DIPLOMARBEIT

MIP Models for (Hop-Constrained) Connected Facility Location

ausgeführt am Institut für
Statistik und Decision Support
der Universität Wien

unter der Anleitung von
Univ.-Prof. Mag. Dr. Immanuel Bomze und Mag. Dr. Ivana Ljubić

eingereicht an der
Technischen Universität Wien durch

Stefan Gollowitzer
Georg Siglgasse 9
A-1090 Wien

Wien, am 05. Dezember 2009

Abstract

The first part of this thesis comprises the first theoretical and computational study on mixed integer programming (MIP) models for the connected facility location problem (ConFL). ConFL combines facility location and Steiner trees: given a set of customers, a set of potential facility locations and some inter-connection nodes, ConFL searches for the minimum-cost way of assigning each customer to exactly one open facility, and connecting the open facilities via a Steiner tree. The costs needed for building the Steiner tree, facility opening costs and the assignment costs need to be minimized.
We model ConFL using eight compact and two exponential mixed integer programming formulations. We also show how to transform ConFL into the Steiner arborescence problem. A full hierarchy between the models is provided.
An extensive computational study is based on two benchmark sets of randomly generated instances with up to 1,300 nodes and 115,000 edges. We empirically compare the presented models for ConFL with respect to the quality of obtained bounds and the corresponding running time. We report optimal values for all but 16 instances for which the obtained gaps are below 0.6%.

In the second part of this work we extend the definition of ConFL to model reliability constraints of the end-user's connections. We develop 15 mixed integer programming models for this problem. Some of these models are extensions from corresponding models for the ConFL, some extend ideas for related problems like the Minimum Spanning or Steiner Tree problem with hop constraints. We provide a hierarchy of the proposed models by comparing the relative quality of the corresponding linear programming lower bounds. We also show how the Hop Constrained ConFL can be modelled as ConFL or Steiner Tree problem on a layered graph.

Acknowledgements

Thank you, Catherine. For being there.
Thank you, my parents. For your endless patience. For your unconditional support.
Thank you, Ivana. For your trust in my work. For the wonderful music. And, finally, for your programming skills.

Thank you, Prof. Bomze. For the kind welcome at your institute.
Thank you, everyone at DTU. For teaching me OR.

Thank you, Markus Chimani, for the proof of Lemma 8b.
Thank you, Luis Gouveia, for very useful discussions on topics related to HCMST and HCSTP.
Thank you, Martin, for your brains.
Thank you, my friends, for all the laughter we had. It kept me going.

Contents

Chapter 1

Introduction

Improving the quality of broadband connections is nowadays one of the highest priorities of telecommunication companies. Solutions are sought that search for the optimal way of "pushing" rapid and high-capacity fiber-optic networks closer to the customers. Developing respective models and answering questions related to the design of "last-mile" networks defines a new challenging area of computer science and operations research. The *Connected Facility Location Problem* (ConFL) models the following telecommunication network design problem: Traditional wired local area networks require copper cable connections between end users. To reduce the signal loss, these lines are limited by a maximum distance. To increase the quality of internet communications, telecommunication companies decide to partially or completely replace the existing copper connection by fiber-optic cables. In order to do so, different strategies, known as *fiber-to-the-home* (FTTH), *fiber-to-the-node* (FTTN), *fiber-to-the-curb* (FTTC) or *fiber-to-the-building* (FTTB), are applied.
ConFL models the FTTN / FTTC strategy: Fiber optic cables run to a cabinet serving a neighborhood. End users connect to this cabinet using the existing copper connections. Expensive switching devices are installed in these cabinets. The problem is to minimize the costs by determining positions of cabinets, deciding which customers to connect to them, and how to reconnect cabinets among each other and to the backbone.

The remainder of this thesis is organized as follows:
In the following chapter will discuss the exact definition of the Connected Facility Location problem and provide an exhaustive literature review on the topic. In Chapter 3 we propose ten mixed integer programming models for ConFL and we show a transformation of ConFL into the Steiner Arborescence (SA) problem. We provide a full hierarchy of the models based on the theoretical comparison of the quality of their lower bounds. In Chapter 4 we give a short description of the Branch-and-Cut framework used for the computational experiments and describe the test instances used. We report the results of our computational experiments. The Hop Constrained ConFL will be discussed in Chapter 5. We provide no less than 15 mixed integer programming models. We compare them with respect to the lower bounds given by solving the respective LP relaxations and provide a hierarchy of the formulations considered. Finally, we give some concluding remarks on the results and on possible topics of future research.

Chapter 2

Preliminaries

In this chapter we discuss the different definitions for the Connected Facility Location problem proposed in the literature. We describe the assumptions made throughout this thesis. Then, we give a literature survey on ConFL and related problems.

2.1 What is Connected Facility Location? - Problem Definition

Gupta et al. [25] define the Connected Facility Location problem as follows: We are given a graph $G = (V, E)$ with a set of customers ($R \subseteq V$), a set of facilities ($F \subseteq V$) and a set of Steiner nodes ($\tilde{S} \subseteq V$) such that $\tilde{S} \cap F = \emptyset$. For all $e \in E$ we are given an edge cost $c_e \geq 0$ and for all $i \in F$ we are given facility opening costs $f_i \geq 0$. Then ConFL consists of finding an assignment of each customer to exactly one facility and connecting these facilities via a Steiner tree. Thereby, assignment costs $c_{ij}, i \in F, j \in R$ are given as the shortest path distance between i and j in G.
The overall costs in this problem are defined as $\sum_{j \in R} d_j c_{i(j)j} + \sum_{i \in \mathcal{F}} f_i + \sum_{e \in T} M c_e$, where $d_j \geq 1$ is demand of customer j, $i(j)$ denotes the facility serving j, $\mathcal{F}$ is the set of open facilities, T is the Steiner tree connecting open facilities and $M \geq 1$ is a constant.
Let $S = \tilde{S} \cup F$ denote the set of *core* nodes. Then we can make the following

Observation 1. *Consider a ConFL instance as defined above, where $S \cap R \neq \emptyset$. Without loss of generality, we can transform this instance into an equivalent one in which: a) $\{S, R\}$ is a non-trivial partition of V and b) all customer demands are equal to one.*

The first transformation can easily be done by replacing all the nodes $u \in S \cap R$, with a pair of nodes, $u_1 \in S$ and $u_2 \in R$, connecting all $i \in S$, core neighbors of u, to u_1, and all $i \in F$, facility neighbors of u to u_2, without changing the edge/assignment costs. Finally, if $u \in F \cap R$, we need to connect customer neighbors to u_1 and add the service link $\{u_1, u_2\}$ into E, set its costs to zero and define $f_{u_1} = f_u$.
Demands different from 1 can be set to 1 by adapting the respective assignment costs. We replace c_{ij} by $d_j c_{ij}$ for all $j \in R$, $i \in F$ and reflect the demand in the cost structure implicitly [36]. Alternatively, we can make d_j copies of customer j, each with demand equal to one (see, e.g., [14]).
For the development of approximation algorithms there are two usual assumptions: The parameter M is used to distinguish between "cheap" assignment and "expensive" core network edges, and c is assumed to be a metric, i.e. the costs satisfy the triangle inequality. As we will see later, both these assumptions are not necessary in our approaches. Therefore, we concentrate on a general cost structure.

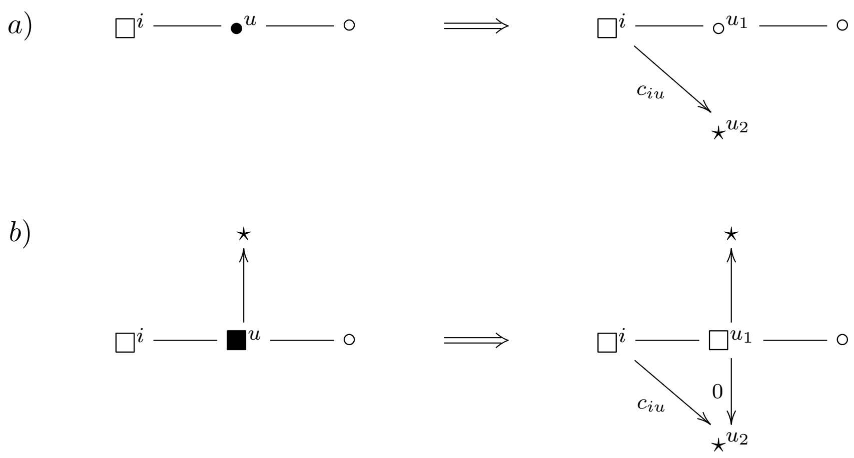


Figure 2.1: Transformations of nodes a) $u \in \tilde{S} \cap R$ and b) $u \in F \cap R$ where $\star \in R$, $\square \in F$, $\circ \in S$, $\blacksquare \in F \cap R$ and $\bullet \in \tilde{S} \cap R$

Definition 1 (ConFL)**.** For a given undirected graph (V, E) with edge costs $c_e \geq 0, e \in E$, facility opening costs $f_i \geq 0, i \in F$, a disjoint partition $\{S, R\}$ of V with $R \subset V$ being the set of customers, $S \subset V$ the set of possible Steiner nodes and $F \subseteq S$ the set of facilities, in the *Connected Facility Location* problem we search for a subset of open facilities such that:

- each customer is assigned to the closest open facility,
- a Steiner tree connects all open facilities, and
- the sum of assignment, facility opening and Steiner tree costs is minimized.

Optionally, a root $r \in F$ may be considered as an open facility always included in the network. In that case, we speak of the *rooted ConFL*. Obviously, every optimal ConFL solution will be a tree where customers (and possibly the root r) are leaves. In the telecommunications field a "central office" connecting to the backbone network is often predefined and may be considered as a root node active in any feasible solution. Therefore, in the following we assume that the root is given in advance. In Section 3 we show how to solve unrooted instances.

2.2 Literature Review

The Connected Facility Location Problem has lately started to attract stronger interest in the scientific community. Compared to some closely related problem classes, there is just a small number of papers on the topic. A large share of publications about ConFL comes from the computer science community who present approximation algorithms of different kinds and qualities. The operations research community has developed a small number of heuristic methods.

2.2.1 Approximation Algorithms

A major part of the publications about ConFL concentrate on approximation algorithms. However, not a single one contains computational results. Thus, no conclusion can be drawn to the practical applicability of the described algorithms.
Karger and Minkoff [29] describe an adapted version of the Steiner tree problem. They consider the distribution of single data items from a root to a set of clients. It is not known beforehand

which clients demand the data item in question. For each client, there is a known probability to become active and request data. Consider caching nodes at a certain cost, i.e. nodes storing the demanded data for resending it to clients becoming active later-on. The problem of finding a tree with minimal expected cost is equal to the Connected Facility Location Problem. The authors gather the clients into clusters connected to a common facility. Second, they connect these facilities by a Steiner tree. They present a bicriterion approximation algorithm producing a solution of at most 41 times the optimum cost.
Krick et al. [33] present a similar problem as the one in [29], although in an other context. They consider a computer network where clients (corresponding to customers) issue read and write requests. The data for the requests is stored in memory modules (facilities) at a certain cost. Read and write requests are served by the nearest installed memory module for the respective client. To keep data consistent throughout the network, all other memory modules are updated with the latest version. This requires connectivity between the memory modules. Krick et al. give a constant approximation algorithm with a larger constant than the one given by Karger and Minkoff [29].
In the context of reserving bandwidth for virtual private networks, Gupta et al. [25] introduce the term Connected Facility Location. They give a proof for ConFL to be NP-hard. They present a first cut-based integer programming formulation. Their formulation will be described and discussed in detail in Section 3.2.1. Their approximation algorithm for ConFL has a constant factor of 10.66. For the closely related *rent-or-buy problem* (RoB), in which all nodes are potential facilities with opening costs equal to 0, the algorithm gives an approximation factor of 9.002.
Swamy and Kumar [48] develop a primal-dual approximation algorithm for ConFL, RoB and k-ConFL. The latter comprises the additional restriction that in an optimum solution at most k facilities can be opened. The integer programming formulation used is the same as in Gupta et al. [25]. As results the authors give approximation ratios of 8.55, 4.55 and 15.55 for ConFL, RoB and k-ConFL, respectively.
The approximation factors have been successively improved in Jung et al. [28] and Williamson and van Zuylen [51]. Finally, Eisenbrand et al. [14] combine approximation algorithms for the basic facility location problem and the connectivity problem of the opened facilities by running a what they call *core detouring scheme*. The randomised version of the approximation algorithm gives new best expected approximation ratios for ConFL (4.00), RoB (2.92) and k-ConFL (6.85). The ratios for the de-randomised version are 4.23, 3.28 and 6.98 respectively.

2.2.2 Heuristics and Exact Methods

Ljubić [36] describes a hybrid heuristic combining Variable Neighborhood Search with a reactive tabu search method. The author compares it with an exact branch-and-cut approach. The corresponding integer programming model for the branch-and-cut approach will be explained in detail and compared to other formulations in Section 3. Ljubić [36] also presents two classes of test instances as a result of combining Steiner tree and uncapacitated facility location instances. Results for these instances with up to 1300 nodes are presented.
Tomazic and Ljubić [49] present a Greedy Randomized Adaptive Search Procedure (GRASP) for the ConFL problem. Results for a new set of test instances with up to 120 nodes (facilities plus customers) are presented.

2.2.3 Related Problems

The Connected Facility Location problem is a combination of two other well-known problems in graph theory. These are the Steiner tree problem (STP) and the Uncapacitated Facility

Location problem (UFL). ConFL contains them both as special cases. For a set of possible facility locations connected to a root via a star, we have UFL. In case each customer can only be served by one predefined facility, we know the set of facilities that needs to be opened in advance. Thus, we then have an STP to solve.

Steiner Tree Problem The Steiner tree problem has been of interest for decades and an enormous number of authors have been working on it. A number of exact methods, heuristics, approximation algorithms and polyhedral approaches have been studied for the classical STP. Good surveys are Hwang, Richards, and Winter [27], Hwang and Richards [26], Maculan [40] and Winter [52]. However, we have a certain interest in a less well known generalization of the STP, the STP with hop constraints.

The Steiner tree problem with hop constraints (HCSTP) There has been intense research on the Minimum Spanning Tree problem with hop constraints (HCMST), a special case of the HCSTP where each node in the graph is a terminal. A recent survey for the HCMST can be found in Dahl et al. [11].
Much less has been said about the Steiner tree problem with hop constraints: The problem was first mentioned by Gouveia [21], who develops a strengthened version of a multi-commodity flow model for the Minimum Spanning and Steiner tree problem. The LP lower bounds of this model are equal to the ones from a Lagrangean relaxation approach of a weaker MIP model introduced in [20]. Results for instances with up to 100 nodes and 350 edges are presented.
Voß [50] presents MIP formulations based on Miller-Tucker-Zemlin subtour elimination constraints. The formulation is then strengthened by disaggregation of the variables indicating used arcs. The author develops a simple heuristic to find starting solutions and improves these with an exchange procedure based on tabu search. Numerical results are given for instances with up to 2500 nodes and 65000 edges.
Gouveia [22] gives a survey of hop-indexed tree and flow formulations for the hop constrained spanning and Steiner tree problem.
Costa, Cordeau, and Laporte [9] give a comparison of three heuristic methods for a generalisation of the HCSTP, namely the Steiner tree problems with revenues, budget and hop constraints (STPRBH). The considered methods comprise a greedy algorithm, a destroy-and-repair method and a tabu search approach. Computational results are reported for instances with up to 500 nodes and 12500 edges.
In [10] Costa, Cordeau, and Laporte introduce two new models for the STPRBH. They are both based on the sub-tour elimination constraints described by [12] and a set of hop constraints of exponential size. They provide a theoretical and computational comparison with the two models based on Miller-Tucker-Zemlin constraints presented in [22, 50].

Rent-or-buy Problem (RoB) The rent-or-buy problem is often viewed as a special case of the ConFL problem. In the RoB problem facility opening costs are 0 and facilities can be opened anywhere. Thus, also customer nodes can act as facilities and have other customers assigned to them. The cost for each edge in a solution to the RoB depends on its adjacent nodes. If an edge is used to assign a customer to a facility, only assignment costs are incurred. If an edge connects two facilities, a comparatively higher cost, i.e. M times the assignment cost, has to be paid for.

The (general) Steiner tree-star problem ((G)STS) The Steiner tree-star problem was introduced by Lee et al. [34]. It arises in the design of some specific telecommunication networks, where bridging occurs. The Steiner tree-star problem is the following: Given a graph with

disjoint sets of possible facility nodes and customers, we want to find a minimum cost tree such that each customer is assigned to a facility and that all open facilities are connected by a Steiner tree. Facility opening costs are incurred for any facility in the solution tree, regardless of whether any customers are assigned to it or not.
Exact methods to solve the STS problem have been described by Lee et al. [34, 35], a tabu search based heuristic was developed by Xu et al. [54]. Khuller and Zhu [30] introduced the *general* Steiner tree-star problem. There, the sets of possible facilities and customers must not be disjoint. Nodes can act in both ways and an open facility can serve the customer in its own place at no additional cost. Khuller and Zhu [30] derive two approximation algorithms for the general STS with approximation factors of 5.16 and 5 respectively.

General Connected Facility Location (GConFL) Bardossy and Raghavan [5] develop a dual-based local search (DLS) heuristic for a family of problems combining facility location decisions with connectivity requirements, namely the (general) Steiner tree-star, ConFL and RoB. They introduce the general ConFL problem, into which any of the aforementioned 4 problem classes can be transformed. The presented DLS heuristic works in two phases. After applying dual-ascent in order to get a lower and upper bound in the first phase, in the second phase a local search procedure is carried out on the facilities and Steiner nodes selected before. Computational results for instances with up to 100 nodes are presented. Running time and the quality of solutions of Ljubić' VNS heuristic and DLS are compared for the set of instances introduced in [36].

Chapter 3

(M)ILP Formulations for ConFL

It is well known that the MIP formulations for Steiner trees and related problems provide stronger lower bounds when defined on directed graphs (see, e.g., [8, 17]). In this chapter we will first describe how to transform undirected instances for ConFL into directed ones. A range of (M)ILP formulations for the ConFL will be presented afterwards. As the exponential size formulations are hard to implement by means of a modeling language, various compact MIP formulations will be described in this section as well. They are either flow formulations or based on sub-tour elimination constraints. We conclude the chapter with some polyhedral results and provide a full hierarchy of 12 different models for ConFL.

3.1 Transformation Into Directed Graphs

Throughout this paper, an arc from i towards j will be denoted by ij, and the corresponding undirected edge by $\{i, j\}$. Let (V, E) be a given instance of ConFL with $\{S, R\}$ being a partition of V and $F \subseteq S$. This instance can be transformed into a bidirected instance (V, A) as follows (cf. [49]):

- Replace core edges $e \in E$ with $e = \{i, j\}, i, j \in S$ by two directed arcs $ij \in A$ and $ji \in A$ with cost $c_{ij} = c_{ji} = c_e$.
- Replace assignment edges $e \in E$ with $e = \{j, k\}, j \in F, k \in R$ by an arc $jk \in A$ with cost $c_{jk} = c_e$ respectively.

Rooting Unrooted Instances To obtain an optimal solution for a directed, unrooted instance (V, A) by solving a model for rooted instances we adapt the input instance and the corresponding model as follows:

- Expand the set of facilities F by adding an artificial root r to $V' = V \cup \{r\}$ with cost $f_r = 0$.
- Expand the set of arcs by adding an arc rj for all core nodes $j \in F$ with $c_{rj} = 0$.
- Limit the number of arcs emanating from the root r to 1, e.g. add the additional constraint $\sum_{j \in F} x_{rj} \leq 1$.

In the remainder of this paper we will refer to the Connected Facility Location problem on directed graphs as the following:

Definition 2 (ConFL on directed graphs)**.** We are given a directed graph (V, A) with edge costs $c_{ij}, ij \in A$, facility opening costs $f_i, i \in F$ and a disjoint partition $\{S, R\}$ of V with $R \subset V$ being the set of customers, $S \subset V$ the set of possible Steiner tree nodes, $F \subset S$ the set of facilities, and the root node $r \in F$. Find a subset of open facilities such that

- each customer is assigned to exactly one open facility,
- a Steiner arborescence rooted in r connects all open facilities, and
- the cost defined as the sum of assignment, facility opening and Steiner arborescence cost, is minimized.

