UNIVERSITÄT KAISERSLAUTERN

WEIGHTED k-CARDINALITY TREES: COMPLEXITY AND POLYHEDRAL STRUCTURE

Matteo Fischetti, Horst W. Hamacher, Kurt Jörnsten
and Francesco Maffioli

Preprint Nr. 228

FACHBEREICH MATHEMATIK

WEIGHTED k–CARDINALITY TREES: COMPLEXITY AND POLYHEDRAL STRUCTURE

Matteo Fischetti, Horst W. Hamacher, Kurt Jörnsten and Francesco Maffioli

Preprint Nr. 228

UNIVERSITÄT KAISERSLAUTERN
Fachbereich Mathematik
Erwin–Schrödinger–Straße
6750 Kaiserslautern

September 1992

WEIGHTED k-CARDINALITY TREES: COMPLEXITY AND POLYHEDRAL STRUCTURE

Matteo Fischetti, University of Bologna (*)
Horst W. Hamacher, University of Kaiserslautern (+, ◇)
Kurt Jørnsten, Norwegian School of Economics, Bergen
Francesco Maffioli, Politecnico (T.U.) of Milano (*, +)

Abstract

We consider the k-CARD TREE problem, i.e., the problem of finding in a given undirected graph G a subtree with k edges, having minimum weight. Applications of this problem arise in oil-field leasing and facility layout. While the general problem is shown to be strongly NP hard, it can be solved in polynomial time if G is itself a tree. We give an integer programming formulation of k-CARD TREE, and an efficient exact separation routine for a set of generalized subtour elimination constraints. The polyhedral structure of the convex hull of the integer solutions is studied.

1 Introduction

We consider a graph $G = (V, E)$ with node set V and edge set E. The cardinality of V and E is $n := |V|$ and $m := |E|$, respectively. For each $e \in E$ a weight $w(e) \in R$ is given, and we denote for each $E' \subseteq E$ with $w(E') := \sum_{e \in E'} w(e)$ the *weight* of E'.

A *tree* in G is a subgraph $T = (V(T), E(T))$ of G such that T contains no cycles and is connected. We often use the notation $w(T)$ for $w(E(T))$. The cardinality $|T|$ of T is the cardinality of $E(T)$. For given k with $1 \leq k \leq n-1$ a *k-cardinality tree* is a tree T with $|T| = k$. If $|T| = n-1$, then T is called a *spanning tree* of G.

Partially supported by:
(*) Research Project MPI 40% of the Italian Ministry of University and Scientific and Technological Research,
(+) NATO Grant RG 85-0240, and
(◇) Grant No. Ha. 19951-1 of the Deutsche Forschungsgemeinschaft

In this paper we study the *minimum (weight) k-cardinality tree problem* (k-CARD TREE)

$$\text{minimize } \{w(T) : T \text{ is } k\text{-cardinality tree}\}.$$

For $k = n - 1$ this problem is nothing but the shortest spanning tree problem and therefore solvable in polynomial time by the greedy algorithm (Kruskal [1956], Prim [1957]). For any fixed k the problem is also polynomially solvable by enumeration.

If we have weights on the nodes instead of on the edges of G, then $w(T)$ is the sum of the node weights of a k-cardinality tree. Notice that the node version of k-CARD TREE is only meaningful for k strictly less than $n - 1$, since otherwise the node weight of the tree is simply the sum of node weights of all nodes in G.

Before we enter the detailed discussion of k-CARD TREE, we will briefly mention two industrial problems for which k-CARD TREE can be used as a suitable model.

The first problem occurs in oil field leasing (Hamacher and Jørnsten [1992]). A government has the following 50% rule for off-shore oil fields. If a company has bought the lease for an oil field it has a certain number of years, say 5 years, to explore this oil field. After the end of this exploration time the company has to return at least 50% of the oil field to the government. But the rules of the government require that the returned part of the oil field is connected. The obvious goal for the company is to return a part with smallest possible value (and thus keep the most valuable part). In Hamacher and Jørnsten [1992] it is assumed that the oil field is a rectangle separated into subsquares. The company who has leased the field has 5 years time to gather information on the value w_i of each subsquare (e.g. by drillings and by the use of extrapolation methods). The part of the oil field which the company will return to the government therefore corresponds to a subset of at least 50% of the subsquares which is connected and has a minimal cumulative value of the weights w_i. In order to model the connectivity, we associate the subsquare partition of the rectangle is associated with its dual graph G which is, obviously, a grid graph (see Figure 1.1). A connected subset of squares in the rectangle then corresponds to a subset of the nodes of G which spans a connected subgraph, where connectivity is now the usual graph theoretic connectivity. Since the weights of the subsquares translate in the dual graph into weights of the nodes we are interested in finding in G a subtree with node cardinality of at least $1/2\ n$ (where n is the number of nodes in G) which has minimum weight, i.e., we are facing the node version of k-CARD-TREE.

The second industrial problem comes from the area of facilities layout, in particular the layout of office buildings. Consider again a rectangular area as in Figure 1.1 (a), but now this rectangle represents an office area which is to be laid out into, say p, offices of a)

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15

b)

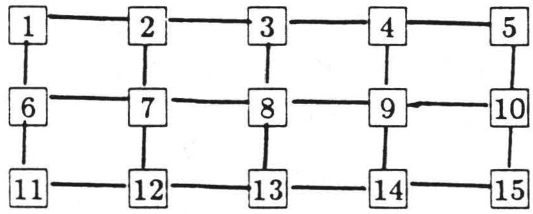

Figure 1.1: a) An oil field represented as rectangle with a subsquare partition; b) The corresponding dual graph G.

given areas consisting of $a_1, \ldots, a_p$ unit squares, respectively, (see Figure 1.2).

In the grid graph G corresponding to the partitioned rectangle, each of the offices corresponds to a connected subgraph of given node cardinality. Hence any feasible solution of the layout problem can be represented by a partitioning of G into p a_j-cardinality subtrees. For more details see Foulds and Hamacher [1992].

In the next section we will show that k-CARD TREE is NP hard. The same result will be proved for the problem of finding a connected subgraph (i.e. not necessarily a subtreœe) of given cardinality. A polynomial algorithm for k-CARD TREE is presented for the case where G is itself a tree. In Section 3 we formulate k-CARD TREE as an integer program and discuss some lower and upper bounds for the problem. The last section contains an investigation of the k-CARD TREE polytope. Several classes of facet defining inequalities are introduced and the corresponding separation problems are discussed.

2 Complexity of k-CARD TREE

In this section we show that the *Steiner-tree problem*, which is well known to be (stronlgy) NP-hard (Garey and Johnson [1979]) can be reduced to k-CARD TREE. In the former problem we are given a subset $S \subseteq V$ and we are trying to find a

A	A	B	E	E	F
A	B	B	E	F	F
A	C	B	E	F	F
A	C	C	C	F	F
A	C	D	D	D	D

Figure 1.2: An office building with 6 offices

minimum weight Steiner tree, i.e. a tree T in G such that $S \subseteq V(T)$ and $w(T)$ is minimum.

One way of approaching this problem is to solve sequentially the problem of finding a minimum weight Steiner tree of cardinality $s-1, s, \ldots, n-1$, respectively, where $s = \mid S \mid$, and take the best of these fixed-cardinality Steiner trees as optimal solution of the Steiner tree problem. Hence, the t-cardinality Steiner tree problem is NP-hard.

We will fix t with $s \leq t \leq n-1$ and show that the t-cardinality Steiner tree problem can be reduced to the $(t+s)$-CARD TREE problem. To that aim we assume that $S := \{1, \ldots, s\}$. We add a new set $S' := \{1', \ldots, s'\}$ of s nodes to G, and edge set $E' := \{\{i, i'\}, i = 1, \ldots, s\}$ with weights $w(\{i, i'\}) = -M$ (see Figure 2.1) where M is sufficiently large, say $M > \sum_{e \in E} \mid w(e) \mid$.

