k - \sigma > 0.$$

In case $0 \leq \sigma \leq k$*, the condition* $\tau - 2\sigma \geq k+1$ *is also necessary.*

Proof. We prove the three parts separately.

Ad (i): Let $T = \{v \in V(K_n) \mid b_v > 0\}$, $S = \{v \in V(K_n) \mid b_v < 0\}$, then:

$$\begin{aligned} \sum_{vw \in E(K_n)} b_v b_w &= \overbrace{\sum_{vw \in E(T)} b_v b_w}^{\leq f_{hm}(\tau,\tau)} + \overbrace{\sum_{vw \in E(S)} b_v b_w}^{\leq f_{hm}(\sigma,\sigma)} + \sum_{vw \in [S,T]} b_v b_w \\ &\leq f_{hm}(\tau,\tau) + f_{hm}(\sigma,\sigma) + \big(\sum_{v \in T} b_v\big)\big(\sum_{w \in S} b_w\big) \\ &= \binom{\tau}{2} + \binom{\sigma}{2} + \tau\,(-\sigma) \end{aligned}$$

The inequality is fulfilled at equality if and only if the bounds on the two first terms are tight. This happens precisely if $\max\{|b_v - b_w| \mid v, w \in T\} \leq 1$ and $\max\{|b_v - b_w| \mid v, w \in S\} \leq 1$, see the definition (7.11) of $f_{hm}(\cdot,\cdot)$.

Ad (ii): We define $q = \lfloor\frac{\tau-\sigma}{k}\rfloor$ and $r = (\tau-\sigma) \bmod k$. (Recall that $\tau - \sigma \geq 0$ by hypothesis.) Two cases are distinguished.

If $r = 0$, then

$$f_{hm}(\tau-\sigma, k) = f_{hm}(qk, k) = \binom{0}{2} q^2 + \binom{k}{2} q^2 + 0 \cdot kqq = \binom{k}{2} q^2.$$

Using this equation and $\tau = \sigma + qk$, we derive the desired result:

$$
\begin{aligned}
&\sum_{vw \in E(K_n)} b_v b_w - f_{hm}(\tau - \sigma, k) \\
&\overset{(i)}{\leq} \binom{\sigma + qk}{2} + \binom{\sigma}{2} - (\sigma + qk)\,\sigma - f_{hm}(qk, k) \\
&= \binom{\sigma + qk}{2} + \binom{\sigma}{2} - (\sigma + qk)\,\sigma - \binom{k}{2} q^2 \\
&= \frac{1}{2}\big[(\sigma^2 + 2kq\sigma - k^2q^2 - \sigma - kq) + (\sigma^2 - \sigma) \\
&\quad - 2(\sigma^2 + kq\sigma) - (k^2q^2 - kq^2)\big] \\
&= \frac{1}{2}\left[-2\sigma + kq^2 - kq\right] \\
&= k\binom{q}{2} - \sigma
\end{aligned}
$$

If $r > 0$, the derivation is more involved but does not require additional insight. We therefore omit the details.

Ad (iii): Let $h(\tau, \sigma) = k\binom{\lfloor \frac{\tau - \sigma}{k} \rfloor}{2} + \lfloor \frac{\tau - \sigma}{k} \rfloor (\tau - \sigma) \bmod k - \sigma$, then

$$
\begin{aligned}
&h(k + 2\sigma, \sigma) \\
&= k\binom{\lfloor \frac{k+\sigma}{k} \rfloor}{2} + \left\lfloor \frac{k + \sigma}{k} \right\rfloor (k + \sigma) \bmod k - \sigma \\
&= k\binom{1 + \lfloor \frac{\sigma}{k} \rfloor}{2} + \left(1 + \left\lfloor \frac{\sigma}{k} \right\rfloor\right) (\sigma \bmod k) - \sigma \\
&= \begin{cases} k\binom{1}{2} + 1\,\sigma - \sigma, & \text{if } 0 \leq \sigma < k, \\ k\binom{2}{2} + 2 \cdot 0 - \sigma, & \text{if } \sigma = k, \\ k\binom{1+q}{2} + (1+q)r - (qk + r), & \text{if } \sigma > kq = \lfloor \frac{\sigma}{k} \rfloor, r = \sigma \bmod k. \end{cases}
\end{aligned}
$$

The result is zero in the first two cases. In the case of $\sigma > k$, we further deduce

$$
h(k + 2\sigma, \sigma) = kq + k\binom{q}{2} + r + qr - (qk + r) = k\binom{q}{2} + qr > 0,
$$

using the identity $\binom{q+1}{2} = \binom{q}{1} + \binom{q}{2}$. The last strict inequality holds because either $q = 1$ and $r > 0$ or $q \geq 2$. Finally, we observe that, for fixed σ, $h(\tau, \sigma)$ is strictly increasing with τ ($\geq \sigma$). Let $q = \lfloor \frac{\tau - \sigma}{k} \rfloor$ and $r = (\tau - \sigma) \bmod k$, then $h(\tau, \sigma) = k\binom{q}{2} + qr - \sigma$. For fixed q, this expression increases with r. Hence, the only problem may arise when p increases and

$\tau-\sigma$ becomes divisible by k. Due to $k\binom{q+1}{2} = k\binom{q}{2}+kq > k\binom{q}{2}+q(k-1)$, this is not the case. The sufficiency and the necessity of the conditions now follow as claimed. □

By Proposition 7.26 (ii), the right-hand side of (7.14) is equivalent to

$$-|S| + k\binom{\left\lfloor \frac{|T|-|S|}{k} \right\rfloor}{2} + \left\lfloor \frac{|T|-|S|}{k} \right\rfloor (|T|-|S|) \bmod k. \qquad (7.15)$$

Hence, the inequality (7.14) is equivalent to the 2-partition inequality (7.9) whenever $|T|-|S| < k$.

The hypermetric inequality (7.12) is also known to be facet-defining is several other cases, where the vertex weights are not restricted to ±1.

Proposition 7.27 (Chopra and Rao [1995]). *The hypermetric inequality (7.12) defines a facet of $\mathcal{P}_{\leq k}(K_n)$, $k \geq 3$, if one of the following conditions holds:*

(i) $2 \leq d \in \mathbb{Z}$, $R^+ = \{v \in V \mid b_v = 1\}$, $R^- = \{v \in V \mid b_v = -1\}$, $S^+ = \{v \in V \mid b_v = d\}$, $S^- = \{v \in V \mid b_v = -d\}$, $V = R^+ \cup R^- \cup S^+ \cup S^-$, $|S^+| = |S^-|$, $|R^+| \geq |R^-| \geq d$, $(|R^+| - |R^-|) \bmod k \not\equiv 0$

(ii) $2 \leq d \in \mathbb{Z}$, $R = \{v \in V \mid b_v = 1\}$, $S = \{v \in V \mid b_v = d\}$, $V = R \cup S$, $|S| \geq k$, $|R| \geq (k - |S| \bmod k)d + 1$, $|S| \bmod k \leq k-2$, $(|R| + d\,|S|) \bmod k \not\equiv 0$

(iii) $2 \leq d \in \mathbb{Z}$, $R = \{v \in V \mid b_v = 1\}$, $S = \{v \in V \mid b_v = d\}$, $V = R \cup S$, $|R| \geq (2k - |S|)d + 1$, $|S| < k$, $(|R| + d\,|S|) \bmod k \not\equiv 0$

Another example for the case of $k = n$ is the claw inequality (7.10). Under the conditions given in Proposition 7.27 (ii) the origin is not feasible, whereas it is feasible under the conditions given in (iii). This follows from Proposition 7.26.

7.3.3 Cycle Inequalities

For the sake of completeness, we also address the cycle inequalities here. The cycle inequalities are introduced for the case $k = 2$ by Deza and Laurent [1992a] and generalized to $k \geq 2$ by Chopra and Rao [1995]. Their left- and right-hand sides also involve the function $f_{hm}(\cdot,\cdot)$ defined in (7.11). This type of inequalities is even more complicated than the hypermetric inequalities (7.12), and it is not addressed in Chapter 8. The reason for mentioning them nevertheless is that under certain conditions the cycle inequalities also separate the origin from $\mathcal{P}_{\leq k}(K_n)$.

Proposition 7.28 (Chopra and Rao [1995]). *Given a complete graph K_n, $3 \leq k \leq n$, and $T \subseteq V(K_n)$ with $|T| \geq k$, let b_v, $v \in V(K_n)$, be integral vertex weights satisfying $\eta = \sum_{v \in V(K_n)} b_v \geq 1$, $T = \{v \in V(K_n) \mid b_v > 0\}$, $\lfloor \sum_{v \in V(K_n)} b_v / k \rfloor \geq 2$, and $(\sum_{v \in V(K_n)} b_v) \bmod k = 1$. Let C be a spanning cycle in the subgraph induced by T. The* cycle inequality

cycle inequality

$$\sum_{vw \in E(K_n)} b_v b_w \, z_{vw} - \big(f_{hm}(\eta, k) - f_{hm}(\eta, k-1)\big) \sum_{vw \in E(C)} z_{vw}$$
$$\geq \sum_{vw \in E(K_n)} b_v b_w - f_{hm}(\eta, k) - \big(|T| - k\big)\big(f_{hm}(\eta, k) - f_{hm}(\eta, k-1)\big) \tag{7.16}$$

is valid for $\mathcal{P}_{\leq k}(K_n)$.

A cycle inequality defines a facet of $\mathcal{P}_{\leq k}(K_n)$ if the following conditions are met: $|T| \geq 2k+1$, $b_v = p$ for some $p \in \mathbb{Z}_+$ and all $v \in T$, and $b_v = -1$ for all $v \in S = \{v \in V(K_n) \mid b_v < 0\}$.

Assume that $\eta = qk + 1$ with $0 \leq q \leq k-2$, then the following holds:

$$\begin{aligned}
& f_{hm}(\eta, k) - f_{hm}(\eta, k-1) \\
&= f_{hm}(qk+1, k) - f_{hm}(q(k-1)+q+1, k-1) \\
&= \left[\binom{1}{2} - \binom{q+1}{2}\right](q+1)^2 + \left[\binom{k-1}{2} - \binom{k-q-1}{2}\right]q^2 \\
&\quad + \Big[1\,(k-1) - (q+1)(k-q-1)\Big](q+1)q \\
&= \binom{q}{2} - kq
\end{aligned}$$

(Some intermediate steps are omitted.) Furthermore, assume that all positive vertex weights differ by at most one and that all negative vertex weights also differ by at most one, then, by Proposition 7.26 (ii), $\sum_{vw \in E(K_n)} b_v b_w - f_{hm}(\eta, k) = k\binom{q}{2} + 1\,q + \sum_{v \in S} b_v$. The right-hand side of the cycle inequality (7.16) reduces to:

$$k\binom{q}{2} + q + \sum_{v \in S} b_v - \big(|T| - k\big)\Big(\binom{q}{2} - kq\Big)$$

We now consider two sets of conditions under which the origin is not valid for the inequality (7.16), but a facet of $\mathcal{P}_{\leq k}(K_n)$ is defined.

First, fix two disjoint subsets S and T of $V(K_n)$ such that $|S| \leq k-2$ and $|T| = 2k+1+|S|$. Let $b_v = 1$ for all $v \in T$, $b_v = -1$ for all $v \in S$, and $b_v = 0$ otherwise. The corresponding cycle inequality defines a facet of $\mathcal{P}_{\leq k}(K_n)$. The right-hand side is:

$$\begin{aligned} & k\binom{2}{2} + 2 - |S| - (2k+1+|S|-k)(\binom{2}{2} - 2k) \\ &= k+2-|S| + (k+1+|S|)(2k-1) \\ &= (k+1)^2 + k^2 + 2(k-1)|S| \\ &> 0 \end{aligned}$$

Second, fix an integer p with $1 \leq p \leq \frac{k}{2} - 1$ and two disjoint subsets $S, T \subseteq V(K_n)$ such that $|T| = 2k+1$ and $|S| = p-1$. Let $b_v = p$ for $v \in T$, $b_v = -1$ for $v \in S$, and $b_v = 0$ otherwise. We have $\eta = p|T| - |S| = p(2k+1) - (p-1) = 2pk+1$. The corresponding cycle inequality defines a facet of $\mathcal{P}_{\leq k}(K_n)$. The right-hand side of (7.16) simplifies as follows under the given assumptions (with some derivations omitted):

$$\begin{aligned} & k\binom{2p}{2} + 2p - (p-1) - \big(2k+1-k\big)\big(\binom{2p}{2} - 2kp\big) \\ &= k\binom{2p}{2} + p + 1 - \big(k+1\big)\big(\binom{2p}{2} - 2kp\big) \\ &= 1 + 2p(k+1) + 2p(k^2 - p) \\ &> 0 \end{aligned}$$

Further details concerning the cycle inequality (7.16) are described by Chopra and Rao [1995].

7.4 Separating Violated Valid Inequalities

Several classes of valid and sometimes facet-defining inequalities for the polytope $\mathcal{P}_{\leq k}(K_n)$ are presented in the previous sections. We are now at the point where a cutting planes algorithm or a branch-and-cut algorithm on the basis of the known valid or even facet-defining inequalities for $\mathcal{P}_{\leq k}(K_n)$ could be developed in order to solve the Minimum k-Partition problem computationally. Cutting planes as well as branch-and-cut algorithms have been successfully applied to solve instances of numerous $\mathcal{NP}$-hard combinatorial optimization problems. For example, the already mentioned Concorde program, by Applegate *et al.* [1997], for the Traveling Salesman Problem is certainly among the most advanced branch-and-cut algorithms.

The reader not familiar with cutting planes and branch-and-cut algorithms may, for example, consult Jünger *et al.* [1995b], Nemhauser and

Wolsey [1988], or Schrijver [1986] for thorough introductions. Roughly speaking, the notion of a "cutting plane" can be explained as follows. We use the example of optimizing a linear objective function over $\mathcal{P}_{\leq k}(K_n)$. Instead of optimizing over the complete description of $\mathcal{P}_{\leq k}(K_n)$ in terms of linear inequalities (which is unknown for general n and k anyhow), the linear function is optimized over $[0,1]^{\binom{n}{2}}$ subject to some set of linear inequalities which are all valid for $\mathcal{P}_{\leq k}(K_n)$.

cutting plane

separate

separation

Assume the resulting optimal solution is a fractional vector z^0, then there exists another linear inequality which is valid for $\mathcal{P}_{\leq k}(K_n)$, but violated by z^0. Since an inequality (usually) defines a half-space delimited by an hyperplane, such an inequality is often called a *cutting plane*. If a cutting plane can be identified, it may be added to the present set of inequalities in order to "cut off" the vector z^0. This is also called to *separate* z^0 from the polytope $\mathcal{P}_{\leq k}(K_n)$. Generally, given a class $\mathcal{C}$ of inequalities and a vector z, the *separation problem* is to check whether all inequalities in $\mathcal{C}$ are satisfied by z and if not to produce at least one violated inequality out of $\mathcal{C}$.

cutting planes algorithm

The optimization process is iterated and possibly other cutting planes are added. If some z^i happens to be an integral vector, then this might be a vertex of $\mathcal{P}_{\leq k}(K_n)$. If so, z^i is optimal in $\mathcal{P}_{\leq k}(K_n)$ with respect to the given linear objective function. Otherwise, there again exists an inequality which is valid for $\mathcal{P}_{\leq k}(K_n)$ and is violated by z^i. We iterate with an appropriate inequality added. An algorithm employing this paradigm is called a *cutting planes algorithm*.

branch-and-cut algorithm

A *branch-and-cut algorithm* may also use "branching:" assume some z^i is fractional and l is one of the fractional coordinates, then two subproblems can be generated. In one of the subproblems, the lth coordinate is fixed to 0, in the other to 1. The vector z^i is infeasible in each of the two subproblems. Both subproblems have to be processed in order to determine which of the "branches" contains the better solution inside $\mathcal{P}_{\leq k}(K_n)$. Of course, more complex branching rules than just branching on a fractional variable can also be used.

We pursue neither of these approaches here. Our initial computational experiments were not encouraging. Among others, we experimented with the program developed by Ferreira, Martin, de Souza, Weismantel, and Wolsey [1996] for the Node Capacitated Graph Partition problem. This is an extension of the Minimum k-Partition problem, where each vertex is assigned a weight and there are upper limits on the weight a partite set may have. Their program, however, did not provide nontrivial lower bounds for the instances we tested. (Johnson, Mehrotra, and Nemhauser [1993] describe a branch-and-cut algorithm which, in addition, uses column generation for the Node Capacitated Graph Partition problem. We also like to mention the branch-and-cut algorithms developed for other related partition problems, which, however, do not comprise the Minimum k-Partition problem. See Grötschel and Wakabayashi [1989] for a branch-and-cut algorithms for the clique partition problem, and Jünger and Rinaldi [1998] for one for the Maximum Cut problem.) We are not aware of any branch-and-cut algorithms for the Minimum k-Partition problem itself.

Another reason not to pursue this further is that the semidefinite program (6.5) turned out to be an appealing alternative when it comes to prove lower bounds on the optimal value of Minimum k-Partition instances.

Nevertheless, we briefly address the computational complexity of finding violated inequalities in the course of a branch-and-cut algorithm. The ILP formulation (7.1) itself is huge for the instances we are interested in. Table 7.1 indicates the actual amount of the $3\binom{n}{3}$ (facet-defining) triangle constraints (7.1a) and $\binom{n}{k+1}$ (facet-defining) clique constraints (7.1b) for three of our test instances, see Chapter 5 and 6. Such vast amounts of constraints can hardly be handled at once by presently available LP-solver. A branch-and-cut algorithm would therefore have to separate violated clique constraints and probably also violated triangle constraints.

	n	k	(7.1a) → $3\binom{n}{3}$	(7.1b) → $\binom{n}{k+1}$
K	267	50	9,410,415	$2.2 \cdot 10^{55}$
B[1]	1971	75	3,822,685,515	$5.5 \cdot 10^{75}$
Sie1	930	43	400,882,080	$3.1 \cdot 10^{138}$

Table 7.1: Number of facet-defining inequalities for $\mathcal{P}_{\leq k}(K_n)$

Proposition 7.19 states that all triangle inequalities (7.1a) listed in our first integer linear programming formulation (7.1) for the Minimum k-Partition problem define facets of the polytope $\mathcal{P}_{\leq k}(K_n)$. There are $3\binom{n}{3}$ inequalities of that type so that the following is obvious.

Observation 7.29. *There exists an algorithms that checks in $\mathcal{O}(n^3)$ many steps whether a given rational vector $z \in [0,1]^{\binom{n}{2}}$, $n \geq 3$, fulfills all triangle constraints* (7.1a)*. If this is not the case, an inequality is returned that is violated by z.*

The same holds for the classes of odd wheel inequalities (7.6) and odd bicycle wheel inequalities (7.7). This is proven by Deza *et al.* [1992], basing on an argument of Gerards [1985]. Both types of inequalities define facets of $\mathcal{P}_{\leq k}(K_n)$ under fairly general conditions, see Proposition 7.19.

Proposition 7.30 (Deza *et al.* [1992]). *The following tasks can be accomplished by a polynomial time algorithm.*

- *Checking whether all q-wheel inequalities* (7.6)*, $q \geq 3$ and odd, are met by a given rational vector $z \in [0,1]^{\binom{n}{2}}$, $n \geq 3$, which meets all triangle inequalities* (7.1a)*. If not, a violated q-wheel inequality is presented.*
- *Checking whether all q-bicycle inequalities* (7.7)*, $q \geq 5$ and odd, are met by a given rational vector $z \in [0,1]^{\binom{n}{2}}$, $n \geq 3$, which meets all triangle inequalities* (7.1a)*. If not, a violated q-bicycle inequality is presented.*

Separating the class of clique inequalities (7.1b), however, is $\mathcal{NP}$-hard if k is considered as part of the input:

Proposition 7.31. *Given the complete graph K_n, $n \geq 3$, and a rational vector $z \in [0,1]^{\binom{n}{2}}$, decide whether the inequality*

$$\sum_{i,j \in Q} z_{ij} \geq 1$$

is met for all $Q \subseteq V(K_n)$ with $|Q| = k+1$. This problem is $\mathcal{NP}$-hard.

Proof. The proof is a simple reduction of the INDEPENDENT SET problem, see Garey and Johnson [1979, GT20], to the separation problem.

