Facetness of Partition Inequalities of the *p−vertex* Spanning Subtree Polytope of a Graph

Belko Soumana Boubacar, Mamane Souleye Ibrahim*
Department of Mathematics and Computer Science, University Abdou Moumouni, Niamey, Niger.

* Corresponding author. Tel.: 0022790392320; email: ibrah_dz@yahoo.fr
Manuscript submitted November 28, 2022; revised January 6, 2023; accepted February 1, 2023.
doi: 10.17706/ijapm.2023.13.3.24-33

Abstract: Given an undirected graph $G = (V,E)$, with $|V| = n$, we consider an integer linear description of the polytope $P_T(G)$ of *p−vertex* spanning subtrees. A *p−vertex* spanning subtree is a subtree that spans $p < n$ vertices of G. The linear description of the polytope $P_T(G)$ is mainly based on well known partition inequalities. The purpose of this paper is to study the facetness of partition inequalities of the polytope $P_T(G)$. In a different approach as what is usually done, we first address constructive algorithms generating *p−vertex* spanning subtrees that incidence vectors are affinely independent. After, we apply such algorithms to show the facetness of these partition inequalities of $P_T(G)$.

Keywords: Graph, algorithm, valid inequality, facet, polytope

1. Introduction

Given the undirected graph $G = (V,E)$, where V is the set of vertices, E the set of edges and such that $|V| = n$. A *p−vertex* spanning subtree of G is a tree of G that spans $p < n$ vertices. Consider the collection θ of all *p−vertex* spanning subtrees of G. For some $T \in \theta$, the incidence vector x of the *p−vertex* spanning subtree is defined as follow:

For all $e \in E$, $x_e = \begin{cases} 1, & e \in T \\ 0, & e \notin T \end{cases}$, for some T in θ.

Assume that each edge, $e \in E$, has a weight $w(e) \in \mathbb{R}_+$, the *p−vertex* spanning subtree problem (*p−VSSP* for short) consists, given p, to find a *p−vertex* spanning subtree T^* with minimum total weight. The total weight of a tree is the sum of the weight of its edges. We denote by *p−VSSP(G)*, the convex hull of incidence vectors of *p−vertex* spanning subtrees of G. Formally, we have:

$$p\text{−}VSSP(G) = conv\{x \in \{0,1\}^E : \text{for all } T \in \theta\}$$

That is *p−VSSP* can be defined as:

$$minimize\{wx : x \in p\text{−}VSSP(G)\}$$

p−VSSP is NP-hard. Indeed, Fischetti and Hamacher *et al.* [1] show that the Steiner tree, known to be strongly NP-hard [2], can be reduced to *p−VSSP*.

In literature, the *p−vertex* spanning subtree problem is also called the *k−cardinality tree problem*. Several studies have been conducted in the literature on the subject. The first Integer Linear Program (ILP) formulation of the *p−vertex* spanning subtree problem is due to Fischetti and Hamacher *et al.* [1]. To define the model, authors consider two types of binary variables, say x_e and y_v, associated to the edge e and the vertex v of the graph, respectively. They also discuss the facial structure of the problem polytope. Maculan and Plateau *et al*. [3] present a flow based linear formulation of the *p−vertex* spanning subtree problem. In their formulation, they first transform the undirected graph into a digraph and add an artificial vertex which may play the role of a root vertex. A vertex in a digraph, say r, is called a root vertex if there exists at least a simple path between the root vertex r and all other vertices of the digraph. After, in addition to binary variables associated to vertices and edges of the graph, they also consider flow continuous positive variables $f_{u,v}^{w} \geq 0$ that define the flow that passes by the arc (u, v) and is destined to the sink vertex w. To efficiently solve the problem using a Branch and Cut algorithm, Chimani and Kandyba *et al*. [4] consider what they call the *k−cardinality arborescence problem*. Indeed, as in Maculan and Plateau *et al.* [3], they also transform the undirected graph that represents an instance of the *p−vertex* spanning subtree problem into a directed instance and create an artificial root vertex. Chimani and Kandyba *et al.* [4] show that their formulation is equivalent to the one introduced by Fischetti and Hamacher *et al.* [1] from a polyhedral point of view.

A partition $\pi = (V_1, V_2, ..., V_r)$ of V is such that $V_1 \cup V_2 \cup ... \cup V_r = V$ and $V_j \cap V_{j'} = \emptyset, \forall j, j' \in \{1, ..., r\}$. Given a partition $\pi = (V_1, V_2, ..., V_r)$, we denote by $\delta(V_1, V_2, ..., V_r)$ the set of edges with endpoints in two different components.

Consider the following linear description of *p−VSSP* [5]. Given a p−vertex spanning subtree T of the convex hull *p−VSSP(G)*, its incidence vector x satisfies the following inequalities:

$$x(E) = p - 1, \tag{1}$$

$$x(\delta(\pi)) \geq 1, \forall \pi, \tag{2}$$

$$x_e \in \{0, 1\}, e \in E. \tag{3}$$

where π is a partition of the vertex set V, with $1 \leq |Vj| \leq p - 1, j = 1, ..., r$.