Lemma 2.1 Any minimum-weight t-cardinality Steiner tree in G corresponds to a minimum weight $(t+s)$-cardinality tree in $G_S = (V \cup S', E \cup E')$.

Proof Let T be a minimum-weight t-cardinality Steiner tree in G. Then $T \cup E'$ is a $(t+s)$-cardinality tree in G_S.

Now consider any other $(t+s)$-cardinality tree $\mathcal{T}$ in G_S.

Case 1 - $\mathcal{T}$ contains no more that $s-1$ edges of E'. Then

$$\begin{aligned} w(\mathcal{T}) &\geq w(\mathcal{T} \cap E) - M(s-1) \\ &\geq (w(T) - M) - M(s-1) \quad \text{by the choice of } M \\ &= w(T) - M\, s = w(T \cup E') \end{aligned}$$

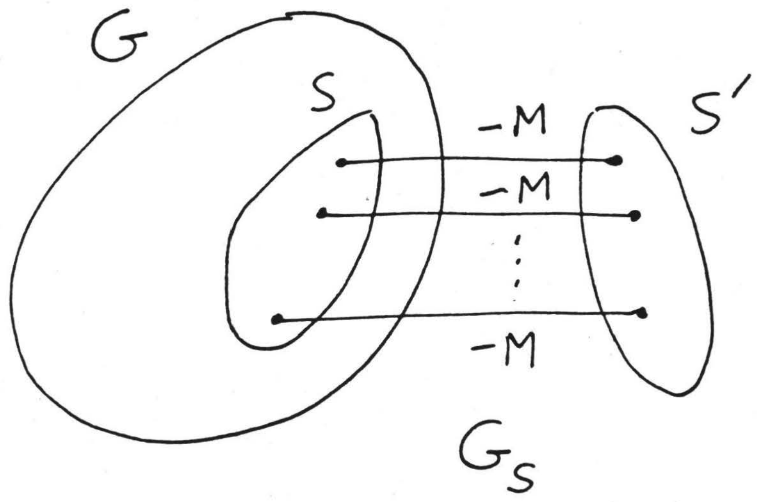

Figure 2.1

Case 2 - $\mathcal{T}$ contains all s edges of E'. Since the nodes of S' are leaves of $\mathcal{T}$, $\mathcal{T} - E'$ is a t-cardinality Steiner tree in G. Since T is a minimum t-cardinality Steiner tree in G we have $w(T) \leq w(\mathcal{T} - E')$ and consequently $w(T \cup E') \leq w(\mathcal{T})$.

Since in both cases $w(T \cup E') \leq w(\mathcal{T})$, $T \cup E'$ is a minimum weight $(t+s)$-cardinality tree in G_S.

Conversely, let T_S be a minimum weight $(t+s)$-cardinality tree in G_S. Since all nodes $i' \in S'$ are leaves of T_S, $T := T_S - E'$ is a t-cardinality Steiner tree in G. If there were another t-cardinality Steiner tree $\mathcal{T}$ in G with $w(\mathcal{T}) < w(T)$, then $w(\mathcal{T} \cup E') < w(T_S)$, contradicting the optimality of T_S. Hence T is a minimum weight t-cardinality Steiner tree in G. □

An immediate consequence of Lemma 2.1 is the following result.

Theorem 2.2 The minimum k-cardinality tree problem is (strongly) NP-hard.

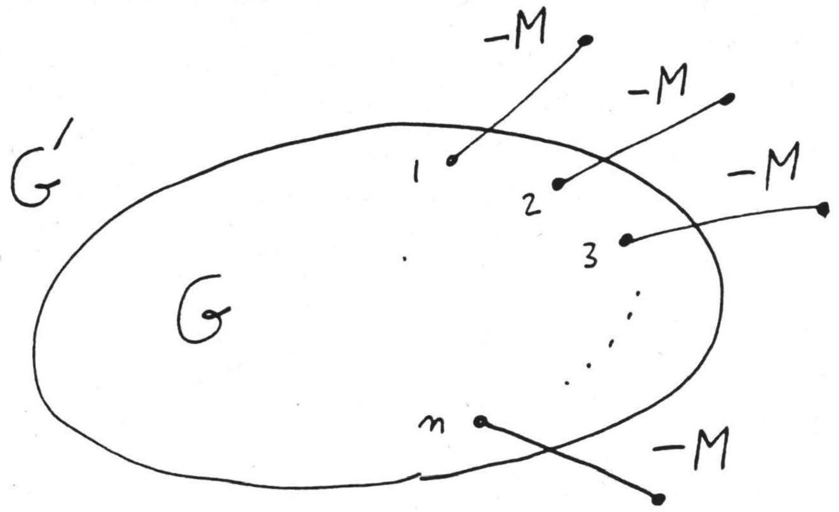

Figure 2.2

The node-weighted version of the Steiner tree problem is easily proved NP-hard (simply insert a new node in each edge). Then a construction similar to the one used above proves that the node-weighted version of the k-cardinality tree problem is strongly NP-hard.

Even if we relax k-CARD TREE to the *k-cardinality subgraph* problem, that is, find a minimum-weight connected subgraph with k edges, the problem remains NP-hard.

Theorem 2.3 The k-cardinality subgraph problem is NP-hard.

Proof Any optimum solution of the $(2k+1)$-cardinality connected subgraph problem on the graph G' of Figure 2.2 corresponds uniquely to a k-cardinality tree of G, since exactly $k+1$ edges of $G' - G$ must belong to the optimum. □

On the other hand when the original graph is itself a tree, k-CARD TREE is polynomially solvable. Note that the obvious idea of applying Prim's algorithm (stopping when k arcs are present, starting in turn from every node of the whole tree, and taking the best result) does not guarantee optimality even in this particular case.

Theorem 2.4 If G is a tree, k-CARD TREE in G is polynomially solvable.

Proof (sketched) An $O(k^2\ n)$ algorithm may be obtained along the following lines. Choose an arbitrary node as the root of G and orient G correspondingly. For each node we solve the h-CARD TREE problem, where $h = 0, 1, \ldots, k$, for the subgraph induced by the descendant nodes. This is trivial for the leaves of G, whereas for the other nodes it can be done in a dynamic fashion working towards the root, requiring in total no more than n merging operations. See Appendix A for the resulting algorithm. □

3 k-CARD TREE as Integer Program

If T is a k-cardinality tree we can introduce characteristic vectors $\boldsymbol{x} = (x_e) \in B^E$ and $\boldsymbol{y} = (y_i) \in B^V$, where $B := \{0,1\}$,

$$x_e = \begin{cases} 1 & \text{if edge } e \in E(T) \\ 0 & \text{otherwise} \end{cases}$$

and

$$y_i = \begin{cases} 1 & \text{if node } i \in V(T) \\ 0 & \text{otherwise} \end{cases}$$

We will often use notation x_{ij} instead of x_e for $e = \{i,j\}$, and for any given $S \subseteq V$,

$$\gamma(S) := \{\{i,j\} \in E : i,j \in S\}$$

and

$$\delta(S) := \{\{i,j\} \in E : i \in S, j \notin S\}.$$

To simplify notation, we write $\delta(i)$ for $\delta(\{i\})$, $i \in V$. Moreover let

$$x(E') := \sum_{e \in E'} x_e$$

and

$$y(V') := \sum_{j \in V'} y_j$$

for any $E' \subseteq E$ and $V' \subseteq V$. The vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ will then satisfy the following conditions

$$x(E) = k, \tag{3.1}$$

$$y(V) = k + 1, \tag{3.2}$$

$$x(\gamma(S)) \leq y(S) - y_t, \ \forall S \subseteq V, \mid S \mid \geq 2, \ t \in S \tag{3.3}$$

Note that when $S = \{i, j\}$, constraints (3.3) become

$$\begin{aligned} x_{ij} &\leq y_i, \\ x_{ij} &\leq y_j. \end{aligned} \tag{3.4}$$

Constraints (3.1) and (3.2) are cardinality conditions. Constraints (3.3) guarantee that T does not contain cycles. All constraints together enforce connectivity. Then the following result holds.

