



The k-edge connected subgraph problem: Valid inequalities and Branch-and-Cut

Fatiha Bendali, Ibrahima Diarrassouba, Mohamed Didi Biha, Ali Ridha Mahjoub, Jean Mailfert

► To cite this version:

Fatiha Bendali, Ibrahima Diarrassouba, Mohamed Didi Biha, Ali Ridha Mahjoub, Jean Mailfert. The k-edge connected subgraph problem: Valid inequalities and Branch-and-Cut. International Workshop on Design of Reliable Communication Networks (DRCN), Oct 2007, La Rochelle, France. pp.Pages 1 - 8, 2007, <10.1109/DRCN.2007.4762290>. <hal-00678058>

HAL Id: hal-00678058

<https://hal.archives-ouvertes.fr/hal-00678058>

Submitted on 12 Mar 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**The k -edge connected subgraph
problem: Valid inequalities and
Branch-and-Cut**

F. Bendali¹, I. Diarrassouba², M. Didi
Biha³, A. R. Mahjoub⁴, J. Mailfert⁵

Research Report LIMOS/RR-07-01

16 avril 2007

¹bendali@isima.fr

²diarrass@isima.fr

³mohamed.didi-biha@univ-avignon.fr

⁴ridha.mahjoub@math.univ-bpclermont.fr

⁵mailfert@isima.fr

Abstract

In this paper we consider the k -edge connected subgraph problem from a polyhedral point of view. We introduce further classes of valid inequalities for the associated polytope, and describe sufficient conditions for these inequalities to be facet defining. We also devise separation routines for these inequalities, and discuss some reduction operations that can be used in a preprocessing phase for the separation. Using these results, we develop a Branch-and-Cut algorithm and present some computational results.

Keywords: k -edge connected subgraph, polytope, facet, separation, reduction operations, Branch-and-Cut.

1 Introduction

One of the main concerns when designing telecommunication networks is to compute network topologies that provide a sufficient degree of survivability. Survivable networks must satisfy some connectivity requirements that is, networks that are still functional after the failure of certain links. As pointed out in [30] (see also [28]), the topology that seems to be very efficient (and needed in practice) is the uniform topology, that is to say that corresponding to networks that survive after the failure of $k - 1$ or less edges, for some $k \geq 2$. The 2-connected topology ($k = 2$) provides an adequate level of survivability since most failure usually can be repaired relatively quickly. However, for many applications, it may be necessary to provide a higher level of connectivity. In this paper, we consider this variant of the survivable network design problem.

A graph $G = (V, E)$ is called k -edge connected (where k is a positive integer) if for every pair of nodes $i, j \in V$, there are at least k edge-disjoint paths between i and j . Given a graph $G = (V, E)$ and a weight function w on E that associates with an edge $e \in E$ the weight $w(e) \in \mathbb{R}$, the k -edge connected subgraph problem (k ECSP for short) is to find a k -edge connected spanning subgraph $H = (V, F)$ of G such that $\sum_{e \in F} w(e)$ is minimum.

The k ECSP is NP -hard for $k \geq 2$ ([20]). When $k = 1$, the k ECSP is nothing but the minimum spanning tree problem, and can be solved in polynomial time. The k ECSP has been extensively studied when $k = 2$ [4, 17, 19, 28, 29, 30, 31, 32, 33]. It has, however, received a little attention when $k \geq 3$.

In this paper we consider the k -edge connected subgraph problem from a polyhedral point of view. We introduce further classes of valid inequalities for the associated polytope, and describe sufficient conditions for these inequalities to be facet defining. We also devise separation routines for these inequalities, and discuss some reduction operations that can be used in a preprocessing phase for the separation. Using these results, we develop a Branch-and-Cut algorithm and present some computational results.

Given a graph $G = (V, E)$ and an edge subset $F \subseteq E$, the 0-1 vector $x^F \in \mathbb{R}^E$ such that $x^F(e) = 1$ if $e \in F$ and $x^F(e) = 0$ if $e \in E \setminus F$ is called the *incidence vector* of F . The convex hull of the incidence vectors of the edge sets of the k -edge connected subgraphs of G , denoted by k ECSP(G), is called the *k -edge connected subgraph polytope* of G .

Let $G = (V, E)$ be a graph. Given $w : E \rightarrow \mathbb{R}$ and F a subset of E , $w(F)$ will denote $\sum_{e \in F} w(e)$. For $W \subseteq V$, we let $\overline{W} = V \setminus W$. If $W \subset V$ is a node subset of G , then the set of edges that have only one node in W is called a *cut* and denoted by $\delta(W)$. We will write $\delta(v)$ for $\delta(\{v\})$. If x^F is the incidence vector of the edge set F of a k -edge connected spanning subgraph of G , then x^F satisfies the following

inequalities :

$$x(e) \geq 0 \quad \text{for all } e \in E, \quad (1)$$

$$x(e) \leq 1 \quad \text{for all } e \in E, \quad (2)$$

$$x(\delta(W)) \geq k \quad \text{for all } W \subset V, W \neq V, W \neq \emptyset. \quad (3)$$

Conversely, any integer solution of the system defined by inequalities (1)-(3) is the incidence vector of the edge set of a k -edge connected subgraph of G . Constraints (1) and (2) are called *trivial inequalities* and constraints (3) are called *cut inequalities*. We will denote by $P(G, k)$ the polytope given by inequalities (1)-(3).

The k ECSP has been studied by Grötschel and Monma [23] and Grötschel et al. [24] within the framework of a more general survivability model. In particular, Grötschel and Monma [23] studied the dimension of the polytope associated with that model as well as some basic facets. It follows from their results that k ECSP is full dimensional if G is $(k + 1)$ -edge connected. In [24], Grötschel et al. studied further facets and polyhedral aspects of that model and devised cutting plane algorithms along with some computational results are discussed. A complete survey of that model and related network design problems can be found in [28].

In [7], Chopra studies the k -edge connected subgraph problem for k odd when multiple copies of an edge may be used. In particular, he characterizes the associated polyhedron for outerplanar graphs (a graph is outerplanar if it can be drawn in the plane on a cycle with none crossing chords). This polyhedron has been previously studied by Cornuéjols et al. [8]. They showed that if a graph is series-parallel (a graph is series-parallel if it can be obtained from a single edge by iterative application of the two operations : (i) addition of a parallel edge, and (ii) subdivision of an edge) and $k = 2$, then the polyhedron is completely described by the nonnegativity and cut inequalities. In [14], Didi Biha and Mahjoub give a complete description of k ECSP(G) for all k when G is series-parallel. In particular, they show that if G is series-parallel and k is even, then k ECSP(G) = $P(G, k)$. Didi Biha and Mahjoub study in [13] the extreme points of $P(G, k)$. They introduce an ordering on the fractional extreme points of $P(G, k)$ and describe some structural properties of the minimal extreme points with respect to that ordering. Using these results, they give sufficient conditions for $P(G, k)$ to be integral.

