A Node Splitting Technique for Two Level Network Design Problems with Transition Nodes

Stefan Gollowitzer, Luis Gouveia and Ivana Ljubić

Abstract The Two Level Network Design (TLND) problem arises when local broadband access networks are planned in areas, where no existing infrastructure can be used, i.e., in the so-called *greenfield deployments*. Mixed strategies of Fiber-To-The-Home and Fiber-To-The-Curb, i.e., some customers are served by copper cables, some by fiber optic lines, can be modeled by an extension of the TLND.
We are given two types of customers (primary and secondary), an additional set of Steiner nodes and fixed costs for installing either a primary or a secondary technology on each edge. The TLND problem seeks a minimum cost connected subgraph obeying a tree-tree topology, i.e., the primary nodes are connected by a rooted primary tree; the secondary nodes can be connected using both primary and secondary technology. In this paper we study an important extension of TLND in which additional transition costs need to be paid for *intermediate facilities* placed at the transition nodes, i.e., nodes where the change of technology takes place. We call this problem TLNDF.
The introduction of transition node costs leads to a problem with a rich structure permitting us to put in evidence reformulation techniques such as modeling in higher dimensional graphs (which in this case are based on a node splitting technique).
We first provide a compact way of modeling intermediate facilities. We then present several generalizations of the facility-based inequalities involving an exponential number of constraints. Finally we show how to model the problem in an extended graph based on node splitting. Our main result states that the connectivity constraints on the splitted graph, projected back into the space of the variables of the original model, provide a new family of inequalities that implies, and even strictly dominates, all previously described cuts. We also provide a polynomial time separation algorithm for the more general cuts by calculating maximum flows *on the splitted graph*. We compare the proposed models both theoretically and computationally.

Stefan Gollowitzer
Department of Statistics and Operations Research, University of Vienna, Austria
e-mail: `stefan.gollowitzer@univie.ac.at`

Luis Gouveia
Departamento de Estatística e Investigação Operacional - Centro de Investigação Operacional, Faculdade de Ciênçias, Universidade de Lisboa, Portugal
e-mail: `legouveia@fc.ul.pt`

Ivana Ljubić
Department of Statistics and Operations Research, University of Vienna, Austria
e-mail: `ivana.ljubic@univie.ac.at`

1 Introduction

The Two Level Network Design (TLND) problem arises in the topological design of hierarchical communication, transportation, and electric power distribution networks. Probably the most important application of TLND is in the context of telecommunication networks, where networks with two cable technologies, fiber optic and copper, are built. In local broadband access networks, if the Fiber-To-The-Home (FTTH) strategy is used, every customer is provided with a distinct fiber optic connection. A cheaper strategy is Fiber-To-The-Curb (FTTC), where the part of the access network closest to the customer uses copper cables and facilities are installed, to convert optical to electrical signals and vice versa. In greenfield deployments, i.e. where there is no existing infrastructure, a mixed strategy of FTTC and FTTH is often preferable. In such a case, telecommunication companies distinguish between *primary* and *secondary customers*. The switching centers, important infrastructure nodes and small businesses are considered as primary customers (i.e., those to be served by fiber optic connections). Single households are not considered as being consumers of a high potential and hence they only need to be supplied using copper cables. The secondary technology is much cheaper, but the guaranteed quality of the connections and bandwidth is significantly below the quality provided by the primary one.

A large body of work has been done for the TLND and its variations (see below). In this study we incorporate two realistic features that have not yet been considered in previous studies of the TLND. Firstly, none of the previous approaches on TLND considers the cost of establishing *intermediate facilities* at *transition nodes*, i.e., nodes in which the change of technology takes place. Typically, at transition nodes, expensive switching devices need to be installed and the respective costs should not be neglected. Secondly, the previous work on the TLND is based on the assumption that all nodes in the network belong to the customer set. We relax this assumption, allowing the existence of *Steiner* nodes as well. We call the new problem the Two Level Network Design Problem with Intermediate Facilities (TLNDF).

This important problem generalizes problems with tree-star and star-tree topologies, like e.g., connected facility location, hierarchical network design, Steiner trees or uncapacitated facility location. We consider an extended graph, where the installation of facilities is modeled as arcs. We show that connectivity constraints on this splitted graph, projected back into the space of the variables of the original model, provide a new family of inequalities that implies, and even strictly dominates, all previously described constraints. We also provide a polynomial time separation algorithm for the more general inequalities by calculating maximum flows *on the splitted graph*. Finally, our computational study demonstrates the efficiency and practical applicability of the new inequalities.

1.1 Problem Definition

We consider the following *generalization* of the two level network design problem:

Definition 1 (TLNDF). We are given an undirected graph $G = (V,E)$ with a root $r \in V$ and a set of customers $R \subseteq V \setminus \{r\}$. To each edge $e \in E$ we associate two installation costs, $c_e^1 \geq c_e^2 \geq 0$. These correspond to the *primary* and *secondary technology*, respectively. The set of customers, R, is partitioned into the set of *primary* and *secondary customers* P and S, respectively. Our goal is to determine a cost-minimal *subtree* of G satisfying the following properties:

(P) each *primary node* in P is connected to the root node by a path that consists of primary edges only,
(S) each *secondary node* in S is connected to the root by a path consisting of primary and/or secondary edges,
(F) facility opening costs $f_i \geq 0$ are payed for each transition node $i \in V$ and
(E) on each edge $e \in E$ at most one of technologies is installed.

