TELECOMMUNICATIONS MODELING, POLICY, AND TECHNOLOGY

OPERATIONS RESEARCH/COMPUTER SCIENCE INTERFACES

Professor Ramesh Sharda
Oklahoma State University

Prof. Dr. Stefan Voß
Universität Hamburg

Bierwirth / *Adaptive Search and the Management of Logistics Systems*
Laguna & González-Velarde / *Computing Tools for Modeling, Optimization and Simulation*
Stilman / *Linguistic Geometry: From Search to Construction*
Sakawa / *Genetic Algorithms and Fuzzy Multiobjective Optimization*
Ribeiro & Hansen / *Essays and Surveys in Metaheuristics*
Holsapple, Jacob & Rao / *Business Modelling: Multidisciplinary Approaches — Economics, Operational and Information Systems Perspectives*
Sleezer, Wentling & Cude/*Human Resource Development And Information Technology: Making Global Connections*
Voß & Woodruff / *Optimization Software Class Libraries*
Upadhyaya et al / *Mobile Computing: Implementing Pervasive Information and Communications Technologies*
Reeves & Rowe / *Genetic Algorithms—Principles and Perspectives: A Guide to GA Theory*
Bhargava & Ye / *Computational Modeling And Problem Solving In The Networked World: Interfaces in Computer Science & Operations Research*
Woodruff / *Network Interdiction And Stochastic Integer Programming*
Anandalingam & Raghavan / *Telecommunications Network Design And Management*
Laguna & Martí / *Scatter Search: Methodology And Implementations In C*
Gosavi/ *Simulation-Based Optimization: Parametric Optimization Techniques and Reinforcement Learning*
Koutsoukis & Mitra / *Decision Modelling And Information Systems: The Information Value Chain*
Milano / *Constraint And Integer Programming: Toward a Unified Methodology*
Wilson & Nuzzolo / *Schedule-Based Dynamic Transit Modeling: Theory and Applications*
Golden, Raghavan & Wasil / *The Next Wave in Computing, Optimization, And Decision Technologies*
Rego & Alidaee/ *Metaheuristics Optimization via Memory and Evolution: Tabu Search and Scatter Search*
Kitamura & Kuwahara / *Simulation Approaches in Transportation Analysis: Recent Advances and Challenges*
Ibaraki, Nonobe & Yagiura / *Metaheuristics: Progress as Real Problem Solvers*
Golumbic & Hartman / *Graph Theory, Combinatorics, and Algorithms: Interdisciplinary Applications*
Raghavan & Anandalingam / *Telecommunications Planning: Innovations in Pricing, Network Design and Management*
Mattfeld / *The Management of Transshipment Terminals: Decision Support for Terminal Operations in Finished Vehicle Supply Chains*
Alba & Martí/ *Metaheuristic Procedures for Training Neural Networks*
Alt, Fu & Golden/ *Perspectives in Operations Research: Papers in honor of Saul Gass' 80th Birthday*
Baker et al/ *Extending the Horizons: Adv. In Computing, Optimization, and Dec. Technologies*
Zeimpekis et al/ *Dynamic Fleet Management: Concepts, Systems, Algorithms & Case Studies*
Doerner et al/ *Metaheuristics: Progress in Complex Systems Optimization*
Goel/ *Fleet Telematics: Real-time management & planning of commercial vehicle operations*
Gondran & Minoux/ *Graphs, Dioïds and Semirings: New models and algorithms*
Alba & Dorronsoro/ *Cellular Genetic Algorithms*
Golden, Raghavan & Wasil/ *The Vehicle Routing Problem: Latest advances and new challenges*

TELECOMMUNICATIONS MODELING, POLICY, AND TECHNOLOGY

Edited by

S. RAGHAVAN
University of Maryland

BRUCE GOLDEN
University of Maryland

EDWARD WASIL
American University

Springer

Editors
S. Raghavan
University of Maryland
College Park, MD, USA

Bruce Golden
University of Maryland
College Park, MD, USA

Edward Wasil
American University
Washington, DC, USA

Series Editors
Ramesh Sharda
Oklahoma State University
Stillwater, Oklahoma, USA

Stefan Voß
Universität Hamburg
Germany

ISBN: 978-0-387-77779-5 e-ISBN: 978-0-387-77780-1

Library of Congress Control Number: 2007943830

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Contents

Preface

This edited book serves as a companion volume to the Ninth INFORMS Telecommunications Conference held in College Park, Maryland, from March 27 to 29, 2008. The 17 papers in this book were carefully selected after a thorough review process.

Rapid advances in telecommunications technology have spawned many new innovative applications. These advances in technology have also fostered new research problems. In a certain sense, each one of the papers in this book is motivated by these advances in technology. Technologies considered range from free-space optical networks and vehicular ad-hoc networks to wave division multiplexing and multiprotocol label switching. The research contained in these papers covers a broad spectrum that includes the design of business models, tools for spectrum auctions, Internet charging schemes, Internet routing policies, and network design problems. Together, these papers address issues that deal with both engineering design and policy.

We thank all of the authors for their hard work and invaluable contributions to this book. We are very pleased with the outcome of this edited book, and hope these papers will give rise to new ideas and research in their respective domains.

S. Raghavan, Bruce Golden, and Edward Wasil

Chapter 1

SINGLE-LAYER CUTS FOR MULTI-LAYER NETWORK DESIGN PROBLEMS

Arie M.C.A. Koster

University of Warwick, Centre for Discrete Mathematics and its Applications (DIMAP), Coventry CV4 7AL, United Kingdom

Arie.Koster@wbs.ac.uk

Sebastian Orlowski, Christian Raack

Zuse Institute Berlin (ZIB), Takustr. 7, D-14195 Berlin

{orlowski,raack}@zib.de

Georg Baier

Siemens AG CT, Discrete Optimization, Munich, Germany

georg.baier@siemens.com

Thomas Engel

Nokia Siemens Networks GmbH & Co. KG, Munich, Germany

thomas.1.engel@nsn.com

Abstract We study a planning problem arising in SDH/WDM multi-layer telecommunication network design. The goal is to find a minimum cost installation of link and node hardware of both network layers such that traffic demands can be realized via grooming and a survivable routing. We present a mixed-integer programming formulation for a predefined set of admissible logical links that takes many practical side constraints into account, including node hardware, several bit-rates, and survivability against single physical node or link failures. This model is solved using a branch-and-cut approach with cutting planes based on either of the two layers. On several realistic two-layer planning scenarios, we show that these cutting planes are still useful in the multi-layer context, helping to increase the dual bound and to reduce the optimality gaps.

Keywords: Telecommunication networks; multi-layer network design; mixed-integer programming; cutting planes.

1. Introduction

During the last decade, dense wavelength division multiplexing (DWDM) has turned out to be the dominant network technology in high-capacity optical backbone networks. It provides a flexible way to expand capacity in optical networks without requiring new cabling. Current DWDM systems usually provide 40 or 80 different wavelengths on a single optical fiber to carry high capacity channels, e. g., 2.5, 10, or 40 Gbit/s per wavelength. Typically, these capacities exhibit economies of scale, such that, for instance, the cost of 10 Gbit/s is only three times the cost of 2.5 Gbit/s. Low-granularity traffic given, for instance, in units of 2 Mbps, can be routed through these high-capacity wavelength channels. Flexible optical network nodes selectively terminate a wavelength or let them pass through to the next fiber, provided that an add/drop multiplexer with sufficient switching capacity has been installed to handle the terminating traffic. Ultra long-haul transmission permits high capacity optical channels via several fiber segments requiring transponders only at the end of the whole path, whose cost depends on the data rate and the length of the chosen path.