Much work has been done on 2ECSP(G). In [31], Mahjoub shows that if G is series-parallel then 2ECSP(G) is completely described by the trivial and cut inequalities. This has been generalized by Baiou and Mahjoub [3] to the Steiner 2-edge connected subgraph polytope, and Didi Biha and Mahjoub [15] to the Steiner k -edge connected subgraph polytope for k even. Mahjoub [31] introduces a general class of valid inequalities for 2ECSP(G). Boyd and Hao [6] describe a class of "comb inequalities" which are valid for 2ECSP(G). This class, as well as that introduced by Mahjoub [31], are special cases of a more general class of inequalities given by Grötschel et al. [24] for the general survivable network polytope. In [4], Barahona and Mahjoub characterize the polytope 2ECSP(G) for the class of Halin graphs. Kerivin et al. [29] describe a general class of valid inequalities for 2ECSP(G) that

generalizes the so-called F -partition inequalities [31]. They also develop a Branch-and-Cut algorithm for the 2ECSP, based on these inequalities and trivial and cut inequalities. In [5], Bienstock et al. describe structural properties of the optimal solution of k ECSP when the weight function satisfies the triangle inequalities (*i.e.* $w(e_1) \leq w(e_2) + w(e_3)$ for every three edges e_1, e_2, e_3 defining a triangle). In particular, they show that every node of a minimum weight k -edge connected subgraph have degree k or $k + 1$. This generalizes results given by Monma et al. [33] for the case when $k = 2$. In [9, 10], Coullard et al. study the Steiner 2-node connected subgraph problem. They devise in [9] a linear time algorithm for this problem on some special classes of graphs. And in [10], they characterize the dominant of the polytope associated with this problem on the graphs which do not have K_4 (the wheel on four nodes) as a minor. In [18], Fonlupt and Naddef characterize the class of graphs for which the system given by inequalities (1) and (3), when $k = 2$, defines the convex hull of the incidence vectors of the tours of G (a tour is a cycle going at least once through each node).

The paper is organised as follows. In the following section we introduce some classes of valid inequalities and describe sufficient conditions for these inequalities to be facet defining. In Section 3, we discuss some graph reduction operations. In Section 4, we describe separation routines for the inequalities described in Section 2 and develop a Branch-and-Cut algorithm for the k ECSP. Our computational results are presented in Section 5, and finally some concluding remarks are given in Section 6.

In the rest of this section we give more definitions and notations. The graphs we consider are finite, undirected, loopless and connected. A graph is denoted by $G = (V, E)$ where V is the *node set* and E is the *edge set*. If $e \in E$ is an edge with endnode u and v , we also write uv to denote e . Given a node subset W , the cut $\delta(W)$ is said to be *proper* if $|W| \geq 2$ and $|V \setminus W| \geq 2$. If W and W' are two disjoint subsets of V , $[W, W']$ will denote the set of edges of G having one endnode in W and the other one in W' . If $\pi = (V_1, \dots, V_p)$, $p \geq 2$, is a partition of V , then we denote by $\delta(\pi)$ the set of edges having their endnodes in different sets. For all $F \subseteq E$, $V(F)$ will denote the set of nodes of the edges of F . For $W \subset V$, we denote by $E(W)$ the set of edges of G having both endnodes in W and $G[W]$ the subgraph induced by W . Given an edge $e = uv \in E$, *contracting* e consists in deleting e , identifying the nodes u and v and in preserving all adjacencies. Contracting a node subset W consists in identifying all the nodes of W and preserving the adjacencies. Given a partition $\pi = (V_1, \dots, V_p)$, $p \geq 2$, we will denote by G_π the subgraph induced by π that is the graph obtained from G by contracting the sets V_i , for $i = 1, \dots, p$. Note that the edge set of G_π is the set $\delta(V_1, \dots, V_p)$. Given a solution $\bar{x} \in P(G, k)$, an inequality $ax \geq \alpha$ is said to be *tight* for \bar{x} if $a\bar{x} = \alpha$.

2 Facets of $k\text{ECSP}(G)$

In this section we present three classes of valid inequalities for $k\text{ECSP}(G)$. We describe some conditions for these inequalities to be facet defining. Separation procedures for these inequalities will be described in Section 4.

2.1 Odd path inequalities

Let $G = (V, E)$ be a $(k + 1)$ -edge connected graph and $\pi = (W_1, W_2, V_1, \dots, V_{2p})$ a partition of V with $p \geq 2$. Let $I_1 = \{4r, 4r + 1, r = 1, \dots, \lfloor \frac{p}{2} \rfloor - 1\}$ and $I_2 = \{2, \dots, 2p - 1\} \setminus I_1$. We say that π induces an *odd path configuration* if

- 1) $|[V_i, W_j]| = k - 1$ for $(i, j) \in (I_1 \times \{1\}) \cup (I_2 \times \{2\})$,
- 2) $|[W_1, W_2]| \leq k - 1$,
- 3) $\delta(V_i) = [V_i, W_1] \cup [V_{i-1}, V_i] \cup [V_i, V_{i+1}]$ (resp. $\delta(V_i) = [V_i, W_2] \cup [V_{i-1}, V_i] \cup [V_i, V_{i+1}]$) if $i \in I_1$ (resp. $i \in I_2$) (see Figure 1 for $k = 3$ and p even).

Let $C = \bigcup_{i=1}^{2p-1} [V_i, V_{i+1}]$. As $[V_l, V_t] = \emptyset$, if $|l - t| > 1$, for $l, t \in \{1, \dots, 2p\}$, C can be seen as an odd path of extremities V_1 and V_{2p} in the graph G_π . With an odd path configuration we associate the inequality

$$x(C) \geq p. \quad (4)$$

Inequalities of type (4) will be called *odd path inequalities*.