Several observations can be made about the solution space of this problem: i) Since $c_e^1 - c_e^2 \geq 0$, there always exists an optimal solution which is a Steiner tree with a *tree-tree topology*, i.e., it is composed of a rooted subtree of primary edges (*primary subtree*) and a union of subtrees of secondary edges (*secondary subtrees*). Each secondary subtree is rooted in a (transition) node of the primary subtree. ii) If facility opening costs are the same for all facility locations, any leaf of the primary subtree will be a primary node. Otherwise, if facility opening costs are location-dependent, placing facilities at locations of Steiner nodes may provide cheaper solutions, i.e., a leaf of the primary subtree may be any node from $V \setminus \{r\}$. iii) This general definition also covers the case in which potential facility locations are a true subset of V (which can be modeled by setting $f_i := \infty$ for the non-facility locations).

As we noted before, the problem discussed here incorporates two new features when compared to the original definition given in Balakrishnan et al. [3], see also Duin and Volgenant [7]. Firstly, the need to consider additional *transition costs* due to the presence of two technologies on the network. The second new feature is that we allow arbitrary subsets of $V \setminus \{r\}$ to be considered as the customer set. This is because in practical applications nodes like street intersections need to be considered as well. Following the spanning tree definition of *multi-level network design* problems given in [2], the TLND problem with Steiner nodes can also be seen as a *three-level network design* problem in which the Steiner nodes are assigned to the third group of customers and the installation costs for the third technology are set to zero.

Literature Review

The concept of two level network design problems (more precisely, *two-level spanning trees*) has been developed in the 80's and early 90's. The *hierarchical network design* problem, in which $R = V \setminus \{r\}$ and $|P| = 2$, was the "initial" variant of the TLND introduced by Current et al. [6]. This problem was later generalized by Duin and Volgenant [7] for $|P| > 2$. Balakrishnan et al. [3] have proposed several network flow based models for this latter problem setting and have compared the linear programming bounds of the proposed formulations. In Balakrishnan et al. [2], the authors have tested a dual ascent method on the model with the strongest linear programming bound. A more recent approach based on a different formulation is described in Gouveia and Telhada [12]. The TLND problem belongs to a class of problems with a tree-tree topology. The reader is referred to [9] where several variants of related problems such as star-tree, tree-star and star-star problems as well as other variants of tree-tree problems are described.

The previous studies on TLND do not incorporate additional constraints. As far as we know, the three exceptions are described in [10, 11, 8]. In the first one, the authors considered the TLND with *weighted hop constraints* defined as follows: given natural numbers w_1 and w_2, our goal is to construct a two-level minimum spanning tree such that for each node k, the unique path from the root to k contains a weighted number of primary and secondary edges (with weights w_1 and w_2, respectively) which does not exceed H. In [11] the two-level minimum spanning tree problem with *secondary distance constraints* stating that each secondary node must not be too far from the primary network, is considered. In the latter work [8], the authors studied the connected facility location problem (ConFL) wich is a TLNDF variant with a tree-star configuration. In [16] a hop constrained variant of connected facility location has been studied.

For a literature overview on *capacitated network design problems* with two technologies, we refer to a recent work of Costa et al. [5], where a problem has been studied with capacities on edges and with fixed installation and non-linear flow costs. In Jongh et al. [13] a *survivable network design problem* with two technologies and facility nodes has been studied.

2 MIP Formulations for the Two Level Network Design Problem

In this section we describe cut based formulations for the TLNDF. We start by presenting a formulation of the original TLND problem *without* modeling the facility opening costs.

Directed Graphs

It is well known that for rooted spanning or Steiner tree problems, models with a stronger linear programming bound are obtained by solving the problem on a directed graph (see, e.g., Magnanti and Wolsey [17]). Thus, we will work on a directed graph $G = (V,A)$ that is obtained from the original undirected graph $G = (V,E)$ as follows: For each edge $e = \{i, j\} \in E$ we include two arcs ij and ji in A with the same cost of the original edge. Since we are modeling an arborescence directed away from the root node, edges $\{r, j\}$ are replaced by a single arc rj only.

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

$$x^1_{ij} = \begin{cases} 1, & \text{if the primary cable technology is installed on arc } ij \\ 0, & \text{otherwise} \end{cases} \qquad \forall ij \in A$$

$$x^2_{ij} = \begin{cases} 1, & \text{if the secondary cable technology is installed on arc } ij \\ 0, & \text{otherwise} \end{cases} \qquad \forall ij \in A, j \notin P$$

Observe that in any feasible solution there will be no secondary arcs entering a primary node (i.e., $x^2_{ij} = 0$, whenever $j \in P$). Therefore, variables corresponding to such arcs will not be considered in our models. However, to simplify the notation, we will allow them in the indexation of the summation terms.

For any $W \subset V$ we denote its complement set by $W^c = V \setminus W$. For any $M, N \subset V$, $M \cap N = \emptyset$, denote the induced cut set of arcs by $(M, N) = \{ij \in A \mid i \in M, j \in N\}$. In particular, let $\delta^-(W) = (W^c, W)$ and $\delta^-(i) = (V \setminus \{i\}, \{i\})$. For a set of arcs $\hat{A} \subseteq A$, we will write $x^\ell(\hat{A}) = \sum_{ij \in \hat{A}} x^\ell_{ij}$, for $\ell = 1, 2$, and $(x^1 + x^2)(\hat{A}) = \sum_{ij \in \hat{A}} x^1_{ij} + x^2_{ij}$.