The corresponding network design problem can be summarized as follows. Given is a set of network nodes together with potential optical fiber connections between them. This optical network is called the *physical layer*. On every fiber, a limited number of lightpath channels can be transmitted simultaneously, each of them corresponding to a capacitated path in the physical network. The nodes together with the lightpath connections form a so-called *logical network* on top of the physical one. Setting aside some technical limitations, any path in the physical network can be used for a lightpath, which leads to many parallel logical links. In practice, however, the set of admissible lightpaths is often restricted to several short paths between each node-pair. A lightpath can be equipped with different bandwidths, and lower-rate traffic demands have to be routed via the lightpaths without exceeding their capacities. A demand may be 1+1-protected, i. e., twice the demand value must be routed such that in case of any single physical link or node failure, at least the demand value survives. To terminate a lightpath, a sufficiently large electrical cross-connect (EXC) must be installed at both end-nodes. The EXC converts the wavelength signal into an electrical SDH signal and extracts lower-rate traffic from it. The latter is either terminated at that node or recombined with other traffic to form new wavelength signals which are sent out on other lightpaths. The goal of the optimization is to minimize total installation cost.

Like in any other publication where an integrated two-layer model is actually used for computations, we do not explicitly assign wavelengths to the lightpaths because finding a suitable wavelength assignment is an extremely hard problem on its own. Instead, we make sure that the maximum number of lightpaths on each fiber is not exceeded, and propose to solve the wavelength assignment and converter installation problem in a subsequent step, as done successfully in [23]. It has been shown in [24] that such an approach causes at most a marginal increase in the overall installation cost on practical instances.

The network planning task is particularly driven by two parameters: the bound on the number of wavelengths per fiber and the transponder prices. A shortage in wavelengths may force the network planner to employ optical channels with high data rates. To keep the total transponder cost low, a suitable set of lightpaths has to be chosen in order to make the best possible use of these high data rates. To draw the maximal benefit out of the optical and the aggregation equipment, both layers have to be optimized together.

Already the optimal design of a single layer network is a challenging task that has been considered by many research groups, see for instance [3, 18, 33, 34] and references therein. A branch-and-cut algorithm enhanced by user-defined, problem-specific cutting planes has been proven to be a very successful solution approach in this context. The combined optimization of two layers significantly increases the complexity of the planning task. This is mainly due to the combined network design problem with integer capacities on the logical layer and the fixed-charge network design problem on the physical layer, and due to the large number of logical links with corresponding integer capacity variables. In previous publications, mixed-integer programming techniques have been used for designing a logical layer with respect to a fixed physical layer [4, 14, 15] or for solving an integrated two-layer planning problem with some simplifying assumptions, like no node hardware or wavelength granularity demands [19, 25]. Recently, Belotti et al. [6] have used a Lagrangean approach for a two-layer network design problem with simultaneous mean demand values and non-simultaneous peak demand values. Orlowski et al. [30] present several heuristics for a two-layer network design problem, which solve a restricted version of the original problem as a sub-MIP within a branch-and-cut framework. Raghavan and Stanojevic [35] consider the case where all logical links are eligible and develop a branch-and-price algorithm with respect to a fixed physical layer for the case of unprotected demands and one facility on the logical links.

In this paper, we present a mathematical model for the described planning problem with a predefined set of logical links and solve it using a branch-and-cut approach with user-defined cutting planes. To our knowledge, this is the first time that so many practically relevant side constraints are taken into account in one integrated two-layer planning model. This includes node hardware, several bit-rates on the logical links, and survivability against physical node and link failures. Despite its practical importance, survivability has not been considered in any previous integrated solution approach for two network layers. This is probably due to the high complexity of the survivable multi-layer network design problem, which is further discussed at the end of Section 2.

On the algorithmic side, we show that a branch-and-cut approach is still useful for an integrated planning of two network layers with all these side constraints, provided that the MIP solver is accelerated by problem-specific cutting plane routines. The algorithm is tested on several network instances provided by Nokia Siemens Networks. By adding a variety of strong single-layer cutting planes for both layers to the solver, we can significantly raise the dual bounds on our network instances. Especially in the unprotected case, most of the optimality gap is closed. With 1+1-protection, the problem is much harder to solve due to the increased problem size and other effects discussed in our computational results. However, the employed cutting planes turn out to be useful also with protection.

The paper is structured as follows. In Section 2, we will present and discuss our mixed-integer programming model. Section 3 describes the used cutting planes and states some known results about their strength. We show in Section 4 how to generate these inequalities during the branch-and-cut algorithm, and provide computational results in Section 5. Eventually, we draw some conclusions in Section 6.

2. Mathematical Model

We will now introduce the mixed-integer programming model on which our cutting planes are based.

Parameters. The physical network is represented by an undirected graph (V, E). The logical network is modeled by an undirected graph (V, L) with the same set of nodes and a fixed set L of admissible logical links. Each logical link represents an undirected path in the physical network. In consequence, any two nodes $i, j \in V$ may be connected by many parallel logical links corresponding to different physical paths, collected in the set $L_{ij} = L_{ji}$. Looped logical links are forbidden, i. e., $L_{ii} = \emptyset$ for all $i \in V$. Let $\delta_L(i) = \cup_{j \in V} L_{ij}$ be the set of all logical links starting or ending at i. Eventually, $L_e \subseteq L$ denotes the set of logical links containing edge $e \in E$, and likewise, $L_i \subseteq L$ refers to the set of logical links containing node $i \in V$ as an inner node.

We consider different types of capacities for logical links, physical links, and nodes. Each logical link $\ell \in L$ has a set M_ℓ of available capacity modules (corresponding to different bit-rates), each of them with a cost of $\kappa_\ell^m \in \mathbb{R}_+$ and a base capacity of $C_\ell^m \in \mathbb{Z}_+$ that can be installed on ℓ in integer multiples.

Similarly, every node $i \in V$ has a set M_i of node modules (representing different EXC types), at most one of which may be installed at i. Module $m \in M_i$ provides a switching capacity of $C_i^m \in \mathbb{Z}_+$ (e. g., in bits per second) at a cost of $\kappa_i^m \in \mathbb{R}_+$. On a physical link $e \in E$, a fiber may be installed at a cost of $\kappa_e \in \mathbb{R}_+$. Each fiber supports up to $B \in \mathbb{Z}_+$ lightpaths.

For the routing part, a set H of undirected point-to-point communication demands is given, which may be *protected* or *unprotected*. Protected demands are expected to survive any single physical node or link failure, whereas unprotected demands are allowed to fail in such a case. Each demand $h \in H$ has a source node, a target node, and a demand value d_h to be routed between these two nodes. Without loss of generality, we may assume the demands to be directed in an arbitrary way. For 1+1-protected demands, d_h refers to twice the original demand value that would have to be routed if the demand was unprotected. By adding constraints that limit the amount of flow for a protected commodity through a node or physical link to $\frac{1}{2}d_h$, it is guaranteed that at least the original demand survives any single physical link or node failure. This survivability model, called *diversification* [2], is a slight relaxation of 1+1-protection, but its solutions can often be transformed into 1+1-solutions.

From the demands, two sets K^p and K^u of protected and unprotected commodities are constructed, where $K := K^p \cup K^u$ denotes the set of all commodities. With every commodity $k \in K$ and every node $i \in V$, a net demand value $d_i^k \in \mathbb{Z}$ is associated such that $\sum_{i \in V} d_i^k = 0$. Every *protected commodity* $k \in K^p$ consists of a single 1+1-protected point-to-point demand, i.e., $d_i^k \neq 0$ only for the source and target node of the demand. In contrast, *unprotected commodities* $k \in K^u$ are derived by aggregating unprotected point-to-point demands at a common source node. Summarizing, every commodity $k \in K$ has a unique source node $s^k \in V$. Unprotected commodities may have several target nodes, whereas protected commodities have a unique target $t^k \in V$. The (undirected) emanating demand of a node $i \in V$, i. e., the total demand value starting or ending at node i, is given by $d_i := \sum_{k \in K} |d_i^k|$. The demand value d^k of a commodity is defined as the demand for k emanating from its source node, i. e., $d^k := d_{s^k}^k > 0$. Notice that for protected commodities, this value is twice the requested bandwidth to ensure survivability.