We are given a graph $G = (V, E)$ for which we want to know whether it contains an independent set of size $k+1$. Let $n = |V|$, and define $z \in [0,1]^{\binom{n}{2}}$ by $z_{ij} = 1$ if $ij \in E$ and $z_{ij} = 0$ otherwise. Then z violates the clique inequality $\sum_{i,j \in Q^*} z_{ij} \geq 1$ if and only if Q^* is an independent set of size $k+1$ in G. □

One may think of several heuristic ways to separate violated clique inequalities. One simple-minded example is the following. Given a rational vector $z \in [0,1]^{\binom{n}{2}}$, let $G_z = (V, E_z)$ denote the subgraph of K_n with $ij \in E_z$ if and only if $z_{ij} < \binom{k+1}{2}^{-1}$. For every clique in G_z of size $k+1$, a corresponding clique constraints is violated by z. Testing whether such a clique exists is, of course, also $\mathcal{NP}$-complete. But a simple greedy heuristic may often find such a clique as long as not very many clique constraints have been separated. Another example is the heuristic described by Krumke [1996, Section 5.5]. This heuristic is a polynomial time 2-approximation algorithm as long as the metric triangle inequalities ($z_{ij} + z_{jk} \geq z_{ik}$) are all satisfied by z.

Recall that the clique inequalities (7.1b) are a special case of the hypermetric inequalities (7.12). The complexity status of separating hypermetric inequalities is not yet fully settled in general. Deza and Laurent [1997, Section 28.4] discuss this issue and give references to related work, e. g., on heuristic approaches to separate hypermetric inequalities. To our best knowledge, the complexity of separating cycle inequalities (7.16) is not fully settled as well. The following holds in general.

Proposition 7.32. *Suppose there exists a class of polytopes $\mathcal{P}_{\mathcal{C}_{n,k}}$, $2 \leq k \leq n$, such that*

(i) $\mathcal{P}_{\leq k}(K_n) \subseteq \mathcal{P}_{\mathcal{C}_{n,k}} \subseteq [0,1]^{\binom{n}{2}}$ *for all* $2 \leq k \leq n$*;*

(ii) the inequality system $\mathcal{C}_{n,k}$ *defining* $\mathcal{P}_{\mathcal{C}_{n,k}}$ *is separable in polynomial time for all* $2 \leq k \leq n$*;*

(iii) there exists a $\delta > 0$ *for which* $\delta \min_{z \in \mathcal{P}_{\leq k}(K_n)} c^T z \leq \min_{z \in \mathcal{P}_{\mathcal{C}_{n,k}}} c^T z$ *holds for all* $c \in \{0,1\}^{\binom{n}{2}}$ *and all* $2 \leq k \leq n$*;*

then $\mathcal{P} = \mathcal{NP}$*.*

Proof. Let $G = (V, E)$ be a graph on n vertices. We define c^G by $c^G_{ij} = 1$ if $ij \in E$ and $c^G_{ij} = 0$ otherwise. The Minimum k-Partition problem associated with c^G has optimal value $\min_{z \in \mathcal{P}_{\leq k}(K_n)} (c^G)^T z = 0$ if and only if G is k-partite (or k-colorable), and at least 1 otherwise.

By assumption *(ii)*, the result of Grötschel *et al.* [1988, Theorem 6.4.9] concerning the use of a strong separation oracle for solving the strong optimization problem in oracle-polynomial time implies that any linear function can be optimized over $\mathcal{P}_{\mathcal{C}_{n,k}}$ in polynomial time. (Notice that $\mathcal{P}_{\mathcal{C}_{n,k}}$ is "well-described.")

Exploiting assumptions *(i)* and *(iii)*, we can therefore check in polynomial time whether the graph G is k-colorable. Since this problem is known to be $\mathcal{NP}$-complete, compare Garey and Johnson [1979, GT4], the assumptions taken together imply $\mathcal{P} = \mathcal{NP}$. □

Recall in this context from Corollary 7.17 that every valid inequality $a^T z \geq a_0$ with positive right-hand side has to have a support of size at least $\binom{k+1}{2}$. For $k = 50, 75, 43$, this is 1275 2850, and 946, respectively. In addition to the problems of identifying violated inequalities with such a large support, there is another potential problem source. If many such inequalities are separated in the course of a branch-and-cut procedure, this may lead to numerical problems in the LP-solver. With this in mind, we proceed to the next chapter. There, we argue that the semidefinite relaxation (6.5) of the MINIMUM K-PARTITION problem can be solved ε-approximately in polynomial time and that its set of feasible solutions can be seen as an reasonable approximation of the polytope $\mathcal{P}_{\leq k}(K_n)$.

CHAPTER 8

Semidefinite Relaxation of the Minimum k-Partition Problem

Lower bounds on the optimal solution of several MINIMUM K-PARTITION instances are reported in Section 6.3. We use these results to bound the unavoidable interference in a frequency assignment problem from below. The bounds are obtained from (approximately) solving the semidefinite relaxations (6.5) from Section 6.2.2. Semidefinite programming is the task of minimizing (or maximizing) a linear objective function over the convex cone of positive semidefinite matrices subject to linear constraints.

Here, we discuss the strength of the semidefinite relaxation. We relate the solution set of the semidefinite relaxation to the polytope $\mathcal{P}_{\leq k}(K_n)$ as defined in Section 7.1. This is done by considering a projection of an affine image of the solution set into $\mathbb{R}^{\binom{n}{2}}$. The image of the projection is called $\Theta_{k,n}$ and contains $\mathcal{P}_{\leq k}(K_n)$. We bound the extent to which the valid and often facet-defining hypermetric inequality (7.12) for $\mathcal{P}_{\leq k}(K_n)$ may be violated by points in $\Theta_{k,n}$. We prove that this bound is tight in several cases. We also show that, for the MINIMUM K-PARTITION problem, neither the LP relaxation of the ILP formulation (6.2)/(7.1) nor the SDP relaxation (6.5) is generally stronger than the other.

The chapter is organized as follows. We fix notation in Section 8.1. A short introduction to semidefinite programming is given in Section 8.2. We treat the semidefinite relaxation (6.5) of the MINIMUM K-PARTITION problem and its connection to the elliptope in Section 8.3. The relation between the polytope $\mathcal{P}_{\leq k}(K_n)$ and the set $\Theta_{k,n}$ is studied in Section 8.4. Finally, we state possible directions of further research in Section 8.5.

8.1 Preliminaries

We recall here basic properties of symmetric and positive (semi-)definite matrices, which are used in the following sections. See Appendix A for general notation.

symmetric and skew-symmetric matrices

A square matrix A is *symmetric* if the matrix is identical to its transpose, i. e., $A = A^T$. The set of $n \times n$-dimensional symmetric matrices is denoted by S_n. The matrix A is *skew-symmetric* if $A = -A^T$. The two sets of n-dimensional symmetric and of skew-symmetric matrices form orthogonal subspaces of $\mathbb{R}^{n\times n}$ of dimension $\binom{n+1}{2}$ and $\binom{n}{2}$, respectively. Every square matrix can be written uniquely as the sum of a symmetric and a skew-symmetric matrix:

$$A = \frac{A + A^T}{2} + \frac{A - A^T}{2}$$

The inner product of two matrices $A, B \in \mathbb{R}^{m\times n}$ is defined here as $\langle A, B\rangle = \sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij}B_{ij}$. For the inner product of a square matrix $A \in \mathbb{R}^{n\times n}$ and a symmetric matrix $X \in S_n$, the skew-symmetric part of A is irrelevant, because

$$\langle A, X\rangle = \langle \frac{A + A^T}{2}, X\rangle + \underbrace{\langle \frac{A - A^T}{2}, X\rangle}_{=0} = \langle \frac{A + A^T}{2}, X\rangle.$$

orthonormal

A matrix $P \in \mathbb{R}^{n\times n}$ is *orthonormal* if its column vectors $p_1, \dots, p_n$ satisfy $\|p_i\| = 1$ (they are unit vectors) and $\langle p_i, p_j\rangle = 0$ for $i \neq j$ (they are pairwise orthogonal). A diagonal matrix $D \in S_n$ is an *orthonormal diagonalization* of a matrix $A \in \mathbb{R}^{n\times n}$ if $D = P^T A\, P$ for some orthonormal matrix $P \in \mathbb{R}^{n\times n}$. The following result from linear algebra states that every symmetric matrix has an orthonormal diagonalization.

orthonormal diagonalization

Proposition 8.1 (orthonormal diagonalization). *Let $A \in S_n$. All eigenvalues of A are real. There exists an orthonormal matrix $P \in \mathbb{R}^{n\times n}$ such that $P^T AP = \Lambda_A$, where Λ_A is a diagonal matrix for which the entries are the eigenvalues of A. The column vectors of P are eigenvectors of A.*

An immediate consequence of the orthonormal diagonalization is that every symmetric matrix can be written as the sum of rank-one matrices.

Proposition 8.2. *Let $A \in S_n$, then $A = \sum_{i=1}^{n} \lambda_i(A) p_i {p_i}^T$, where $\lambda_i(A)$, $i = 1, \dots, n$ are the eigenvalues of A and the p_i's are associated eigenvectors.*

positive (semi-)definite

A symmetric matrix $A \in S_n$ is *positive semidefinite* or $A \succeq 0$, for short, if $x^T Ax \geq 0$ for all $x \in \mathbb{R}^n$. If, in addition, $x^T Ax > 0$ for all $x \neq 0$, then the matrix is *positive definite* or $A \succ 0$, for short.The subsets of S_n consisting of all positive semidefinite and positive definite matrices are denote here by S_n^+ and S_n^{++}, respectively. The following two propositions give known characterizations of positive semidefinite and positive definite matrices. S_n^+, S_n^{++}

Proposition 8.3. *Let A be a symmetric $n \times n$-matrix. The following properties are equivalent:*

(i) A is positive semidefinite.

(ii) All eigenvalues of A are nonnegative.

(iii) A can be written as the product of a matrix C of $\operatorname{rank}(A)$ *and its transpose, $A = CC^T$.*

(iv) $\langle A, B\rangle \geq 0$ for all matrices $B \in S_n^+$ (Fejer's Trace Theorem).

(v) $\det(A_{II}) \geq 0$ for every principal submatrix A_{II} of A.

Moreover, for each positive semidefinite matrix A holds:

(i) If $B \in S_n^+$ and $\langle A, B\rangle = 0$, then $AB = 0$.

(ii) A diagonal element is dominating all entries, $\exists i : A_{ii} = \max\{|A_{kl}| \mid 1 \leq k, l \leq n\}$, and if a diagonal element is zero, so are all entries in the corresponding row and column, i. e., $A_{ii} = 0$ implies $A_{ij} = 0$ for all j.

(iii) If $B \in \mathbb{R}^{n\times n}$ is a regular matrix, then $A \in S_n^+ \iff B^TAB \in S_n^+$.

Proposition 8.4. *Let A be a symmetric $n \times n$-matrix. The following properties are equivalent:*

(i) A is positive definite.

(ii) All eigenvalues of A are positive.

(iii) A is the product of a regular matrix C and its transpose, $A = CC^T$.

(iv) $\det(A_{I_jI_j}) > 0$, $j = 1, \ldots, n$ and $I_1 \subsetneq \cdots \subsetneq I_n$, for a nested sequence of principal submatrices.

Moreover, for each positive definite matrix A holds:

(i) If $B \in S_n^{++}$ is another positive definite matrix, then $\langle A, B\rangle > 0$.

(ii) If $B \in \mathbb{R}^{n\times n}$ is regular, then $A \in S_n^{++} \iff B^TAB \in S_n^{++}$.

(strictly) diagonally dominant

A matrix $A \in \mathbb{R}^{n \times n}$ is *diagonally dominant* if $|A_{ii}| \geq \sum_{j=1,j\neq i}^{n} |A_{ij}|$ holds for all $i = 1, \ldots, n$. In case the inequalities are all strictly fulfilled, then the matrix is *strictly diagonally dominant.* The following sufficient criteria for being positive (semi-)definite are direct consequences of Geršgorin's disc theorem.

Proposition 8.5. *Let $A \in S_n$ be diagonally dominant with nonnegative entries on the principal diagonal, then A is positive semidefinite. In case the principal diagonal is positive and A is strictly diagonally dominant, then A is positive definite.*

The trace of a square matrix is the sum of its eigenvalues. Bounds on the inner product of two positive semidefinite matrices are easily obtainable from this fact.

Proposition 8.6. *Let $A, B \in S_n^+$. Then $\langle A, B \rangle$ can be bounded from below and from above:*

$$\begin{aligned} \lambda_{min}(A)\lambda_{max}(B) \leq \lambda_{min}(A)\operatorname{tr}(B) & \\ \leq \langle A, B \rangle \leq & \\ & \lambda_{max}(A)\operatorname{tr}(B) \leq n\lambda_{max}(A)\lambda_{max}(B) \end{aligned}$$

Given a symmetric matrix for which it is known that some principal submatrix is positive definite, the following theorem states the necessary and sufficient condition under which the entire matrix is positive semidefinite.

Theorem 8.7 (Schur complement). *Let $A \in S_m^{++}$, $B \in \mathbb{R}^{m \times n}$, $C \in S_n$, then*

$$\begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \succeq 0 \iff C \succeq B^T A^{-1} B.$$

A partial characterization of the cone S_n^+ of positive semidefinite matrices is the following.

Proposition 8.8. *The set of positive semidefinite matrices S_n^+ is a full-dimensional, closed, and pointed cone in the vector space S_n of symmetric matrices. The positive definite matrices S_n^{++} are the interior of this cone.*

Proof (Folklore). Obviously, S_n^+ is a nonempty cone. In order to see that S_n^+ is pointed, pick any $A \in S_n^+$ other than the matrix containing only zeros. There exists a vector $x \in \mathbb{R}^n$ such that $x^T A x > 0$. Thus, $x^T(-A)x < 0$, and $-A \notin S_n^+$. Now, consider the set of symmetric matrices that have ones at positions (i,i), (j,j), (i,j) and (j,i) and zeros elsewhere for possibly equal i and j. There are $\binom{n+1}{2}$ such matrices, all of which are positive semidefinite and mutually linearly independent. Thus, S_n^+ is full-dimensional. Finally, a symmetric matrix is positive semidefinite if and only if all principal subdeterminants are nonnegative, and it follows from continuity that S_n^+ is a closed set. □

polar cone The *polar cone* C^* of a cone $C \subseteq \mathbb{R}^n$ is the set $C^* = \{y \in \mathbb{R}^n \mid \forall x \in C : \langle y,x\rangle \geq 0\}$. Hence, another way of stating Fejer's Trace Theorem, see Proposition 8.3, is to say that the polar cone of positive semidefinite matrices coincides with itself, that is, S_n^+ is *self-polar*. This fact is important in the duality theory of semidefinite programming. *self-polar*

Proposition 8.9. $S_n^+ = S_n^{+*}$.

We state the simple proof.

Proof (Folklore). $S_n^+ \subseteq S_n^{+*}$: Consider $A \in S_n^+$ and let $\Lambda_A = P\,A\,P^T$ be its eigenvalue decomposition with $PP^T = I_n$, see Proposition 8.1. Then, for every positive semidefinite $B \in S_n^+$,

$$\begin{aligned}\langle A,B\rangle &= \langle P\Lambda_A P^T, B\rangle = \langle P\Lambda_A, BP\rangle = \langle \Lambda_A, P^T BP\rangle \\ &= \sum_{i=1}^{n} \lambda_i(A)(P_{i\cdot})^T B\,P_{i\cdot} \geq 0,\end{aligned}$$

since $\lambda_i(A) \geq 0$ and $(P_{i\cdot})^T BP_{i\cdot} \geq 0$. Thus, $A \in S_n^{+*}$.

$S_n^+ \supseteq S_n^{+*}$: The square matrix xx^T of rank one is positive semidefinite all $x \in \mathbb{R}^n$. If $A \in S_n^{+*}$, then $x^TAx = \langle A, xx^T\rangle \geq 0$, and hence, $A \in S_n^+$. □

Given two convex sets F and C with $F \subseteq C$, the set F is called a *face* of C if $x,y \in C, \alpha \in]0,1[, \alpha x + (1-\alpha)y \in F \implies x,y \in F$. The cone S_n^+ has the following faces. *face*

Theorem 8.10. *Each face F of S_n^+ is one of the sets $\emptyset$, $\{0\}$ or $\{PWP^T \mid W \in S_k^+\}$ for some k, $1 \leq k \leq n$ and some $P \in \mathbb{R}^{n\times k}$ with $P^TP = I_k$.*

Every positive semidefinite matrix can be expressed as $\sum_{i=1}^n \lambda_i x_i {x_i}^T$ with $\lambda_i \geq 0$ according to Proposition 8.2. For each $x \in \mathbb{R}^n$ the set $\{\lambda xx^T \mid \lambda \geq 0\}$ is a face of S_n^+. None of these faces can be expressed as the convex combination of smaller faces of S_n^+. Hence, $\{X = xx^T \mid \|x\| = 1, x \in \mathbb{R}^n\}$ is a minimal generating system of S_n^+ (strictly speaking, a restriction like "the first nonzero coordinate of each x is positive" has to be added). This implies that the cone S_n^+ is not polyhedral, i.e., S_n^+ cannot be described as the intersection of finitely many hyperplanes for $n > 1$. Since the faces of S_n^+ have dimension $\binom{k+1}{2}$ for some k, there are gaps of more than one between the dimensions of nested faces.

Let A_i, $i = 1, \dots, m$, be symmetric matrices from S_n, then a *linear operator* $\mathcal{A}\colon S_n \to \mathbb{R}^m$ is defined via

$$X \mapsto \mathcal{A}X = \begin{bmatrix} \langle A_1, X\rangle \\ \vdots \\ \langle A_m, X\rangle \end{bmatrix}.$$

The *adjoint operator* $\mathcal{A}^T\colon \mathbb{R}^m \to S_n$ of $\mathcal{A}$ is given by $y \mapsto \sum_{i=1}^m y_i A_i$, and

$$\langle X, \mathcal{A}^T y\rangle = \langle \mathcal{A}X, y\rangle = \sum_{i=1}^m y_i \langle A_i, X\rangle = \langle \sum_{i=1}^m y_i A_i, X\rangle.$$

The following three simple results are used in Section 8.4.

Observation 8.11. *Let $X \in S_n^+$, $m > n$, and $J \subseteq \{1, \dots, m\}$ with $|J| = n$. There exists a matrix $\tilde{X} \in S_m^+$ such that $X = \tilde{X}_{JJ}$.*

Proof. Let $\tilde{X}$ be the identity matrix I_m with the submatrix I_{JJ} replaced by X. □

Notice that if X is positive definite, then the constructed matrix $\tilde{X}$ is also positive definite.

$D^{\alpha,\beta}(n)$

$E(m,n)$

In the course of our further calculations, matrices of a particular structure are of importance. Let $D^{\alpha,\beta}(n)$ denote the symmetric square matrix of order $n \geq 1$ with all entries on the principal diagonal equal to α and all other entries equal to β. Let $E(m,n) \in \mathbb{R}^{m \times n}$ be the matrix with all entries equal to 1. The following properties of $D^{\alpha,\beta}(n)$ are easily observed.

Proposition 8.12. *For $n \geq 1$, the determinant of $D^{\alpha,\beta}(n)$ is given by*

$$\det(D^{\alpha,\beta}(n)) = \big(\alpha - \beta\big)^{n-1} \big(\alpha + (n-1)\beta\big).$$

For $\beta \notin \{\frac{-\alpha}{n-1}, \alpha\}$ the matrix $D^{\alpha,\beta}(n)$ is regular and its inverse is

$$D^{\alpha,\beta}(n)^{-1} = \frac{1}{\big(\alpha - \beta\big)\big(\alpha + (n-1)\beta\big)} \cdot D^{\alpha+(n-2)\beta, -\beta}(n). \tag{8.1}$$

$D^{\alpha,\beta}(n)$ is positive semidefinite if and only if $\alpha \geq \beta \geq \frac{-\alpha}{n-1}$; it is positive definite if and only if strict inequality holds in both cases. (In case $n = 1$, $D^{\alpha,\beta}(n) = \big[\alpha\big]$ and β is assumed to be 0 in the above formulas. Moreover, the condition "$\beta \geq \frac{-\alpha}{n-1}$" becomes void.)