Constraint (1) guarantees the cardinality condition. Indeed, *p−vertex* spanning subtrees may contain $(p-1)$ edges. Constraints (2) are partition inequalities that permit simultaneously to eliminate cycles in any solution of *p−VSSP(G)* and to make such a solution connected. Constraints of type (3) are integrality constraints. We denote by $P_T(G)$ the p − vertex spanning subtree polytope defined by constraints (1), (2) and trivial constraints $x_e \geq 0, \forall e \in E$ and $x_e \leq 1, \forall e \in E$.

Grotschel and Monma [6] showed that inequalities (1), (2) and trivial inequalities $x_e \geq 0$ and $x_e \leq 1$ suffice to describe the spanning tree polytope. Note that in this case $p = n$ and the subsets V_i, $i = 1, ..., r$ that form the partition are such that $1 \leq |V_i| \leq n - 1$.

Partition inequalities appear in the linear description of survivable network problem. Its facetness has been extensively studied. Partition inequalities have been firstly introduced by Grotschel and Monma *et al*. [7] for the link survivable network problem with all vertex types equal to 2. These valid inequalities generalize the so-called cut inequalities. Partition inequalities are defined as follows:

Let $\pi = (V_1, V_2, ..., V_p)$, $p \geq 3$ be a partition of V.

$$x(\delta(\pi)) \geq \begin{cases} p - 1 \; if \; I_2 = \emptyset \\ p \quad \text{otherwise} \end{cases}$$

where $I_2 = \{i: con(V_i = 2, i = 1, \dots, p)\}$, $con(V_i) = min(r(V_i), r(V \setminus V_i))$ and $r(V_i) = max(r(u) : u \in V_i)$. $r(u)$ is the vertex u connectivity type.

In the case of the k edge connectivity subgraph problem, partition inequalities are as follows:

$$x(\delta(\pi)) \geq \left\lceil \frac{kp}{2} \right\rceil$$

Grotschel and Monma *et al.* [8] gave sufficient conditions for the partition inequalities to be facet defining for the link survivable network problem. For other cases of linear description of the survivable network problem with partition inequalities, one can refer to [9–12].

Note also the importance of valid inequalities (facets) to solve combinatorial optimization problems, as the *p−vertex* subtree problem, formulated as linear programs. For an exact resolution or in view to strengthen the relaxations of such linear programs, branch and cut and cutting plane algorithms are usually used. Chimani and Chimani *et al.* [4] present a branch and cut algorithm for an exact resolution of the *p−vertex* subtree problem. The linear formulation of the *p−vertex* subtree problem considered in [4] is defined on the space of variables associated both to vertices and edges of the graph. Ibrahim and Maculan *et al*. [13, 14] consider some family of valid inequalities of the *s−t* shortest path problem. Using lifting procedures, they show that these lifted valid inequalities contribute significantly to strengthen the linear relaxation of a flow based formulation of the shortest path problem. Ibrahim and Maculan *et al*. [15] address an efficient cutting plane algorithm to solve the minimum weighted dicycle problem in planar digraphs.

In this paper, we study the facetness of the well known partition inequalities of a new linear description of the *p−vertex* spanning subtree polytope $P_T(G)$ [5]. Such a polytope is mainly based on partition inequalities. In the case of the p−vertex spanning subtree problem, these partition inequalities are as follows:

$$x(\delta(\pi)) \geq 1$$

For this, in a different approach as what is usually done, we first address constructive algorithms generating *p−vertex* spanning subtrees that incidence vectors are affinely independent and satisfied partition inequalities with equality. After, resorting to these constructive algorithms, we discuss the facetness of inequalities that define the subtree polytope.

The paper is organized as follows. In Section 2, we present some theoretical results that are useful in the sequel. That are some technical lemmas. We also consider the new integer linear program of *p−VSSP* introduced in [5]. Such an ILP-program is defined on the space of variables associated with the edges of the graph. In Section 3, we discuss the facetness of partition inequalities of the polytope $P_T(G)$. For this purpose, we first devise a constructive algorithm to generate p−subtrees with affinely independent incidence vectors. Moreover, these incidence vectors satisfied a given valid inequality with equality. After, applying these algorithms, we prove the facetness of partition inequalities of the *p−vertex* spanning subtree polytope $P_T(G)$ defined by constrainst (1), (2) and trivial constraints $x_e \geq 0, \forall e \in E$ and $x_e \leq 1, \forall e \in E$.

