A Catalog of Steiner Tree Formulations

Michel X. Goemans*

Department of Mathematics, Room 2-372, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

Young-soo Myung†

Department of Business Administration, Dankook University, Cheonan, Chungnam 330, South Korea

We present some existing and some new formulations for the Steiner tree and Steiner arborescence problems. We show the equivalence of many of these formulations. In particular, we establish the equivalence between the classical bidirected dicut relaxation and two vertex weighted undirected relaxations. The motivation behind this study is a characterization of the feasible region of the dicut relaxation in the natural space corresponding to the Steiner tree problem.

INTRODUCTION

We refer to undirected graphs as *graphs* and to directed graphs as *digraphs*. In a graph $G = (V, E)$, the elements of E are called *edges* and the edge e between the vertices i and j is denoted by $\{i, j\}$ or $\{j, i\}$. In a digraph $D = (V, A)$, the elements of A are called *arcs* and the arc a between i and j is denoted by (i, j). (i, j) and (j, i) do not represent the same arc. From any graph $G = (V, E)$, we can obtain a bidirected graph $D_G = (V, A)$ by replacing every edge of E by two arcs in opposite direction, i.e., $A = \{(i, j) : \{i, j\} \in E\}$.

Given a graph $G = (V, E)$ and a set $T \subseteq V$ of *terminals*, a *Steiner tree* is a tree spanning T. We do not require its leaves to be terminals. Let l be a cost function defined on the edge set E. The Steiner tree problem is the problem of finding a Steiner tree of minimum cost, where the cost of a tree is the sum of the costs of its edges. Given a digraph $D = (V, A)$ and a root vertex r, a set of arcs is called an r-arborescence of D if it forms a (not necessarily spanning) tree directed away from the root r. For a set T of terminals and a specified root vertex $r \in T$, we define a *Steiner arborescence* as an r-arborescence spanning T. The Steiner arborescence problem is the problem of finding a minimum-cost Steiner arborescence. Let $T_r = T\backslash\{r\}$ and $V_r = V\backslash\{r\}$.

The Steiner tree and Steiner arborescence problems have extensively been studied in the literature. Two recent surveys on these Steiner problems have summarized formulations and solution methods [20, 29]. Maculan [20] emphasizes exact algorithms and integer programming formulations, whereas Winter [29] considers exact algorithms, heuristics, and polynomially solvable special cases.

We associate to any Steiner tree an incidence vector x such that $x_e = 1$ if edge $e \in E$ is part of the Steiner tree and 0 otherwise. Let $\mathcal{P}_x$ denote the convex hull of incidence vectors of Steiner trees in a graph G. $\mathcal{P}_x$ is called the Steiner tree polytope. Similarly, the inci-

*Supported by Air Force contract AFOSR-89-0271 and DARPA contract DARPA-89-J-1988.

†On leave at the Operations Research Center, MIT. Partial Support from the Yonam Foundation.

CCC 0028-3045/93/01019-10

dence vector w of a Steiner arborescence B is defined by $w_a = 1$ if $a \in B$ and 0 otherwise. The Steiner arborescence polytope, denoted by $\mathcal{P}_w$, is the convex hull of incidence vectors of Steiner arborescences.

We shall describe linear programming (LP) relaxations of the Steiner tree and Steiner arborescence problems. An LP relaxation for the Steiner tree problem is a linear program of the form

$$\text{Minimize}\left\{z = \sum_{e \in E} l_e x_e : x \in R_x\right\},$$

where R_x is a polyhedral region with $\mathcal{P}_x \subseteq R_x$. More generally, we allow this definition to include *extended* relaxations of the form

$$\text{Minimize}\left\{z = \sum_{e \in E} l_e x_e : (x, s) \in R_{xs}\right\},$$

where $\mathcal{P}_x$ is contained within the projection $\text{proj}_x(R_{xs})$ of R_{xs} onto the x variables defined as $\text{proj}_x(R_{xs}) = \{x : (x, s) \in R_{xs} \text{ for some } s\}$. We regard two relaxations as *equivalent for a class* $\mathcal{L}$ of cost functions $l : E \rightarrow \mathbf{R}$ if their optimal values are equal for any $l \in \mathcal{L}$. Of course, this does not necessarily imply that their feasible regions are equal. However, if two relaxations, defined by R_{xs} and R'_{xt}, are equivalent for *all* cost functions of x, then $\text{proj}_x(R_{xs}) = \text{proj}_x(R'_{xt})$ and vice versa. In this case, we say that R_{xs} and R'_{xt} are extended descriptions of $R = \text{proj}_x(R_{xs}) = \text{proj}_x(R'_{xt})$. These concepts are defined analogously for the Steiner arborescence problem.

The most classical relaxation for the Steiner arborescence problem is the dicut relaxation introduced by Wong [30]. This relaxation can also be used for the Steiner tree problem since any instance of the problem can be equivalently formulated as a bidirected Steiner arborescence problem. Chopra and Rao [6] showed that this approach leads to better relaxations than do simple undirected relaxations for the Steiner tree problem. As a result, there has been little emphasis on undirected relaxations in recent years. In this paper, we show that undirected relaxations can be as tight as bidirected ones, provided that we introduce some auxiliary variables. In particular, by considering vertex variables that either keep track of the vertices spanned or the degrees of the vertices in the Steiner tree, we obtain two undirected relaxations that are equivalent to the bidirected dicut relaxation. These relaxations are valid only for nonnegative cost functions. We also introduce tighter bounded analogs to these relaxations that appear to be equivalent.

The paper is organized as follows: In Section 1, we review classical formulations for the Steiner tree and Steiner arborescence problems and we consider the use of bidirected relaxations for the Steiner tree problem. In Section 2, we introduce two simple extended undirected relaxations involving vertex variables and we prove their equivalence to the bidirected dicut relaxation. Bounded analogs to these relaxations are presented in Section 3. Finally, in Section 4, we show that the polyhedra defined in Section 2 are the dominants of their bounded analogs of Section 4. This implies that all relaxations defined in this paper are equivalent for all nonnegative cost functions. In Section 4, we also prove that the choice of the root vertex is unimportant when bidirecting an undirected instance.