We are interested under which conditions on α, β, γ, δ, and ε the matrix

$$A = \begin{bmatrix} D^{\alpha,\beta}(s) & \gamma E(s,t) \\ \gamma E(t,s) & D^{\delta,\varepsilon}(t) \end{bmatrix} \tag{8.2}$$

is positive semidefinite. They can be derived by means of the Schur Complement, see Theorem 8.7.

Proposition 8.13. *Given integers $s, t \geq 1$, the matrix*

$$A = \begin{bmatrix} D^{\alpha,\beta}(s) & \gamma E(s,t) \\ \gamma E(t,s) & D^{\delta,\varepsilon}(t) \end{bmatrix}$$

is positive semidefinite if and only if $D^{\alpha,\beta}(s)$, $D^{\delta,\varepsilon}(t)$ are both positive semidefinite and $\big(\alpha + (s-1)\beta\big)\big(\delta + (t-1)\varepsilon\big) \geq st\gamma^2$ holds.

Proof. Clearly, the composite matrix cannot be positive semidefinite unless $D^{\alpha,\beta}(s)$ and $D^{\delta,\varepsilon}(t)$ are both positive semidefinite.

We first deal with the case of $D^{\alpha,\beta}(s)$ being positive definite. By the Schur Complement Theorem 8.7, A is positive semidefinite if and only if $D^{\delta,\varepsilon}(t) \succeq \gamma\, E(t,s)\, D^{\alpha,\beta}(s)^{-1}\, \gamma\, E(s,t)$. We compute the expression on the right-hand side of this inequality:

$$\begin{aligned} \gamma\, E(t,s)\, D^{\alpha,\beta}(s)^{-1}\, \gamma\, E(s,t) &= \frac{\gamma^2\, E(t,s)\, D^{\alpha+(s-2)\beta,-\beta}(s)\, E(s,t)}{\big(\alpha-\beta\big)\big(\alpha+(s-1)\beta\big)} \\ &= \frac{\gamma^2\; (\alpha-\beta)\; E(t,s)\, E(s,t)}{\big(\alpha-\beta\big)\big(\alpha+(s-1)\beta\big)} \\ &= \frac{\gamma^2\, s}{\alpha+(s-1)\beta}\, E(t,t) \end{aligned}$$

For $\omega = \frac{\gamma^2\, s}{\alpha+(s-1)\beta}$, $D^{\delta-\omega,\varepsilon-\omega}(t) = D^{\delta,\varepsilon}(t) - \gamma\, E(t,s)\, D^{\alpha,\beta}(s)^{-1}\, \gamma\, E(t,s)$. By Proposition 8.12, this matrix is positive semidefinite if and only if $\delta \geq |\varepsilon|$ and $\varepsilon - \omega \geq \frac{-(\delta-\omega)}{t-1}$. After resubstituting for ω, the latter condition reads as $\big(\alpha + (s-1)\beta\big)\big(\delta + (t-1)\varepsilon\big) \geq st\gamma^2$.

Finally, let $D^{\alpha,\beta}(s)$ be positive semidefinite. Then $D^{\alpha+\theta,\beta}(s)$ is positive definite for all $\theta > 0$. Let $A(\theta)$ denote the matrix A with α being replaced by $\alpha + \theta$. From what is proven so far, $A(\theta)$ is positive semidefinite if and only if $\big(\alpha + \theta + (s-1)\beta\big)\big(\delta + (t-1)\varepsilon\big) \geq st\gamma^2$ for all $\theta > 0$. The claim now follows from the fact that $S^{+}_{(s+t)}$ is closed. □

8.2 Semidefinite Programming

In a semidefinite program a linear objective function is minimized (or maximized) over the cone of positive semidefinite matrices subject to linear constraints. This cone is closed and convex but not polyhedral. The duality theory for semidefinite programming is not as smooth as that of linear programming. A gap between the optimal primal and dual objective function value is possible. This gap vanishes under a simple condition, which holds for the MINIMUM K-PARTITION relaxation.

We give a brief introduction to semidefinite programming here. More comprehensive treatments of this topic are, for example, given by Alizadeh [1995], Helmberg [2000], and can be found in the book edited by Wolkowicz, Saigal, and Vandenberghe [2000].

primal semidefinite program

The generic *primal semidefinite program* reads as follows:

$$\min\langle C, X\rangle \quad \text{s.t.} \quad \mathcal{A}X - b \in K, X \succeq 0 \qquad \text{(P-SDP)}$$

K is one of the following convex cones: $\{0\}^m$, $\mathbb{R}^m_+$, or $\{0\}^{m_1} \times \mathbb{R}^{m_2}_+$. This formulation is not entirely standard but has been used before. The appearance of the cone K may seem awkward at first. In the cases $K = \{0\}^m$, $K = \mathbb{R}^m_+$, and $K = \{0\}^{m_1} \times \mathbb{R}^{m_2}_+$ the corresponding semidefinite programs have equality constraints, inequality constraints, or a mixture of both. Hence, all cases of linear constraints can be represented adequately. Moreover, we will also have a nice formulation when turning to the dual program.

If the inner product in the objective function is spelled out and the effect of the operator $\mathcal{A}$ of X is written explicitly, then it is obvious that the objective and all restrictions are indeed linear. The only nonlinear ingredient is the condition "$X \succeq 0$."

What constraints does $X \succeq 0$ impose on the entries of X? We give some examples. Straight from the definition of positive semidefiniteness follows that all diagonal elements have to be nonnegative. Moreover, the absolute value of each off-diagonal element is bounded from above by the maximum of the diagonal elements in its row and column. A strengthening of this constraint is obtained by considering the determinant of 2×2 principal submatrices. Let $J = \{i, j\}$ with $1 \leq i, j \leq n$, then the submatrix $X_{JJ} = \begin{bmatrix} X_{ii} & X_{ij} \\ X_{ij} & X_{jj} \end{bmatrix}$ is itself positive semidefinite, and a short computation yields

$$\det(\begin{bmatrix} X_{ii} & X_{ij} \\ X_{ij} & X_{jj} \end{bmatrix}) \geq 0 \iff X_{ii}, X_{jj} \geq 0 \quad \text{and} \quad |X_{ij}| \leq \sqrt{X_{ii}X_{jj}}.$$

Hence, $|X_{ij}| \leq \sqrt{X_{ii}X_{jj}}$ has to hold for all i and j.

In order for a real, symmetric, 3×3-matrix to be positive semidefinite, it is necessary but not sufficient that all principal 2×2 submatrices are positive semidefinite, recall Proposition 8.3. We give an example for the insufficiency. The parameterized matrix

$$\begin{bmatrix} 1 & a & c \\ a & 1 & b \\ c & b & 1 \end{bmatrix}$$

has the determinant $1 + 2abc - a^2 - b^2 - c^2$. Hence, this matrix is positive semidefinite if and only if $-1 \leq a, b, c \leq 1$ and $1 + 2abc - a^2 - b^2 - c^2 \geq 0$. If we set $a = b = 1$ and $c = 0$, then all three 2×2 principal submatrices are positive semidefinite, but the determinant is -1 and the matrix itself is thus not in S_3^+. Similarly, the positive semidefiniteness of a matrix in S_n^+ can generally not be solely guaranteed by the fact that all its $(n-1) \times (n-1)$ principal submatrices are in S_{n-1}^+. We do not pursue this further and turn to the duality theory of semidefinite programming instead.

The dual semidefinite program is obtained by a standard Lagrangian approach:

$$\begin{aligned} \inf_{X \succeq 0} \langle C, X \rangle \quad \text{s.t.} \quad \mathcal{A}X - b \in K &= \inf_{X \in S_n^+} \sup_{y \in K^*} \langle C, X \rangle + \langle b - \mathcal{A}X, y \rangle \\ &\geq \sup_{y \in K^*} \inf_{X \in S_n^+} \langle C, X \rangle + \langle b - \mathcal{A}X, y \rangle \\ &= \sup_{y \in K^*} \inf_{X \in S_n^+} \langle b, y \rangle + \langle C - \mathcal{A}^T y, X \rangle \\ &= \sup_{y \in K^*} \langle b, y \rangle \quad \text{s.t.} \quad C - \mathcal{A}^T y \in S_n^{+*} \end{aligned}$$

Using that S_n^+ is self-dual, i.e., $\left(S_n^+\right)^* = S_n^+$, we may define the generic *dual semidefinite program* as follows:

dual semidefinite program

$$\max \langle b, y \rangle \quad \text{s.t.} \quad C - \mathcal{A}^T y \succeq 0, y \in K^* \qquad \text{(D-SDP)}$$

The dual cones of $K = \{0\}^m$, $K = \mathbb{R}_+^m$, and $K = \{0\}^{m_1} \times \mathbb{R}_+^{m_2}$ are $\mathbb{R}^m$, $\mathbb{R}_+^m$, and $\mathbb{R}^{m_1} \times \mathbb{R}_+^{m_2}$, respectively. In other words, a dual variable associated to an equality constraint is unrestricted, whereas a dual variable associated to an inequality constraint has to be nonnegative. This is in perfect accordance with what we know from linear programming.

Notice that in the same fashion as above, a more general optimization problem can be "dualized." One may replace the condition "$X \succeq 0$" (or, equivalently, "$X \in S_n^+$") by the condition $X \in L$ for any convex cone L, see Ben-Israel, Charnes, and Kortanek [1971], for example. Instead of the self-dual cone of semidefinite matrices, the polar cone L^* then appears in the constraint section of the dual program. Take $L = \mathbb{R}_+^n$ and $K = \{0\}^m$, for example, then the primal program is a linear program with equality constraints and nonnegative variables. The corresponding dual program has inequality constraints and its variables are unrestricted. Thus, classical weak LP-duality appears as special case.

Although (D-SDP) does not look like a semidefinite program at first, it is one nevertheless. The feasible sets of the primal and of the dual program are both intersections of an affine subspace with the semidefinite cone, see Nesterov and Nemirovskii [1994].

Weak duality holds as explained above.

Proposition 8.14 (Weak duality). *For a semidefinite program and its dual the following holds:*

$$\sup\{\langle b, y\rangle \mid C - \mathcal{A}^T y \succeq 0, y \in K^*\} \leq \inf\{\langle C, X\rangle \mid \mathcal{A}X - b \in K, X \succeq 0\}$$

with $\inf \emptyset = +\infty$ *and* $\sup \emptyset = -\infty$.

Strong duality does not always hold. Here is an example with a duality gap, taken from Vandenberghe and Boyd [1996].

Example 8.15 (Missing strong duality). *Consider*

$$\min\langle \begin{bmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, X\rangle \quad s.\,t. \quad \mathcal{A}X = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T, \; X \succeq 0$$

with

$$A_1 = \begin{bmatrix} 0 & \frac{-1}{2} & 0 \\ \frac{-1}{2} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, A_3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, A_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

The corresponding dual semidefinite program is

$$\max y_1 \quad s.\,t. \quad Z = C - \sum_{i=1}^{4} y_i A_i = \begin{bmatrix} -y_2 & \frac{y_1+1}{2} & -y_3 \\ \frac{y_1+1}{2} & 0 & -y_4 \\ -y_3 & -y_4 & -y_1 \end{bmatrix} \succeq 0.$$

Since the diagonal element Z_{22} is zero, the elements $Z_{12} = \frac{y_1+1}{2}$ and $Z_{32} = -y_4$ in the corresponding row and column have to be zero as well. Consequently, $y_1 = -1$, and the maximum of -1 is achieved by $y = \begin{bmatrix} -1 & 0 & 0 & 0 \end{bmatrix}^T$. The optimal value of the primal semidefinite program is 0. *Hence, there is a* duality gap *of 1.*

Conditions are known under which a duality gap does not occur.

Definition 8.16 (strictly feasible). (P-SDP) *is* strictly feasible *if there is a solution which is positive definite, i. e., if the interior of the set of solutions is nonempty. Analogously,* (D-SDP) *is strictly feasible if the interior of the set of dual solutions is nonempty.*

strictly feasible

Recall that K is one of the convex cones $K = \{0\}^m$, $K = \mathbb{R}^m_+$, or $K = \{0\}^{m_1} \times \mathbb{R}^{m_2}_+$. In these cases strong duality holds if the dual program is strictly feasible, see, e. g., Helmberg [2000]. Stoer and Witzgall [1970], for example, describe more general results on strong duality.

Theorem 8.17 (Strong duality). *Assume* (D-SDP) *is strictly feasible, then*

$$\inf\big\{\langle C, X\rangle \mid X \succeq 0, \mathcal{A}X - b \in K\big\} = \sup\big\{\langle b, y\rangle \mid y \in K^*, C - \mathcal{A}^T y \succeq 0\big\}.$$

If this value is finite, that is, in case the primal problem is feasible, then the optimal value is attained for some $X \succeq 0$ with $\mathcal{A}X - b \in K$.

The strict feasibility in the theorem is indeed necessary. Here is a folklore example to illustrate this.

Example 8.18. *Consider*

$$\min X_{11} \quad s.\,t. \quad X = \begin{bmatrix} X_{11} & 1 \\ 1 & X_{22} \end{bmatrix} \succeq 0$$

and its dual

$$\max 2y_1 \quad s.\,t. \quad \begin{bmatrix} 1 & -y_1 \\ -y_1 & 0 \end{bmatrix} \succeq 0.$$

The primal problem is strictly feasible, e. g., $X \succ 0$ for $X_{11} = X_{22} = 2$. The dual optimal solution 0 is attained for $y_1 = 0$, which is, in fact, the only feasible solution. Due to the condition $X \succeq 0$, X_{11} and X_{22} must satisfy $X_{11}, X_{22} \geq 0$ and $X_{11}X_{22} \geq 1$. Consequently, $X_{11} \geq \frac{1}{X_{22}}$. For $X_{22} \to \infty$, the right-hand side tends to 0. The primal optimum value is not attained.

Assume that we want to solve

$$\min \alpha\langle C, X\rangle + \beta \quad \text{s.t.} \quad \mathcal{A}X - b \in K, X \succeq 0, \tag{8.3}$$

where α, β are real numbers, $\alpha > 0$, then we may as well solve

$$\min\langle C, X\rangle \quad \text{s.t.} \quad \mathcal{A}X - b \in K, X \succeq 0, \tag{8.4}$$

multiply the optimal value by α and add β. The optimal solutions themselves are the same in both cases. A minor technical difference concerns the dual programs. In the former case, the dual is

$$\max\langle b, y\rangle + \beta \quad \text{s.t.} \quad \alpha C - \mathcal{A}^T y \succeq 0, y \in K^*,$$

whereas scaling and shifting the objective function of the dual to (8.4) yields

$$\max \alpha\langle b, \bar{y}\rangle + \beta \quad \text{s.t.} \quad C - \mathcal{A}^T \bar{y} \succeq 0, \bar{y} \in K^*.$$

Those programs are equivalent. This can be seen by substituting $\alpha\bar{y}$ for y in the first dual and canceling α in the constraints. (Recall that $\alpha > 0$ and that S_n^+ is a cone.)

Hence, in order to obtain a lower bound on the optimal value of the primal program (8.3), we may try to find a feasible solution to

$$\max\langle b, \bar{y}\rangle \quad \text{s.t.} \quad C - \mathcal{A}^T \bar{y} \succeq 0, \bar{y} \in K^*.$$

If $\hat{y}$ is such a feasible solution, then $\alpha\langle b, \hat{y}\rangle + \beta$ is a lower bound. This procedure is applied a number of times in Section 8.4, and we summarize it for reference.

Observation 8.19. *Let y be a feasible solution to*

$$\max\langle b, \bar{y}\rangle \quad \textit{s. t.} \quad C - \mathcal{A}^T \bar{y} \succeq 0, \bar{y} \in K^*.$$

Then $\alpha\langle b, y\rangle + \beta$ is a lower bound for

$$\min \alpha\langle C, X\rangle + \beta \quad \textit{s. t.} \quad \mathcal{A}X - b \in K, X \succeq 0.$$

Finally, we turn to the computational complexity of solving semidefinite programs. Such programs can in general not be solved in polynomial time. One reason is that the feasible set is not generally contained in a sufficiently bounded area around the origin. In fact, it is not even known whether testing the solution set for nonemptiness is in $\mathcal{NP}$ in the Turing model of computation, see Ramana [1997] and the references contained therein. If, however, the feasible set of the semidefinite program is known to be contained in the hypercube $[-1, 1]^{\binom{n+1}{2}}$, for example, then an optimal solution can be approximated with arbitrary precision in polynomial time. This is a consequence of the general theory on optimization over circumscribed convex bodies developed by Grötschel *et al.* [1988].

In order to make this more precise, let us recall what the weak optimization problem is. The definition relies on the following two notions. Given a convex set $K \subseteq \mathbb{R}^n$ and a real number $\varepsilon > 0$, the *ε-ball around K* is defined by

ε-ball around K

$$B\big(K,\varepsilon\big) = \big\{x \in \mathbb{R}^n \mid \|x - y\| \leq \varepsilon \text{ for some } y \in K\big\},$$

where $\|x - y\| = \sqrt{\langle x - y, x - y\rangle}$ is the Euclidean norm in $\mathbb{R}^n$, and the *interior ε-ball* is defined as

interior ε-ball

$$B\big(K,-\varepsilon\big) = \big\{x \in K \mid B\big(\{x\},\varepsilon\big) \subseteq K\big\}.$$

Definition 8.20 (Grötschel *et al.* [1988]). *An instance of the* weak optimization problem *consists of a compact and convex set K, a vector $c \in \mathbb{Q}^n$, and a rational number $\varepsilon > 0$. The task is to either*

weak optimization

- *find a vector $y \in \mathbb{Q}^n$ such that $y \in B\big(K,\varepsilon\big)$ and $\langle c, x\rangle \leq \langle c, y\rangle + \varepsilon$ for all $x \in B\big(K,-\varepsilon\big)$, or*
- *assert that $B\big(K,-\varepsilon\big)$ is empty.*

A simple adaptation of the proof of Theorem 9.3.30 by Grötschel *et al.* [1988] shows that for every fixed $\varepsilon > 0$ the weak optimization problem for a semidefinite program is solvable in a running time which is polynomially bounded in the two parameters n and R. Here, n is the dimension of the space, and R is the radius of a ball around the origin which is known to contain an optimal solution.

The hypercube $[-1,1]^{\binom{n+1}{2}}$ is contained in $B\big(\{0\},n\big)$, and the Corollary 4.2.7 of Grötschel *et al.* [1988], concerning the use of a weak separation oracle for efficiently solving the weak optimization problem, immediately yields the following.

Proposition 8.21. *Let $F \subseteq [-1,1]^{\binom{n+1}{2}}$ be the feasible set of a semidefinite program with m linear constraints. Then, for every fixed $\varepsilon > 0$, the weak optimization problem over F can be solved in polynomial time in m and n.*

If ε is considered part of the input, then the running time of the ellipsoid method, used for the proof of Corollary 4.2.7 by Grötschel *et al.* [1988], depends exponentially on the coding length of ε.

In the above mentioned counterexample for the general solvability of semidefinite programs in polynomial time the radius R grows exponentially in the size of the problem formulation.

8.3 The Minimum k-Partition Relaxation

In Section 6.2.2, the program (6.4) is stated as an alternative formulation of the MINIMUM K-PARTITION problem and the program (6.5) is its semidefinite relaxation. After having elaborated on the basics of semidefinite programming, we now come back to this relaxation. We study the semidefinite program (6.5) from a particular point of view. Namely, we relate the set of feasible solutions to the semidefinite program to the k-partition polytope $\mathcal{P}_{\leq k}(K_n)$ as defined through (6.4) in Section 6.2.2, see also (7.1) in Section 7.1.