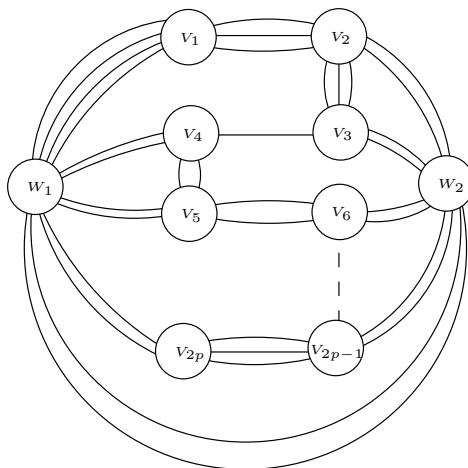


FIG. 1 – An odd path configuration with $k = 3$ and p even.

We have the following.

Theorem 2.1 *Inequality (4) is valid for $k\text{ECSP}(G)$.*

Proof. As $|[V_i, W_j]| = k - 1$ and $x(\delta(V_i)) \geq k$ is valid for $kECSP(G)$, for $(i, j) \in (I_1 \times \{1\}) \cup (I_2 \times \{2\})$, we have

$$x([V_{2s-1}, V_{2s}]) + x([V_{2s}, V_{2s+1}]) \geq 1 \text{ for } s = 1, \dots, p-1, \quad (5)$$

$$x([V_{2s}, V_{2s+1}]) + x([V_{2s+1}, V_{2s+2}]) \geq 1 \text{ for } s = 1, \dots, p-1. \quad (6)$$

By multiplying each inequality (5) (resp. inequality (6)), corresponding to $s \in \{1, \dots, p-1\}$ by $\frac{p-s}{p}$ (resp. $\frac{s}{p}$) and summing these inequalities, we obtain

$$\sum_{i \in I} x([V_i, V_{i+1}]) + \sum_{i \in \bar{I}} \frac{p-1}{p} x([V_i, V_{i+1}]) \geq p-1, \quad (7)$$

where $I = \{2, 4, 6, \dots, 2p-2\}$ and $\bar{I} = \{1, \dots, 2p-1\} \setminus I$.

By considering the cut inequality induced by $W_1 \cup V_1 \cup (\bigcup_{i \in I_1} V_i)$ (resp. $W_1 \cup V_1 \cup$

$(\bigcup_{i \in I_1} V_i) \cup V_{2p}$) if p is odd (resp. even) we have

$$x([W_1, W_2]) + \sum_{i \in \bar{I}} x([V_i, V_{i+1}]) \geq k.$$

As $|[W_1, W_2]| \leq k-1$, it follows that

$$\frac{1}{p} \sum_{i \in \bar{I}} x([V_i, V_{i+1}]) \geq \frac{1}{p}. \quad (8)$$

By summing inequalities (7) and (8) and rounding up the right hand side, we get inequality (4). □

In what follows, we give sufficient conditions for inequality (4) to be facet defining.

Theorem 2.2 *Inequality (4) defines a facet for $kECSP(G)$ if the following hold.*

- 1) *The subgraphs $G[W_1]$, $G[W_2]$ and $G[V_i]$, for $i = 1, \dots, 2p$, are $(k+1)$ -edge connected;*
- 2) *$|[W_1, W_2]| = k-1$, $|[V_1, W_1]| = k$ and $|[V_{2p}, W_1]| = k$ (resp. $|[V_{2p}, W_2]| = k$) if p is even (resp. odd).*

Proof. We will show the result for p even, the proof is similar if p is odd. Let e_i

be a fixed edge of $[V_i, V_{i+1}]$, for $i = 1, \dots, 2p-1$. Let $E_0 = \bigcup_{s=1}^p [V_{2s-1}, V_{2s}]$, $E_1 =$

$\bigcup_{s=1}^{p-1} [V_{2s}, V_{2s+1}]$, $\bar{E} = \delta(\pi) \setminus (E_0 \cup E_1)$, $\tilde{E} = E \setminus (E_0 \cup E_1 \cup \bar{E})$ and $E' = \bar{E} \cup \tilde{E}$.

Inequality (4) can be written as

$$x(E_0) + x(E_1) \geq p. \quad (9)$$

Suppose that conditions 1) and 2) above hold and let us denote inequality (9) by $ax \geq \alpha$. Let $\mathcal{F} = \{x \in k\text{ECSP}(G) \mid ax = \alpha\}$. Let $S = E' \cup \{e_{2s-1}, s = 1, \dots, p\}$. Clearly, S induces a k -edge connected subgraph of G and x^S satisfies (9) with equality. This implies that \mathcal{F} is a proper face of $k\text{ECSP}(G)$. Now suppose that there exists a facet defining inequality $bx \geq \beta$ such that $\mathcal{F} \subseteq \{x \in k\text{ECSP}(G) \mid bx = \beta\}$. We will show that $ax \geq \alpha$ and $bx \geq \beta$ are equivalent. As G is $(k+1)$ -edge connected and thus, $k\text{ECSP}(G)$ is full dimensional, it suffices to show that $b = \rho a$ with $\rho > 0$.

To this end, first observe, from conditions 1) and 2) that $S_f = S \setminus \{f\}$, for $f \in \tilde{E}$ induces a k -edge connected subgraph of G . Moreover, x^{S_f} satisfies (9) with equality. Hence, $ax^{S_f} = \alpha$ and then $bx^{S_f} = \beta$. This implies that $b(f) = bx^S - bx^{S_f} = 0$.

Now, let $e \in [V_{2s-1}, V_{2s}] \setminus \{e_{2s-1}\}$ for some $s \in \{1, \dots, p\}$. Let $S_1 = (S \setminus \{e_{2s-1}\}) \cup \{e\}$. The set S_1 induces a k -edge connected subgraph of G and $ax^{S_1} = \alpha$. It then follows that $bx^{S_1} = \beta$, implying that

$$b(e) = \rho_{2s-1} \text{ for all } e \in [V_{2s-1}, V_{2s}], \text{ for } s = 1, \dots, p, \text{ for some } \rho_{2s-1} \in \mathbb{R}^*. \quad (10)$$

Similarly, for an edge $e \in [V_{2s}, V_{2s+1}] \setminus \{e_{2s}\}$, for some $s \in \{1, \dots, p-1\}$, one can consider the edge sets $S_2 = E' \cup \left(\bigcup_{i=1}^{p-1} \{e_{2i}\}\right) \cup \{e_1\}$ and $S_3 = (S_2 \setminus \{e_{2s}\}) \cup \{e\}$. Since S_2 and S_3 induce k -edge connected subgraphs of G and $ax^{S_2} = ax^{S_3} = \alpha$, it follows that $bx^{S_2} = bx^{S_3}$ and then

$$b(e) = \rho_{2s} \text{ for all } e \in [V_{2s}, V_{2s+1}], \text{ for } s = 1, \dots, p-1, \text{ for some } \rho_{2s} \in \mathbb{R}^*.$$