The examples described in the next sections use the following symbols: ▩ represents the root node, ∘ represents a Steiner node. □ represents a primary customer, △ represents a secondary customer. Whenever we solve a problem as the Steiner tree problem, terminals are denoted by ◊.

2.1 Modeling the TLND Problem

The following formulation models the TLND with the set of primary nodes P (that may also be an empty set), and the set of secondary nodes S *without* facility opening costs.

$$(TLND) \quad \min \sum_{ij \in A} (c^1_{ij} x^1_{ij} + c^2_{ij} x^2_{ij})$$

$$x^1(\delta^-(W)) \geq 1 \qquad \forall W \subseteq V \setminus \{r\},\ W \cap P \neq \emptyset \tag{x1}$$

$$(x^1 + x^2)(\delta^-(W)) \geq 1 \qquad \forall W \subseteq V \setminus \{r\},\ W \cap S \neq \emptyset \tag{x12}$$

$$(x^1 + x^2)(\delta^-(i)) \leq 1 \qquad \forall i \in V \tag{1}$$

$$x^1_{ij}, x^2_{ij} \in \{0, 1\} \quad \forall ij \in A \tag{2}$$

The *primary connectivity constraints* (x1) ensure that for every primary node i, there is a path between r and i containing only primary arcs. The *secondary connectivity constraints* (x12) ensure that every secondary node is connected to the root by a path containing primary and/or secondary arcs. The in-degree constraints (1) ensure that the overall solution is a subtree and they are redundant if the edge costs are non-negative.

This gives a valid model for the TLND. In [2, 3] a directed MIP formulation based on network flows has been presented. It is easy to show that the set of feasible solutions of the LP-relaxation of the *TLND* model is the projection onto the space of (x^1, x^2) variables of this flow model. This result follows immediately from the max-flow min-cut theorem. Thus, the two models produce the same linear programming bound.

2.2 Modeling Facility Opening Costs

At each node in which a change of technology takes place, expensive facilities (e.g., multiplexors, splitters) need to be installed. In order to model these facility opening costs, we will use variables z_i:

$$z_i = \begin{cases} 1, & \text{if a facility is installed in node } i \\ 0, & \text{otherwise} \end{cases} \qquad \forall i \in V$$

For a set $W \subseteq V$, we will write $z(W) = \sum_{i \in W} z_i$.

2.2.1 Basic Coupling Constraints

To ensure that a facility is open, whenever a change of technology takes place, we request that every secondary arc $jk \in A$ used in a solution is either preceded by another secondary arc entering node j, or there is an open facility at node j. These constraints are an adaptation of degree-inequalities proposed by Khoury et al. [14] for the Steiner tree problem. Our problem can then be modeled as follows:

$$(TLNDF) \quad \min \sum_{ij \in A} (c^1_{ij} x^1_{ij} + c^2_{ij} x^2_{ij}) + \sum_{i \in V} z_i f_i$$

$$z_j + \sum_{ij \in A, i \neq k} x^2_{ij} \geq x^2_{jk} \qquad \forall jk \in A, k \notin P \tag{3}$$

$$z_i \in \{0,1\} \quad \forall i \in V \tag{4}$$

$$(\text{x1}), (\text{x12}), (1), (2)$$

In this model, the indegree constraints (1) are not redundant even if the arc- and facility costs are non-negative. These constraints namely prevent building of secondary cycles that would satisfy (3) without opening a facility at position j. Together with connectivity constraints (x12), the *basic coupling constraints* (3) guarantee that if a facility is installed at node j, then j is the root of a secondary subtree. This model does not prevent from opening facilities along a secondary path, but this will never be the case in an optimal solution.

2.2.2 Generalized $x^2 - z$ Coupling Constraints

One can generalize the coupling constraints (3) in the following way: Let k be any secondary customer or Steiner node and $W \subseteq V \setminus \{k\}$. Then, the *generalized* $x^2 - z$ *coupling constraints* can be written as follows:

$$z(W) + x^2(W^c_k, W) \geq x^2(W, \{k\}) \quad \forall k \in V \setminus (P \cup \{r\}),$$
$$W \subseteq V \setminus \{k\}, W_k = W \cup \{k\}. \tag{5}$$

Note that if $W = \{j\}$, for $j \neq k$, we obtain (3). In [10], the authors consider a similar generalization technique of degree-constraints by Khoury et al. [14] in the context of the two-level minimum spanning tree problem with weighted hop constraints. The formulation obtained by replacing constraints (3) by (5) is denoted by $TLNDF^+$.

The convex hull of feasible LP-solutions of $TLNDF^+$ is (for some instances even strictly) contained in the polytope defined by the LP-relaxation of the *TLNDF* model.

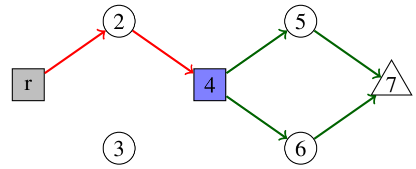

Fig. 1 Example for Lemma 1. All primary arc costs are 1, secondary arc and facility costs are 1/2.