We briefly recall how the semidefinite program (6.5) is obtained. Given are a complete graph K_n on n vertices together with an edge weighting $w\colon E(K_n) \to \mathbb{R}$ and an integer k, $2 \leq k \leq n$. We assume for notational convenience that the vertex set of K_n is $\{1, \dots, n\}$ and that the edge set is $\big\{ij \mid 1 \leq i < j \leq n\big\}$. According to Lemma 6.4, we may pick a set U of k unit vectors $u_1, \dots, u_k \in \mathbb{R}^n$ such that $\langle u_i, u_j\rangle = \frac{-1}{k-1}$ for all pairs of distinct vectors. Let $T_k\colon \mathbb{R} \to \mathbb{R}$ be the affine transformation $x \mapsto \frac{k-1}{k}x + \frac{1}{k}$, mapping 1 onto 1 and $\frac{-1}{k-1}$ onto 0. With the vectors as labels, the MINIMUM K-PARTITION problem can be stated as follows:

$$\min_{\substack{\phi\colon V \to U \\ i \mapsto \phi_i}} \sum_{ij \in E(K_n)} w_{ij} T_k(\langle \phi_i, \phi_j\rangle) \tag{8.5}$$

The weight of an edge is accounted for if and only if its endpoints have the same label. Every such labeling of the vertex set defines a matrix $X = \big[\langle \phi_i, \phi_j\rangle\big]_{1 \leq i,j \leq n} \in \mathbb{R}^{n \times n}$with the following properties: (i) X is positive semidefinite; (ii) all entries on the principal diagonal of X are equal to 1; (iii) all off-diagonal entries of X are equal to $\frac{-1}{k-1}$ or 1; and (iv) X has rank at most k.

In case the last property is not enforced and the second last is relaxed to the request that all off-diagonal entries are between $\frac{-1}{k-1}$ and 1, an optimal matrix X can be found by solving the semidefinite program:

$$\begin{aligned}
&\min \frac{k-1}{2k}\langle W, X\rangle + \frac{1}{2k}\langle W, E\rangle \\
&\text{s. t.} \\
&\langle E^{ii}, X\rangle = 1 && \forall i = 1, \dots, n \\
&\langle E^{ij}, X\rangle \geq \frac{-1}{k-1} && \forall i, j \in \{1, \dots, n\}, i < j \\
&X \succeq 0
\end{aligned} \tag{8.6}$$

This is the same semidefinite program as (6.5) in Section 6.2.2, only stated in a different way. Here, W denotes the symmetric matrix obtained from the vector $w \in \mathbb{R}^{\binom{n}{2}}$ by letting $W_{ii} = 0$ and $W_{ij} = w_{ij}$ for all $1 \leq i < j \leq n$. Moreover, $E^{ij}(n)$ denotes the symmetric $n \times n$-matrix with an 1-entry at positions (i, j) and (j, i) and zeros elsewhere. If the dimension is clear from the context we simply write E^{ij}. Let

E^{ij}

$$\Psi_{k,n} = \left\{X \in S_n^+ \mid X_{ii} = 1, X_{ij} \geq \frac{-1}{k-1}, i, j \in \{1, \ldots, n\}\right\} \tag{8.7}$$

denote the set of feasible solutions of (8.6), then this semidefinite program can be rewritten as

$$\min \frac{1}{2} \langle W, T_k(X) \rangle \quad \text{s.t.} \quad X \in \Psi_{k,n}.$$

Since all the diagonal elements of elements $X \in \Psi_{k,n}$ are equal to 1, all off-diagonal elements are confined to take values between $\frac{-1}{k-1}$ and 1. Thus, $\Psi_{k,n}$ is contained in the hypercube in S_n with vertex coordinates from $\{-1, 1\}$, and Proposition 8.21 implies that the semidefinite program (8.6) is ε-approximately solvable in polynomial time.

polynomial time

In the case of $k = 2$, $\Psi_{k,n}$ is the *elliptope*

elliptope

$$\mathcal{E}_n = \left\{X \in S_n^+ \mid X_{ii} = 1, i \in \{1, \ldots, n\}\right\}.$$

For $k > 2$, $\Psi_{k,n}$ is obtained by intersecting the elliptope $\mathcal{E}_n$ with the half-spaces defined by $X_{ij} \geq \frac{-1}{k-1}$ for all $i, j \in \{1, \ldots, n\}$. Projections of the elliptope $\mathcal{E}_3$ and the truncated elliptope $\Psi_{3,3}$ on the set of upper triangular matrix entries are depicted in Figure 8.1. The elliptope is studied extensively in the literature in terms of the following notions.

A boundary point A_0 of the nonpolyhedral, but convex $\mathcal{E}_n$ is an *extreme point* if $\{A_0\}$ is a face of $\mathcal{E}_n$; it is called a *vertex* if the cone of normal vectors to the hyperplanes supporting $\mathcal{E}_n$ at A_0 is full-dimensional. Moreover, we denote the smallest face of $\mathcal{E}_n$ containing A by $F_{\mathcal{E}_n}(A)$. The following characterizations are taken from Deza and Laurent [1997], but most of them are originally due to Laurent and Poljak [1995, 1996b].

extreme point
vertex

Theorem 8.22. *The elliptope $\mathcal{E}_n$ has the following properties.*

(i) The vertices of $\mathcal{E}_n$ are the matrices xx^T for $x \in \{\pm 1\}^n$. There are 2^{n-1} many of them.

(ii) Let A be a boundary point of $\mathcal{E}_n$ with $\operatorname{rank}(A) = r$ and let $l = \dim(F_{\mathcal{E}_n}(A))$, then

$$\max\left(0, \binom{r+1}{2} - n\right) \leq l \leq \binom{r}{2}.$$

Furthermore, for all integers $r, l \geq 0$ satisfying the above inequality, a boundary point A of $\mathcal{E}_n$ with rank r and $l = \dim(F_{\mathcal{E}_n}(A))$ exists.

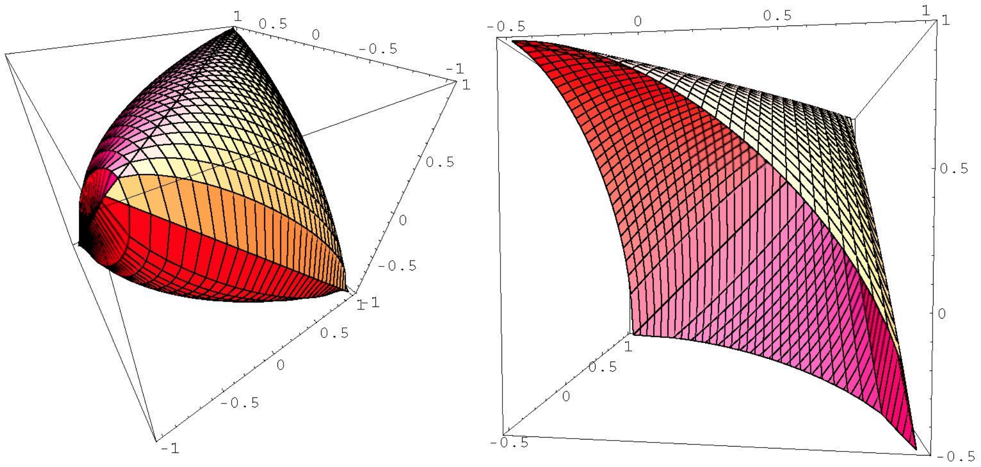


Figure 8.1: $\mathcal{E}_3$ and $\Psi_{3,3}$

(iii) Let F be a polyhedral face of $\mathcal{E}_n$ with dimension k, then $\binom{k+1}{2} \leq n-1$. Conversely, for every integer $k \geq 1$ satisfying $\binom{k+1}{2} \leq n-1$ exists a k-dimensional polyhedral face of $\mathcal{E}_n$.

(iv) Given $b \in \mathbb{R}^n$, the optimal value of the maximization problem

$$\max\langle bb^T, X\rangle \quad s.\,t. \quad X \in \mathcal{E}_n$$

is attained at a vertex of the elliptope if and only if

- $\min_{S\subseteq\{1,\ldots,n\}}\left|\sum_{i\in S} b_i - \sum_{j\in\bar{S}} b_j\right| = 0$ *and* $|b_i| \leq \sum_{j\neq i}|b_j|$ *for all* $i = 1,\ldots,n$; *or*
- $|b_i| > \sum_{j\neq i}|b_j|$ *for some* $i \in \{1,\ldots,n\}$.

balanced vector
gap of a vector

A vector $b \in \mathbb{R}^n$ satisfying $|b_i| \leq \sum_{j\neq i}|b_j|$ for all $i = 1,\ldots,n$ is called *balanced*, and the quantity $\min_{S\subseteq\{1,\ldots,n\}}\left|\sum_{i\in S} b_i - \sum_{j\in\bar{S}} b_j\right|$ is also known as the *gap* $\gamma(b)$ of b in the literature.

The next section deals with the relation between the positive semidefinite relaxation (8.6) of the Minimum k-Partition problem and the integer linear programming formulation (6.2) given in Section 6.2.2, see also (7.1) in Section 7.1. For this purpose, the set of feasible solutions to (8.6) is mapped injectively from $\mathbb{R}^{\binom{n+1}{2}}$ to $\mathbb{R}^{\binom{n}{2}}$ in such a way that the objective function values are preserved. The image of this injection is called $\Theta_{k,n}$. We study relations between $\Theta_{k,n}$ and $\mathcal{P}_{\leq k}(K_n)$.

The affine transformation T_k is extended from $\mathbb{R}$ to S_n (which is isomorphic to $\mathbb{R}^{\binom{n+1}{2}}$) by letting

$$T_k \colon S_n \to S_n, S \mapsto \frac{k-1}{k} S + \frac{1}{k} E(n,n).$$

(Recall that $E(n,n)$ is the $n \times n$ matrix with all entries being equal to one.) Which dimension applies will be clear from the context. Let

$$\zeta_{k,n} \colon S_n \to \mathbb{R}^{\binom{n}{2}}, \; X \mapsto \zeta_{k,n}(X) = z \quad \text{with} \quad z_{ij} = \big(T_k(X)\big)_{ij} \quad \text{for } i < j,$$

and consider

$\Theta_{k,n}$

$$\Theta_{k,n} = \zeta_{k,n}(\Psi_{k,n}) = \big\{\zeta_{k,n}(X) \mid X \in \Psi_{k,n}\big\}. \tag{8.8}$$

The restriction of $\zeta_{k,n}$ onto $\Psi_{k,n}$ is one-to-one and $\zeta_{k,n}|_{\Psi_{k,n}} \colon \Psi_{k,n} \to \Theta_{k,n}$ is an affine bijection. Moreover, for any given $X \in \Psi_{k,n}$ and any given $w \in \mathbb{R}^{\binom{n}{2}}$ the identity

$$\frac{1}{2}\langle W, T_k(X)\rangle = \langle w, \zeta_{k,n}(X)\rangle$$

holds,where W is again the symmetric matrix obtained from w by letting $W_{ii} = 0$ and $W_{ij} = w_{ij}$ for all $1 \leq i < j \leq n$.

A direct consequence of our definitions is as follows.

Observation 8.23. *The optimization problems*

$$\min \frac{1}{2}\langle W, T_k(X)\rangle \; s.\,t. \; X \in \Psi_{k,n} \quad \textit{and} \quad \min\langle w, z\rangle \; s.\,t. \; z \in \Theta_{k,n}$$

are equivalent.

The affine image $\Theta_{k,n}$ of the truncated elliptope $\Psi_{k,n}$ contains the polytope $\mathcal{P}_{\leq k}(K_n)$ and is itself contained in the hypercube $[0,1]^{\binom{n}{2}}$.

Proposition 8.24. *For every $k \geq 2$ and $n \geq 3$ with $k \leq n$, the set $\Theta_{k,n}$ is convex, and*

$$\mathcal{P}_{\leq k}(K_n) \subseteq \Theta_{k,n} \subseteq [0,1]^{\binom{n}{2}}.$$

We call $\Theta_{k,n}$ a *semidefinite relaxation* of $\mathcal{P}_{\leq k}(K_n)$. A related connection between the MAXIMUM CUT polytope and the elliptope is observed by Laurent and Poljak [1995, Lemma 4.1]. See also their Theorem 2.5, which characterizes the vertices of the elliptope.

semidefinite relaxation

Proof of Proposition 8.24. The set $\Theta_{k,n}$ is obtained by projecting the previously scaled and translated convex set $\Psi_{k,n}$. As such, it is itself convex.

Let $V_1, \ldots, V_l$ be a partition of the vertex set $V(K_n)$ into $l \leq k$ many sets, and let $z \in \mathbb{R}^{\binom{n}{2}}$ be the characteristic vector of this partition, i. e., $z_{ij} = 1$ if i and j are in the same class and $z_{ij} = 0$ otherwise. Moreover, let $U = \{u_1, \ldots, u_k\}$ be a set of k unit vectors such that the scalar product for every pair of vectors is $\frac{-1}{k-1}$, see Lemma 6.4. Finally, let $\phi\colon V(K_n) \to U$ be the mapping that assigns each vertex in V_1 to u_1, each vertex in V_2 to u_2, and so on. The matrix $X = \left[\langle \phi_i, \phi_j \rangle\right]_{1 \leq i,j \leq n}$ is then contained in $\Psi_{k,n}$ and $\zeta_{k,n}(X) = z$. Consequently, $\mathcal{P}_{\leq k}(K_n) \subseteq \zeta_{k,n}(\Psi_{k,n}) = \Theta_{k,n}$.

Finally, we observe that every matrix $X \in \Psi_{k,n}$ satisfies $\frac{-1}{k-1} \leq X_{ij} \leq 1$ for all (off-diagonal) entries. The left-hand side is explicitly enforced by the corresponding conditions. The right-hand side is implicitly enforced by fixing the entries on the principal diagonal to 1. (Recall that the absolute value of every off-diagonal element in a positive semidefinite matrix is bounded from above by the maximum of the diagonal elements in its row and column.) Hence, $\zeta_{k,n}\colon S_n \to \mathbb{R}^{\binom{n}{2}}$ maps every $X \in \Psi_{k,n}$ onto a vector z with $0 \leq z_{ij} \leq 1$ for all $ij \in E(K_n)$. Thus, $\Theta_{k,n} = \zeta_{k,n}(\Psi_{k,n}) \subseteq [0,1]^{\binom{n}{2}}$. □

Moreover, $\Theta_{k,n}$ contains only integral points from $\mathcal{P}_{\leq k}(K_n)$.

Proposition 8.25. *Given integers k, n with $2 \leq k < n$, then $\Theta_{k,n}$ and $\mathcal{P}_{\leq k}(K_n)$ contain the same integral points.*

Proof. Let $\bar{z}$ be an integral (binary) vector in $\Theta_{k,n}$. If at all, $\bar{z}$ violates triangle constraints (7.1a) or clique constraints (7.1b) by an integral amount. Let $\bar{X}$ denote the preimage of $\bar{z}$ under the mapping $\zeta_{k,n}$. All entries of the positive semidefinite matrix $\bar{X}$ are either $\frac{-1}{k-1}$ or $+1$.

No triangle constraint (7.1a) is violated either, because such a violation would imply that $\bar{X}$ has one of the matrices

$$\begin{bmatrix} 1 & \frac{-1}{k-1} & 1 \\ \frac{-1}{k-1} & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 1 & \frac{-1}{k-1} \\ 1 & 1 & 1 \\ \frac{-1}{k-1} & 1 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & \frac{-1}{k-1} \\ 1 & \frac{-1}{k-1} & 1 \end{bmatrix}$$

as a principal submatrix. In any case, the determinant is $-\left(\frac{k}{k-1}\right)^2 < 0$. Hence, none of these matrices appears as a principal submatrix of $\bar{X}$.

According to Lemma 6.4, no subset Q of size larger than k can induce a submatrix $\bar{X}_{QQ}$ with all its off-diagonal elements equal to $\frac{-1}{k-1}$. Thus, at least one off-diagonal element in $\bar{X}_{QQ}$ equals 1 for each set Q of size $k+1$, and, consequently, no clique constraint (7.1b) is violated by $\bar{z}$. □

The following observation is used in some of the following proofs.

Observation 8.26. *For every matrix $C \in S_n$, the semidefinite programs*

$$\begin{aligned} &\min \sum_{1\le i,j\le n} C_{ij}X_{ij} \qquad s.\,t. \\ &\langle E^{ii}, X_{ii}\rangle = 1, \quad \langle E^{ij}, X_{ij}\rangle \ge \frac{-1}{k-1}, \quad \forall\, i,j \in \{1,\dots,n\}, i<j \\ &X \in S_n^+ \end{aligned} \tag{8.9}$$

and

$$\begin{aligned} &\max \sum_{i=1}^{n} y_{ii} \; - \sum_{1\le i<j\le n} \frac{y_{ij}}{k-1} \qquad s.\,t. \\ &C - \sum_{1\le i\le j\le n} y_{ij}\, E^{ij} \;\in\; S_n^+, \quad y_{ii} \in \mathbb{R}, \quad y_{ij} \in \mathbb{R}_+ \end{aligned} \tag{8.10}$$

are dual to each other and are both strictly feasible.

The dual variable associated to the primal constraint $\langle E^{ii}, X\rangle = 1$ is y_{ii} and that associated to the primal constraint $\langle E^{ij}, X\rangle \ge \frac{-1}{k-1}$ is y_{ij}.

Proof. The two programs are simply (P-SDP) and (D-SDP), see Section 8.2, specialized with particular constraints.

The identity matrix I_n is positive definite and fulfills all inequality constraints of (8.9) with strict inequality. Hence, it is in the relative interior of the solution space, and the first program is strictly feasible.

The vector $y \in \mathbb{R}^{\binom{n}{2}}$ with $y_{ii} = -\sum_{j=1}^{n}|C_{ij}|-n$ for all i and $y_{ij} = 1$ for all $i<j$ is a feasible dual solution. All sign restrictions on y are strictly met, and the matrix $C - \sum_{1\le i\le j\le n} y_{ij}\, E^{ij}$ is positive definite, because it is strictly diagonally dominant (see Proposition 8.5). Therefore, the program (8.10) is strictly feasible, too. □

8.4 The semidefinite relaxation $\Theta_{k,n}$ and $\mathcal{P}_{\le k}(K_n)$

As reported in Section 6.3, a nontrivial lower bound on the optimal value of Minimum k-Partition problem can often be obtained from solving the positive semidefinite relaxation (6.5)/(8.6). In search of an explanation of this, our approach is to bound the maximal possible violation of facet-defining inequalities for $\mathcal{P}_{\le k}(K_n)$ by points in $\Theta_{k,n}$. The results presented here are obtained in this context. We also show that neither the solution set of the LP relaxation of the MINIMUM K-PARTITION problem's ILP formulation (7.1) is generally contained in $\Theta_{k,n}$ nor vice versa.

While writing down this material, we became aware of the strong connection to the elliptope and the related work of Laurent and Poljak [1995, 1996a,b] as well as Laurent, Poljak, and Rendl [1997]. With their results in mind, some of our findings are more easily stated and sometimes also more easily proved. We also discovered that some of the results had been known before, in particular, Proposition 8.28. We give our original proofs, nevertheless. Sometimes, we indicate an alternative proof as well.

A close connection between the elliptope $\mathcal{E}_n$ and the polytope $\mathcal{P}_{\leq k}(K_n)$ is given by the following result, which characterizes $\mathcal{E}_n$ in terms of hypermetric inequalities, compare with Laurent and Poljak [1995].

Lemma 8.27.

$$\mathcal{E}_n = \Big\{X \in S_n \mid X_{ii} = 1 \textit{ for } i = 1, \ldots, n; \\ 2 \sum_{1 \leq i < j \leq n} b_i b_j X_{ij} \geq -\sum_{i=1}^{n} b_i^2 \quad \textit{for all } b \in \mathbb{Z}^n\Big\}$$

If X is positive semidefinite, then, in particular, $b^T X b \geq 0$ for all $b \in \mathbb{Z}^n$. The inequality in Lemma 8.27 is merely a reformulation of this. Conversely, $b^T X b \geq 0$ for all $b \in \mathbb{Z}^n$ implies that this also holds for all $b \in \mathbb{Q}^n$. Because $\mathbb{Q}^n$ is dense in $\mathbb{R}^n$, the latter implies that $b^T X b \geq 0$ for all $b \in \mathbb{R}^n$, i. e., X is positive semidefinite.

Recall from Section 7.3 that the hypermetric inequalities (7.12) are valid for $\mathcal{P}_{\leq k}(K_n)$ and that they are also facet-defining under certain conditions. If those inequalities are "shifted a little," they become valid for $\Theta_{k,n}$. The shift is obtained by changing the constant on the right-hand side of the inequalities. The necessary change is bounded by $\frac{k}{8}$, see Figure 8.2. Later, we state conditions under which this bound is tight and others under which the bound can be improved.