Variables. The model comprises four classes of variables representing the flow and different capacity types. First, for a logical link $\ell \in L$ and a module $m \in M_\ell$, the logical link capacity variable $y_\ell^m \in \mathbb{Z}_+$ represents the number of modules of type m installed on ℓ. For a physical link $e \in E$, the binary physical link capacity variable $z_e \in \{0, 1\}$ indicates whether e is equipped with a fiber or not. Similarly, for a node $i \in V$ and a node module $m \in M_i$, the binary variable $x_i^m \in \{0, 1\}$ denotes whether module m is installed at node i or not. Eventually, the routing of the commodities is modeled by flow variables. In order to model diversification of protected commodities, we need fractional flow variables $f^k_{\ell,ij}, f^k_{\ell,ji} \in \mathbb{R}_+$ representing the flow for commodity $k \in K$ on logical link $\ell \in L_{ij}$ directed from i to j and from j to i, respectively. For notational convenience, $f^k_\ell := f^k_{\ell,ij} + f^k_{\ell,ji}$ denotes the total flow for $k \in K$ on $\ell \in L_{ij}$ in both directions.

In our model, a flow variable $f^k_{\ell,ij}$ for commodity k and logical link $\ell \in L_{ij}$ is omitted if any of the following conditions is satisfied: (i) $j = s^k$, (ii) $k \in K^p$ and $i = t^k$, and (iii) $k \in K^p$ and ℓ contains the source or target node of k as an inner node. The first two types of variables represent flow into the unique source node or out of the unique target node of a protected commodity. They are not generated in order to reduce cycle flows in the edge-flow formulation. For aggregated unprotected commodities, we have to allow flow from one target node to another, and thus flow out of target nodes. The third type of variables would allow flow to be routed through an end-node u of a protected commodity without terminating at that node, and then back to u on another logical link. As such routings are not desired in practice, we exclude flow variables whose logical link contains an end-node of the corresponding commodity as an inner node. Again, in the unprotected case, such variables have to be admitted because commodities may consist of several aggregated demands.

Objective and Constraints. The objective and constraints of our mixed-integer programming model read as follows:

$$\min \quad \sum_{i\in V}\sum_{m\in M_i} \kappa_i^m x_i^m + \sum_{\ell\in L}\sum_{m\in M_\ell} \kappa_\ell^m y_\ell^m + \sum_{e\in E} \kappa_e z_e \tag{1}$$

$$\text{s.t.} \quad \sum_{j\in V}\sum_{\ell\in L_{ij}} (f^k_{\ell,ij} - f^k_{\ell,ji}) = d_i^k \qquad \forall\, i\in V,\ \forall\, k\in K \tag{2}$$

$$\sum_{m\in M_\ell} C_\ell^m y_\ell^m - \sum_{k\in K} f_\ell^k \geq 0 \qquad \forall\, \ell \in L \tag{3}$$

$$\sum_{\ell\in L_i} f_\ell^k + \sum_{\ell\in\delta_L(i)} \frac{1}{2} f_\ell^k \leq \frac{1}{2} d^k \qquad \forall\, i\in V,\ \forall\, k\in K^p \tag{4}$$

$$f^k_{\ell,s^k,t^k} \leq \frac{1}{2} d^k \qquad \forall\, k\in K^p,\ \ell = e = \{s^k, t^k\} \tag{5}$$

$$\sum_{m\in M_i} x_i^m \leq 1 \qquad \forall\, i \in V \tag{6}$$

$$2 \sum_{m \in M_i} C_i^m x_i^m - \sum_{\ell \in \delta_L(i)} \sum_{m \in M_\ell} C_\ell^m y_\ell^m \geq d_i \qquad \forall\, i \in V \tag{7}$$

$$B z_e - \sum_{\ell \in L_e} \sum_{m \in M_\ell} y_\ell^m \geq 0 \qquad \forall\, e \in E \tag{8}$$

$$f_{\ell,ij}^k, f_{\ell,ji}^k \in \mathbb{R}_+,\ y_\ell^m \in \mathbb{Z}_+,\ x_i^m, z_e \in \{0,1\} \tag{9}$$

The objective (1) aims at minimizing the total installation cost. The flow-conservation (2) and capacity constraints (3) describe a multi-commodity flow and modular capacity assignment problem on the logical layer. For protected commodities, the flow diversification constraints (4) restrict the flow through an intermediate node to half the demand value. In this way, the original demand is guaranteed to survive single node failures as well as single physical link failures, except for the direct physical link between source s^k and target t^k. This exception is covered by the variable bound (5). In fact, to reduce cycle flows in the LP, we set an upper bound of d^k and $\frac{1}{2}d^k$ on *all* flow variables for unprotected and protected commodities, respectively. The generalized upper bound constraints (6) guarantee that at most one node module is installed at each node. The node switching capacity constraints (7) ensure that the switching capacity of the network element installed at a node is sufficient for all traffic that can potentially be switched at that node. Since all traffic is counted twice, it is compared to twice the installed node capacity. Eventually, the physical link capacity constraints (8) make sure that the maximum number of modules on a physical link is not exceeded, and set the physical link capacity variables to 1 whenever a physical link is used.

Discussion of the model. There are three main challenges in solving this planning task using standard MIP techniques. First, lower granularity traffic has to be routed in integer capacity batches on the logical links, which in turn have to be supported by the physical network. This is a capacitated network design problem with modular integer capacities on the logical layer (see [3, 8, 27]) combined with an additional fixed-charge network design or Steiner tree problem (see [12, 13, 16, 17, 21, 32]) on the physical layer. Both types of problems are well studied and strong valid inequalities are known, but integrated approaches have been rarely considered. Second, the logical lightpath graph is complete and may even contain many parallel links corresponding to different paths on the fiber graph. This leads to a large number of integer capacity variables and an even larger number of flow variables. Even if these are fractional, the time required for solving the LP relaxations during the branch-and-cut process becomes a critical factor as the network size increases. Third, indirect interdependencies, e. g., between physical fibers and the switching capacity of a node module, are hard to detect for a black-box MIP solver.

Several particular design choices in our model deserve a brief discussion. First, we assume a fractional multi-commodity flow on the logical layer although SDH requires an integer routing in practice. This is motivated by our observation that in good solutions, the routing is often nearly integer even if this is not required, and by the fact that relaxing the integrality conditions on the flow variables significantly reduces the computation times. If an integral routing is indispensable, it can be obtained in a postprocessing step, which usually does not deteriorate the cost of the solutions very much if properly done. Notice that the lower bound computed for the model with fractional flow can also be used to assess the quality of the postprocessed integral solutions.

Second, we aggregate unprotected demands by their source node. Compared to using point-to-point commodities also in the unprotected case, this standard approach (see [8], for instance) reduces the number of commodities from $\mathcal{O}(|V|^2)$ to $\mathcal{O}(|V|)$, which leads to a much smaller ILP formulation. As every solution of the aggregated formulation can be transformed into a solution of the model with disaggregated commodities and vice versa, the aggregation does not affect the LP bound.

Third, we assume a predefined set of logical links for computational reasons. The consideration of all possible physical paths as logical links in combination with the practical side constraints and the survivability requirements would ask for a branch-and-cut-and-price approach with a nontrivial pricing problem already in the root node. Such an approach clearly can only be successful if the problem with a limited set of logical links can be solved efficiently. For a branch-and-price approach that deals with all possible logical links using a simplified model without survivability, the interested reader is referred to [35].

3. Cutting Planes

Backed by theoretical results of polyhedral combinatorics, cutting plane procedures have been proven to be a feasible approach to improve the performance of mixed integer programming solvers for many single-layer network design problems. In this section we show how an appropriate selection of these inequalities can be adapted to our problem setting. Their separation and some computational results are given in Sections 4 and 5, respectively.

3.1 Cutting Planes on the Logical Layer

On the logical layer, we consider *cutset inequalities* and *flow-cutset inequalities*. These cutting planes have, for instance, been studied in [3, 8, 11, 26, 34] for a variety of network settings (e. g., directed, undirected, and bidirected link models, single or multiple capacity modules, etc.) and have been successfully used within branch-and-cut algorithms for capacitated single-layer network design problems [7, 8, 18, 33].

