TECHNISCHE UNIVERSITÄT WIEN
Institut für Computergraphik und Algorithmen

The Two-Level Diameter Constrained Spanning Tree Problem

Luis Gouveia, Markus Leitner, Ivana Ljubić

Forschungsbericht / Technical Report

TR–186–1–12–02

October 18, 2012

Favoritenstraße 9-11 / E186, A-1040 Wien, Austria
Tel. +43 (1) 58801-18601, Fax +43 (1) 58801-18699
www.cg.tuwien.ac.at

The Two-Level Diameter Constrained Spanning Tree Problem

Luis Gouveia · Markus Leitner · Ivana Ljubić

Abstract In this article, we introduce the Two-Level Diameter Constrained Spanning Tree Problem (2-DMSTP) which generalizes the classical DMSTP by considering two sets of nodes with different latency requirements. We first observe that any feasible solution to the 2-DMSTP can be viewed as a DMST that contains a diameter constrained Steiner tree. This observation allows us to prove graph theoretical properties related to the centers of each tree which are then exploited to develop mixed integer programming formulations, strengthening valid inequalities, and symmetry breaking constraints. In particular, we propose a novel modeling approach based on a three-dimensional layered graph. In an extensive computational study we show that a branch-and-cut based on the latter model is highly effective in practice.

Keywords Networks/graphs: tree algorithms · Integer programming: formulations · Layered graphs

Mathematics Subject Classification (2000) 90C11 Mixed integer programming · 90C27 Combinatorial optimization · 90C57 Polyhedral combinatorics, branch-and-bound, branch-and-cut

L. Gouveia is supported by National Funding from FCT - Fundação para a Ciência e Tecnologia, under the project: PEst-OE/MAT/UI0152. M. Leitner is supported by the Austrian Science Fund (FWF) under grant I892-N23. I. Ljubić is supported by the APART Fellowship of the Austrian Academy of Sciences. These supports are greatly acknowledged.

L. Gouveia
Faculdade de Ciênçias, Universidade de Lisboa, DEIO, CIO Bloco C/2, Campo Grande, 1749-016 Lisbon, Portugal. E-mail: legouveia@fc.ul.pt

M. Leitner
Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstr. 9-11, 1040 Vienna, Austria. E-mail: leitner@ads.tuwien.ac.at

I. Ljubić
Department of Statistics and Operations Research, University of Vienna, Brünnerstr. 72, 1210 Vienna, Austria. E-mail: ivana.ljubic@univie.ac.at

1 Introduction and Motivation

Given a graph $G = (V, E)$ with edge costs $c_e \geq 0$, $e \in E$, the diameter constrained spanning tree problem (DMSTP) is to find a minimum spanning tree such that the distance (in number of edges) between every pair of nodes is at most some parameter $D_m \in \mathbb{N}$, $D_m > 1$. The DMSTP has a wide range of applications: in telecommunications, data compression, or parallel computing (see, e.g., Deo and Abdalla [3], Noronha et al. [14]). In telecommunication networks, for example, when multicasting is employed, the network latency between a pair of users is directly proportional to the length of the routing path. In a tree-multicasting, the maximum pairwise latency equals the diameter of that tree. The optimization goal consist of finding a minimum-cost spanning tree such that the maximum latency is restricted by some parameter D_m (see, e.g., Vik et al. [16]). In a more realistic scenario, media types transmitted over a network may range from text to video streaming, with different latency requirements being imposed for transmitting text, voice over IP, or video streaming in multi-player online games. Throughout this paper we will assume that there are two disjoint groups of nodes - those with more stringent latency requirements (e.g., video streaming users) and the remaining ones whose latency requirements are less demanding. At first glance, the assumption that there are two subsets with different latency requirements might look too simplistic, but as we shall see later, it is already rich enough to impose different and interesting research challenges.

More formally, we introduce a new problem which is a generalization of the DMSTP to which we will refer to as the *Two-Level Diameter Constrained Spanning Tree Problem* (2-DMSTP): Given a graph $G = (V, E)$ (where $|V| = n$) with edge costs $c_e \geq 0$, $e \in E$, and with the set of nodes V partitioned into two subsets: P, the subset of *primary* and more important nodes, and S, the subset of *secondary* and less important nodes. Two limits on the maximum length of communication paths are imposed: The maximum distance between nodes in P is allowed to be at most D', and the maximum distance between nodes in S or between a node in P and a node in S is allowed to be at most D ($D > D' > 1$). The optimization goal consists of finding a minimum-cost spanning tree that satisfies these length restrictions.

Notice that for $D' = 2$ and $D = 3$ the 2-DMSTP can be solved in polynomial time by adapting an enumeration approach for the DMSTP with $D_m = 3$ (see, e.g., Gouveia et al. [7]). We fix a central edge $\{i, j\} \in E$ and attach each node in S to the closest node in the set $\{i, j\}$. All the remaining nodes in P need to be attached either to node i or to node j, whichever is cheaper. The obtained tree is feasible for the 2-DMSTP, and to find an optimal solution, we repeat this procedure for all edges $\{i, j\} \in E$ and choose the cheapest solution.

On the other hand, when $|P| = 1$ or $|S| = 1$ (but not both) we obtain the *Hop Constrained Minimum Spanning Tree Problem* (HMSTP) which is to find a minimum cost spanning tree such that the distance from a given node to any other node, is at most H_m. These particular cases indicate that the 2-DMSTP is also NP-hard in general. In fact, we can show the following result:

Lemma 1 *The 2-DMSTP is NP-hard for $D \geq 4$, and any $2 \leq D' < D$.*

Proof The proof follows by reduction from the HMSTP which is NP-hard for $H_m \geq 2$ (see, e.g., [13]). Let us consider an instance G_H of the HMSTP with hop limit $H_m = 2$ and root 1. Attach now to the root a star built out of a set of nodes $p \in P$, with the center p_0 of that star directly connected to 1. The obtained graph is now an instance of the 2-DMSTP with $D = 4$ and $D' = 2$. Each optimal solution of this graph corresponds to an optimal solution of the HMSTP for $H_m = 2$ which concludes the proof. □

Our Contribution In this paper we first observe that the 2-DMSTP can be viewed as a diameter constrained spanning tree with diameter D that contains a diameter constrained *Steiner tree* with *terminal set* P and diameter D'. This permits us to use *center properties* for each of these two subproblems and to develop mixed integer programming (MIP) models that are more efficient than the traditional formulations based on the pairwise distance constraints.

The results presented in this paper are threefold: (a) Graph theoretical results: In the first part of the paper we study feasible 2-DMSTP solutions from a graph theoretical perspective. We obtain upper bounds on the distance between the two centers and we provide necessary and sufficient conditions for those centers to be at minimum distance. (b) MIP models: We propose two new models, both relying on the concept of layered graphs. Layered graphs have been shown (see Gouveia et al. [8]) to provide the strongest MIP models for the DMSTP, both, from theoretical and computational perspective. Our first model can be viewed as an intersection of two layered graphs that independently model the Steiner tree and the spanning tree. On the other hand, the new graph theoretical results regarding the relative location of the two centers permit us to embed these properties in a layered graph construction. To do this, we propose a novel three-dimensional layered graph approach that also incorporates distance constraints w.r.t. primary nodes in its structure. To break symmetries, we use theoretical results regarding the minimum distance between the centers. (c) Computational results: Branch-and-cut algorithms are developed for the two proposed layered graph approaches. They are computationally tested on a set of benchmark instances for the DMSTP. They show that the novel three-dimensional layered graph model performs highly effective in practice.

Outline of the Paper In the remainder of this section we give short literature overview and provide a generic MIP model for the 2-DMSTP. The graph theoretical results are given in Section 2. Section 3 deals with the MIP modeling issues for the 2-DMSTP, where the two layered graph approaches, along with the sets of valid inequalities are proposed. The results of our computational study are provided in Section 4.

1.1 Related Literature

Since 2-DMSTP was not studied before, in this section we summarize the previous work on the MIP approaches to the DMSTP (see, e.g., Gruber [9] for further literature on the DMSTP). Several single-commodity flow models were proposed and tested in Achutan et al. [1]. Multi-commodity flow models with tighter linear programming (LP) relaxations were studied in Gouveia and Magnanti [4]. The authors use the idea of a *central node* or a *central edge* that serves as the source for the commodities. This approach allows for a reduction of the number of commodities by a factor of n, while preserving the tight LP bounds. Several other approaches for the DMSTP (see, e.g., Gruber and Raidl [10], Santos et al. [15]) have used the properties of tree centers as well. In Gouveia et al. [5] the authors introduced an approach that views the DMSTP with odd diameter as being composed of a directed spanning tree (from an artificial root node) together with two constrained paths, a shortest and a longest path, from the root node to any node in the tree. The authors proposed and tested an extended flow based model derived from this idea. In Gouveia et al. [6], an alternative modeling approach for odd diameters is proposed: the approach views the DMSTP as an intersection of two trees directed out of the end-nodes of the central edge. A constraint programming approach has been proposed by Noronha et al. [14] where the obtained computational results indicate that the approach cannot compete with MIP based approaches yet. Gruber and Raidl [11] applied a heuristic separation technique in a branch-and-cut algorithm applied to an MIP formulation based on *jump constraints*. The current state-of-the-art approach for the DMSTP has been proposed by Gouveia et al. [8] where the DMSTP is modeled as a Steiner tree problem on a layered graph. The authors showed that the layered graph approach outperforms all previous MIP based approaches both in theory and practice.

1.2 Generic MIP Model for the 2-DMSTP

As noted before, any feasible solution of the 2-DMSTP can be interpreted as a diameter constrained spanning tree with diameter at most D that contains a diameter constrained Steiner tree with terminal set P and diameter at most D'. Let x^1_e be binary variables indicating whether edge e is inside the primary Steiner tree and let x^2_e be binary variables indicating whether edge e is in the spanning tree. Then the problem can be modeled in a generic way as follows:

$$\min_{e \in E} c_e x^2_e$$

$$\{e : x^1_e = 1\} \quad \text{is a } \textit{Steiner tree} \text{ with diameter } D' \text{ and terminal set } P \tag{1}$$

$$\{e : x^2_e = 1\} \quad \text{is a } \textit{spanning tree} \text{ with diameter } D \tag{2}$$

$$x^1_e \leq x^2_e \quad \text{for all } e \in E \tag{3}$$

$$x^1, x^2 \in \{0,1\}^{|E|} \tag{4}$$

Thus, instead of using all-pairs-distance-constraint-based models, this generic formulation allows us to model each of the subproblems (1) and (2) independently and to reduce the number of distance constraints by exploiting the center properties of each of the two subproblems (cf. Section 3).

2 Graph Theoretical Properties of 2-DMSTP Trees

In this section we provide answers to the following questions: (a) What is the maximum distance between the two centers in any feasible solution? (b) What are necessary and sufficient conditions for this distance to be minimal? These properties are used later on in Section 3 to derive new layered graph MIP formulations for the 2-DMSTP.