1. A REVIEW OF CLASSICAL INTEGER PROGRAMMING FORMULATIONS

Given a graph $G = (V, E)$ and a set S of vertices, $\delta(S)$ represents the set of edges in E with exactly one endpoint in S, whereas $E(S)$ represents the set of edges in E with both endpoints in S. The corresponding notions for a digraph $D = (V, A)$ are as follows: For a set $S \subseteq V$, $\delta^-(S)$ denotes the set of arcs $\{(i, j) \in A : i \notin S, j \in S\}$, $\delta^+(S) = \delta^-(V \setminus S)$ and $A(S) = \{(i, j) : i \in S, j \in S\}$. For simplicity, we write $\delta^-(i)$ [resp., $\delta^+(i)$ or $\delta(i)$] instead of $\delta^-(\{i\})$ [resp., $\delta^+(\{i\})$ or $\delta(\{i\})$]. If x is defined on the elements of a set M (typically M is an edge set E, an arc set A, or a vertex set V), then we denote $\Sigma_{i \in N} x_i$ for $N \subseteq M$ by $x(N)$. The only exceptions are $\delta(.)$, $\delta^-(.)$, $\delta^+(.)$, $E(.)$, and $A(.)$, which were defined previously.

1.1. Classical Formulations for the Steiner Tree Problem

A Steiner tree can be seen as a minimal‡ subgraph having a path between any pair of terminals. In fact, we can even restrict our attention to pairs containing a specified vertex $r \in T$. This vertex r plays the role of root for the Steiner tree. This definition of Steiner trees in terms of minimal subgraphs can be used to formulate the Steiner tree problem as an integer program when all cost coefficients are nonnegative. For this purpose, we introduce some flow variables and consider the following program [3]:

$$\text{Minimize} \sum_{e \in E} l_e x_e$$

(IP^u_{xf}) subject to:

$$(x, f) \in (S_{xf} \cap (\mathbf{Z}^{|E|} \times \mathbf{R}^{|A \times T_r|}))$$

‡With respect to inclusion.

where

$$S_{xf} = \{(x, f): f^k(\delta^+(i)) - f^k(\delta^-(i)) = \begin{cases} 1 & i = r \\ -1 & i = k \\ 0 & i \in V\backslash\{k, r\} \end{cases} \text{ and } k \in T_r \quad (1)$$

$$f_{ij}^k \le x_e \qquad e = \{i, j\} \in E \text{ and } k \in T_r \quad (2)$$

$$f_a^k \ge 0 \qquad a \in A \text{ and } k \in T_r\},$$

and $D = (V, A)$ is the bidirected graph obtained from $G = (V, E)$ by bidirecting every edge of E. The constraints (1) imply the existence of a unit flow from r to k, and if x_e is integral, this means that there exists a path from r to k in the subgraph $\{e \in E: x_e \ge 1\}$.

Using the max-flow min-cut theorem, the projection S_x of S_{xf} onto the x variables can be characterized as

$$S_x = \{x: x(\delta(S)) \ge 1 \qquad r \notin S \text{ and } S \cap T \neq \emptyset \quad (3)$$
$$x_e \ge 0 \qquad e \in E\}$$

and $S_x \supseteq \mathcal{P}_x$. For a set S with $r \notin S$ and $S \cap T \neq \emptyset$, the set of edges of the form $\delta(S)$ constitutes a so-called Steiner cut, and, as a result, the inequalities (3) are known as Steiner cut inequalities. (IP_{xf}^u) can thus be reformulated by the classical *cut formulation* [1]:

$$\text{Minimize } \sum_{e \in E} l_e x_e$$

(IP_x^u) subject to:

$$x \in (S_x \cap \mathbf{Z}^{|E|}).$$

Moreover, the fact that $S_x = \text{proj}_x(S_{xf})$ implies that the linear programming (LP) relaxations of (IP_{xf}^u) and (IP_x^u) obtained by relaxing the integrality constraints on all x_e's are equivalent. Notice that (IP_x^u) is a natural formulation for the Steiner tree problem that has exponential size, whereas (IP_{xf}^u) is an extended formulation that is compact (namely, it has a polynomial number of constraints and a polynomial number of variables). Therefore, the value of their LP relaxations (LP_x^u) and (LP_{xf}^u) can be computed in polynomial time either using an interior-point algorithm on the polynomial-sized formulation (LP_{xf}^u) or the ellipsoid algorithm on (LP_x^u) since the separation problem over the Steiner cut inequalities can be solved in polynomial time as a sequence of $|T| - 1$ maximum flow problems (one for each $k \in T_r$).

Goemans and Bertsimas [14] show that, if the cost function satisfies the triangle inequality, the linear program (LP_x^u) can be simplified considerably without affecting its optimal value. As a result of their study, the value of (LP_x^u) can be computed à la Held and Karp [15] by solving a sequence of minimum spanning tree problems with Lagrangean costs [14].

In some cases, S_x is integral, i.e., it is equal to its integer hull $int\_hull(S_x)$ defined as the convex hull $conv(S_x \cap \mathbf{Z}^{|E|})$ of its integer points. For example, this happens when $|T| = 2$ or when G is acyclic. The case $|T| = 2$ corresponds to the shortest path problem in an undirected graph with no negative cycles. However, even for the spanning tree problem on a cycle ($T = V$ and G is a cycle), S_x is not equal to $int\_hull(S_x)$. It would be natural to expect a complete characterization for this very simple case. Although (LP_x^u) appears to be a fairly loose relaxation, its value has been shown to be within a factor of $1/(2 - 2/|T|)$ of the optimal value of the Steiner tree problem [14].