To be precise, the inequalities on the logical layer are valid for the polyhedron P defined by the multi-commodity flow constraints (2) and the capacity constraints (3). That is,

$$P := \text{conv}\left\{(f,y) \in \mathbb{R}_+^{n_1} \times \mathbb{Z}_+^{n_2} \mid (f,y) \text{ satisfies } (2), (3)\right\},$$

where $n_1 := 2|K||L|$ and $n_2 := \sum_{\ell \in L} |M_\ell|$. As P is a relaxation of the model discussed in Section 2, the inequalities are also valid for that model.

We introduce the following notation. For any subset $\emptyset \neq S \subset V$ of the nodes V, let

$$L_S := \{\ell \in L \mid \ell \in L_{ij},\ i \in S,\ j \in V \setminus S\}$$

be the set of logical links having exactly one end-node in S. Furthermore, define $d_S^k := \sum_{i \in S} d_i^k \geq 0$ to be the total demand value to be routed over the cut L_S for commodity $k \in K$. By reversing the direction of demands and exchanging the corresponding flow variables, we may w. l. o. g. assume that $d_S^k \geq 0$ for all $k \in K$ (i. e., the commodity is directed from S to $V \setminus S$, or the end-nodes of k are either all in S or all in $V \setminus S$). This reduction is done implicitly in our code. More generally, let $d_S^Q := \sum_{k \in Q} d_S^k$ denote the total demand value to be routed over the cut L_S for all commodities $k \in Q$.

Mixed-integer rounding (MIR). In order to derive strong valid inequalities on the logical layer we aggregate model inequalities and apply a strengthening of the resulting base inequalities that is known as *mixed-integer rounding* (MIR). It exploits the integrality of the capacity variables. Further details on mixed-integer rounding can be found in [28], for instance.

Let $a, c, d \in \mathbb{R}$ with $c > 0$ and $\frac{d}{c} \notin \mathbb{Z}$ and define $a^+ := \max(0, a)$. Furthermore, let

$$r_{a,c} := a - c(\lceil \tfrac{a}{c} \rceil - 1) > 0$$

be the remainder of the division of a by c if $\frac{a}{c} \notin \mathbb{Z}$, and c otherwise. A function $f : \mathbb{R} \to \mathbb{R}$ is called *subadditive* if $f(a) + f(b) \geq f(a+b)$ for all $a, b \in \mathbb{R}$. The MIR function

$$F_{d,c} :\ \mathbb{R} \to \mathbb{R} :\ a \mapsto \lceil \tfrac{a}{c} \rceil\, r_{d,c} - (r_{d,c} - r_{a,c})^+$$

is subadditive and nondecreasing with $F_{d,c}(0) = 0$. If $d/c \notin \mathbb{Z}$ then $\bar{F}_{d,c}(a) := \lim_{t \searrow 0} \frac{F_{d,c}(at)}{t} = a^+$ for all $a \in \mathbb{R}$; otherwise $\bar{F}_{d,c}(a) = a$ for all $a \in \mathbb{R}$ [33]. Because of these properties, applying this function to the coefficients of a valid inequality yields another valid inequality [29]. In particular, if a valid inequality contains continuous flow variables and integer capacity variables then applying $F_{d,c}$ to its capacity coefficients and $\bar{F}_{d,c}$ to its flow coefficients yields a valid inequality. More details and explanations can be found in [33] where it is also shown that the MIR function $F_{d,c}$ is integral if a, c, and d are integral, and that $|F_{d,c}(a)| \leq |a|$ for all $a \in \mathbb{R}$. Both properties are desirable from a numerical point of view.

Cutset inequalities. Let L_S be a cut in the logical network as defined above. Obviously, the total capacity on the cut links L_S must be sufficient to accommodate the total demand over the cut:

$$\sum_{\ell \in L_S} \sum_{m \in M_\ell} C_\ell^m y_\ell^m \geq d_S^K. \tag{10}$$

Since all coefficients are nonnegative in (10) and $y_\ell^m \in \mathbb{Z}_+$, we can round down all coefficients to the value of the right-hand side (if larger). For notational convenience we assume from now on $C_\ell^m \leq d_S^K$ for all $\ell \in L_S$ and $m \in M_\ell$. Mixed-integer rounding exploits the integrality of the capacity variables. Setting $c > 0$ to any of the available capacities on the cut and applying the MIR-function $F_c := F_{d_S^K,c}$ to the coefficients and the right-hand side of (10) results in the *cutset inequality*

$$\sum_{\ell \in L_S} \sum_{m \in M_\ell} F_c(C_\ell^m) y_\ell^m \geq F_c(d_S^K). \tag{11}$$

A crucial necessary condition for (11) to define a facet for P is that the two subgraphs defined by the network cut are connected, which is trivially fulfilled if L contains logical links between all node pairs.

Flow-cutset inequalities. Cutset inequalities can be generalized to flow-cutset inequalities, which have nonzero coefficients also for flow variables. Like cutset inequalities, flow-cutset inequalities are derived by aggregating capacity and flow-conservation constraints on a logical cut L_S and applying a mixed-integer rounding function to the coefficients of the resulting inequality. However, the way of aggregating the inequalities is more general. Various special cases of flow-cutset inequalities have been discussed in [3, 8, 11, 33, 34]. Necessary and sufficient conditions for flow-cutset inequalities to define a facet of P can be found in [34].

Consider fixed nonempty subsets $S \subset V$ of nodes and $Q \subseteq K$ of commodities. Assume that logical link $\ell \in L_S$ has end-nodes $i \in S$ and $j \in V \setminus S$. We will denote by $f_{\ell,-}^k := f_{\ell,ji}^k$ inflow into S on ℓ while $f_{\ell,+}^k := f_{\ell,ij}^k$ refers to outflow from S on ℓ. We now construct a base inequality to which a suitable mixed-integer rounding function will be applied. First, we obtain a valid inequality from the sum of the flow conservation constraints (2) for all $i \in S$ and all commodities $k \in Q$:

$$\sum_{\ell \in L_S} \sum_{k \in Q} (f_{\ell,+}^k - f_{\ell,-}^k) \geq d_S^Q$$

Given a subset $L_1 \subseteq L_S$ of cut links and its complement $\bar{L}_1 := L_S \setminus L_1$ with respect to the cut, we can relax the above inequality by omitting the inflow variables and by replacing the flow by the capacity on all links in L_1:

$$\sum_{\ell \in L_1} \sum_{m \in M_\ell} C_\ell^m y_\ell^m + \sum_{\ell \in \bar{L}_1} \sum_{k \in Q} f_{\ell,+}^k \geq d_S^Q. \tag{12}$$

Again we may assume $C_\ell^m \leq d_S^K$ for all $\ell \in L_1$ and $m \in M_\ell$.

Let $c > 0$ be the capacity of a module available on the cut and define $F_c := F_{d_S^Q,c}$ and $\bar{F}_c := \bar{F}_{d_S^Q,c}$. Applying these functions to the base inequality (12) results in the *flow-cutset inequality*

$$\sum_{\ell \in L_1} \sum_{m \in M_\ell} F_c(C_\ell^m) y_\ell^m + \sum_{\ell \in \bar{L}_1} \sum_{k \in Q} f_{\ell,+}^k \geq F_c(d_S^Q). \tag{13}$$

Notice that $\bar{F}_c(1) = 1$, so the coefficients of the flow variables remain unchanged. This inequality can be generalized to a flow-cutset inequality also containing inflow-variables [33]. By choosing $L_1 = L_S$ and $Q = K$, inequality (13) reduces to the cutset inequality (11).

3.2 Cutting Planes on the Physical Layer

If the fixed-charge cost values κ_e are zero then the corresponding variables z_e can be assumed equal to 1 in any optimal solution. If, on the other hand, this cost is positive, the variables will take on fractional values in linear programming (LP) relaxations. By the demand routing requirements, we know that certain pairs of nodes have to be connected not only on the logical layer but also on the physical layer. Consequently, the variables z_e have to satisfy certain connectivity constraints. Note that information of the physical layer is combined with the demands here, skipping the intermediate logical layer.