Notation and Definitions. In the following, we assume that $T = (V, E_\mathrm{T})$, $E_\mathrm{T} \subseteq E$, is a spanning tree of G with diameter *at most* D that contains a Steiner tree $T' = (V'_\mathrm{T}, E'_\mathrm{T})$, $P \subseteq V'_\mathrm{T} \subseteq V$, $E'_\mathrm{T} \subseteq E_\mathrm{T}$, whose diameter is *at most* D'. We will denote a feasible 2-DMSTP solution as a pair (T, T'). Let $d_{uv} \in \mathbb{N}$ denote the length of the path between u and v in T, i.e., its number of edges, and let $\varepsilon_T(u) = \max_{v \in V} d_{uv}$ be the *eccentricity* of u, i.e., the maximum number of edges on the path between u and any other node within the tree T. Similarly, let $\varepsilon_{T'}(u)$ denote the *eccentricity* of u within T'. Thus, any feasible 2-DMSTP solution given by T and T' has to satisfy:

$$\max_{u \in V} \varepsilon_T(u) \leq D \text{ and } \max_{u \in V'_\mathrm{T}} \varepsilon_{T'}(u) \leq D'.$$

Whenever it is clear from context, we will write $\varepsilon(u)$ instead of $\varepsilon_T(u)$. Given an edge $e = \{i, j\}$, we can also define the *edge eccentricity* $\varepsilon(e)$ as follows $\varepsilon(e) = \min\{\varepsilon(i), \varepsilon(j)\}$. For a node u and an edge $e = \{i, j\}$ in T, let $d_{ue} = d_{eu} = \min\{d_{ui}, d_{uj}\}$ be the distance from u to the node of e that is closer to u. Similarly, the distance between two edges $e = \{i, j\}$ and $f = \{k, l\}$ of T, is given as $d_{ef} = \min_{u \in \{i,j\}, v \in \{k,l\}} d_{uv}$. Notice that for the latter, we obtain a distance of zero if the two edges are either adjacent or if $e = f$.

The following two properties play a central role in the graph theoretical results associated to spanning/Steiner trees with bounded diameter:

Central node Property: A tree T has diameter no more than an *even* integer D if and only if for some node p of T (the *central node*) $\varepsilon(p) \leq D/2$, i.e., the path to any other node of the tree from node p contains at most $D/2$ edges.

Central edge Property: A tree T has diameter no more than an *odd* integer D if and only if for some edge $e = \{p, q\}$ of T (the *central edge*) $\varepsilon(e) \leq \lfloor D/2 \rfloor$, i.e., the path to any other node of the tree from either node p or node q contains at most $\lfloor D/2 \rfloor$ edges.

We will use generic notation (c, c') to denote a pair of centers for (T, T'). Whenever D (D') is even, c (c') will denote a central node, otherwise it will denote a central edge. Furthermore, c' and c will also be called *primary* and *secondary center*, respectively.

Central Path: Given a feasible 2-DMSTP solution (T, T'), for each pair of centers (c, c') the unique path in T between c and c' including c and c' is called the *central path* of (T, T') with respect to (c, c'). Note that, due to this definition we distinguish between the *length of the central path*, i.e., its number of edges, and the *distance between the centers* which is given by $d_{cc'}$ and hence will be less than the length of the central path unless both c and c' are central nodes, i.e., unless both D and D' are even.

Finally, for nodes u, v and z belonging to the same path of a tree we use $u - v - z$ to state that node v is *between* nodes u and z on this path. Similarly, $u - e - v$ states that edge e is *between* nodes u and v and $u - e - e'$ denotes that edge e is between node u and edge e'. For the set of edges E we will define the set of arcs A by introducing two oppositely directed arcs for each edge. For a subset $W \subset V$, we use $\delta^-(W) = \{(i,j) \in A \mid i \notin W, j \in W\}$ and $\delta^+(W) = \{(i,j) \in A \mid i \in W, j \notin W\}$ to denote the ingoing and outgoing cutset, respectively.

2.1 Maximum Distance Between the Centers

Given a feasible 2-DMSTP solution (T, T'), for each of the two trees, T and T' there exists a central node or central edge. Notice that if the diameters are tight these centers are unique, but this does not need to be a case for an arbitrary feasible solution. The following proposition gives a tight upper bound on the distance between these two centers for an arbitrary feasible solution.

Proposition 1 *Given a feasible 2-DMSTP solution (T, T'), there exist centers of T and T' such that the distance between them is at most $\lfloor D/2 \rfloor - \lceil D'/2 \rceil$.*

In order to prove this proposition, we will use the lemmas stated below. We will provide the proofs for the case D' and D being even, and the same proofs can be easily adapted for the remaining cases.

Lemma 2a *If D' is even and $D = D' + 2m$, $m \in \mathbb{N}$, then there exists a central node p of T' such that $\varepsilon(p) \leq D/2 + m$.*

Proof Assume first that the diameter D' of T' is tight, in which case there exists a unique central node p in T'. To show that the result holds, assume the opposite, i.e., let $\varepsilon(p) > D/2 + m$. Let $w \in V$ be a node with maximal distance to p, i.e., $w = \text{argmax}_{v \in V} d_{pv}$. Let q be the node adjacent to p on the path from p to w. Consider now an arbitrary node $z \in V'_{\mathrm{T}}$ and the path between z and q in T'. We distinguish the following two cases: (i) If $p - q - z$ holds, then, $d_{qz} < d_{pz} \leq D'/2$; (ii) Otherwise, if $z - p - q$ holds, then $d_{qz} = d_{pz} + 1$. Since $z - p - q - w$ also holds, we have $d_{qz} = d_{zw} - d_{qw}$. By assumption, $d_{pw} > D/2 + m$ and therefore $d_{qw} \geq D/2 + m$ and we also have $d_{zw} \leq D$. Hence, $d_{qz} \leq D - D/2 - m = D'/2$ and thus $d_{qz} \leq D'/2$ for any $z \in V'_{\mathrm{T}}$. But then, it follows that the node q is also a center of T', which is a contradiction.

Assume now that the diameter D' is not tight. Without loss of generality let p be a central node of T' such that its eccentricity in T is minimal among all possible central nodes, i.e., for each central node u of T' we have $\varepsilon(p) \leq \varepsilon(u)$. Assume again that $\varepsilon(p) > D/2 + m$ and let w and q be the nodes constructed as above. By the same arguments as above, it follows that q is another center of T'. Since we assumed p to be a central node minimizing the eccentricity $\varepsilon(.)$, there must exist $t \in V$ such that $d_{qt} \geq \varepsilon(p)$. Since T is a tree and w is a node with maximum distance from p, the path from w to t is such that $w - q - p - t$. Thus $d_{wt} = d_{pw} + d_{pt} \geq \varepsilon(p) + \varepsilon(p) - 1$. Due to our original assumption we have $d_{wt} > D + 2m - 1$ which contradicts the fact that the diameter of T is at most D for any $m \geq 1$. □

For a feasible solution (T, T') the following lemma shows that either the centers of T and T' coincide, or the upper bound w.r.t. their distance can be tight.

Lemma 2b *If D' is even, $D = D' + 2m$, $m \in \mathbb{N}$, there exist central nodes p of T' and r of T such that they either coincide, i.e., $p = r$, or $d_{pr} = \varepsilon(p) - D/2$.*

Proof From Lemma 2a, it follows that there exists a central node p of T' such that $\varepsilon(p) \leq D/2 + m$. If $\varepsilon(p) \leq D/2$, then p is also a central node of T and $r = p$ and the result holds.

If $\varepsilon(p) > D/2$, then we will find the center r of T as follows. Let $(p = v_0, v_1, \ldots, v_{\varepsilon(p)} = w)$ be the path from p to a node w with maximum distance from p. Let $d_l = \varepsilon(p) - D/2$ and notice that $d_l \geq 1$. We set r to be the node on the $p - w$ path in T such that $d_{pr} = d_l$, which also implies that $d_{rw} = D/2$. It only remains to show that r is a center of T. To see this, consider an arbitrary node $z \in V$. We distinguish the following two cases:

(i) The path from z to w is of the form $z - r - w$: Then, we have $d_{zr} = d_{zw} - d_{rw} \leq D/2$ since the diameter of T is at most D.
(ii) The path from z to w does not contain r. Then z is a successor of r when directing T away from the root p. Thus $d_{rz} = d_{pz} - d_{pr} \leq \varepsilon(p) - m \leq D/2$.

Therefore, for any $z \in V$, we have $d_{rz} \leq D/2$ and thus r is a center of T, which satisfies the desired properties, which concludes the proof. □

By similar arguments, one can prove the following lemmas, stating the analogous relations for the other cases, i.e., when D' or D is odd.

Lemma 3 *If D' is odd and $D = D' + 2m$, $m \in \mathbb{N}$, then there exists a central edge e' of T' such that $\varepsilon(e') \leq \lfloor D/2 \rfloor + m$. Furthermore, there exists a central edge e of T such that either $e = e'$ or $d_{e'e} = \varepsilon(e') - \lceil D/2 \rceil$.*

Lemma 4 *If D' is even and $D = D' + 2m - 1$, $m \in \mathbb{N}$, then there exists a central node p of T' such that $\varepsilon(p) \leq \lfloor D/2 \rfloor + m$. Furthermore, there exists a central edge e of T such that either $d_{pe} = 0$, i.e., $e = \{p, v\}$, or $d_{pe} = \varepsilon(p) - \lceil D/2 \rceil$.*

Lemma 5 *If D' is odd and $D = D' + 2m - 1$, $m \in \mathbb{N}$, then there exists a central edge e' of T' such that $\varepsilon(e') \leq D/2 + m - 1$. Furthermore, there exists a central node r of T such that either $d_{e'r} = 0$, i.e., $e' = \{r, v\}$, or $d_{e'r} = \varepsilon(e') - D/2$.*

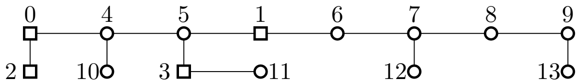


Fig. 1 A feasible solution of the 2-DMSTP with $P = \{0, 1, 2, 3\}$, $D' \geq 4$, and $D \geq 9$.

Summarizing the results of Lemmas 2a-5, we can state the following: There exist centers c' and c of T' and T, respectively, such that they either coincide or we have $d_{c'c} \leq \lfloor D/2 \rfloor - \lceil D'/2 \rceil$.

This result also implies that the *length of the central path* (which also includes the central edges) is at most $\lceil D/2 \rceil - \lfloor D'/2 \rfloor$. To illustrate the result of Proposition 1, consider the solution given in Fig. 1 with $D' = 4$ and $D = 9$. The distance between the two unique centers 4 and $\{1, 6\}$ is exactly $\lfloor D/2 \rfloor - \lceil D'/2 \rceil = 2$ which proves that our upper bound is tight. On the other hand, consider again the solution given in Fig. 1 but with $D' = 4$ and $D = 10$. From Proposition 1 it follows that there must exist a pair of central nodes such that their distance is at most three. For this particular solution, however, node 4 is the unique center of the primary tree, while we may choose either node 1 or node 6 as the center of the secondary tree. Hence, the length of the central path can also be much smaller than the obtained upper bound. This suggests the following question: Can we provide a lower bound on the length of the central path (assuming that centers of the two trees do not coincide)?

2.2 Solutions with Minimum Distance Between the Centers

As observed above, if the diameters are not tight, there exist various choices for choosing centers of T and T'. Assume that for a given instance there does not exist an optimal solution such that the primary and secondary center coincide. Then, among all possible pairs of centers, we are interested in characterizing a pair with minimum distance. The results introduced in this section establish necessary and sufficient conditions for those centers to be at minimum distance. It turns out that either the centers coincide or the subtrees attached to them will have maximum depth. In Section 3.3 we will use this result to derive symmetry breaking inequalities for our model(s).

Proposition 2 *If there does not exist centers (c, c') of (T, T') that coincide, then the length of the central path between c and c' is minimal if and only if there exist two distinct nodes, v in T and w in T', such that $v - c - c' - w$ holds, and $d_{cv} = \varepsilon(c) = \lfloor D/2 \rfloor$, $d_{c'w} = \varepsilon_{T'}(c') = \lfloor D'/2 \rfloor$.*

Proof We will prove this result for D and D' even. The remaining cases can be shown in a similar way. Let r and p be central nodes of T and T', respectively, and let $(r = v_0, v_1, \ldots, v_{l-1}, v_l = p)$ denote the central path from r to p in T.