To model the problem, we will use the following binary variables:

$$x_{ij} = \begin{cases} 1, & \text{if } ij \text{ belongs to the solution} \\ 0, & \text{otherwise} \end{cases} \quad \forall ij \in A \qquad z_i = \begin{cases} 1, & \text{if } i \text{ is open} \\ 0, & \text{otherwise} \end{cases} \quad \forall i \in F$$

We will use the following notation: $A_R = \{ij \in A \mid i \in F, j \in R\}$, $A_S = \{ij \in A \mid i, j \in S\}$. Furthermore, for any $W \subset V$ we denote by $\delta^-(W) = \{ij \in A \mid i \notin W, j \in W\}$ and $\delta^+(W) = \{ij \in A \mid i \in W, j \notin W\}$.
The examples described throughout this thesis use the following symbols: ■ represents the root node, ○ represents a Steiner node. □ represents a facility, ⋆ represents a customer. Superscripts mark node labels.
Arc costs different from 1 are displayed next to the respective arc. Facility opening and assignment costs are all 1 in all examples. All the values of facility node variables stated in the descriptions below refer to optimal LP solutions. The core network is presented as undirected graph, except in Example 5.

3.2 Models for Connected Facility Location

3.2.1 Cut-Based Formulations

In the literature there are two different exponential size formulations for ConFL. They are both based on cuts and differ in strength.

Cut Set Formulation of Gupta et al. [25] Gupta et al. [25] first introduced an undirected ILP formulation for ConFL. To ensure comparability, a directed version will be presented here. One might think of any ConFL solution as a Steiner arborescence rooted at r with customers as leaves and with node weights that need to be payed for any node that is adjacent to a customer. Therefore, instead of requiring connectivity among open facilities and assignment of customers to open facilities, we are going to ask for the solution that ensures a directed path between r and any customer $j \in R$, using the arcs from A.
The cut-based model reads then as follows:

$$
\begin{aligned}
(CUT_R) \quad & \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i \\
\text{s.t.} \quad & \sum_{uv \in \delta^-(U)} x_{uv} \geq \sum_{j \in U : jk \in A_R} x_{jk} \quad \forall U \subseteq S \setminus \{r\}, U \cap F \neq \emptyset,\ \forall k \in R \qquad (3.1) \\
& \sum_{jk \in A_R} x_{jk} = 1 \quad \forall k \in R \qquad (3.2) \\
& x_{jk} \leq z_j \quad \forall jk \in A_R \qquad (3.3) \\
& z_r = 1 \qquad (3.4) \\
& x_{ij} \in \{0,1\} \quad \forall ij \in A \qquad (3.5) \\
& z_i \in \{0,1\} \quad \forall i \in F \qquad (3.6)
\end{aligned}
$$

The objective comprises the cost for the Steiner arborescence ($\sum_{ij \in A_S} x_{ij} c_{ij}$), the cost to connect customers to facilities (that we also refer to as *assignment cost*, i.e. $\sum_{ij \in A_R} x_{ij} c_{ij}$) and the facility opening cost ($\sum_{i \in F} z_i f_i$). Constraints (3.2) ensure that every customer is connected to at least one facility, constraints (3.3) ensure that each facility is opened if customers are assigned to it, equation (3.4) defines the root node. Inequalities (3.1) represent the set of connectivity cuts. For every subset $U \subseteq S \setminus \{r\}$ and for each customer $k \in R$, an open arc from a facility in U toward j, necessitates a directed path from r towards U. Constraints (3.2) can be replaced by inequality in case that $c_{ij} > 0$, for all $ij \in A_R$. Furthermore, the same optimization problem with continuous assignment variables x_{ij}, for all $ij \in A_R$, returns an optimal ConFL solution. This is because the underlying assignment matrix is totally unimodular, whenever z_i values are fixed to zero or one.

Observation 2. *Using equations* (3.2)*, we can re-write constraints* (3.1) *as follows:*

$$
\sum_{uv \in \delta^-(U)} x_{uv} + \sum_{jk \in A_R : j \notin U} x_{jk} \geq 1, \quad \forall U \subseteq S \setminus \{r\}, U \cap F \neq \emptyset,\ \forall k \in R. \qquad (3.7)
$$

Denote by $W = S \setminus U$, and let $A_S^W := \delta^+(W) \cap A_S$ and $A_R^W = \delta^+(W) \cap A_R$. Now, we can interpret these constraints as follows: every cut separating customer k from r (involving all arcs from $A_S \cup A_R$) has to be greater than or equal to one, i.e.:

$$
\sum_{uv \in A_S^W} x_{uv} + \sum_{jk \in A_R^W} x_{jk} \geq 1, \quad \forall W \subseteq S,\ r \in W,\ W \cap F \neq F,\ \forall k \in R.
$$

Figure 3.1 illustrates an example of these cut set inequalities.

According to the result of Swamy and Kumar [48], the integrality gap of the LP-relaxation of (CUT_R) is not greater than 8.55, if c is a metric, and core costs are M times more expensive than the assignment costs ($M \geq 1$).

Ljubić' Cut Set Formulation Ljubić [36] presents a slightly different formulation where the cuts are defined according to the open facilities:

$$
\begin{aligned}
(CUT_F) \quad & \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i \\
\text{s.t.} \quad & \sum_{uv \in \delta^-(W)} x_{uv} \geq z_i \quad \forall W \subseteq S \setminus \{r\},\ \forall i \in W \cap F \neq \emptyset \qquad (3.8)
\end{aligned}
$$

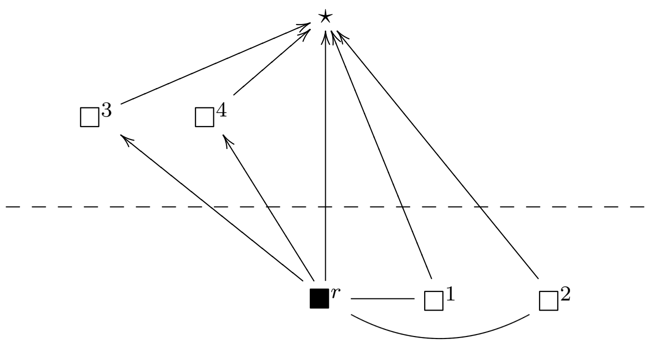


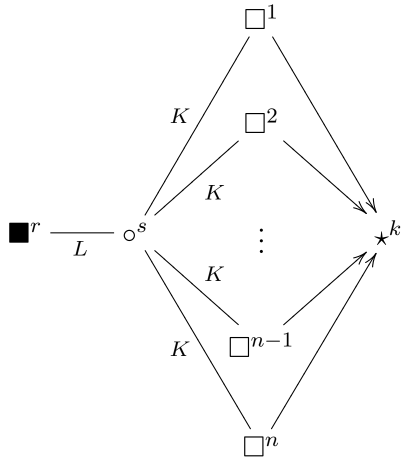
Figure 3.1: Graphic illustration for cut inequalities (3.2). $W = \{r, 1, 2\}$, $U = \{3, 4\}$

(3.2) - (3.6)

Lemma 1. *There are instances for which the values of the LP-relaxation of the CUT_F model can be as bad as $\frac{1}{|F|-1} OPT$, where OPT denotes the integer optimal solution.*

Proof. Example 1 illustrates such a situation. In this example $n := |F| - 1$. The optimal solution value for the LP relaxation of CUT_L is $v_{LP}(CUT_L) = \frac{L}{n} + K + 3$ and the optimal integer solution value is $OPT = L + K + 3$. For $K >> L$, we get $\frac{vlpCUT_L}{OPT} \approx \frac{1}{n}$. □

Example 1. The cost structure is as follows: all facility opening and assignment costs are 1. $c_{rs} = L$ and $c_{si} = K$, for all $i \in \{1, \dots, n\}$.



3.2.2 Flow-based Formulations

Extending flow formulations for the (prize-collecting) Steiner tree problem (see, e.g., [37, 47]), several ways to model ConFL as a flow problem are possible. One option is to have a flow from the root to each customer. Alternatively, flow can be allowed from the root node to open facilities only, with additional constraints ensuring customers to be assigned to an open facility. Further it is possible to consider just one single commodity or separate commodities for each customer or facility respectively.
In the following we propose six different flow formulations for ConFL. The strength of the different formulations is discussed later in Section 3.3.

Single-Commodity Flow Between Root and Facilities This single commodity-flow formulation with flow between root node and facilities is an extension of the single-commodity flow formulation for the prize-collecting Steiner tree problem (see, e.g., Ljubić [37]). The amount of flow terminating in a facility is linked to the variable indicating whether the facility is open or not. For all $ij \in A_S$, continuous variable g_{ij} denotes the amount of flow that is simultaneously routed from r toward all open facilities over arc ij.

$$(SCF_F) \quad \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i$$

$$\text{s.t.} \quad \sum_{ji \in A_S} g_{ji} - \sum_{ij \in A_S} g_{ij} = \begin{cases} z_k & i = k,\, k \in F \\ -\sum_{k \in F} z_k & i = r \\ 0 & i \in S \setminus \{F\} \end{cases} \qquad \forall i \in S \tag{3.9}$$

$$0 \le g_{ij} \le (|F| - 1) \cdot x_{ij} \quad \forall ij \in A_S \tag{3.10}$$

$$(3.2)\text{ - }(3.6)$$

Constraints (3.9) ensure that each facility $j \in F$ receives z_j units of flow from the root. The coupling constraints (3.10) ensure that on every arc ij, there is enough capacity to simultaneously route that flow. They also force an arc ij to be installed if there is a flow sent through it. Model SCF_F comprises $O(|A|)$ constraints and $O(|A|)$ binary and continuous variables.
The following result is due to the usage of "big-M" constraints in (3.10):

Lemma 2. *There are instances for which*

a) the values of the LP-relaxation of the SCF_F model can be as bad as $\frac{1}{|F|-1} OPT$, and

b) the ratio $\frac{\upsilon_{LP}(SCF_F)}{\upsilon_{LP}(CUT_F)} \approx \frac{1}{|F|}$.

Proof. a) The same example given in Figure 1 provides $\upsilon_{LP}(SCF_F) = \frac{L}{n} + \frac{K}{n} + 3$ which gives ratio $\frac{\upsilon_{LP}(SCF_F)}{OPT} \approx \frac{1}{|F|-1}$.

b) If $K >> L$ in the same example, we obtain $\frac{\upsilon_{LP}(SCF_F)}{\upsilon_{LP}(CUT_F)} = \frac{\frac{L}{n}+\frac{K}{n}+3}{\frac{L}{n}+K+3} = \frac{1}{|F|-1} \approx \frac{1}{|F|}$. □

Single-Commodity Flow between Root and Customers We now consider single commodity-flow from the root node to each of the customers. At the expense of more flow variables this allows us to drop constraints (3.2) used in SCF_F:

$$(SCF_R) \quad \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i$$

$$\text{s.t.} \quad \sum_{ji \in A_S} f_{ji} - \sum_{ij \in A} f_{ij} = \begin{cases} 1 & i \in R \\ -|R| & i = r \\ 0 & i \in S \setminus \{r\} \end{cases} \qquad \forall i \in V \tag{3.11}$$

$$0 \le f_{ij} \le |R| \cdot x_{ij} \quad \forall ij \in A \tag{3.12}$$

$$(3.3)\text{ - }(3.6)$$

Constraints (3.11) ensure that each customer receives one unit of flow from the root node and constraints (3.12) are similar to (3.10). However, one easily observes that, although redundant for the MIP formulation, assignment constraints (3.2) can strengthen the quality of lower bounds. We denote by SCF_R^+ the formulation SCF_R extended by (3.2). Models SCF_R and SCF_R^+ comprise $O(|A|)$ constraints and $O(|A|)$ binary variables.

Lemma 3. *There are instances for which*

a) the values of the LP-relaxation of the SCF_R (SCF_R^+) model can be as bad as $\frac{1}{|R|} OPT$, and

b) the ratio $\frac{\upsilon_{LP}(SCF_R)}{\upsilon_{LP}(CUT_R)} \approx \frac{1}{|R|}$.

Multi-Commodity Flow with One Commodity per Facility The two flow formulations presented above can be improved by disaggregation of commodities.
Choosing one commodity per facility, each variable indicating an open facility is linked to a distinct commodity. A multi-commodity flow formulation with one commodity per facility is given by:

$$(MCF_F) \quad \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i$$

$$\text{s.t.} \quad \sum_{ji \in A_S} g_{ji}^k - \sum_{ij \in A_S} g_{ij}^k = \begin{cases} z_k & i = k \\ -z_k & i = r \\ 0 & i \neq k, r \end{cases} \qquad \forall i \in S \quad \forall k \in F \tag{3.13}$$

$$0 \leq g_{ij}^k \leq x_{ij} \qquad \forall ij \in A_S, \ \forall k \in F \tag{3.14}$$

(3.2) - (3.6)

Equations (3.13) are the flow preservation constraints defining the flow from the root node to each facility. These constraints ensure the existence of a connected path from r to every open facility. The stronger coupling constraints ensure that the arc is open if a flow is sent through it. Formulation MCF_F comprises $O(|A_S||F| + |A_R|)$ constraints, $O(|A_S||F|)$ continuous and $O(|A|)$ binary variables.

Multi-Commodity Flow with One Commodity per Customer Another choice for the commodities we use, is the set of customers. Assigning a commodity of size 1 to each customer allows to remove the **z** variables from the flow preservation constraints. Using one commodity per customer, ConFL can be stated as:

$$(MCF_R) \quad \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i$$

$$\text{s.t.} \quad \sum_{ji \in A} f_{ji}^k - \sum_{ij \in A} f_{ij}^k = \begin{cases} 1 & i = k \\ -1 & i = r \\ 0 & i \neq k, r \end{cases} \qquad \forall i \in V \quad \forall k \in R \tag{3.15}$$

$$0 \leq f_{ij}^k \leq x_{ij} \qquad \forall ij \in A, \ \forall k \in R \tag{3.16}$$

(3.3) - (3.6)

Formulation MCF_R comprises $O(|A||R|)$ constraints, $O(|A||R|)$ continuous and $O(|A|)$ binary variables.

Observation 3. *Variables x_{ij}, $ij \in A_R$, are redundant in this formulation, as every LP-optimal solution of MCF_R also satisfies:*

$$f_{jk}^l = \begin{cases} x_{jk}, & \text{if } l = k \\ 0, & \text{otherwise} \end{cases} \qquad \forall l \in R, \quad \forall jk \in A_R.$$

Therefore, constraints (3.2) are redundant, for both, the MCF_R model and its LP-relaxation. However, we keep variables $x_{ij}, ij \in A_R$ in this model for better readability.

Strong Formulations Comprising Common Flow Variables

Polzin and Daneshmand [47] have developed a formulation which they call *Common Flow* formulation for the Steiner arborescence problem. It is based on a disaggregation of multi commodity-flow formulation with additional 4-index variables. These variables indicate the common flow from the root towards any pair of terminals. For ConFL this gives two choices on the common flows considered, towards facilities or towards customers. The variant, where common flows towards facilities are considered, is an extension of MCF_F, the other one is an augmentation of MCF_R and it is the strongest one among all formulations presented in this paper (see Section 3.3).

Common Flow Between Root and Facilities Let $\bar{g}_{ij}^{kl}$ denote the common flow towards facilities k and l, $k, l \in F, k \neq l$, over an arc ij. Then a MIP formulation of ConFL using common flows from the root to facilities is given by:

$$(CF_F) \quad \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i$$

$$\text{s.t.} \quad \sum_{ji \in A_S} g_{ji}^k - \sum_{ij \in A_S} g_{ij}^k = \begin{cases} z_k & i = k \\ -z_k & i = r \\ 0 & i \neq k, r \end{cases} \quad \forall i \in S \quad \forall k \in F \tag{3.17}$$

$$\sum_{ij \in A_S} \bar{g}_{ij}^{kl} - \sum_{ji \in A_S} \bar{g}_{ji}^{kl} \leq \begin{cases} \min(z_k, z_l) & i = r \\ 0 & \forall i \in S \setminus \{r\} \end{cases} \quad \forall i \in S \quad \forall k, l \in F \tag{3.18}$$

$$0 \leq \bar{g}_{ij}^{kl} \leq \min(g_{ij}^k, g_{ij}^l) \quad \forall ij \in A_S, \quad \forall k, l \in F \tag{3.19}$$

$$0 \leq g_{ij}^k + g_{ij}^l - \bar{g}_{ij}^{kl} \leq x_{ij} \quad \forall ij \in A_S, \quad \forall k, l \in F \tag{3.20}$$

$$(3.2) - (3.6)$$

Constraints (3.17) are flow preservation constraints as in MCF_F. Constraints (3.18) ensure that the common flow from the root toward facilities k and l is non-increasing. Inequalities (3.19) define the relation between common flow and commodity flow variables. The coupling constraints (3.20) ensure that the arc is installed whenever there is a flow sent through it.
Formulation CF_F comprises $O(|A_S||F|^2)$ constraints, $O(|A_S||F|^2)$ continuous and $O(|A|)$ binary variables.

Common Flow Between Root and Customers Starting from the MCF_R model, we can now derive the other common flow formulation. Let $\bar{f}_{ij}^{kl}$ denote the common flow towards customers k and l, $k \neq l$. Then the common flow formulation with flows from the root to customers is given by:

$$(CF_R) \quad \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i$$

$$\text{s.t.} \quad \sum_{ji \in A} f_{ji}^k - \sum_{ij \in A} f_{ij}^k = \begin{cases} 1 & i = k \\ -1 & i = r \\ 0 & i \neq k, r \end{cases} \quad \forall k \in R \tag{3.21}$$

$$\sum_{ij \in A_S} \bar{f}_{ij}^{kl} - \sum_{ji \in A_S} \bar{f}_{ji}^{kl} \leq \begin{cases} 1 & i = r \\ 0 & \forall i \in S \setminus \{r\} \end{cases} \quad \forall i \in V \quad \forall k, l \in R \tag{3.22}$$

$$0 \leq \bar{f}_{ij}^{kl} \leq \min(f_{ij}^k, f_{ij}^l) \quad \forall ij \in A, \quad \forall k, l \in R \tag{3.23}$$

$$0 \leq f_{ij}^k + f_{ij}^l - \bar{f}_{ij}^{kl} \leq x_{ij} \quad \forall ij \in A, \quad \forall k, l \in R \tag{3.24}$$

$$(3.3) - (3.6)$$

Constraints (3.21) are flow preservation constraints as in MCF_R. Inequalities (3.22) ensure that the common flow from the root to customers k and l is non-increasing. Constraints (3.23)-

(3.24) are equivalents of (3.19) - (3.20). Formulation CF_R comprises $O(|A||R|^2)$ constraints, $O(|A||R|^2)$ continuous and $O(|A|)$ binary variables.

3.2.3 Formulations Based on Sub-tour Elimination Constraints

Another well-studied group of formulations for problems on graphs are based on sub-tour elimination. We present here one compact and one exponential size model.