Connectivity problems have been studied on several occasions, in particular in the context of the Steiner Tree problem and fixed-charge network design, e. g., [10, 32]. Let $S \subset V$ be a set of nodes and $\delta(S)$ the corresponding cut in the physical network. If some demand has to cross the cut then the inequality

$$\sum_{e \in \delta(S)} z_e \geq 1 \tag{14}$$

ensures that at least one physical link is installed on the cut. If a protected demand has to cross the cut, the right-hand side can even be set to 2 because the demand must be routed on at least two physically disjoint paths.

If the demand graph (defined by the network nodes and edges corresponding to traffic demands) has p connected components (usually $p = 1$) then

$$\sum_{e \in E} z_e \geq |V| - p \tag{15}$$

is valid, because the installed physical links can consist of at most p connected components as well, each one being at least a tree. If protected demands exist and the demand graph is connected, inequality (15) can be strengthened by setting the right hand side to $|V|$. If protected demands exist for all demand end nodes, this inequality is however dominated by the inequalities (14) for all demand end nodes as single node subsets.

4. Separation and Implementation

We used the branch-and-cut framework SCIP 0.90 [1] with CPLEX 10.1 [20] as the underlying LP solver to tackle the multi-layer problem introduced in Section 2. At every node of the search tree, SCIP applies various primal heuristics to compute feasible solutions, as well as built-in and application-specific separators to cut off fractional solutions. For the cutting planes described in Section 3, three *separation problems* are addressed: Given a fractional point, find a cutset inequality (11), a flow-cutset inequality (13) or one of the fixed-charge inequalities (14) and (15) cutting off this point, or decide that no such inequality exists. After calling all of its own and all user-defined separators, SCIP selects the best inequalities based on criteria such as the Euclidean distance to the current fractional point and the degree of orthogonality to the objective function. In the following we will describe the separation algorithm that we have implemented for each of the considered inequalities.

4.1 Cutset Inequalities

As explained in Section 3.1, a cutset inequality (11) is completely determined by its base inequality (10), which in turn depends only on the choice of the cut in the logical network. Our separation procedure works as follows:

1 Choose a subset S of nodes and compute the corresponding cut links L_S.

2 Compute the base inequality (10) corresponding to this logical cut.

3 For all different capacity coefficients c occurring in the base inequality, compute the cutset inequality (11) using the function $F_{d_S^K,c}$ and check it for violation.

In this way, the task reduces to finding a suitable cut in the logical network. In general, it is $\mathcal{NP}$-hard to find a cut where the cutset inequality is maximally violated, see [7]. We apply a heuristic shrinking procedure to the logical network, similar to what has been done in [7, 18, 33] for single-layer problems. Define the link weights $w_\ell := s_\ell + \pi_\ell$ where s_ℓ and π_ℓ are the slack and the dual value of the capacity constraint (3) for link ℓ with respect to the current LP solution. We iteratively shrink links with the largest weight w_ℓ, aggregating parallel logical links if necessary, until k nodes are left. Using a value of k between 2 and 6, we enumerate all cuts in the shrunken graph. The definition of w_ℓ is based on the heuristic argument that a cutset inequality is most likely to be violated if the slack of the base inequality is small. We thus want to keep links in the shrunken graph that have a small slack in the capacity constraints, i. e., we have to shrink links with a large slack s_ℓ. Since many capacity constraints are usually tight in the LP solutions, many slacks are 0. For those we use the dual values as a second sorting criterion. In addition to the described shrinking procedure we check all cutset inequalities for violation that correspond to single-node cuts, that is $S = \{i\}$ for all $i \in V$.

4.2 Flow-cutset Inequalities

For separating a flow-cutset inequality, a suitable set S of nodes, a subset Q of commodities, a capacity c, and a partition $(L_1, \bar{L}_1)$ of the cut links L_S have to be chosen. We apply two different separation heuristics. Both restrict the separation procedure to special subclasses of flow-cutset inequalities. However, already with this restriction a large number of violated inequalities is found.

The first heuristic considers commodity subsets Q that consist of a single commodity $k \in K$ and node-sets S consisting of one or two end-nodes of k. After fixing S and k and choosing an available capacity $c > 0$ on the cut, a partition of the cut links that maximizes the violation for flow-cutset inequalities is obtained by setting

$$L_1 := \left\{ \ell \in L_S \mid \sum_{m \in M_\ell} F_c(C_\ell^m)\bar{y}_\ell^m \le \sum_{k \in Q} \bar{f}_{\ell,+}^k \right\}, \tag{16}$$

where $(\bar{f}, \bar{y})$ are flow and capacity values on the logical graph in the current LP solution, see Atamtürk [3]. The calculation of L_1 is done in linear time.

The second, more time-consuming heuristic finds a most violated flow-cutset inequality for a fixed single commodity $k \in K$ and a fixed capacity c, see [3]. The crucial observation is that once k and c are fixed, the two values compared in (16) are known, and thus the partition of the potential cut links into L_1 and $\bar{L}_1$. The only remaining question is which links are part of the cut. This question can be answered in polynomial time by defining the logical link weights $w_\ell := \min\{\sum_{m \in M_\ell} F_c(C_\ell^m)\bar{y}_\ell^m, \bar{f}_{\ell,+}^k\}$ and searching for a minimum-weighted cut between the end-nodes of the commodity with respect to these weights (introducing artificial super-source and super-target nodes if necessary).

Table 1.1. Network instances used for testing cutting planes

instance	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert L\rvert$	$\lvert H\rvert$	$\lvert M_i\rvert$	$C_\ell^1, C_\ell^2, C_\ell^3$	physical cost?
Germany17	17	26	674	121	16	1, 4, 16	no
Germany17-fc	17	26	564	121	16	1, 4, 16	yes
Ring15	15	16	184	78	5	16, 64, 256	no
Ring7	7	8	32	10	5	16, 64, 256	no

4.3 Physical Layer Cutset Inequalities

The single tree inequality (15) can simply be added to the initial MIP formulation. The number of components of the demand graph is determined using depth-first search.

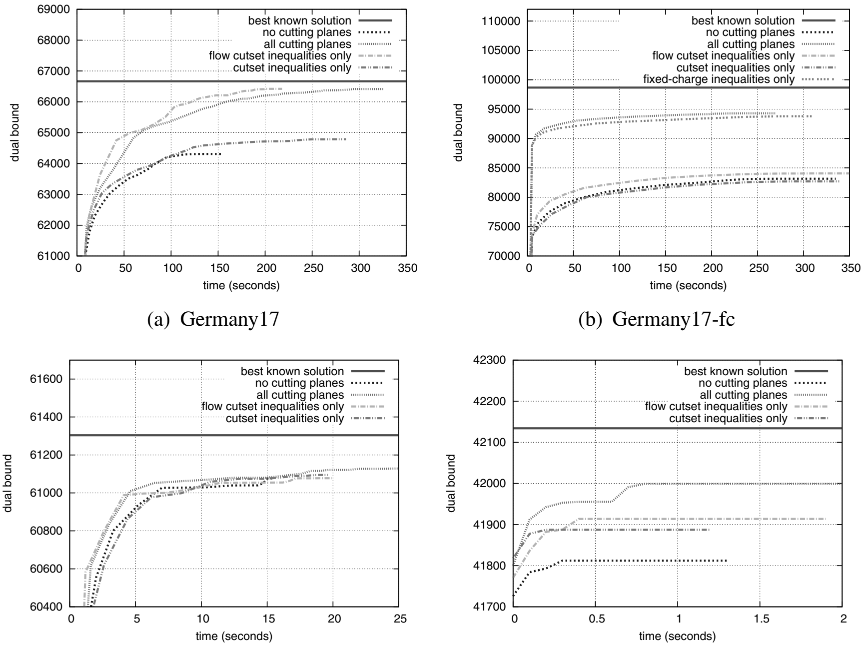
The physical cutset inequalities (14) can be separated using a min-cut algorithm. The weight of a physical link e is set to its capacity value $\bar{z}_e$ in the current LP solution, which is exactly its contribution to the left-hand side of the inequality if the link is part of the cut. Then a minimum cut with respect to these weights is searched between every pair of nodes, and the corresponding cutset inequality is tested for violation. Assuming all demands are either protected or unprotected, the right-hand side of the inequality does not depend on the cut, and thus this procedure is exact, i. e., a violated inequality exists if and only if this algorithm finds it. In addition, we test all cuts defined by single nodes $i \in V$ in each iteration, as these cuts turned out to be quite important.