Consider the edge sets $S_4 = (S_2 \setminus \{e_1\}) \cup \{e_{2s-1}\}$ and $S_5 = (S_2 \setminus \{e_1, e_{2s}\}) \cup \{e_{2s-1}, e_{2s+1}\}$ for some $s \in \{1, \dots, p-1\}$. It is clear that S_4 and S_5 induce k -edge connected subgraphs of G and that $ax^{S_4} = ax^{S_5} = \alpha$. We have that $bx^{S_4} = bx^{S_5} = \beta$, yielding

$$b(e_1) = b(e_{2s}) = b(e_{2s+1}). \quad (11)$$

From (10) and (11), it follows that

$$b(e) = \rho \text{ for all } e \in E_0 \cup E_1, \text{ for some } \rho \in \mathbb{R}^*.$$

Let $e \in [V_i, W_j]$ for $(i, j) \in (I_1 \times \{1\}) \cup (I_2 \times \{2\})$ and $S_6 = (S_2 \setminus \{e_1\}) \cup \{e_{i-1}\}$ (resp. $S_6 = (S_2 \setminus \{e_1\}) \cup \{e_i\}$) if i is even (resp. odd). It is not hard to see that S_6 and $S_6 \setminus \{e\}$ induce k -edge connected subgraphs of G and that their incidence vector satisfy $ax \geq \alpha$ with equality. Hence, $b(e) = bx^{S_6} - bx^{S_6 \setminus \{e\}} = 0$.

For all $e \in [W_1, W_2]$, the edge set $S_7 = S \setminus \{e\}$ induces a k -edge connected subgraph of G and satisfies $ax \geq \alpha$ with equality. Hence $ax^{S_7} = \alpha$ and $bx^{S_7} = bx^S = \beta$. Thus, we obtain $b(e) = 0$ for all $e \in [W_1, W_2]$.

All together, we have then shown that

$$b(e) = \begin{cases} \rho & \text{if } e \in E_0 \cup E_1, \\ 0 & \text{if not.} \end{cases}$$

Thus, $b = \rho a$ with $\rho \in \mathbb{R}$. Since $bx \geq \beta$ defines a facet of $k\text{ECSP}(G)$, one should have that $\rho > 0$, which terminates the proof of the theorem. \square

2.2 Lifting procedure for odd path inequalities

In what follows we are going to describe a lifting procedure for the odd path inequalities. This will permit to extend these inequalities to a more general class of valid inequalities.

A general lifting procedure for inequalities (4) can be described as follows (see [34]). Consider a graph $G = (V, E)$ and a valid inequality $ax \geq \alpha$ for k ECSP(G). Let $G' = (V, E')$ be a graph obtained from G by adding an edge e , that is $E' = E \cup \{e\}$. Then the inequality

$$ax + a(e)x(e) \geq \alpha \quad (12)$$

is valid for k ECSP(G') where $a(e) = \alpha - \gamma$ with $\gamma = \min\{ax \mid x \in k$ ECSP(G') and $x(e) = 1\}$. Moreover, if $ax \geq \alpha$ is facet defining for k ECSP(G), then inequality (12) is also facet defining for k ECSP(G'). Note that if more than one edge have to be added to G , then their lifting coefficients will depend on the order in which they will be added to G . Moreover, if edges $e_1, \dots, e_{k-1}, e_k, \dots, e_t$ are added to G in this order and $a(e_k)$ is the lifting coefficient of e_k with respect to this order, then $a(e_k) \leq a'(e_k)$ where $a'(e_k)$ is the lifting coefficient of e_k in any order $e_{i_1}, \dots, e_{i_{k-1}}, \dots, e_{i_t}$ such that $i_l = l$ for $l = 1, \dots, k-1$ and $i_s = k$ for some $s > k$.

Theorem 2.3 *Let $G = (V, E)$ be a graph and $\pi = (W_1, W_2, V_1, \dots, V_{2p})$, $p \geq 2$, a partition of V which induces an odd path configuration. Let C , I_1 and I_2 be defined as in Section 2.1. Let $U_1 = \bigcup_{i \in I_1} V_i$, $U_2 = \bigcup_{i \in I_2} V_i$ and $W = U_2 \cup V_{2p}$ (resp. $W = U_2$) if p is odd (resp. even). If $G' = (V, E \cup L)$ is a graph obtained from G by adding an edge set L , then the following inequality is valid for k ECSP(G')*

$$x(C) + \sum_{e \in L} a(e)x(e) \geq p, \quad (13)$$

with

$$a(e) = \begin{cases} 1 & \text{if } e \in \left(\bigcup_{j=1,2} [W_j, U_1 \cup U_2] \cup [W_1, W_2] \right) \cap L \text{ or} \\ & e \in \left(\bigcup_{j=1,2p} [V_j, U_1 \cup U_2] \cup \left(([V_1, V_{2p} \cup W_2] \cup [V_{2p}, W_1 \cup W_2]) \cap \delta(W) \right) \right) \cap L; \\ 2 & \text{if } e \in [V_i, V_j] \cap L, \ i, j \in \{2, \dots, 2p-1\} \text{ with } i < j \text{ and } i \text{ even, } j \text{ odd;} \\ \lambda & \text{if } e \in [V_i, V_j] \cap L \text{ with } i, j \in \{2, \dots, 2p-1\}, \ i < j \text{ and } i \text{ odd} \\ & \text{or } i \text{ and } j \text{ have same parity;} \\ 0 & \text{otherwise,} \end{cases}$$

where λ is the lifting coefficient obtained using the lifting procedure above. Moreover, we have $1 \leq \lambda \leq 2$.

Proof. Easy.

□

Observe that the lifting coefficients of the edges other than those between two subsets V_i and V_j such that $i, j \in \{2, \dots, 2p-1\}$, $i < j$, i is odd or i and j have the same parity do not depend on the order of their adding in G . Inequalities (13) will be called *lifted odd path inequalities*. As it will turn out, these inequalities are very useful for our Branch-and-Cut algorithm.