Lemma 1. *Let $\mathscr{P}_{TLNDF^+}$ and $\mathscr{P}_{TLNDF}$ denote the polytopes associated with LP-relaxations of models TLNDF$^+$ and TLNDF, respectively. Then, $\mathscr{P}_{TLNDF^+} \subseteq \mathscr{P}_{TLNDF}$ and there exist instances for which the strict inequality holds.*

Proof. Constraints (3) are contained in the set (5): (3) for $jk \in A, k \notin P$ is derived from (5) for $W = \{j\}$. Figure 1 shows an example where the strict inequality holds: Consider the LP-optimal solution for *TLNDF* in which $x^1_{r2} = x^1_{24} = 1$ and $x^2_{45} = x^2_{57} = x^2_{46} = x^2_{67} = z_4 = 0.5$. $\upsilon_{LP}(TLNDF) = 3.25$ but constraint (5) is violated for $W = \{4,5,6\}$ and $k = 7$, so $\upsilon_{LP}(TLNDF^+) = 3.5 > \upsilon_{LP}(TLNDF)$. □

Constraints (5) can be rewritten in several equivalent ways which permit an easier comparison with other inequalities. In fact, by adding $x^2(W^c,k)$ to both sides, we can rewrite (5) as follows:

$$z(W) + x^2(\delta^-(W_k)) \geq x^2(\delta^-(k)) \quad \forall k \notin P, \forall W \subseteq V \setminus \{k\}, W_k = W \cup \{k\}. \quad \text{(x2-z)}$$

2.2.3 Generalized $x^1 - z$ Coupling Constraints

We have shown how to relate variables z and x_2. We show next how to relate variables z and x_1: For a given $k \in S$ and $W = V \setminus \{k\}$ we can rewrite inequalities (x2-z) as $z(V \setminus \{k\}) \geq x^2(\delta^-(k))$. By using the in-degree constraint (1), we obtain:

$$z(V \setminus \{k\}) + x^1(\delta^-(k)) \geq 1 \quad \forall k \in S$$

The latter constraints can be generalized for subsets $W \cap S \neq \emptyset$ in the following way:

$$z(W^c) + x^1(\delta^-(W)) \geq 1 \quad \forall W \subseteq V \setminus \{r\}, W \cap S \neq \emptyset \quad \text{(x1-z)}$$

These new inequalities describe the fact that for any subset W containing a secondary node, either there is a primary path between a node from W and r, or there is an open facility in the complementary set W^c.

Observation 1 *The set of inequalities* (x1-z) *cannot replace the coupling constraints* (3) *in the model TLNDF, i.e.* (x1-z) *are not sufficient for modeling the TLND problem with facility nodes. However,* (x1-z) *can be used to strengthen the model TLNDF$^+$.*

We denote the model *TLNDF*$^+$ extended by (x1-z) as $TLNDF^+_{x1-z}$.

Next, we will show that connectivity constraints (x1), (x12) and both groups of generalized coupling constraints are special cases of a more general group of constraints. These can be derived if we model the problem in a new graph obtained by node-splitting as described below.

3 The Node-Splitting Model

We can model the TLNDF problem as the Steiner arborescence problem in a slightly modified graph $G_{NS} = (V_{NS}, A_{NS})$ with the root r' and the set of terminals R_{NS}, as follows:

$$
\begin{aligned}
V_{NS} &:= V' \cup V'' \cup S \text{ where} & A_{NS} &:= A' \cup A'' \cup A_z \cup A_S \text{ where}\\
V' &:= \{i' \mid i \in V\}, & A' &:= \{i'j' \mid ij \in A\},\\
V'' &:= \{i'' \mid i \in V\}, & A'' &:= \{i''j'' \mid ij \in A\},\\
&S \text{ is the set of secondary nodes;} & A_z &:= \{i'i'' \mid i \in V\},\\
R_{NS} &:= P' \cup S \text{ where} & A_S &:= \{i'i \mid i' \in V', i \in S\}\\
P' &= \{i' \mid i \in P\}; & &\quad \cup \{i''i \mid i'' \in V'', i \in S\}.
\end{aligned}
$$

The graph G_{NS} is composed of several components: i) a subgraph $G' = (V', A')$ which corresponds to the primary network (it contains nodes and arcs that may be included in the primary subtree); ii) a subgraph $G'' = (V'', A'')$ that corresponds to the secondary network (it contains nodes and arcs that may be contained in the secondary subtrees); iii) arcs linking nodes in G' to the corresponding copy in G'' and representing potential facilities and iv) another copy of the secondary nodes with arcs incoming from their representatives in graphs G' and G'' (see Figure 2). Arc costs C_{ij}, $ij \in A_{NS}$ are assigned accordingly to the arcs in G', G''. The arcs linking the two subgraphs are assigned costs $C_{i'i''} := f_i$, for all $i \in V$. To the arcs of the set A_S costs of zero are assigned.

If, for a primary node $i \in P$, its copy $i'' \in V''$ belongs to the optimal solution, there will be no ingoing arcs into i'' (with the only exception of $i'i''$). Therefore, we can reduce the size of G_{NS}, by removing all ingoing arcs of primary nodes in V''. This corresponds to setting $x^2_{ij} := 0$ for all $ij \in A$ such that $j \in P$, as already described in Section 2. Observe that we need a third copy of secondary nodes in G_{NS}, namely the set S, since it is not clear for secondary nodes whether they will be connected within the primary or the secondary subtree.