$\Rightarrow$: First assume (r, p), $r \neq p$, is a pair of central nodes such that $l = d_{pr} \geq 1$ is minimal. If $\nexists v \in V$ such that $d_{rv} = D/2$ and $v - r - p$ holds, then clearly $\varepsilon(v_1) \leq D/2$ would hold and hence v_1 would be a central node of T with

$d_{v_1 p} < d_{pr}$ which contradicts the assumption that d_{pr} is minimal. Likewise, if $\nexists w \in V'_{\mathrm{T}}$ such that $d_{pw} = D'/2$ and $r - p - w$ holds, then $\varepsilon_{T'}(v_{l-1}) \leq D'/2$ and thus v_{l-1} would be a central node of T' closer to r than p.

$\Leftarrow$: If there exist v in T and w in T' such that $d_{rv} = D/2$, $d_{pw} = D'/2$, and $v - r - p - w$ holds, then clearly r and p are the only nodes from the path between r and p that are central nodes of T and T', respectively. Hence (r, p) is a pair of central nodes with minimal distance. □

The proposition above also points out how to find centers (c, c') with minimum distance between them, when given a feasible solution (T, T') with an arbitrary pair of centers $(\bar{c}, \bar{c}')$. Consider, e.g., the solution given in Fig. 1 with $D' = 6$ and $D = 10$ in which case the set of feasible primary centers is $\{0, 4, 5\}$ and the set of feasible secondary centers is $\{1, 6\}$. Now start with any feasible pair of centers, say $p = 0$ and $r = 6$. Observe that among all nodes u such that $p - r - u$ holds, node 13 is the node with maximum distance from r and that $d_{r,13} = 4$. Since $D/2 = 5$, using Proposition 2 we conclude that (p, r) is not a pair of central nodes with minimum distance. Now, from the proof of Proposition 2 we know that $r' = 1$ is also a valid center of the secondary tree. Since $d_{r',13} = 5 = D/2$, r' is the secondary central node closest to the current primary center p. For the primary center, however, among all nodes u such that $u - p - r'$, node 2 is the node with maximum distance $d_{pu} = 1 < D'/2$. Thus, we observe that $p' = 4$ and $p'' = 5$ are primary central nodes closer to the secondary center 1. Finally, we obtain a pair of central nodes $(p'', r') = (5, 1)$ with minimum distance that satisfies the conditions of Proposition 2.

3 MIP Formulations for the 2-DMSTP

In this section, we describe two ways of modeling the 2-DMSTP using layered graphs. The first model that we will refer to as the *two trees model* (2T) considers the intersection of two layered graphs, one of them to model the tree T, the other one to model the subtree T'. Additional coupling constraints relate ("intersect") the two models. The second model is a *three-dimensional layered graph model* (3dLG) that incorporates the properties of Proposition 1 directly into its structure. Results of Proposition 2 are then used to break the symmetries in this model.

3.1 Two Trees Model

To describe this model we first review the layered graph model for the Steiner / spanning tree problem with a single diameter bound $D_m \geq 4$ which is then used as building block the for the (2T) model (cf. Section 3.1.2).

3.1.1 Review: Modeling the Diameter Constrained Steiner Tree Problem

The hop constrained and diameter constrained *spanning tree* problems have been recently modeled and solved successfully by using a branch-and-cut approach on adequate layered graphs (see [8]). We will now review this approach to the DMSTP with the modification for the *Steiner tree variant* (DSTP) as it will be a building block of our (2T) model described below. The approach relies on the idea to model the DSTP as a directed Steiner tree problem in an extended (layered) graph. For a DSTP on graph $G = (V, E)$ with required node set $R \subset V$ and an even diameter D_m, we add a dummy root node 0, set $H_m := D_m/2$ and construct a layered graph $G_L = (V_L, A_L)$ as follows:

- $V_L = \{(0)\} \cup \{i_h : i \in V, 0 \le h \le H_m - 1\} \cup R_L$, where $R_L = \{i_{H_m} : i \in R\}$
- $A_L = A_0 \cup A_1 \cup A_2$ where
  - $A_0 = \{((0), i_0) : i \in V\}$,
  - $A_1 = \{(i_h, j_{h+1}) : (i,j) \in A,\ 0 \le h \le H_m - 2\} \cup \{(i_{H_m-1}, j_{H_m}) : (i,j) \in A,\ j \in R\}$,
  - $A_2 = \{(i_h, i_{H_m}) : i \in R, 0 \le h \le H_m - 1\}$.

The costs of arcs in $A_0 \cup A_2$ are set to zero, and the costs of arcs in A_1 are set to the corresponding c_{ij} values. Then, the DSTP can be modeled as the directed Steiner tree problem on G_L with root 0, the set of terminals equal to R_L, and an extra constraint stating that the out degree of the root node is equal to one. The latter constraint ensures the connectivity of the solution, and that the node at the layer zero, chosen by an optimal solution in the layered graph is a central node of the corresponding optimal tree.

We associate nonnegative variables X^0_{0i} to each arc $((0), i_0) \in A_0$, X^h_{ij} to arcs $(i_{h-1}, j_h) \in A_1$, and X^h_{ii} to arcs $(i_{h-1}, i_{H_m}) \in A_2$. For a subset $\hat{A} \subset A_L$, by $X[\hat{A}]$ we denote the sum of X-variables associated to the arcs of this set. Let $DSTP(R, D_m)$ denote the set of all incidence vectors X corresponding to feasible Steiner trees in G_L that correspond to Steiner trees in G. In [8] the authors use the well known cut set formulation for Steiner trees to derive the currently strongest MIP model for the DMSTP, so we have:

$$\begin{aligned} DSTP(R, D_m) = \{X \in \{0,1\}^{|A_L|} \mid\ & X[\delta^-(W)] \ge 1, \forall W \subset V_L \setminus \{0\}, W \cap R_L \ne \emptyset, \\ & X[\delta^+(0)] = 1, \quad X[\delta^-(i_{H_m})] = 1,\ \forall i_{H_m} \in R_L, \\ & \sum_{h=0}^{H_m-1} X[\delta^-(i_h)] \le 1,\ \forall i \in V \setminus R_L\} \end{aligned} \tag{5}$$

Using binary arc decision variables $a_{ij} \in \{0,1\}$, $\forall (i,j) \in A$, the problem is then solved as

$$\min\{\sum_{(i,j)\in A} c_{ij} a_{ij} \mid a_{ij} = \sum_{h=1}^{H_m} X^h_{ij},\ \forall (i,j) \in A, X \in DSTP(R, D_m)\} \tag{6}$$

Gouveia et al. [8] also showed that a similar approach is possible when D_m is odd, in which case a central edge needs to be selected. This edge is intended to be placed at the layer zero so that the distance of the remaining nodes from that edge is bounded by $H_m := \lfloor D_m/2 \rfloor$. Since an edge cannot be explicitly placed at the layer zero, an additional layer "-1" is introduced, together with nodes i_{-1} for each $i \in V$ and arcs (i_0, j_{-1}) for all $(i,j) \in A$ with costs c_{ij} which represent the potential central edges. Finally, zero-cost arcs (i_{-1}, i_0), $\forall i \in V$, are added to G_L. Again, the DSTP is modeled as the directed Steiner tree problem with the root 0, the set of terminals R_L and *two* additional constraints: (a) the out-degree of the root is one, and (b) the number of solution arcs from layer zero to level "-1" is exactly one. The latter two constraints ensure that the arc connecting a node from layer zero to layer "-1" is exactly the central edge we were looking for. For deriving the corresponding MIP formulation, we need to replace the linking constraints from (6) by $a_{ij} = \sum_{h=-1}^{H_m} X_{ij}^h$, for all $(i,j) \in A$ and to add $\sum_{(i_0, j_{-1}) \in A_\mathrm{L}} X_{ij}^{-1} = 1$ to (5).

In the remainder of this paper, for the DMSTP with diameter bound D_m, the set of all feasible incidence vectors $X \in 2^{|A_\mathrm{L}|}$ of arcs on the layered graph will be denoted by $DMSTP(D_m)$ (cf. the definition given in (5)).

3.1.2 The (2T) Model

The generic MIP formulation introduced in Section 1.2 permits us to use the best known formulations for the DMSTP / DSTP and merge them into a single model by coupling the variables associated to corresponding solutions. Using the layered graph approach introduced in Section 3.1.1 to model diameter constrained trees in general, we can model the 2-DMSTP as follows:

$$
\begin{aligned}
\min \sum_{e \in E} & c_e x_e && \\
& X^1 \in DSTP(P, D') && (7)\\
& X^2 \in DMSTP(D) && (8)\\
& \sum_h \left(X_{ij}^{h,1} + X_{ji}^{h,1} \right) \le x_e \qquad \forall e = \{i,j\} \in E && (9)\\
& \sum_h \left(X_{ij}^{h,2} + X_{ji}^{h,2} \right) = x_e \qquad \forall e = \{i,j\} \in E && (10)\\
& x \in \{0,1\}^{|E|}
\end{aligned}
$$

This model is based on the intersection of two layered graphs: Graph G_L^1 is used to model diameter constrained Steiner trees with terminal set equal to P and the diameter bounded by D'. Graph G_L^2 is used to model diameter constrained spanning trees in G with the diameter bounded by D. Binary variables X^1 and X^2 are the incidence vectors of diameter constrained solutions on graphs G_L^1 and G_L^2, respectively. Variables X^1 and X^2 are finally linked to undirected edge variables using constraints (9) and (10).

Although the model appears to be quite intuitive (given the generic formulation introduced in Section 1.2) and may be moderately successful in solving the problem, it contains two main drawbacks:

(i) The two subproblems are only loosely coupled together in terms of original variables, and
(ii) The model is undirected, i.e., the coupling constraints (9) and (10) only relate layered arc variables to undirected edge design variables x_e and it remains unclear how to direct this model.

In the next subsection we show how to strengthen the lower bounds of this model by adding some additional valid inequalities derived from the graph theoretical properties of feasible solutions.

3.1.3 Strengthening the (2T) Model

Due to Proposition 1, several strengthening coupling constraints between arc variables X^1 and X^2 can be derived as shown below. If D' and D are even, $D = D' + 2m$, let r and p be the centers of T and T', respectively. We have:

$$X_{ij}^{h,1} \leq \sum_{l=1}^{m+h} X_{ij}^{l,2} + \sum_{l=1}^{m-h+1} X_{ji}^{l,2} \quad \forall (i,j) \in A,\ 1 \leq h \leq \min\{m, D'/2\} \tag{11}$$

$$X_{ij}^{h,1} \leq \sum_{l=h-m}^{h+m} X_{ij}^{l,2} \quad \forall (i,j) \in A,\ m+1 \leq h \leq \lfloor D'/2 \rfloor \tag{12}$$

To see that these inequalities are valid notice that if arc (i,j) is at distance $h \leq \min\{m, D'/2\}$ from the primary center p, then this arc may belong to the central path. Regarding the location of the arc (i,j) with respect to the two centers p and r, we distinguish two cases: (i) If $r - p - i - j$ holds, then the same arc (i,j) is used in X^2, and its distance from the secondary center r is at most $m+h$. (ii) If $p-i-j-r$ holds, then in the X^2-solution arc (j,i) is used, and its distance from r is at most $m - h + 1$. These facts are incorporated in inequalities (11), for all $(i,j) \in A$. On the other hand, if arc (i,j) is at distance $m < h \leq \lfloor D'/2 \rfloor$ from the primary center p, then the same arc has to be used in the spanning tree X^2, and its distance from the secondary center r is at most $h + m$, and at least $h - m$, which is expressed using inequalities (12).

If D' and D are odd, $D = D' + 2m$, $m \in \mathbb{N}$, we have to replace (11) by the following two sets of inequalities:

$$X_{ij}^{-1,1} \leq X_{ji}^{-1,2} + \sum_{l=1}^{m} (X_{ij}^{l,2} + X_{ji}^{l,2}) \quad \forall (i,j) \in A \tag{13}$$

$$X_{ij}^{h,1} \leq X_{ji}^{-1,2} + \sum_{l=1}^{m-h} X_{ji}^{l,2} + \sum_{l=1}^{m+h} X_{ij}^{l,2} \quad \forall (i,j) \in A, 1 \leq h \leq \min\{m, \lfloor D'/2 \rfloor\} \tag{14}$$

Inequalities (13) exploit the fact that the primary and secondary central edges e' and e either coincide, or that the distance of e' from e is at most $m - 1$ (see Lemma 3). In case the edges coincide, there are two possible edge orientations, and by considering only one of them, we are able to break the symmetries. Inequalities (14) are the adaptation of (11) in which the secondary central arc is considered. Again, by choosing one of the two possible directions, we break the symmetries of this model.