Theorem 3.1 Given $x \in B^E$ and $y \in B^V$ satisfying (3.1)-(3.3), let $E' := \{e \in E : x_e = 1\}$ and $V' := \{i \in V : y_i = 1\}$. Then E' and V' are edge set and node set, respectively, of a k-cardinality tree T.

Proof By (3.1) and (3.3) E' is the edge set of a forest in G with cardinality k (notice that (3.3) implies $x(\gamma(S)) \leq \mid S \mid -1$). Hence the cardinality of the node set $V(E')$ incident with some $e \in E'$ satisfies

$$\mid V(E') \mid \geq k + 1 \tag{3.5}$$

If we consider any $e = \{i, j\} \in E'$, then $x_e = 1$ and (3.4) implies $y_i = y_j = 1$. Hence $V(E') \subseteq V'$ and we get from (3.2)

$$\mid V(E') \mid \leq \mid V' \mid = k + 1. \tag{3.6}$$

From (3.5) and (3.6) we conclude that $\mid V(E') \mid = \mid V' \mid = k + 1$, i.e., $V(E') = V'$. Since (E', V') is a forest with k edges and $k + 1$ nodes it is a k-cardinality tree. □

Hence we can write k-CARD TREE as the following Integer Linear Program (ILP).

$$\text{minimize} \sum_{e \in E} w(e) \, x_e \tag{3.7}$$

subject to (3.1), (3.2), (3.3), and $x_e \in \{0,1\} \ \forall e \in E$, $y_i \in \{0,1\} \ \forall i \in V$.

Notice that this formulation can easily incorporate weights on the nodes. If we drop the integrality constraints we get a valid LP relaxation. In order to deal with the exponentially many constraints of type (3.3) we choose a subset of these constraints

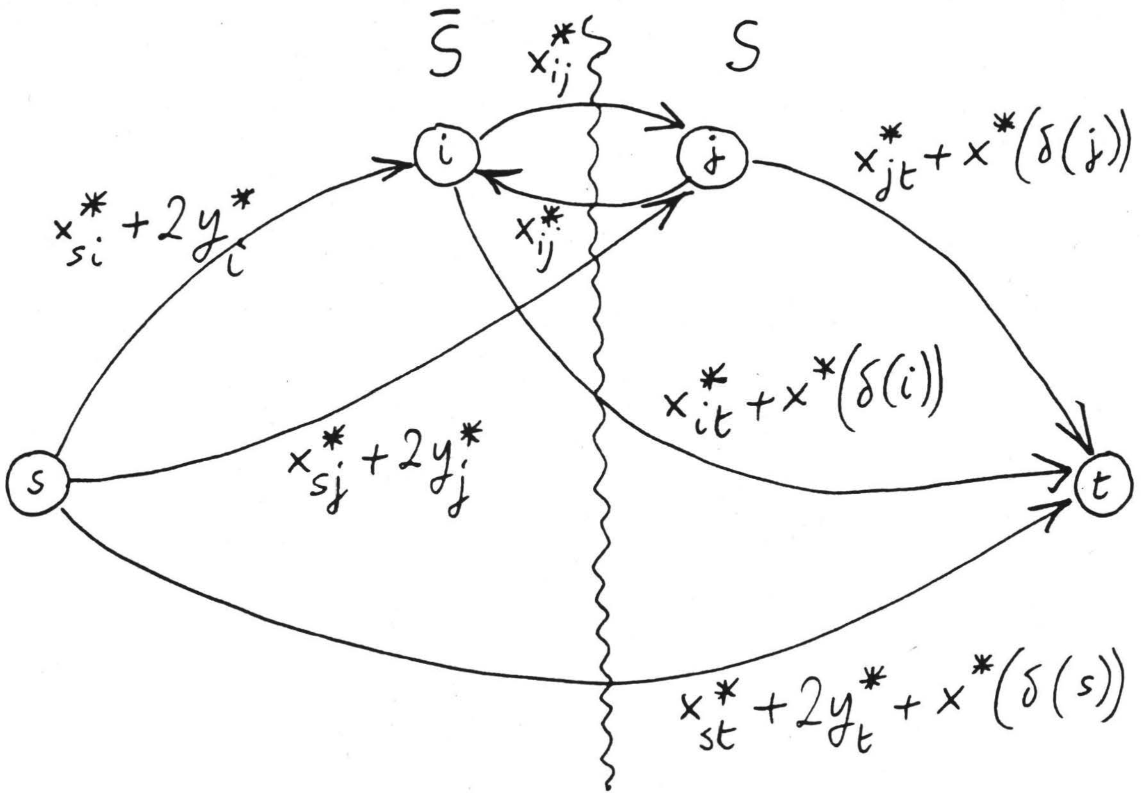

Figure 3.1: Network of the separation problem (capacities on arcs)

and compute an optimum solution (x^*, y^*) of the corresponding LP problem. With respect to this solution we then apply the following *separation routine* for finding a most violated inequality of type (3.3), if any.

Using the identity

$$\sum_{i\in\bar{S}} x(\delta(i)) + x(\delta(S)) + 2\,x(\gamma(S)) = 2x(E)$$

where $\bar{S} := V - S$, we rewrite (3.3) as

$$2y(S) + x(\delta(S)) + \sum_{i\in\bar{S}} x(\delta(i)) \;\geq\; 2(x(E) + y_t). \tag{3.8}$$

Therefore for any node $s \neq t$ the most violated constraint of type (3.8) such that $t \in S$ and $s \in \bar{S}$ corresponds to the minimum capacity (s,t)-cut in a flow network with source s and sink t (see Figure 3.1).

Consequently an all pairs maximum flow algorithm solves the separation problem for constraints (3.3). Notice that Gomory-Hu type algorithms are not applicable since the structure of the network changes with each source-sink pair. Nevertheless we obtain the following result

Theorem 3.2 A most violated constraint (3.3) can be found with at most $2n - 2$ maximum flow computations.

Proof Let h be such that $y_h^* \geq y_i^*$, $\forall\, i \neq h$. We will show that it is sufficient to solve the $2n-2$ maximum flow problems where $s = h$ or $t = h$. Indeed, let $(\bar{S}, S)$ be a minimum capacity (s,t)-cut with $h \notin \{s,t\}$. If $h \in \bar{S}$, then the same cut will separate h and t, and hence we can replace s by h. If $h \in S$, then (3.3) (or (3.8)) shows that replacing t by h produces a cut which is not worse than $(\bar{S}, S)$. □

Using the ILP formulation (3.7) and Theorem 3.2 we can solve k-CARD TREE using a cutting-plane branch-and-bound scheme. Upper bounds can be computed as follows.

The Prim algorithm (Prim [1957]) chooses a root node i and grows, at iteration ℓ, a tree rooted at i with cardinality ℓ. If we stop the algorithm after the k-th iteration we get therefore a k-cardinality tree T_i rooted at i. Hence

$$\mathrm{U} := \min \{w(T_i) : i = 1, \ldots, n\}$$

is an upper bound of k-CARD TREE.

Since this procedure always yields a subtree of an optimum spanning tree of the graph, an improved upper bound can be obtained as follows. Extend the Prim k-cardinality tree (corresponding to the bound U) to an optimum spanning tree and apply the procedure of Appendix A (which proves Theorem 2.4) to it. Notice also that the idea of extending any given heuristic solution of k-CARD TREE to an arbitrary spanning tree, and then applying this procedure, can be used as a refinement tool.

4 Facets of the k-CARD TREE Polytope

In this section we assume $3 \leq k \leq n-2$, thus excluding the easy solvable cases $k = 1, 2$, as well as the spanning tree case arising when $k = n - 1$. Moreover, we suppose that G is a (simple) complete graph, i.e., $m = n(n-1)/2$. We will study the facial structure of the k-CARD TREE polytope, defined as

$$P := \mathrm{conv}\,\{(\boldsymbol{x}, \boldsymbol{y}) \in B^{|E|+|V|} : (3.1), (3.2), \text{and } (3.3) \text{ hold}\}.$$

We first address the dimension of P.