To provide an ILP model, we assign binary variables X_{ij} to all arcs $ij \in A_{NS}$. Denote by $X(\delta^-(\tilde{W}))$ the sum of X variables that correspond to the arcs in the directed cut $(\tilde{W}^c, \tilde{W})$ in G_{NS}. Based on the classical cut set model for Steiner trees (cf. [4]) we derive the following ILP formulation:

$$(SA) \quad \min \sum_{ij \in A_{NS}} C_{ij} X_{ij} \tag{6}$$

$$\text{s.t.} \quad X(\delta^-(\tilde{W})) \geq 1 \qquad \forall \tilde{W} \subseteq V_{NS} \setminus \{r'\}, \tilde{W} \cap R_{NS} \neq \emptyset \tag{7}$$

$$\sum_{ij \in A} (X_{i'j'} + X_{i''j''}) \leq 1 \qquad \forall j \in V \tag{8}$$

$$X_{ij} \in \{0,1\} \quad \forall ij \in A_{NS} \tag{9}$$

Constraints (7) are the classical connectivity cuts, inequalities (8) state that of all ingoing edges of both copies of a node in G at most one is allowed to be open.

Lemma 2. *The TLNDF problem can be modeled as the Steiner arborescence problem with additional degree constraints on some node pairs on the graph G_{NS} with the root r' and terminal set R_{NS}.*

Proof. We map each binary solution of formulation *SA* into the variable space of *TLNDF* as follows: $X_{i'j'} \to x^1_{ij}$, $X_{i''j''} \to x^2_{ij}$ and $X_{i'i''} \to z_i$. Let now $\mathbf{X}$ be an LP optimal solution for *SA*. The mapping of $\mathbf{X}$ then satisfies all constraints of *TLNDF*: Connectivity cuts (7) imply (x1) and (x12), together with degree constraints (8) they ensure (1). Finally, constraints (3) are also satisfied since we have:

$$z_j + \sum_{ij \in A, i \neq k} x^2_{ij} = X_{j'j''} + \sum_{i \neq k} X_{i''j''} \geq X_{j''k''} = x^2_{jk}.$$

The last inequality is implied by (7) and (8). $\square$

Let $Proj_{x^1,x^2,z}(\mathscr{P}_{SA})$ denote the projection of the polytope obtained as the convex hull of the LP-solutions of the *SA* formulation into the space of (x^1,x^2,z) variables. In this projection, we set $x^1_{ij} := X_{i'j'}$, $x^2_{ij} := X_{i''j''}$ for all $ij \in A$ and $z_i := X_{i'i''}$ for all $i \in V$.

Theorem 1. *The SA formulation is at least as strong as the previously defined formulation $TLNDF^+_{x1-z}$, i.e., $Proj_{x^1,x^2,z}(\mathscr{P}_{SA}) \subseteq \mathscr{P}_{TLNDF^+_{x1-z}}$.*

To prove this result, we need to analyze the cut set inequalities defined in the *SA* model and their projection onto the original graph G.

Lemma 3. *Cut set inequalities* (7) *such that $\delta^-(\tilde{W}) \cap A_S \neq \emptyset$ are redundant in the model SA.*

Proof. Consider a cut set $\tilde{W} \subseteq V_{NS} \setminus \{r'\}$, $\tilde{W} \cap S \neq \emptyset$, such that $\delta^-(\tilde{W}) \cap A_S \neq \emptyset$. We will show that in that case, $X(\delta^-(\tilde{W})) \geq 1$ is dominated by another cut set inequality $X(\delta^-(\tilde{U})) \geq 1$ where $\tilde{U}$ is defined as stated below. We need to distinguish the following two cases:

i) If for all $i \in S \cap \tilde{W}$, $i'i \in \delta^-(\tilde{W})$ and $i''i \notin \delta^-(\tilde{W})$, a dominating cut is given for $\tilde{U} = \tilde{W} \cup \bigcup_{i \in \tilde{W}} \{i'\}$.
ii) For all other $\tilde{W}$ the dominating cut is obtained by removing nodes $i \in S$ from $\tilde{W}$ if $i'' \in \tilde{W}$ and $i' \notin \tilde{W}$ and adding nodes i' and i'' to $\tilde{W}$ for $i \in S \cap \tilde{W}$ such that $i', i'' \notin \tilde{W}$. $\square$

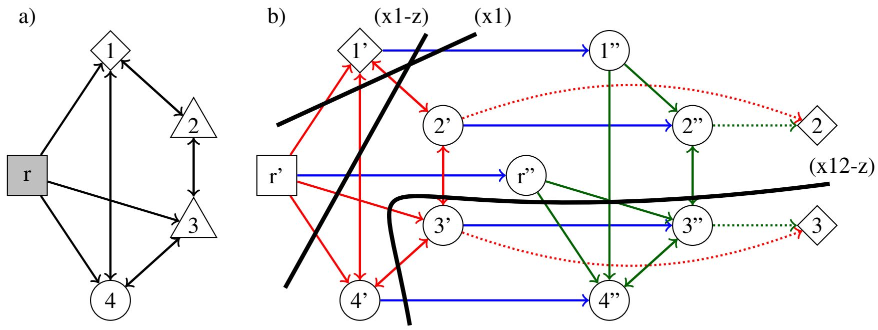

Fig. 2 a) Instance of *TLNDF*; b) Transformed instance and illustration of cuts (x1) for $W = \{1\}$, (x1-z) for $W = \{2,3,4\}$ and cuts (x12-z) for $W' = \{3\}$ and $W'' = \{3,4\}$

We will refer to the cut set inequalities such that $\delta^-(\tilde{W}) \cap A_S = \emptyset$ as the *non-dominated cut set inequalities*.