The Steiner tree problem on a graph $G = (V, E)$ can be transformed into a Steiner arborescence problem by considering the digraph $D_G = (V, A)$ obtained by bidirecting every edge of G, choosing arbitrarily a root vertex $r \in T$ and defining the cost of the arc $(i, j) \in A$ by $c_{ij} = l_e$ where $e = \{i, j\}$. This approach leads to much better formulations for the Steiner tree problem. For this reason, undirected relaxations have recently been given much less attention than have bidirected relaxations.

1.2. Classical Formulations for the Steiner Arborescence Problem

The Steiner arborescence problem can be formulated in a similar way as is the Steiner tree problem. A Steiner arborescence can be seen as a minimal digraph having directed paths between the root r and any other terminal. Therefore, when the cost function is nonnegative, the Steiner arborescence problem can be formulated by the following integer program [30], known as the multicommodity flow formulation:

$$\text{Minimize } \sum_{a \in A} c_a w_a$$

(IP_{wf}) subject to:

$$(w, f) \in (Q_{wf} \cap (\mathbf{Z}^{|A|} \times \mathbf{R}^{|A \times T_r|}))$$

where

$$Q_{wf} = \{(w,f): f^k(\delta^+(i)) - f^k(\delta^-(i)) = \begin{cases} 1 & i = r \\ -1 & i = k \\ 0 & i \in V\backslash\{k, r\} \end{cases} \text{ and } k \in T_r$$

$$f_a^k \leq w_a \qquad a \in A \text{ and } k \in T_r$$

$$f_a^k \geq 0 \qquad a \in A \text{ and } k \in T_r\}.$$

Since no Steiner arborescence contains any arc incoming to the root, we shall assume that A has no such arc. Equivalently, we could assume that $w_a = 0$ for $a \in \delta^-(r)$. This assumption is made throughout the paper and turns out to be useful when dealing with bidirected graphs.

Again, by the max-flow min-cut theorem, the projection of Q_{wf} onto the w variables can be expressed as [20]:

$$Q_w = \{w: w(\delta^-(S)) \geq 1 \qquad r \notin S \text{ and } S \cap T \neq \emptyset \quad (4)$$

$$w_a \geq 0 \qquad a \in A\}, \quad (5)$$

with $\mathcal{P}_w \subseteq Q_w$. The inequalities (4) are known as Steiner dicut inequalities. This leads to the classical *dicut* formulation for the Steiner arborescence problem:

$$\text{Minimize } \sum_{a \in A} c_a w_a$$

(IP_w) subject to:

$$w \in (Q_w \cap \mathbf{Z}^{|A|}).$$

Relaxing the integrality on w, we obtain the linear programming relaxations (LP_{wf}) and (LP_w). These LP relaxations are equivalent and, by the same argument as for the Steiner tree relaxations, their common optimal value can be computed in polynomial time. Wong [30] proposed a dual ascent method to obtain good approximations on this optimal value.

The polyhedron Q_w is equal to its integer hull when $|T| = 2$, when $T = V$ [10] or when the underlying graph is series-parallel ([26], see also [13, 17, 28] for a slight generalization). The case $T = V$ corresponds to the minimum-cost arborescence problem.

Tighter relaxations for the Steiner arborescence problem have been proposed. Chopra and Rao [7] derived classes of facet-defining valid inequalities that can be used to strengthen Q_w. Myung [24] also derived a class of facet-defining inequalities with 0-1 coefficients for the set covering formulation of the problem. Liu [18] gave a formulation for the Steiner arborescence problem based on an extended complete characterization of its polyhedron when $|T| = 3$.

1.3. The Bidirected Case

As we have previously mentioned, the Steiner tree problem is equivalent to the bidirected Steiner arborescence problem. Therefore, any formulation or relaxation for the Steiner arborescence problem can be used for the Steiner tree problem. This gives formulations not in the space of the edge variables x but in the space of the arc variables w (and possibly some additional variables). To obtain a formulation in the natural set of variables, we need to use the linear transformation $x_e = w_{ij} + w_{ji}$ for all $e = \{i, j\}$ and eliminate the w variables, i.e., project onto the x variables. In particular, for Q_w, we first define $Q_{xw} = \{(x, w): w \in Q_w$ and $x_e = w_{ij} + w_{ji}$ for all $e = \{i, j\} \in E\}$ and then consider $Q_x = \text{proj}_x(Q_{xw})$. Clearly, $\mathcal{P}_x \subseteq Q_x$ and $Q_x \subseteq S_x$, since, for $x \in Q_x$, we have $x(\delta(S)) = w(\delta^-(S)) + w(\delta^+(S)) \geq 1$ by (4) and (5). Furthermore, from the results for Q_w, we know that $Q_x = int\_hull(Q_x)$ if $|T| = 2$, $T = V$, or G is series-parallel.

Little is known about Q_x. Although Q_x seems to depend on the choice of the root vertex r, we shall prove at the end of this paper that, in fact, it is independent of r. To the best of our knowledge, this fact, although believed by many authors, has not been established before. On the other hand, a complete description of Q_x by linear inequalities is unknown. Chopra and Rao [6] described some of the linear inequalities defining Q_x, namely, the class of Steiner partition inequalities and the class of odd hole inequalities. Goemans [12] obtained many more such inequalities and showed that Q_x has a very rich and complicated structure. There is therefore no hope of obtaining a simple description of Q_x by linear inequalities. This has motivated many researchers to focus their attention on bidirected formulations. Our main goal in this paper is to show that simple *undirected* formulations can be as tight as bidirected formulations provided that auxiliary variables are allowed. For this purpose, we shall present three simple extended descriptions of Q_x. Two of them are new and presented in the next section. The last one is obtained by "undirectizing" Q_{wf}.