2.3 F -partition inequalities

In [31], Mahjoub introduced a class of valid inequalities for 2ECSP(G) as follows. Let (V_0, V_1, \dots, V_p) , $p \geq 2$, be a partition of V and $F \subseteq \delta(V_0)$ with $|F|$ odd. By adding the inequalities

$$x(\delta(V_i)) \geq 2 \quad \text{for } i = 1, \dots, p, \quad (14)$$

$$-x(e) \geq -1 \quad \text{for } e \in F, \quad (15)$$

$$x(e) \geq 0 \quad \text{for } e \in \delta(V_0) \setminus F, \quad (16)$$

we obtain $2x(\Delta) \geq 2p - |F|$ where $\Delta = \delta(V_0, V_1, \dots, V_p) \setminus F$. Dividing by 2 and rounding up the right hand side lead to

$$x(\Delta) \geq p - \frac{|F| - 1}{2}. \quad (17)$$

Inequalities (17) are called *F -partition inequalities*. These inequalities can be straightforwardly extended for all $k \geq 2$. In fact, in a similar way, one can show that given a partition (V_0, V_1, \dots, V_p) , $p \geq 2$, of V and $F \subseteq \delta(V_0)$, the inequality

$$x(\delta(V_0, V_1, \dots, V_p) \setminus F) \geq \left\lceil \frac{kp - |F|}{2} \right\rceil, \quad (18)$$

is valid for k ECSP(G). Note here that $|F|$ can be either odd or even. Also note that if kp and $|F|$ have the same parity, then the corresponding inequality (18) is implied by the cut and the trivial inequalities.

In what follows, we describe sufficient conditions for inequalities (18) to be facet defining. Theorems 2.4 and 2.5 describe these conditions for k odd and k even, respectively.

Theorem 2.4 *Let $G = (V, E)$ be a graph and $k \geq 3$ an odd integer. Let $\pi = (W, V_1, \dots, V_{2l+1})$,*

U_1, \dots, U_{2l+1}), $l \geq 2$ if $k = 3$ and $l \geq 1$ if $k \geq 5$, be a partition of V such that

- 1) $G[W]$, $G[V_i]$, $G[U_i]$, $i = 1, \dots, 2l+1$, are $(k+1)$ -edge connected;
- 2) $||[W, V_i]|| \geq k - 2$ for $i = 1, \dots, 2l+1$;
- 3) $||[U_i, U_{i+1}]|| \geq \frac{k-1}{2}$, $i = 1, \dots, 2l+1$ (the indices are modulo $2l+1$);
- 4) $||[V_i, V_{i+1}]|| \geq 1$, $i = 1, \dots, 2l+1$ (the indices are modulo $2l+1$);
- 5) $||[V_i, U_i]|| \geq 1$ and $||[V_i, U_{i-1}]|| \geq 1$, $i = 1, \dots, 2l+1$ (for convenience we will let $U_0 = U_{2l+1}$) (See Figure 2 for an illustration with $k = 5$ and $l = 2$).

Let F_i be an edge subset of $[W, V_i]$ such that $|F_i| = k - 2$, $i = 1, \dots, 2l + 1$ and let $F = \bigcup_{i=1}^{2l+1} F_i$. Then the F -partition inequality

$$x(\delta(\pi) \setminus F) \geq l(k + 2) + \left\lceil \frac{k}{2} \right\rceil + 1 \quad (19)$$

induced by π and F , defines a facet of $k\text{ECSP}(G)$.

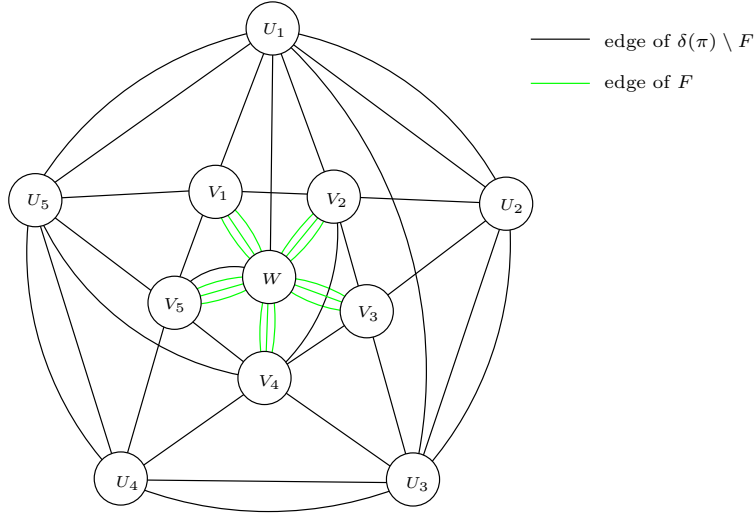


FIG. 2 – An F -partition configuration with $k = 5$

Proof. First observe that, by conditions 1) - 5), G is $(k + 1)$ -edge connected and hence $k\text{ECSP}(G)$ is full dimensional. Let us denote inequality (19) by $ax \geq \alpha$ and let $\mathcal{F} = \{x \in k\text{ECSP}(G) \mid ax = \alpha\}$. Clearly, \mathcal{F} is a proper face of $k\text{ECSP}(G)$. Now suppose that there exists a facet defining inequality $bx \geq \beta$ such that $\mathcal{F} \subseteq \{x \in k\text{ECSP}(G) \mid bx = \beta\}$. We will show that there exists $\rho > 0$ such that $b = \rho a$.

Let e_i be an edge of $[V_i, V_{i+1}]$, $i = 1, \dots, 2l + 1$ (the indices are modulo $2l + 1$) and f_i and f'_i be edges of $[V_i, U_{i-1}]$ and $[V_i, U_i]$, respectively, for $i = 1, \dots, 2l + 1$. Note that $U_0 = U_{2l+1}$. Let T_i be an edge subset of $[U_i, U_{i+1}]$ of $\frac{k-1}{2}$ edges, for $i = 1, \dots, 2l + 1$ (the indices are modulo $2l + 1$).