In the rest of this section, we give further definitions and notations. Throughout the paper, we deal with the complete undirected graph $G = (V,E)$, with $V = \{1, 2, ..., n\}$, $E = \{e_{k,l} = (k, l), 1 \leq k \leq n-1, 2 \leq l \leq n\}$. That is, we have $|E| = m = \frac{n(n-1)}{2}$. We denote an edge as a pair of vertices. Let τ_i be the incidence vector of the *p−vertex* spanning subtree T_i. $E(Ti)$ is the edge set of the subtree T_i and $|E(Ti)|$ is the number of edges of $E(Ti)$. Recall that vectors $\tau_1, \tau_2, ..., \tau_q$ are said to be affinely independent, if there exists some coefficients λ_i, $1 \leq i \leq q$ such that the unique solution of systems $\sum_{i=0}^{q} \lambda_i \tau_i = 0$ and $\sum_{i=0}^{q} \lambda_i = 0$ is $\lambda_i = 0, i = 1, ..., q$. By $\delta([V_j: V_{j'}])$, we mean the edge set having one of its endpoint in V_j and the other in $V_{j'}$. Given two components, V_j and $V_{j'}$ of a partition π, by $e_{k,l}^{j,j'}$ or $f_{k,l}^{j,j'}$, we define the edge (k, l) such that vertices k and l belong to components V_j and $V_{j'}$, respectively. $e_{k,l}^{j}$ or $f_{k,l}^{j}$ denotes the edge (k, l) with both vertices k and l belonging to the component V_j. We denote by $E(V_i)$ the set of edge having both its endpoints in V_i and $G[V_i] = (V_i, E(V_i))$ the subgraph induced by V_i. The degree $d_G(u)$ of a given vertex u of G is the number of edges having the vertex u as endpoint. We call all leaf vertices u–rooted *p–vertex* subtree T in **G**, a *p–vertex* subtree having the vertex u as a root such that $d_G(u) = p - 1$ and $d_G(v) = 1$, for all others vertices v of T. As an example, subtrees $T_{1,1}$, $T_{1,2}$ and $T_{1,3}$ depicted in Fig. 1 are all leaf vertices 1–rooted *p–vertex* spanning subtrees, with the vertex 1 as a root and $p = 4$.

2. Some Theoretical Results

In what follows, we give some technical lemmas which will be useful in the proof of results of the next section.

Lemma 1: *From the set $Q_\tau = \{\tau_1\}$, where τ_1 is the incidence vector of a p–vertex spanning subtree T_1, by sequentially setting $Q_\tau := Q_\tau \cup \{\tau_i\}$ such that the p–vertex spanning subtree T_i contains an edge e that is not contained by any subtree $T_{i'}$ having its corresponding incidence vector in Q_τ ($\tau_{i'} \in Q_\tau$), then we construct a set Q_τ that elements are affinely independent.*

Proof. Assume that $Q_\tau = \{\tau_1, \tau_2, ..., \tau_q\}$ is a set of affinely independent subtree incidence vectors and consider the subtree T_{q+1}, (with τ_{q+1} as incidence vector), that contains the edge e with the condition that $e \notin E(T_i)$, $i = 1, ..., q$. By applying the definition of affine independence with respect to the set $Q_\tau \cup \{\tau_{q+1}\}$, one can write $\sum_{i=1}^{q} \lambda_i + \lambda_{q+1} = 0$. As $\sum_{i=1}^{q} \lambda_i = 0$, we deduce that $\lambda_{q+1} = 0$. On the other hand, the vectors τ_1, τ_2, ..., τ_q representing the subtrees T_1, T_2, ..., T_q are affinely independent, this proves that the incidences vectors $\tau_1, \tau_2, ..., \tau_q, \tau_{q+1}$ are affinely independent.

Lemma 2: *Consider a set $\{\tau_1, \tau_2, ..., \tau_q\}$ of affinely independent incidence vectors of p–vertex spanning subtrees T_i, $i = 1, ..., q$, constructed according to Lemma 1, that all pass through an edge, say $e_{1,2} = (1, 2)$. If the p–vertex spanning subtrees $T_{q+1}, ..., T_{q+l}$, (with incidence vectors $\tau_{q+1}, ..., \tau_{q+l}$) contain the edge $e_{1,2}$ and $(p-3)$ other edges $e_{1,k} = (1, k)$, $k \in \{3, ..., p-1\}$ such that for each subtree T_{q+j} , $j \in \{1, ..., l\}$, there exists an edge $e_{1,k} = (1, k)$, $3 \le k \le p-1$ with $e_{1,k} \notin E(T_{q+j})$ and $e_{1,k} \in E(Tq+j')$, $j' \in \{1, ..., l\} \setminus \{j\}$. Then incidence vectors $\tau_1, \tau_2, ..., \tau_q, \tau_{q+1}, ..., \tau_{q+l}$ are affinely independent.*

Proof. As vectors $\tau_1, \tau_2, ..., \tau_q$ are affinely independent according to Lemma 1 and the fact that all *p–vertex* spanning subtrees pass through the edge e1,2 = (1, 2), applying the affine independence definition, we have $\sum_{j=1}^{q} \lambda_j + \sum_{j=1}^{l} \lambda_j = 0$. On the other hand, each subtree T_{q+j} is such that there exists an edge $e_{1,k} = (1, k)$, $k \in \{3, ..., p-1\}$ contained by all subtrees $T_{q+j'}$, $j' \in \{1, ..., l\} \setminus \{j\}$ except the subtree T_{q+j} . So, we can write $(p-3)$ equations of the form $\sum_{j' \neq j} \lambda_{q+j'} + \sum_{j=1}^{q} \lambda_j = 0$. This finally implies that $\lambda_{q+j} = 0$, $j = \{1, ..., l\}$ and shows that vectors $\tau_1, \tau_2, ..., \tau_q, \tau_{q+1}, ..., \tau_{q+l}$ are affinely independent.