For this purpose, consider

$$Q_{xf} = \{(x, f): f^k(\delta^+(i)) - f^k(\delta^-(i)) = \begin{cases} 1 & i = r \\ -1 & i = k \\ 0 & i \in V\backslash\{k, r\} \end{cases} \quad \text{and } k \in T_r$$

$$f^h_{ij} + f^k_{ji} \le x_e \qquad e = \{i, j\} \in E \text{ and } h, k \in T_r \tag{6}$$

$$f^k_a \ge 0 \qquad a \in A \text{ and } k \in T_r\}.$$

The difference between Q_{xf} and S_{xf} resides in the constraints (6). These constraints couple the flow corresponding to two commodities on arcs of opposite direction. This technique to strengthen multicommodity flow formulations has been used by Martin [21] and Balakrishnan et al. [2].

Q_{xf} precisely constitutes an extended description of Q_x.

Proposition 1. $Q_{xf} = \{(x, f)$: *There exists* $(w, f) \in Q_{wf}$ *with* $x_e = w_{ij} + w_{ji}$ *for all* $e = \{i, j\} \in E\}$.

Proof. If $(w, f) \in Q_{wf}$ and $x_e = w_{ij} + w_{ji}$ for all $e = \{i, j\} \in E$, then $(x, f) \in Q_{xf}$ since, by (2), $f^h_{ij} + f^k_{ji} \le w_{ij} + w_{ji} = x_e$ for $e = \{i, j\} \in E$.

On the other hand, assume that $(x, f) \in Q_{xf}$. By (6), $\max_h f^h_{ij} + \max_k f^k_{ji} \le x_e$ for $e = \{i, j\} \in E$. Hence, we can choose w such that $\max_h f^h_{ij} \le w_{ij}$ and $x_e = w_{ij} + w_{ji}$ for all $e = \{i, j\} \in E$ [e.g., take $w_{ij} = \frac{1}{2}(x_e + \max_h f^h_{ij} - \max_k f^k_{ji})$]. Clearly, $(w, f) \in Q_{wf}$. ∎

Since Q_x is integral when $|T| = 2$, $T = V$, or when G is series-parallel,[§] Q_{xf} gives a compact description of the dominant of the Steiner tree polytope for these cases. In particular, when $T = V$, we have a compact extended description of the dominant of the spanning tree polytope by a system of linear inequalities.[‖] A description of this dominant in the space of the x variables is given by Fulkerson [9] (see also [4]):

$$\{x: x(\delta(V_1, \ldots, V_k)) \ge k - 1 \qquad (V_1, \ldots, V_k)$$

$$x_e \ge 0 \qquad e \in E\}$$

where $(V_1, \ldots, V_k)$ is any partition of V and $\delta(V_1, \ldots, V_k)$ denotes the set of edges whose endpoints belong to different members of the partition.

Whether a linear program can be expressed in compact way is an important question (see [31]). Two general techniques have been proposed to derive compact descriptions. In [22], Martin derives polynomial-sized formulations from separation algorithms. Most of the compact formulations presented in this paper can be obtained in this way. In particular, Martin gave another compact description of the spanning tree polytope. The other technique is based upon dynamic programming algorithms for the associated combinatorial optimization problems [23]. For example, a compact extended description of the Steiner tree polytope for series-parallel graphs can be obtained from a dynamic programming algorithm based upon the decomposition of these graphs [23].

2. TWO OTHER EXTENDED FORMULATIONS FOR Q_x

In this section, we show that simple descriptions of Q_x can be obtained by introducing some vertex variables. We present two such descriptions.

The first description is obtained by keeping track of which vertices are spanned by the Steiner tree. Consider additional variables y_i for $i \in N = V\backslash T$ (N is the set of *Steiner vertices*) with the meaning that $y_i = 1$ if vertex i is spanned by the Steiner tree and 0 otherwise. If we know which vertices are spanned by the Steiner tree, then we can use Edmonds' complete description [8] of the spanning tree polytope to obtain a (partial) description of the Steiner tree polytope:

$$P'_{xy} = \{(x, y) \in \mathbf{R}^{|E|+|N|}: x(E) = y(N) + |T| - 1 \tag{7}$$

$$x(E(S)) \le y(N \cap S) + |T \cap S| - 1 \qquad S \cap T \ne \emptyset \tag{8}$$

$$x(E(S)) \le y(S - k) \qquad k \in S \subseteq N \tag{9}$$

$$x_e \ge 0 \qquad e \in E \tag{10}$$

$$y_k \le 1 \qquad k \in N\} \tag{11}$$

where $S - k = S\backslash\{k\}$. Constraints (8) and (9) are called *generalized subtour elimination* constraints. These constraints together with (7) and (10) enforce that, whenever $y_k \in \{0, 1\}$ for all $k \in N$, x is a convex

[§] However Q_{xf} itself is not integral [21].

[‖] In fact, Martin [21] showed that if we add the constraint $x(E) = |V| - 1$ to Q_{xf}, we obtain an extended description of the spanning tree polytope. This is a slightly weaker result.

combination of incidence vectors of trees spanning $T \cup \{k \in N: y_k = 1\}$.

Relaxations based on P'_{xy} or on similar polyhedra have been considered by Lucena [19] and Goemans [12]. The separation problem for the constraints (8) and (9) can be solved by a sequence of $|V|$ minimum cut problems (see [27] or Section III.3.7. in [25]). Therefore, optimizing over P'_{xy} can be done in polynomial time. P'_{xy} is integral when $T = V$ [8] or when G is series-parallel [12].

To relate this polytope to Q_x, we consider a relaxation of P'_{xy}. It involves the same root vertex r as used in the definition of Q_w (although we shall show later on that Q_x is independent of r). Let

$$
\begin{aligned}
P_{xy} = \{(x, y) \in \mathbf{R}^{|E|+|V|}: x(E) &= y(V_r) && && (12)\\
x(E(S)) &\le y(S) - 1 && S \cap T \neq \emptyset && (13)\\
x_e &\ge 0 && e \in E && (14)\\
y_k &\ge 1 && k \in T_r && (15)\\
y_r &= 1\}. && && (16)
\end{aligned}
$$

In the next theorem, we show that P_{xy} constitutes an extended description of Q_x.