5. Computational Results

5.1 Test Instances and Settings

For our computational experiments we used the network instances summarized in Table 1.1. In addition to the number of nodes, physical, and logical links, the number $|H|$ of communication demands is given from which the commodities were constructed ($|K| = |V| - 1$ if all demands are unprotected and $|K| = |H|$ if all demands are protected). Further we report the number $|M_i|$ of node modules installable at each node and the size of the installable logical link modules. Eventually, Table 1.1 indicates whether the instance has physical link cost or not. The first three instances are realistic scenarios provided by Nokia Siemens Networks, whereas the small ring network Ring7 has been constructed out of the larger instance Ring15 in order to study the effect of the cutting planes on the number of branch-and-cut nodes needed to prove optimality.

Germany17 and Germany17-fc are based on a physical 17-node German network available at SNDlib [31]. In both networks, the set of admissible



(a) Germany17 (b) Germany17-fc

(c) Ring15 (d) Ring7

Figure 1.1. Unprotected demands: dual bound at the root node

logical links consists of 3–5 short paths in the physical network between each pair of nodes. Ring15 consists of a physical ring with a chord, representing a regional subnetwork connected to a larger national network. The set of logical links consists basically of the two possible logical links for each node pair, one in each physical direction of the ring. Ring7 has been constructed from Ring15 by successively removing nodes with the smallest emanating demand value. Because in our ring instances, every node is a demand end-node and the demand graph is connected, nearly all physical links have to be used in any feasible solution. We thus do not consider ring variants with physical link cost because doing so would basically add a constant to the objective function. In all networks, up to three capacity modules corresponding to 2.5, 10, and 40 Gbit/s can be installed on each logical link, depending on its physical path length.

All computations were done on a Linux-operated machine with a 2×3 GHz Intel P4 processor and 2 GB of memory. In a first series of test runs, we assumed unprotected demands with physical fibers supporting $B = 40$ wavelengths. In a second series, we made all demands 1+1-protected, assuming $B = 80$ wavelengths in order to allow for feasible solutions with the doubled demand values. We have used extended versions of the MIP-based heuristics

Table 1.2. Number of violated cutset inequalities (11), flow-cutset inequalities (13), and fixed-charge inequalities (14) found in the root node of branch-and-bound tree without separating SCIP's Gomory and c-mir cuts

	# cuts unprotected			# cuts protected		
instance	cutset	flow-cutset	fixed-charge	cutset	flow-cutset	fixed-charge
Germany17	37	1521	-	4	940	-
Germany17-fc	34	1046	35	7	844	20
Ring15	66	652	-	26	489	-
Ring7	41	98	-	15	24	-

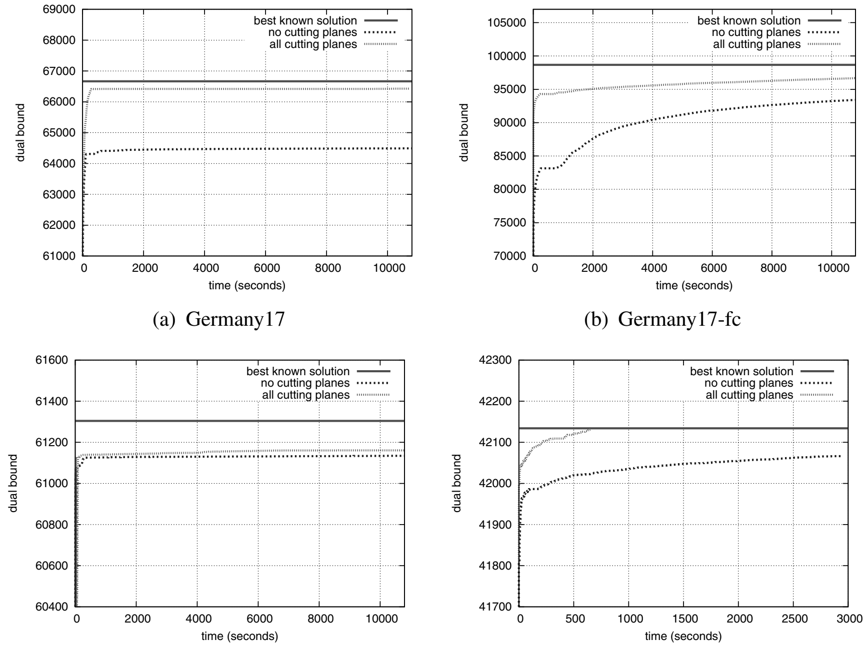
from [30] in all tests.To reduce the complexity of the problem, we also applied preprocessing and probing techniques, as described in [22].

5.2 Unprotected Demands

As cutting planes are primarily thought to increase the lower bound of the LP-relaxation, we first consider the effect of the different types of cutting planes on the lower bound at the branch-and-bound root node. We separated each of the classes cutset inequalities, flow-cutset inequalities and fixed-charge inequalities on its own as well as all together. Figure 1.1 shows the improvement over time of the lower bound in the root node of the search tree for all test instances. The solid red line at the top marks the value of the best known solution, which cannot be exceeded by the dual bound curves. The line "no cutting planes" refers to the dual bound with SCIP's built-in general-purpose cuts only.

It can be seen that in the two Germany17 instances and on the small ring network, our cutting planes reduce the gap between the lower bound and the best known solution at the root node by 50–75%. In all three problem instances, flow-cutset inequalities performed better than cutset inequalities, which is in contrast to the results presented by Raack et al. [33] for a single-layer problem. There might be several reasons for this effect. A good candidate is the structural difference between single-layer networks and the logical layer in multi-layer problems: the logical layer graph (V, L) contains edges between almost all node pairs, whereas only a few links cross a cut in single layer graphs. Further, we have implemented our cutting planes as callbacks in SCIP, whereas in [33], CPLEX was used as the underlying branch-and-cut framework, which means that different general-purpose cutting planes have been used.

For the problem Germany17-fc with physical cost, most of the optimality gap comes from the z_e variables whose values are highly fractional and close to zero in the solution of the LP-relaxation. A major part of this gap is closed by the fixed-charge inequalities that operate on the physical layer. Of course,



(a) Germany17 (b) Germany17-fc

(c) Ring15 (d) Ring7

Figure 1.2. Unprotected demands: dual bound during 3h test runs

the contribution of these inequalities changes with the ratio of the cost of the physical fiber links on the one hand and the logical wavelength links and the node hardware on the other hand.

In contrast to these three instances, the problem-specific cutting planes have only a marginal effect on the dual bound for Ring15 compared to SCIP's built-in general-purpose cuts. This is probably due to the fact that already in SCIP's default settings, the dual bound at the end of the root node is within 0.4 % of the optimal solution value, so there is not much room for improvement at all. We also observed that on this instance, our cuts seem to interfere with the c-mir and Gomory cuts separated by SCIP. Both classes are based on a mixed-integer rounding procedure similar to the one described in Section 3. With these two classes of cuts disabled in SCIP, our inequalities could reduce the relative distance between the root dual bound and the best known solution from 3.8 % to 0.4 %, thus achieving the same dual bound as SCIP's cutting planes. The number of violated cutting planes found in this setting is reported in Table 1.2 for all instances.