Lemma 3: *Consider a set $\{\tau_1, \tau_2, ..., \tau_q\}$ of affinely independent incidence vectors of p–vertex spanning sutrees T_i, $i = 1, ..., q$, constructed according to Lemma 1, that all pass through an edge, say $e_{1,2} = (1, 2)$. Let T_{q+1} (with τ_{q+1} as incidence vector) be a p–vertex spanning subtree that do not pass by $e_{1,2}$. Then the incidence vectors $\tau_1, \tau_2, ..., \tau_q, \tau_{q+1}$ are affinely independent.*

Proof. Applying the definition of affine independence, as $\sum_{j=1}^{q+1} \lambda_j = 0$ and the fact that all subtrees pass by the edge $e_{1,2}$, except the p–vertex spanning subtree T_{q+1}, we deduce that $\sum_{j=1}^{q} \lambda_j = 0$ implying that $\tau_{q+1} = 0$. That shows that the incidence vectors $\tau_1, \tau_2, ..., \tau_q, \tau_{q+1}$ are affinely independent.

Theorem 1: *The dimension of $P_T(G)$ is equal to $m - 1$* [5].

In the following section, consider one right hand side partition constraints (2) of the ILP formulation of the *p–VSSP* presented the introduction, we present a constructive algorithm that constructs *(m–1) p–vertex* spanning subtrees that corresponding incidence vectors satisfy a constraint of type (2) with equality and are affinely independent. After, we resort to the algorithm to show that constraints (2) are facet defining of the polytope $P_T(G)$.

3. Facetness of Partition Inequalities of $P_T(G)$

3.1. A Constructive Algorithm

Consider a partition $\pi = (V_1, ..., V_r)$ of V defined such that

(a) $G[V_j], j = 1, . . . , r$ are connected and

(b) $|V_1| = |V_2| = . . . = |V_{r-1}| = p - 1$ and $1 \leq |V_r| \leq p - 1$.

From the condition (b), w.l.o.g, let order vertices of each component $V_j, j = 1, ..., r$ of π as follows:

$$V_j = \{(j - 1)(p - 1) + 1, . . . , j(p - 1)\}, j = 1, . . . , r - 1$$

and

$$V_r = \{(r - 1)(p - 1) + 1, . . . , n\}, j = r.$$

Saying that $x(\delta(\pi)) = 1$ implies that there exists a pair of components $(V_j, V_{j'}), j \in \{1, ..., r - 1\}$ and $j' \in \{j + 1, ..., r\}$ such that $x(\delta([V_j : V_{j'}])) = 1$. Let check solutions with incidence vectors satisfying equations $x(\delta([V_j : V_{j'}])) = 1, j \in \{1, ..., r - 1\}, j' \in \{j + 1, ..., r\}$ that are affinely independent. Such solutions may be among *p–vertex* spanning subtrees that pass through a unique edge $e_{k,l}^{j,j'}$, with

$$(j - 1)(p - 1) + 1 \leq k \leq j(p - 1)$$

and

$$(j' - 1)(p - 1) + 1 \leq l \leq j'(p - 1),$$

where $j' < r$.

If $j' = r$, we have

$$(j' - 1)(p - 1) + 1 \leq l \leq n.$$

As defined above, we recall that $e_{k,l}^{j,j'}$ is the edge (k, l) with k belonging to the component V_j and l to $V_{j'}$.

Consider all leaf vertices $[(j - 1)(p - 1) + 1]$–rooted *p–vertex* spanning subtrees $T_{j,1}$, (with incidence vector $\tau_{j,1}$), $j = 1, ..., r - 1$, such that

$$E(T_{j,1}) = \{((j-1)(p-1)+1, (j-1)(p-1)+2)), ((j-1)(p-1)+1, (j-1)(p-1)+3)), . . . , ((j - 1)(p - 1) + 1, j(p - 1)), ((j - 1)(p - 1) + 1, j(p - 1) + 1))\}.$$

One can check that among all edges of $T_{j,1}$, only $((j - 1)(p - 1) + 1, j(p - 1) + 1))$ belongs to $[V_j : V_{j+1}]$. The fact that $|E(T_{j,1})| = p - 1, j = 1, ..., r - 1$, it remains $(m - (r - 1)(p - 1))$ edges of G that are not yet used.