Theorem 2. $proj_x(P_{xy}) = Q_x$.

Proof. Let P_x denote $\text{proj}_x(P_{xy})$. We first show that $Q_x \subseteq P_x$. Let $x \in Q_x$ and $w \in Q_w$ be such that $x_e = w_{ij} + w_{ji}$ for $e = \{i, j\} \in E$. Define $y_i = w(\delta^-(i))$ for $i \in V_r$ and $y_r = 1$. We claim that $(x, y) \in P_{xy}$, implying that $x \in P_x$. First, (x, y) satisfies (12) since $x(E) = w(A) = \sum_{i \in V_r} w(\delta^-(i)) = y(V_r)$. To show that (x, y) satisfies (13) for $S \cap T \neq \emptyset$, we consider two cases. If $r \notin S$, then

$$
\begin{aligned}
x(E(S)) &= w(A(S))\\
&= \sum_{i \in S} w(\delta^-(i)) - w(\delta^-(S))\\
&= y(S) - w(\delta^-(S))\\
&\overset{(4)}{\le} y(S) - 1.
\end{aligned}
$$

If $r \in S$, then

$$
\begin{aligned}
x(E(S)) &= w(A(S))\\
&= \sum_{i \in S \setminus \{r\}} w(\delta^-(i)) - w(\delta^-(S))\\
&= y(S \setminus \{r\}) - w(\delta^-(S))\\
&\le y(S \setminus \{r\}) = y(S) - 1.
\end{aligned}
$$

Finally, it is obvious that (x, y) satisfies (14), (15), and (16).

To show that $P_x \subseteq Q_x$, consider an $(x, y) \in P_{xy}$. We would like to prove the existence of a vector $w \in Q_w$ with $x_e = w_{ij} + w_{ji}$ for all $\{i, j\} \in E$. Given $(x, y) \in P_{xy}$, define

$$
\begin{aligned}
P_w^{xy} = \{w: w_{ij} + w_{ji} &= x_e && e = (i, j) \in E\\
w(\delta^-(i)) &= y_i && i \in V_r\\
w_a &\ge 0 && a \in A\},
\end{aligned}
$$

where, as usually, $D_G = (V, A)$ denotes the bidirected graph corresponding to $G = (V, E)$ in which the arcs incoming to r have been removed.

Claim 1. $P_w^{xy} \subseteq Q_w$.

Let $w \in P_w^{xy}$. By definition, w satisfies all nonnegativity constraints (5).

Consider now a set S such that $r \notin S$ and $S \cap T \neq \emptyset$. We have

$$
\begin{aligned}
w(\delta^-(S)) &= \sum_{i \in S} w(\delta^-(i)) - w(A(S))\\
&= y(S) - x(E(S))\\
&\overset{(13)}{\ge} 1,
\end{aligned}
$$

proving that w also satisfies (4). ◇

Claim 2. $P_w^{xy} \neq \emptyset$.

P_w^{xy} can be interpreted as the set of feasible flows w in a transportation network (N, L). In this network, the set N of nodes consists of E and V_r, the set L of arcs is $\{(e, j): e = \{i, j\} \in E\}$, the supply at node $e \in E$ is x_e, the demand at node $i \in V_r$ is y_i and w_{ij} denotes the flow on the arc (e, j), where $e = \{i, j\}$. Although the arcs in L are uncapacitated, we can view this transportation problem as a capacitated transportation problem on a complete bipartite network with capacities either zero or infinity. Such a transportation problem is feasible iff, for any set M of nodes with no arc in L leaving M, the total supply in M is less or equal to the total demand in M. This version of the max-flow min-cut theorem follows from Gale's characterization [11] of the feasible capacitated transportation networks. In our case, the set M of nodes must satisfy $(M \cap E) \subseteq E((M \cap V_r) \cup \{r\})$ and the condition on the supplies and demands is $x(M \cap E) \le y(M \cap V_r)$. This latter condition is clearly satisfied for any such M since

$$x(M \cap E) \le x(E((M \cap V_r) \cup \{r\})) \overset{(13)}{\le} y((M \cap V_r) \cup \{r\}) - 1 = y(M \cap V_r).$$

Therefore, P_w^{xy} is nonempty. ◇

This completes the proof of the theorem. ■

The other description is obtained by introducing vertex variables $z_i (i \in V)$ whose values are functions of the degree d_i of vertex i in the Steiner tree. Consider the polyhedron

$$R'_{xz} = \{(x, z):\ z(V) = 2 \tag{17}$$

$$x(\delta(S)) + z(S) \ge \begin{cases} 2 & S \cap T \ne \emptyset \\ x(\delta(k)) + z_k & k \in S \subseteq N \end{cases} \tag{18}$$

$$x(\delta(k)) + z_k \le 2 \qquad k \in V \tag{19}$$

$$x_e \ge 0 \qquad e \in E\}. \tag{20}$$

The variables z_k for $k \in T$ can be easily eliminated from this formulation since, from (18) for $S = \{k\}$ and (19), we derive that $z_k = 2 - x(\delta(k))$ for $k \in T$. If x is the incidence vector of a Steiner tree whose degree at vertex i is d_i, then $(x, z) \in R'_{xz}$ for $z_i = 0$ if $d_i = 0$ (vertex i is not spanned) and $z_i = 2 - d_i$ if $d_i \ge 1$. To verify that (x, z) satisfies (18), it suffices to realize that $x(\delta(S)) + z(S)$ is equal to twice the number of connected components of the forest induced by the vertices in S spanned by the Steiner tree.