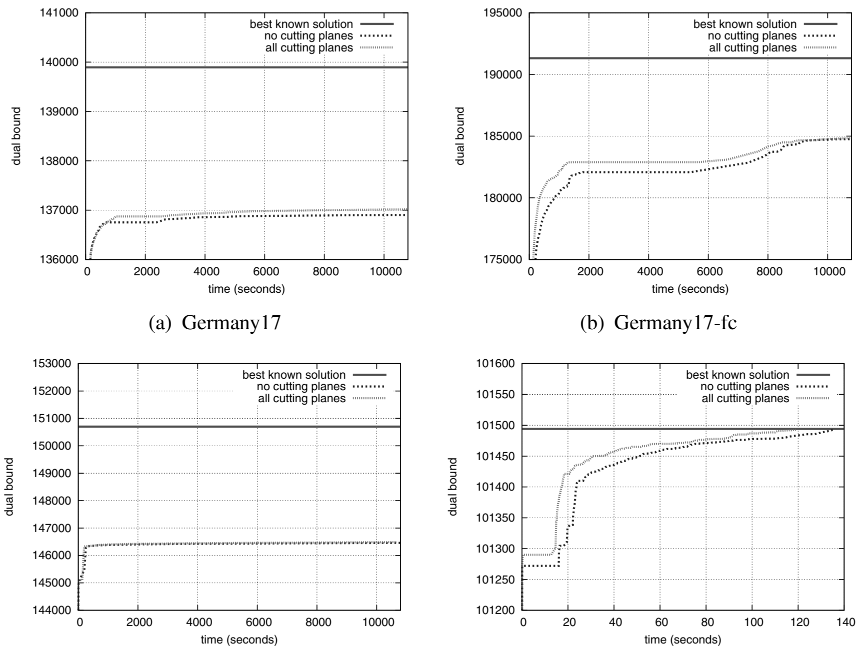
In a second study, we have investigated the lasting effect of the cutting planes on the dual bound in longer computations. Figure 1.2 shows the development of the dual bound with and without all cutting planes from Section 3 during a computation with a time limit of 3 hours for all four test instances, compared to the best known solution. Similarly to most of SCIP's own cutting planes, we separated our inequalities only at the root node of the branch-and-cut tree.

By applying all separators we could solve the problem Ring7 to optimality within 10 minutes, whereas without our cutting planes the computation was aborted after nearly one hour with a nonzero optimality gap due to the memory limit of 2 GB. The size of search tree was 1.2 million unexplored nodes at this point (and 4 million explored nodes). Figure 1.4 shows the relative gap between the dual bound and the best known solution (defined as (*bestsol* − *dual*)/*dual*), which overestimates the relative distance of the dual bound to the optimal solution value. As the figure shows, this gap could be reduced by factor 10 on Germany17 and by factor 2 on Germany17-fc by raising the lower bound only. It can be seen from Figures 1.2 and 1.4 that the dual bounds obtained with our cutting planes are very close to their maximum possible value. In fact, as also the upper bound improved in both cases, the relative gap between the dual bound and the best solution found in that specific run (as opposed to the best solution known at all) could be improved from 4 % to 0.36 % and from 12.4 % to 3.1 %, respectively. For Ring15 the improvement of the dual bound by the cutting planes was much smaller than for the other instances, probably for the reasons discussed above.

5.3 Protected Demands

In the case of protected demands, we first of all would like to point out that the problem size drastically increases compared to the unprotected case. Instead of $|V|-1$ commodities, $|H|$ commodities have to be routed, increasing the number of variables and constraints considerably. Consequently, solving the initial LP relaxation, as well as reoptimizing the LP after adding a cutting plane or a branching constraint, takes more time with protection than without.

With 1+1 protected demands, the cutting planes have only a marginal effect of the dual bound. Figure 1.3 shows the increase of the dual bound in a three hour test run with and without cutting planes (again, the solid red line at the top indicates the best known solution value). It can be seen that the dual bound always increases, but only by a very limited amount. Figure 1.4 shows the corresponding change in the relative gap between the dual bound after three hours and the best known solution. More detailed investigations revealed that the small progress is mainly due to the strength of the general-purpose c-mir and Gomory cuts generated by SCIP. Experiments where these cuts were turned off showed that our inequalities still contribute significantly to closing the optimality gap at the root node. Table 1.2 shows the number of violated inequalities found at the root node in this setting. Only slightly lower



(a) Germany17 (b) Germany17-fc

(c) Ring15 (d) Ring7

Figure 1.3. Protected demands: lower bound in 3h test runs

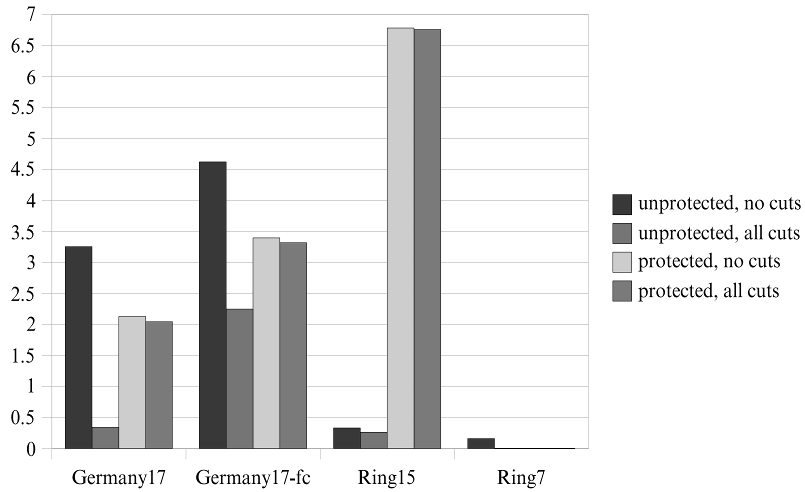


Figure 1.4. Relative gap (in %) between best dual bound after 3h and best known solution

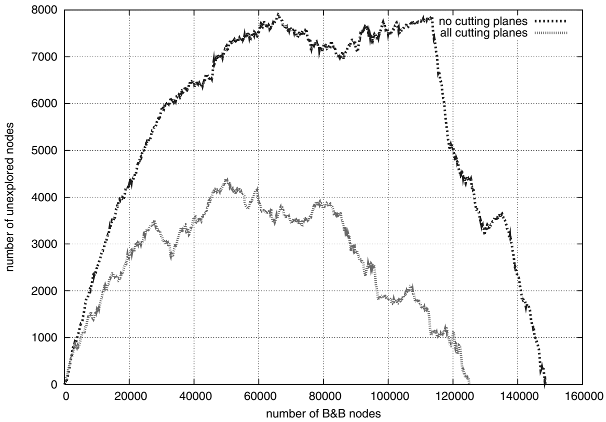


Figure 1.5. Number of unexplored branch-and-cut nodes on the protected Ring7 network

numbers of violated inequalities are found with c-mir and Gomory cuts turned on, but their impact on the dual bound is limited in such a case, cf. Figure 1.3.

The strength of the general-purpose cuts originates from the potential to include all inequalities from the original formulation, as well as cutting planes added later in the solution process. In contrast, our cutting planes only take capacity and flow conservation constraints into account. The inclusion of survivability requirements into the generation of cutset and flow-cutset inequalities might accelerate the increase of the lower bound compared to SCIP. For this, the polyhedral studies of Bienstock and Muratore [9] and of Balakrishnan et al. [5] for single layer survivability network design could be a good starting point. We suspect that cuts that make use of such problem-specific information will outperform the general-purpose cuts of SCIP, as in the unprotected case.

Nevertheless, the cutset inequalities and flow-cutset inequalities seem to have a lasting effect on the performance of the branch-and-bound algorithm as can be shown for the small ring network Ring7. This instance could be solved to optimality in both cases. But as Figure 1.5 shows, the maximum number of unexplored nodes in the search tree was roughly halved by our cutting planes, even though they were added only in the root node. Moreover, optimality was proven about 13 % faster (cf. Figure 1.3(d)) and with 16 % less nodes.