Proposition 8.28. *Given an integral $k \geq 2$ and an integral weight b_i for every vertex $i \in V(K_n)$. The inequality*

$$\sum_{ij \in E(K_n)} b_i b_j z_{ij} \geq \frac{1}{2k}\Big(\Big(\sum_{i \in V(K_n)} b_i\Big)^2 - k \sum_{i \in V(K_n)} b_i^2\Big) \tag{8.11}$$

is valid for $\Theta_{k,n}$.

Proof. Let $\tilde{b}$ be the edges weighting obtained by setting $\tilde{b}_{ij} = b_i b_j$ for integral vertex weights b_i. Moreover, let $\tilde{B}$ denote the symmetric matrix with $\tilde{b}$ on its off-diagonal positions and zeros on the principal diagonal.

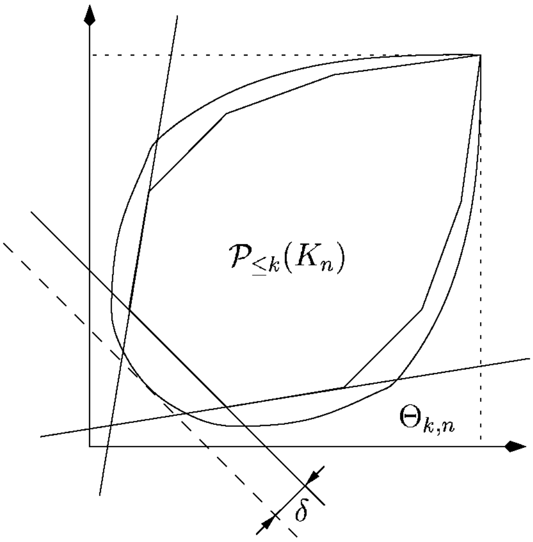


Figure 8.2: Shifting the hypermetric inequalities (7.12) for the k-partition polytope $\mathcal{P}_{\leq k}(K_n)$ by $\delta \leq \frac{k}{8}$ yields a valid inequality for $\Theta_{k,n}$.

Recall that solving

$$\min \sum_{ij \in E(K_n)} b_i b_j \, z_{ij} \text{ s.t. } z \in \Theta_{k,n} \quad \text{and} \quad \min \frac{1}{2}\langle \tilde{B}, T_k(X) \rangle \text{ s.t. } X \in \Psi_{k,n}$$

is equivalent. We argue that the optimal value of the minimization problem on the right is bounded from below by the right-hand side of (8.11). To this end, we show that $y \in \mathbb{R}^{\binom{n}{2}}$ with $y_{ii} = -b_i^2$ and $y_{ij} = 0$ if $i \neq j$ is feasible for the dual program (8.10) with $\tilde{B}$ in place of C. The claim then follows from the fact that the objective function value of y scaled by $\frac{k-1}{2k}$ and shifted by $\frac{1}{2k}\, 2 \sum_{ij \in E(K_n)} b_i b_j$ matches the right-hand side of (8.11), recall Remark 8.19.

First, the vector y as defined above is feasible: $\tilde{B} - \sum_{i=1}^{n} (-b_i^2)\, E^{ii} - \sum_{1 \leq i < j \leq n} 0 \cdot E^{ij} = bb^T \succeq 0$. Second, we have

$$\begin{aligned}
&\frac{k-1}{2k}\Big(\sum_{i \in V(K_n)} -b_i^2\Big) + \frac{2}{2k} \sum_{ij \in E(K_n)} b_i b_j \\
&= \frac{1}{2k}\Big(\sum_{i \in V(K_n)} b_i^2 + 2 \sum_{ij \in E(K_n)} b_i b_j - k \sum_{i \in V(K_n)} b_i^2\Big) \\
&= \frac{1}{2k}\Big(\big(\sum_{i \in V(K_n)} b_i\big)^2 - k \sum_{i \in V(K_n)} b_i^2\Big)
\end{aligned}$$

as desired. This completes the proof. □

The preceding proof uses a feasible solution to the dual program (8.10) in order to give lower bound on the optimal value of (8.9) with the primal cost matrix $C = bb^T$, $b \in \mathbb{Z}^n$. This dual solution y is zero in all entries corresponding to the constraints restricting the off-diagonal values. Hence, the obtained bound also holds without those restrictions, and, in fact, the same bound is obtained from Lemma (8.27) as follows.

The function $\zeta_{k,n}$ maps any matrix $X \in \Psi_{k,n}$ to a vector $z \in \Theta_{k,n}$ such that $X_{ij} = \frac{1}{k-1}(k\, z_{ij} - 1)$ holds for every $1 \leq i < j \leq n$. Plugging this into the inequality given in Lemma (8.27) yields

$$2 \sum_{1 \leq i < j \leq n} b_i b_j \frac{k\, z_{ij} - 1}{k-1} = \frac{2k}{k-1} \sum_{1 \leq i < j \leq n} b_i b_j z_{ij} - \frac{2}{k-1} \sum_{1 \leq i < j \leq n} b_i b_j \geq - \sum_{i=1}^{n} b_i^2 .$$

This is equivalent to the inequality given in Proposition 8.28.

The difference between the right-hand side of the hypermetric inequalities (7.12) and the right-hand side of (8.11) is bounded by a term that depends on the relation between the sum of the vertex weights b_i and k.

Proposition 8.29. *Given an integer $k \geq 2$ and an integral weight b_i for every vertex $i \in V(K_n)$. Then the difference between the right-hand side of the hypermetric inequality (7.12) for the polytope $\mathcal{P}_{\leq k}(K_n)$ and the right-hand side of the hypermetric inequality (8.11) for $\Theta_{k,n}$ is*

$$\Big(\big(\sum_{i \in V(K_n)} b_i\big) \bmod k\Big) \frac{k - (\sum_{i \in V(K_n)} b_i) \bmod k}{2k} .$$

This expression is bounded from above by $\frac{k}{8}$. The bound is attained if $(\sum_{i \in V(K_n)} b_i) \bmod k = \frac{k}{2}$.

Proof. Let $p = \left\lfloor (\sum_{i \in V(K_n)} b_i)/k \right\rfloor$ and $r = (\sum_{i \in V(K_n)} b_i) \bmod k$, then simply plugging these parameters into the right-hand side of (7.12) yields

$$\begin{aligned} & f_{hm}\Big(\sum_{i \in V(K_n)} b_i, k\Big) + \Big(\big(\sum_{i \in V(K_n)} b_i\big) \bmod k\Big) \frac{k - (\sum_{i \in V(K_n)} b_i) \bmod k}{2k} \\ &= f_{hm}(pk + r, k) + r \frac{k-r}{2k} . \end{aligned}$$

We expand this expression as follows:

$$
\begin{aligned}
& f_{hm}(pk + r, k) + r\frac{k-r}{2k} \\
&= \left(-\frac{k\,p^2}{2} + \frac{k^2\,p^2}{2} - \frac{r}{2} - p\,r + k\,p\,r + \frac{r^2}{2}\right) + \left(\frac{r}{2} - \frac{r^2}{2\,k}\right) \\
&= \frac{1}{2k}\left(-k^2\,p^2 + k^3\,p^2 - k\,r - 2\,k\,p\,r + 2k^2\,p\,r + k\,r^2 + k\,r - r^2\right) \\
&= \frac{1}{2k}\left(k^3\,p^2 + 2k^2\,p\,r + k\,r^2 - k^2\,p^2 - 2\,k\,p\,r - r^2\right) \\
&= \frac{(k-1)\,(p\,k+r)^2}{2k}
\end{aligned}
$$

The right-hand side of (8.11) is:

$$
\begin{aligned}
& \frac{1}{2k}\Big(\big(\sum_{i\in V(K_n)} b_i\big)^2 - k \sum_{i\in V(K_n)} b_i^2\Big) \\
&= \frac{1}{2k}\Big(2k \sum_{ij\in E(K_n)} b_i b_j - k\big(\sum_{i\in V(K_n)} b_i\big)^2 + \big(\sum_{i\in V(K_n)} b_i\big)^2\Big) \\
&= \sum_{ij\in E(K_n)} b_i b_j - \frac{(k-1)\left(\sum_{i\in V(K_n)} b_i\right)^2}{2k} \\
&= \sum_{ij\in E(K_n)} b_i b_j - f_{hm}\big(\sum_{i\in V(K_n)} b_i, k\big) \\
&\qquad - \big(\big(\sum_{i\in V(K_n)} b_i\big) \bmod k\big)\frac{k - \left(\sum_{i\in V(K_n)} b_i\right) \bmod k}{2k}
\end{aligned}
$$

The first part of the claim follows from this. As far as the second part is concerned, we observe that $r\frac{k-r}{2k}$ is a quadratic polynomial in r. Its maximum of $\frac{k}{8}$ is attained for $r = \frac{k}{2}$. This completes the proof. □

Other than the constraints on the variables to be binary, the integer linear programming formulation (7.1) linked to $\mathcal{P}_{\leq k}(K_n)$ contains only constraints on triangles and on cliques of size $k+1$. Both classes of constraints are facet-defining hypermetric inequalities for $\mathcal{P}_{\leq k}(K_n)$. Recall from Section 7.4 that the class of triangle inequalities (7.1a) can be separated in running time $\mathcal{O}(n^3)$ and that separating violated clique inequalities (7.1b) is $\mathcal{NP}$-hard if k is considered as part of the input. Recall also from Section 7.3 that every inequality separating the origin and the polytope $\mathcal{P}_{\leq k}(K_n)$ has a support of size at least $\frac{k(k+1)}{2}$. With this in mind, we observe the following.

Proposition 8.30. *Given the complete graph K_n and an integer k with $4 \leq k \leq n$, then for every $z \in \Theta_{k,n}$*

$$z_{ij} + z_{jl} - z_{il} \;\leq\; 1 + \frac{\sqrt{2(k-2)(k-1)} - (k-2)}{k} \quad \left[< \sqrt{2}\right] \tag{8.12}$$

holds for every triangle and

$$\sum_{ij \in E(Q)} z_{ij} \;\geq\; 1 - \frac{k-1}{2k} \quad \left[> \frac{1}{2}\right] \tag{8.13}$$

holds for every clique Q of size $k+1$ in K_n. Both bounds are tight.

In other words, all triangle inequalities (7.1a) and all clique constraints (7.1b) are "more than half satisfied" by every point in $\Theta_{k,n}$ in the sense that the violation is bounded by $\frac{1}{2}$ rather than by 1, which is worst possible. Both bounds are special cases of results to follow. A direct proof can be given by considering a principal 3×3-submatrix of $X \in \Psi_{k,n}$ in the first case and by applying Lemma 6.4 in the second case.

Before we continue investigating the relation between the polytope $\mathcal{P}_{\leq k}(K_n)$ and its semidefinite relaxation $\Theta_{k,n}$, we look at the relation between $\Theta_{k,n}$ and the solution set of the LP relaxation of the ILP formulation (7.1) to which $\mathcal{P}_{\leq k}(K_n)$ is associated. From Proposition 8.30 follows that $\Theta_{k,n}$ contains points which are infeasible for the LP relaxation of (7.1). Hence, $\Theta_{k,n}$ is not contained in the solution set of the LP relaxation. In general, the reverse inclusion does not hold either. In order to see this, we fix integers k and n such that $4 \leq k < \sqrt{n}$. Let $\tilde{z} \in \mathbb{R}^{\binom{n}{2}}$ be the vector with all coordinates equal to $\frac{1}{k+1}$. Then $\tilde{z}$ is feasible for the LP relaxation of (7.1) because $0 < \tilde{z}_{ij} < 1$ for all ij and $\tilde{z}$ satisfies all triangle inequalities (7.1a) as well as all clique inequalities (7.1b). The vector $\tilde{z}$ is, however, not contained in $\Theta_{k,n}$, because the valid inequality (8.11), with $b_i = 1$ for every vertex i, is violated by $\tilde{z}$:

$$\sum_{ij \in E(K_n)} \tilde{z}_{ij} = \binom{n}{2}\frac{1}{k+1} = \frac{n\,(n-1)}{2\,(k+1)} \;\not\geq\; \frac{n\,(n-k)}{2k} = \frac{1}{2k}\left(n^2 - kn\right)$$

This follows from $k\,(n-1) < (k+1)(n-k) \iff 0 < n - k^2$ and our assumption $\sqrt{n} > k$. In summary, the following holds.

Proposition 8.31. *Given two integers k and n with $4 \leq k < \sqrt{n}$, then neither $\Theta_{k,n}$ is contained in the solution set of the LP relaxation of* (7.1) *nor is the converse true.*

This is interesting, since the weak optimization problem over $\Theta_{k,n}$ can be solved in polynomial time, see Proposition 8.21, whereas solving the LP relaxation of (7.1) is $\mathcal{NP}$-hard, see Proposition 7.31.

We now turn back to studying relations between $\mathcal{P}_{\leq k}(K_n)$ and $\Theta_{k,n}$. The left-hand side of the inequality dealt with in the next proposition matches that of the general clique inequality (7.13), which is facet-defining for $\mathcal{P}_{\leq k}(K_n)$ if the size of the clique is larger than k but not an integer multiple of k.

Proposition 8.32. *Given the complete graph K_n and an integer k with $2 \leq k < n$. Let Q be a clique in K_n of size larger than k. Then*

$$\sum_{ij \in E(Q)} z_{ij} \geq \frac{|Q|}{2k}\big(|Q| - k\big) \tag{8.14}$$

is valid for $\Theta_{k,n}$ and there is a point $\bar{z} \in \Theta_{k,n}$ satisfying the inequality (8.14) at equality.

A proof using Lemma 6.4 is possible, but we give a more constructive argument using SDP duality theory.

Proof. The inequality (8.14) is obtained from (8.11) by setting $b_i = 1$ for all $i \in Q$ and $b_i = 0$ otherwise. Hence, it is valid for $\Theta_{k,n}$.

Let $q = |Q|$. We show that there exists a feasible solution to the optimization problem

$$\min \frac{1}{2}\langle D^{0,1}(q), T_k(X)\rangle \quad \text{s.t.} \quad X \in \Psi_{k,q}$$

with objective function value $\frac{q}{2k}(q-k)$. (See Proposition 8.12 for the definition of $D^{0,1}(q)$.) The claim then follows from Observation 8.11.

The matrix $D^{1,\frac{-1}{q-1}}(q)$ is primal feasible, because all its entries on the principal diagonal are equal to 1 and it is positive semidefinite, see Proposition 8.12. The corresponding objective function value is as desired:

$$\begin{aligned}
&\frac{1}{2}\langle D^{0,1}(q), T_k(D^{1,\frac{-1}{q-1}}(q))\rangle \\
&= \frac{k-1}{2k}\langle D^{0,1}(q), D^{1,\frac{-1}{q-1}}(q)\rangle + \frac{1}{2k}\langle D^{0,1}(q), E(q,q)\rangle \\
&= \frac{k-1}{2k}\,\frac{-q(q-1)}{q-1} + \frac{1}{2k}q(q-1) \\
&= \frac{1}{2k}(-kq + q + q^2 - q) \\
&= \frac{q}{2k}(q-k)
\end{aligned}$$

□

Under certain conditions on the relation among k, $|S|$, and $|T|$, a "shifted 2-partition inequality" is tight for $\Theta_{k,n}$.

Proposition 8.33. *Given the complete graph K_n, $n \geq 3$, and an integer k with $2 \leq k \leq n$. Let S and T be two nonempty, disjoint subsets of $V(K_n)$ with $|S| \leq |T|$. Then*

$$z(E(S)) + z(E(T)) - z([S,T]) \geq \frac{1}{2k}\Big(\big(|T| - |S|\big)^2 - k\big(|T| + |S|\big)\Big) \quad (8.15)$$

is valid for $\Theta_{k,n}$. Furthermore, there is a point $\bar{z} \in \Theta_{k,n}$ satisfying the inequality (8.15) *at equality if one of the following conditions holds:*

(i) $|S| = 1$ *and* $|T| \geq k - 1$*;*

(ii) $|S| \geq 2$, $|S| + |T| \leq k$ *and either* $|T| \leq |S|^2$, *or* $|T| > |S|^2$ *together with* $k \leq \frac{|T|^2 - |S|^2}{|T| - |S|^2}$.

Proof. The inequality (8.15) is obtained from (8.11) by setting

$$b_i = \begin{cases} +1, & i \in T, \\ -1, & i \in S, \\ \ \ 0, & \text{otherwise,} \end{cases}$$

and is thus valid for $\Theta_{k,n}$.

Let $s = |S|$, $t = |T|$, and let $C^{s,t}$ be the following symmetric matrix of dimension $(s+t) \times (s+t)$:

$$C^{s,t} = \begin{bmatrix} D^{0,+1}(s) & -E(s,t) \\ -E(t,s) & D^{0,+1}(t) \end{bmatrix}$$

(See Proposition 8.12 for the notation.) We show that feasible solutions exist to the optimization problem

$$\min \frac{1}{2}\langle C^{s,t}, T_k(X)\rangle \quad \text{s.t.} \quad X \in \Psi_{k,(s+t)}$$

with objective function value $\frac{(t-s)^2 - k(t+s)}{2k}$. The claim then follows from Observation 8.11. We give different primal feasible solutions for the cases $s = 1$ and $s \geq 2$.

In case $s = 1$, we assume $t \geq k - 1 (\geq 2)$ and let

$$X = \begin{bmatrix} 1 & \frac{1}{t}E(1,t) \\ \frac{1}{t}E(t,1) & D^{1,-1/t}(t) \end{bmatrix}.$$

The matrix X has only 1-entries on its principal diagonal and all of its off-diagonal elements are no less than $\frac{-1}{k-1}$. Moreover, X is positive semidefinite. This is verified by checking the conditions of Proposition 8.13:

$$1 \geq -\frac{1}{t} \geq \frac{-1}{t-1} \qquad \text{and} \qquad 1 + (t-1)\frac{-1}{t} = \frac{1}{t} = 1\,t\Big(\frac{1}{t}\Big)^2$$

Consequently, $X \in \Psi_{k,n}$. For later reference, we state that $\langle C^{1,t}, X\rangle = 2(-t)\frac{1}{t} + 2\binom{t}{2}\frac{-1}{t} = -2 - (t-1) = -(t+1) = -(t+s)$.

Now, the case $s \geq 2$ is considered. We assume that $k \geq s+t$ and that either $t \leq s^2$, or $t > s$ together with $k \leq \frac{t^2-s^2}{t-s^2}$ holds. Let $\alpha = \frac{t(k-t)-s(k-1)}{s(s-1)(k-1)}$, $\beta = \frac{-1}{k-1}$, $\gamma = \frac{k-t}{s(k-1)}$, and set

$$X = \begin{bmatrix} D^{1,\alpha}(s) & \gamma E(s,t) \\ \gamma E(t,s) & D^{1,\beta}(t) \end{bmatrix}.$$

We check the conditions given in Propositions 8.12 and 8.13 in order to prove that X is positive semidefinite. First, $D^{1,\alpha}(s)$ is positive semidefinite if $1 \geq \alpha \geq -\frac{1}{s-1}$, that is:

$$\begin{array}{rlclcl}
 & 1 & \geq & \alpha & \geq & -\dfrac{1}{s-1} \\
\iff & s(s-1)(k-1) & \geq & t(k-t) - s(k-1) & \geq & -s(k-1) \\
\iff & s^2(k-1) & \geq & t(k-t) & \geq & 0 \\
\overset{k\geq 2}{\iff} & s^2 & \geq & t\dfrac{k-t}{k-1} & \geq & 0
\end{array}$$

Both of the latter inequalities hold due to the conditions on the relations among s, t, and k. (This can be verified by case distinction.) Second, $D^{1,\beta}(t)$ is positive semidefinite if $1 \geq \beta \geq -\frac{1}{s-1}$, i.e., $1 \geq \frac{-1}{k-1} \geq \frac{-1}{t-1}$, which holds because $2 \leq t \leq k$. Finally,

$$\begin{aligned}
&\big(1 + (s-1)\alpha\big)\big(1 + (t-1)\beta\big) \\
&= \frac{s(k-1) + t(k-t) - s(k-1)}{s(k-1)}\;\frac{k-1-t+1}{k-1} \\
&= \frac{t(k-t)}{s(k-1)}\;\frac{k-t}{k-1} \geq st\,\frac{(k-t)^2}{s^2(k-1)^2} = st\gamma^2
\end{aligned}$$

also holds due to $s,t \geq 2$. Thus, X is positive semidefinite. Next, we show that $X \in \Psi_{k,n}$ by checking that the off-diagonal entries of X are no less than $\frac{-1}{k-1}$. This is obviously true for β and γ. For α, we distinguish the cases $t = s$ and $t > s$. If $t = s$, then $\alpha = \frac{-1}{k-1}$. If $t > s$, then:

$$\begin{aligned}
& \alpha = \frac{t(k-t) - s(k-1)}{s(s-1)(k-1)} \geq \frac{-1}{k-1} \\
\overset{k,s\,\geq\,2}{\Longleftrightarrow} \quad & t(k-t) - s(k-1) \geq -s(s-1) \\
\Longleftrightarrow \quad & k(t-s) \geq t^2 - s^2 \\
\overset{t\,>\,s}{\Longleftrightarrow} \quad & k \geq t + s
\end{aligned}$$

This holds by assumption.