With respect to all leaf vertices $[(j - 1)(p - 1) + 1]$–rooted p–vertex spanning subtrees $T_{j,1}$, (with incidence vector $\tau_{j,1}$, $j = 1, . . . , r-1$, and the component V_j, $j = 1, . . . , r-1$, unused edges having its endpoints in two different components V_j and $V_{j'}$ are in form

$$f_{k,l}^{j,j'}, k = (j - 1)(p - 1) + 1, l \in \{j(p - 1) + 2, ..., n\}.$$

$$f_{k,l}^{j,j'}, k \in \{(j - 1)(p - 1) + 2, ..., j(p - 1)\}, l \in \{j(p - 1) + 1, ..., n\}.$$

And the unused edges having their both endpoints in a same component are

$$f_{k,l}^{j,j'}, k \in \{(j-1)(p-1)+2, . . . , j(p-1)-1\}, l \in \{(j-1)(p-1)+3, . . . , j(p-1)\} \text{ if } j < r.$$

$$f_{k,l}^{j,j'},\ k \in \{(r-1)(p-1)+1, ..., n-1\},\ l \in \{(r-1)(p-1)+2, ..., n\} \text{ if } j = r.$$

In the following algorithm, from $T_{j,1}$, we contruct subtrees that corresponding incidence vectors are affinely independent.

Algorithm 1: Computation of aff. indpt. subtree vectors of $\{x: \ x(\delta(\pi)) = 1\}$

Data: $G = (V, E)$ A loopless complete undirected graph, with $|V| = n$.

Result: Set $\mathcal{T}$ of subtrees with aff. indpt. vectors that pass by an unique edge of a given V partition π

begin
- **for** $j \leftarrow 1$ *To* $r-1$ **do**
  - $i \leftarrow 2$
  - **for** $k \leftarrow j(p-1)+2$ *To* n **do**
    - Construct $T_{j,i}$ such that $E(T_{j,i}) = (E(T_{j,1}) \setminus \{((j-1)(p-1)+1, j(p-1)+1)\}) \cup \{((j-1)(p-1)+1, k)\}$
    - $i \leftarrow i+1$
  - **end**
- **end**

end

begin
- **for** $j \leftarrow 1$ *To* $r-1$ **do**
  - $i \leftarrow n - j(p-1)+1$
  - **for** $k \leftarrow (j-1)(p-1)+2$ *To* $j(p-1)$ **do**
    - **for** $l \leftarrow k+1$ *To* $j(p-1)+1$ **do**
      - $E(T_{j,i}) = (E(T_{j,1}) \setminus \{((j-1)(p-1)+1, l)\}) \cup \{(k,l)\}$
      - $i \leftarrow i+1$
    - **end**
    - **for** $l \leftarrow j(p-1)+2$ *To* n **do**
      - $E(T_{j,i}) = (E(T_{j,1}) \setminus \{((j-1)(p-1)+1, j(p-1)+1)\}) \cup \{(k,l)\}$
      - $i \leftarrow i+1$
    - **end**
  - **end**
- **end**
- **if** $(|V_r| > 1)$ **then**
  - $j \leftarrow r-1$
  - $i \leftarrow (p-1)[(n-1) - j(p-1)] + \frac{(p-1)(p-2)}{2} + 2$
  - **for** $k \leftarrow j(p-1)+1$ *To* $n-1$ **do**
    - **for** $l \leftarrow k+1$ *To* n **do**
      - **if** $k \leftarrow j(p-1)+1$ **then**
        - $E(T_{j,i}) = (E(T_{j,1}) \setminus \{((j-1)(p-1)+1, j(p-1))\}) \cup \{(k,l)\}$
        - $i \leftarrow i+1$
      - **end**
      - **else**
        - $E(T_{j,i}) = (E(T_{j,1}) \setminus \{((j-1)(p-1)+1, j(p-1)), ((j-1)(p-1)+1, j(p-1)+1)\}) \cup \{((j-1)(p-1)+1, k), (k,l)\}$
        - $i \leftarrow i+1$
      - **end**
    - **end**
  - **end**
- **end**

end

begin
- **for** $j \leftarrow 1$ *To* $r-2$ **do**
  - $i \leftarrow (p-1)[(n-1) - j(p-1)] + \frac{(p-1)(p-2)}{2} + 2$
  - **for** $k \leftarrow (j-1)(p-1)+2$ *To* $j(p-1)$ **do**