Miller-Tucker-Zemlin Formulation One very simple strategy for sub-tour elimination was proposed by Miller, Tucker and Zemlin [44] and has been applied to a number of problems, including (Asymmetric) Traveling Salesman, Vehicle Routing, Minimum Spanning Tree and Steiner Tree Problem [10, 13, 19, 45]. In addition to x and z variables, we now introduce *level variables* $u_i \geq 0$, for all $i \in S$, determining the level of node i in the tree solution. The root node is assigned to the level zero.
Using Miller-Tucker-Zemlin (MTZ) constraints, ConFL can be stated as:

$$(MTZ) \quad \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i$$

$$\sum_{i \in S \setminus \{k\}} x_{ij} \geq x_{jk} \qquad \forall j \in S \setminus \{r\}, k \in V \tag{3.25}$$

$$|S| \cdot x_{ij} + u_i \leq u_j + |S| - 1 \qquad \forall ij \in A_S \tag{3.26}$$

$$u_r = 0 \tag{3.27}$$

$$u_i \geq 0 \qquad \forall i \in S \setminus \{r\} \tag{3.28}$$

(3.2) - (3.6)

Constraints (3.25) limit the out-degree of a node by its in-degree. Constraints (3.26) are Miller-Tucker-Zemlin sub-tour elimination constraints, setting the difference $u_j - u_i$ for an open arc ij to exactly 1, thereby eliminating cycles in the Steiner tree connecting the facilities. Constraint (3.27) sets the level of the root node to zero.
Formulation MTZ comprises $O(|A|)$ constraints, $O(|S|)$ continuous and $O(|A|)$ binary variables. The formulation is small in the number of constraints and variables, compared to the aforementioned formulations based on flows or cut sets. The quality of the lower bounds, i.e. the strength of the formulations will be analyzed in the subsequent section.
Note that for our computational experiments we replaced constraints (3.26) by the following stronger ones:

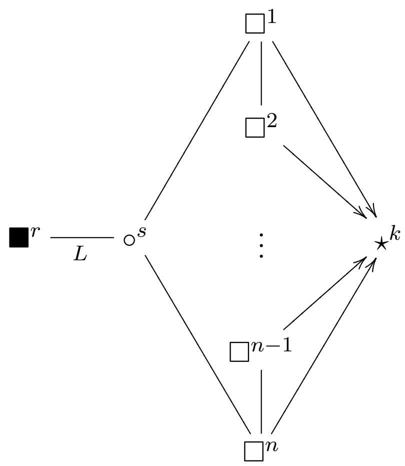
$$(|S| - 2) \cdot x_{ji} + |S| \cdot x_{ij} + u_i \leq u_j + |S| - 1 \qquad \forall ij \in A_S$$

However, the polyhedral results in Section 3.3 are for the weaker model.

Lemma 4. *The values of the LP-relaxation of the MTZ model can be arbitrarily bad.*

Proof. Consider Example 2: The LP-solution opens each facility with $1/n$, and builds one directed cycle of $\{s\} \cup \{1, \ldots, n\}$ where for each arc ij in the cycle $x_{ij} = 1/n$. It assigns $v_{LP}(MTZ) = 4 + \frac{1}{n}$ and $OPT = L + 4$, which gives ratio $\frac{v_{LP}(MTZ)}{OPT} \approx \frac{1}{L}$. □

Example 2. In this example $n := |F| - 1$. The cost structure is as follows: all facility opening, arc opening and assignment costs are 1, except for $c_{rs} = L$, where $L \gg 0$ is an arbitrarily large number.



Formulation Based on Generalized Sub-tour Elimination Constraints To model the Steiner tree in the core network, one might consider another formulation extended by the following node variables:

$$w_i = \begin{cases} 1, & \text{if } i \text{ belongs to the solution,} \\ 0, & \text{otherwise} \end{cases}, \qquad \forall i \in S$$

Such model has been used for the node-weighted Steiner tree problems (see, e.g., [16, 39, 41]).

$$(GSEC) \quad \min \sum_{ij \in A} x_{ij} c_{ij} + \sum_{i \in F} z_i f_i$$

$$\sum_{uv \in A: u,v, \in U} x_{uv} \leq \sum_{i \in U \setminus \{k\}} w_i \quad \forall U \subset S, \forall k \in U \tag{3.29}$$

$$\sum_{uv \in A} x_{uv} = \sum_{i \in S \setminus \{r\}} w_i \tag{3.30}$$

$$w_i \geq z_i \quad \forall i \in F \tag{3.31}$$

$$0 \leq w_i \leq 1 \quad \forall i \in S \tag{3.32}$$

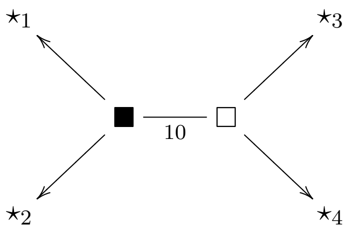
$$(3.2) \text{ - } (3.6)$$

Equality (3.30) ensures that the set of edges is equal to the number of selected nodes minus one. In order to ensure the tree structure, sub-tours are eliminated by deploying constraints (3.29). Since facility nodes can also be used only as Steiner nodes, in which case $w_i = 1$ and $z_i = 0$, inequalities (3.31) must hold.
We will see in the following section that the results known for Steiner trees with respect to *GSEC*, directly apply to ConFL.

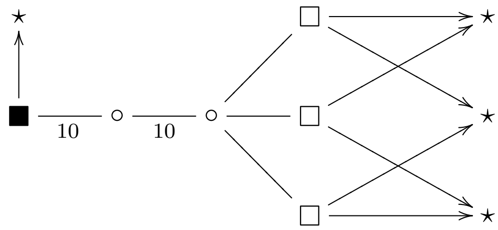
3.3 Polyhedral Comparison

In this section we provide a theoretical comparison of the MIP models described above with respect to optimal values of their LP-relaxations. The examples given below are used in the proofs of this section.

Example 3. The underlying network is given in the figure below. The facility node variable is 1/4 for SCF_R and 1 for all other models.



Example 4. This example is a small variant of Example 1. It will show the weakness of models where the flows are only defined on the core subgraph A_S. Facility node variables are 1/8 for SCF_R and 1/2 for all other models.



Example 5. The core network is directed and there is exactly one customer that can be assigned to each facility. Thus, every facility needs to be open in a feasible solution. The underlying graph is shown in Figure 3.2. Facility node variables are 1/5 for SCF_R and 1 for all other models. A version of this example was described by Polzin and Daneshmand [47].

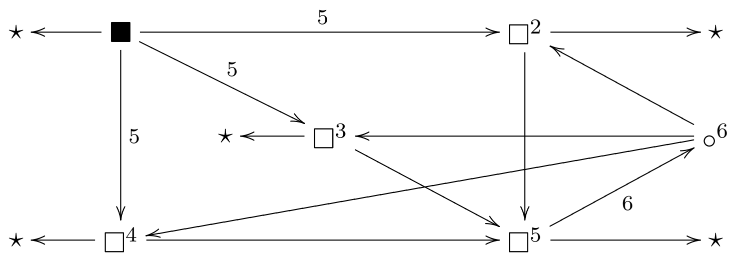
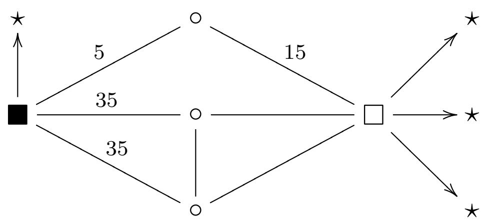
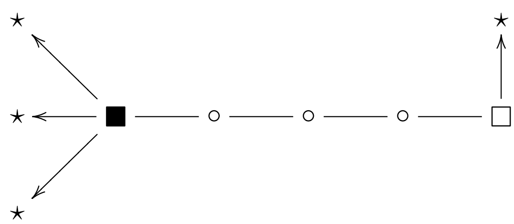


Figure 3.2: Example 5

Example 6. The example shown below will demonstrate the weakness of Miller-Tucker-Zemlin constraints. The facility node variable is 1/4 for SCF_R and 1 for all other models. In the LP solution for model *MTZ* there is a cycle consisting of the arcs of weight 1. The open facility is not connected to the root.



Example 7. The example shown below will demonstrate the weakness of "big-M" constraints in the models comprising single commodity flow. The facility node variable is 1/4 for SCF_R and 1 for all other models.



Let $v_{LP}(.)$ denote the optimal solution value of the LP relaxation of a given model. By comparing the optimal LP solution values for the aforementioned examples, provided by the models in Section 3, we can state the following

Lemma 5. *The following pairs of formulations are incomparable with respect to the quality of lower bounds:*

	Ex. 3	Ex. 4	Ex. 5	Ex. 6	Ex. 7
MTZ	16	18	20	9	10
SCF_F	11	$14\frac{3}{8}$	$14\frac{1}{5}$	16	8
SCF_R	$7\frac{1}{4}$	$18\frac{1}{8}$	7	$17\frac{1}{4}$	$3\frac{1}{4}$
SCF_R^+	11	$22\frac{1}{4}$	$14\frac{1}{5}$	21	7
MCF_F	16	18	22	26	10
MCF_R	16	28	22	26	10
CF_F	16	18	24	26	10
CF_R	16	28	24	26	10

Table 3.1: Optimal LP solutions for Examples 3 - 7

a) MTZ *and* SCF_F*,*
b) MTZ *and* SCF_R (SCF_R^+)*,*
c) SCF_F *and* SCF_R (SCF_R^+)*,*
d) SCF_R (SCF_R^+) *and* MCF_F*,*
e) SCF_R (SCF_R^+) *and* CF_F*,*
f) MCF_R *and* CF_F*.*

Proof. a) In Example 3 we have $\upsilon_{LP}(SCF_F) = 11 < 16 = \upsilon_{LP}(MTZ)$ and in Example 6 we have $\upsilon_{LP}(MTZ) = 9 < 10 = \upsilon_{LP}(SCF_F)$.

b) In Example 3 we have $\upsilon_{LP}(SCF_R) = 7.25 < \upsilon_{LP}(SCF_R^+) = 11 < \upsilon_{LP}(MTZ) = 16$ and in Example 6 we have $\upsilon_{LP}(MTZ) = 9 < 17.25 = \upsilon_{LP}(SCF_R) < \upsilon_{LP}(SCF_R^+) = 21$.

c) In Example 4 we have $\upsilon_{LP}(SCF_F) = 14.325 < 18.125 = \upsilon_{LP}(SCF_R)$ and in Example 7 we have $\upsilon_{LP}(SCF_R) = 3.25 < \upsilon_{LP}(SCF_R^+) = 7 < \upsilon_{LP}(SCF_F) = 8$.

d) For Example 4 we have $\upsilon_{LP}(SCF_R) = 18.125 > 18 = \upsilon_{LP}(MCF_F)$. For Example 3 we have $\upsilon_{LP}(SCF_R) = 7.25 < \upsilon_{LP}(SCF_R) = 11 < \upsilon_{LP}(MCF_F) = 16$.

e) For Example 3 we have $\upsilon_{LP}(SCF_R) = 7.25 < \upsilon_{LP}(SCF_R^+) = 11 < \upsilon_{LP}(CF_F) = 16$, for Example 4 we have $\upsilon_{LP}(CF_F) = 18 < \upsilon_{LP}(SCF_R) = 18.125 < \upsilon_{LP}(SCF_R^+) = 22.25$.

f) Consider Examples 4 and 5. For Example 4 we have $\upsilon_{LP}(CF_F) = 18 < 28 = \upsilon_{LP}(MCF_R)$, for Example 5 we have $\upsilon_{LP}(MCF_R) = 22 < 24 = \upsilon_{LP}(CF_F)$.

□

Denote by $\mathcal{P}_.$ the polytope of the LP-relaxation of any of the MIP models described above, and with $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_.)$ the natural projection of that polytope onto the space of variables $\mathbf{x}$ and $\mathbf{z}$.

Lemma 6. *The following results hold:*

a) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CF_F}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MCF_F}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{SCF_F})$*, and*

b) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CF_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MCF_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{SCF_R^+}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{SCF_R})$*.*

Proof. The results follow immediately from the corresponding results for Steiner trees, see e.g., [47]. Instances that prove the strict inclusion can be found in Table 3.1. □

Lemma 7. *The following results hold:*

a) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MCF_F}) = \mathcal{P}_{CUT_F} = Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{GSEC})$*, and*

b) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MCF_R}) = \mathcal{P}_{CUT_R}$*.*

Proof.

a) The first equality follows from the min-cut max-flow theorem, the second one follows from the related result for node-weighted Steiner trees, see e.g. [41].

b) This result follows from the min-cut max-flow theorem.

□

Lemma 8. *The following results hold:*

a) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MCF_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MCF_F})$ *and*

b) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CF_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CF_F})$.

Proof.

a) According to Lemma 7, it is enough to show this relationship by comparing $\mathcal{P}_{CUT_R}$ and $\mathcal{P}_{CUT_F}$. Then it is easy to see that every solution $(\mathbf{x}', \mathbf{z}') \in \mathcal{P}_{CUT_R}$ also belongs to $\mathcal{P}_{CUT_F}$. Example 4, with $\upsilon_{LP}(CUT_R) = 28 > 18 = \upsilon_{LP}(CUT_F)$, proves that the opposite is not true.

b) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CF_R}) \subseteq Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CF_F})$**:** Let $(\mathbf{f}', \overline{\mathbf{f}}', \mathbf{x}', \mathbf{z}')$ be in $\mathcal{P}_{CF_R}$. We define the capacities on the subgraph $G_S = (S, A_S)$ as x_{ij}, for all $ij \in A_S$. Since $x_{ij} = max_{k \in R} f^k_{ij}$, and $z_i = max_{ij \in A_R} x_{ij}$, there will be enough capacity to independently route z_i units of flow, for all $i \in F$, such that $z_i > 0$. Now, we are going to construct $(\mathbf{g}, \overline{\mathbf{g}}, \mathbf{x}, \mathbf{z}) \in \mathcal{P}_{CF_F}$ as follows: We fix the ordering of the outgoing arcs of every node $i \in S$ and then apply an adapted Ford-Fulkerson maximum flow algorithm. To define $\mathbf{g}$, we send z_i units of flow from r towards $i \in F$, for all $i \in F$ such that $z_i > 0$. When searching for augmenting paths, we always follow the fixed ordering. Therefore, the outgoing arcs of a node always get saturated in the same order, independently on the commodity under consideration. It follows directly from construction that the common flow $\overline{\mathbf{g}}$ for any pair of facilities k and l, once it splits up, will never meet again, i.e., ineqalities (3.18) will be satisfied.

$Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CF_F}) \nsubseteq Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CF_R})$**:** Consider Example 4, where $\upsilon_{LP}(CF_R) = 28 > 18 = \upsilon_{LP}(CF_F)$.

□

Lemma 9. *Formulation MCF_F (i.e., CUT_F, GSEC) is strictly stronger than formulation MTZ, i.e.* $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MCF_F}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MTZ})$.

Proof. Let C_S denote the set of all cycles in S. Then Padberg and Sung [46] show that constraints (3.26) are equivalent to cycle constraints of the following form:

$$\sum_{ij \in C} x_{ij} \leq |C| - \frac{|C|}{|S|} \qquad \forall C \subseteq C_S \tag{3.33}$$

By inequalities

$$\sum_{ij \in C} x_{ij} \leq |C| - 1 \leq |C| - \frac{|C|}{|S|} \qquad \forall C \subseteq C_S$$

constraints (3.33) are implied by generalized subtour elimination constraints (3.29). For Example 6 $\upsilon_{LP}(MTZ) = 9 < 26 = \upsilon_{LP}(GSEC)$. Thus, $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{GSEC}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{MTZ})$. □

3.3.1 Reformulation as the Steiner Arborescence Problem

As has already been observed in [49], the ConFLP can be transformed into the Steiner Arborescence Problem. This transformation is done by using the well-known *node splitting* technique that has proven useful for different network design problems, see e.g., [3, 7].
To solve an instance of ConFL as SA, we use the following procedure:

- Generate a directed graph $\tilde{G} = (\tilde{V}, \tilde{A})$ with costs $\tilde{\mathbf{c}} : \tilde{A} \mapsto \mathrm{R}_0^+$, as follows:
  - Initialize $\tilde{V} = V$, $\tilde{A} = A$ and $\tilde{\mathbf{c}} = \mathbf{c}$.
  - For any facility node i, add a node i' to the graph, connect i to i', and set $\tilde{c}_{ii'} = f_i$.
  - Replace arcs $ik \in A_R$ by $i'k$.
- Solve the Steiner arborescence problem on the transformed graph $\tilde{G}$ with customers as terminals.

Recall that, given a directed graph $\tilde{G} = (\tilde{V}, \tilde{A})$, with arc weights $\tilde{\mathbf{c}} : \tilde{A} \mapsto \mathbb{R}$, a root $r \in \tilde{V}$, and a set of terminal nodes $R \subset \tilde{V}$, the Steiner arborescence problem searches for the cheapest subtree rooted at r that connects all terminals. Figure 3.3 shows a simple example that illustrates the transformation of ConFL into the SA problem, according to the procedure described above:

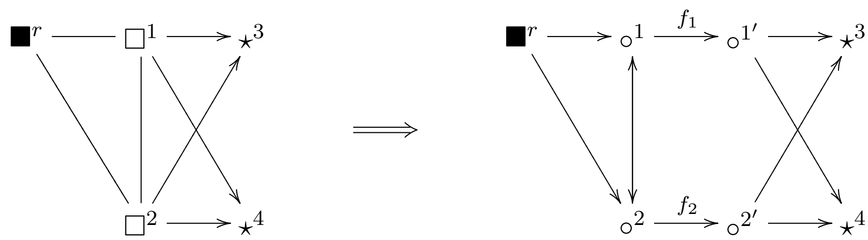


Figure 3.3: Initial undirected ConFL instance and transformed SA instance

For each facility $i \in F$, i corresponds to node's function as Steiner node, while i' corresponds to its function as open facility. With this transformation we ensure that the arc ii' belongs to a solution if and only if facility i is open. Similarly, facility i is used as Steiner node if and only if i belongs to the solution, but arc ii' does not. A similar, but undirected transformation has been used by Bardossy and Raghavan to transform (G)STS, ConFL and RoB into the GConFL [5].
To solve the SA problem as a MIP, let us define binary variables v_{ij} as follows:

$$v_{ij} = \begin{cases} 1, & \text{if } ij \text{ belongs to the solution} \\ 0, & \text{otherwise} \end{cases}, \quad \forall ij \in \tilde{A}.$$

We extend the directed cut-based formulation for Steiner trees (originally proposed by Chopra and Rao [8]) by the root out-degree constraint as follows:

$$(SA) \qquad \min \sum_{ij \in \tilde{A}} \tilde{c}_{ij} v_{ij} \tag{3.34}$$

$$\sum_{ij \in \delta^-(W)} v_{ij} \geq 1, \qquad \forall W \subseteq \tilde{V} \setminus \{r\}, W \cap R \neq \emptyset \tag{3.35}$$

$$v_{rr'} = 1 \tag{3.36}$$

$$v_{ij} \in \{0, 1\} \quad \forall ij \in \tilde{A} \tag{3.37}$$

Let us denote by

$$Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{SA}) = \{(\mathbf{x},\mathbf{z}) \in [0,1]^{|A|} \times [0,1]^{|F|} \mid \mathbf{v} \in \mathcal{P}_{SA} \text{ and}$$

$$x_{kl} = v_{kl} \ \forall kl \in A_S;\ x_{ij} = v_{i'j} \ \forall ij \in A_R;\ z_i = v_{ii'} \ \forall i \in F\},$$

the projection of the $\mathcal{P}_{SA}$ polytope onto the space of variables $(\mathbf{x},\mathbf{z})$.
We show the following result:

Lemma 10. *The LP-relaxation of the Steiner arborescence formulation is equally strong as the LP-relaxation of CUT_R, i.e.:*

$$Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{SA}) = \mathcal{P}_{CUT_R}.$$

Proof. We prove equality by showing mutual inclusion:

$Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{SA}) \subseteq \mathcal{P}_{CUT_R}$: Let $\mathbf{v}'$ be an optimal fractional solution of the LP-relaxation of SA, and $(\mathbf{x}',\mathbf{z}')$ its projection into $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{SA})$. Obviously, (3.1), (3.2) and (3.4) are satisfied by $(\mathbf{x}',\mathbf{z}')$. It only remains to show that $x'_{ij} \leq z'_i, \forall ij \in A_R$. Let us assume that $\exists i \in F$, $\exists ij \in A_R$ such that $x'_{ij} > z'_i$. Without loss of generality assume also that $c_{ij} > 0$. In $\tilde{G}$, $x'_{ij} > z'_i$ implies that $v'_{i'j} > v'_{ii'}$. Given graph $\tilde{G}$ with capacities v'_{ij} on the arcs, the only possibility to send flow from r to j over i' is through the arc ii'. But given the capacity of $v'_{ii'} < v'_{i'j}$, and given the objective function (3.34), it follows that we can find another LP-solution $\mathbf{v}''$ whose objective value is strictly less than $\tilde{\mathbf{c}}^{\mathbf{t}}\mathbf{v}'$, without violating connectivity constraints (3.35), by simply setting $v''_{ij} := v'_{ii'}$ and keeping the rest of values unchanged. This however contradicts the assumption that $\mathbf{v}'$ is an optimal LP-solution.