Theorem 4.1 Equations (3.1) and (3.2) define a minimal equality set of P, i.e., $\dim(P) = m + n - 2$.

Proof The two equations (3.1) and (3.2) are clearly linearly independent, i.e., $\dim(P) \leq m + n - 2$. We now prove that the reverse inequality holds as well. Let

$$\alpha\, x + \beta\, y = \gamma \tag{4.1}$$

be any equation satisfied by all $(x, y) \in P$. We have to show that (4.1) is a linear combination of (3.1) and (3.2).

Let $e^* \in E$ and $j^* \in V$ be chosen arbitrarily. We will assume without loss of generality

$$\alpha_{e^*} = \beta_{j^*} = 0. \tag{4.2}$$

Indeed, if this is not the case one can subtract from (4.1) the two equations (3.1) and (3.2) weighted by α_{e^*} and β_{j^*}, respectively. By construction, the resulting equation still holds for all $(x, y) \in P$, and satisfies the "normalization" condition (4.2); moreover, it is linearly dependent on (3.1) - (3.2) if and only if the original equation (4.1) is.

We now show that, under assumption (4.2), one necessarily has $\alpha = 0$, $\beta = 0$, and $\gamma = 0$, thus proving the linear dependence of (4.1) on (3.1) - (3.2). Indeed, let $e \in E - \{e^*\}$ be any edge, and consider two k-cardinality trees T_{e^*} and T_e such that $e^* \in E(T_{e^*})$, $e \in E(T_e)$, $V(T_e) = V(T_{e^*})$, and $E(T_{e^*}) - \{e^*\} = E(T_e) - \{e\}$ (see Figure 4.1.a).

The characteristic vectors of T_e and T_{e^*} must satisfy (4.1), and therefore $\alpha_e = \alpha_{e^*}$ holds, with $\alpha_{e^*} = 0$ by assumption. Because of the arbitrariness of e, this leads to $\alpha_e = 0$ all $e \in E$. Consider now any $j \in V - \{j^*\}$, and let T_{j^*} and T_j be two k-cardinality trees such that $j^* \in V(T_{j^*})$, $j \in V(T_j)$, and $V(T_{j^*}) - \{j^*\} = V(T_j) - \{j\}$ (see Figure 4.1.b). The characteristic vectors of T_{j^*} and T_j satisfy (4.1), hence (since $\alpha = 0$) one has $\beta_j = \beta_{j^*} = 0$. It then follows that $\alpha = 0$ and $\beta = 0$, from which one has $\gamma = 0$ since otherwise no (x, y) would satisfy (4.1), which is impossible since clearly $P \neq \phi$. □

We now address to so-called *trivial* facets of P.

Theorem 4.2 The inequalities

$$x_f \geq 0, \quad f \in E \tag{4.3}$$

$$y_h \leq 1, \quad h \in V \tag{4.4}$$

define (trivial), facets of P.

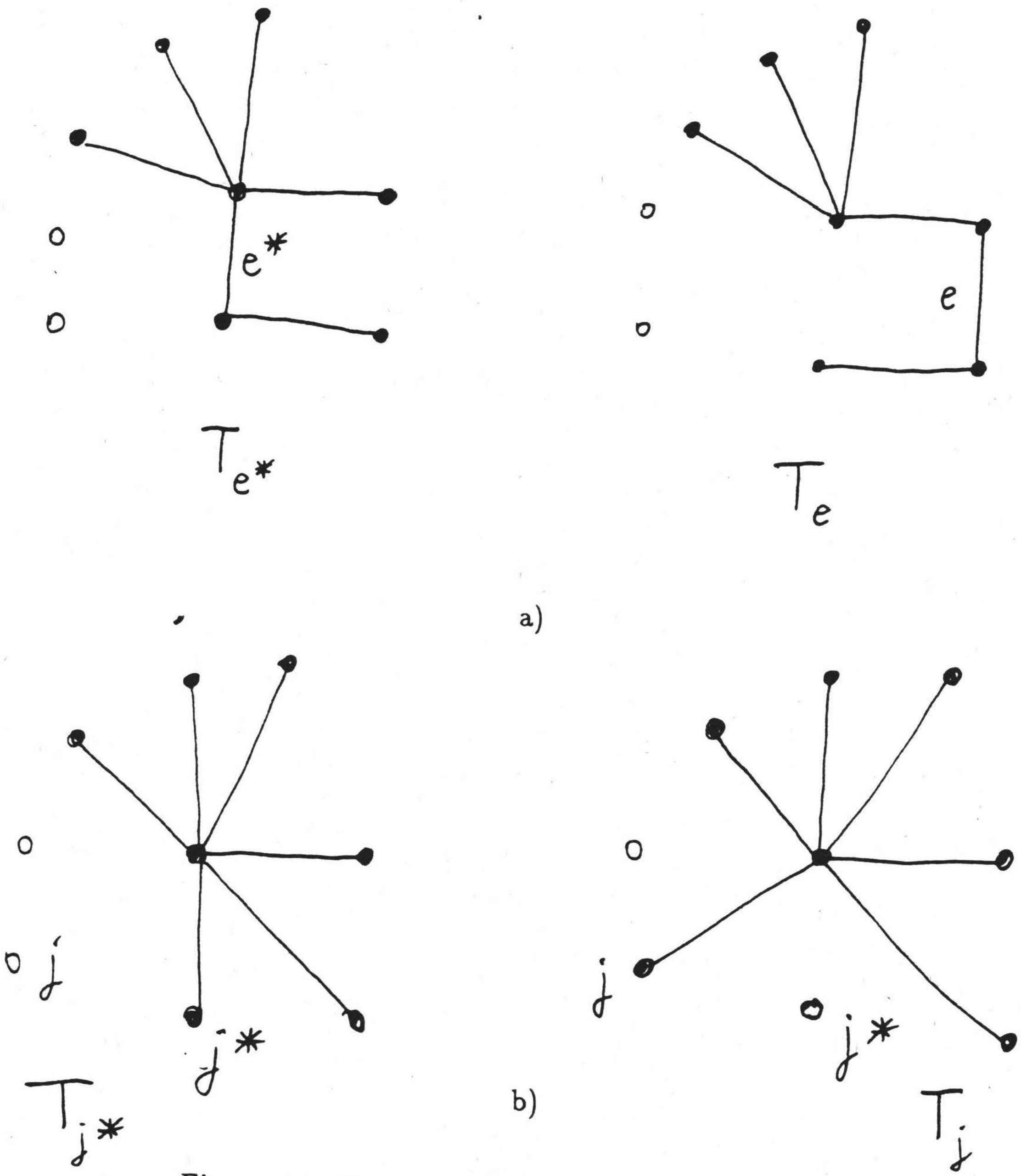

Figure 4.1: Illustration for the proof of Theorem 4.1

Proof The proof is along the same lines as that of Theorem 4.1.

Consider first the inequality $x_f \geq 0$ for any given $f = \{i,j\} \in E$, and let $F := \{(\boldsymbol{x},\boldsymbol{y}) \in P : x_f = 0\}$ be the associated face of P. We have to prove $\dim(F) = \dim(P) - 1$, i.e., that (3.1), (3.2) and

$$x_f = 0 \tag{4.5}$$

define an equality set for F. To this end, let (4.1) be any equation satisfied by all $(\boldsymbol{x},\boldsymbol{y}) \in F$. We will show that (4.1) is a linear combination of (3.1), (3.2), and (4.5). By exploiting these latter 3 equations, one can always assume that the three "normalization" conditions

$$\alpha_{e^*} = \alpha_f = \alpha_{j^*} = 0 \tag{4.6}$$

hold, where $e^* \in E - \delta(\{i,j\})$ and $j^* \in V - \{i,j\}$ are chosen arbitrarily. It is then not hard to see, using the tree constructions outlined in Figure 4.1, that $\boldsymbol{\alpha} = \mathbf{0}$, $\boldsymbol{\beta} = \mathbf{0}$ (and hence, since $F \neq \phi$, $\gamma = 0$) hold, from which the claim follows.