Let E_0 be the set of edges not in F and having both endnodes in the same element of π . First, we will show that $b(e) = 0$, for all $e \in E_0 \cup F$. To this end, let $i_0 \in \{1, \dots, 2l + 1\}$ and consider the edge sets

$$E_1 = \{e_{i_0+2r}, r = 0, \dots, l\} \cup \{f'_i, i = 1, \dots, 2l + 1\} \cup \left(\bigcup_{i=1}^{2l+1} T_i \right),$$

$$E_2 = E_1 \cup F \cup E_0.$$

It is not hard to see that E_2 induces a k -edge connected subgraph of G . Note that there is $k + 1$ edges incident to V_{i_0} in E_2 . Now, observe that for any edge $e \in F_{i_0}$,

we have that $E'_2 = E_2 \setminus \{e\}$ also induces a k -edge connected subgraph of G . As x^{E_2} and $x^{E'_2}$ belong to \mathcal{F} , it follows that $bx^{E_2} = bx^{E'_2} = \beta$, implying that $b(e) = 0$ for all $e \in F_{i_0}$. As i_0 is arbitrarily chosen, we obtain that $b(e) = 0$ for all $e \in F$. Moreover, as the subgraphs induced by $W, V_1, \dots, V_{2l+1}, U_1, \dots, U_{2l+1}$ are all $(k+1)$ -edge connected, the subgraph induced by $E_2 \setminus \{e\}$ for all $e \in E_0$ is also k -edge connected. This yields as before $b(e) = 0$ for all $e \in E_0$. Thus, $b(e) = 0$ for all $e \in F \cup E_0$.

Next, we will show that $b(e) = \rho$ for all $e \in \delta(\pi) \setminus F$, for some $\rho \in \mathbb{R}$. Let g_i be a fixed edge of T_i and let $T'_i = T_i \setminus \{g_i\}$, $i = 1, \dots, 2l+1$. Consider the edge sets

$$\begin{aligned} E_3 &= \{f_i, f'_i, i = 1, \dots, 2l+1\} \cup \left(\bigcup_{i=1}^l T_{2i} \right) \cup T_{2l+1} \cup \left(\bigcup_{i=0}^{l-1} T'_{2i+1} \right), \\ E_4 &= E_3 \cup F \cup E_0, \\ E'_4 &= (E_4 \setminus g_{2l+1}) \cup \{g_1\}. \end{aligned}$$

Note that $g_1 \notin T'_1$ and thus $g_1 \notin E_4$, and that $g_{2l+1} \in E_4$. One can easily check that E_4 and E'_4 both induce k -edge connected subgraphs of G . Moreover, we have that x^{E_4} and $x^{E'_4}$ belong to \mathcal{F} . Thus, $bx^{E_4} = bx^{E'_4} = \beta$ and therefore we get $b(g_{2l+1}) = b(g_1)$. As g_1 and g_{2l+1} are arbitrary edges of T_1 and T_{2l+1} , respectively, it follows that $b(e) = b(e')$ for all $e \in T_1$ and $e' \in T_{2l+1}$. Moreover, we have that T_1 and T_{2l+1} are arbitrary subsets of $[U_1, U_2]$ and $[U_{2l+1}, U_1]$, respectively. This implies that $b(e) = b(e')$ for all $e \in [U_1, U_2]$ and $e' \in [U_{2l+1}, U_1]$. Consequently, by symmetry, we get

$$\begin{aligned} b(e) &= \rho' \text{ for all } e \in [U_i, U_{i+1}], i = 1, \dots, 2l+1 \text{ (the indices are modulo } 2l+1\text{)} \\ &\text{for some } \rho' \in \mathbb{R}. \end{aligned}$$

Now let

$$E_5 = (E_4 \setminus \{f_1\}) \cup \{e_{2l+1}\}.$$

Clearly, E_5 induces a k -edge connected subgraph of G and x^{E_5} belongs to \mathcal{F} , implying that $bx^{E_4} = bx^{E_5} = \beta$. Hence $b(f_1) = b(e_{2l+1})$. In a similar way, we can show that $b(f'_{2l+1}) = b(e_{2l+1})$. As f_1, f'_{2l+1} and e_{2l+1} are arbitrary edges of $[U_{2l+1}, V_1]$, $[V_{2l+1}, U_{2l+1}]$ and $[V_{2l+1}, V_1]$, respectively, we obtain that $b(e) = \rho''_{2l+1}$ for all $e \in [U_{2l+1}, V_1] \cup [V_{2l+1}, U_{2l+1}] \cup [V_{2l+1}, V_1]$. By exchanging the roles of V_{2l+1}, V_1, U_{2l+1} and V_i, V_{i+1}, U_i , for $i = 1, \dots, 2l$, we obtain by symmetry that

$$\begin{aligned} b(e) &= \rho''_i \text{ for all } e \in [U_i, V_i] \cup [V_i, V_{i+1}] \cup [V_{i+1}, U_i], \\ &i = 1, \dots, 2l+1 \text{ (the indices are modulo } 2l+1\text{)}, \text{ for some } \rho''_i \in \mathbb{R}. \end{aligned} \tag{21}$$

Consider the edge set

$$E'_5 = (E_5 \setminus \{f_1\}) \cup \{e_1\}.$$

Obviously, E'_5 induces a k -edge connected subgraph of G . As x^{E_4} and $x^{E'_5}$ belong to \mathcal{F} , it follows, in a similar way, that $b(e_1) = b(f_1)$. From (21), we have that $\rho''_1 = \rho''_{2l+1}$. By symmetry, it then follows that $\rho''_i = \rho''_j$ for $i, j = 1, \dots, 2l+1$, $i \neq j$, and therefore

$$\begin{aligned} b(e) &= \rho'' \text{ for all } e \in [U_i, V_i] \cup [V_i, V_{i+1}] \cup [V_{i+1}, U_i], \\ &\text{for } i = 1, \dots, 2l+1 \text{ (the indices are modulo } 2l+1\text{)}, \text{ for some } \rho'' \in \mathbb{R}. \end{aligned} \tag{22}$$

Let $e \in ([V_{2l+1}, W] \setminus F_{2l+1}) \cup [V_{2l+1}, V_j]$, $j \in \{1, \dots, 2l+1\} \setminus \{1, 2l, 2l+1\}$. It is not hard to see that $E_6 = (E_4 \setminus \{f'_{2l+1}\}) \cup \{e\}$ induces a k -edge connected subgraph of G and that $x^{E_6} \in \mathcal{F}$. This implies that $bx^{E_6} = bx^{E_4} = \beta$ and hence $b(e) = b(f'_{2l+1})$. By (22), we then obtain that $b(e) = \rho''$ for all $e \in ([V_{2l+1}, W] \setminus F_{2l+1}) \cup [V_{2l+1}, V_i]$ for $i \in \{1, \dots, 2l+1\} \setminus \{1, 2l, 2l+1\}$. By exchanging the roles of V_{2l+1} and V_i , $i = 1, \dots, 2l$, we obtain by symmetry that $b(e) = \rho''$ for all $e \in ([V_i, W] \setminus F_i) \cup [V_i, V_j]$, $i = 1, \dots, 2l+1$ and $j \in \{1, \dots, 2l+1\} \setminus \{i-1, i, i+1\}$ (the indices are taken modulo $2l+1$ and $V_0 = V_{2l+1}$).