To relate R'_{xz} to Q_x, we consider a relaxation of R'_{xz} involving the root vertex r:

$$R_{xz} = \{(x, z):\ z(V) = 2 \tag{21}$$

$$x(\delta(S)) + z(S) \ge 2 \qquad S \cap T \ne \emptyset \tag{22}$$

$$x(\delta(r)) + z_r = 2 \tag{23}$$

$$x_e \ge 0 \qquad e \in E\}. \tag{24}$$

Juenger and Pulleyblank [16] showed that R_{xz} constitutes an extended description of the dominant of the spanning tree polytope when $T = V$.

Theorem 3. *$proj_x(R_{xz}) = Q_x$.*

Proof. Using Theorem 2, we shall instead show that $P_x = R_x$, where $P_x = \text{proj}_x(P_{xy})$ and $R_x = \text{proj}_x(R_{xz})$.

We first show that $P_x \subseteq R_x$. Let $(x, y) \in P_{xy}$ and define $z_i = 2y_i - x(\delta(i))$ for $i \in V$. We claim that $(x, z) \in R_{xz}$. First, (x, z) satisfies $z(V) = 2y(V) - 2x(E) \overset{(12)}{=} 2y_r \overset{(16)}{=} 2$ and, hence, (21) holds. For a set S with $S \cap T \ne \emptyset$, we have

$$x(\delta(S)) + z(S) = x(\delta(S)) + 2y(S) - x(\delta(S)) - 2x(E(S)) \overset{(13)}{\ge} 2.$$

Thus, (x, z) satisfies (22). Since (23) and (24) are also satisfied, $(x, z) \in R_{xz}$.

To show that $R_x \subseteq P_x$, consider an $(x, z) \in R_{xz}$ and define $y_i = \frac{1}{2}(x(\delta(i)) + z_i)$ for $i \in V$. We claim that $(x, y) \in P_{xy}$. (12) holds since

$$y(V_r) \overset{(23)}{=} y(V) - 1 = x(E) + \tfrac{1}{2}z(V) - 1 \overset{(21)}{=} x(E).$$

(x, y) satisfies also (13):

$$y(S) - 1 = \tfrac{1}{2}x(\delta(S)) + x(E(S)) + \tfrac{1}{2}z(S) - 1 \overset{(22)}{\ge} x(E(S)).$$

Moreover, (15) follows from (22) for $S = \{k\}$ and (16) follows from (23). Therefore, $(x, y) \in P_{xy}$. This completes the proof of the theorem. ■

From this relationship between R_{xz} and P_{xy}, a polynomial time separation algorithm for the constraints (22) can be easily derived.

We have presented four extended descriptions of Q_x, namely, Q_{xw}, Q_{xf}, P_{xy}, and R_{xz}. The latter two have the advantage of requiring just $|V|$ additional variables and this makes them more attractive to be used in a cutting plane algorithm.

3. TIGHTER RELAXATIONS

Both Q_x and Q_w are of blocking type, i.e., their recession cones (or characteristic cones) consist of all nonnegative vectors. As a result, all relaxations mentioned in the previous sections are useful only for nonnegative cost functions. In this section, we are interested in bounded analogs to Q_x and Q_w. More precisely, we shall describe polyhedra Q'_x and Q'_w, which are, respectively, contained within Q_x and Q_w and whose integer members are the incidence vectors of Steiner trees and Steiner arborescences [i.e., *int_hull*$(Q'_x) = \mathcal{P}_x$ and *int_hull*$(Q'_w) = \mathcal{P}_w$].

Q'_x is at least as complex as Q_x. We should therefore not expect a simple description of it in the space of the x variables. However, as for Q_x, we derive simple extended descriptions of Q'_x. Two of these descriptions appear to be the polyhedra P'_{xy} and R'_{xz} introduced in the previous section.

Theorem 4. *$proj_x(P'_{xy}) = proj_x(R'_{xz})$.*

Proof. The proof is almost identical to the proof of Theorem 3. Let $P'_x = \text{proj}_x(P'_{xy})$ and $R'_x = \text{proj}_x(R'_{xz})$.

Let $(x, y) \in P'_{xy}$. Define $z_i = 2y_i - x(\delta(i))$ for $i \in V$, where, by convention, $y_i = 1$ if $i \in T$. We already know that (x, z) satisfies (21)–(24), i.e., (17), (18) for $S \cap T \neq \emptyset$, and (20). (18) for $k \in S \subseteq N$ also holds since

$$\begin{aligned} &x(\delta(S)) + z(S - k) \\ &= x(\delta(S)) + 2y(S - k) - x(\delta(S - k)) - 2x(E(S - k)) \\ &\overset{(9)}{\geq} x(\delta(S)) + 2x(E(S)) - x(\delta(S - k)) - 2x(E(S - k)) \\ &= x(\delta(k)). \end{aligned}$$

Moreover, from (11), (x, z) satisfies (19) and, hence, belongs to R'_{xz}.

On the other hand, consider an $(x, z) \in R'_{xz}$ and define $y_i = \frac{1}{2}(x(\delta(i)) + z_i)$ for $i \in V$. We already know that (x, y) satisfies (12)–(16). From (19), $y_k \leq 1$ and, hence, (x, y) satisfies (11) and $y_k = 1$ for $k \in T$. To show that $(x, y) \in P'_{xy}$ (where y is restricted to its components in N), we simply need to check (9) for $k \in S \subseteq N$:

$$\begin{aligned} y(S - k) &= x(E(S - k)) + \tfrac{1}{2}x(\delta(S - k)) + \tfrac{1}{2}z(S - k) \\ &\overset{(18)}{\geq} x(E(S - k)) + \tfrac{1}{2}x(\delta(S - k)) + \tfrac{1}{2}x(\delta(k)) \\ &\quad - \tfrac{1}{2}x(\delta(S)) = x(E(S)). \qquad \blacksquare \end{aligned}$$