6. Conclusions

In this work, we have presented a mixed-integer programming model for a two-layer SDH/WDM network design scenario. The model includes many practically relevant side constraints like many parallel logical links, various bit-rates, node capacities, and survivability with respect to physical node and link failures. To accelerate the solution process for this planning task, we have applied a variety of network design specific cutting planes that are known to be strong in single-layer network design to either of the two layers, namely cutset inequalities and flow-cutset inequalities on the logical layer and fixed-charge inequalities on the physical one. These cutting planes have been used as callbacks within the branch-and-cut framework SCIP and tested on several realistic planning scenarios provided by Nokia Siemens Networks.

With unprotected demands, our cutting planes significantly raised the lower bounds until close to the optimal solution value. With 1+1 protection against physical failures, they also helped to improve the dual bounds, but less than in the unprotected case. This is partly due to the fact that with protection, many of our cutting planes were already found by SCIP alone, and partly due to the impact of the survivability constraints on the structure of the polyhedron. We expect that adapting previous results for survivable single-layer network design to the multi-layer setting could further raise the lower bound in these cases. Moreover, new classes of specific multi-layer cuts have to be found for multi-layer problems with protected demands.

References

[1] T. Achterberg. *Constraint Integer Programming*. PhD thesis, Technische Universität Berlin, 2007. `http://opus.kobv.de/tuberlin/volltexte/2007/1611/`.

[2] D. Alevras, M. Grötschel, and R. Wessäly. A network dimensioning tool. ZIB Technical Report SC-96-49, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 1996.

[3] A. Atamtürk. On capacitated network design cut-set polyhedra. *Mathematical Programming*, 92:425–437, 2002.

[4] G. Baier, T. Engel, A. Autenrieth, and P. Leisching. Mehrperiodenplanung optischer Transportnetze. In *7. ITG-Fachtagung Photonische Netze, Leipzig, Germany*, volume 193, pages 153–160. VDE-Verlag, April 2006.

[5] A. Balakrishnan, T. Magnanti, J. Sokol, and Y. Wang. Modeling and solving the single facility line restoration problem. Technical report, MIT Press, 1998.

[6] P. Belotti, A. Capone, G. Carello, F. Malucelli, F. Senaldi, and A. Totaro. Design of multi-layer networks with traffic grooming and statistical multiplexing. In *Proceedings of the 3rd International Network Optimization Conference (INOC 2007), Spa, Belgium*, April 2007.

[7] D. Bienstock, S. Chopra, O. Günlük, and C.Y. Tsai. Mininum cost capacity installation for multicommodity flows. *Mathematical Programming*, 81:177–199, 1998.

[8] D. Bienstock and O. Günlük. Capacitated network design – polyhedral structure and computation. *INFORMS Journal on Computing*, 8(3):243–259, 1996.

[9] D. Bienstock and G. Muratore. Strong inequalities for capacitated survivable network design problems. *Mathematical Programming*, 89(1):127–147, 2000.

[10] S. Chopra. On the spanning tree polyhedron. *Operations Research Letters*, 8:25–29, 1989.

[11] S. Chopra, I. Gilboa, and S. T. Sastry. Source sink flows with capacity installation in batches. *Discrete Applied Mathematics*, 86:165–192, 1998.

[12] S. Chopra and M. Rao. The Steiner tree problem I: Formulations, compositions and extension of facets. *Mathematical Programming*, 64(2):209–229, 1994.

[13] S. Chopra and M. Rao. The Steiner tree problem II: Properties and classes of facets. *Mathematical Programming*, 64(2):231–246, 1994.

[14] G. Dahl, A. Martin, and M. Stoer. Routing through virtual paths in layered telecommunication networks. *Operations Research*, 47(5):693–702, 1999.

[15] M. Dawande, R. Gupta, S. Naranpanawe, and C. Sriskandarajah. A traffic-grooming algorithm for wavelength-routed optical networks. *INFORMS Journal on Computing*, 2006. To appear.

[16] M. Grötschel, C. Monma, and M. Stoer. Facets for polyhedra arising in the design of communication networks with low-connectivity constraints. *SIAM Journal on Optimization*, 2(3):474–504, August 1992.

[17] M. Grötschel, C. Monma, and M. Stoer. Design of survivable networks. volume Networks of *Handbooks in Operations Research and Management Science*, chapter 10, pages 617–672. North-Holland, Amsterdam, 1993.

[18] O. Günlük. A branch-and-cut algorithm for capacitated network design problems. *Mathematical Programming*, 86:17–39, 1999.

[19] H. Höller and S. Voss. A heuristic approach for combined equipment-planning and routing in multi-layer SDH/WDM networks. *European Journal of Operational Research*, 171(3):787–796, June 2006.

[20] ILOG CPLEX Division, 889 Alder Avenue, Suite 200, Incline Village, NV 89451, USA. *CPLEX 10.1 Reference Manual*, 2006.

[21] T. Koch and A. Martin. Solving steiner tree problems in graphs to optimality. *Networks*, 32(3):207–232, 1998.

[22] A. M. C. A. Koster, S. Orlowski, C. Raack, G. Baier, and T. Engel. Single-layer cuts for multi-layer network design problems. ZIB Report ZR-07-21, Konrad-Zuse-Zentrum für Informationstechnik Berlin, August 2007.

[23] A. M. C. A. Koster and A. Zymolka. Minimum converter wavelength assignment in all-optical networks. In *Proceedings of ONDM 2004*, pages 517–535, Ghent, Belgium, 2004. The 8th IFIP Working Conference on Optical Network Design & Modelling.

[24] A. M. C. A. Koster and A. Zymolka. Tight LP-based lower bounds for wavelength conversion in optical networks. *Statistica Neerlandica*, 61(1):115–136, 2007.

[25] E. Kubilinskas and M. Pióro. An IP/MPLS over WDM network design problem. In *Proceedings of the 2nd International Network Optimization Conference (INOC 2005), Lisbon, Portugal*, volume 3, pages 718–725, March 2005.

[26] T. L. Magnanti and P. Mirchandani. Shortest paths, single origin-destination network design and associated polyhedra. *Networks*, 33:103–121, 1993.

[27] T. L. Magnanti, P. Mirchandani, and R. Vachani. Modelling and solving the two-facility capacitated network loading problem. *Operations Research*, 43:142–157, 1995.

[28] H. Marchand and L. A. Wolsey. Aggregation and mixed integer rounding to solve MIPs. *Operations Research*, 49(3):363–371, 2001.

[29] G. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. John Wiley & Sons, 1988.

[30] S. Orlowski, A. M. C. A. Koster, C. Raack, and R. Wessäly. Two-layer network design by branch-and-cut featuring MIP-based heuristics. In *Proceedings of the 3rd International Network Optimization Conference (INOC 2007), Spa, Belgium*, April 2007.

[31] S. Orlowski, M. Pióro, A. Tomaszewski, and R. Wessäly. SNDlib 1.0–Survivable Network Design Library. In *Proceedings of the 3rd International Network Optimization Conference (INOC 2007), Spa, Belgium*, April 2007. `http://sndlib.zib.de`.

[32] F. Ortega and L. A. Wolsey. A branch-and-cut algorithm for the single-commodity, uncapacitated, fixed-charge network flow problem. *Networks*, 41:143–158, 2003.

[33] C. Raack, A. M. C. A. Koster, S. Orlowski, and R. Wessäly. Capacitated network design using general flow-cutset inequalities. ZIB Report 07-14, Konrad-Zuse-Zentrum für Informationstechnik Berlin, June 2007. Submitted to Networks.

[34] C. Raack, A. M. C. A. Koster, and R. Wessäly. On the strength of cut-based inequalities for capacitated network design polyhedra. ZIB-Report 07-08, Konrad-Zuse-Zentrum für Informationstechnik Berlin, June 2007.

[35] S. Raghavan and D. Stanojević. WDM optical design using branch-and-price. Robert H. Smith School of Business, University of Maryland, April 2007.