$\mathcal{P}_{CUT_R} \subseteq Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{SA})$: Let $(\mathbf{x}',\mathbf{z}')$ be a fractional solution satisfying (3.1)-(3.4), and let us assume that the corresponding solution $\mathbf{v}'$ from $\mathcal{P}_{SA}$ is not feasible. In other words, assume that there exists a cut-set $\tilde{W} \subseteq \tilde{V} \setminus \{r\}, \tilde{W} \cap R \neq \emptyset$, such that $\sum_{ij \in \delta^-(\tilde{W})} v_{ij} < 1$. Obviously, there must exist at least one $i \in F \setminus \{r\}$, such that $ii' \in \delta^-(\tilde{W})$. We now construct a new cut-set $\tilde{W}_n$ such that $\delta^-(\tilde{W}_n) = \delta^-(\tilde{W}) \cup \{i'j \mid j \in \tilde{W}\} \setminus \{ii'\}$. Obviously, if $\sum_{ij \in \delta^-(\tilde{W})} v_{ij} < 1$, then also $\delta^-(\tilde{W}_n) < 1$. By repeating this procedure for all $i \in F$ such that $ii' \in \delta^-(\tilde{W})$, we end up with a cut-set containing only arcs from $A_R \cup A_S$, that violates inequality (3.35), which is a contradiction.

□

3.3.2 Full Hierarchy of Formulations

The hierarchical scheme given in Figure 3.4 summarizes the relations between the LP relaxations of the MIP models considered throughout this paper. An arrow specifies that the target formulation is strictly stronger than the source formulation. A dashed connection specifies that the formulations are not comparable to each other.
Note that we do not display formulation SCF_R^+ separately, because it has the same relations as the formulation SCF_R.
Note that all three models SCF_F, MCF_F and CF_F may have lower bounds as bad as $OPT/|F|$. Model CF_R is the strongest one among all considered throughout this paper. Observe that there are several other tree models known for Steiner trees, that can directly be interpreted in ConFL context. Therefore we do not mention them here, but refer the interested reader to Magnanti and Wolsey [41] and Polzin and Daneshmand [47].

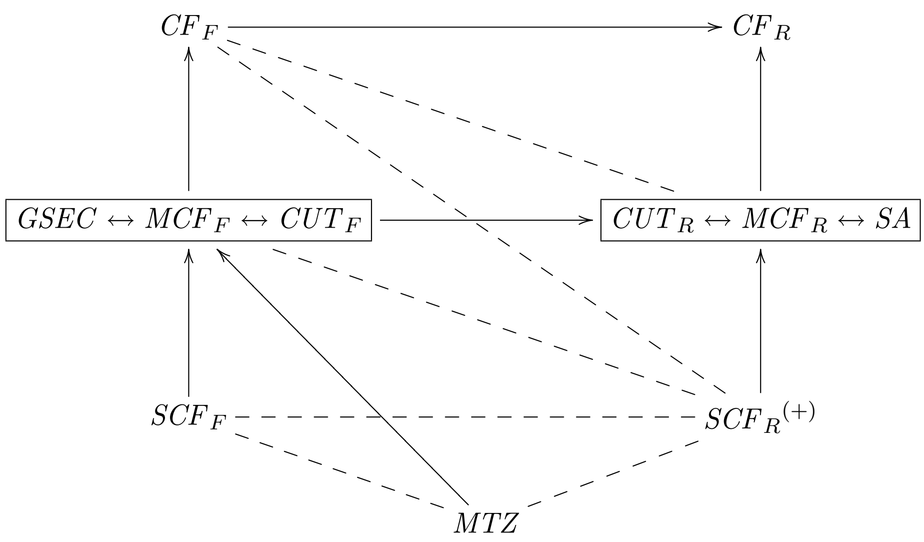


Figure 3.4: Relations between LP-relaxations of MIP models for ConFL

Chapter 4

Computational Results

This chapter contains a short description of the Branch-and-Cut framework used to solve cut based models. Descriptions of the instance sets used in our computational experiments are given. Finally we report the results of our exhaustive computational experiments.

All experiments were performed on a Intel Core2 Quad 2.33 GHz machine with 3.25 GB RAM, where each run was performed on a single processor. For solving the linear programming relaxations and for a generic implementation of the branch-and-cut approach, we used the commercial packages IBM CPLEX (version 11.2) [2] and ILOG Concert Technology (version 2.7).

4.1 Branch-and-Cut Framework

To solve CUT_F and CUT_R models to optimality, we used the branch-and-cut framework developed by I. Ljubic. We now describe the main features of that algorithm. A detailed description of this framework can be found in [18].

The strengthening flow-balance constraints are used to initialize the LP model, together with several subsets of trivial inequalities. The separation of both types of cut set inequalities is done by means of the maximum flow algorithm. Instead of inserting one violated inequality per iteration, techniques known as *back cuts*, *nested cuts* and *minimum cardinality cuts* are used to speed up the separation procedure (see also [38]). Higher branching priorities in all implementations are assigned to potential facility nodes.
Finally, in every node of the branch-and-bound tree, upper bounds are calculated by means of a primal heuristic. The heuristics works in three phases: In the first phase, LP-values of facility nodes are rounded up to determine the set of facilities to be opened. In the second, construction phase, the set of open facilities is used to supply all customers, on one side, and is connected by a Steiner tree, on the other side. In the local improvement phase, a peeling procedure is applied that iteratively removes redundant branches of the core network.

4.2 Test Instances

In our computational study, two groups of instances were considered:

Randomly Generated Graphs From [49] For this set of instances the parameters for the generation were set as follows: $|S| \in \{20, 50, 100\}$, $|R| \in \{20, 50, 100\}$. Edges of the core network

are generated with probability $p(S) \in \{0.1, 0.5, 1\}$, while the connections between facilities and customers are established with probability $p(R) \in \{0.18, 0.55, 1\}$. Edge weights were uniformly randomly set to an integer value between 50 and 100. Finally, the facility opening costs were uniformly randomly assigned to values between 150 and 200. Increasing only the core costs did not significantly change the behavior of the GRASP algorithm for this set of instances. The core network was generated by MAPLE, using the parameters given above. Finally, customers are randomly linked to the existing nodes using the density values $p(R)$.
As the original instances are unrooted we selected the facility with the highest index for the root node respectively.

Graphs Derived From OR-library [6] and UflLib [1] We consider another class of benchmark instances, obtained by merging data from two public sources. In general, we combine an UFLP instance with an STP instance, to generate ConFL input graphs in the following way: first $|F|$ nodes of the STP instance are selected as potential facility locations, and the node with index 1 is selected as the root. The number of facilities, the number of customers, opening costs and assignment costs are provided in UFLP files. STP files provide edge-costs and additional Steiner nodes.

- We consider two sets of non-trivial UFLP instances from UflLib [1]:
  - `mp-{1,2}` and `mq-{1,2}` instances have been proposed by Kratica et al. [32]. They are designed to be similar to UFLP real-world problems and have a large number of near-optimal solutions. There are 6 classes of problems, and for each problem $|F| = |R|$. We took 2 representatives of the 2 classes MP and MQ of sizes 200×200 and 300×300, respectively.
  - The `gs-{250,500}a-{1,2}` benchmark instances were initially proposed by Koerkel [31] (see also Ghosh [15]). Here we chose two representatives of the 250×250 and 500×500 classes, respectively. The authors drew uniformly at random connection costs from [1000, 2000], and the facility opening costs from [100, 200].
- STP instances: Instances `{c,d}n`, for $n \in \{5, 10, 15, 20\}$ were chosen randomly from the OR-library [6] as representatives of medium size instances for the STP. These instances define the core networks with between 500 and 1000 nodes and with up to 25,000 edges.

Combined with assignment graphs, the largest instances of this data set contain 1,300 nodes and 115,000 edges.

4.3 Testing Randomly Generated Instances

For the following tests we turn the primal heuristics off, in order to compare lower bounds of all presented MIP formulations. Furthermore, our preliminary results have shown that turning all CPLEX general purpose cuts speeds up the performance. Therefore, and in order to avoid biased results, all the results reported in this paper are obtained without usage of these cuts.

LP-gaps We first test the performance and the quality of lower bounds for proposed formulations. For that purpose, we run the models as linear programs. Table 4.1 provides the average gaps calculated as $(OPT - v_{LP}(.))/OPT$, where optimal values are obtained by running the branch-and-cut approach (see below). The set of 81 instances is divided into 3 groups according to the size of the core- and the assignment-subgraph.

Not surprisingly, the worst gaps are obtained by running SCF_R model in which "big-M" constraints affect all the arcs in G. Comparing gap values of SCF_F model on these three groups, we observe that the gap increases with the size of the nodes of the core network. This is also not surprising, since "big-M" constraints of the SCF_F model affect only the core network. We observe that there is a correlation between the size of the two subgraphs and the quality of obtained lower bounds for the other models as well. The gaps obtained by MTZ model are surprisingly good, and very close to those obtained by MCF_F. The best LP-gaps are obtained by MCF_R model. Interestingly, the most difficult instances for the latter three models appear to be those with the equal number of facilities and customers.
Finally, we tried to make the same experiment with CF_F and CF_R models, but apparently in almost all cases the execution has been terminated because of memory overconsumption.

$\|S\|$	$\|R\|$	MTZ	SCF_F	SCF_R	MCF_F	MCF_R
20	100	1.36 %	5.44 %	96.24 %	1.33 %	0.73 %
50	50	2.57 %	7.33 %	93.28 %	2.51 %	1.36 %
100	20	2.48 %	8.33 %	85.19 %	2.43 %	1.22 %

Table 4.1: Average Integrality Gaps for selected MIP formulations

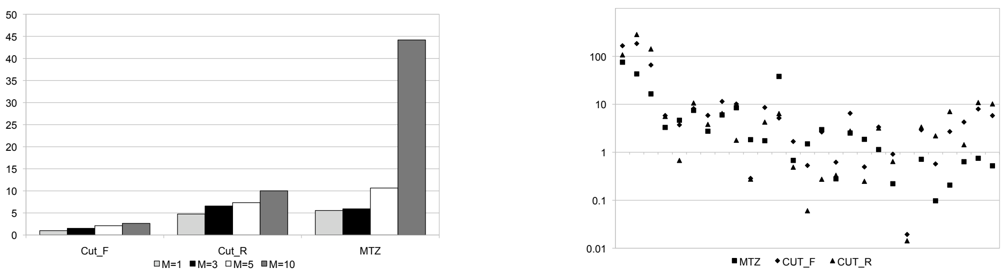
Solving MIPs Table 4.2 shows the running times in seconds ($t[s]$) and the number of branch-and-bound nodes ($B\&B$) needed to solve this set of instances. Each row corresponds to three instances generated according to the same probabilities $p(R)$ and $p(S)$. We provide values for $t[s]$ and $B\&B$ averaged over the respective group. We set the time limit to 1000 seconds. If at least one of the three instances per group is not solved to optimality, we denote this by "-".
As expected, due to the weak lower bounds of the SCF_R^+, most of the instances could not be solved to optimality within the given time limit. The exceptions are graphs with complete bipartite structure of the assignment subgraph A_R that appear to be easy for SCF_R^+. The second worse performance was shown by the MCF_R model, which is easily explained by its huge number of variables.
This test gives two surprising results:

1. Despite the fact, that the integrality gap of model CUT_F can be as bad as $\frac{1}{|F|}$ it outperforms even the strongest cut set based model CUT_R with respect to the running time. On average, the number of $B\&B$ nodes needed by CUT_F is 2.3 times larger than for CUT_R. However, averaged over all 81 instances, CUT_F is about 4.6 times faster than CUT_R.

2. The compact MTZ model with arbitrarily bad lower bounds performs comparatively well. It outperforms CUT_R: the average running time over all instances for MTZ is 1.06 times less than the corresponding time for CUT_R.

$\|S\|$	$\|R\|$	$p(S)$	$p(R)$	Opt	MTZ $t[s]$	MTZ $B\&B$	SCF^+_R $t[s]$	SCF^+_R $B\&B$	MCF_R $t[s]$	MCF_R $B\&B$	CUT_F $t[s]$	CUT_F $B\&B$	CUT_R $t[s]$	CUT_R $B\&B$
20	100	0.1	0.18	9,768	**0.10**	1	-	-	2.00	0	**0.10**	0	0.47	0
20	100	0.5	0.18	9,577	0.29	10	-	-	2.77	0	**0.09**	0	0.48	0
20	100	1.0	0.18	9,554	0.56	48	-	-	9.53	2	**0.12**	0	0.57	0
20	100	0.1	0.55	7,428	2.52	103	-	-	26.92	36	**1.57**	70	5.09	43
20	100	0.5	0.55	7,289	1.46	52	-	-	301.27	31	**1.26**	55	8.18	38
20	100	1.0	0.55	7,316	1.97	57	-	-	-	-	**1.41**	67	9.24	49
20	100	0.1	1.00	6,675	2.39	48	1.59	29	10.54	4	**1.21**	28	4.04	4
20	100	0.5	1.00	6,683	2.02	25	1.41	22	110.56	10	**1.40**	37	6.50	11
20	100	1.0	1.00	6,632	1.97	25	1.20	27	258.67	9	**1.05**	19	9.92	11
50	50	0.1	0.18	5,295	4.47	171	-	-	-	-	**2.50**	81	20.39	45
50	50	0.5	0.18	5,019	10.61	242	-	-	-	-	**2.09**	22	22.04	28
50	50	1.0	0.18	4,987	4.43	42	-	-	-	-	**3.38**	67	16.31	37
50	50	0.1	0.55	4,045	5.24	123	-	-	217.10	12	**3.97**	94	9.05	14
50	50	0.5	0.55	4,011	**6.67**	55	-	-	-	-	7.09	118	38.06	31
50	50	1.0	0.55	3,896	7.52	47	-	-	-	-	**4.13**	74	28.82	21
50	50	0.1	1.00	3,615	4.91	51	**1.10**	4	25.53	3	1.81	29	2.77	4
50	50	0.5	1.00	3,596	5.44	26	2.18	16	284.24	5	**1.64**	21	4.87	8
50	50	1.0	1.00	3,596	7.30	16	**2.17**	21	-	-	2.74	21	10.28	13
100	20	0.1	0.18	2,489	1.84	16	251.27	171,598	122.51	6	**1.22**	14	2.72	3
100	20	0.5	0.18	2,463	10.43	35	-	-	-	-	**5.30**	33	23.57	16
100	20	1.0	0.18	2,487	144.35	378	-	-	-	-	**7.79**	44	42.62	39
100	20	0.1	0.55	1,921	4.44	51	118.32	27,557	43.20	7	**1.75**	30	2.68	9
100	20	0.5	0.55	1,876	8.66	31	-	-	-	-	4.08	29	**3.26**	5
100	20	1.0	0.55	1,873	14.16	13	-	-	-	-	**2.12**	13	4.95	4
100	20	0.1	1.00	1,638	1.16	4	0.82	4	3.07	0	**0.29**	3	1.02	0
100	20	0.5	1.00	1,638	2.70	1	1.26	1	8.98	0	**0.47**	2	1.59	0
100	20	1.0	1.00	1,633	7.26	2	1.31	1	21.08	0	**0.91**	2	2.59	0

Table 4.2: Running times (in seconds) and the number of Branch-and-Bound nodes for selected MIP formulations with CPLEX cuts turned off.



(a) Average slow-down factors for three MIP models and for $M \in \{1, 3, 5, 10\}$.

(b) Speed-up factors obtained by using branching priorities for facility nodes against default branching times.

Figure 4.1: Results for randomly generated instances from [49].

Testing the influence of the factor M In the following test, we multiply the costs of the core network by a factor $M \in \{3, 5, 10\}$. Our goal is to test the influence of the cost structure of the core network on the overall performance of proposed MIP models. For that purpose, we select the best performing models according to the results obtained above, namely: MTZ, CUT_F and CUT_R. As a reference value, we take the average running time the model CUT_L needed to solve the problems with $M = 1$ to optimality. For each of the three MIP models, and for each of possible M values, we divide the corresponding average running time with the reference time to calculate the so-called *slow down factor* shown in Figure 4.1(a).
The obtained slow down factors indicate that the MTZ model is the most affected by increasing the costs of the core network: MTZ needs about 7 times more time to solve the instances to optimality, if the costs of the core network are multiplied by factor $M = 10$. This result is due to decreasing quality of lower bounds of the MTZ model with increasing M values. On the other hand, models CUT_F and CUT_R are not so much affected by that effect: in the worst case, when $M = 10$, the average running time increases by roughly a factor of 2.6 and 2.1 for CUT_F and CUT_R, respectively. We also observe that CUT_F outperforms MTZ by a factor of 5 for $M = 1$, and by a factor of 16 for $M = 10$.

Branching We also tested our branching strategy described in Section 4.1 against the CPLEX default branching strategy. For each of the 27 instance density settings, Figure 4.1(b) shows the speed up factor obtained by dividing two running times: one needed to solve the instance with default CPLEX setting to optimality and the other one obtained with our branching strategy. The values are averaged over three instances per setting. In most of the cases our branching strategy significantly reduces the overall running time. On average over all 81 instances, our branching strategy outperforms CPLEX default branching by a factor of 1.4, 3.3 and 2.9, when models MTZ, CUT_F and CUT_R are solved, respectively.

4.4 Testing Larger Graphs

The set of instances is divided into three groups according to the underlying instance for the assignment graph. We refer to them as `mp`, `mq` and `qs` group. Tables 4.3 and 4.4 report on the results obtained trough this experiment. Note that the optimal values, as well as lower bounds reported in this paper differ from those reported in [36]. This is due to in-degree inequalities used in [36], that turned out to model the Steiner tree star problem, instead of ConFL.

Comparing Two Branch-and-Cut Approaches: First, we compare the two branch-and-cut approaches by running them with the proposed primal heuristic. Regarding 32 instances obtained by combining `stein` and `mp/q` instances, CUT_F solves all 32 instances to provable optimality within 213 seconds on average. The gaps we report for each model were calculated as

$$gap[\%] = \frac{UB - LB}{UB},$$

where UB and LB are the upper and lower bound obtained by the respective model. In addition, we report on the running time in seconds (t [s]), the model CUT_F needs to solve the instances of the `mp/q` group to optimality. Note that CUT_R solves only 7 out of 32 `mp/q` instances to optimality. For the majority of instances CUT_R does not branch at all, as it has not finished the cutting plane phase at the root node of the branch-and-bound tree. This is because the assignment graphs for these instances are complete bipartite, which means that many dense cuts of the CUT_R model need to be separated.

Comparing MIP Models Initialized with Best Upper Bound: Second, we run all three models, MTZ, CUT_F and CUT_R, but we deactivate the primal heuristic. Instead, we initialize the models with the best upper bound found in the previous setting. For the `gs` group of instances, the best lower and upper bounds obtained with this setting can be found in the right hand half of Table 4.4. Each of the models MTZ and CUT_R solves only 8 instances to optimality. For the `mp` subgroup, MTZ gives much smaller gaps though, on average 0.17% compared to 1.42% for CUT_R. For the group of `mq` instances MTZ also outperformes CUT_R with an average gap of 1.86% vs. 3.18% for the latter.
In the last group of large scale instances derived from the `gs` group, the performance of MTZ is comparatively better. CUT_F obtains the smallest gap in 11 cases, but MTZ performs best on 7 instances. Not a single instance of `gs` group has been solved to optimality. Note that for this last group of instances the cost structure is special. The factor M, describing the scale between core and assignment costs is about 0.001.