If D' is even and D is odd, $D = D' + 2m - 1$, $m \in \mathbb{N}$, the length of the central path is at most m (see Lemma 4), i.e., the distance between the two centers is at most $m - 1$. This fact is exploited by inequalities (15)–(17).

$$X_{ij}^{h,1} \leq X_{ji}^{-1,2} + \sum_{l=1}^{m-h} X_{ji}^{l,2} + \sum_{l=1}^{m+h-1} X_{ij}^{l,2} \quad \forall (i,j) \in A, 1 \leq h \leq \min\{m-1, D'/2\} \quad (15)$$

$$X_{ij}^{m,1} \leq X_{ji}^{-1,2} + \sum_{l=1}^{2m-1} X_{ij}^{l,2} \quad \forall (i,j) \in A \quad (16)$$

$$X_{ij}^{h,1} \leq \sum_{l=h-m}^{h+m-1} X_{ij}^{h,2} \quad \forall (i,j) \in A,\ m+1 \leq h \leq D'/2 \quad (17)$$

Finally, if D' is odd and D is even, $D = D' + 2m - 1$, $m \in \mathbb{N}$, we consider inequalities (18)–(20), which exploit the results of Lemma 5.

$$X_{ij}^{-1,1} \leq \sum_{l=1}^{m} (X_{ij}^{l,2} + X_{ji}^{l,2}) \quad \forall (i,j) \in A \quad (18)$$

$$X_{ij}^{h,1} \leq \sum_{l=1}^{m-h} X_{ji}^{l,2} + \sum_{l=1}^{m+h} X_{ij}^{l,2} \quad \forall (i,j) \in A, 1 \leq h \leq \min\{m-1, \lfloor D'/2 \rfloor\} \quad (19)$$

$$X_{ij}^{h,1} \leq \sum_{l=h-m+1}^{h+m} X_{ij}^{l,2} \quad \forall (i,j) \in A,\ m \leq h \leq \lfloor D'/2 \rfloor \quad (20)$$

3.2 Three-Dimensional Layered Graph Model (3dLG)

In this section, by exploiting in a different manner the graph theoretical results of Section 2 we show that the problem can be viewed as a *single directed Steiner tree* problem on a more sophisticated layered graph, more precisely a three-dimensional layered graph, with a moderate number of additional constraints. Many of the graph theoretical relations established in Section 2 will be satisfied by the new graph, making unnecessary the inclusion of additional constraints as it was done in the previous subsection to strengthen the (2T) model.

To make the explanation easier, we define the upper bounds on eccentricities for the primary and secondary center as $H' := \lfloor D'/2 \rfloor$ and $H := \lfloor D/2 \rfloor$, respectively. As before, in order to simplify the explanation of the new model we start by assuming that D and D' are even and let r and p be the centers of T and T', respectively. Let us now *direct* the tree T by making it a rooted

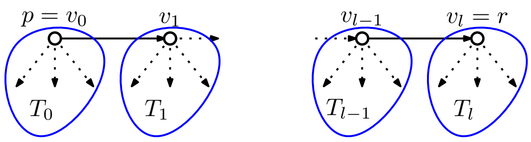


Fig. 2 Feasible 2-DMSTP solution presented as a spanning arborescence rooted at the primary center. Arborescences dangling on the central path are denoted by $T_0, \ldots, T_l$.

arborescence with the root equal to p. We can view the solution as composed by a directed path between p and r, together with several subarborescences whose roots are nodes in the path. More precisely, the solution contains a directed path from p to r ($p = v_0, v_1, \ldots, v_{l-1}, v_l = r$), $d_{pr} = l$. Each of the nodes v_i, $i = 0, \ldots, l$, is a root of a subarborescence of T, denoted by T_i (see Fig. 2) and each subarborescence T_i, $i = 0, \ldots, l$, satisfies the following properties:

(P1) The maximal length of a path from T' in T_0 is H'
(P2) The maximal length of a path from T' in T_i is at most $H' - i$
(P3) The maximal length of any path in T_l is H, and
(P4) The maximal length of any path in T_i is at most $H - l + i$.

We note that these properties indicate the distance of a node in P (or in S) to the path. For the moment we will replace (P4) which gives the maximum distance from a node S to the main path by (P4'):

(P4') The maximal length of any path in T_i is at most H.

This relaxed problem can be modeled as an arborescence in a large (three-dimensional) layered graph in which each of the arborescences T_i is modeled as a separate (two-dimensional) layered graph. The constraints on the lengths of a path in each of layered graphs associated to T_i are guaranteed explicitly by imposing a certain number of layers on these graphs. The central path will be directed from the primary center to the secondary center and Proposition 1 gives an upper bound on the length of this path. This path links the layered graphs associated to each subarborescence and we define the *depth of the layered graph* as the length of this path. In this way, we are able to obtain a fully directed model, that is a model that views each feasible solution of this relaxation (replacing (P4) by (P4')) as a spanning arborescence rooted at the primary central node.

The main difficulty of this approach is that property (P4) is not guaranteed by the graph construction. (For the sake of completeness we state that the problem modeled in this way is the original problem where the maximum distance between nodes in S is augmented by the maximum length of the central path.) Thus, we need to add extra constraints to guarantee this condition.

Each node of the layered graph is denoted by i_h^d - it is associated to the original node i at layer h and depth d. Arcs between nodes of different depths only exist at layer zero and the path realized at layer zero represents the central path. For a fixed depth d, and fixed hop-limits H_1 and H_2 associated to nodes from P and S, respectively, we first explain how to construct a layered graph

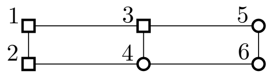


Fig. 3 An exemplary instance with $P = \{1, 2, 3\}$, $S = \{4, 5, 6\}$, $D' = 4$, and $D = 8$.

$G_{2L}^{H_1,H_2,d} = (V_{2L}^{H_1,H_2,d}, A_{2L}^{H_1,H_2,d})$ which will ease the necessary definitions later on. The graph $G_{2L}^{H_1,H_2,d}$ $(0 < H_1 \leq H_2)$ is constructed so that copies of each node i from P are made at layers 0 to H_1 and nodes from S are copied at all layers between 0 and H_2. Arcs are superimposed so that for each edge $e = \{i, j\}$ from E, arcs are added between the copies of i and j (and j and i) in two consecutive layers.We have:

$$\begin{aligned} V_{2L}^{H_1,H_2,d} = & \{i_h^d : i \in V,\ 0 \leq h \leq H_1\} \cup \{i_h^d : i \in S,\ H_1 < h \leq H_2\} && (21)\\ A_{2L}^{H_1,H_2,d} = & \{(i_h^d, j_{h+1}^d) : (i,j) \in A,\ 0 \leq h \leq H_1 - 1\} \cup \\ & \{(i_{H_1}^d, j_{H_1+1}^d) : (i,j) \in A, i \in V, j \in S\} \cup \\ & \{(i_h^d, j_{h+1}^d) : (i,j) \in A, \{i,j\} \cap P = \emptyset,\ H_1 < h \leq H_2 - 1\} && (22) \end{aligned}$$

In the following, the case when both D' and D are even is discussed in detail before showing the necessary adaptations for the other cases. We will use the following notation:

$$d_{\max} = \lceil D/2 \rceil - \lfloor D'/2 \rfloor \quad \text{and} \quad \tilde{n} = D' \bmod 2,$$

where $d_{\max}$ denotes the maximum length of the central path (including the possibly existing central edges), and parameter $\tilde{n}$ indicates whether there is a primary central edge, in which case (P2) is reformulated as "The maximal length of a path from T' in T_i is at most $H' - i + 1$, for all $i = 1, \ldots, l$".

We model the 2-DMSTP on G_{3L} as a Steiner arborescence problem with the set of terminals R_{3L} and some additional constraints, where $G_{3L} = (V_{3L}, A_{3L})$ is defined as follows:

$$V_{3L} = \{(0)\} \cup R_{3L} \cup V_{2L}^{H',H,0} \cup (\bigcup_{d=1}^{d_{\max}} V_{2L}^{H'-d+\tilde{n},H,d}) \text{ with } R_{3L} = \{(i) : i \in V\} \text{ and}$$

$$A_{3L} = A_{3L}^{0} \cup (\bigcup_{d=1}^{d_{\max}} A_{3L}^{\mathrm{C,d}}) \cup (\bigcup_{d=0}^{d_{\max}} A_{3L}^{\mathrm{Z,d}}) \cup A_{2L}^{H',H,0} \cup (\bigcup_{d=1}^{d_{\max}} A_{2L}^{H'-d+\tilde{n},H,d}), \text{ where}$$

$$A_{3L}^{0} = \{((0), i_0^0) : i \in V\} \text{ (zero costs)}$$

$$A_{3L}^{\mathrm{C,d}} = \{(i_0^{d-1}, j_0^d) : (i,j) \in A,\ i_0^{d-1}, j_0^d \in V_{3L}\}$$

$$A_{3L}^{\mathrm{Z,d}} = \{(i_h^d, (i)) : i \in V, i_h^d \in V_{3L}\} \text{ (zero costs)}$$

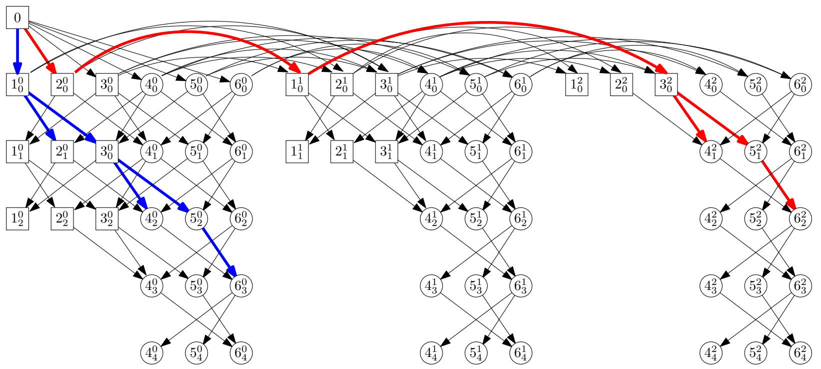


Fig. 4 3-dimensional layered graph for the instance given in Fig. 3 and two possible embeddings of the solution $\{\{1,2\},\{1,3\},\{3,4\},\{3,5\},\{5,6\}\}$. For the solution shown in blue, 1 is the primary and secondary center, while for the solution shown in red, 2 is the primary center and 3 is the secondary center. Thus for the latter solution, the central path consists of two arcs. For simplicity nodes in $R_{3\mathrm{L}}$ and their ingoing arcs are not drawn.