Generalized Cut Set Constraints

We will now define the generalized cut set constraints for the TLNDF that are obtained by projecting the non-dominated inequalities among the ones in (7) into the space of (x_1, x_2, z). For an arbitrary cut set $\tilde{W} \subset V_{NS} \setminus \{r'\}$, $\tilde{W} \cap R_{NS} \neq \emptyset$, let us denote the projected subsets of the original graph G as follows:

$$W' = \{i \in V \mid i' \in \tilde{W}\} \text{ and } W'' = \{i \in V \mid i'' \in \tilde{W}\}$$

Then, the projected cut set inequalities (7), that we will refer to as *generalized cut set constraints*, can be written as:

$$x^1(\delta^-(W')) + x^2(\delta^-(W'')) + z(W'' \setminus W') \geq 1 \quad r \notin W', W' \cap W'' \cap S \neq \emptyset \text{ or } W' \cap P \neq \emptyset. \qquad \text{(x12-z)}$$

Observe that all the previously studied inequalities are special cases of this constraint (see Figure 2):

i) If $W'' = \emptyset$, we obtain the primary connectivity constraints (x1).
ii) If $W' = W''$, we obtain the secondary connectivity cuts (x12).
iii) If $W'' = V$, we obtain the generalized coupling constraints (x1-z).
iv) For a given $k \in S$, and a subset $W \subseteq V \setminus \{r, k\}$, the generalized (x2-z) constraint corresponds to (x12-z) for $W' = \{k\}$, $k \in S$, and $W'' = W \cup \{k\}$.

This implicitly proves Theorem 1, i.e., the projection of every feasible LP-solution of the formulation *SA* is also feasible to $TLNDF^+_{x1-z}$.
We conclude this section by noting that even more general classes of inequalities can be obtained by considering non-trivial cases in which $W' \cap W'' \neq \emptyset, W', W''$.

Lemma 4. *The generalized connectivity constraints* (x12-z) *can be separated in polynomial time.*

To separate the constraints (x12-z), one needs to apply the max-flow algorithm on the splitted graph G_{NS} as described in the next section.

4 Computational Study

In this section we report details of the implementation of our Branch-and-Cut algorithm, how we derived the set of benchmark instances and the computational results we obtained.

4.1 The Branch-and-Cut Algorithm

To implement our models, we used the Gurobi [1] Branch-and-Cut framework, version 3.0.2. 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.

4.1.1 Initialization

As the Gurobi MIP solver requires a compact model for initialization, we used the following Miller-Tucker-Zemlin connectivity constraints (10)-(13) and trivial degree-constraints (14):

$$u_r = 1 \tag{10}$$

$$|V|(x^1_{ij} + x^2_{ij}) + (|V| - 2)(x^1_{ji} + x^2_{ji}) + u_i - u_j \leq |V| - 1 \quad ij \in A, j \notin P \tag{11}$$

$$|V|(x^1_{ij}) + (|V| - 2)(x^1_{ji}) + u_i - u_j \leq |V| - 1 \quad ij \in A, j \in P \tag{12}$$

$$\sum_{ij \in A : i \neq k} x^1_{ij} \geq x^1_{jk} \quad j \neq r \tag{13}$$

$$\sum_{ij \in A} x^1_{ij} \geq z_j \quad j \in F \setminus \{r\} \tag{14}$$

In addition, our model comprises in-degree constraints (1) and coupling constraints (3).

4.1.2 Separation

Separating (x1) **and** (x12) **Cuts:**

We separate violated cut set inequalities (x1),(x12) and (x12-z) in every node of the the Branch-and-Bound tree (BnB). To obtain inequalities (x1), we solve a maximum flow problem on the graph $G = (V, A)$. The capacities on each arc are set to the value of the x^1-variable for the respective arc in the current fractional solution. Cut set inequalities (x12) are obtained in a similar fashion. The capacities are equal to the sum of variables x^1 and x^2 for each arc.

Separating (x12-z) **Cuts:**

To obtain violated constraints of the largest and strongest group (x12-z), we solve the maximum flow problem on the *splitted graph* G_{NS}. The weights for arcs in A',

A'' and A_z are set to the value of the corresponding variable in the current fractional solution. For arcs in A_S, the weight is set to 1, as cuts containing these arcs are dominated by others (cf. Lemma 3).

General Settings:

To improve the computational efficiency of our separation, we search for nested and minimum cardinality cuts. To do so, all capacities in the respective graph are increased by some $\varepsilon > 0$. Thus, every detected violated cut contains the least possible number of arcs. The LP is resolved after adding at most 50 violated inequalities of type (x1), (x12) or (x12-z). Finally, we randomly permute the order in which customers are chosen to find violated cuts. To ensure comparability, we fix the seed value for the computations reported in Section 4.3.