45 Construct $T_{j,i}$ such that $E(T_{j,i}) = (E(T_{j,1}) \setminus \{(j-1)(p-1)+1, k\} \cup \{(j(p-1)+1, (j(p-1)+2)\}$
46 $i \leftarrow i+1$
47 **end**
48 **end**
49 $j \leftarrow r-1$
50 **if** $(r=2)$ **then**
51 $i \leftarrow \frac{1}{2}[(n-1)-j(p-1)][n-(j-2)(p-1)] + \frac{(p-1)(p-2)}{2} + 2$
52 **end**
53 **else**
54 $i \leftarrow [(n-1)-j(p-1)][(n-1)-(j-1)(p-1)] + \frac{(p-1)(p-2)}{2} + 2$
55 **end**
56 **if** $(|\ V_r\ | > 1)$ **then**
57 **for** $k \leftarrow (j-1)(p-1)+2$ *To* $j(p-1)-1$ **do**
58 Construct $T_{j,i}$ such that $E(T_{j,i}) = (E(T_{j,1}) \setminus \{(j-1)(p-1)+1, k\} \cup \{j(p-1)+1, j(p-1)+2\}$
59 $i \leftarrow i+1$
60 **end**
61 **end**
62 **else**
63 **for** $k \leftarrow (j-2)(p-1)+2$ *To* $(j-1)(p-1)-1$ **do**
64 Construct $T_{j,i}$ such that $E(T_{j,i}) = (E(T_{j,1}) \setminus \{(j-1)(p-1)+1, j(p-1)+1\} \cup \{(j-1)(p-1)+1, k\}$
65 **end**
66 **end**
67 $\mathcal{T} \leftarrow \{T_{j,i}\}$,
68 with $1 \le i \le (p-1)[(n-1)-j(p-1)] + \frac{p(p-1)}{2}$, $1 \le j \le r-2$.
69 If $j = r-1$, $r > 2$,
70 $1 \le i \le [(n-1)-j(p-1)][(n-1)-(j-1)(p-1)] + \frac{(p+1)(p-2)}{2}$.
71 If $j = r-1$, $r = 2$,
72 $1 \le i \le \frac{1}{2}[(n-1)-j(p-1)][(n-1)-(j-2)(p-1)] + \frac{(p+1)(p-2)}{2}$.
73 **return** $\mathcal{T}$
74 **end**

Theorem 2: *Algorithm 1 constructs (m−1) p−vertex spanning subtrees that incidence vectors satisfy a given partition inequality with equality and are affinely independent.*

Proof. Steps 1-40 of Algorithm 1 construct $[(m-(r-1)(p-2)]$ p−vertex spanning subtrees, $T_{j,i}$, $i = 1, \ldots, (p-1)[(n-1) - j(p-1)] + \frac{(p-1)(p-2)}{2} + 1$ for $j = 1, \ldots, r-2$.

If $j = r-1$ and $r > 2$, $i = 1, \ldots, [(n-1)-j(p-1)][(n-1)-(j-1)(p-1)] + \frac{(p-1)(p-2)}{2} + 1$.

In the case, $r = 2$, we have $i = 1, \ldots, \frac{1}{2}[(n-1)-j(p-1)][n-(j-2)(p-1)] + \frac{(p-1)(p-2)}{2} + 1$.

All these subtrees contain the edge $((j-1)(p-1)+1, (j-1)(p-1)+2)$ and by Lemma 1, its corresponding incidence vectors are affinely independent. Indeed, at each step, the current subtree $T_{j,1}$ obtained from $T_{j,1}$, includes an edge that do not belong to any others previously generated subtrees, $T_{j,I'}$, with $i' = 1, \ldots, i-1$.

Such edges are the ones represented by dashed lines in Figs. 1 and 2 for $n = 8$, $p = 4$ and $r = 3$). We recall that subtrees $T_{j,1}$ are the all leaf vertices $[(j-1)(p-1)+1]$−rooted subtrees with

$$E(T_{j,1}) = \{((j-1)(p-1)+1, (j-1)(p-1)+2), ((j-1)(p-1)+1, (j-1)(p-1)+2), \ldots, ((j-1)(p-1)+1, j(p-1)), ((j-1)(p-1)+1, j(p-1)+1)\}, \text{ with } j = 1, \ldots, r-1.$$

After, by applying steps 41-66 of Algorithm 1, we add to the first $[(m-(r-1)(p-2)]$ already constructed subtrees, $((r-1)(p-2)-1)$ subtrees,

$$T_{j,i},\ i = (p-1)[(n-1)-j(p-1)] + \frac{(p-1)(p-2)}{2} + 2, \ldots, (p-1)[(n-1)-j(p-1)] + \frac{p(p-1)}{2},\ j = 1, \ldots, r-2,$$

$$T_{j,i},\ i = (p-1)[(n-1)-j(p-1)]+ \frac{(p-1)(p-2)}{2} +2, \ldots, [(n-1)-j(p-1)][(n-1)-(j-1)(p-1)] + \frac{(p+1)(p-2)}{2},\ j = r-1,\ r > 2$$

and

$$T_{j,i},\ i = \frac{1}{2}[(n-1)-j(p-1)][n-(j-2)(p-1)] + \frac{(p-1)(p-2)}{2}\ (p-1)(p-2)+2, \ldots, \frac{1}{2}[(n-1)-j(p-1)][n-(j-2)(p-1)] + \frac{(p+1)(p-2)}{2},\ j = r-1,\ r = 2$$

That incidence vectors are affinely independent according to Lemmas 2 and 3, (Fig. 3). Thus, incidence vectors of these corresponding *p−vertex* spanning subtrees are affinely independents. This completes the proof.