Fig. 4 illustrates an example in which $D = 8$, $D' = 4$ and, for the sake of simplicity, the nodes of $R_{3\mathrm{L}}$ and the arcs from $\bigcup_{d=0}^{d_{\max}} A_{3\mathrm{L}}^{\mathrm{Z,d}}$ are not shown. The costs of the arcs from $A_{3\mathrm{L}}^{0} \cup (\bigcup_{d=0}^{d_{\max}} A_{3\mathrm{L}}^{\mathrm{Z,d}})$ are set to zero, and the costs of the arcs from $\bigcup_{d=1}^{d_{\max}} A_{3\mathrm{L}}^{\mathrm{C,d}} \cup A_{2\mathrm{L}}^{H',H,0} \cup (\bigcup_{d=1}^{d_{\max}} A_{2\mathrm{L}}^{H'-d+\tilde{n},H,d})$ are set to the original cost, c_{ij}, for each of the corresponding arcs $(i,j) \in A$. The primary center is chosen from the subgraph $V_{2\mathrm{L}}^{H',H,0}$ and if primary and secondary center do not coincide, the central path is modeled using the arcs from $\cup_{d=1}^{d_{\max}} A_{3\mathrm{L}}^{\mathrm{C,d}}$. By construction, graph $G_{3\mathrm{L}}$ satisfies properties (P1)-(P3). However, since we do not know the exact length l of the central path, the property (P4) is ensured by explicitly imposing some extra constraints in the model. The following binary variables are used in our MIP formulation:

- X_{0j}^{00}, for arcs $((0), j_0^0) \in A_{3\mathrm{L}}^0$,
- X_{ij}^{hd} for arcs $(i_{h-1}^d, j_h^d) \in A_{2\mathrm{L}}^{H,H',0}$ or $\in A_{2\mathrm{L}}^{H-d+\tilde{n},H',d}$ for $d = 1, \ldots, d_{\max}$,
- X_{ij}^{0d} for arcs $(i_0^{d-1}, j_0^d) \in A_{3\mathrm{L}}^{\mathrm{C,d}}$ where $d = 1, \ldots, d_{\max}$, and
- X_{ii}^{hd} for arcs $(i_{h-1}^d, (i)) \in A_{3\mathrm{L}}^{\mathrm{Z,d}}$ where $d = 0, \ldots, d_{\max}$.

Furthermore, we use binary variables z_d, $0 \le d \le d_{\max}$, with $z_d = 1$ if the distance between the two chosen centers is at least d and zero otherwise. The reason why these variables are included is to enforce the statement of property (P4) that is not implicitly guaranteed by $G_{3\mathrm{L}}$. Finally, arc variables $a_{ij} \in \{0,1\}$, $\forall (i,j) \in A$, are used to project the arcs from $G_{3\mathrm{L}}$ into G.

3.2.1 D' and D even

For D and D' are even, the 2-DMSTP on $G_{3\mathrm{L}}$ is modeled as follows:

$$\min \sum_{(i,j)\in A} c_{ij} a_{ij} \tag{23}$$

$$\text{s.t.} \quad X[\delta^-(W)] \geq 1 \qquad \forall W \subset V_{3\text{L}},\ W \cap R_{3\text{L}} \neq \emptyset,\ (0) \notin W \tag{24}$$

$$\sum_{d=0}^{\min\{d_{\max},H'\}} \sum_{h=0}^{H'-d} X[\delta^-(i_h^d)] = 1 \qquad \forall i \in P \tag{25}$$

$$\sum_{d=0}^{d_{\max}} \sum_{h=0}^{H} X[\delta^-(i_h^d)] = 1 \qquad \forall i \in S \tag{26}$$

$$\sum_{i\in V} X[\delta^-(i_0^d)] = z_d \qquad d = 0,\ldots,d_{\max} \tag{27}$$

$$a_{ij} = \sum_{d=0}^{d_{\max}} \sum_{h=1}^{H} X_{ij}^{hd} + \sum_{d=1}^{d_{\max}} X_{ij}^{0d} \qquad \forall (i,j) \in A \tag{28}$$

$$\sum_{t=0}^{d-1} \sum_{h=H-d+t+1}^{H} X[\delta^-(i_h^t)] \leq 1 - z_d \qquad \forall i \in V, d = 1,\ldots,d_{\max} \tag{29}$$

$$X \in \{0,1\}^{|A_{3\text{L}}|},\ z \in \{0,1\}^{d_{\max}+1},\ a \in \{0,1\}^{|A|} \tag{30}$$

Directed cutset constraints (24) ensure connectivity between the artificial root (0) and every *terminal node* in $R_{3\text{L}}$, while the indegree constraints (25) and (26) state that across all layers and all depths, each node $i \in V$ is visited exactly once. Inequalities (27) ensure that for each depth, at most one node is chosen at layer zero, and establish the connection to variables z_d. Constraints (28) link the arcs of the layered graph with the arcs in A. Notice that for the sake of simplicity, in writing the summation terms, (28) also includes reference to variables X_{ij}^{hd} that do not exist in our layered graph (and these can be considered as fixed to zero). Finally, inequalities (29) are added to forbid too long paths between secondary nodes. They state that if the length of the central path is at least $d+1$, then, each of the subtrees dangling at the depth d' ($d' \leq d$) can contain paths whose length is at most $H-d+d'$, i.e., the nodes at the layers $\geq H-d+d'$ are forbidden. Note, that without inequalities (29), we have a valid model for the previously mentioned relaxation of the 2-DMSTP obtained by replacing (P4) by (P4') where the maximum distance between secondary nodes is at most $D+d_{\max}$.

Slight modification to this model need to be made for the remaining cases, when D' or D is odd. These cases are studied in the remainder of this section.

3.2.2 D' odd, D odd

If D and D' are odd, the main difference to the previous case is that we now have central edges instead of central nodes and, as a consequence the maximum number of allowed layers for primary nodes in each of the subgraphs for a fixed depth starts to decrease from $d = 2$ rather than from $d = 1$. This fact is incorporated in the previous definition of G_{3L} by considering $\tilde{n}$ which is equal to one in this case. To correctly model this case, we replace (25) by (31), replace constraints (29) by inequalities (32), and set $z_1 = 1$. Otherwise, the model and variables are defined analogously.

$$\sum_{h=0}^{H'} X[\delta^-(i_h^0)] + \sum_{d=1}^{\min\{d_{\max},H'\}} \sum_{h=0}^{H'-d+1} X[\delta^-(i_h^d)] = 1 \qquad \forall i \in P \tag{31}$$

$$\sum_{t=0}^{d-2} \sum_{h=H-d+t+2}^{H} X[\delta^-(i_h^t)] \le 1 - z_d \qquad i \in V,\ d = 2, \ldots, d_{\max} \tag{32}$$

3.2.3 D' even, D odd

If D is odd and D' is even, the primary tree has a central node (which is chosen at layer zero and depth zero) and thus the maximum number of allowed layers for primary nodes starts decreasing from $d = 1$, i.e., $\tilde{n} = 0$. Again, the previous model and all variables are defined analogously. Since D is odd, however, the secondary tree has a central edge and thus we replace constraints (29) by inequalities (32) and set $z_1 = 1$.

3.2.4 D' odd, D even

If D is even and D' is odd, we have a primary central edge which is chosen from $A_{3L}^{C,1}$ and a secondary central node which is chosen at layer zero and depth greater than or equal to one. Thus, the maximum feasible layer of secondary nodes at depth zero is equal to $H - 1$ and hence we replace $V_{2L}^{H',H,0}$ by $V_{2L}^{H',H-1,0}$ and $A_{2L}^{H',H,0}$ by $A_{2L}^{H',H-1,0}$ in the previously given definition of G_{3L}. As for the case when D' and D are odd, the fact that the maximum layer of primary nodes starts decreasing from $d = 2$ rather than from $d = 1$ is captured by the definition of $\tilde{n}$. Furthermore, we replace (25) by (31) and set $z_1 = 1$. Otherwise, both the model and the variables are defined analogously to the case when both D' and D are even.

3.3 Symmetry Breaking Constraints for (3dLG)

In case the length of the central path can be less than $d_{\max}$, there will be different feasible Steiner trees on the layered graph modeling the same solution in terms of original variables, cf. Fig 4. To avoid this situation, we use Proposition 2 to derive corresponding symmetry breaking constraints for the (3dLG) model. These constraints will ensure that each solution modeled by the (3dLG) model will have minimal distance between the two chosen centers.

D' even, D even Assume that nodes p and r are the centers of T' and T, respectively. If $p \neq r$, then $z_1 = 1$. In that case, we want to make sure that the centers are at minimal distance, and hence with constraints (33) we enforce that there must exist a node $i \in P$, such that $d_{pi} = \varepsilon_{T'}(p) = H'$ and $i - p - r$ holds. Furthermore, constraints (34) and (35) ensure that if $d_{pr} = l$, $1 \leq l \leq m$, then for at least one node $i \in V$ with $p - r - i$, we have $d_{ri} = \varepsilon(r) = H$.

$$\sum_{i \in P} X[\delta^-(i^0_{H'})] \geq z_1 \tag{33}$$

$$\sum_{i \in S} X[\delta^-(i^d_H)] \geq z_d - z_{d+1} \qquad d = 1, \dots, d_{\max} - 1 \tag{34}$$

$$\sum_{i \in S} X[\delta^-(i^{d_{\max}}_H)] \geq z_{d_{\max}} \tag{35}$$

D or D' odd In all the remaining cases (when at least one of the diameters is odd), constraints (33) are replaced by (36), and (34) are replaced by (37):

$$\sum_{i \in P} X[\delta^-(i^0_{H'})] \geq z_2 \tag{36}$$

$$\sum_{i \in S} X[\delta^-(i^d_H)] \geq z_d - z_{d+1} \qquad d = 2, \dots, d_{\max} - 1 \tag{37}$$

Below we describe additional constraints that can further break the symmetries of our model.

D and D' odd In case that the two center edges coincide, for every solution we can obtain a symmetric solution by exchanging all chosen nodes / arcs on depth zero and one. Thus, we consider a further set of symmetry breaking constraints (38). These simply guarantee that, if the two central edges are identical, i.e., if $z_2 = 0$, the index of the node chosen at layer zero and depth zero is smaller than the one chosen at layer zero and depth one:

$$X[\delta^-(i^0_0)] + \sum_{j<i} X[\delta^-(j^1_0)] \leq 1 + z_2 \qquad i \in V \tag{38}$$

D' even, D odd There is an additional case, where symmetric solutions may cause difficulties. This situation occurs whenever an optimal solution exists in which either node of the secondary center edge can be chosen as primary central node. We observe, that in this situation, z_2 will be equal to zero (since the centers coincide) and no primary node will be chosen at layer H' and depth zero, since the node chosen at layer zero and depth one would not be a primary center otherwise. By adding constraints (39) we ensure that if both nodes incident to the secondary central edge are primary central nodes, we chose the one with smaller index at depth zero.

$$X_{ji}^{01} \leq z_2 + \sum_{p \in P} X[\delta^-(p_{H'}^0)] \qquad (j,i) \in A, j > i \qquad (39)$$

D′ odd, D even For the case when either node of the primary central edge can be chosen as secondary central node, we observe that both a node i chosen at layer zero and depth zero and a node j chosen at layer zero and depth one are valid secondary centers, if no node is active at layer H and depth one. Thus, by constraints (40) we ensure that in this case the node chosen at layer zero and depth zero has a smaller index than the one at layer zero and depth one.

$$X_{ji}^{01} \leq z_2 + \sum_{k \in S} X[\delta^-(k_H^1)] \qquad (j,i) \in A, j > i \qquad (40)$$

4 Branch-and-Cut Algorithms. Computational Study.

In this section we compare the computational performance of three branch-and-cut algorithms (B&C) for the following models: (i) (2T), (ii) $(2\mathrm{T})^+$ (which is a variant of (2T) with additional inequalities (11)–(18), cf. Section 3.1.3), and (iii) (3dLG). B&C approaches are implemented in C++ using IBM CPLEX 12.4. The evaluation and comparison of the three approaches is conducted on a benchmark instances from [8] that have been frequently used for testing the HMSTP and the DMSTP approaches. We choose the first instance from each of the groups of random (TR) and Euclidean instances (TC) with 31, 41, and 61 nodes. For the sake of simplicity, in the following we will use 30, 40, and 60 to refer to instances with 31, 41, and 61 nodes, respectively. Each graph is complete, and we use the first $|P|$ nodes of an instance as primary nodes. Furthermore, for each instance set we set $D' = 3, \ldots 6$, and for each D' we consider $D = D' + i$, for $i = 2, \ldots, 5$. Regarding the number of primary nodes, we choose $|P| \in \{5, 10, \ldots, |V| - 1\}$ for instances with 31 and 41 nodes, respectively, and $|P| \in \{10, 20, \ldots, 60\}$ for instances with 61 nodes.