The proof for inequalities (4.4) is perfectly analogous. Indeed, by assuming normalization $\alpha_{e^*} = \beta_{j^*} = \beta_h = 0$ for arbitrarily chosen $e^* \in \delta(h)$ and $j^* \in V - \{h\}$, one can use the construction of Figure 4.1.a (where h now coincides with one of the two extreme nodes of e^*) to show $\alpha_e = \alpha_{e^*} = 0$ for all $e \in E - \{e^*\}$, and that of Figure 4.1.b (where h is chosen as the "central" node of the star) to prove $\beta_j = \beta_{j^*} = 0$ for all $j \in V - \{h, j^*\}$. □

Notice that the inequalities $y_h \geq 0$ $(h \in V)$ and $x_f \leq 1$ $(f \in E)$ are dominated by (3.4), and hence do not define facets of P.

Theorem 4.3 Inequalities (3.3) with $| S | < n - k$ define facets of P.

Proof Let $F := \{(\boldsymbol{x},\boldsymbol{y}) \in P : x(\gamma(S)) = y(S) - y_t\}$ be the proper face of P induced by any inequality (3.3) having $| S | < n - k$. Notice that, because of this latter assumption, we have $| \bar{S} | \geq k + 1$, hence there exist k-cardinality trees covering no nodes in S. We will prove that 3 linearly independent equations (3.1), (3.2), and

$$x(\gamma(S)) = y(S) - y_t, \tag{4.7}$$

define an equality set of F. To this end, let equation (4.1) be satisfied by all $(\boldsymbol{x},\boldsymbol{y}) \in F$, and assume (without loss of generality, as explained in the previous proofs) it satisfies the normalization conditions

$$\alpha_{e^*} = \alpha_{f^*} = \beta_{j^*} = 0, \tag{4.8}$$

where $e^* \in \gamma(S) \cap \delta(t)$, $f^* \in \gamma(\bar{S})$, and $j^* \in S - \{t\}$ are chosen arbitrarily. We prove that $\boldsymbol{\alpha} = \mathbf{0}$ and $\boldsymbol{\beta} = \mathbf{0}$ (and hence $\gamma = 0$) hold.

Let $\mathcal{F}$ be the family of the k-cardinality trees T whose characteristic vector belongs to the face F, i.e., such that either $t \in V(T)$ and $E(T) \cap \gamma(S)$ defines a tree spanning $V(T) \cap S$, or $V(T) \cap S = \phi$.

(a) To see that $\alpha_e = 0$ holds for each $e \in \gamma(S) - \{e^*\}$, consider two k-cardinality trees $T_e, T_{e^*} \in \mathcal{F}$ such that $e \in E(T_e)$, $e^* \in E(T_{e^*})$, $V(T_e) = V(T_{e^*})$, and $E(T_e) - \{e\} = E(T_{e^*}) - \{e^*\}$; see Figure 4.2.a, where T_e is drawn in continuous line. The characteristic vector of both T_e and T_{e^*} must satisfy $\boldsymbol{\alpha x} + \boldsymbol{\beta y} = \gamma$, from which one immediately has $\alpha_e = \alpha_{e^*}$, where $\alpha_{e^*} = 0$ because of normalization (4.8).

(b) Using a construction very similar to that of case (a) above, and illustrated in Figure 4.2.b, one easily obtains $\alpha_f = \alpha_{f^*} = 0$ for all $f \in \gamma(\bar{S}) - \{f^*\}$.

(c) Consider now any $e := \{i, j\} \in \delta(S)$ with $i \in S$ and $j \in \bar{S}$ (possibly $i = t$). Let w be any node in $\bar{S} - \{j\}$, and consider the k-cardinality tree $T_e \in \mathcal{F}$ of Figure 4.2.c having $E(T_e) \cap \delta(S) = \{e, \{t, w\}\}$. Comparing T_e with the tour T_f obtained by replacing e with $f := \{j, w\}$, one concludes that $\alpha_e = \alpha_f$, where $\alpha_f = 0$ has been proved in case (b) above.

(d) We now prove $\beta_j = \beta_{j^*} = 0$ for all $j \in V - \{t, j^*\}$. Let T_j and T_{j^*} belong to $\mathcal{F}$ and satisfy $j \in V(T_j)$, $j^* \in V(T_{j^*})$, $V(T_j) - \{j\} = V(T_{j^*}) - \{j^*\}$; see Figure 4.2.d for an illustration of T_j. Since, as already shown, $\alpha_e = 0$ for all $e \in E$, the existence of T_j and T_{j^*} implies $\beta_j = \beta_{j^*}$, where $\beta_{j^*} = 0$ from (4.8).

(e) It remains to be shown that $\beta_t = 0$ holds as well. To this end it suffices to consider any $T \in \mathcal{F}$ with $V(T) \subseteq \bar{S}$, and compare it with the k-cardinality tree T' obtained by disconnecting from T one of its leaves, say node j, and then connecting t to the resulting subtree (see Figure 4.2.e). This yields $\beta_t = \beta_j = 0$, as required.

Therefore $\boldsymbol{\alpha} = \mathbf{0}$ and $\boldsymbol{\beta} = \mathbf{0}$ hold, as well as $\gamma = 0$ since otherwise $F = \phi$. The theorem follows. □