Remark 1: Note that the incidence vectors of *p−vertex* spanning subtrees generated from $T_{j,1}$, $T_{j+1,1}$, ..., $T_{r-1,1}$, by applying Algorithm 1, are affinely independent. Indeed, they do not have any edge in common.

3.2. Application

In this paragraph, we feature the fact that the proof of the facetness of partition inequalities is a direct application of Algorithm 1.

Theorem 3 *Let π be a partition of V satisfying conditions (a) and (b) defined above. The inequality $x(\delta(\pi)) \geq 1$ define a facet of $P_T(G)$.*

Proof. By virtue of Theorem 2 and under conditions (a) and (b), it's possible to create *(m−1) p−vertex* spanning subtrees that incidence vectors satisfy an inequality of type (2), $x(\delta(\pi)) \geq 1$, with equality and are affinely independent.

On the other hand, consider the inequality $x(\delta(\pi)) \geq 1$, there exists several subtrees that incidence vectors strictly satisfy $x(\delta(\pi)) \geq 1$. As an example, all p−vertex spanning subtrees that cover *(p−2)* vertices of a component, say V_j, $j \in \{1, ..., r\}$ may have two edges of type $e_{k,l}^{j,j'}$, $k \in V_j$, $l \in V_{j'}$. The incidence vectors of these subtrees are such that $x(\delta(\pi)) = 2 > 1$. This proves that an inequality of type (2) is not an equation and completes the proof.

Example 1 Consider the graph G with $V = \{1, 2, ..., 8\}$, $E = \{(u, v): 1 \leq u \leq 7, 2 \leq v \leq 8, u < v\}$, $n = 8$, $p = 4$, $r = 3$ and $T_{j,1}$ is such that $E(T_{j,1}) = \{((j-1)(p-1)+1, (j-1)(p-1)+2), ((j-1)(p-1)+1, (j-1)(p-1)+2), ..., ((j-1)(p-1)+1, j(p-1)), ((j-1)(p-1)+1, j(p-1)+1)\}$, $j = 1, ..., r-1$. Figs. 1–3 shows the *p−vertex* subtrees constructed by applying the above constructive procedure that incidence vectors are affinely independent.

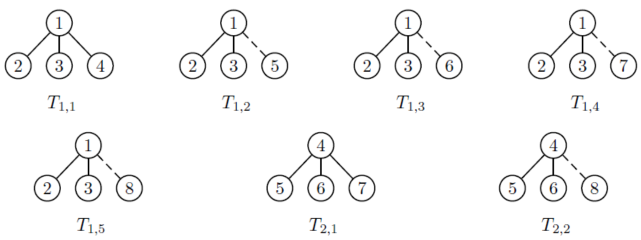
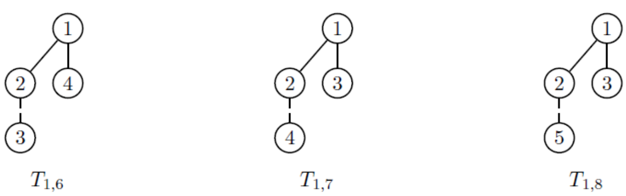


Fig. 1. Subtrees constructed by applying Steps 1–9 of Algorithm 3.



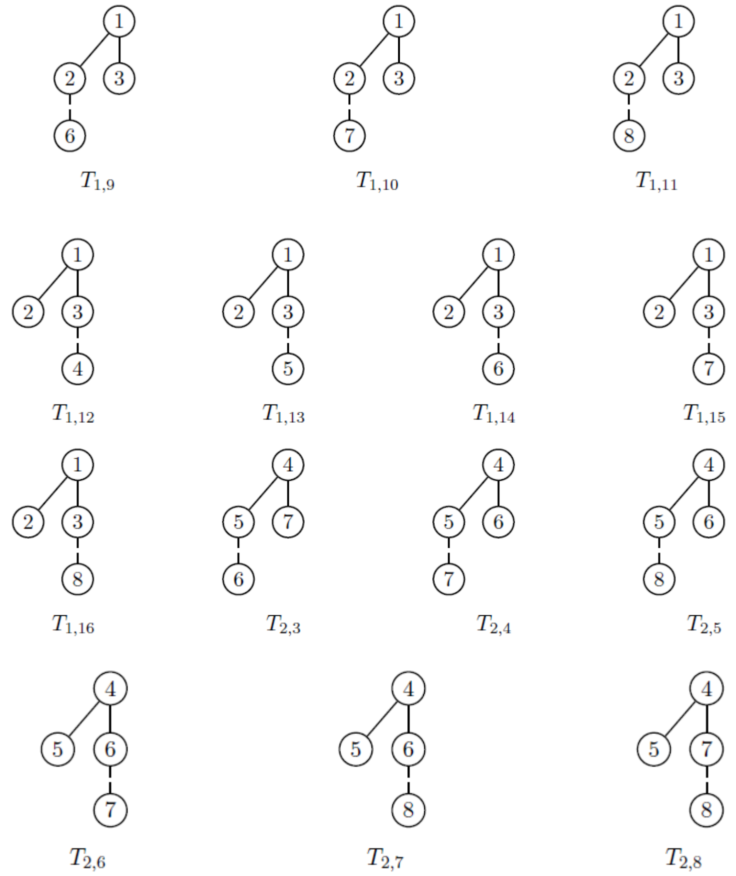


Fig. 2. Subtrees constructed by applying Steps 10–40 of Algorithm 3.