			PH on, no UB given					PH off, best UB given						
			CUT_R		CUT_F			MTZ		CUT_R		CUT_F		
Stein	UFL	*OPT*	*gap*[%]	*B&B*	*gap*[%]	*B&B*	t [s]	*gap*[%]	*B&B*	*gap*[%]	*B&B*	*gap*[%]	*B&B*	t [s]
c05	mp1	2,691.5	**0.00**	13	**0.00**	27	73	0.34	605	**0.00**	23	**0.00**	33	50
c10	mp1	2,661.7	**0.00**	17	**0.00**	17	67	**0.00**	86	**0.00**	23	**0.00**	25	47
c15	mp1	2,634.7	1.45	1	**0.00**	15	100	0.15	1084	1.39	3	**0.00**	17	73
c20	mp1	2,618.7	1.91	3	**0.00**	33	185	**0.00**	58	1.50	1	**0.00**	11	104
d05	mp1	2,677.9	**0.00**	9	**0.00**	27	62	**0.00**	19	**0.00**	9	**0.00**	37	40
d10	mp1	2,676.5	2.39	0	**0.00**	21	92	0.24	542	2.39	1	**0.00**	21	66
d15	mp1	2,635.7	1.05	5	**0.00**	13	67	**0.00**	43	**0.00**	15	**0.00**	11	41
d20	mp1	2,619.7	1.59	0	**0.00**	27	229	0.06	49	1.59	1	**0.00**	15	82
c05	mp2	2,692.5	**0.00**	11	**0.00**	15	37	**0.00**	58	**0.00**	17	**0.00**	13	26
c10	mp2	2,661.5	**0.00**	9	**0.00**	5	27	**0.00**	97	**0.00**	7	**0.00**	11	23
c15	mp2	2,640.5	0.61	3	**0.00**	10	47	0.13	1772	0.89	0	**0.00**	5	28
c20	mp2	2,626.5	**0.00**	11	**0.00**	11	55	0.06	300	**0.00**	11	**0.00**	11	43
d05	mp2	2,710.6	**0.00**	25	**0.00**	19	41	**0.00**	1048	**0.00**	31	**0.00**	17	31
d10	mp2	2,682.5	1.14	0	**0.00**	29	50	0.26	574	0.94	3	**0.00**	27	50
d15	mp2	2,647.5	0.53	7	**0.00**	7	43	**0.00**	14	0.53	7	**0.00**	7	31
d20	mp2	2,628.5	2.14	0	**0.00**	11	222	0.09	70	2.14	0	**0.00**	11	142
c05	mq1	3,907.0	3.08	1	**0.00**	53	261	1.56	11	3.08	1	**0.00**	41	193
c10	mq1	3,866.5	4.12	0	**0.00**	35	214	1.49	20	4.12	0	**0.00**	37	146
c15	mq1	3,842.5	3.09	0	**0.00**	41	183	1.61	12	3.09	0	**0.00**	35	142
c20	mq1	3,826.5	3.08	0	**0.00**	33	289	1.43	7	3.08	0	**0.00**	35	173
d05	mq1	3,879.0	2.56	1	**0.00**	31	210	**0.00**	25	2.12	3	**0.00**	51	127
d10	mq1	3,869.1	2.99	0	**0.00**	43	242	1.72	15	2.92	0	**0.00**	29	156
d15	mq1	3,843.5	2.68	3	**0.00**	61	173	1.07	28	2.02	5	**0.00**	37	134
d20	mq1	3,828.5	2.80	0	**0.00**	45	483	1.87	5	2.80	0	**0.00**	39	387
c05	mq2	3,768.6	2.89	0	**0.00**	73	561	2.99	10	2.88	0	**0.00**	71	283
c10	mq2	3,732.6	5.14	0	**0.00**	63	320	2.99	9	5.14	1	**0.00**	50	190
c15	mq2	3,689.6	2.31	0	**0.00**	41	259	1.23	6	2.31	0	**0.00**	69	231
c20	mq2	3,686.5	4.58	0	**0.00**	45	620	2.33	3	4.03	0	**0.00**	27	317
d05	mq2	3,741.5	2.60	0	**0.00**	47	276	1.34	8	2.59	0	**0.00**	73	236
d10	mq2	3,720.9	**4.24**	0	**0.00**	31	285	4.07	6	2.52	0	**0.00**	43	396
d15	mq2	3,696.5	3.96	0	**0.00**	41	328	1.49	5	2.44	0	**0.00**	33	198
d20	mq2	3,685.5	5.73	0	**0.00**	27	727	2.60	2	5.73	0	**0.00**	33	402

Table 4.3: Results for large scale instances I: The best obtained gaps per setting and instance are shown in bold.

		PH on, no UB given						PH off, best UB given							
				CUT_R		CUT_F				MTZ		CUT_R		CUT_F	
Stein	UFL	best UB	best LB	*gap*[%]	*B&B*	*gap*[%]	*B&B*	best UB	best LB	*gap*[%]	*B&B*	*gap*[%]	*B&B*	*gap*[%]	*B&B*
c5	gs250a-1	258,568.0	258,088.8	0.27	2	**0.19**	162	258,540.0	258,112.9	0.20	180	0.27	5	**0.17**	289
c10	gs250a-1	258,480.0	257,955.7	0.25	1	**0.20**	147	258,464.0	257,986.5	0.20	201	0.20	7	**0.18**	227
c15	gs250a-1	258,387.0	257,823.3	**0.22**	0	-	-	258,387.0	257,858.5	**0.20**	280	0.23	3	-	-
c20	gs250a-1	258,250.0	257,786.4	0.50	0	**0.18**	15	258,250.0	257,798.6	**0.18**	28	0.52	0	0.49	28
c5	gs250a-2	258,287.0	257,724.9	**0.22**	0	0.31	68	258,077.0	257,744.4	0.23	125	0.42	2	**0.13**	192
c10	gs250a-2	257,990.0	257,600.0	0.24	0	**0.15**	92	257,990.0	257,625.1	**0.14**	120	0.22	3	0.19	175
c15	gs250a-2	257,911.0	257,564.4	0.45	0	**0.13**	17	257,911.0	257,536.4	**0.15**	109	0.27	1	-	-
c20	gs250a-2	258,193.0	257,462.5	0.53	0	**0.28**	6	258,054.0	257,471.5	0.28	11	0.53	0	**0.23**	15
c5	gs500a-1	513,476.0	510,860.9	0.53	0	**0.51**	0	513,364.0	510,866.9	0.51	0	**0.49**	0	0.55	0
c10	gs500a-1	513,148.0	510,733.5	0.48	0	**0.47**	0	513,091.0	510,734.9	0.47	0	0.52	0	**0.46**	2
c15	gs500a-1	512,919.0	510,637.7	0.47	0	**0.45**	0	512,919.0	510,635.8	**0.45**	0	0.47	0	**0.45**	0
c20	gs500a-1	513,158.0	510,568.0	0.51	0	**0.50**	0	513,131.0	510,568.0	-	-	0.52	0	**0.50**	0
c5	gs500a-2	513,663.0	510,844.5	0.61	0	**0.55**	0	513,544.0	510,846.2	0.55	0	0.61	0	**0.53**	0
c10	gs500a-2	513,357.0	510,717.7	0.57	0	**0.51**	0	513,357.0	510,719.7	**0.52**	0	0.55	0	**0.52**	0
c15	gs500a-2	513,127.0	510,616.9	**0.49**	0	**0.49**	0	513,127.0	510,617.4	**0.49**	0	**0.49**	0	**0.49**	0
c20	gs500a-2	513,511.0	510,545.7	**0.58**	0	0.59	0	513,254.0	510,545.7	-	-	**0.53**	0	0.58	0

Table 4.4: Results for large scale instances II: The best obtained gaps per setting and instance are shown in bold.

Chapter 5

Hop Constrained ConFL

In this chapter we first motivate and introduce the Hop Constrained Connected Facility Location problem. Then we develop MIP models of different classes for it. Some of these models are extensions from corresponding models for the ConFL, some extend ideas for related problems like the Minimum Spanning or Steiner Tree problem with hop constraints. We show how the Hop Constrained ConFL can be modelled as ConFL on a layered graph. Later in this chapter, we compare the proposed models with respect to the quality of lower bounds provided by their LP relaxations and provide a full hierarchy of formulations.

5.1 Introduction

We have shown that the Fiber-to-the-Curb strategy is modelled by the Connected Facility Location problem (ConFL). However, in thereby obtained simply connected graphs reliability against single arc failures is not provided. Economic arguments do not allow the installation of 2-connected last mile networks. Therefore, the reliability of end-user connections is maintained by limiting the number of nodes between them and the 2-connected backbone network. We model these reliability constraints within the Fiber-to-the-Curb strategy by generalizing the ConFL to the Hop Constrained ConFL.

5.1.1 Problem Definition

In Chapter 2 we have discussed a number of issues regarding the exact definition of the Connected Facility Location problem. We recall the definition for the *rooted ConFL* given there:

Definition 3 (rooted ConFL)**.** We are given an undirected graph (V, E) with edge costs $c_e \geq 0, e \in E$, facility opening costs $f_i \geq 0, i \in F$, a disjoint partition $\{S, R\}$ of V with $R \subset V$ being the set of customers, $S \subset V$ the set of possible Steiner nodes, $F \subseteq S$ the set of facilities, and the root node $r \in F$. Find a subset of open facilities such that:

- each customer is assigned to the closest open facility,
- a Steiner tree connects all open facilities, and
- the sum of assignment, facility opening and Steiner tree costs is minimized.

Based on this definiton the *Hop Constrained Connected Facility Location Problem* is then:

Definition 4 (HC ConFL)**.** Given an instance of the rooted ConFL as defined above, find a solution that is valid for ConFL and in which there are at most H hops, i.e. edges, between the root and any open facility.

Observation 4. *Using the transformation given in Section 2.1, any HC ConFL instance, in which $S \cap R \neq \emptyset$, can be transformed into an equivalent one such that $\{S, R\}$ is a proper partition of V.*

HC ConFL is not in APX, i.e. it not possible to have polynomial time heuristics that guarantee a constant approximation ratio. This result can be obtained by applying an error-preserving reduction from SET COVER. Using this technique, Manyem and Stallmann [43] show that the Hop Constrained Steiner Tree problem (HCSTP) is not in APX, even if all edge weights are equal to one. Furthermore, Manyem [42] shows that HCSTP is not in APX, even if the edge weights satisfy the triangle inequality. Obviously, HCSTP is a special case of HC ConFL, in which every facility supplies exactly one customer. Therefore, this non-approximability results apply to HC ConFL as well.

5.2 (M)ILP Formulations for HC ConFL

Problem formulations on directed graphs often give better lower bounds than their undirected equivalents (see, e.g., [41]). In Chapter sec:MIPmodels we describe a transformation of undirected ConFL instances into directed ones. In the remainder of this thesis we will focus on the Hop Constrained Connected Facility Location problem on directed graphs defined as follows:

Definition 5 (HC ConFL on directed graphs)**.** We are given a directed graph (V, A) with edge costs $c_{ij} \geq 0, ij \in A$, facility opening costs $f_i \geq 0, i \in F$ and a disjoint partition $\{S, R\}$ of V with $R \subset V$ being the set of customers, $S \subset V$ the set of possible Steiner tree nodes, $F \subset S$ the set of facilities, and the root node $r \in F$. Find a subset of open facilities such that

- each customer is assigned to exactly one open facility,
- a Steiner arborescence rooted in r connects all open facilities,
- the cost defined as the sum of assignment, facility opening and Steiner arborescence cost, is minimized and
- there are at most H hops between the root and any open facility.

All models in this chapter employ the same notation as the ones in Chapter 3

5.2.1 Cut-Based Formulations

Two types of cut set formulations for the HCSTP and HCMST can be found in the literature: path-based and jump-based, respectively. The earlier have been mentioned by [10], the latter are a development of [11].

Cut Set Formulations with Path Constraints

Cut Set Formulation based on the one in Gupta et al. [25] Recall formulation CUT_R for the ConFL problem. By adding an exponential number of constraints that limit the number of hops we can generalize it to model HC ConFL:
Let $P = \{(i_1, j_1), \dots, (i_l, j_l)\}$ with $i_1 = r$ and $j_{k-1} = i_k,\ k = 2 \dots l$ denote a path originating at the root node with l arcs. For a given number l, let $\mathcal{P}_l$ be the set of all such paths P consisting

of l arcs.
Then we can formulate HC ConFL as follows:

$$(CUT^P_R) \quad \min \sum_{ij \in A} c_{ij} x_{ij} + \sum_{i \in F} f_i z_i$$

$$\text{s.t.} \quad \sum_{uv \in \delta^-(U)} x_{uv} + \sum_{jk \in A_R : j \notin U} x_{jk} \geq 1 \qquad \forall U \subseteq S \setminus \{r\}, U \cap F \neq \emptyset,\ \forall k \in R \tag{5.1}$$

$$\sum_{uv \in P} (x_{uv} + x_{vu}) \leq H + 1 \quad \forall P \in \mathcal{P}_{H+2} \tag{5.2}$$

$$\sum_{jk \in A_R} x_{jk} = 1 \qquad \forall k \in R \tag{5.3}$$

$$x_{jk} \leq z_j \qquad \forall jk \in A_R \tag{5.4}$$

$$z_r = 1 \tag{5.5}$$

$$x_{ij} \in \{0, 1\} \quad \forall ij \in A_S \tag{5.6}$$

$$x_{jk} \in [0, 1] \quad \forall jk \in A_R \tag{5.7}$$

$$z_i \in \{0, 1\} \quad \forall i \in F \tag{5.8}$$

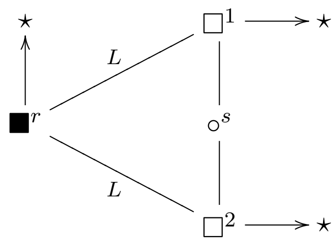
The objective comprises the cost for the Steiner arborescence ($\sum_{ij \in A_S} c_{ij} x_{ij}$), the cost to connect customers to facilities (that we also refer to as *assignment cost*, i.e. $\sum_{ij \in A_R} c_{ij} x_{ij}$) and the facility opening cost ($\sum_{i \in F} f_i z_i$). Inequalities (5.1) represent the set of connectivity cuts. For every subset $U \subseteq S \setminus \{r\}$ and for each customer $k \in R$, an open arc from a facility in U toward j, necessitates a directed path from r towards U. Constraints (5.2) are path-based hop constraints. For any path consisting of $H + 2$ arcs at most $H + 1$ arcs are allowed to be open in a valid solution. Constraints (5.3) ensure that every customer is connected to one facility, constraints (5.4) ensure that each facility is opened if customers are assigned to it, equation (5.5) defines the root node. Constraints (5.3) can be replaced by inequality in case that $c_{ij} > 0$, for all $ij \in A_R$.

The path constraints (5.2) can be replaced by the ones initially presented in [10]:

$$\sum_{uv \in P} x_{uv} \leq H + 1 \qquad \forall P \in \mathcal{P}_{H+2} \tag{5.9}$$

However, constraints (5.2) strictly dominate these, as can be seen in Example 8. This result also holds for the HCSTP and HCMST problem.

Example 8. The hop limit H is 2. The solution in which $x_{r1} = x_{r2} = x_{1s} = x_{s2} = x_{2s} = x_{s1} = 0.5$ is only valid for model CUT^P_R where constraints (5.2) are replaced by the weaker constraints (5.9).



An Adaption of Ljubić' Cut Set Formulation If we replace (5.1) and (5.2) with the following groups of constraints,

$$\sum_{uv \in \delta^-(W)} x_{uv} \geq z_i \qquad \forall W \subseteq S \setminus \{r\},\ \forall i \in W \cap F \neq \emptyset \text{ and} \tag{5.10}$$

$$\sum_{uv \in P} (x_{uv} + x_{vu}) \leq H \qquad \forall P \in \mathcal{P}_{H+1},\ P \subseteq A_S, \tag{5.11}$$

we obtain a hop constrained extension of CUT_F. We refer to it as CUT_F^P. We have shown in Chapter 3 that the lower bounds of CUT_F are up to $|F| - 1$ times worse than the bounds of CUT_R in the absence of hop constraints. HC ConFL contains ConFL as a special case. Thus, this results still holds for HC ConFL.
LP relaxations of both, CUT_R^P and CUT_F^P, can be solved in polynomial time. Connectivity constraints are separated using the maximum flow algorithm (see, e.g., [18]). A polynomial time separation algorithm for path constraints is given in [10].

Cut Set Formulations with Jump Constraints

To formulate cut set based models for HC ConFL with jump constraints we borrow the notation proposed in [11]:
Let $S_0, S_1, \ldots, S_{H+1}$ be a partition of S, such that none of the subsets is empty and that the root node $r \in S_0$ and $S_{H+1} \cap F \neq \emptyset$. We call $J = J(S_0, S_1, \ldots, S_{H+1}) = \bigcup_{(i,j):i<j-1} [S_i, S_j]$ where $[S_i, S_j] = \{uv \in A_S : u \in S_i, v \in S_j\}$ a H-jump. Using J_H, the set of all possible H-jumps, we can formulate hop constraints on the core graph by using the following jump inequalities:

$$\sum_{ij \in J} x_{ij} \geq z_l \quad \forall J \in J_H,\ l \in F \cap S_{H+1}. \tag{5.12}$$

In the following, let CUT_F^J denote the formulation given by replacing constraints (5.11) by (5.12) in formulation CUT_F^P.
These particular jump constraints represent a new way to model hop constraints. They can be applied to all hop constrained network design problems with node variables, like the hop constrained prize-collecting STP or STPRBH.
Let $S_0, S_1, \ldots, S_{H+2}$ be a partition of S, such that none of the subsets is empty and that the root node $r \in S_0$ and $S_{H+2} \cap R \neq \emptyset$. We call $J = J(S_0, S_1, \ldots, S_{H+2}) = \bigcup_{(i,j):i<j-1} [S_i, S_j]$ a $(H+1)$-jump. Using J_{H+1}, the set of all possible $(H+1)$-jumps, we can formulate hop constraints on the core and assignment graph by using the following jump inequalities

$$\sum_{ij \in J} x_{ij} \geq 1 \quad \forall J \in J_{H+1}. \tag{5.13}$$

In the following let CUT_R^J denote the formulation given by replacing constraints (5.2) by (5.13) in formulation CUT_R^P.
An illustration of these jump sets is given in Figure 5.1.
It is an open question whether the LP relaxation of the models with jump constraints is polynomially solvable. There is a conjecture, that the separation of jump constraints is an NP hard problem.

5.2.2 Flow-based Formulations

Multi-Commodity Flow Formulations

Balakrishnan and Altinkemer [4] and Gouveia [20] have used multi-commodity flow formulations for network design problems with hop constraints. In both papers the authors limit the amount

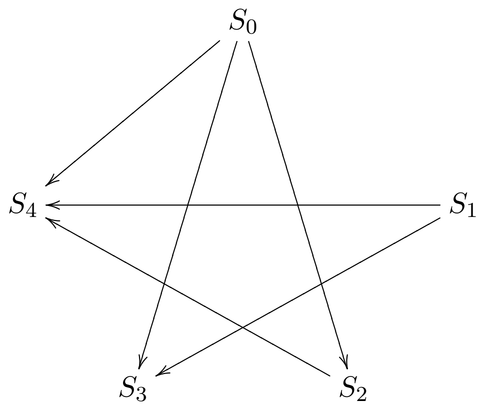


Figure 5.1: Illustration of the arcs contained in a jump for $H = 3$

of flow for each commodity by the hop limit. Together with flow preservation constraints this leads to valid MIP models for HC ConFL.

Multi-Commodity Flow with One Commodity per Facility Choosing one commodity per facility, each variable indicating an open facility is linked to a distinct commodity. A multi-commodity flow formulation with one commodity per facility is given by:

$$(HCMCF_F) \quad \min \sum_{ij \in A} c_{ij} x_{ij} + \sum_{i \in F} f_i z_i$$

$$\text{s.t.} \quad \sum_{ji \in A_S} g^k_{ji} - \sum_{ij \in A_S} g^k_{ij} = \begin{cases} z_k & i = k \\ -z_k & i = r \\ 0 & i \neq k, r \end{cases} \qquad \forall i \in S \quad \forall k \in F \setminus \{r\} \tag{5.14}$$

$$0 \leq g^k_{ij} \leq x_{ij} \qquad \forall ij \in A_S,\ \forall k \in F \setminus \{r\} \tag{5.15}$$

$$\sum_{ij \in A_S} g^k_{ij} \leq H \qquad \forall k \in F \setminus \{r\} \tag{5.16}$$

$$(5.3)\text{ - }(5.8)$$

Equations (5.14) are the flow preservation constraints defining the flow from the root node to each facility. These constraints ensure the existence of a connected path from r to every open facility. The coupling constraints (5.15) ensure that the arc is open if a flow is sent through it. The maximum number of hops on the path from r to k is modelled by inequalities (5.16).