For any edge e between U_{2l+1} and either W , U_j , $j \in \{1, \dots, 2l+1\} \setminus \{1, 2l, 2l+1\}$, or V_i , $t \in \{1, \dots, 2l+1\} \setminus \{1, 2l+1\}$, the edge set

$$E_7 = (E_4 \setminus \{f'_{2l+1}, f_1\}) \cup \{e, e_{2l+1}\}$$

induces a k -edge connected subgraph of G . Since x^{E_4} and x^{E_7} belong to \mathcal{F} we have that $bx^{E_7} = b^{E_4} = \beta$ and $b(f'_{2l+1}) + b(f_1) = b(e) + b(e_{2l+1})$. As by (22), $b(f'_{2l+1}) = b(f_1) = b(e_{2l+1}) = \rho''$, we get $b(e) = \rho''$. By exchanging the roles of U_{2l+1} and U_i , $i = 1, \dots, 2l$ we obtain that $b(e) = \rho''$ for all $e \in [U_i, W] \cup [U_i, U_j] \cup [U_i, V_t]$, $i = 1, \dots, 2l+1$, $j \in \{1, \dots, 2l+1\} \setminus \{i, i+1\}$ and $t \in \{1, \dots, 2l+1\} \setminus \{i-1, i, i+1\}$.

As x^{E_2} and x^{E_4} belong to \mathcal{F} , we have that $bx^{E_2} = bx^{E_4} = \beta$. Thus, from (20) and (22), we obtain that $\rho' = \rho'' = \rho$, for some $\rho \in \mathbb{R}$.

All together we obtain that

$$b(e) = \begin{cases} \rho & \text{if } e \in E \setminus (E_0 \cup F), \\ 0 & \text{if not.} \end{cases}$$

Thus, $b = \rho a$ with $\rho \in \mathbb{R}$. Since $bx \geq \beta$ defines a facet of $k\text{ECSP}(G)$, one should have $\rho > 0$, which ends the proof of the theorem. \square

In what follows, we describe sufficient conditions for inequalities (18) to be facet defining when k is even. Consider a graph $G = (V, E)$ and an even integer $k = 2q$ with $q \geq 1$, a *generalized odd-wheel configuration* is given by an integer $l \geq 1$, a set of positive integers $\{p_1, \dots, p_{2l+1}\}$ and a partition $\pi = (V_0, V_i^s, i = 1, \dots, 2l+1, s = 0, \dots, p_i)$ of V such that

- 1) $G[V_0]$ and $G[V_i^s]$ are $(k+1)$ -edge connected, for $s = 1, \dots, p_i$ and $i = 1, \dots, 2l+1$;
- 2) $|[V_i^0, V_{i+1}^0]| \geq 2q$, for $i = 1, \dots, 2l+1$ (the indices are modulo $2l+1$);
- 3) $|[V_i^s, V_i^{s+1}]| \geq 2q$, for $s = 0, \dots, p_i$ and $i = 1, \dots, 2l+1$ (for convenience we will let $V_i^{p_i+1} = V_0$);
- 4) $[V_i^s, V_i^t] = \emptyset$, for $s, t \in \{1, \dots, p_i\}$, $|s-t| > 1$ and $(s, t) \neq (0, p_i+1)$, and $i = 1, \dots, 2l+1$;
- 5) $[V_i^s, V_t^t] = \emptyset$, for $s \in \{1, \dots, p_i\}$, $t \in \{1, \dots, p_t\}$, $i, t \in \{1, \dots, 2l+1\}$, $i \neq t$ (See Figure 3).

Let F_i^0 be an edge subset of $[V_0, V_i^{p_i}]$ of q (resp. $q-1$) edges if q is odd (resp. even)

$$\text{and } F = \bigcup_{i=1}^{2l+1} F_i^0.$$

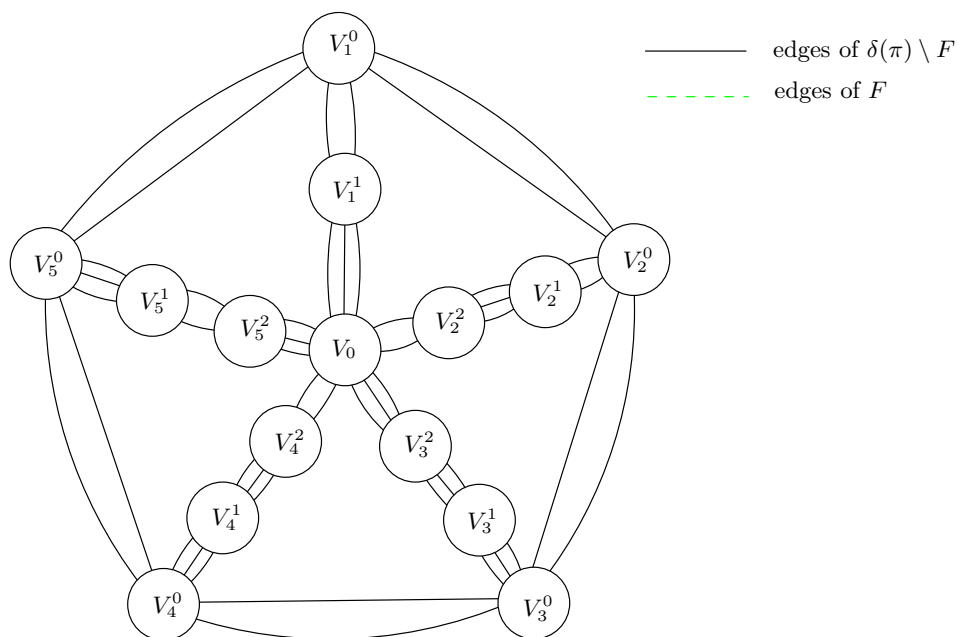


FIG. 3 – A generalized odd-wheel configuration with $k = 4$

With a generalized odd-wheel configuration with q odd (resp. even) we associate the following F -partition inequality induced by the partition π and F

$$\begin{aligned}
 x(\delta(\pi) \setminus F) &\geq q \sum_{i=1}^{2l+1} p_i + ql + \frac{q+1}{2} \\
 (\text{resp. } x(\delta(\pi) \setminus F) &\geq q \sum_{i=1}^{2l+1} p_i + (q+1)l + \frac{q+2}{2}).
 \end{aligned} \tag{23}$$

Inequality of type (23) will be called *generalized odd-wheel inequality*. We have the following theorem given without proof, the proof follows the same lines as that of Theorem 2.4.