Again, evaluating $\langle C^{s,t}, X \rangle$ yields:

$$\begin{aligned}
& \langle C^{s,t}, X \rangle \\
&= 2\binom{s}{2}\alpha + 2\binom{t}{2}\beta - 2st\gamma \\
&= s(s-1)\frac{t(k-t) - s(k-1)}{s(s-1)(k-1)} + t(t-1)\frac{-1}{k-1} - 2st\frac{k-t}{s(k-1)} \\
&= \frac{t(k-t) - s(k-1)}{k-1} - \frac{t(t-1)}{k-1} - \frac{2t(k-t)}{k-1} \\
&= \frac{tk - t^2 - sk + s - t^2 + t - 2tk + 2t^2}{k-1} \\
&= -(s+t)
\end{aligned}$$

Thus, in both cases the corresponding solution X yields $\langle C^{s,t}, X \rangle = -(s+t)$, and the resulting objective function values are:

$$\begin{aligned}
& \frac{k-1}{2k}\langle C^{s,t}, X \rangle + \frac{1}{2k}\langle C^{s,t}, E(s+t, s+t) \rangle \\
&= -\frac{k-1}{2k}(s+t) + \frac{t(t-1) + s(s-1) - 2st}{2k} \\
&= \frac{-ks - kt + s + t + t^2 - t + s^2 - s - 2st}{2k} \\
&= \frac{(t-s)^2 - k(t+s)}{2k}
\end{aligned}$$

This concludes the proof. □

The treatment of the case $|S| = 1$ in Proposition 8.33 is not fully satisfactory, because the most prominent representative of the 2-partition inequality, namely, the triangle inequality, is not covered. The case of $|S| = 1$ and $2 \leq |T| \leq k - 2$ is therefore considered separately.

Proposition 8.34. *Given the complete graph K_n and an integer k with $4 \leq k \leq n$. Let S and T be two disjoint subsets of $V(K_n)$ with $1 = |S| < |T| \leq k-2$. Then*

$$\begin{aligned} &z(E(S)) + z(E(T)) - z([S,T]) \\ &\quad \geq -1 - \frac{\sqrt{t(k-t)(k-1)} - (k-t)}{k} \qquad \left[> -\sqrt{t}\right] \end{aligned} \tag{8.16}$$

is valid for $\Theta_{k,n}$, and a point $\bar{z} \in \Theta_{k,n}$ fulfills (8.16) *at equality.*

In the proof, we again exhibit primal and dual solutions with matching objective function values. For the first time, however, the dual variables y_{ij} linked to the primal constraints $\langle E^{ij}, X\rangle \geq -1/(k-1)$ are positive.

Proof. We give solutions X and y to the dual programs (8.9) and (8.10), respectively, with matching objective function values for the primal cost matrix $C^{1,t}$ (see the proof of Proposition 8.33 for notation). We then compute $\frac{k-1}{2k}\langle C^{1,t}, X\rangle + \frac{1}{2k}\langle C^{1,t}, E(1+t, 1+t)\rangle$ and show that this is the desired value.

We first construct a solution y to the maximization problem (8.10). Let $a = \sqrt{\frac{k-1}{t(k-t)}}$. A short computation reveals that $0 < a \leq 1$ provided $1 \leq t \leq k-2$. Let $y_{11} = -\frac{1}{a}$, $y_{ii} = -a$ for $i = 2, \ldots, 1+t$, $y_{1j} = y_{j1} = 0$ for all $j = 2, \ldots 1+t$, and $y_{ij} = 1-a$ for all $i, j \in \{2, \ldots, 1+t\}, i < j$. The vector y is a feasible solution because $y_{ij} \geq 0$ for all $i < j$ and

$$C^{1,t} - \sum_{1 \leq i < j \leq n} y_{ij} E^{ij} = \begin{bmatrix} 1/a & -E(1,t) \\ -E(t,1) & D^{a,a}(t) \end{bmatrix} \succeq 0.$$

The latter is a direct consequence of Proposition 8.13.

The objective function evaluates to:

$$\begin{aligned} &\sum_{i=1}^{n} y_{ii} - \frac{1}{k-1}\sum_{i \neq j} y_{ij} \\ &= 1\left(-\sqrt{\frac{k-1}{t(k-t)}}\right)^{-1} + t\left(-\sqrt{\frac{k-1}{t(k-t)}}\right) - \frac{t(t-1)}{k-1}\left(1 - \sqrt{\frac{k-1}{t(k-t)}}\right) \\ &= -\sqrt{\frac{t(k-t)}{k-1}} - \underbrace{\frac{t(k-1) - t(t-1)}{k-1}\sqrt{\frac{k-1}{t(k-t)}}}_{=\sqrt{\frac{t(k-t)}{k-1}}} - \frac{t(t-1)}{k-1} \\ &= -2\sqrt{\frac{t(k-t)}{k-1}} - \binom{t}{2}\frac{2}{k-1} \end{aligned}$$

Next, we argue that the matrix

$$X = \begin{bmatrix} 1 & \sqrt{\frac{k-t}{t(k-1)}}\,E(1,t) \\ \sqrt{\frac{k-t}{t(k-1)}}\,E(t,1) & D^{1,-1/(k-1)}(t) \end{bmatrix}$$

is a primal feasible solution. Given that $1 < t \leq k-2$, all off-diagonal entries are at least as large as $\frac{-1}{k-1}$. By Proposition 8.13, X is positive semidefinite. We check the only condition that is not trivially fulfilled:

$$1\left(1 + (t-1)\frac{-1}{k-1})\right) = \frac{k-t}{k-1} \overset{!}{=} 1\,t\,\left(\sqrt{\frac{k-t}{t(k-1)}}\right)^2$$

The corresponding objective function value is

$$\langle C^{1,t}, X\rangle = -2t\sqrt{\frac{k-t}{t(k-1)}} - \binom{t}{2}\frac{2}{k-1} = -2\sqrt{\frac{t(k-t)}{k-1}} - \binom{t}{2}\frac{2}{k-1}.$$

Furthermore, we obtain the following for the dual transformed objective function value:

$$\begin{aligned} & \frac{k-1}{2k}\left(-2\sqrt{\frac{t(k-t)}{k-1}} - \binom{t}{2}\frac{2}{k-1}\right) + \frac{t(t-1)-2t}{2k} \\ = & -\frac{k-1}{k}\sqrt{\frac{t(k-t)}{k-1}} - \binom{t}{2}\frac{1}{k} - \frac{t}{k} + \binom{t}{2}\frac{1}{k} \\ = & -1 - \frac{\sqrt{t(k-t)(k-1)} - (k-t)}{k} \end{aligned}$$

The claims concerning the validity and tightness of the inequality (8.16) is thereby proven, compare Proposition 8.11 and Remark 8.19.

Finally, we show that $-\sqrt{t}$ bounds the above term from below. A straightforward application of l'Hôspital's rule yields that the expression $-1 - \frac{\sqrt{t(k-t)(k-1)}-(k-t)}{k}$ converges to $-\sqrt{t}$ as k goes to infinity. It remains to check that the value of the expression is bounded from below by $-\sqrt{t}$:

$$\begin{aligned} & -\sqrt{t} < -1 - \frac{\sqrt{t(k-t)(k-1)} - (k-t)}{k} \\ \iff & k\sqrt{t} - t > \sqrt{t(k-t)(k-1)} \\ \overset{k>t>1}{\iff} & tk^2 - 2tk\sqrt{t} + t^2 > t(k-t)(k-1) \quad [= tk^2 - t(t+1)k + t^2] \\ \overset{t>0}{\iff} & -2k\sqrt{t} > -t(t+1)k \\ \overset{k>0}{\iff} & \sqrt{t} < \frac{t+1}{2} \end{aligned}$$

The last inequality holds for all $t \geq 2$. This completes the proof. □

8.5 Summary and Outlook

The semidefinite program (8.6) is a relaxation of the combinatorial Minimum k-Partition problem. It is known for more than a decade that such a semidefinite program is, in principle, solvable in polynomial time. Merely within the last one or two years, however, SDP solvers have matured to the point, where the semidefinite programs associated to graphs of sizes in the order of a few hundred vertices become computationally tractable in practice. We are now in the position to solve our large semidefinite relaxations with a sufficient degree of accuracy in tolerable running times.

The lower bounds obtained in Section 6.3 on the optimal value of a minimal k-partition are higher than we had expected. The previously known computational studies on related problems like the Maximum Cut problem (with $k = 2$) or the graph partition problem with given sizes for the partite sets and with values for k up to 4, see Wolkowicz and Zhao [1999], can hardly give an indication of what to expect for the Minimum k-Partition problem for values of k between 39 and 76.

We attribute the strength of the bound to a large extent to the "shifted hypermetric inequalities" (8.11), which are implicit in the semidefinite relaxation. Consequently, the solutions to the semidefinite relaxation always fulfill at least partially every single valid (and often facet-defining) hypermetric inequality (7.12) for the polytope $\mathcal{P}_{\leq k}(K_n)$. In particular, with respect to the ILP formulation (7.1) of the Minimum k-Partition problem, all triangle constraints (7.1a) are violated by at most $\sqrt{2} - 1$, and all clique constraints (7.1b) are violated less than $\frac{1}{2}$. Hence, in the LP relaxation obtained from the ILP (7.1) by dropping the integrality constraints, this corresponds to the simultaneous partial fulfillment of all (exponentially many) constraints.

The semidefinite relaxation (8.6) of the Minimum k-Partition problem thus appears as an intriguing alternative to the classical LP relaxation. Due to the enormous amount of constraints the latter seems to be hardly solvable in practice. If this bottleneck is to be by-passed by adding the triangle and clique constraints as model cuts, then the separation problem for the clique inequalities, in particular, has to be solved effectively.

As an alternative to developing a traditional branch-and-cut algorithm for solving Minimum k-Partition problems with guaranteed quality, we may as well consider a branch-and-cut algorithm on the basis of the semidefinite relaxation and an SDP solver. Successful applications of this kind are reported by Helmberg [1995]; Helmberg, Rendl, Vanderbei, and Wolkowicz [1996]; Helmberg and Rendl [1998]; Helmberg and Weismantel [1998]; Karisch, Rendl, and Clausen [1998], for example. In their conclusion, Karisch *et al.* [1998] state: "Our results compare favorably to previously published ones [for graph bisection], which were obtained with cutting plane methods based on linear programming relaxations." As pointed out before, it is not even clear how competitive lower bounds an LP-based branch-and-cut algorithm for the MINIMUM K-PARTITION problem could provide for the instances we are interested in, see Section 6.3.

In the context of an SDP-based branch-and-cut algorithm for the MINIMUM K-PARTITION problem, the following points deserve consideration.

- Further investigations concerning the relation between the polytope $\mathcal{P}_{\leq k}(K_n)$ and its semidefinite relaxation $\Theta_{k,n}$ in order to give a better theoretical underpinning of our empirically observed lower bounds.

- A computational study of the strength of the semidefinite relaxation on a larger set of MINIMUM K-PARTITION instances with known optimal solution.

- For which classes of valid inequalities for $\mathcal{P}_{\leq k}(K_n)$ is the separation problem (heuristically) well solvable in practice and what is the effect on the strength of the corresponding relaxation?

Another interesting issue is the generation of good k-partitions on the basis of the solution for the semidefinite relaxation (8.6). How well, for example, does randomized rounding perform? Which other heuristics are of practical use?

Appendix A

Notation

The following survey on basic notation and facts from linear algebra, linear programming, and graph theory is primarily intended to serve as a glossary. For comprehensive introductions see, for example, Chvátal [1983] or Padberg [1995] for *linear programming*; Schrijver [1986] or Nemhauser and Wolsey [1988] for *integer linear programming*; Schrijver [1986] or Ziegler [1994] for *polyhedral theory*; Wolkowicz *et al.* [2000] for *semidefinite programming*; West [1996] or Diestel [1997] for *graph theory*; Cormen *et al.* [1990] for *algorithms and data structures*; and Garey and Johnson [1979] as well as Ausiello *et al.* [1999] for *computational complexity.*

Basics

The cardinality of the set A is denoted by $|A|$. The Cartesian product of two sets A and B is written as $A \times B$. The set difference of A and B is $A \setminus B$. The *intersection* and *union* of A and B are denoted by $A \cap B$ and $A \cup B$, respectively. The symbols $\subseteq$ and $\subsetneq$ denote set inclusion and proper set inclusion, respectively.

$|A|$

$A \times B,\ A \setminus B$

$A \cap B,\ A \cup B$

$A \subseteq B,\ A \subsetneq B$

The sets of real, rational, and integer numbers are denoted by $\mathbb{R}$, $\mathbb{Q}$, and $\mathbb{Z}$, respectively. Their restrictions to the nonnegative numbers are denoted by $\mathbb{R}_+$, $\mathbb{Q}_+$, and $\mathbb{Z}_+$. The symbol $\mathbb{K}$ is used to represent $\mathbb{Q}$ and $\mathbb{R}$ in cases where the definition or property applies to both fields. The set of all column vectors with n components, $n \geq 1$, and entries from some set B is denoted by B^n, in particular, $\mathbb{R}^n, \mathbb{Q}^n, \mathbb{Z}^n$.

$\mathbb{R}, \mathbb{Q}, \mathbb{Z}$

$\mathbb{R}_+, \mathbb{Q}_+, \mathbb{Z}_+$

$\mathbb{K}$

$\mathbb{R}^n, \mathbb{Q}^n, \mathbb{Z}^n$

Let k, n be a nonnegative integers. Then $k!$ is equal to 1 if $k = 0$ and equal to $1 \cdot \ldots \cdot k$ otherwise. Moreover, $\binom{n}{k}$ is equal to $\frac{n!}{(n-k)!\,k!}$ if $0 \leq k \leq n$ and equal to 0 otherwise. The expression $n \bmod k$ stands for the remainder of the integer division of n divided by k.

$k!$

$\binom{n}{k}$

$n \bmod k$

For $x \in \mathbb{K}$, the *ceiling* $\lceil x \rceil$ *of* x is the smallest integer larger than or equal to x, and the *floor* $\lfloor x \rfloor$ *of* x is the largest integer less or equal to x.

$\lceil x \rceil$

$\lfloor x \rfloor$

For $x, y \in \mathbb{R}$, $[x, y]$ and $]x, y[$ denote the closed and open interval of real numbers between x and y, respectively. Analogously, $[x, y]_{\mathbb{Q}}$ and

$[x, y],]x, y[$

$[x, y]_{\mathbb{Q}},]x, y[_{\mathbb{Q}}$

$]x,y[_{\mathbb{Q}}$ denote the closed and open interval between the rational numbers x and y.

incidence vector

support

For a finite set E, the function $x\colon E \to \mathbb{K}$ is identified with an $|E|$-dimensional column vector $x \in \mathbb{K}^{|E|}$. Given a finite set E and a subset $F \subseteq E$, the *incidence vector of F in E* is the $|E|$-dimensional vector $\chi^F \in \{0,1\}^E$ with $\chi^F(e) = 1$ if $e \in F$ and $\chi^F(e) = 0$ otherwise. The *support* of a function $f\colon X \to Y$ is the set $\{x \in X \mid f(x) \neq 0\}$. The expression f_x is used as an alternative to $f(x)$.

$\exists, \forall$

The symbols $\exists$ and $\forall$ stand for the *universal* and *existential quantifier*.

Linear Algebra

$\mathbb{K}^{m\times n}$

$A_{ij}, A_{i\cdot}, A_{\cdot j}, A_{IJ}$

The set $\mathbb{K}^{m\times n}$ consists of all $(m \times n)$-matrices with entries from $\mathbb{K}$. The n-dimensional column vectors are identified with $\mathbb{K}^{n\times 1}$. For a matrix $A \in \mathbb{K}^{m\times n}$, the entry in row i and column j is referenced by A_{ij}, the ith row by $A_{i\cdot}$, the jth column by $A_{\cdot j}$, and the submatrix consisting of the elements in the rows contained in set I and the columns contained in set J is referenced by A_{IJ}. If $I = J$, then A_{II} is *principal submatrix* of A.

$\det(A)$

$\mathrm{tr}(A)$

singular

regular

A^{-1}, I_n

The *determinant* of a square matrix $A \in \mathbb{K}^{n\times n}$ is denoted by $\det(A)$, and its *trace*, i. e., the sum of its diagonal elements, is denoted by $\mathrm{tr}(A)$. A square matrix A is *singular* if its determinant is zero, and it is nonsingular or *regular* otherwise. The multiplicative inverse of a regular matrix A is denoted by A^{-1}, and I_n denotes the *identity matrix* in $\mathbb{K}^{n\times n}$.

A^T

$\langle\cdot,\cdot\rangle$

$\langle AB,C\rangle = \langle A,CB^T\rangle$

$\langle A, xx^T\rangle = x^TAx$

scalar product

$\|\cdot\|$

The transpose of a matrix $A \in \mathbb{K}^{m\times n}$ is the matrix $A^T \in \mathbb{K}^{n\times m}$ with the columns of A forming the rows of A^T. In particular, the transpose of a column vector is a row vector. The *inner product* on $\mathbb{K}^{m\times n}$ is $\langle\cdot,\cdot\rangle\colon \mathbb{K}^{m\times n} \times \mathbb{K}^{m\times n} \to \mathbb{K}$ with $(A,B) \mapsto \langle A,B\rangle = \sum_{i=1}^{m}\sum_{j=1}^{n} A_{ij}B_{ij} = \mathrm{tr}(B^TA)$. The identity $\langle AB,C\rangle = \langle A,CB^T\rangle$ holds for all matrices with compatible dimensions. Furthermore, the identity $\langle A,xx^T\rangle = x^TAx$ holds for every vector $x \in \mathbb{K}^n$ and every matrix $A \in \mathbb{K}^{n\times n}$. The inner product of two (column) vectors is also called *scalar product*. The Euclidean *norm* of a vector $x \in \mathbb{K}^n$ is $\|x\| = \sqrt{\langle x,x\rangle}$.

positive
(semi-)definite

S_n^+, S_n^{++}

A matrix $A \in \mathbb{K}^{n\times n}$ is symmetric if $A = A^T$. A symmetric matrix is *positive semidefinite* ($A \succeq 0$) if $x^TAx \geq 0$ for all vectors $x \in \mathbb{K}^n$. If, furthermore, $x^TAx = 0$ implies $x = 0$, then A is *positive definite* ($A \succ 0$). The sets of $n \times n$-dimensional positive semidefinite and positive definite matrices are denoted by S_n^+ and S_n^{++}, respectively.

eigenvalue

eigenvector

A scalar $\lambda \in \mathbb{K}$ is an *eigenvalue* of a matrix $A \in \mathbb{K}^{n\times n}$ if $A\,x = \lambda\,x$ for some $x \in \mathbb{K}^n, x \neq 0$. If λ is an eigenvalue of A, then all vectors $x \in \mathbb{K}^n$ satisfying $A\,x = \lambda\,x$ are *eigenvectors* of A associated to the eigenvalue λ.

The determinant of a matrix A is the product of all its eigenvalues, and the trace is the sum of all its eigenvalues.