B&C Configuration for (2T) and $(2\mathrm{T})^+$ We initialize the MIP model with the compact constraints from (5) for G_{L}^1 and G_{L}^2 and we add a compact number of connectivity constraints:

$$\sum_{(i_{h-1}, j_h) \in A_{\mathrm{L}}^{\ell},\ i \neq k} X_{ij}^{h,\ell} \geq X_{jk}^{h+1,\ell} \qquad \forall (j_h, k_{h+1}) \in A_{\mathrm{L}}^{\ell},\ \ell = 1,2 \qquad (41)$$

to ensure connectivity on each of the two layered graphs. Furthermore, in CPLEX, we assign higher branching priorities to edge variables than to layered arc variables. Regarding $(2\mathrm{T})^+$, the MIP is initialized with inequalities (11)–(18), as our preliminary tests have shown that this variant performs better than a dynamic separation of these constraints.

B&C Configuration for (3dLG) The highest branching priority is given to z-variables, followed by the arc design variables a, which in turn are given higher branching priority than layered arc variables. We initialize the B&C with (25)-(29) and with the appropriate compact connectivity cuts (similar to (41)) applied to each of the subgraphs $G_{2L}^{H'-d+\tilde{n},H,d}$. In addition we add:

$$z_d \geq z_{d+1} \qquad d = 0, \dots, d_{\max} - 1 \qquad (42)$$

Symmetry breaking constraints (cf. Section 3.3) are dynamically separated.

Experimental Set-Up All experiments have been performed on a single core of an Intel Xeon processor with 2.53 GHz using at most 3GB RAM. We used the single threaded variant of IBM CPLEX 12.4 and an absolute time limit of 10 000 CPU-seconds has been applied in all experiments. For the separation of cutset inequalities in (5) and (24), we run the maximum flow algorithm of Cherkassky and Goldberg [2]. In all separation variants, we used nested and backcuts, cf. [12]), and inserted at most 100 violated cuts in each iteration.

4.1 Computational Experiments

We first analyze the overall performance of (2T), $(2T)^+$ and (3dLG) on instances with 31 and 41 nodes with respect to their LP gaps (see Fig. 5a) and with respect to the CPU-time needed by each of the B&C approaches (see Fig. 5b).

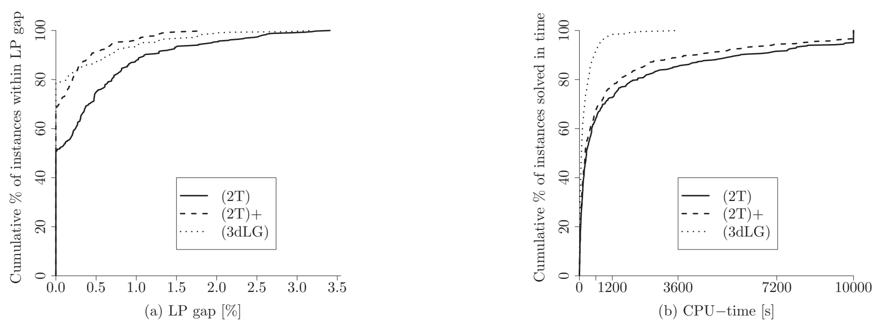


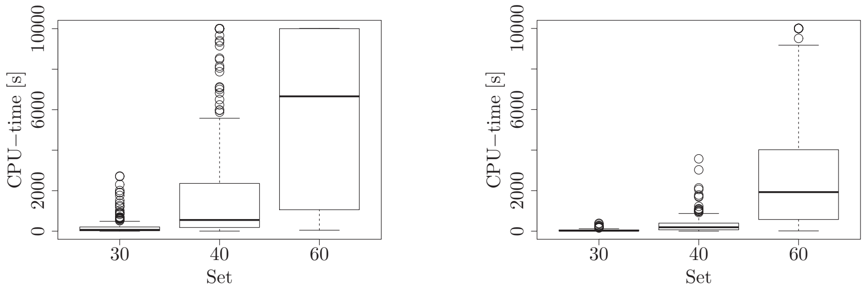
Fig. 5 LP gaps and B&C CPU-times for (2T), $(2T)^+$, and (3dLG).

From Fig. 5a we conclude that adding inequalities (11)–(18) to model (2T) significantly increases the obtained LP bounds. Furthermore, neither $(2T)^+$ nor (3dLG) dominates the other with respect to the quality of LP bounds. This is not surprising given that (3dLG) is directed (as opposed to $(2T)^+$) but does not incorporate the maximum allowed distance between secondary

Table 1 Numbers of instances solved to optimality ($\#_{\text{solved}}$), geometric means of CPU-times in seconds, and average optimality gaps in [%], grouped by $|P|$.

		$\#_{\text{solved}}$			CPU-time [s]			Gap [%]		
$\|P\|$	#	(2T)	(2T)$^+$	(3dLG)	(2T)	(2T)$^+$	(3dLG)	(2T)	(2T)$^+$	(3dLG)
5	64	**64**	**64**	**64**	58	**46**	109	**0.0**	**0.0**	**0.0**
10	96	82	**87**	**87**	319	**260**	273	5.2	3.1	**0.0**
15	64	62	**64**	**64**	256	174	**83**	**0.0**	**0.0**	**0.0**
20	96	85	89	**94**	487	377	**149**	3.1	6.3	**0.0**
25	64	60	63	**64**	367	281	**33**	**0.0**	**0.0**	**0.0**
30	96	89	92	**96**	801	805	**147**	2.1	3.1	**0.0**
35	32	28	27	**32**	1512	1429	**89**	3.2	3.2	**0.0**
40	64	41	39	**64**	3209	3732	**474**	15.7	15.7	**0.0**
50	32	12	14	**32**	4971	5039	**749**	31.4	31.4	**0.0**
60	32	12	9	**32**	5455	6970	**1343**	19.0	25.2	**0.0**

nodes into its structure. From a practical perspective, we observe that (3dLG) outperforms (2T)$^+$ with respect to the number of instances for which the LP relaxation is integral, while the smallest maximum LP-gap is obtained for (2T)$^+$. With respect to the CPU-times needed for solving the integer models, (3dLG) clearly outperforms (2T)$^+$, which in turn performs better than its simpler variant (2T).



(a) (2T)$^+$ (b) (3dLG)

Fig. 6 CPU-times of (2T)$^+$ and (3dLG) for instance sets 30, 40, and 60.

To analyze the performances and possibly existing individual advantages of (2T), (2T)$^+$, and (3dLG) in more detail, we present numbers of solved instances, geometric means of CPU-times (in seconds), and average optimality gaps (in %) grouped by the numbers of primary nodes, size of the given instance graph, and diameter values (see Tables 1, 2, and 3, respectively). Furthermore, Fig. 6 provides the distribution of CPU-times for the various settings of D, D', and $|P|$, grouped by the size of the given instance graph.

From Table 1 we are able to draw the important conclusion that model (2T)$^+$ performs comparably well to (3dLG) whenever the number of primary nodes is very small. We argue that this is due to the fact that the size of G_{3L} as well as the number of constraints needed to ensure property (P4) decreases

Table 2 Numbers of instances solved to optimality ($\#_{\text{solved}}$), geometric means of CPU-times in seconds, and average optimality gaps in [%], grouped by the instance graph.

		$\#_{\text{solved}}$			CPU-time [s]			Gap [%]		
Inst	#	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)
TC30	96	95	**96**	**96**	122	116	**37**	**0.0**	**0.0**	**0.0**
TR30	96	**96**	**96**	**96**	41	41	**14**	**0.0**	**0.0**	**0.0**
TC40	128	117	123	**128**	818	648	**200**	1.6	1.6	**0.0**
TR40	128	117	117	**128**	777	555	**120**	**0.0**	0.8	**0.0**
TC60	96	45	48	**85**	4161	4111	**2026**	26.1	30.3	**0.0**
TR60	96	65	68	**96**	2531	2380	**1018**	10.5	9.5	**0.0**

Table 3 Numbers of instances solved to optimality ($\#_{\text{solved}}$), geometric means of CPU-times in seconds, and average optimality gaps in [%], grouped by (D', D).

			$\#_{\text{solved}}$			CPU-time [s]			Gap [%]		
D'	D	#	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)
3	5	40	36	37	**40**	448	196	**93**	**0.0**	**0.0**	**0.0**
3	6	40	35	35	**40**	310	263	**56**	0.1	0.1	**0.0**
3	7	40	36	35	**40**	402	484	**149**	2.5	0.1	**0.0**
3	8	40	35	35	**40**	352	428	**132**	0.1	0.1	**0.0**
4	6	40	37	38	**40**	259	184	**52**	**0.0**	**0.0**	**0.0**
4	7	40	37	38	**39**	418	317	**110**	**0.0**	**0.0**	**0.0**
4	8	40	36	35	**39**	333	313	**110**	**0.0**	**0.0**	**0.0**
4	9	40	31	31	**39**	688	617	**143**	**0.0**	5.0	**0.0**
5	7	40	33	37	**40**	1391	914	**394**	2.6	**0.0**	**0.0**
5	8	40	34	36	**39**	958	811	**280**	5.1	5.0	**0.0**
5	9	40	29	31	**38**	1733	1786	**584**	7.6	12.5	**0.0**
5	10	40	30	32	**38**	1683	1677	**534**	10.1	12.5	**0.0**
6	8	40	36	37	**40**	335	285	**81**	10.0	7.5	**0.0**
6	9	40	32	32	**39**	603	606	**225**	17.5	17.5	**0.0**
6	10	40	33	32	**39**	522	508	**212**	15.0	17.5	**0.0**
6	11	40	25	27	**39**	1009	1055	**373**	22.5	25.0	**0.0**

with the increasing number of primary nodes. This also explains, why, on the contrary to (2T) and $(2T)^+$, model (3dLG) scales remarkably well with respect to the increasing number of primary nodes. Our observations due to the results shown in Table 2 are twofold: (i) Euclidean instances (TC) generally seem to be much harder to solve than random instances (TR), and (ii) the (3dLG) model clearly scales better to larger instance graphs than the two variants of (2T). The latter fact is also strongly supported by Fig. 6. From Table 3, we conclude that the difficulty of the instances increases with increasing difference between D and D'. Furthermore, as expected from what is known for the DMSTP, even diameter cases seem to be easier to solve than situations where at least one diameter is odd.

Overall, we conclude that (3dLG) provides superior performance and outperforms the two variants of the two-trees model for almost all variations of input parameters. Further note that for (3dLG) only very few instances could not be solved to proven optimality within the given time limit (11 out of 640) and the remaining gap between upper and lower bounds is usually very small (less than 0.25%). This is in contrast to the performance of models (2T) and

$(2T)^+$, in particular for large instances with 61 nodes. For model $(2T)^+$ 92 out of 640 instances were not solved to optimality and the maximum obtained gaps were about 100%. These findings are also supported by the more detailed results presented in Tables 4, 5, 6, and 7 in the Appendix.

5 Conclusions

In this article, we introduced the Two-Level Diameter Constrained Spanning Tree Problem (2-DMSTP) which generalizes the DMSTP by considering two sets of nodes with different maximum latency requirements. Studying graph theoretical properties related to the centers of each tree allowed us to propose two MIP formulations based on layered graphs, strengthening valid inequalities and symmetry breaking constraints. In particular our second model which is based on a novel three-dimensional layered graph approach (3dLG) turned out to perform extremely well in practice. Our computational study has shown that the (3dLG) approach works particularly well for graphs where the proportion of the number of primary nodes over the number of secondary nodes is large. Besides considering a generalization of the problem to more than two different sets of nodes (k-DMSTP), further interesting aspects to be studied in the future include the study of a model that intersects two three-dimensional layered graph models, one incorporating primary and a second incorporating secondary distance constraints in its structure. In such a model explicit distance constraints become redundant, since each distance is guaranteed by one of the two layered graphs. It is, however, not at all obvious how to propose effective linking constraints in order to obtain a model that dominates in theory the ones proposed in this article. Finally, we point out that the idea of the three-dimensional graph can be easily extended to model problems involving more complicated distance constraints and which are not easy to model with a single layered graph (in fact, we claim that they cannot be modeled with a single layered graph). As one example consider the problem where the maximum depth of each of the subarborescences depends on the distance of its root from the two end points of the central path. Hence, this concept may turn out to be fruitful for other network design problems as well.