When $|\ S\ | \geq n - k$, instead, the inequality (3.3) is dominated by

$$x(\gamma(S)) \leq y(S) - 1. \tag{4.9}$$

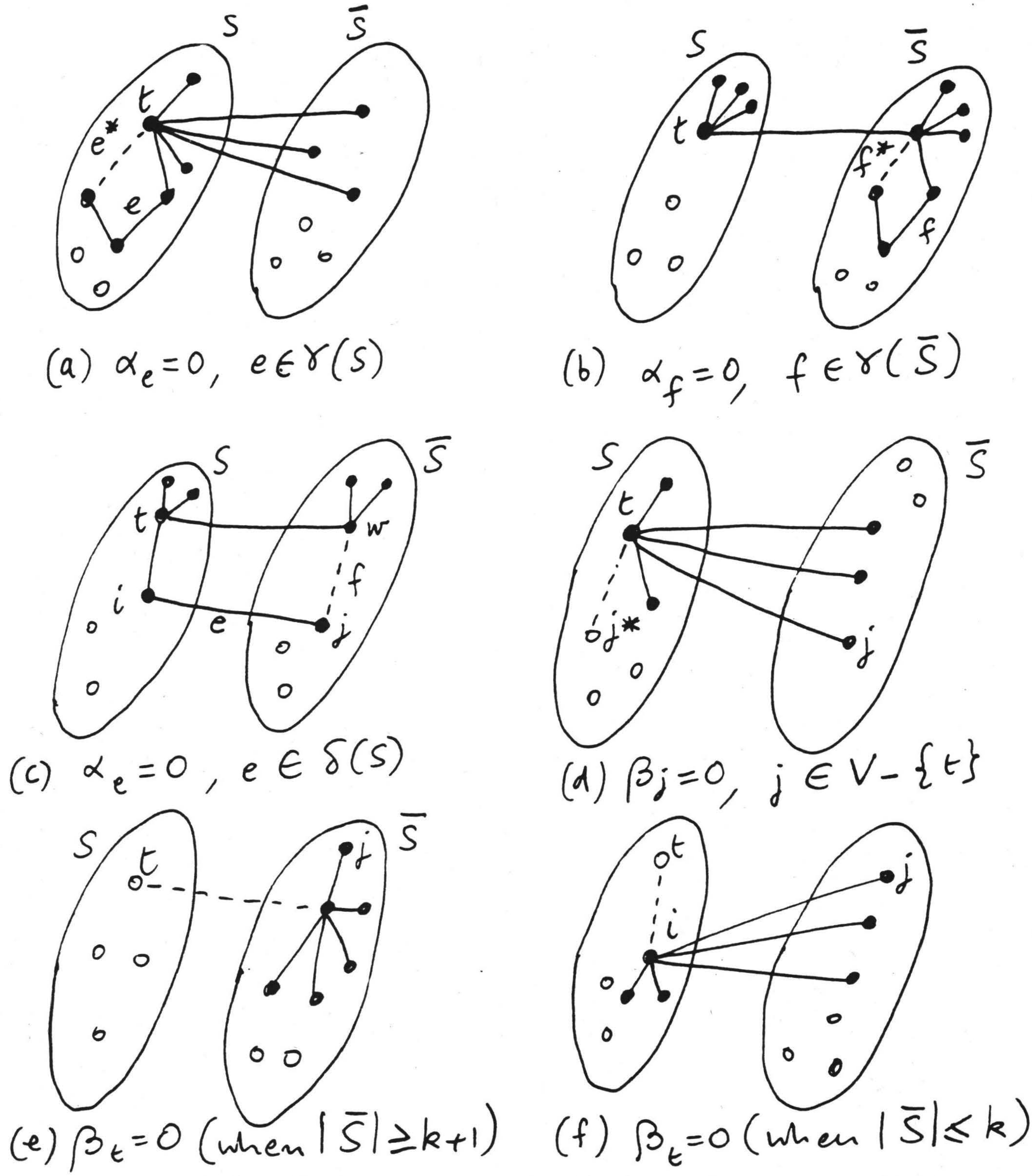

Figure 4.2: k-cardinality tree constructions for the proof of Theorems 4.3 and 4.4

To see that this inequality is valid for P, it is sufficient to note that a k-cardinality tree T would violate (4.9) only in case $V(T) \subseteq \bar{S}$, impossible since $| V(T) | = k+1$ whereas $| \bar{S} | \leq k$. When $S = V$, (4.9) define the improper face of P, i.e., is satisfied with equality by all $(\boldsymbol{x}, \boldsymbol{y}) \in P$.

Theorem 4.4 Inequalities (4.9) with $n-k \leq | S | \leq n-1$ define facets of P.

Proof We claim that (3.1), (3.2) and

$$x(\gamma(S)) = y(S) - 1 \tag{4.10}$$

define a minimal equality set for the face $F := \{(\boldsymbol{x}, \boldsymbol{y}) \in P : \text{(4.10) holds}\}$ induced by (4.9). This amounts to showing that every given equation $\boldsymbol{\alpha x} + \boldsymbol{\beta y} = \gamma$ satisfied by all $(\boldsymbol{x}, \boldsymbol{y}) \in F$, is indeed a linear combination of (3.1), (3.2), and (4.10).

We distinguish between two cases.

1. $| \bar{S} | \geq 2$: In this case all the constructions outlined at points (a) to (d) of the proof of Theorem 4.3, still apply to our case, as they all involve k-cardinality trees for which $y_t = 1$. (Notice that, when $k = 3$, the tree of Figure 4.2.b still covers node T, since $| \bar{S} | \leq 3$ and hence f and f^* must be adjacent.)

   Assuming normalization (4.8), this leads to $\alpha_e = 0$ for all $e \in E$, and $\beta_j = 0$ for all $j \in V - \{t\}$. It then remains to be proved that $\beta_t = 0$. Indeed, let T be the k-cardinality tree of Figure 4.2.f, and T' the one with edge set $E(T') := (E(T) \cup \{\{i,t\}\}) - \{\{i,j\}\}$. The characteristic vector of both these trees belongs to F, hence one has $\beta_t = \beta_j = 0$, as required.

2. $| \bar{S} | = 1$. In this case, let $\bar{S} = \{h\}$, and note that (4.10) is equivalent to $x(\delta(h)) \geq y_h$. We can then without loss of generality assume that the normalization condition (4.8) holds for arbitrarily chosen $e^* \in E - \delta(h)$, $f^* \in \delta(h)$, and $j^* \in V - \{h\}$. Under this assumption, it is then easy to see (by means of simple tree constructions perfectly analogous to those used before) that $\boldsymbol{\alpha} = \mathbf{0}$ and $\boldsymbol{\beta} = \mathbf{0}$ necessarily hold. □

We next analyze two new classes of valid inequalities for P.

Let $i, j \in V$, $i \neq j$, and $S \subset V$ be such that $i \in S$ and $j \in \bar{S}$. Then the *cut inequality*

$$x(\delta(S)) \geq y_i + y_j - 1 \tag{4.11}$$

is valid for P, as $y_i = y_j = 1$ implies that the cut-set $\delta(S)$ must be crossed at least once. However, (4.11) is not facet-inducing since it can be obtained by adding the following two inequalities of class (3.3):

$$x(\gamma(S)) \leq y(S) - y_i$$

$$x(\gamma(\bar{S})) \leq y(\bar{S}) - y_j,$$

thus obtaining

$$x(E) - x(\delta(S)) \leq y(V) - y_i - y_j,$$

which is equivalent to (4.11) since, from (3.1) and (3.2), $x(E) = y(V) - 1$ holds for all $(\boldsymbol{x}, \boldsymbol{y}) \in P$.

The second class we consider is covered by the following theorem.

Theorem 4.5 Assuming $k \leq n - 3$, for every $i \in V$, the inequality

$$x(\delta(i)) \leq k\ y_i \tag{4.12}$$

is valid and facet-inducing for P.

Proof Validity is trivial. We prove that (3.1), (3.2) and

$$x(\delta(i)) = k\ y_i \tag{4.13}$$

define a minimal equality set for F, the face of P induced by (4.12). As in the previous proofs in this section, this amounts to showing that every equation $\boldsymbol{\alpha x} + \boldsymbol{\beta y} = \gamma$ satisfied by all $(\boldsymbol{x}, \boldsymbol{y}) \in F$ is a linear combination of (3.1), (3.2), and (4.13). Let us assume, without loss of generality, that the normalization conditions (4.8) hold for arbitrarily chosen $e^* \in E - \delta(i)$, $f^* \in \delta(i)$, and $j^* \in V - \{i\}$. We will prove that $\boldsymbol{\alpha} = \mathbf{0}$ and $\boldsymbol{\beta} = \mathbf{0}$ hold, from which the claim will follow.

The construction of Figure 4.1.a (with node i uncovered by T_{e^*}) immediately shows that $\alpha_e = \alpha_{e^*} = 0$ for all $e \in E - \delta(i)$. Similarly, the construction outlined in Figure 4.1.b (with i uncovered by both T_j and T_j^*) yields $\beta_j = \beta_{j^*} = 0$ for all $j \in V - \{i\}$.

To see that $\alpha_f = \alpha_{f^*} = 0$ for every $f \in \delta(i)$, it is then enough to compare two k-cardinality trees T_f and T_{f^*} with $f \in T_f$, $f^* \in T_{f^*}$, $E(T_f) - \{f\} = E(T_{f^*}) - \{f^*\} \subset \delta(i)$.

Finally, $\beta_i = 0$ derives from comparing any two k-cardinality trees T and T' such that $E(T) \subset \delta(i)$ and $E(T') \subset E - \delta(i)$. $\square$

Notice that, in case $k = n - 1$, (4.12) can be obtained by adding $x_e \leq y_i$ for all $e \in \delta(i)$. When $k = n - 2$, instead, (4.12) can be obtained by adding the $n - 1 = k + 1$ inequalities

$$x_{ij} \leq y_i + y_j - 1, \qquad j \in V - \{i\}$$

which are particularizations of (4.9) arising when $S = \{i, j\}$ (note that $|S| = 2 = n - k$, as required for the validity of (4.9)), plus the equation $y(V) = k + 1 = n - 1$.

We conclude the section with a discussion of possible equivalences among the facet-inducing inqualities of the classes covered by Theorems 4.2 to 4.5.