A vector $x \in \mathbb{K}^n$ is a *linear combination* of vectors in $X \subseteq \mathbb{K}^n$ if a finite number of vectors $x^1, \ldots, x^t \in X$ and scalars $\lambda_1, \ldots, \lambda_t \in \mathbb{K}$ exist such that $x = \sum_{i=1}^t \lambda_i x^i$. The vector x is a *conic combination* of the x^is if, in addition, $\lambda_i \geq 0$ for all i. It is an *affine combination* if $\sum_{i=1}^t \lambda_i = 1$, and a *convex combination* if $\sum_{i=1}^t \lambda_i = 1$ as well as $\lambda_i \geq 0$ for all i. The linear, affine, and convex hull as well as the cone of X, denoted by $\operatorname{lin}(X)$, $\operatorname{aff}(X)$, $\operatorname{conv}(X)$, and $\operatorname{cone}(X)$, are the sets of all linear, affine, convex, and conic combinations, respectively. A set X is *convex* if $X = \operatorname{conv}(X)$. A set X satisfying $X = \operatorname{cone}(X)$ is a *cone*. A cone X is *pointed* if $x \in X$ implies $-x \notin X$ for every nonzero vector x.

affine, conic, convex, linear combination

convex (pointed) cone

A set $X \subseteq \mathbb{K}^n$ of vectors is *linearly independent* if only one linear combination of vectors in X is equal to the zero-vector in $\mathbb{K}^n$. The *dimension* $\dim X$ of X is maximal number of linearly independent vectors from X, that is, the dimension of the linear hull of X as a subspace of $\mathbb{K}^n$.

linearly independent

$\dim X$

Polyhedral Theory

Given a vector $a \in \mathbb{K}^n \setminus \{0\}$ and a scalar $a_0 \in \mathbb{K}$, then the set $\{x \in \mathbb{K}^n \mid a^T x = a_0\}$ is a *hyperplane*, and $\{x \in \mathbb{K}^n \mid a^T x \leq a_0\}$ is the *half-space* delimited by the hyperplane. The finite intersection of half-spaces, given by $\{x \in \mathbb{K}^n \mid Ax \leq b\}$ with $A \in \mathbb{K}^{m \times n}$, $b \in \mathbb{K}^m$, is a *polyhedron*. A set $S \subseteq \mathbb{K}^n$ is bounded if it is contained in a set $\{x \in \mathbb{K}^n \mid \|x\| \leq r\}$ for some $r \in \mathbb{K}$. A bounded polyhedron is a *polytope*.

hyperplane

half-space

polyhedron

polytope

The inequality $a^T x \leq a_0$ for $a \in \mathbb{K}^n \setminus \{0\}$, $a_0 \in \mathbb{K}$ is *valid* for a polyhedron P if P is contained in the half-space $\{x \in \mathbb{K}^n \mid a^T x \leq a_0\}$ and it is *tight for P* if it is valid for P and the hyperplane $\{x \in \mathbb{K}^n \mid a^T x = \alpha\}$ contains at least one point in P.

valid

tight for P

The set $P \cap \{x \in \mathbb{K}^n \mid a^T x = a_0\}$ is the *face of P induced by* $a^T x \leq a_0$. A zero-dimensional face is a *vertex*. A face $F \subsetneq P$ of a polyhedron P is a *facet of P* if it is a maximal face with respect to inclusion. An inequality $a^T x \leq a_0$ is *facet-defining for P* if it is valid for P and $F = \{x \in P \mid a^T x = a_0\}$ is a facet of P. An equivalent characterization of a facet is that $\dim F = \dim P - 1$.

face of P

vertex

facet of P

facet-defining

Linear Programming

A polyhedron $P = \{x \in \mathbb{K}^n \mid Ax \leq b\} \subseteq \mathbb{K}^n$, a vector $c \in \mathbb{K}^n$, and an objective define a *linear program* or LP, for short. The minimization and

linear program

maximization versions are

$$\max\{c^T x \mid x \in P\} \quad \text{and} \quad \min\{c^T x \mid x \in P\}. \tag{A.1}$$

optimal solution
dual program

A vector $\bar{x} \in P$ attaining the maximum (minimum) in (A.1), provided this exists, is an *optimal solution*. In case P is a nonempty polytope, at least one vertex of P is an optimal solution. The *dual program* to a linear program $\max\{c^T x \mid Ax \leq b, x \geq 0\}$ is $\min\{b^T y \mid A^T y \geq c, y \geq 0\}$.

Theorem A.1 (Duality of linear programming). *Let $A \in \mathbb{K}^{m \times n}$, $b \in \mathbb{K}^m$, and $c \in \mathbb{K}^n$. In case*

$$\{x \in \mathbb{K}^n \mid Ax \leq b, x \geq 0\} \neq \emptyset \quad \textit{and} \quad \{y \in \mathbb{K}^m \mid A^T y \geq c, y \geq 0\} \neq \emptyset$$

the optimal solution values of

$$\max\{c^T x \mid Ax \leq b, x \geq 0\} \quad \textit{and} \quad \min\{b^T y \mid A^T y \geq c, y \geq 0\} \tag{A.2}$$

are finite, and

$$\bar{x} \in \{x \in \mathbb{K}^n \mid Ax \leq b, x \geq 0\} \quad \textit{and} \quad \bar{y} \in \{y \in \mathbb{K}^m \mid A^T y \geq c, y \geq 0\}$$

exist such that $c^T \bar{x} = b^T \bar{y}$.

binary/integer linear program

LP relaxation

A linear program turns into an *integer linear program* (ILP) if all variables are required to take integer values. In the special case of a *binary linear program* the values of the variables are restricted to 0 and 1. The *LP relaxation* of an integer linear program is obtained by dropping the integrality constraints.

Graph Theory

Our graph theoretic nomenclature is mostly taken from West [1996].

(simple) graph
vertex/edge set

incident, adjacent
neighborhood
degree
isolated vertex
graph complement
subgraph

A *simple graph* G with n vertices and m edges consists of a *vertex set* $V(G) = \{v_1, \ldots, v_n\}$ and an *edge set* $E(G) = \{e_1, \ldots, e_m\}$. Each edge is an unordered pair of distinct vertices. The edge $\{v, w\}$ is also written as vw. If $e = vw \in E(G)$, then e is *incident* to v and w, the vertices v and w are the *endpoints* of e, and v and w are *adjacent*. The *neighborhood* of a vertex are its adjacent vertices. The *degree* of a vertex is the number of incident edges. A vertex in a graph is *isolated* if its degree is zero.

The *graph complement* $\bar{G}$ of a graph G is a graph on the same vertex set as G with $vw \in E(\bar{G})$ if and only if $vw \notin E(G)$. A *subgraph* of a graph G is a graph H such that $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. The subgraph H of G is an *induced subgraph* if every edge in $E(G)$ with both endpoints in $V(H)$ is in $E(H)$. If H is an induced subgraph of G with vertex set S, then this is written as $H = G[S]$.

induced subgraph
$G[S]$

A nonempty subset Q of the vertices of $V(G)$ is a *clique* in G if every pair of vertices in Q is an edge in $E(G)$. A set $S \subseteq V(G)$ is an *independent set* if $G[S]$ contains no edge. A clique (independent set) is *maximal* if no larger clique (independent set) contains it, and it is *maximum* if no clique (independent set) of larger size exists. The size of a maximum clique in a graph G is its *clique number*, denoted by $\omega(G)$, and the size of a maximum independent set is its *independence* or *stability number*, denote by $\alpha(G)$.

clique
independent set
clique number
independence number

A graph is *complete* if its vertex set is a clique. K_n is complete graph on n vertices. A vertex is *simplicial* if its neighborhood is a clique.

complete graph,K_n
simplicial vertex

A *vertex labeling* of a graph G (with elements from a set Y) is a function $f\colon V(G) \to Y$. Analogously, an *edge labeling* is a function $g\colon E(G) \to Y$.

vertex labeling
edge labeling

A graph is *bipartite* if its vertex set can be partitioned into at most two independent sets; it is *k-partite* if its vertex set may be partitioned into k or fewer independent sets. A k-partite graph is *k-colorable*, and a k-coloring of a graph is a vertex labeling $f\colon V(G) \to \{1, \ldots, k\}$. The coloring is proper if $f(v) \neq f(w)$ for every pair of adjacent vertices. The minimum number k such that a graph G is properly k-colorable is its *chromatic number*, denoted by $\chi(G)$.

bipartite
k-partite
k-colorable
chromatic number

A *walk* of length k in a graph is a sequence v_0, e_1, v_1, e_2, $\ldots$, e_k, v_k of vertices and edges such that $e_i = v_{i-1}v_i$ for all $i = 1, \ldots, k$. The endpoints of the walk are v_0 and v_k. A *path* is a walk containing no vertex more than once. A *Hamiltonian path* contains every vertex of the graph. A *cycle* or *tour* is a walk with both endpoints being the same vertex and no repeated vertex otherwise. A cycle of length 3 is also called a *triangle*. The graph C_n contains n vertices and its edge set is a cycle. An edge vw is a *(l-)chord* with respect to a walk if $v = v_i$ and $w = v_{i+l}$ for some i.

walk
Hamiltonian path
cycle, tour
triangle
C_n
(l-)chord

A graph is *connected* if a path exists between any two vertices. A *tree* is a connected graph which does not contain a cycle. A *component* of a graph is a maximal induced subgraph that is connected. A *shortest path* between two vertices v and w is a path with endpoints v and w of shortest length. In a connected graph, the *diameter* is the maximal length of a shortest path; otherwise, the diameter is infinite. An *(edge) cut* in a graph is a subset of the edges for which its removal disconnects the graph. For a partition of the vertex set into two nonempty, disjoint sets S and T, the cut $[S, T]$ contains all edges with one endpoint in S and the other endpoint in T. A graph is *2-(edge) connected* if every cut has

connected, tree
component
shortest path
diameter
(edge) cut
2-connected

spanning

size at least 2. A subgraph H of a graph G is *spanning* if $V(H) = V(G)$ and H is connected.

(simple) digraph
vertex/arc set

head, tail
acyclic
orientation

A *simple directed graph* D with n vertices and m arcs consists of a *vertex set* $V(D) = \{v_1, \dots, v_n\}$ and an *arc set* $A(D) = \{a_1, \dots, a_m\}$, where each arc is an ordered pair of distinct vertices. We write vw for the arc (v, w). The vertex v is the *head* and the vertex w the *tail* of the arc vw. A digraph not containing any directed cycle is *acyclic*. A digraph D is an *orientation* of a graph G if $V(G) = V(D)$ and $vw \in E(G)$ if and only if either $vw \in E(D)$ or $wv \in E(D)$.

Asymptotic Function Growths

$\mathcal{O}(\cdot), \Omega(\cdot)$

Given a function $f\colon \mathbb{Z}_+ \to \mathbb{Z}_+$, let

$$\mathcal{O}(f) = \bigl\{g\colon \mathbb{Z}_+ \to \mathbb{Z}_+ \mid \exists\, c, a, n_0 \in \mathbb{Z}_+ \; \forall n \geq n_0 : g(n) \leq c\, f(n) + a\bigr\}$$

and

$$\Omega(f) = \bigl\{g\colon \mathbb{Z}_+ \to \mathbb{Z}_+ \mid \exists\, c, a, n_0 \in \mathbb{Z}_+ \; \forall n \geq n_0 : c\, g(n) + a \geq f(n)\bigr\}$$

Every function $g \in \mathcal{O}(f)$ grows asymptotically no more than f, whereas every $g \in \Omega(f)$ grows asymptotically at least as much as f.

Bibliography

Aardal, K., Hipolito, A., van Hoesel, C., Jansen, B., Roos, C., and Terlaky, T. EUCLID CALMA Radio Link Frequency Assignment Project: A branch-and-cut algorithm for the frequency assignment problem. Technical Annex T-2.2.1 A, T.U. Eindhoven RLFAP Group and T.U. Delft RLFAP Group, 1995.

Aardal, K., Hurkens, C., Lenstra, J., and Tiourine, S. Algorithms for frequency assignment problems. *CWI Quaterly*, 9(1 & 2):1–8, 1996.

Aarts, E. H. and Lenstra, J. K. (eds.). *Local Search in Combinatorial Optimization.* John Wiley & Sons Ltd., 1997.

Ahuja, R. K., Magnanti, T., and Orlin, J. B. *Network Flows.* Prentice Hall, 1992.

Alizadeh, F. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5(1):12–51, 1995.

Allen, S., Smith, D., and Hurley, S. Lower bounding techniques for frequency assignment. *Discrete Mathematics*, 197/198:41–52, 1999.

Applegate, D., Bixby, R., Chvátal, V., and Cook, W. Concorde: A computer code for the traveling salesman problem. 1997. URL `http://www.caam.rice.edu/~keck/concorde.html`.

Applegate, D., Bixby, R., Chvátal, V., and Cook, W. On the solution of traveling salesman problems. In *Proceedings of the International Congress of Mathematicians Berlin 1998*, vol. III of *Documenta Mathematica.* Deutsche Mathematiker Vereinigung, 1998.

Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A., and Protasi, M. *Complexity and Approximation: combinatorial optimization problems and their approximability properties.* Springer-Verlag, 1999.

Barahona, F. and Mahjoub, A. On the cut polytope. *Mathematical Programming*, 36:157–173, 1986.

Beckmann, D. and Killat, U. Frequency planning with respect to interference minimization in cellular radio networks. Tech. Rep. TD(99) 032, COST 259, Vienna, Austria, 1999.

Ben-Israel, A., Charnes, A., and Kortanek, K. O. Asymptotic duality over closed convex sets. *Journal of Mathematical Analysis and Applications*, 35:677–691, 1971.

Björklund, P., Värbrand, P., and Yuan, D. Optimal GSM network planning with frequency hopping. Tech. rep., Linköping Institute of Technology, Norrköpping, Sweden, 2000. Presented at the INFORMS Telecommuncitations Conference, Boca Ration, USA.

Bodlaender, H. L. A tourist guide through treewidth. *Acta Cybernetica*, 11(1–2):1–21, 1993.

Borndörfer, R. *Aspects of Set Packing, Partitioning, and Covering.* Ph.D. thesis, Technische Universität Berlin, Fachbereich Mathematik, Berlin, Germany, 1998.

Borndörfer, R., Eisenblätter, A., Grötschel, M., and Martin, A. Frequency assignment in cellular phone networks. *Annals of Operations Research*, 76:73–93, 1998a.

Borndörfer, R., Eisenblätter, A., Grötschel, M., and Martin, A. The orientation model for frequency assignment problems. Tech. Rep. TR 98-1, Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB), 1998b. URL `http://www.zib.de/PaperWeb/abstracts/TR-98-1`.

Bovet, D. P. and Crescenzi, P. *Introduction to the Theory of Complexity.* Series in Computer Science. Prentice Hall, 1994.

Brélaz, D. New methods to color the vertices of a graph. *Communications of the ACM*, 22(4):169–174, 1979.

Burer, S., Monteiro, R. D., and Zhang, Y. Interior-point algorithms for semidefinite programming based on a nonlinear programming formulation. Tech. Rep. TR 99-27, Department of Computational and Applied Mathematics, Rice Unviversity, 1999.

Buttmann-Beck, N. and Hassin, R. Approximation algorithms for minimum sum p-clustering. *Discrete Applied Mathematics*, 89:125–142, 1998.

Carlsson, M. and Grindal, M. Automatic frequency assignment for cellular telephones using constraint satisfaction techniques. In *Proceedings of the Tenth International Conference on Logic Programming*, pp. 648–665. 1993.

Carraghan, R. and Pardalos, P. An exact algorithms for the maximum clique problem. *Operations Research Letters*, 9:375–382, 1990.

Castelino, D., Hurley, S., and Stephens, N. A tabu search algorithm for frequency assignment. *Annals of Operations Research*, 63:301–319, 1996.

Chopra, S. and Rao, M. R. The partition problem. *Mathematical Programming*, 59:87–115, 1993.

Chopra, S. and Rao, M. R. Facets of the k-partition polytope. *Discrete Applied Mathematics*, 61:27–48, 1995.

Chvátal, V. *Linear Programming.* W. H. Freeman and Company, 1983.

Cormen, T. H., Leiserson, C. E., and Rivest, R. L. *Introduction to Algorithms.* The MIT Press, 1990.

Correia, L. M. (ed.). *COST 259: Wireless Flexible Personalized Communications.* John Wiley & Sons Ltd., 2001.

Costa, D. On the use of some known methods for T-colorings of graphs. *Annals of Operations Research*, 41:343–358, 1993.

Damosso, E. and Correia, L. M. (eds.). *COST 231: Digital mobile radio towards future generation systems.* European Commission, 1999.

Deza, M. M., Grötschel, M., and Laurent, M. Complete descriptions of small multicut polytopes. *Applied Geometry and Discrete Mathematics—The Victor Klee Festschrift*, 4:221–252, 1991.

Deza, M. M., Grötschel, M., and Laurent, M. Clique-web facets for multicut polytopes. *Mathematics of Operations Research*, 17:981–1000, 1992.

Deza, M. M. and Laurent, M. Facets for the cut cone I. *Mathematical Programming*, 56:121–160, 1992a.

Deza, M. M. and Laurent, M. Facets for the cut cone II: clique-web facets. *Mathematical Programming*, 56:161–188, 1992b.

Deza, M. M. and Laurent, M. *Geometry of Cuts and Metrics*, vol. 15 of *Algorithms and Combinatorics*. Springer-Verlag, 1997.

Diestel, R. *Graph Theory.* Springer-Verlag, 1997.

Dueck, G. and Scheuer, T. Threshold accepting: a general purpose optimization algorithm appearing superior to simulated annealing. *Journal of Computational Physics*, pp. 161–175, 1990.

Duque-Antón, M. and Kunz, D. Channel assignment based on neural network algorithms. In *Proc. DMR IV*, pp. 5.4.1–5.4.9. Oslo, Norway, 1990.

Duque-Antón, M., Kunz, D., and Rüber, B. Channel assignment for cellular radio using simulated annealing. *IEEE Transactions on Vehicular Technology*, 42(1):14–21, 1993.

Eidenbenz, S., Stamm, C., and Widmayer, P. RA3DIO – Wellenausbreitung in 3D. *Computerworld Schweiz*, 37:26–29, 1999.

Eisenblätter, A. Combinatorial lower bounds for co-channel interference in GSM-networks (preliminary report). Tech. Rep. TD(98) 104, COST 259, Duisburg, Germany, 1998.

Eisenblätter, A. and Kürner, T. Benchmarking frequency allocation strategies. Tech. Rep. TD(00) 044, COST 259, Bergen, Norway, 2000.

Eisenblätter, A., Kürner, T., and Fauß, R. Radio planning algorithms for interference reduction in cellular networks. In *Communications for the Millennium. COST 252/259 Joint Workshop, University of Bradford, 21–22 Apr. 1998.*, pp. 87–92. 1998.

Eisenblätter, A., Kürner, T., and Fauß, R. Analysis of C/I-ratio thresholds for frequency planning. Tech. Rep. TD(99) 012, COST 259, Thessaloniki, Greece, 1999.

Erdős, P., Rubin, A. L., and Taylor, H. Choosability in graphs. *Congressus Numerantium*, 26:125–157, 1979.

FAP web. *FAP web—A website about Frequency Assignment Problems.* Andreas Eisenblätter and Arie Koster, 2000. URL `http://fap.zib.de/`.

Ferracioli, M. and Verdone, R. A general methodology based on the handover rate for network planning of cellular radio networks based on ATM. *IEEE Journal on Selected Areas in Communications*, 18(3), 2000.

Ferreira, C. E., Martin, A., de Souza, C. C., Weismantel, R., and Wolsey, L. A. The node capacitated graph partitioning problem: a computational study. *Mathematical Programming*, 74A(3):247–266, 1996.

Fischetti, M., Lepschy, C., Minerva, G., Jacur, G. R., and Toto, E. Frequency assignment in mobil radio systems using branch-and-cut techniques. *European Journal of Operational Research*, 123:241–255, 2000.

Frieze, A. and Jerrum, M. Improved approximation algorithms for max k-cut and max bisection. *Algorithmica*, 18:67–81, 1997.

Gamst, A. Some lower bounds for a class of frequency assignment problems. *IEEE Transactions on Vehicular Technology*, VT-35(1):8–14, 1986.

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

Garg, N., Vazirani, V. V., and Yannakakis, M. Approximate max-flow min-(multi)cut theorems and their applications. *SIAM Journal on Computing*, 25:235–251, 1996.