References

1. N.R. Achutan, L. Caccetta, P. Caccetta, and J.F. Geelen. Computational methods for the diameter restricted minimum weight spanning tree problem. *Australasian Journal of Combinatorics*, 10:51–71, 1994.
2. B. V. Cherkassky and A. V. Goldberg. On implementing push-relabel method for the maximum flow problem. *Algorithmica*, 19:390–410, 1994.
3. N. Deo and A. Abdalla. Computing a diameter-constrained minimum spanning tree in parallel. In G.C. Bongiovanni et al., editors, *CIAC*, volume 1767 of *Lecture Notes in Computer Science*, pages 17–31. Springer, 2000.

4. L. Gouveia and T.L. Magnanti. Network flow models for designing diameter-constrained minimum-spanning and Steiner trees. *Networks*, 41 (3):159–173, 2003.
5. L. Gouveia, T.L. Magnanti, and C. Requejo. A 2-path approach for odd-diameter-constrained minimum spanning and Steiner trees. *Networks*, 44 (4):254–265, 2004.
6. L. Gouveia, T.L. Magnanti, and C. Requejo. An intersecting tree model for odd-diameter-constrained minimum spanning and Steiner trees. *Annals OR*, 146(1):19–39, 2006.
7. L. Gouveia, T.L. Magnanti, and C. Requejo. Tight models for special cases of the diameter-constrained minimum spanning tree problem. In *Proceedings of the 3rd International Network Optimization Conference (INOC), Spa*, 2007.
8. L. Gouveia, L. Simonetti, and E. Uchoa. Modeling hop-constrained and diameter-constrained minimum spanning tree problems as Steiner tree problems over layered graphs. *Mathematical Programming*, 128:123–148, 2011.
9. M. Gruber. *Exact and Heuristic Approaches for Solving the Bounded Diameter Minimum Spanning Tree Problem*. PhD thesis, Vienna University of Technology, Vienna, Austria, 2009.
10. M. Gruber and G.R. Raidl. A new 0-1 ILP approach for the bounded diameter minimum spanning tree problem. In *Proceedings of the 2nd International Network Optimization Conference (INOC), Lisbon*, pages 178–185, 2005.
11. M. Gruber and G.R. Raidl. (meta-)heuristic separation of jump cuts in a branch&cut approach for the bounded diameter minimum spanning tree problem. In V. Maniezzo et al., editors, *Matheuristics*, volume 10 of *Annals of Information Systems*, pages 209–230. Springer, 2010.
12. 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 Programming*, 105:427–449, 2006.
13. P. Manyem and M.F.M. Stallmann. Some approximation results in multicasting. Technical Report TR-96-03, North Carolina State University at Raleigh, NC, USA, 1996.
14. T.F. Noronha, C.C. Ribeiro, and A.C. Santos. Solving diameter-constrained minimum spanning tree problems by constraint programming. *International Transactions in Operational Research*, 17(5):653–665, 2010.
15. A.C. Santos, A. Lucena, and C.C. Ribeiro. Solving diameter constrained minimum spanning tree problems in dense graphs. In C.C. Ribeiro and S.L. Martins, editors, *WEA*, volume 3059 of *Lecture Notes in Computer Science*, pages 458–467. Springer, 2004.
16. K.H. Vik, P. Halvorsen, and C. Griwodz. Multicast tree diameter for dynamic distributed interactive applications. In *INFOCOM*, pages 1597–1605. IEEE, 2008.

Appendix

Table 4 Results for solving the LP relaxations on instances with 31 and 41 nodes: Numbers of instances solved within the given time limit ($\#_{\text{solved}}$), geometric means of CPU-times in seconds, and average LP gaps [%] with respect to instance graph and $|P|$. Note that for this set of experiments, all preprocessing and presolving routines of CPLEX as well as its build in additional cuts have been turned off. Average LP gaps are given only over those instances solved by all three models.

			$\#_{\text{solved}}$			CPU-time [s]			Gap [%]		
Inst	$\|P\|$	#	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)
TC30	5	16	**16**	**16**	**16**	41	**33**	198	**0.1**	**0.1**	0.4
TC30	10	16	**16**	**16**	**16**	88	**84**	121	**0.1**	**0.1**	**0.1**
TC30	15	16	**16**	**16**	**16**	320	168	**109**	0.3	**0.1**	**0.1**
TC30	20	16	**16**	**16**	**16**	191	153	**83**	0.5	0.3	**0.0**
TC30	25	16	14	**16**	**16**	636	335	**90**	0.3	0.1	**0.0**
TC30	30	16	**16**	**16**	**16**	275	341	**87**	**0.0**	**0.0**	**0.0**
TC40	5	16	**16**	**16**	**16**	393	**304**	2205	**0.1**	**0.1**	0.5
TC40	10	16	**16**	**16**	**16**	303	**285**	607	**0.1**	**0.1**	0.2
TC40	15	16	**16**	**16**	**16**	832	**433**	497	0.4	**0.1**	0.2
TC40	20	16	15	**16**	**16**	993	486	**417**	0.6	**0.2**	**0.2**
TC40	25	16	15	**16**	**16**	1318	1021	**364**	0.6	**0.2**	**0.2**
TC40	30	16	14	15	**16**	2005	1251	**437**	0.3	**0.1**	**0.1**
TC40	35	16	13	15	**16**	1776	1692	**420**	0.1	**0.0**	0.1
TC40	40	16	15	13	**16**	2154	2518	**617**	**0.0**	**0.0**	**0.0**
TR30	5	16	**16**	**16**	**16**	11	**7**	26	1.1	0.4	**0.3**
TR30	10	16	**16**	**16**	**16**	33	**23**	59	1.2	0.3	**0.2**
TR30	15	16	**16**	**16**	**16**	99	**56**	68	0.2	**0.1**	**0.1**
TR30	20	16	**16**	**16**	**16**	85	**59**	**59**	**0.0**	**0.0**	**0.0**
TR30	25	16	**16**	**16**	**16**	206	181	**71**	0.1	**0.0**	**0.0**
TR30	30	16	**16**	**16**	**16**	201	233	**62**	**0.0**	**0.0**	**0.0**
TR40	5	16	**16**	**16**	**16**	145	**89**	382	0.4	**0.1**	0.4
TR40	10	16	**16**	**16**	**16**	536	**306**	402	1.2	**0.2**	0.5
TR40	15	16	**16**	**16**	**16**	891	**401**	549	0.9	**0.2**	**0.2**
TR40	20	16	14	**16**	**16**	736	640	**423**	0.9	0.2	**0.1**
TR40	25	16	15	14	**16**	931	658	**383**	0.1	**0.0**	**0.0**
TR40	30	16	13	**16**	**16**	2004	1273	**975**	0.3	**0.2**	**0.2**
TR40	35	16	13	14	**16**	2598	1411	**777**	**0.4**	**0.4**	**0.4**
TR40	40	16	12	11	**16**	2718	3211	**633**	**0.5**	**0.5**	**0.5**

Table 5 Numbers of instances solved to optimality ($\#_{\text{solved}}$), geometric means of CPU-times in seconds, and average optimality gaps in [%] with respect to instance graph and $|P|$.

			$\#_{\text{solved}}$			CPU-time [s]			Gap [%]		
Inst	$\|P\|$	#	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)
TC30	5	16	**16**	**16**	**16**	43	**38**	73	**0.0**	**0.0**	**0.0**
TC30	10	16	**16**	**16**	**16**	73	76	**51**	**0.0**	**0.0**	**0.0**
TC30	15	16	**16**	**16**	**16**	183	140	**49**	**0.0**	**0.0**	**0.0**
TC30	20	16	**16**	**16**	**16**	131	104	**24**	**0.0**	**0.0**	**0.0**
TC30	25	16	15	**16**	**16**	213	196	**19**	**0.0**	**0.0**	**0.0**
TC30	30	16	**16**	**16**	**16**	204	299	**29**	**0.0**	**0.0**	**0.0**
TC40	5	16	**16**	**16**	**16**	336	**302**	947	**0.0**	**0.0**	**0.0**
TC40	10	16	**16**	**16**	**16**	249	**249**	301	**0.0**	**0.0**	**0.0**
TC40	15	16	15	**16**	**16**	619	374	**250**	**0.0**	**0.0**	**0.0**
TC40	20	16	14	**16**	**16**	887	480	**207**	**0.0**	**0.0**	**0.0**
TC40	25	16	14	**16**	**16**	1168	629	**145**	**0.0**	**0.0**	**0.0**
TC40	30	16	14	15	**16**	1231	938	**116**	6.3	6.2	**0.0**
TC40	35	16	13	13	**16**	1629	1527	**85**	6.3	6.3	**0.0**
TC40	40	16	15	15	**16**	1871	2553	**121**	**0.0**	**0.0**	**0.0**
TC60	10	16	3	**7**	**7**	7602	**6072**	7216	31.4	18.8	**0.1**
TC60	20	16	10	10	**14**	3929	3038	**2593**	18.8	31.3	**0.0**
TC60	30	16	13	13	**16**	2344	2609	**1778**	6.3	12.5	**0.0**
TC60	40	16	10	9	**16**	2991	3444	**1279**	25.0	31.3	**0.0**
TC60	50	16	5	5	**16**	4492	4995	**899**	43.8	43.8	**0.0**
TC60	60	16	4	4	**16**	5521	5833	**1809**	31.4	43.8	**0.0**
TR30	5	16	**16**	**16**	**16**	8	**5**	13	**0.0**	**0.0**	**0.0**
TR30	10	16	**16**	**16**	**16**	**23**	**23**	**23**	**0.0**	**0.0**	**0.0**
TR30	15	16	**16**	**16**	**16**	49	43	**28**	**0.0**	**0.0**	**0.0**
TR30	20	16	**16**	**16**	**16**	45	53	**10**	**0.0**	**0.0**	**0.0**
TR30	25	16	**16**	**16**	**16**	101	105	**6**	**0.0**	**0.0**	**0.0**
TR30	30	16	**16**	**16**	**16**	122	174	**14**	**0.0**	**0.0**	**0.0**
TR40	5	16	**16**	**16**	**16**	102	**73**	159	**0.0**	**0.0**	**0.0**
TR40	10	16	**16**	**16**	**16**	428	190	**112**	**0.0**	**0.0**	**0.0**
TR40	15	16	15	**16**	**16**	767	407	**139**	**0.0**	**0.0**	**0.0**
TR40	20	16	14	15	**16**	869	621	**121**	**0.0**	6.2	**0.0**
TR40	25	16	15	15	**16**	727	485	**71**	**0.0**	**0.0**	**0.0**
TR40	30	16	14	**16**	**16**	1925	1261	**132**	**0.0**	**0.0**	**0.0**
TR40	35	16	15	14	**16**	1404	1338	**93**	**0.0**	0.1	**0.0**
TR40	40	16	12	9	**16**	2325	3111	**169**	0.1	0.2	**0.0**
TR60	10	16	15	**16**	**16**	770	**617**	1433	0.1	**0.0**	**0.0**
TR60	20	16	15	**16**	**16**	742	**580**	710	0.1	**0.0**	**0.0**
TR60	30	16	16	**16**	**16**	1905	1694	**911**	**0.0**	**0.0**	**0.0**
TR60	40	16	4	6	**16**	8151	7091	**1932**	37.7	31.4	**0.0**
TR60	50	16	7	9	**16**	5502	5083	**624**	18.9	18.9	**0.0**
TR60	60	16	8	5	**16**	5389	8328	**998**	6.5	6.6	**0.0**

Table 6 Numbers of instances solved to optimality ($\#_{\text{solved}}$), geometric means of CPU-times in seconds, and average optimality gaps in [%] with respect to D and D' for Euclidean instances.