Two given valid inequalities $\boldsymbol{\alpha x} + \boldsymbol{\beta y} \leq \gamma$ and $\bar{\boldsymbol{\alpha}}\boldsymbol{x} + \bar{\boldsymbol{\beta}}\boldsymbol{y} \leq \bar{\gamma}$ are called *equivalent* with respect to P when any of the two can be obtained as a linear combination of the other and of the 2 equations (3.1) and (3.2). In other words, the two inequalities are equivalent if and only if there exist 3 real multipliers μ_1, μ_2 and μ_3, where $\mu_1 \geq 0$, such that

$$\bar{\alpha}_e = \mu_1 \, \alpha_e + \mu_2, \quad e \in E \tag{4.14}$$

$$\bar{\beta}_j = \mu_1 \, \beta_j + \mu_3, \quad j \in V \tag{4.15}$$

and

$$\bar{\gamma} = \mu_1 \, \gamma + \mu_2 \, k + \mu_3(k + 1). \tag{4.16}$$

Equivalent inequalities can be considered as different formulations of the same valid constraint.

It is well known that two facet-inducing inequalities define the same facet of P if and only if they are equivalent. Therefore one is interested in studying conditions under which two different inequalities define in fact the same facet of P.

For instance, it is not hard to see that the two inequalities

$$x(\gamma(S)) \leq y(S) - 1$$

and

$$x(\gamma(\bar{S})) + x(\delta(\bar{S})) \geq y(\bar{S}),$$

although different, are equivalent and hence define the same facet of P (assuming $|S| \geq n - k$, see Theorem 4.4).

We next define a canonical form that facilitates a rigorous comparison between inequalities. (Canonical forms for the asymmetric travelling salesman problem have been proposed by Balas and Fischetti [1992 a, b], and by Fischetti [1991] for the directed Steiner tree problem).

Definition 4.6 An inequality $\alpha x + \beta y \leq \gamma$ is said to be in *canonical form* iff:

(i) $\min\{\alpha_e : e \in E\} = 0$;

(ii) $\max\{\beta_j : j \in V\} = 0$;

(iii) the coefficients $\alpha_e (e \in E)$, $\beta_j (j \in V)$, and γ are relatively prime integers.

Every inequality $\alpha x + \beta y \leq \gamma$ with rational coefficients can be put in canonical form by subtracting equations (3.1) and (3.2) weighted by $\min\{\alpha_e : e \in E\}$ and $\max\{\beta_j : j \in V\}$, respectively, and then multiplying the resulting formulation by a convenient scaling factor.

Theorem 4.7 Two inequalities $\alpha x + \beta y \leq y$ and $\bar{\alpha} x + \bar{\beta} y \leq \bar{\gamma}$ in canonical form are equivalent if and only if they are the same.

Proof The "if" part is trivial. Suppose now that the two inequalities are equivalent, i.e., conditions (4.14) to (4.16) hold. We have to show that $\mu_1 = 1$, $\mu_2 = \mu_3 = 0$ necessarily hold when both inequalities are in canonical form. Indeed, let $e, \bar{e} \in E$ and $j, \bar{j} \in V$ be such that $\alpha_e = \bar{\alpha}_{\bar{e}} = \beta_j = \bar{\beta}_{\bar{j}} = 0$, see Definition 4.6. From $\bar{\alpha}_e = \mu_1 \alpha_e + \mu_2$ we have $\mu_2 \geq 0$ (since $\bar{\alpha}_e \geq 0$ and $\alpha_e = 0$), whereas from $\bar{\alpha}_{\bar{e}} = \mu_1 \alpha_{\bar{e}} + \mu_2$ we obtain $\mu_2 \leq 0$ (since $\bar{\alpha}_{\bar{e}} = 0$, $\mu_1 \geq 0$, and $\alpha_{\bar{e}} \geq 0$). Therefore $\mu_2 = 0$. Analogously, from $\bar{\beta}_j = \mu_1 \beta_j + \mu_3$ we have $\mu_3 \leq 0$ (as $\bar{\beta}_j \leq 0$ and $\beta_j = 0$), and from $\bar{\beta}_{\bar{j}} = \mu_1 \beta_{\bar{j}} + \mu_3$ we derive $\mu_3 \geq 0$ (as $\bar{\beta}_{\bar{j}} = 0$, $\mu_1 \geq 0$, and $\beta_{\bar{j}} \leq 0$). This proves $\mu_3 = 0$. Therefore one has $\bar{\alpha} = \mu_1 \alpha$, $\bar{\beta} = \mu_1 \beta$, and $\bar{\gamma} = \mu_1 \gamma$, where $\mu_1 = 1$ because of the scaling condition (iii) of Definition 4.6. □

Using the above result one can easily prove the following

Theorem 4.8 All the inequalities covered by Theorems 4.2 to 4.5 define distinct facets of P.

Proof The trivial inequalities (4.3) and (4.4) of Theorem 4.2 attain, respectively, the canonical form

$$x(E - \{f\}) \leq k, \quad f \in E \tag{4.3'}$$

$$-y(V - \{h\}) \leq -k, \quad h \in V. \tag{4.4'}$$

The inequalities (3.3) and (4.9) of Theorems 4.3 and 4.4 are already stated in canonical form, as well as the inequalities (4.12) of Theorem 4.5. All canonical forms are readily seen to be different, hence the claim follows from Theorem 4.7. □

References

E. Balas and M. Fischetti, [1992 a] "The fixed-outdegree 1-arborescence polytope", *Mathematics of Operations Research* (to appear).

E. Balas and M. Fischetti. [1992 b] "A lifting procedure for the asymmetric traveling salesman polytope and a large new class of facets", *Mathematical Programming* (to appear).

U. Faigle and W. Kern, "Computational Complexity of some maximum average weight problems with precedence constraints", Memorandum No. 899, Faculty of Applied Mathematics, University of Twente (1990).

M. Fischetti, "Facets of two Steiner arborescence polyhedra", *Mathematical Programming* 51, (1991) 401-419.

L.R. Foulds and H.W. Hamacher, "A new integer programming approach to (restricted) facilities layout problems allowing flexible facility shapes", Report 1992-3, Department of Management Systems, University of Waikato (1992).

M.R. Garey and D.S. Johnson, *Computers and Intractability: a Guide to the Theory of NP-completeness*, W.H. Freeman & Co., 1979.

H.W. Hamacher, and K. Joernstern, "Optimal Relinquishments", Technical Report, Norwegian School of Economics and Business Administration (1992).

F. Maffioli, "Finding a best subtree of a tree", Rep. n. 91.041, Dip. di Elettronica, Politecnico di Milano, (1991).

R.C. Prim, "Shortest Connection Networks and Some Generalizations", *Bell Syst. Techn. J. 36* (1957) 1389-1401.

Appendix A – Finding a best Subtree of a Tree

From now on we assume that the given simple graph $G = (V, E)$ is a tree on $n = | V |$ nodes and with $| E | = n - 1$ edges; edge (i, j) has weight w_{ij}.

The choice of any node r of G as a *root* induces a unique orientation of the arcs of G from (or into) the root.

We shall use the orientation *from* the root and continue to indicate by G the directed tree obtained with this operation.

Definition A.1 The root of G is the *ancestor* of all other nodes of G.

Definition A.2 A node j is a *child* of node i iff $(i, j) \in G$.

Definition A.3 A node j is a *descendant* of node i iff there exists in G a (directed) path from i to j.

Definition A.4 The subtree $P(i)$ of G containing all descendants of node i is called a *downtree* of G (rooted at i).

The leaves of G originate downtrees with zero arcs, whereas G itself is the downtree rooted at r.

For any downtree P of G let us indicate with $z^P(h)$ the value of a minimum weight subtree of P having exactly h arcs. Whenever h is greater than the number of arcs of P we set $z^P(h) = \infty$ by definition.. We also indicate by $z_1^P(h)$ and $z_o^P(h)$ the values of an optimum subtree of P with exactly h arcs constrained respectively to contain and not to contain the root of P.