4.1.3 Primal Heuristic

We use a primal heuristic (PH) to find incumbent solutions. The PH is entirely carried out on the graph G_{NS}. It consists of the following steps:

1. **Construct primary subtree:** Primary customers are connected to the root node by the arcs in the shortest path to the copy of that customer in V'. For all nodes taken into the primary subtree, ingoing secondary arcs are removed.
2. **Construct secondary subtree:**
    a. Using zero costs on all arcs in the primary subtree, the shortest paths $P(i')$ and $P(i'')$ from the root to $i' \in V'$ and $i'' \in V''$ are calculated for all $i \in S$. Let $H'(i) = |P(i'') \cap A'|$ and $H''(i) = |P(i'') \cap A''|$.
    b. Let $Q = S$. For all $i \in Q$ such that $H''(i) = 0$ add $P(i')$ and remove i from Q.
    c. Sort Q according to (H', H'') in decreasing order and repeat until $Q = \emptyset$: Add $P(i'')$ and remove i from Q.
3. **Pruning of primary subtree:** Superfluous *leaves* are iteratively removed from the primary subgraph: Secondary customers, that are part of the primary and a secondary subtree in which no facility is installed as well as Steiner nodes are removed.
4. **Pruning and repairing of secondary subtree:** Superfluous nodes are removed from the secondary subgraph and infeasible parts of the solution repaired: Steiner node leaves and secondary customer leaves in V'' are iteratively removed, if their respective copy in the primary subtree is used. For each secondary customer with in- and out-going arcs in both A' and A'', we remove the ingoing arcs in A'' and open a facility at this node.

We use the information from the current best LP solution to adjust the weights for calculating the shortest paths. We set the weight w for an arc in G_{NS} to $(1 - v)c$ where v is the corresponding variable and c is the initial cost.

4.2 Instances

For our computational study we transform instances of the Steiner tree problem (STP) using the following procedure:

- First, 30% of STP terminals are chosen as primary customers, the remaining 70% are defined as secondary customers. The primary customer with the lowest index is set as root node.
- The Steiner nodes in the STP instance are Steiner nodes in the TLNDF instance.
- As potential facility nodes we chose the root node, primary and secondary customers.
- Primary edge costs equal edge costs of the STP instance. For each secondary edge e, the cost c_e^2 is defined as qc_e^1, where q is uniformly randomly chosen from $[0.25, 0.5]$.
- Facility opening costs are uniform and equal 0.5 times the average primary edge costs.

The parameters for generating instances have been carefully chosen so that trivial solutions (e.g., optimal solutions that do not contain secondary subtrees) are avoided. The sets B, C, D and E of the Steinlib library [15] have been used in our computational study.

4.3 Results

We compared the computational performance of three different settings (two of which using cuts derived from the splitted graph):

i) Model *TLNDF*, in which the basic coupling constraints (3) are inserted at once and the (x1) and (x12) cuts are separated within the branch-and-bound (BnB) tree.
ii) In the second setting, after all violated (x1) cuts have been detected, (x12) are separated. Finally, after no more violated (x1) and (x12) cuts can be found, generalized connectivity constraints (x12-z) are separated.
iii) In the third setup, we refrained from separating inequalities (x12), i.e., after no more violated (x1) cuts can be found, generalized connectivity constraints (x12-z) are separated.

In a preliminary test we tested our three approaches on the instances of set B. The maximum runtime was 5.28, 8.11 and 4.47 seconds respectively. As a consequence we only give detailed results for the larger sets C,D and E.

In Table 1 we show the key figures of our computational study. The first column indicates the group of instances, in columns 2, 3 and 4 we state the (maximum) number of nodes, edges and terminals (i.e. union of primary and secondary customers) of the largest instance of each group, respectively. In the third segment of the upper part we show the number of instances in this group solved to optimality within 1000 seconds of running time. The last segment shows the average running times for the subset of instances that was solved to optimality by all three

Table 1 Computational comparison of three different branch-and-cut settings.

	$\|V\|$	$\|E\| \leq$	$\|T\| \leq$	#OPT i)	#OPT ii)	#OPT iii)	$t\ [s]$ i)	$t\ [s]$ ii)	$t\ [s]$ iii)
`c01-10`	500	1000	250	**10**	**10**	**10**	**46.52**	69.54	73.48
`c11-20`	500	12500	250	1	3	**6**	55.84	56.53	**20.86**
`d01-10`	1000	2000	500	**10**	9	**10**	**180.61**	253.66	271.11
`d11-20`	1000	25000	500	2	2	**4**	172.38	110.89	**39.19**
`e01-10`	2500	5000	1250	**6**	5	5	191.71	81.68	**38.60**
`e11-20`	2500	62500	1250	2	**3**	**3**	178.42	179.29	**80.16**

	$\|V\|$	$\|E\| \leq$	$\|T\| \leq$	avg $gap[\%]$ i)	avg $gap[\%]$ ii)	avg $gap[\%]$ iii)	iii) #OPT found in $\leq$ 1h	iii) #OPT found in $\leq$ 2h	iii) #OPT found in $\leq$ 24h
`c01-10`	500	1000	250	**0.00%**	**0.00%**	**0.00%**	10	10	10
`c11-20`	500	12500	250	5.93%	3.88%	**1.05%**	7	9	10
`d01-10`	1000	2000	500	**0.00%**	**0.00%**	**0.00%**	10	10	10
`d11-20`	1000	25000	500	4.19%	4.17%	**0.75%**	6	6	9
`e01-10`	2500	5000	1250	**0.11%**	0.15%	0.20%	6	8	10
`e11-20`	2500	62500	1250	5.78%	**5.34%**	5.35%	3	5	5

Table 2 Average running times vs. graph density and vs. number of terminals, respectively. Values are normalized according to the first column in each segment.