One can easily show, that for any solution with binary entries in $\mathbf{x}$ and $\mathbf{z}$ and fractional flows there exists a solution of equal cost, in which the flows are binary as well. Thus, we do not need integrality constraints for the flow variables. This holds for all other formulations based on multi-commodity flow in this chapter as well.

Formulation $HCMCF_F$ comprises $O(|A_S||F| + |A_R|)$ constraints, $O(|A_S||F|)$ continuous and $O(|A|)$ binary variables.

Multi-Commodity Flow with One Commodity per Customer Another choice for the commodities we use, is the set of customers. Assigning a commodity of demand 1 to each customer allows to remove the $\mathbf{z}$ variables from the flow preservation constraints. Using one commodity per customer, HC ConFL can be stated as:

$$(HCMCF_R) \quad \min \sum_{ij\in A} c_{ij}x_{ij} + \sum_{i\in F} f_i z_i$$

$$\text{s.t.} \quad \sum_{ji\in A} f_{ji}^k - \sum_{ij\in A} f_{ij}^k = \begin{cases} 1 & i=k \\ -1 & i=r \\ 0 & i\neq k,r \end{cases} \qquad \forall i \in V \quad \forall k \in R \tag{5.17}$$

$$0 \leq f_{ij}^k \leq x_{ij} \qquad \forall ij \in A,\ \forall k \in R \tag{5.18}$$

$$\sum_{ij\in A} f_{ij}^k \leq H+1 \ \ \forall k \in R \tag{5.19}$$

$$(5.4) \text{ - } (5.8)$$

Constraints (5.17) and (5.18) guarantee the existence of a directed path from the root r to customer k. Together with constraints (5.19) this path contains at most $H+1$ arcs. Formulation $HCMCF_R$ comprises $O(|A||R|)$ constraints, $O(|A||R|)$ continuous and $O(|A|)$ binary variables. Note that in this formulation variables x_{jk} can be replaced by flows f_{jk}^k for all jk in A_R, as we have already shown in Chapter 3.

Hop Indexed Multi-Commodity Flow Formulations

Gouveia [21] develops a hop indexed formulation for the HCMST and HCSTP. It uses the usual multi-commodity flow variables with an additional hop-index. We use a similar formulation in which we reduce the number of backbone variables to handle HC ConFL.
As for the MCF models, there are two choices on the commodities considered, facilities or customers. The variant, where facilities resemble commodities, is an extension of $HCMCF_F$, the other one is based on $HCMCF_R$.

Hop Indexed Multi-Commodity Flow Between Root and Facilities Let g_{ij}^{kp} denote the flow towards facility $k \in F$, over arc ij, at position p of the path from r to k. Then a MIP formulation of HC ConFL using hop-indexed multi-commodity flows from the root to facilities is given by:

$$(HD_F) \quad \min \sum_{ij\in A} c_{ij}x_{ij} + \sum_{i\in F} f_i z_i$$

$$\text{s.t.} \quad \sum_{ji\in A_S} g_{ji}^{k,p-1} - \sum_{ij\in A} g_{ij}^{kp} = 0 \qquad \forall k \in F\setminus\{r\},\ i \in S\setminus\{r,k\},\ p=2,\dots,H \tag{5.20}$$

$$\sum_{rj\in A_S} g_{rj}^{k1} = z_k \qquad \forall k \in F\setminus\{r\} \tag{5.21}$$

$$\sum_{p=1}^{H}\sum_{jk\in A_S} g_{jk}^{kp} = z_k \qquad \forall k \in F\setminus\{r\} \tag{5.22}$$

$$g_{ij}^{kp} = 0 \qquad \forall ij \in A_S,\ k \in F\setminus\{r\}, \begin{cases} i \notin \{r,k\},\ p=1 \\ i=r,\ p=2,\dots,H \end{cases} \tag{5.23}$$

$$\sum_{p=1}^{H} g_{ij}^{kp} \leq x_{ij} \qquad \forall ij \in A_S,\ k \in F\setminus\{r\} \tag{5.24}$$

$$g_{ij}^{kp} \geq 0 \qquad \forall ij \in A_S,\ k \in F\setminus\{r\},\ p=1,\dots,H \tag{5.25}$$

$$(5.3) - (5.8)$$

Equations (5.20) - (5.22) are flow conservation constraints. Equalities (5.20) set the outflows of a commodity equal to the inflows of the same commodity one position earlier. Constraints (5.21) ensure that z_k units of commodity k leave the root, constraints (5.22) ensure they terminate in the respective facility. Constraints (5.23) fix some flows to zero: Flows at position one are limited to arcs emanating from the root, flows at a higher position than one don't emanate from the root. Inequalities (5.24) ensure an arc is in the solution if flow is sent through it.
In contrast to the model in [21] we do not consider variables g_{kk}^{kp} in our model. Thus, commodity flows can end in the respective facility at any position. All flows fixed to zero in (5.23) could be removed from the model but are kept to simplify the notation of constraints (5.20) - (5.22). Formulation HD_F comprises $O(H|S||F| + |A_S||F|)$ constraints, $O(H|A_S||F|)$ continuous and $O(|A|)$ binary variables.

Hop Indexed Multi-Commodity Flow Between Root and Customers Based on the $HCMCF_R$ model, we can now derive a different hop-indexed formulation. Let f_{ij}^{kp} denote the flow towards customer $k \in R$, over arc ij, at position p of the path from r to k. The formulation using hop-indexed multi-commodity flows from the root to customers is then given by:

$$(HD_R) \quad \min \sum_{ij \in A} c_{ij} x_{ij} + \sum_{i \in F} f_i z_i$$

$$\text{s.t.} \quad \sum_{ji \in A_S} f_{ji}^{k,p-1} - \sum_{ij \in A} f_{ij}^{kp} = 0 \qquad \forall i \in S \setminus \{r\},\ k \in R,\ p = 2, \dots, H+1 \tag{5.26}$$

$$\sum_{rj \in A} f_{rj}^{k1} = 1 \qquad \forall k \in R \tag{5.27}$$

$$\sum_{p=1}^{H+1} \sum_{jk \in A_R} f_{jk}^{kp} = 1 \qquad \forall k \in R \tag{5.28}$$

$$f_{ij}^{kp} = 0 \qquad \forall ij \in A,\ k \in R, \begin{cases} i \neq r,\ p = 1 \\ i = r,\ p = 2, \dots, H+1 \end{cases} \tag{5.29}$$

$$\sum_{p=1}^{H+1} f_{ij}^{kp} \leq x_{ij} \qquad \forall ij \in A,\ k \in R \tag{5.30}$$

$$f_{ij}^{kp} \geq 0 \qquad \forall ij \in A,\ k \in R,\ p = 1, \dots, H+1 \tag{5.31}$$

$$(5.4) - (5.8)$$

Constraints (5.26), (5.27) and (5.28) are flow preservation constraints similar to the ones in HD_F. Constraints (5.29) fix some flows to zero as in HD_F: Flows at position one are only allowed to emanate from the root node. No flows in a later position can occur on arcs leaving the root. Inequalities (5.30) ensure an arc is in the solution if there is flow on it.
Formulation HD_R comprises $O(|S||R|H+|A||R|)$ constraints, $O(|A||R|H)$ continuous and $O(|A|)$ binary variables.

5.2.3 A Formulation Based on Sub-tour Elimination Constraints

Miller-Tucker-Zemlin Formulation Miller-Tucker-Zemlin constraints [44] have been applied to a number of problems. Besides Connected Facility Location [18] we shall mention the models for the Hop Constrained Minimum Spanning and Steiner Tree Problem [10, 19]. In addition to $\mathbf{x}$ and $\mathbf{z}$ variables, we now introduce *hop variables* $u_i \geq 0$, for all $i \in S$. These indicate the distance in hops of each node i from the root. The root node has a distance of zero.

Using the Miller-Tucker-Zemlin ($HCMTZ$) constraints (see, e.g., [22]), HC ConFL can be stated as:

$$
\begin{aligned}
(HCMTZ) \quad \min \sum_{ij \in A} c_{ij} x_{ij} + \sum_{i \in F} f_i z_i & & \\
\sum_{ij \in A_S} x_{ij} \geq x_{jk} & \quad \forall j \in S \setminus \{r\},\ k \in V & (5.32)\\
(H+1)x_{ij} + u_i \leq u_j + H & \quad \forall ij \in A_S & (5.33)\\
u_r = 0 & & (5.34)\\
u_i \geq 0 & \quad \forall i \in S \setminus \{r\} & (5.35)\\
(5.3) - (5.8) & &
\end{aligned}
$$

Constraints (5.32) limit the out-degree of a node by its in-degree. Constraints (5.33) are Miller-Tucker-Zemlin sub-tour elimination constraints, setting the difference $u_j - u_i$ for an open arc ij to at least 1. They thereby eliminate cycles in the Steiner tree connecting the facilities and paths on the core graph with more than H arcs. Constraint (5.34) sets the hop variable to zero for the root node. Formulation $HCMTZ$ comprises $O(|A|)$ constraints, $O(|S|)$ continuous and $O(|A|)$ binary variables.

5.2.4 Hop-indexed Tree Formulations

Gouveia [22] proposes a hop-indexed tree model for the Hop Constrained STP. Voß [50] states that this is a disaggregation of the formulation $HCMTZ$ (see Section 5.3).
To model HC ConFL, there are two options for the hop-indexed variables. We can consider them on the whole graph or alternatively we can separate core and assignment graph and link them by the z-variables indicating the use of facilities.

Hop Constraints on the Entire Graph Let X_{ij}^p indicate whether arc $ij \in A$ is used at the p-th position from the root node. Then we can model HC ConFL as follows:

$$
\begin{aligned}
(HOP_R) \quad \min \sum_{p=1}^{H+1} \sum_{ij \in A} c_{ij} X_{ij}^p + \sum_{i \in F} f_i z_i & & \\
\sum_{\substack{i \in S \setminus \{k\}: \\ ij \in A_S}} X_{ij}^{p-1} \geq X_{jk}^p & \quad \forall jk \in A,\ j \neq r,\ p = 2, \ldots, H+1 & (5.36)\\
\sum_{p=1}^{H+1} \sum_{jk \in A_R} X_{jk}^p = 1 & \quad \forall k \in R & (5.37)\\
\sum_{p=1}^{H+1} X_{jk}^p \leq z_j & \quad \forall jk \in A_R,\ j \neq r & (5.38)\\
X_{ij}^p = 0 & \quad \forall ij \in A, \begin{cases} i = r,\ p = 2, \ldots, H+1 \\ i \neq r,\ p = 1 \end{cases} & (5.39)\\
X_{ij}^p \in \{0,1\} & \quad \forall ij \in A,\ p = 1, \ldots, H+1 & (5.40)\\
(5.8) & &
\end{aligned}
$$

Constraints (5.36) are connectivity constraints. As X_{ij}^p are integer, they eliminate cycles as well. Constraints (5.38) ensure a facility is opened if it serves a customer. Constraints (5.37)

ensure each customer is served. Equations (5.39) fix some of the X_{ij}^p to zero: Arcs emanating from the root can only be 1 hop away from it. Conversely, all other arcs are at least two hops away from the root.
For the polyhedral comparison in Section 5.3 we define the projection of $(\mathbf{X}', \mathbf{z}') \in \mathcal{P}_{HOP_R}$ onto the space of $(\mathbf{x}, \mathbf{z})$ as follows: $x_{ij} := \sum_{p=1}^{H+1} X'^p_{ij}$ for all ij in A and $z_i := z'_i$ for all i in F.

Hop Constraints on the Core Graph We separate core and assignment graph and link them by variables $z_j, j \in F$. After replacing variables $X_{ij}^p, ij \in A_R$ from formulation HOP_R by assignment variables $x_{ij}, ij \in A_R$, we can formulate HC ConFL using hop constraints only on the core graph:

$$(HOP_F) \quad \min \sum_{p=1}^{H} \sum_{ij \in A_S} c_{ij} X_{ij}^p + \sum_{jk \in A_R} c_{jk} x_{jk} + \sum_{i \in F} f_i z_i$$

$$\sum_{\substack{i \in S \setminus \{k\}: \\ ij \in A_S}} X_{ij}^{p-1} \geq X_{jk}^p \qquad \forall jk \in A_S,\ j \neq r,\ p = 2, \dots, H \tag{5.41}$$

$$\sum_{ij \in A_S} \sum_{p=1}^{H} X_{ij}^p \geq z_j \qquad \forall j \in F \setminus \{r\} \tag{5.42}$$

$$X_{ij}^p = 0 \qquad ij \in A_S, \begin{cases} i = r,\ p = 2, \dots, H \\ i \neq r,\ p = 1 \end{cases} \tag{5.43}$$

$$X_{ij}^p \in \{0, 1\} \qquad \forall ij \in A_S,\ p = 1, \dots, H \tag{5.44}$$

$$(5.3) - (5.5), (5.7), (5.8)$$

Constraints (5.41) are connectivity constraints like (5.36). Constraints (5.42) link opening facilities to their in-degree. Constraints (5.43) are similar to (5.39).
We define the projection of $(\mathbf{X}', \mathbf{x}', \mathbf{z}') \in \mathcal{P}_{HOP_F}$ onto the space of $(\mathbf{x}, \mathbf{z})$ as follows: $x_{ij} := \sum_{p=1}^{H} X'^p_{ij}$ for all ij in A_S; $x_{jk} := x'_{jk}$ for all jk in A_R; $z_i := z'_i$ for all i in F.

5.2.5 Modelling Hop Constraints on a Layered Graph

Gouveia et al. [24] model the Minimum Spanning Tree problem with hop constraints (HCMST) as Steiner tree problem on a so-called layered graph. This allows to apply all algorithms developed for the STP to the HCMST. Additionally, the directed cut model on this layered graph turns out to be stronger than the models considered before.
We extend this idea and develop two variants of a layered graph to model the HC ConFL as directed ConFL problem. In the first one we disaggregate only the core graph and leave the assignment graph unchanged. We denote the models on this graph by LG_x. For the second variant we disaggregate both, core and assignment graph. We also disaggregate variables $\mathbf{z}$ for each layer. The models on this graph are denoted by $LG_{x,z}$.

Layered Core Graph LG_x

Consider a graph $LG_x = (V_L, A_x)$ defined as follows:

$$\begin{aligned}
V_L &:= \{r\} \cup S_L \cup R \text{ where}\\
&\quad F_L = \{(i,p) : 1 \le p \le H, i \in F \setminus \{r\}\},\\
&\quad S_L = F_L \cup \{(i,p) : 1 \le p \le H-1, i \in S \setminus F\} \text{ and}\\
A_x &:= \bigcup_{i=1}^{6} A_i \text{ where}\\
&\quad A_1 = \{(r,(j,1)) : rj \in A_S\},\\
&\quad A_2 = \{((i,p),(j,p+1)) : 1 \le p \le H-2, (i,j) \in A_S\},\\
&\quad A_3 = \{((i,H-1),(j,H)) : ij \in A_S, i \in S \setminus \{r\}, j \in F \setminus \{r\}\},\\
&\quad A_4 = \{((i,p),(i,H)) : 1 \le p \le H-1, i \in F \setminus \{r\}\},\\
&\quad A_5 = \{((j,H),k) : jk \in A_R, j \ne r\} \text{ and}\\
&\quad A_6 = \{rk : rk \in A_R\}
\end{aligned}$$

In this directed graph the set of facilities is given by $\{(i,H) : i \in F\}$. The facility opening and assignment costs are left unchanged. The arc costs between (i,p) and $(j,p+1)$ are given as c_{ij}. Finally, arcs between (i,p) and (i,H) are assigned costs of 0 for all $p = 1,\dots,H-1$ and i in F.

Lemma 11. *Any HC ConFL instance can be transformed into an equivalent directed ConFL instance on the layered graph LG_x as described above.*

Figure 5.2 illustrates the transformation of an instance for HC ConFL to an instance for ConFL on LG_x. We link binary variables to the arcs in A_x as follows: X^1_{rj} corresponds to $(r,(j,1)) \in A_1$,

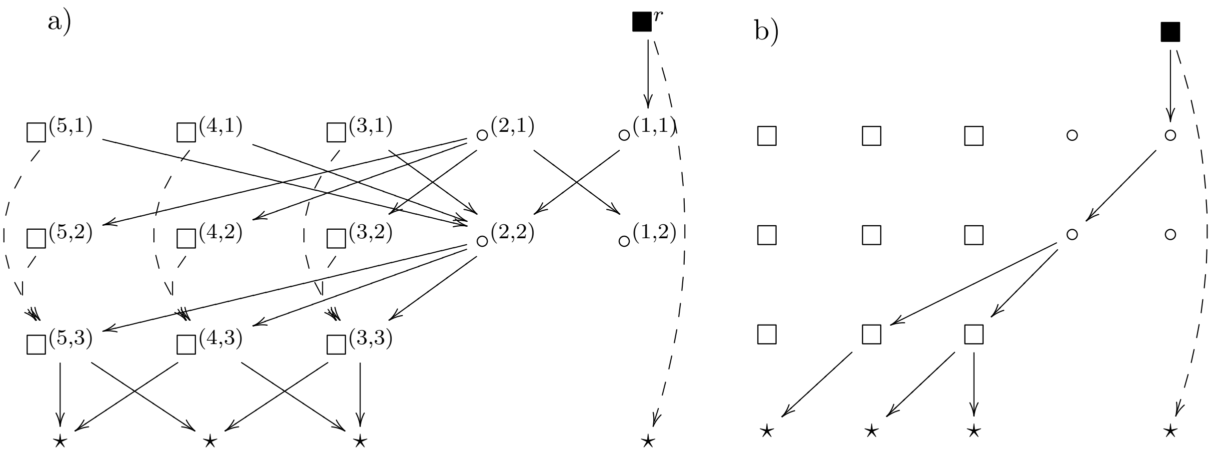


Figure 5.2: a) Layered graph (V_L, A_x) for Example 10; b) An optimal integer solution

X^p_{ij} to $((i,p-1),(j,p)) \in A_2$, X^H_{ij} to $((i,H-1),(j,H)) \in A_3$, X^p_{ii} to $((i,p-1),(i,H)) \in A_4$, X_{jk} to $((j,H),k) \in A_5$ and X^1_{rk} to $rk \in A_6$.
Let $X[V_L \setminus W, W]$ denote the sum of all variables X^p_{ij} and X_{jk} in the cut $\delta^-(W)$ in LG_x defined by $W \subset V_L$ and $r \notin W$. The two cut set based models for ConFL, CUT_F and CUT_R (see

Section 3.2.1) lead to two new formulations, LG_xCUT_F and LG_xCUT_R as follows:

$$(LG_xCUT_F) \quad \min \sum_{rj\in A} c_{rj}X^1_{rj} + \sum_{ij\in A, i\neq r} c_{ij}\sum_{p=2}^{H} X^p_{ij} \; + \sum_{jk\in A_R, j\neq r} c_{jk}X_{jk} + \sum_{i\in F} f_i z_i$$

$$X[V_L \setminus W, W] \geq z_i \qquad \forall W \in S_L,\ r \notin W,\ (i,H)\in W,\ i\in F\setminus\{r\} \tag{5.45}$$

$$X^1_{rk} + \sum_{jk:((j,H),k)\in A_5} X_{jk} = 1 \qquad \forall k \in R \tag{5.46}$$

$$X_{jk} \leq z_j \qquad \forall ((j,H),k)\in A_5 \tag{5.47}$$

$$\mathbf{X} \in \{0,1\}^{|A_x|} \tag{5.48}$$

$$(5.5), (5.8)$$

Constraints (5.45) are cuts on LG_x between sets containing the root and a facility i respectively. Equalities (5.46) ensure each customer is assigned to a facility. Inequalities (5.47) necessitate a facility to be open if customers are assigned to it.

$$(LG_xCUT_R) \quad \min \sum_{rj\in A} c_{rj}X^1_{rj} + \sum_{ij\in A, i\neq r} c_{ij}\sum_{p=2}^{H} X^p_{ij} + \sum_{jk\in A_R, j\neq r} c_{jk}X_{jk} + \sum_{i\in F} f_i z_i$$

$$X[V_L \setminus W, W] \geq 1 \qquad \forall W \subset V_L\setminus\{r\}, W\cap R \neq \emptyset \tag{5.49}$$

$$(5.5), (5.8), (5.47), (5.48)$$

Inequalities (5.49) are cuts on LG_x between sets that contain the root and at least one customer respectively.
We define the projection of $(\mathbf{X}', \mathbf{z}') \in \mathcal{P}_{LG_xCUT_{\{F,R\}}}$ onto the space of $(\mathbf{x}, \mathbf{z})$ variables as follows: $x_{rj} := {X'}^1_{rj}$ for all rj in A_S; $x_{ij} := \sum_{p=1}^{H} {X'}^p_{ij}$ for all ij in A_S with j in F;$x_{ij} := \sum_{p=1}^{H-1} {X'}^p_{ij}$ for all ij in A_S with j in $S\setminus F$;$x_{jk} := {X'}_{jk}$ for all jk in A_R, $j\neq r$; $x_{rj} := {X'}^1_{rj}$ for all rj in A_R and $z_i := z'_i$ for all i in F.