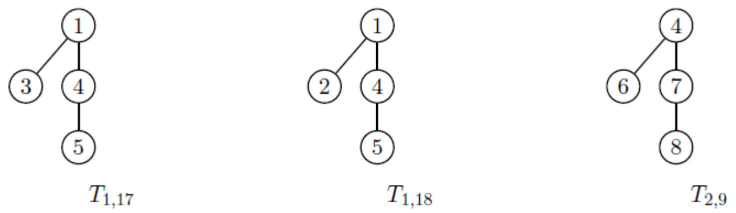


Fig. 3. Subtrees constructed by applying Steps 41–66 of Algorithm 3.

4. Conclusion

We discuss the facetness of partition inequalities of a new linear formulation of the minimum weighted spanning subtree problem. We first address a constructive algorithm that generate a set of subtrees spanning with affinely independent corresponding incidence vectors. Moreover, these incidence vectors satisfy a given partition inequality of the spanning subtree polytope. Consider the polytope associated to this linear formulation and unlike the traditional approach that consists to look for the affine subspace of the subtree polytope, to show the facetness of these partition inequalities, we resort to these algorithms.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Mamane Souleye Ibrahim conducted the research and wrote the paper; Belko Soumana Boubacar contributed to improve the actual form of the paper. All authors had approved the final version.

Acknowledgment

We gratefully acknowledge the referees for their careful reading and constructive comments.

References

[1] Fischetti, M., Hamacher, H. W., Jornsten, K., & Maffioli, F. (1994). Weighted k−cardinality trees: Complexity and polyhedral structure. *Networks, 24*, 11–21.

[2] Garey, M. Y., & Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of Np-Completeness*. W. H. Freeman & Co Ltd.

[3] Maculan, N., Plateau, G., & Lisser, A. (2003). Integer linear models with a polynomial number of variables and constraints for some classical combinatorial optimization problems. *Pesquisa Operacional*, *23*, 161–168.

[4] Chimani, W., Kandyba, M., Ljubic, I., & Mutzel, P. (2008), Obtaining, optimal k−cardinality trees fast. *Journal of Experimental Algorithms, 14*, 27–36.

[5] Ibrahim, M. S., & Boubacar, B. S. (2022). A p−vertex spanning polytope of a graph. *Asian Journal of Science and Technology*, *13(12),* 12350–12358.

[6] Grotschel, M., & Monma, C. L. (1990). Integer polyhedra arising from certain network design problem with connectivity constraints. *SIAM Journal of Discrete mathematics*, *3(4)*, 502–523.

[7] Grotschel, M., Monma, C. L., & Stoer, M. (1992). Facets for polyhedral arising in the design of communication networks with low-connectivity constraints. *SIAM Journal of Optimization*, *2(3)*, 474–504.

[8] Grotschel, M., Monma, C. L., & Stoer, M. (1991). Polyhedral approaches to network survivability. *Reliability of Computer and Communication Networks Series Discrete Mathematics and Computer Science*, *5*, 121–141.

[9] Kerivin, H., & Mahjoub, A. R. (2005). Design of survivable networks: A survey. *Networks*, *46(1)*, 1–21.

[10] Grotschel, M., Monma, C. L., & Stoer, M. (1992). Computational results with a cutting plane algorithm for designing communication networks with low-connectivity constraints. *Operations Research, 40*, 309–330.

[11] Grotschel, M., Monma, C. L., & Stoer, M. (1995). Polyhedral and computational investigations for designing communication networks with high survivability requirements. *Operations Research*, *43(6)*, 1012–1024.

[12] Grotschel, M., Monma, C. L., and Stoer, M. (1995). Design of survivable networks. *Handbook in Operations Research and Management Science, 7*, 617–671.

[13] Ibrahim, M. S., Maculan, N., & Minoux, M. (2014). A note on the NP-hardness of the separation problem on some valid inequalities for the elementary shortest path problem. *Pesquisa Operacional, 34(1)*, 117–124.

[14] Ibrahim, M. S., Maculan, N., & Minoux, M. (2015). Valid inequalities and lifting procedures for the shortest path problem in digraphs with negative cycles. *Optimization Letters*, *9*, 345–357.

[15] Ibrahim, M. S., Maculan, N., & Ouzia, H. (2016). An efficient cutting plane algorithm solving the minimum weighted elementary directed cycle in planar digraphs. *RAIRO-Oper. Res*., *50*, 665–675.