Motivated by the results of Section 2, we would like to characterize $\text{proj}_x(P'_{xy})$ or $\text{proj}_x(R'_{xz})$ in the space of the w variables. In the following Theorem 5, we show that, in this space, this polytope takes an especially attractive form:

$$\begin{aligned} Q'_w = \{w:\ & w(\delta^-(S)) \geq 1 && r \notin S \text{ and } S \cap T \neq \emptyset && (25) \\ & w(\delta^-(S)) \geq w(\delta^-(k)) && k \in S \subseteq N && (26) \\ & w(\delta^-(k)) \leq 1 && k \in V_r && (27) \\ & w_a \geq 0 && a \in A\}. && (28) \end{aligned}$$

Constraints (26) can be interpreted as saying that the maximum flow from r to k in the network with capacity w_a on arc $a \in A$ has value precisely $w(\delta^-(k))$. Moreover, constraints (27) imply that this value is at most 1. Optimizing over Q'_w can be done in polynomial time since the separation problem over (25) and (26) can be solved by a sequence of $|V| - 1$ maximum flow problems. When the underlying path graph is series-parallel, Q'_w is precisely the Steiner arborescence polytope $\mathcal{P}_w$ [13].

Theorem 5. *$P'_x = Q'_x$, where $P'_x = proj_x(P'_{xy})$ and $Q'_x = \{x : x_e = w_{ij} + w_{ji}$ for all $e = \{i, j\} \in E$ for some $w \in Q'_w\}$.*

Proof. The proof is very similar to the proof of Theorem 2 and is therefore just outlined. First, we consider a vector $w \in Q'_w$ and we let $x_e = w_{ij} + w_{ji}$ for $e = \{i, j\} \in E$. Define $y_i = w(\delta^-(i))$ for $i \in N$. It is easy to verify that $(x, y) \in P'_{xy}$. This proves that $Q'_x \subseteq P'_x$.

To show that $P'_x \subseteq Q'_x$, we consider an $(x, y) \in P'_{xy}$

and we define

$$
\begin{aligned}
P_w'^{xy} = \{w\colon\ & w_{ij} + w_{ji} = x_e && e = \{i, j\} \in E \\
& w(\delta^-(i)) = y_i && i \in N \\
& w(\delta^-(i)) = 1 && i \in T_r \\
& w_a \ge 0 && a \in A\}.
\end{aligned}
$$

By the same technique as in the proof of Theorem 2, it can be shown that $P_w'^{xy} \subseteq Q_w'$ and that $P_w'^{xy} \neq \emptyset$. This completes the proof of the theorem. ∎

Since P'_{xy} (or R'_{xz}) is independent of the vertex r, so is Q'_x. Without extended undirected descriptions of Q'_x, this might have been difficult to prove because Q'_w clearly depends on the vertex r.

For the Steiner tree problem, we have seen that the introduction of vertex variables y improves the formulation. This is typically not the case for Steiner arborescence formulations since y_k can be easily eliminated as it must be equal to $w(\delta^-(k))$. For example, introducing vertex variables y in Q'_w, we obtain

$$
\begin{aligned}
Q'_{wy} = \{(w, y)\colon\ & w(\delta^-(S)) \ge 1 && r \notin S \text{ and } S \cap T \neq \emptyset \\
& w(\delta^-(S)) \ge y_k && k \in S \subseteq N \\
& y_k = w(\delta^-(k)) && k \in V_r \\
& y_k \le 1 && k \in V_r \\
& w_a \ge 0 && a \in A\}.
\end{aligned}
$$

This formulation for the vertex weighted Steiner arborescence problem is considered in Chopra and Gorres [5]. By the same argument as in Theorem 5, we have the $P'_{xy} = \{(x, y) : x_e = w_{ij} + w_{ji} \text{ for all } e = \{i, j\} \in E \text{ and some } (w, y) \in Q'_{wy}\}$. This contradicts the belief (see, e.g., [5]) that vertex weighted bidirected relaxations are tighter than their undirected counterparts.

4. Q_w IS THE DOMINANT OF Q'_w

Although $Q_w \neq Q'_w$, it appears that Q_w and Q'_w lead to relaxations that are equivalent for all *nonnegative* cost functions. This follows from the following theorem:

Theorem 6. *Q_w is equal to the dominant $dom(Q'_w)$ of Q'_w, where $dom(Q'_w) = \{w : w \ge w' \text{ with } w' \in Q'_w\}$.*

Proof. If $w' \in Q'_w$ and $w \ge w'$, then, clearly, $w \in Q_w$. This proves that $dom(Q'_w) \subseteq Q_w$.

To prove that $Q_w \subseteq dom(Q'_w)$, we show that the minimal members in Q_w belong to Q'_w. Consider a $w \in Q_w$ such that $\bar{w} \notin Q_w$ for all $\bar{w} \le w$, $\bar{w} \neq w$. Clearly, w satisfies (25) and (28).

Suppose that w violates the inequality (26) for some S and k. As it will be useful later, we do not assume that $S \subseteq N$. Among all such inequalities, choose the one for which $|S|$ is minimal. If $w_a = 0$ for all $a \in (\delta^-(k)\backslash\delta^-(S))$, then $w(\delta^-(k)) = w(\delta^-(S))$ and this is a contradiction. Let $a = (i, k) \in (\delta^-(k)\backslash\delta^-(S))$ (i.e., $i \in S$) with $w_a > 0$. Since w_a cannot be decreased without violating one of the constraints defining Q_w, there exists R with $r \notin R$, $R \cap T \neq \emptyset$, $a \in \delta^-(R)$, and $w(\delta^-(R)) = 1$. By submodularity of $w(\delta(.))$, we have $w(\delta^-(S)) + w(\delta^-(R)) \ge w(\delta^-(S \cup R)) + w(\delta^-(S \cap R))$. Since $r \notin (S \cup R)$ and $(S \cup R) \cap T \neq \emptyset$, (4) says that $w(\delta^-(S \cup R)) \ge 1 = w(\delta^-(R))$. Therefore, $w(\delta^-(S)) \ge w(\delta^-(S \cap R))$. This implies that w also violates (26) for $S \cap R$ and k. Since $i \in S\backslash R$, we have $|S \cap R| < |S|$ and this contradicts the minimality of S.