Layered Core and Assignment Graph $LG_{x,z}$

Consider a graph $LG_{x,z} = (V_L, A_{x,z})$ where V_L is defined as above and:

$$A_{x,z} := \bigcup_{i=1}^{3} A_i \cup A_6 \cup A_7 \text{ where}$$

A_1, A_2, A_3 and A_6 are defined as for A_x and

$$A_7 = \{((j,p),k) : 1 \leq p \leq H, jk \in A_R, j \neq r\}.$$

In this directed graph the set of facilities is given by $\{(i,p) : i \in F, p = 1,\ldots,H\}$. The facility opening costs are f_i for all (i,p) with $p = 1,\ldots,H$. Assignment costs are c_{jk} for all $((j,p),k)$ in A_7. The arc costs between (i,p) and $(j,p+1)$ are given as c_{ij}.

Lemma 12. *Any HC ConFL instance can be transformed into an equivalent directed ConFL instance on the layered graph $LG_{x,z}$ as described above.*

Figure 5.2.5 illustrates the transformation of an instance for HC ConFL to an instance for ConFL on $LG_{x,z}$.
We link binary variables X to the arcs in A_1 to A_3 and A_6 as above and to arcs in A_7 as follows: X^p_{jk} corresponds to $((j,p),k) \in A_7$. Additionally, we link variables Z^p_i to each (i,p) in F_L.

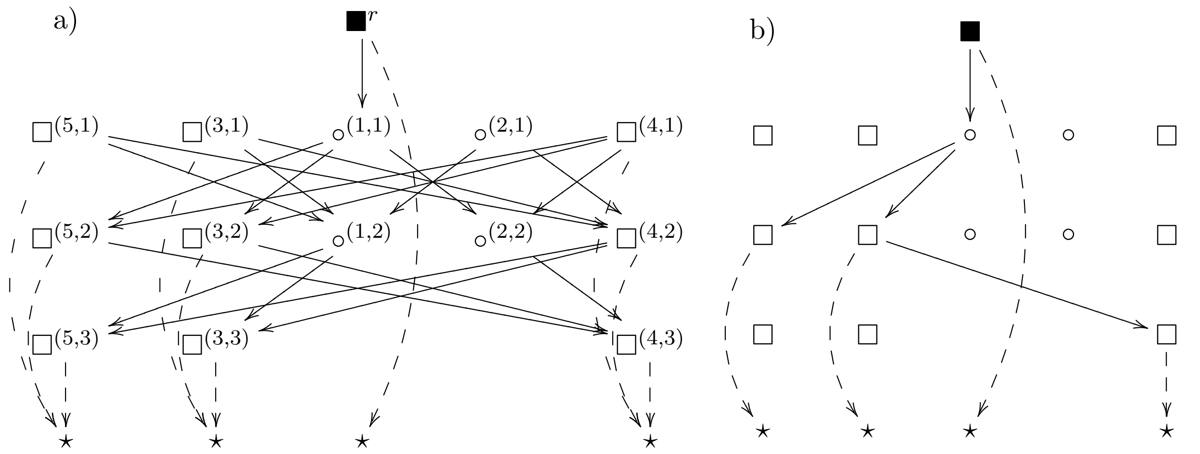


Figure 5.3: a) Layered graph $(V_L, A_{x,z})$ for Example 12; b) An optimal solution

Let $X[V_L \setminus W, W]$ denote the sum of all variables X_{ij}^p in the cut $\delta^-(W)$ in $LG_{x,z}$ defined by $W \subset V_L$ and $r \notin W$. Again, we can derive two formulations for HC ConFL on $LG_{x,z}$:

$$(LG_{x,z}CUT_F) \quad \min \sum_{rj\in A} c_{rj}X_{rj}^1 + \sum_{ij\in A, i\neq r} c_{ij}\sum_{p=1}^{H} X_{ij}^p + \sum_{i\in F\setminus\{r\}} f_i \sum_{p=1}^{H} Z_i^p + f_r z_r$$

$$X[V_L \setminus W, W] \geq Z_i^p \qquad \forall (i,p) \in F_L \cap W \neq \emptyset, r \notin W \tag{5.50}$$

$$X_{rk}^1 + \sum_{jk\in A_R, j\neq r}\sum_{p=1}^{H} X_{jk}^p = 1 \qquad \forall k \in R \tag{5.51}$$

$$X_{jk}^p \leq Z_j^p \qquad \forall jk \in A_R,\ p = 1,\dots,H,\ j \neq r \tag{5.52}$$

$$\mathbf{X} \in \{0,1\}^{|A_{x,z}|} \tag{5.53}$$

$$(5.5)$$

Constraints (5.50) are cuts on $LG_{x,z}$ between the root r and each facility at a level p, (i,p). Equalities (5.51) are assignment constraints. Inequalities (5.52) necessitate a facility at a level p to be open if customers are assigned to it.

Lemma 13. *We can replace connectivity cuts* (5.50) *by the following stronger ones:*

$$X[V_L \setminus W, W] \geq \sum_{p=1}^{H} Z_i^p \qquad \forall i \in F \setminus \{r\} : (i,p) \in F_L \cap W, r \notin W \tag{5.54}$$

Proof. The validity of these constraints follows from the fact that for each $i \in F$, the facilities (i,p) with $p = 1,\dots,H$ serve the same subset of customers with the same assignment costs. Therefore, any optimal solution will open at most one among those facilities, i.e. $\sum_{p=1}^{H} Z_i^p \leq 1$ for all i in $F \setminus \{r\}$. □

$$(LG_{x,z}CUT_R) \quad \min \sum_{rj \in A} c_{rj} X^1_{rj} + \sum_{ij \in A, i \neq r} c_{ij} \sum_{p=1}^{H} X^p_{ij} + \sum_{i \in F \setminus \{r\}} f_i \sum_{p=1}^{H} Z^p_i + f_r z_r$$

$$X[V_L \setminus W, W] \geq 1 \qquad \forall W \subset V_L \setminus \{r\}, W \cap R \neq \emptyset \tag{5.55}$$

$$(5.5), (5.52), (5.53)$$

Inequalities (5.55) are cuts on $LG_{x,z}$ between sets containing the root and a customer respectively. We define the projection of $(\mathbf{X}', \mathbf{Z}') \in \mathcal{P}_{LG_{x,z}CUT_{\{F,R\}}}$ onto the space of $(\mathbf{x}, \mathbf{z})$ variables as follows: $x_{rj} := X'^1_{rj}$ for all rj in A_S; $x_{ij} := \sum_{p=1}^{H} X'^p_{ij}$ for all ij in A_S with j in F; $x_{ij} := \sum_{p=1}^{H-1} X'^p_{ij}$ for all ij in A_S with j in $S \setminus F$; $x_{jk} := \sum_{p=1}^{H} X'^p_{jk}$ for all jk in A_R, $j \neq r$; $x_{rj} := X'^1_{rj}$ for all rj in A_R and $z_i := \sum_{p=1}^{H} Z'^p_i$ for all i in F.

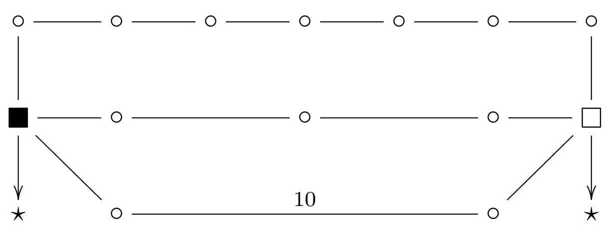
Modelling HC ConFL as STP on a Layered Graph

In Section 3.3.1 we have shown that by splitting facility nodes one can model ConFL as the Steiner arborescence problem on the transformed graph. If we apply this transformation to the corresponding instances on the layered graphs LG_x and $LG_{x,z}$, we end up with two ways of formulating HC ConFL as the Steiner arborescence problem.
However, we have shown for ConFL that this transformation does not lead to improved LP lower bounds. Thus, we do not consider the Steiner arborescence models in the theoretical discussion provided below.

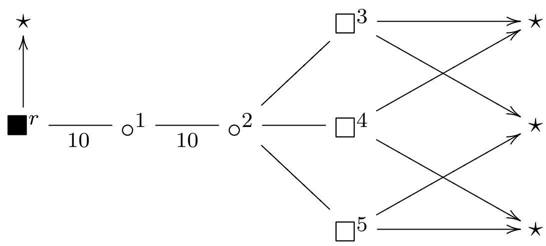
5.3 Polyhedral Comparison

In this section we provide a theoretical comparison of the MIP models described above with respect to optimal values of their LP-relaxations. The examples given below are used in the proofs of this section. In Example 11 $H = 5$, in all other examples $H = 3$.

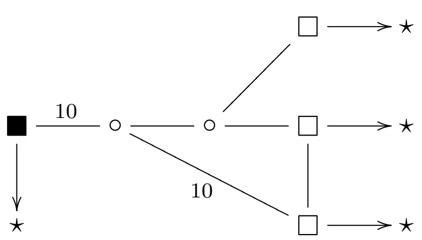
Example 9.



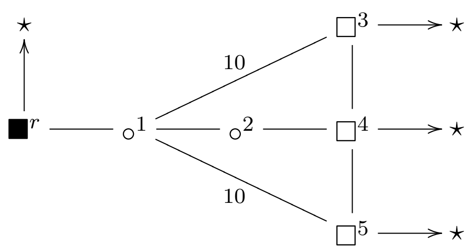
Example 10.



Example 11.



Example 12.



Let $v_{LP}(.)$ denote the optimal solution value of the LP relaxation of a given model. By comparing the optimal LP solution values for the aforementioned examples, provided by the models in Section 5.2, we can state the following

	Ex. 9	Ex. 10	Ex. 11	Ex. 12
OPT	16.00	29.00	22.00	30.00
$HCMTZ$	8.25	18.00	21.50	13.54
$HOP_{\{F,R\}}$	16.00	29.00	21.50	30.00
CUT_F^P	8.50	18.00	22.00	19.80
CUT_R^P	8.50	28.00	22.00	19.80
$HCMCF_F$	16.00	18.00	22.00	21.00
$HCMCF_R$	16.00	28.00	22.00	21.00
HD_F	16.00	18.00	22.00	30.00
HD_R	16.00	28.00	22.00	30.00

Table 5.1: Optimal LP solutions for Examples 9 - 12

Lemma 14. *The following pairs of formulations are incomparable with respect to the quality of lower bounds:*
a) $HOP_{\{F,R\}}$ *and* $CUT^P_{\{F,R\}}$ *and b)* $HOP_{\{F,R\}}$ *and* $HCMCF_{\{F,R\}}$

Proof. Consider the optimal LP solution values for Examples 9, 11 and 12 in Table 5.1. □

Denote by $\mathcal{P}_.$ the polytope of the LP-relaxation of any of the MIP models described above, and by $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_.)$ the natural projection of that polytope onto the space of variables $\mathbf{x}$ and $\mathbf{z}$.

Lemma 15. *The following results hold:*

a) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_{x,z}CUT_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_{x,z}CUT_F})$,
b) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_xCUT_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_xCUT_F})$,
c) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HD_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HD_F})$,
d) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HOP_R}) = Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HOP_F})$
e) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HCMCF_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HCMCF_F})$,
f) $\mathcal{P}_{CUT_R^P} \subset \mathcal{P}_{CUT_F^P}$ *and*
g) $\mathcal{P}_{CUT_R^J} \subset \mathcal{P}_{CUT_F^J}$.

Proof.

a), b) The inclusions follow directly from the results for ConFL in Chapter 3.

c) To show that every $(\mathbf{x},\mathbf{z},\mathbf{f})$ in $\mathcal{P}_{HD_R}$ can be projected into $\mathcal{P}_{HD_F}$ one needs to adapt the proof provided for the common flow models for ConFL in Chapter 3.

d) To prove the relation we describe a mapping from any solution of HOP_F to a solution of HOP_R of the same objective value and vice versa.

$Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HOP_R}) \subseteq Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HOP_F})$ Let $(\mathbf{X},\mathbf{z}) \in \mathcal{P}_{HOP_R}$. Then, w.l.og., for any facility j equation $z_j = \max_{k\in R}\sum_{p=1}^{H+1} X^p_{jk}$ holds. Let $(\mathbf{X}',\mathbf{x}',\mathbf{z}')$ be defined as: $z'_j := z_j$ for all j in F; $X'^p_{ij} := X^p_{ij}$ for all ij in A_S, $p = 1,\dots,H$ and $x'_{jk} := \sum_{p=1}^{H+1} X^p_{jk}$ for all jk in A_R. Then $(\mathbf{X}',\mathbf{x}',\mathbf{z}') \in \mathcal{P}_{HOP_F}$: Inequalities (5.41) and (5.43) follow from (5.36) and (5.39). Constraints (5.3) and (5.4) are implied by (5.38) and (5.37) respectively. For all $j \in F\setminus\{r\}$ let $k^j := \arg\max_{k\in R}\sum_{p=2}^{H+1} X^p_{jk}$. Then, w.l.o.g., we have $z_j = \sum_{p=2}^{H+1} X^p_{jk^j}$ and further we have

$$z'_j = z_j = \sum_{p=2}^{H+1} X^p_{jk^j} \overset{(5.36)}{\leq} \sum_{p=2}^{H+1}\sum_{ij\in A_S} X^{p-1}_{ij} = \sum_{p=1}^{H}\sum_{ij\in A_S} X'^p_{ij},$$

hence equations (5.42) also hold for $(\mathbf{X}',\mathbf{x}',\mathbf{z}')$.

$Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HOP_F}) \subseteq Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HOP_R})$ Let $(\mathbf{X}', \mathbf{x}', \mathbf{z}') \in \mathcal{P}_{HOP_F}$ and $(\mathbf{X}, \mathbf{z})$ defined as $z_j := z'_j$ for all j in F and $X^p_{ij} := X'^p_{ij}$ for all ij in A_S, $p = 1, \ldots, H$. Let $x'_{jk} > 0$ with $j \in F\backslash\{r\}, k \in R$. From equations (5.42) and (5.4) we have $x'_{jk} \leq z'_j \leq \sum_{ij \in A_S} \sum_{p=1}^{H} X'^p_{ij}$. From the right hand side we can select $X^{*p}_{ij,k}$ with $X^{*p}_{ij,k} \leq X'^p_{ij}$ such that $\sum_{ij \in A_S} \sum_{p=1}^{H} X^{*p}_{ij,k} = x'_{jk}$. We can do this for all $jk \in A_R$. Let

$$X^{p+1}_{jk} := \sum_{ij \in A_S} X^{*p}_{ij,k} \qquad \forall j \in F\backslash\{r\}, k \in R,\ p = 1, \ldots, H \tag{5.56}$$

and

$$X^1_{rk} := x'_{rk} \qquad \forall k \in R.$$

Then we can show that $(\mathbf{X}, \mathbf{z}) \in \mathcal{P}_{HOP_R}$:
Equations (5.36) follow from the definitions. Constraints (5.38) follow from

$$\sum_{p=1}^{H+1} X^p_{jk} \overset{(5.43)}{=} 0 + \sum_{p=2}^{H+1} X^p_{jk} \overset{(5.56)}{=} \sum_{ij \in A_S} \sum_{p=1}^{H} X^{*p}_{ij,k} = x'_{jk} \leq z'_j = z_j \qquad \forall jk \in A_R.$$

Constraints (5.37) follow from

$$\sum_{p=1}^{H+1} \sum_{jk \in A_R} X^p_{jk} = X^1_{rk} + \sum_{p=1}^{H} \sum_{ij \in A_S} X^{*p}_{ij,k} = \tag{5.57}$$

$$= X^1_{rk} + \sum_{ij \in A_S} \sum_{p=1}^{H+1} X^{*p}_{ij,k} = \tag{5.58}$$

$$= \sum_{jk \in A_R} x'_{jk} \overset{(5.3)}{=} 1. \tag{5.59}$$

e) The relation holds for the case without hop constraints (cf. Chapter 3). From equations (5.17) we have $\sum_{ik \in A_R} f^k_{ik} = 1$ for all $k \in R$. Thus, we have $\sum_{ij \in A_S} f^k_{ij} \leq H$ for all $k \in R$. These constraints are at least as strong as (5.16). Therefore, $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HCMCF_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HCMCF_F})$.

f) We have shown the relation for the case without hop constraints (i.e. $H = |S| - 1$) in Chapter 3. Constraints (5.2) dominate (5.11), thus $\mathcal{P}_{CUT^P_R} \subset \mathcal{P}_{CUT^P_F}$ holds for the hop constrained case as well.

g) The inclusion follows from arguments analogously to the ones in f).

□

Lemma 16. *The projections $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_.)$ of sets of feasible LP solutions of the following formulations are identical:*
a) $LG_{x,z}CUT_R$ and LG_xCUT_R, b) $LG_{x,z}CUT_F$ and LG_xCUT_F

Proof. To prove the relation in a), we describe a mapping from any vector in $\mathcal{P}_{LG_xCUT_R}$ to a vector in $\mathcal{P}_{LG_{x,z}CUT_R}$ and vice versa, such that both have the same objective function value and that their projections onto the space of $(\mathbf{x}, \mathbf{z})$ are the same. The proof for b) uses the same arguments.

$Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_{x,z}CUT_R}) \subseteq Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_x CUT_R})$ Let $(\mathbf{X},\mathbf{Z}) \in Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_{x,z}CUT_R})$ and $(\mathbf{X}',\mathbf{z}')$ defined as follows: $X'^p_{ij} := X^p_{ij}$ for all edges in A_1, A_2 and A_3; $X'^p_{jj} := \max_{k\in R} X^p_{jk}$ for all edges in A_4; $X'_{jk} := \sum_{p=1}^{H} X^p_{jk}$ for all edges in A_5; $X'_{rk} := X^1_{rk}$ for all edges in A_6; $z_i := \sum_{p=1}^{H} Z^p_i$. Then, obviously, $(\mathbf{X}',\mathbf{z}') \in Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_x CUT_R})$.

$Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_x CUT_R}) \subseteq Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_{x,z}CUT_R})$ Let $(\mathbf{X}',\mathbf{z}') \in Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_x CUT_R})$ and $(\mathbf{X},\mathbf{Z})$ defined as follows: $X^p_{ij} := X'^p_{ij}$ for all edges in A_1, A_2 and A_3; $X^1_{rk} := X'_{rk}$ for all edges in A_6; $X^H_{jk} := \min(\delta^-((j,H)), X'_{jk})$ and $X^p_{jk} := \min(\delta^-((j,p)), x'_{jk} - \sum_{q=p+1}^{H} X'^q_{jk})$ recursively, starting with $p = H-1$. Then $(\mathbf{X},\mathbf{Z}) \in Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_{x,z}CUT_R})$.