Obviously for any h,

$$z^P(h) = \min\{z_o^P(h),\ z_1^P(h)\}. \tag{A.1}$$

Note that when P is a downtree of a leaf of G

$$z_o^P(h) = \infty,\ h \geq 0,$$

$$z_1^P(h) = \begin{cases} 0, & \text{if } h = 0 \\ \infty, & \text{if } h > 0 \end{cases}.$$

We need to define two elementary operations on downtrees, that we call ADDFATHER and MERGE, respectively.

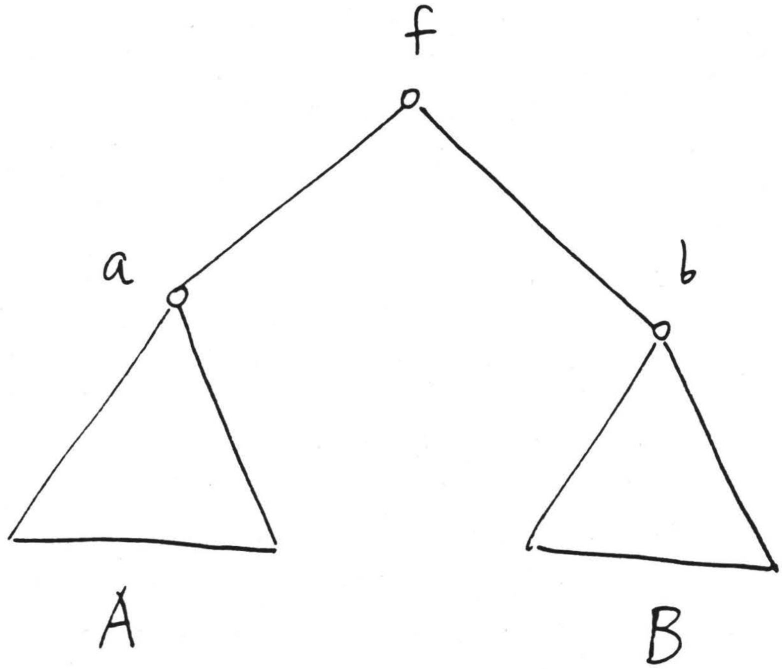

Figure A.1

During operation ADDFATHER (a, b, f) we replace two downtrees A and B whose roots a and b are children of the same node f (their *father*), by a single downtree F having root f (see Figure A.1).

It is easy to see that the following relations allow to compute the values of z_o, z_1 (and therefore from (A.1), the value of z) for F once we know their values for A and B. For $h \geq 0$, we have

$$z_o^F(h) = \min\{z^A(h), z^B(h)\} \tag{A.2}$$

$$z_1^F(h) = \min\{w_{fa} + z_1^A(h-1), w_{fb} + z_1^B(h-1), w_{fa} + w_{fb} + \eta\} \tag{A.3}$$

where

$$\eta := \min\{z_1^A(\alpha) + z_1^B(\beta) : \alpha + \beta = h - 2\}.$$

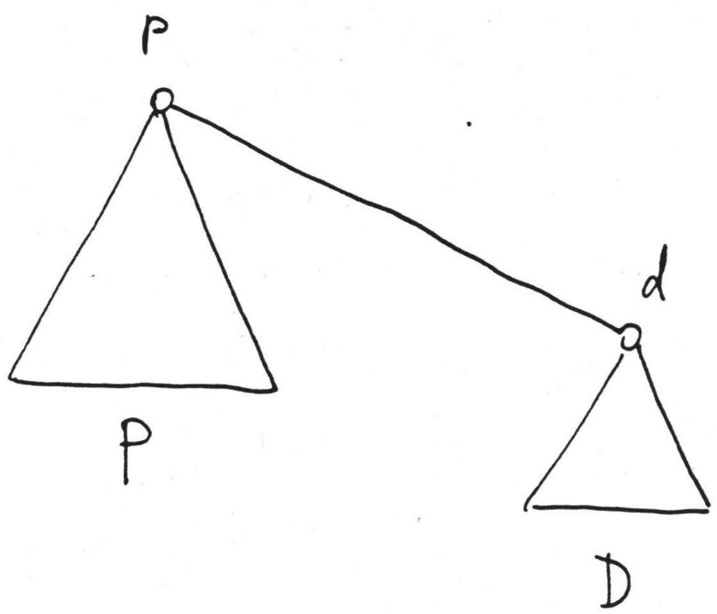

Figure A.2

Operation MERGE (p, d) is visualized in Figure A.2 and consists in adding a downtree D whose root d is a child of node p to the downtree P having p as its root, thus forming a single bigger downtree Q.

For this operation we have

$$z_o^Q(h) = \min\{z_o^P(h), z^D(h)\} \tag{A.4}$$

and

$$z_1^Q(h) = \min\{z_1^P(h),\ w_{pd} + \vartheta\} \tag{A.5}$$

where

$$\vartheta := \min\{z_1^P(\alpha) + z_1^D(\beta) : \alpha + \beta = h - 1\}.$$

A third, simpler operation, which could also be considered as a particular case of ADDFATHER, is depicted in Figure A.3.

We call this operation ASCEND (ℓ, c) and we have (see Figure A.3 for the meaning of symbols)

$$z_o^L(h) = z^C(h), \tag{A.6}$$

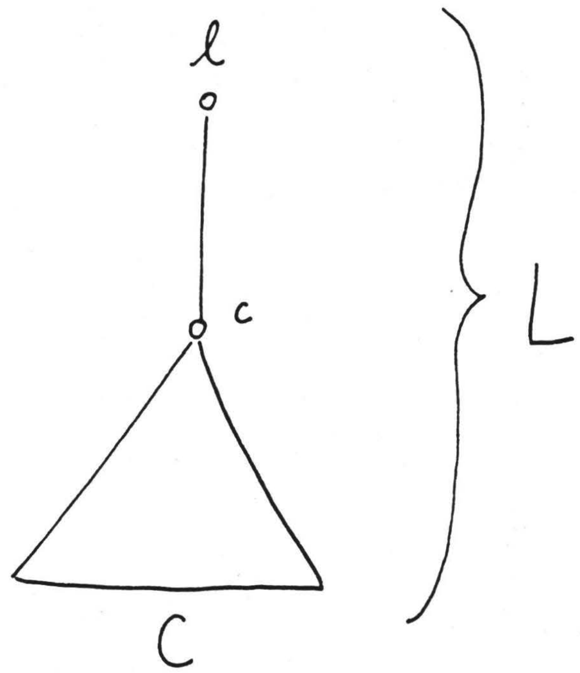

Figure A.3

$$z_1^L(h) = w_{\ell c} + z_1^C(h-1). \tag{A.7}$$

The properties of downtrees considered sofar allow to implement an algorithm for the optimum k-cardinality tree problem when G is a tree. During the algorithm each downtree takes the name of its root, and is characterized by an associated table with 3 columns and $k+1$ rows. The first, second and third column stores the values of $z_o(h)$, $z_1(h)$ and $z(h)$ respectively. Each row corresponds to a different value of h, from $h=0$ to $h=k$. At the beginning the set of downtrees contains only leaves for which the above mentioned table is as in Figure A.4.

z_o	z_1	z	
∞	0	0	$h=0$
∞	∞	∞	$h=1$
	$\vdots$		
∞	∞	∞	$h=k$

Figure A.4

```
begin
  choose a node of G as the root r and orient G;
  initialize the set of downtrees S to contain a downtree j for each leaf j of G;
  each downtree is characterized by a table as in Figure A.4;
  repeat
    while ∃ downtree i whose root is the only child of node j do
      ASCEND (j, i);
    while ∃ downtrees a and b whose roots have the same father f do
      ADDFATHER (a, b, f);
    while ∃ downtree d whose root is child of the root of downtree p do
      MERGE (p, d)
  until | S |= 1 ;
  output z^r(k)
end
```

Figure A.5

The algorithm is sketched in Figure A.5 and it is easy to see that its worst case complexity is $O(k^2\ n)$.

We recently became aware that an algorithm similar but with a higher complexity, was independently introduced in Faigle and Kern (1990).