Gerards, A. Testing the odd bicycle wheel inequalites for the bipartite subgraph polytope. *Mathematics of Operations Research*, 10:359–360, 1985.

Goemans, M. X. and Williamson, D. P. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, 1995.

Goldschmidt, O. and Hochbaum, D. S. A polynomial algorithm for the k-cut problem for fixed k. *Mathematics of Operations Research*, 19(1):24–37, 1994.

Gotzner, U., Gamst, A., and Rathgeber, R. Statial traffic distribution in cellular networks. In *Proc. IEEE VTC'97 Ottawa Canada*, pp. 1994–1998. IEEE, 1997.

Grace, D. *Distributed Dynamic Channel Assignment for the Wireless Environment.* Ph.D. thesis, University of York, York, UK, 1999.

Grace, D., Burr, A. G., and Tozer, T. C. Performance of a distributed dynamic channel assignment algorithm incorporating power control in a wireless environment. In *Globecom 98*. Sydney, Australia, 1998.

Grötschel, M., Lovász, L., and Schrijver, A. *Geometric Algorithms and Combinatorial Optimization.* Springer-Verlag, 1988.

Grötschel, M. and Wakabayashi, Y. A cutting plane algorithm for a clustering problem. *Mathematical Programming*, 45:59–96, 1989.

Grötschel, M. and Wakabayashi, Y. Facets of the clique partitioning polytope. *Mathematical Programming*, 47:367–387, 1990.

GSM Association. 2000. URL `http://www.gsmworld.com`.

Haberland, D. *Azyklische Subdigraphenprobleme und Frequenzzuweisung im Mobilfunk.* Master's thesis, Technische Universität Berlin, Fachbereich Mathematik, Berlin, Germany, 1996.

Hale, W. K. Frequency assignment: Theory and applications. In *Proceedings of the IEEE*, vol. 68, pp. 1497–1514. IEEE, 1980.

Hao, J.-K., Dorne, R., and Galinier, P. Tabu search for the frequency assignment in cellular radio networks. *Journal of Heuristics*, 4:47–62, 1998.

Håstad, J. Some optimal inapproximability results. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pp. 1–10. ACM, 1997.

Hebermehl, M. *Heuristische Algorithmen zur Lösung von Färbungsproblemen im Mobilfunk*. Master's thesis, Technische Universität Berlin, Fachbereich Mathematik, Berlin, Germany, 1996.

Hellebrandt, M. and Heller, H. A new heuristic method for frequency assignment. Tech. Rep. TD(00) 003, COST 259, Valencia, Spain, 2000.

Heller, H. private communication, 2000.

Helmberg, C. *An interior point method for semidefinite programming and max-cut bounds*. Ph.D. thesis, Technische Universität Graz, Graz, Austria, 1995.

Helmberg, C. Semidefinite programming for combinatorial optimization. Habilitationsschrift, 2000.

Helmberg, C. and Rendl, F. Solving quadratic (0,1)-problems by semidefinite programs and cutting planes. *Mathematical Programming*, 82(3):291–315, 1998.

Helmberg, C., Rendl, F., Vanderbei, R. J., and Wolkowicz, H. An interior point method for semidefinite programming. *SIAM Journal on Optimization*, 6(2), 1996.

Helmberg, C. and Weismantel, R. Cutting plane algorithms for semidefinite relaxations. In P. M. Pardalos and H. Wolkowicz (eds.), *Topics in Semidefinite and Interior-Point Methods*, vol. 18 of *Fields Institute Communications Series*, pp. 197–213. American Mathematical Society, 1998.

Hurley, S., Smith, D., and Thiel, S. FASoft: A system for discrete channel frequency assignment. *Radio Science*, 32(5):1921–1939, 1997.

Janssen, J. and Kilakos, K. Polyhedral analysis of channel assignment problems: (i) tours. Tech. Rep. LSE-CDAM-96-17, London School of Economics, 1996.

Janssen, J. and Kilakos, K. An optimal solution to the "Philadelphia" channel assignmetn problem. *IEEE Transactions on Vehicular Technology*, 48(3):1012–1014, 1999.

Jaumard, B., Marcotte, O., and Meyer, C. Estimation of the quality of cellular networks using column generation techniques. Tech. Rep. G-98-02, GERAD, University of Montreal, Montreal, Canada, 1998.

Jaumard, B., Marcotte, O., and Meyer, C. Mathematical models and exact methods for channel assignment in cellular networks. In B. Sansò and P. Soriano (eds.), *Telecommunications Network Planning*, chap. 13, pp. 239–255. Kluwer Academic Publishers, 1999.

Jünger, M. and Rinaldi, G. Relaxations of the max cut problem and computation of spin glass ground states. In P. Kischka, H.-W. Lorenz, U. Derigs, W. Domschke, P. Kleinschmidt, and R. Möhring (eds.), *Operations Research Proceedings 1997, Selected Papers of the Symposium on Operations Research (SOR'97)*, pp. 74–83. Springer-Verlag, 1998.

Johnson, D., Pataki, G., and Alizadeh, F. Seventh DIMACS implementation challenge: Semidefinite and related optimization problems. DIMACS, 2000. URL `http://dimacs.rutgers.edu/Challenges/Seventh/`.

Johnson, E. L., Mehrotra, A., and Nemhauser, G. L. Min-cut clustering. *Mathematical Programming*, 62:133–151, 1993.

Jünger, M., Martin, A., Reinelt, G., and Weismantel, R. Quadratic 0/1 optimization and a decomposition approach for the placement of electronic circuits. *Mathematical Programming*, 63:257–279, 1994.

Jünger, M., Reinelt, G., and Rinaldi, G. *Network Models*, vol. 7 of *Handbooks in Operations Research and Management Science*, chap. The traveling salesman problem, pp. 225–330. Elsevier Science B. V., 1995a.

Jünger, M., Reinelt, G., and Thienel, S. Practical problem solving with cutting plane algorithms in combinatorial optimization. In W. Cook, L. Lovász, and P. D. Seymour (eds.), *Combinatorial Optimization*, vol. 20, pp. 111–152. American Mathematical Society, 1995b.

Kann, V., Khanna, S., Lagergren, J., and Panconesi, A. Hardness of approximating max k-cut and its dual. *Chicago Journal of Theoretical Computer Science*, 2, 1997. URL `http://www.cs.uchicago.edu/publications/cjtcs`.

Karger, D., Motwani, R., and Sudan, M. Approximate graph coloring by semidefinite programming. In *FOCS 94*, pp. 2–13. 1994.

Karger, D., Motwani, R., and Sudan, M. Approximate graph coloring by semidefinite programming. *Journal of the ACM*, 45(2):246–265, 1998.

Karisch, S. E., Rendl, F., and Clausen, J. Solving graph bisection problems with semidefinite programming. *INFORMS Journal on Computing*, 1998. To appear.

Kennedy, K., Vries, E. D., and Koorevaar, P. Performance of a distributed DCA algorithm under inhomogeneous traffic modelled from an operational GSM network. In *Proc. of VTC'98–48th IEEE Vehicular Technology Conference*. Phoenix, Arizona, USA, 1998.

Koster, A. M. C. A. *Frequency Assignment – Models and Algorithms*. Ph.D. thesis, Universiteit Maastricht, Maastricht, The Netherlands, 1999.

Krumke, S. O. *On the Approximability of Location and Network Design Problems*. Ph.D. thesis, Julius-Maximilians-Universität Würzburg, Würzburg, Germany, 1996.

Kürner, T., Cichon, D. J., and Wiesbeck, W. Concepts and results for 3D digital terrain-based wave propagation models: An overview. *IEEE Journal on Selected Areas in Communications*, 11(7):1002–1012, 1993.

Kürner, T. and Fauß, R. Investigation of path loss algorithms at 1900 MHz. Tech. Rep. TD(94) 109, COST 231, Darmstadt, Germany, 1994.

Kürner, T., Fauß, R., and Wäsch, A. A hybrid propagation modelling approach for DCS 1800 macro cells. In *Proc. IEEE VTC'96*, pp. 1628–1632. Atlanta, Georgia, USA, April 28–May 1, 1996.

Laurent, M. and Poljak, S. On a positive semidefinite relaxation of the cut polytope. *Linear Algebra and its Applications*, 223/224:439–461, 1995.

Laurent, M. and Poljak, S. Gap inequalities for the cut polytope. *European Journal of Combinatorics*, 17(2–3):233–254, 1996a.

Laurent, M. and Poljak, S. On the facial structure of the set of correlation matrices. *SIAM Journal on Matrix Analysis and Applications*, 17(3):530–547, 1996b.

Laurent, M., Poljak, S., and Rendl, F. Connection between semidefinite relaxations of the max-cut and stable set problems. *Mathematical Programming*, 77B(2):225–246, 1997.

Lin, S. and Kernighan, B. An effective heuristic algorithm for the traveling salesman problem. *Operations Research*, 21:498–516, 1973.

Löbel, A. *MCF Version 1.0 – A network simplex implementation*, 1997. URL `http://www.zib.de/Optimization/Software/Mcf`.

Lovász, L. On the Shannon capacity of a graph. *IEEE Transactions on Information Theory*, IT-25:1–7, 1979.

LEDA 3.6.1. *The LEDA User Manual Version 3.6.1.* LEDA Software GmbH, D-66123 Saarbrücken, Germany, 1998. URL http://www.mpi-sb.mpg.de/LEDA.

Majewski, K., Hallmann, E., and Volke, A. From propagation predictions to frequency planning: An approach capturing radio link control options. In *AP2000 Millennium Conference on Antennas and Propagation.* Davos, Switzerland, 2000.

Malesińska, E. *Graph-Theoretical Models for Frequency Assignment Problems.* Ph.D. thesis, Technische Universität Berlin, Fachbereich Mathematik, Berlin, Germany, 1997.

Mehlhorn, K. and Näher, S. *LEDA: a platform for combinatorial and geometric computing.* Cambridge University Press, 1999.

Menolascino, R. and Pizarroso, M. STORMS Project Final Report. ACTS Programm AC016, European Union, 1999. URL http://www.infowin.org/ACTS/RUS/PROJECTS/FINAL-REPORTS/fr-016.pdf.

Metzger, B. H. Spectrum management technique. Presentation at 38th National ORSA meeting (Detroit, MI), 1970.

Mittelmann, H. Benchmarks for optimization software. 2000. URL http://plato.la.asu.edu/bench.html.

Mouly, M. and Pautet, M.-B. *The GSM System for Mobile Communications.* Cell & Sys, France, 1992.

Murphey, R. A., Pardalos, P. M., and Resende, M. G. C. Frequency assignment problems. In D.-Z. Du and P. M. Pardalos (eds.), *Handbook of Combinatorial Optimization*, vol. A. Kluwer Academic Publishers, 1999.

Nemhauser, G. L. and Wolsey, L. A. *Integer and Combinatorial Optimization.* John Wiley & Sons Ltd., 1988.

Nesterov, Y. and Nemirovskii, A. *Interior-point polynomial algorithms in convex programming.* No. 13 in SIAM Studies in Applied Mathematics. Society for Industrial and Applied Mathematics, 1994.

Nielsen, T. T. and Wigard, J. *Performance Enhancements in a Frequency Hopping GSM Network.* Kluwer Academic Publishers, 2000.

Oosten, M., Rutten, J., and Spieksma, F. The facial structure of the clique partitioning polytope. Tech. Rep. Rep. 95-09, Universiteit Maastricht, Maastricht, The Netherlands, 1995. URL ftp://ftp.unimaas.nl/pub/grey_files/fdaw/1995/rep95-09.ps.

Padberg, M. *Linear Optimization and Extensions.* Springer-Verlag, 1995.

Papadimitriou, C. H. *Computational Complexity.* Addison-Wesley, 1994.

Plehn, J. Applied frequency assignment. In *Proceedings of the IEEE Vehicular Technology Conference.* IEEE, 1994.

Poljak, S. and Tuza, Z. Maximum cuts and large bipartite subgraphs. In W. Cook, L. Lovász, and P. Seymour (eds.), *Combinatorial Optimization*, vol. 20 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pp. 188–244. American Mathematical Society, 1995.

Ramana, M. V. An exact duality theory for semidefinite programming and its complexity implications. *Mathematical Programming*, 77(2):129–162, 1997.

Raychaudhuri, A. *Intersection Assignments, T-Colourings and Powers of Graphs.* Ph.D. thesis, Rutgers University, 1985.

Raychaudhuri, A. Further results on T-coloring and frequency assignment problems. *SIAM Journal on Discrete Mathematics*, 7(4):605–613, 1994.

Redl, S. M., Weber, M. K., and Oliphant, M. W. *An Introduction to GSM.* Mobile Communications Series. Artech House Publishers, 1995.

Roberts, F. S. From garbage to rainbows: generalizations of graph coloring and their applications. In Y. Alavi, G. Chartrand, O. Oellermann, and A. Schwenk (eds.), *Graph Theory, Combinatorics, and Applications*, vol. 2, pp. 1031–1052. John Wiley & Sons Ltd., 1991a.

Roberts, F. S. T-colorings of graphs: recent results and open problems. *Discrete Mathematics*, 93:229–245, 1991b.

Rutten, J. *Polyhedral Clustering.* Ph.D. thesis, Universiteit Maastricht, Maastricht, The Netherlands, 1998.

Sahni, S. K. and Gonzalez, T. F. P-complete approximation problems. *Journal of the ACM*, 23:555–565, 1976.

Schneider, K. *Frequenzzuweisung im Mobilfunk mittels lokaler Suche.* Master's thesis, Technische Universität Berlin, Fachbereich Mathematik, 1997.

Schrijver, A. *Theory of Linear and Integer Programming.* John Wiley & Sons Ltd., 1986.

Schulz, A. S. *Polytopes and Scheduling.* Ph.D. thesis, Technische Universität Berlin, Fachbereich Mathematik, Berlin, Germany, 1996.

Smith, D., Allen, S. M., Hurley, S., and Watkins, W. J. Frequency assignment: Methods and algorithms. In NATO (ed.), *RTO IST Symposium on "Frequency Assignment, Sharing and Conservation in Systems (Aerospace)", Aalborg, Denmark, 5–7 Oct. 1998*, vol. MP-13 of *RTO*, pp. K-1–K-18. Aalborg, Denmark, 1998.

Smith, D. and Hurley, S. Bounds for the frequency assignment problem. *Discrete Mathematics*, 167–168:571–582, 1997.

Stoer, J. and Witzgall, C. *Convexity and Optimization in Finite Dimension I*. Springer-Verlag, 1970.

Sturm, J. F. Sedumi. 1998. URL `http://www2.unimaas.nl/~sturm/software/sedumi.html`.

Tesman, B. List T-colorings of graphs. *Discrete Applied Mathematics*, 45:277–289, 1993.

Thienel, S. *ABACUS - A Branch-And-CUt System*. Ph.D. thesis, Universität zu Köln, Köln, Germany, 1995.

Tutschku, K., Mathar, R., and Niessen, T. Interference minimization in wireless communciation systems by optimal cell site selection. In *3rd European Personal Mobile Communication Conference*. Paris, France, 1999.

Vandenberghe, L. and Boyd, S. Semidefinite programming. *SIAM Review*, 38(1):49–95, 1996.

Wessäly, R. *DImensioning Survivable Capacitated NETworks*. Ph.D. thesis, Technische Universität Berlin, Fachbereich Mathematik, Berlin, Germany, 2000.

West, D. B. *Introduction to graph theory*. Prentice Hall, 1996.

Wolkowicz, H., Saigal, R., and Vandenberghe, L. (eds.). *Handbook on Semidefinite Programming*, vol. 27. Kluwer Academic Publishers, 2000.

Wolkowicz, H. and Zhao, Q. Semidefinite programming relaxations for the graph partitioning problem. *Discrete Applied Mathematics*, 96–97:461–479, 1999.

www.emc-database.com. 2000. URL `http://www.emc-database.com`.

Ziegler, G. M. *Lectures on Polytopes*. Springer-Verlag, 1994.

Index

Zusammenfassung

Eine Schlüsseltechnologie im Informationszeitalter ist die mobile Telekommunikation. Diese kann durch Interferenz empfindlich gestört werden. Für GSM-Mobilfunknetze untersucht die Dissertation wie sich Interferenz möglichst weitgehend vermeiden läßt, indem die verfügbaren Frequenzen geeignet an die Basisstationen zugewiesen werden. Mathematisch wird dieses Ziel als Minimierung der Gesamtinterferenz aufgefaßt.

Heuristische Methoden zur Lösung des Frequenzzuweisungsproblems werden entwickelt und anhand der Laufzeiten und Ergebnisse für Planungsfälle, die der Praxis entstammen, verglichen. Der Großteil der Methoden eignet sich aufgrund des guten Laufzeitverhaltens für den interaktiven Einsatz bei der Netzplanung. Die Resultate sind im Vergleich mit denen des besten derzeit bekannten (aber deutlich langsameren) Verfahrens durchaus akzeptabel. Eine Auswahl der Methoden ist heute bei der E-Plus Mobilfunk GmbH & Co. KG erfolgreich im Einsatz.

Weiterhin wird in der Dissertation der Frage nachgegangen, wieviel Interferenz in einem gegebenen Netz bei der Frequenzzuweisung unvermeidbar ist. Die Ergebnisse entsprechender Berechnungen werden verwendet, um (im mathematischen Sinne) Qualitätsgarantien für Frequenzzuweisungen hinsichtlich der Interferenzvermeidung zu geben. Im besten betrachteten Fall verursacht eine Frequenzzuweisung nur doppelt soviel Interferenz wie nachweislich unvermeidbar.

Das Frequenzzuweisungsproblem läßt sich zu einem k-Partitionierungsproblem eines vollständigen Graphen relaxieren. Dem k-Partitionierungsproblem ist (ausgehend von einer Formulierung als ganzzahliges lineares Programm) eine Polyederklasse zugeordnet, wobei die Ecken der Polyeder jeweils die k-Partitionierungen des zugehörigen Graphen darstellen. Anstelle der sonst üblichen polyedrischen Relaxierungen wird eine nichtpolyedrische Umschreibung des Polyeders analysiert, die sich als Lösungsmenge eines semidefiniten Programmes ergibt. Dieses Programm läßt sich für festes $\varepsilon > 0$ in Polynomialzeit ε-optimal lösen (im Gegensatz zur linearen Relaxierung des ganzzahligen Programmes – $\mathcal{P} \neq \mathcal{NP}$ vorausgesetzt).

Die Lösung der semidefiniten Programme führt zu den derzeit mit Abstand besten unteren Abschätzungen der unvermeidbaren (Gleichkanal-)Interferenz. Zudem handelt es sich um eine der ersten Anwendungen von semidefiniter Programmierung bei großen industriellen Problemen mit kombinatorischem Hintergrund.

Schlüsselwörter: GSM, Frequenzzuweisung, Minimale k-Partitionierung, Heuristiken, Semidefinite Programmierung, Ganzzahlige Programmierung, Polyeder.
Mathematics Subject Classification (MSC 2000): 90C27 90C35 90B18 90C22 90C57

Lebenslauf

Andreas Eisenblätter
geboren am 27. August 1967 in Hilden/Rhld.

1974–1978: Grundschule

1978–1980: Integrierte Gesamtschule Nieder-Roden

1980–1987: Gymnasium in Haan, Abitur im Mai 1987

1987–1990: Studium der Technischen Informatik an der Berufsakademie Stuttgart

Sept. 1990: Diplom-Ingenieur (BA), Vertiefungsrichtung „Prozeßdatenverarbeitung“

1990–1991: Zivildienst beim Studentenwerk Düsseldorf A.ö.R.

1988–1992: Teilzeitstudium der Mathematik an der Fernuniversität Hagen

1992–1995: Mathematikstudium an der Universität Heidelberg

1992–1994: Studiengangzweithörer in Informatik an der Fernuniversität Hagen

1992–1994: Freier Mitarbeiter am European Networking Center der IBM, Heidelberg

1994–1995: Austauschstudent an der Univ. of Illinois at Urbana-Champaign, USA

Sept. 1995: Diplom in Mathematik an der Universität Heidelberg, Spezialgebiet „Komplexitätstheorie“, Betreuer: Prof. Dr. Ambos-Spies

seit 1995: Wissenschaftlicher Mitarbeiter am Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB)

Juni 2000: Mitgründer der Ariston Consulting GmbH (jetzt Atesio GmbH)