□

Lemma 17. *The following results hold:*

a) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_x CUT_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HD_R}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HCMCF_R}) \subset \mathcal{P}_{CUT^P_R}$,

b) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{LG_x CUT_F}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HD_F}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HCMCF_F}) \subset \mathcal{P}_{CUT^P_F}$,

c) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HD_R}) \subset \mathcal{P}_{CUT^J_R} \subset \mathcal{P}_{CUT^P_R}$,

d) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HD_F}) \subset \mathcal{P}_{CUT^J_F} \subset \mathcal{P}_{CUT^P_F}$ *and*

e) $Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HD_F}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HOP_{\{F,R\}}}) \subset Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HCMTZ})$.

Proof. Strict inclusions in a) to e) follow from the optimal LP solution values in Table 5.1.

a) The first inclusion can be shown by adapting the proof in [24]. It is strict because of the example in the same paper. Note that by disaggregating constraints (5.30) and introducing variables X^p_{ij} formulation HD_R becomes an equivalent of formulation LG_R. $HCMCF_R$ is an aggregation of HD_R ($f^k_{ij} := \sum_{p=1}^{H+1} f^{kp}_{ij}$).
Let $(\mathbf{f},\mathbf{x},\mathbf{z}) \in \mathcal{P}_{HCMCF_R}$ and assume that there exists a path P of length $H+2$ such that $\sum_{ij\in P} x_{ij} > H+1$, i.e. $(\mathbf{x},\mathbf{z}) \notin \mathcal{P}_{CUT^P_R}$. Let further $i'j' := \arg\max_{ij\in P} x_{ij}$. Then there exists $k \in R$ such that $f^k_{i'j'} = x_{i'j'}$. We denote this amount of flow by $a > 0$. In the worst case, a units of flow are sent through the whole path P. The complementary flow of commodity k is sent outside of P. Even if only a single edge is used on this complementary route, constraints (5.19) imply $H+1 \geq (H+2)a + 1 - a = (H+1)a + 1$. Therefore, $a \leq \frac{H}{H+1}$. But then $\sum_{ij\in P} x_{ij} \leq (H+2)a < H+1$, which is a contradiction.

b) The inclusions follow from similar arguments as used in a).

c) Assume that $(\mathbf{x},\mathbf{z}) \in Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{HD_R})$ and $(\mathbf{x},\mathbf{z}) \notin Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CUT^J_R})$. Then there exists a $(H+1)$-jump J where $S_{H+2} \cap R \neq \emptyset$ and such that $\sum_{ij\in J} x_{ij} = 1-\epsilon$, and $\epsilon > 0$. Because of the flow preservation constraints (5.26) - (5.28) there needs to be a flow of ϵ on the path $P = \{ij : i \in S_i, j \in S_{i+1}, i = 0,\ldots,H+1\}$. This flow uses $H+2$ hops and cannot be composed of flow variables f^{kp}_{ij}, $p = 1,\ldots,H+1$, which is a contradiction.
Furthermore, assume that $(\mathbf{x},\mathbf{z}) \in Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CUT^J_R})$ and optimal and $(\mathbf{x},\mathbf{z}) \notin Proj_{\mathbf{x},\mathbf{z}}(\mathcal{P}_{CUT^P_R})$. Let $P' = \{(r,i_1),(i_1,i_2),\ldots,(i_H,i_{H+1}),(i_{H+1},i_{H+2})\}$ be the path for which constraint (5.2) is violated, i.e. inequality $\sum_{ij\in P'}(x_{ij}+x_{ji}) > H+1$ holds. For the jump J with $S_0 = \{r\}$, $S_1 = \{i_1\}$, $S_2 = \{i_2\}$, ..., $S_H = \{i_H\}$, $S_{H+1} = \{i_{H+1}\}$, $S_{H+2} = V \setminus \{r, i_1, \ldots, i_{H+1}\}$ we have $\sum_{ij\in J} x_{ij} \geq 1$. By adding these two inequalities we get

$$\sum_{j=1}^{H+2} \sum_{ki_j\in A} x_{ki_j} \geq \sum_{j=1}^{H+2} \sum_{ki_j\in J\cup P'} x_{ki_j} > H+3,$$

thus, for at last one $j \in \{1, \dots, H+2\}$, the in-degree of i_j is strictly greater than 1, which is a contradiction to $(\mathbf{x}, \mathbf{z})$ being an optimal LP solution.
In a similar way one can show that the jump formulation for HCSTP is stronger than the path formulation. (Gouveia [23]).

d) See the arguments in c).

e) Let $(\mathbf{x}, \mathbf{z}, \mathbf{g})$ be an arbitrary solution for the LP-relaxation of HD_F and let $(\mathbf{X}', \mathbf{x}', \mathbf{z}')$ be defined as follows: ${x'}_{jk} := x_{jk}$ for all jk in A_R; $z'_j := z_j$ for all j in F and ${X'}^p_{ij} := \max_{k \in F} g^{kp}_{ij}$ for all ij in A_S, $p = 1, \dots, H$. Then $(\mathbf{X}', \mathbf{x}', \mathbf{z}') \in \mathcal{P}_{HOP_F}$:
From equations (5.22) and the definition of X' we have

$$z_j = \sum_{p=1}^{H} \sum_{ij \in A_S} g^{jp}_{ij} \leq \sum_{p=1}^{H} \sum_{ij \in A_S} \max_{k \in F} g^{kp}_{ij} = \sum_{p=1}^{H} \sum_{ij \in A_S} {X'}^p_{ij}$$

for compliance with equations (5.42). With $k^* := \arg\max_{k \in F} g^{kp}_{ij}$ estimations

$$ {X'}^p_{ij} = \max_{k \in F} g^{kp}_{ij} \overset{(5.20)}{\leq} \sum_{\substack{l \in S \setminus \{j\}: \\ li \in A_S}} g^{k^*,p-1}_{li} \leq \sum_{\substack{l \in S \setminus \{j\}: \\ li \in A_S}} \max_{k \in F} g^{k,p-1}_{li} = \sum_{\substack{l \in S \setminus \{j\}: \\ li \in A_S}} {X'}^{p-1}_{li}$$

give equations (5.41). Constraints (5.3) - (5.5), (5.7), (5.8) are common in both models and thus met trivially. *HCMTZ* is an aggregation of *HOP* (cf. [50]; $u_j := \sum_{p=1}^{H} p X^p_{ij}$).

□

5.3.1 Full Hierarchy of Formulations

The hierarchical scheme given in Figure 5.4 summarizes the relationships between the LP relaxations of the MIP models considered throughout this chapter. An arrow specifies that the target formulation is strictly stronger than the source formulation. A double-headed arrow denotes formulations of equal strength. Whenever formulations are not comparable or we do not know their relation, this is not indicated in the figure for the sake of simplicity.

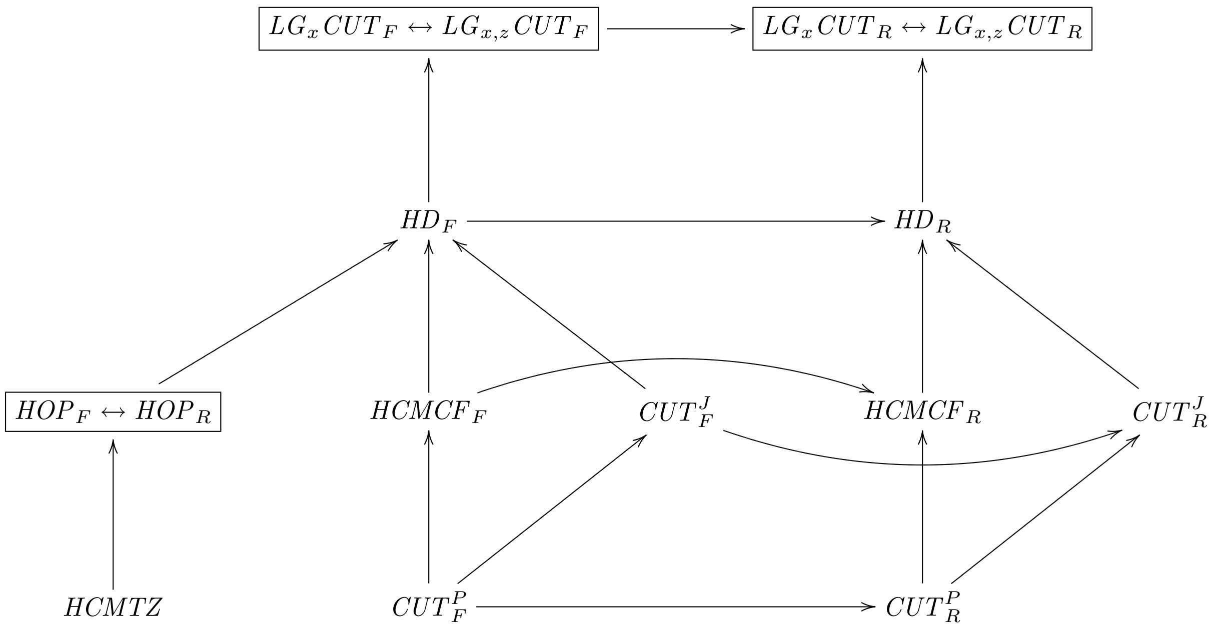


Figure 5.4: Relations between LP-relaxations of MIP models for ConFL

Chapter 6

Conclusion

We provide a first theoretical comparison of MIP models for ConFL. We show that there are basically two groups of models, derived from the way the connectivity requirements in the whole graph are defined. Our "F" models require connectivity between open facilities and the root node, and in addition a proper assignment of customers. We derive the stronger "R" models by requiring connectivity between customers and the root node. We also present the weak Miller-Tucker-Zemlin formulation. It follows a sub-tour elimination concept instead of a connectivity-based one. In contrast to known results for the traveling salesman problem [53], we show that *MTZ* is not dominated by the two single commodity flow models. The second interesting result is that the integrality gap of all "F" models is not a constant value.
In our computational study we also obtain two surprising results. First, the branch-and-cut algorithm for the correspondingly weaker "F" cut based model significantly outperforms all other models in practice. Second, the weak but small *MTZ* formulation performs comparatively well, and in most cases outperforms even the branch-and-cut derived for the stronger "R" model.

Following the theoretical results for ConFL we introduce the Hop Constrained ConFL. We provide an extensive theoretical comparison of LP relaxations of 15 MIP models for it. We also introduce new sets of inequalities to model the corresponding (prize-collecting) HCSTP and HCMST. In particular, directed path constraints are shown to be strictly stronger than those originally proposed by Costa et al. [10]. To model the prize-collecting HCSTP we introduce a new set of jump inequalities.
We follow the same concept as for ConFL to basically derive two groups of models. We describe a transformation of the HC ConFL into the ConFL on two variants of a layered graph. This leads to the strongest models in the presented hierarchy. A disaggregation of variables indicating facilities in the layered graphs turns out not to improve the quality of the LP lower bounds.
The relation between the jump formulation and the two models based on path constraints and hop indexed multi-commodity flows extends to HCSTP and HCMST. The relation between jump constraints and the ones derived for the multi-commodity flow formulation remains an open question for all three problems, HC ConFL, HCSTP and HCMST. We believe that formulation *HCMTZ* is weaker than CUT_F^P. This relation is not known for HCSTP and HCMST either.

Bibliography

[1] UflLib. URL http://www.mpi-inf.mpg.de/departments/d1/projects/benchmarks/UflLib/.

[2] IBM CPLEX. URL http://www.ilog.com/products/cplex/.

[3] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows*. Prentice Hall, 1993.

[4] A. Balakrishnan and K. Altinkemer. Using a hop-constrained model to generate alternative communication network design. *INFORMS Journal on Computing*, 4(2):192–205, 1992.

[5] M. G. Bardossy and S. Raghavan. Dual-based local search for the connected facility location and related problems. Technical report, Smith School of Business and Institute for Systems Research, University of Maryland, 2009.

[6] J. E. Beasley. OR-library. URL http://people.brunel.ac.uk/~mastjjb/jeb/orlib/steininfo.html.

[7] S. Chen, I. Ljubić, and S. Raghavan. The regenerator location problem. *Networks*, 2009. to appear.

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

[9] A. M. Costa, J.-F. Cordeau, and G. Laporte. Fast heuristics for the Steiner tree problem with revenues, budget and hop constraints. *European Journal of Operational Research*, 190 (1):68–78, 2008.

[10] A. M. Costa, J.-F. Cordeau, and G. Laporte. Models and branch-and-cut algorithms for the Steiner tree problem with revenues, budget and hop constraints. *Networks*, 53(2):141–159, 2009.

[11] G. Dahl, L. Gouveia, and C. Requejo. On formulations and methods for the hop-constrained minimum spanning tree problem. In P. M. Pardalos and M. Resende, editors, *Handbook of Optimization in Telecommunications*, pages 493–515. Springer, 2006.

[12] G. Dantzig, D. Fulkerson, and S. Johnson. Solution of a large-scale traveling-salesman problem. *Operations Research*, 2:393–410, 1954.

[13] M. Desrochers and G. Laporte. Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints. *Operations Research Letters*, 10(1):27 – 36, 1991.

[14] F. Eisenbrand, F. Grandoni, T. Rothvoß, and G. Schäfer. Approximating connected facility location problems via random facility sampling and core detouring. In *SODA '08: Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1174–1183. SIAM, 2008.

[15] D. Ghosh. Neighborhood search heuristics for the uncapacitated facility location problem. *European Journal of Operations Research*, 150:150–162, 2003.

[16] M. X. Goemans. The Steiner tree polytope and related polyhedra. *Mathematical Programming*, 63:157–182, 1994.

[17] M. X. Goemans and Y. Myung. A catalog of Steiner tree formulations. *Networks*, 23, 1993.

[18] S. Gollowitzer and I. Ljubić. MIP models for connected facility location: A theoretical and computational study. Technical Report Nr. [2009-07], Department of Statistics and Decision Support Systems, Faculty of Business, Economics, and Statistics, University of Vienna, Austria, 2009.

[19] L. Gouveia. Using the Miller-Tucker-Zemlin constraints to formulate a minimal spanning tree problem with hop constraints. *Comput. Oper. Res.*, 22(9):959–970, 1995.

[20] L. Gouveia. Multicommodity flow models for spanning trees with hop constraints. *European Journal of Operational Research*, 95(1):178–190, November 1996.

[21] L. Gouveia. Using variable redefinition for computing lower bounds for minimum spanning and Steiner trees with hop constraints. *INFORMS J. on Computing*, 10(2):180–188, 1998.

[22] L. Gouveia. Using hop-indexed models for constrained spanning and Steiner tree models. In B. Sanso and P. Soriano, editors, *Telecommunications network planning*, pages 21–32. Kluwer, 1999.

[23] L. Gouveia. Personal communication.

[24] L. Gouveia, L. G. Simonetti, and E. Uchoa. Modelling hop-constrained and diameter-constrained minimum spanning tree problems as Steiner tree problems over layered graphs. *Optimization Online*, 2008.

[25] A. Gupta, J. Kleinberg, A. Kumar, R. Rastogi, and B. Yener. Provisioning a virtual private network: A network design problem for multicommodity flow. In *In Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 389–398, 2001.

[26] F. K. Hwang and D. S. Richards. Steiner tree problems. *Networks*, 22:55–89, 1992.

[27] F. K. Hwang, D. S. Richards, and P. Winter. *The Steiner Tree Problem*, volume 53 of *ADM*. North-Holland, Amsterdam, Netherlands, 1992.

[28] H. Jung, M. Hasan, and K.-Y. Chwa. Improved primal-dual approximation algorithm for the connected facility location problem. *Combinatorial Optimization and Applications*, pages 265–277, 2008.

[29] D. R. Karger and M. Minkoff. Building Steiner trees with incomplete global knowledge. In *FOCS '00: Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, pages 613–623, 2000.

[30] S. Khuller and A. Zhu. The general Steiner tree-star problem. *Information Processing Letters*, 84(4):215–220, 2002.

[31] M. Koerkel. On the exact solution of large-scale simple plant location problems. *European Journal of Operations Research*, 39:157–173, 1989.

[32] J. Kratica, D. Tošić, V. Filipović, and I. Ljubić. Solving the simple plant location problem by genetic algorithms. *RAIRO - Operations Research*, 35(1):127–142, 2001.

[33] C. Krick, H. Räcke, and M. Westermann. Approximation algorithms for data management in networks. In *SPAA '01: Proceedings of the thirteenth annual ACM symposium on Parallel algorithms and architectures*, pages 237–246. ACM, 2001.

[34] Y. Lee, L. Lu, Y. Qiu, and F. Glover. Strong formulations and cutting planes for designing digital data service networks. *Telecommunication Systems*, 2(1):261–274, 1993.

[35] Y. Lee, Y. Chiu, and J. Ryan. A branch and cut algorithm for a Steiner tree-star problem. *INFORMS Journal on Computing*, 8(3):194–201, 1996.

[36] I. Ljubić. A hybrid VNS for connected facility location. In T. Bartz-Beielstein, M. J. B. Aguilera, C. Blum, B. Naujoks, A. Roli, G. Rudolph, and M. Sampels, editors, *Hybrid Metaheuristics*, volume 4771 of *Lecture Notes in Computer Science*, pages 157–169. Springer, 2007.

[37] I. Ljubić. *Exact and Memetic Algorithms for Two Network Design Problems.* PhD thesis, Faculty of Computer Science, Vienna University of Technology, November 2004.

[38] I. Ljubić, R. Weiskircher, U. Pferschy, G. Klau, P. Mutzel, and M. Fischetti. An algorithmic framework for the exact solution of the prize-collecting Steiner tree problem. *Mathematical Progamming, Series B*, 105(2-3):427–449, 2006.

[39] A. Lucena and M. G. C. Resende. Strong lower bounds for the prize-collecting Steiner problem in graphs. *Discrete Applied Mathematics*, 141:277–294, 2004.

[40] N. Maculan. The Steiner tree problem in graphs. *Annals of Discrete Mathematics*, 31: 185–212, 1987.

[41] T. Magnanti and L. Wolsey. Optimal trees. *Handbook in Operations Research and Management Science*, pages 503–615, 1995.

[42] P. Manyem. Constrained spanning, Steiner trees and the triangle inequality. In C. Pearce and E. Hunt, editors, *Optimization*, volume 32 of *Springer Optimization and Its Applications*, pages 355–367. Springer New York, 2009.

[43] P. Manyem and M. F. M. Stallmann. Some approximation results in multicasting. Technical report, Raleigh, NC, USA, 1996.

[44] C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *Journal of the ACM*, 7(4):326–329, 1960.

[45] T. Öncan, I. K. Altınel, and G. Laporte. Invited review: A comparative analysis of several asymmetric traveling salesman problem formulations. *Comput. Oper. Res.*, 36(3):637–654, 2009.

[46] M. Padberg and T.-Y. Sung. An analytical comparison of different formulations of the travelling sales man problem. *Mathematical Programming Series A and B*, 52(2):315–357, 1991.

[47] T. Polzin and S. V. Daneshmand. A comparison of Steiner tree relaxations. *Discrete Applied Mathematics*, 112(1-3):241–261, 2001.

[48] C. Swamy and A. Kumar. Primal-dual algorithms for connected facility location problems. *Algorithmica*, 40:245–269, 2004.

[49] A. Tomazic and I. Ljubić. A GRASP algorithm for the connected facility location problem. In *SAINT*, pages 257–260. IEEE Computer Society, 2008.

[50] S. Voß. The Steiner tree problem with hop constraints. *Annals of Operations Research*, 86: 321–345, 1999.

[51] D. P. Williamson and A. van Zuylen. A simpler and better derandomization of an approximation algorithm for single source rent-or-buy. *Operations Research Letters*, 35(6): 707–712, 2007.

[52] P. Winter. Steiner problem in networks: a survey. *Networks*, 17(2):129–167, 1987. ISSN 0028-3045. doi: http://dx.doi.org/10.1002/net.3230170203.

[53] R. T. Wong. Integer programming formulations of the traveling salesman problem. *Proceedings of the IEEE international conference of circuits and computers*, pages 149–52, 1980.

[54] J. Xu, S. Y. Chiu, and F. Glover. Tabu search for dynamic routing communications network design. *Telecommunication Systems*, 8(1):55–77, 1997.