Suppose now that w violates the inequality (27) for $k \in V_r$. Let $a = (i, k) \in \delta^-(k)$ with $w_a > 0$. Since w_a cannot be decreased, there exists S with $r \notin S$, $S \cap T \neq \emptyset$, $a \in \delta^-(S)$ and $w(\delta^-(S)) = 1$. Hence, $w(\delta^-(S)) < w(\delta^-(k))$. By the above argument, this gives a contradiction. ∎

Corollary 7. $Q_x = dom(Q'_x)$.

Theorem 6 and Corollary 7 imply that most of the relaxations considered in this paper are equivalent for all nonnegative cost functions.

Corollary 8. *The LP relaxations obtained by optimizing over the following polyhedra are all equivalent for all nonnegative cost functions*:

$$Q_w,\ Q_{wf},\ Q_x,\ Q_{xf},\ P_{xy},\ R_{xz},\ Q'_w,\ P'_{xy},\ \textit{and}\ R'_{xz}.$$

From Corollary 7 and the fact that Q'_x is independent of r, we obtain the following previously mentioned result:

Theorem 9. Q_x is independent of the root r.

Many thanks to Laurence Wolsey and Bill Pulleyblank for describing relaxations closely related to P'_{xy} and R'_{xz}.

REFERENCES

[1] Y. P. Aneja, An integer linear programming approach to the Steiner problem in graphs. *Networks* **10** (1980) 167–178.

[2] A. Balakrishnan, T. L. Magnanti, and R. T. Wong, A dual-ascent procedure for large-scale uncapacitated network design. *Operations Res.* **37** (1989) 716–740.

[3] J. E. Beasley, An algorithm for the Steiner problem in graphs. *Networks* **14** (1984) 147–159.

[4] S. Chopra, On the spanning tree polyhedron. Working Paper, New York University (1988).

[5] S. Chopra and E. Gorres, On the node weighted Steiner tree problem. Working Paper, New York University (1990).

[6] S. Chopra and M. R. Rao, The Steiner tree problem. I: Formulations, compositions and extensions of facets. Working Paper, New York University (1988).

[7] S. Chopra and M. R. Rao, The Steiner tree problem. II: Properties and classes of facets. Working Paper, New York University (1988).

[8] J. Edmonds, Submodular functions, matroids and certain polyhedra, *Combinatorial Structures and Their Applications* (R. Guy et al., Eds.). Gordon and Breach, New York (1970).

[9] D. R. Fulkerson, Blocking and anti-blocking pairs of polyhedra. *Math. Programming* **1** (1971) 168–194.

[10] D. R. Fulkerson, Packing rooted directed cuts in a weighted directed graph. *Math. Programming* **6** (1974) 1–13.

[11] D. Gale, A theorem of flows in networks. *Pacific J. Math.* **7** (1957) 1073–1082.

[12] M. X. Goemans, The Steiner tree polytope and related polyhedra. Working Paper (1991).

[13] M. X. Goemans, Arborescence polytopes for series-parallel graphs. Working Paper (1991), to appear in *Discrete Applied Mathematics*.

[14] M. X. Goemans and D. Bertsimas, Survivable networks, linear programming relaxations and the parsimonious property. Working Paper OR 216-90, MIT (1990), to appear in *Math. Programming*.

[15] M. Held and R. M. Karp, The traveling-salesman problem and minimum spanning trees: Part II. *Math. Programming* **1** (1971) 6–25.

[16] M. Juenger and W. R. Pulleyblank, private communication with W. R. Pulleyblank (1991).

[17] W. Liu, Extended formulations and polyhedral projections. PhD Thesis, Department of Combinatorics and Optimization, University of Waterloo (1988).

[18] W. Liu, A lower bound for the Steiner tree problem in directed graphs. *Networks* **20** (1990) 765–778.

[19] A. Lucena, Tight bounds for the Steiner problem in graphs. Working Paper (1991).

[20] N. Maculan, "The Steiner problem in graphs. *Ann. Discrete Math.* **31** (1987) 185–212.

[21] R. K. Martin, A sharp polynomial size linear programming formulation of the minimum spanning tree problem. Working Paper, University of Chicago (1986).

[22] R. K. Martin, Using separation algorithms to generate mixed integer model reformulations. Working Paper, University of Chicago (1988).

[23] R. K. Martin, R. L. Rardin, and B. A. Campbell, Polyhedral characterization of discrete dynamic programming. *Operations Res.* **38** (1990) 127–138.

[24] Y.-s. Myung, Valid inequalities and facets for the Steiner problem in a directed graph. Working Paper (1991).

[25] G. Nemhauser and L. A. Wolsey, Integer and combinatorial optimization. Wiley, New York (1988).

[26] A. Prodon, T. M. Liebling, and H. Gröflin, Steiner's problem on 2-trees. Research Report RO 850315, Ecole Polytechnique de Lausanne (1985).

[27] J. M. W. Rhys, A selection problem of shared fixed costs and network flows. *Management Sci.* **17** (1970) 200–207.

[28] M. Schaffers, Network flow design III. Polyhedral characterization of the single source fixed costs problem on series-parallel graphs. CORE Discussion Paper, Université Catholique de Louvain (1991).

[29] P. Winter, Steiner problem in networks: A survey. *Networks* **17** (1987) 129–167.

[30] R. T. Wong, A dual ascent approach for Steiner tree problems on a directed graph. *Math. Programming* **28** (1984) 271–287.

[31] M. Yannakakis, Expressing combinatorial optimization problems by linear programs. Working Paper, AT&T Bell Laboratories (1988).

Received July 1991
Accepted June 1992