	$\|E\|/\|V\|$ 1.25	2	5	25	$\|T\|$ 5	10	$\frac{1}{6}\|V\|$	$\frac{1}{4}\|V\|$	$\frac{1}{2}\|V\|$
`c01-20`	1.0	2.5	18.3	206.7	1.0	1.0	12.2	23.0	100.4
`d01-20`	1.0	1.3	6.1	158.8	1.0	1.6	43.3	80.4	143.3
`e01-20`	1.0	345.7	699.9	1195.6	1.0	2.1	296.7	503.1	256.1

approaches within 1000 seconds. In segment three of the lower part we state the average gaps of each instance group after 1000 seconds of running time. In segment four we report the number of optimal solutions found by approach iii) within 1h, 2h and 24h, respectively. From the number of instances solved to optimality and the average running times one can see, that for sparse graphs (`.1-10`) the approach based only on connectivity cuts (x1) and (x12) is competitive to the generalized cut set constraints. For denser graphs (`.11-20`) the two new approaches (namely ii) and iii) involving (x12-z) cuts) perform much better: For instances with few arcs, there is little difference in the LP bounds provided by the models with and without constraints (x12-z). Constraints (x1) and (x12) are cheaper to separate, but as the instances grow larger and denser, the advantage of better LP bounds provided by cuts (x12-z) outweighs this.

Table 2 illustrates how the running time performance of the approach iii) depends on the graph density (the second segment) and on the number of terminals (the third segment). Instances C, D, and E have been divided into groups according to their density ($|E|/|V|$) and the number of terminals ($|T|$), respectively. We observe that the average running times increase exponentially with the density and the number of terminals.

5 Conclusions

For the TLNDF we have introduced several new families of valid inequalities combining network design and facility location variables. The so-called generalized cut inequalities (x12-z) are the strongest among those inequalities and can be derived from a cut-set model for Steiner arborescence applied on a splitted graph. We have seen that the separation of (x12-z) cuts is not only computationally tractable, but it also outperforms the standard compact approach of modeling facility nodes. Finally, we have tested our approach on a set of 78 benchmark instances with up to 2500 nodes and 62500 edges. We have been able to solve 60 (66) instances to provable optimality in less than 1h (2h). From the remaining 12 instances 6 were solved optimally after 1 day and for 6 we obtained solutions less than 2% from optimum.

6 Acknowledgements

The authors would like to thank the referees for their comments and suggestions. Luis Gouveia was funded by Centro de Investigação Operacional [MATH-LVT-Lisboa-152]. Ivana Ljubić was partially supported by the Austrian Science Fund (FWF): [T334].

References

[1] Gurobi Optimization. URL `http://www.gurobi.com/`.
[2] A. Balakrishnan, T. L. Magnanti, and P. Mirchandani. A dual-based algorithm for multi-level network design. *Management Science*, 40(5):567–581, 1994.
[3] A. Balakrishnan, T. L. Magnanti, and P. Mirchandani. Modeling and heuristic worst-case performance analysis of the two-level network design problem. *Management Science*, 40(7):846–867, 1994.
[4] S. Chopra, E. R. Gorres, and M. R. Rao. Solving the Steiner Tree Problem on a Graph Using Branch and Cut. *INFORMS Journal on Computing*, 4(3): 320–335, 1992.
[5] A. M. Costa, P. M. França, and C. Lyra Filho. Two-level network design with intermediate facilities: An application to electrical distribution systems. *Omega*, 39(1):3–13, 2011.
[6] J. R. Current, C. S. ReVelle, and J. L. Cohon. The hierarchical network design problem. *European Journal of Operational Research*, 27(1):57 – 66, 1986.
[7] C. Duin and T. Volgenant. The multi-weighted Steiner tree problem. *Annals of Operations Research*, 33(6):451–469, 1991.

[8] S. Gollowitzer and I. Ljubić. MIP models for connected facility location: A theoretical and computational study. *Computers & Operations Research*, 38 (2):435–449, 2011.

[9] E. Gourdin, M. Labbé, and H. Yaman. Telecommunication and location - a survey. In Z. Drezner and H. Hamacher, editors, *Facility Location: Applications and Theory*. Springer, 2001.

[10] L. Gouveia and E. Janssen. Designing reliable tree networks with two cable technologies. *European Journal of Operational Research*, 105(3):552 – 568, 1998.

[11] L. Gouveia and J. Telhada. Distance-constrained hierarchical networks. In *Proceedings of International Network Optimization Conference, INOC 2005*, pages B2.416–B2.421, 2005.

[12] L. Gouveia and J. Telhada. The multi-weighted Steiner tree problem: A reformulation by intersection. *Computers & Operations Research*, 35(11):3599–3611, 2008.

[13] A. D. Jongh, M. Gendreau, and M. Labbé. Finding disjoint routes in telecommunications networks with two technologies. *Operations Research*, 47(1): 81–92, 1999.

[14] B. N. Khoury, P. M. Pardalos, and D.-Z. Du. A test problem generator for the Steiner problem in graphs. *ACM Trans. Math. Softw.*, 19(4):509–522, 1993.

[15] T. Koch, A. Martin, and S. Voß. SteinLib: An updated library on steiner tree problems in graphs. Technical report, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2000. URL `http://elib.zib.de/steinlib`.

[16] I. Ljubić and S. Gollowitzer. Modeling the hop constrained connected facility location problem on layered graphs. In *Electronic Notes in Discrete Mathematics*, volume 36, pages 207–214, 2010. Proceedings of the International Symposium on Combinatorial Optimization (ISCO).

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