				$\#_{\text{solved}}$			CPU-time [s]			Gap [%]		
Inst	D'	D	#	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)
TC30	3	5	6	**6**	**6**	**6**	43	19	**13**	**0.0**	**0.0**	**0.0**
TC30	3	6	6	**6**	**6**	**6**	54	33	**9**	**0.0**	**0.0**	**0.0**
TC30	3	7	6	**6**	**6**	**6**	76	83	**14**	**0.0**	**0.0**	**0.0**
TC30	3	8	6	**6**	**6**	**6**	36	58	**12**	**0.0**	**0.0**	**0.0**
TC30	4	6	6	**6**	**6**	**6**	49	40	**14**	**0.0**	**0.0**	**0.0**
TC30	4	7	6	**6**	**6**	**6**	114	109	**27**	**0.0**	**0.0**	**0.0**
TC30	4	8	6	**6**	**6**	**6**	114	112	**26**	**0.0**	**0.0**	**0.0**
TC30	4	9	6	**6**	**6**	**6**	351	441	**43**	**0.0**	**0.0**	**0.0**
TC30	5	7	6	**6**	**6**	**6**	159	107	**73**	**0.0**	**0.0**	**0.0**
TC30	5	8	6	**6**	**6**	**6**	173	184	**81**	**0.0**	**0.0**	**0.0**
TC30	5	9	6	5	**6**	**6**	469	527	**142**	**0.0**	**0.0**	**0.0**
TC30	5	10	6	**6**	**6**	**6**	470	457	**134**	**0.0**	**0.0**	**0.0**
TC30	6	8	6	**6**	**6**	**6**	63	62	**26**	**0.0**	**0.0**	**0.0**
TC30	6	9	6	**6**	**6**	**6**	113	156	**55**	**0.0**	**0.0**	**0.0**
TC30	6	10	6	**6**	**6**	**6**	126	121	**85**	**0.0**	**0.0**	**0.0**
TC30	6	11	6	**6**	**6**	**6**	287	297	**95**	**0.0**	**0.0**	**0.0**
TC40	3	5	8	7	**8**	**8**	206	63	**22**	**0.0**	**0.0**	**0.0**
TC40	3	6	8	**8**	**8**	**8**	53	63	**15**	**0.0**	**0.0**	**0.0**
TC40	3	7	8	**8**	**8**	**8**	125	142	**60**	**0.0**	**0.0**	**0.0**
TC40	3	8	8	**8**	**8**	**8**	149	183	**38**	**0.0**	**0.0**	**0.0**
TC40	4	6	8	**8**	**8**	**8**	735	384	**115**	**0.0**	**0.0**	**0.0**
TC40	4	7	8	**8**	**8**	**8**	853	568	**204**	**0.0**	**0.0**	**0.0**
TC40	4	8	8	**8**	**8**	**8**	794	804	**192**	**0.0**	**0.0**	**0.0**
TC40	4	9	8	5	7	**8**	2421	1500	**286**	**0.0**	**0.0**	**0.0**
TC40	5	7	8	**8**	**8**	**8**	2249	1416	**643**	**0.0**	**0.0**	**0.0**
TC40	5	8	8	**8**	**8**	**8**	1157	947	**388**	**0.0**	**0.0**	**0.0**
TC40	5	9	8	**8**	**8**	**8**	1785	1879	**735**	**0.0**	**0.0**	**0.0**
TC40	5	10	8	6	**8**	**8**	2686	2329	**677**	0.1	**0.0**	**0.0**
TC40	6	8	8	**8**	**8**	**8**	1016	725	**235**	**0.0**	**0.0**	**0.0**
TC40	6	9	8	7	7	**8**	1954	1363	**514**	**0.0**	**0.0**	**0.0**
TC40	6	10	8	**8**	7	**8**	1702	1697	**490**	**0.0**	12.5	**0.0**
TC40	6	11	8	4	6	**8**	3920	3625	**908**	25.0	12.5	**0.0**
TC60	3	5	6	**6**	**6**	**6**	311	228	**221**	**0.0**	**0.0**	**0.0**
TC60	3	6	6	5	5	**6**	594	653	**112**	0.1	0.1	**0.0**
TC60	3	7	6	**6**	**6**	**6**	608	914	**609**	**0.0**	**0.0**	**0.0**
TC60	3	8	6	**6**	**6**	**6**	672	885	**375**	**0.0**	**0.0**	**0.0**
TC60	4	6	6	3	4	**6**	5406	3938	**1206**	0.1	**0.0**	**0.0**
TC60	4	7	6	3	4	**5**	7494	5734	**2602**	0.2	0.1	**0.0**
TC60	4	8	6	2	2	**5**	8789	8000	**2493**	0.2	0.2	**0.0**
TC60	4	9	6	0	0	**5**	10000	10000	**3601**	0.2	33.5	**0.0**
TC60	5	7	6	3	5	**6**	6820	5211	**4211**	16.7	0.1	**0.0**
TC60	5	8	6	3	3	**5**	6890	8032	**4202**	16.8	33.4	**0.0**
TC60	5	9	6	1	0	**4**	9691	10000	**7374**	50.1	66.7	**0.0**
TC60	5	10	6	2	2	**4**	9742	9835	**7072**	33.4	50.0	**0.1**
TC60	6	8	6	3	3	**6**	7682	7865	**2366**	50.0	50.0	**0.0**
TC60	6	9	6	1	1	**5**	9350	9665	**5542**	83.3	83.3	**0.0**
TC60	6	10	6	1	1	**5**	9438	9787	**5822**	66.7	66.7	**0.0**
TC60	6	11	6	0	0	**5**	10000	10001	**7169**	100.0	100.0	**0.0**

Table 7 Numbers of instances solved to optimality ($\#_{\text{solved}}$), geometric means of CPU-times in seconds, and average optimality gaps in [%] with respect to D and D' for random instances.

				$\#_{\text{solved}}$			CPU-time [s]			Gap [%]		
Inst	D'	D	#	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)	(2T)	$(2T)^+$	(3dLG)
TR30	3	5	6	**6**	**6**	**6**	484	127	**38**	**0.0**	**0.0**	**0.0**
TR30	3	6	6	**6**	**6**	**6**	238	139	**27**	**0.0**	**0.0**	**0.0**
TR30	3	7	6	**6**	**6**	**6**	332	405	**41**	**0.0**	**0.0**	**0.0**
TR30	3	8	6	**6**	**6**	**6**	239	251	**59**	**0.0**	**0.0**	**0.0**
TR30	4	6	6	**6**	**6**	**6**	7	6	**2**	**0.0**	**0.0**	**0.0**
TR30	4	7	6	**6**	**6**	**6**	11	17	**6**	**0.0**	**0.0**	**0.0**
TR30	4	8	6	**6**	**6**	**6**	11	11	**9**	**0.0**	**0.0**	**0.0**
TR30	4	9	6	**6**	**6**	**6**	16	23	**10**	**0.0**	**0.0**	**0.0**
TR30	5	7	6	**6**	**6**	**6**	75	64	**19**	**0.0**	**0.0**	**0.0**
TR30	5	8	6	**6**	**6**	**6**	45	50	**11**	**0.0**	**0.0**	**0.0**
TR30	5	9	6	**6**	**6**	**6**	108	150	**31**	**0.0**	**0.0**	**0.0**
TR30	5	10	6	**6**	**6**	**6**	91	95	**28**	**0.0**	**0.0**	**0.0**
TR30	6	8	6	**6**	**6**	**6**	6	6	**2**	**0.0**	**0.0**	**0.0**
TR30	6	9	6	**6**	**6**	**6**	14	21	**8**	**0.0**	**0.0**	**0.0**
TR30	6	10	6	**6**	**6**	**6**	10	11	**8**	**0.0**	**0.0**	**0.0**
TR30	6	11	6	**6**	**6**	**6**	19	26	**23**	**0.0**	**0.0**	**0.0**
TR40	3	5	8	7	7	**8**	1451	554	**165**	0.1	**0.0**	**0.0**
TR40	3	6	8	7	7	**8**	1064	857	**87**	0.1	0.1	**0.0**
TR40	3	7	8	7	6	**8**	866	984	**213**	0.1	0.1	**0.0**
TR40	3	8	8	6	6	**8**	703	723	**233**	0.1	0.1	**0.0**
TR40	4	6	8	**8**	**8**	**8**	174	136	**28**	**0.0**	**0.0**	**0.0**
TR40	4	7	8	**8**	**8**	**8**	369	165	**63**	**0.0**	**0.0**	**0.0**
TR40	4	8	8	**8**	**8**	**8**	194	155	**65**	**0.0**	**0.0**	**0.0**
TR40	4	9	8	**8**	7	**8**	490	319	**66**	**0.0**	**0.0**	**0.0**
TR40	5	7	8	**8**	**8**	**8**	2963	1398	**380**	**0.0**	**0.0**	**0.0**
TR40	5	8	8	**8**	**8**	**8**	1683	1015	**219**	**0.0**	**0.0**	**0.0**
TR40	5	9	8	6	7	**8**	4167	2964	**493**	0.1	0.1	**0.0**
TR40	5	10	8	7	7	**8**	2759	2821	**355**	0.1	**0.0**	**0.0**
TR40	6	8	8	**8**	**8**	**8**	202	136	**32**	**0.0**	**0.0**	**0.0**
TR40	6	9	8	**8**	**8**	**8**	561	400	**95**	**0.0**	**0.0**	**0.0**
TR40	6	10	8	**8**	**8**	**8**	433	356	**74**	**0.0**	**0.0**	**0.0**
TR40	6	11	8	5	6	**8**	1087	933	**112**	0.1	12.6	**0.0**
TR60	3	5	6	4	4	**6**	3677	3056	**2100**	0.2	0.2	**0.0**
TR60	3	6	6	3	3	**6**	2487	2192	**1255**	0.2	0.3	**0.0**
TR60	3	7	6	3	3	**6**	2898	3533	**2925**	16.9	0.3	**0.0**
TR60	3	8	6	3	3	**6**	3323	4077	**2685**	0.3	0.3	**0.0**
TR60	4	6	6	**6**	**6**	**6**	1002	670	**145**	**0.0**	**0.0**	**0.0**
TR60	4	7	6	**6**	**6**	**6**	1455	1061	**316**	**0.0**	**0.0**	**0.0**
TR60	4	8	6	**6**	5	**6**	722	722	**257**	**0.0**	**0.0**	**0.0**
TR60	4	9	6	**6**	5	**6**	1164	1040	**297**	**0.0**	**0.0**	**0.0**
TR60	5	7	6	2	4	**6**	8864	6153	**2296**	0.4	0.2	**0.0**
TR60	5	8	6	3	5	**6**	5707	3557	**1420**	16.9	0.1	**0.0**
TR60	5	9	6	3	4	**6**	5469	6111	**3309**	0.2	16.8	**0.0**
TR60	5	10	6	3	3	**6**	5308	5989	**3830**	33.5	33.5	**0.0**
TR60	6	8	6	5	**6**	**6**	1821	1747	**306**	16.7	**0.0**	**0.0**
TR60	6	9	6	4	4	**6**	2064	2452	**1083**	33.3	33.3	**0.0**
TR60	6	10	6	4	4	**6**	1693	1655	**727**	33.3	33.3	**0.0**
TR60	6	11	6	4	3	**6**	2815	3637	**1849**	16.7	33.4	**0.0**