Theorem 2.5 *Inequalities (23) define facets of $k\text{ECSP}(G)$.*

2.4 SP -partition inequalities

In [7], Chopra introduces a class of valid inequalities for the $k\text{ECSP}$ when the graph G is outerplanar, k is odd, and each edge can be used more than once. Let $G = (V, E)$ be an outerplanar graph and $k \geq 1$ an odd integer. He showed that if $\pi = (V_1, \dots, V_p)$, $p \geq 2$, is a partition of V , then the inequality

$$x(\delta(V_1, \dots, V_p)) \geq \left\lceil \frac{k}{2} \right\rceil p - 1 \tag{24}$$

is valid for $k\text{ECSP}(G)$.

Didi Biha and Mahjoub [14] extended this result for a general graph and when each edge can be used at most once. They showed that if G is a graph and $\pi = (V_1, \dots, V_p)$, $p \geq 2$, is a partition of V such that G_π is series-parallel, then inequality (24) is valid for $k\text{ECSP}(G)$. They called inequalities (24) *SP-partition inequalities*. They also described necessary conditions for inequality (24) to be facet defining and showed that if G is series-parallel and k is odd then $k\text{ECSP}(G)$ is defined by the trivial, cut inequalities and *SP-partition inequalities*. More necessary conditions for inequalities (24) to be facet defining are given in [12]. In particular, Diarrassouba and Slama [12] show the following.

Theorem 2.6 [12] *Let $G = (V, E)$ be a $(k + 1)$ -edge connected graph and $k \geq 3$ an odd integer. Let $\pi = (V_1, \dots, V_p)$, $p \geq 2$, be a partition of V such that G_π is series-parallel. If the *SP-partition inequality* induced by π defines a facet of $k\text{ECSP}(G)$ different from trivial inequalities then*

- 1) $||[V_i, V_{i+1}]| \geq \lceil \frac{k}{2} \rceil$, for $i = 1, \dots, p$ (the indices are modulo p);
- 2) G_π is outerplanar.

In what follows, we shall give some sufficient conditions for inequalities (24) to be facet defining.

Theorem 2.7 *Let $G = (V, E)$ be a graph and $k \geq 3$ an odd integer. Let $\pi = (V_1, \dots, V_p)$, $p \geq 2$, be a partition of V such that G_π is outerplanar. Then the *SP-partition inequality* induced by π is facet defining for $k\text{ECSP}(G)$ if the following conditions hold*

- 1) $G[V_i]$ is $(k + 1)$ -edge connected for $i = 1, \dots, p$;
- 2) $||[V_i, V_{i+1}]| \geq \lceil \frac{k}{2} \rceil$, $i = 1, \dots, p$ (the indices are modulo p) (See Figure 4 for an illustration with $k = 3$).

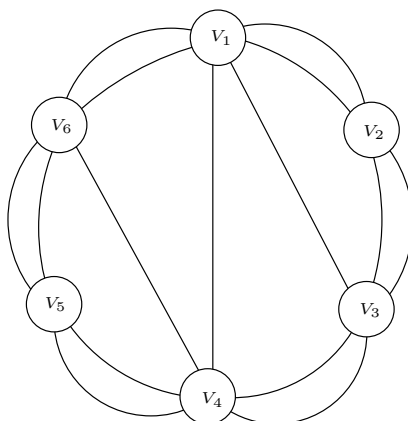


FIG. 4 – An outerplanar configuration with $k = 3$

Proof. Note that since G_π is outerplanar and conditions 1) and 2) hold, G is $(k + 1)$ -edge connected. It then follows that $k\text{ECSP}(G)$ is full dimensional. Let

us denote by $ax \geq \alpha$ the SP -partition inequality induced by π and let $\mathcal{F} = \{x \in k\text{ECSP}(G) \mid ax = \alpha\}$. Clearly \mathcal{F} is a proper face of $k\text{ECSP}(G)$. Now suppose that there exists a facet defining inequality $bx \geq \beta$ different from the trivial inequalities such that $\mathcal{F} \subseteq \{x \in k\text{ECSP}(G) \mid bx = \beta\}$. We will show that there exists $\rho > 0$ such that $b = \rho a$.

Let T_i be an edge subset of $[V_i, V_{i+1}]$, $i = 1, \dots, p$ (the indices are modulo p) of $\frac{k+1}{2}$ edges and let $T'_i = T_i \setminus \{g_i\}$, where g_i is a fixed edge of T_i . Consider

$$\begin{aligned} E_0 &= \bigcup_{i=1}^p E(V_i), \\ E_1 &= \left(\bigcup_{i=1}^{i_0-1} T_i \right) \cup \left(\bigcup_{i=i_0+1}^p T_i \right) \cup T'_{i_0}, \text{ for some } i_0 \in \{1, \dots, p\}, \\ E_2 &= E_1 \cup E_0. \end{aligned}$$

Note that $g_{i_0} \notin T'_{i_0}$ and hence $g_{i_0} \notin E_2$, and $g_{i_0+1} \in E_2$. It is not hard to see that E_2 and $E'_2 = (E_2 \setminus \{g_{i_0+1}\}) \cup \{g_{i_0}\}$ induce k -edge connected subgraphs of G and that x^{E_2} and $x^{E'_2}$ belong to \mathcal{F} . Thus, we have that $bx^{E_2} = bx^{E'_2} = \beta$ and that $b(g_{i_0}) = b(g_{i_0+1})$. As g_{i_0} and g_{i_0+1} are arbitrary edges of T_{i_0} and T_{i_0+1} , respectively, it follows that $b(e) = b(e')$ for all $e \in T_{i_0}$ and $e' \in T_{i_0+1}$. Moreover, since T_{i_0} and T_{i_0+1} are arbitrary subsets of $[V_{i_0}, V_{i_0+1}]$ and $[V_{i_0+1}, V_{i_0+2}]$, respectively, we obtain $b(e) = b(e')$ for all $e \in [V_{i_0}, V_{i_0+1}]$ and $e' \in [V_{i_0+1}, V_{i_0+2}]$, $i_0 = 1, \dots, p$. Consequently, by symmetry, we have that

$b(e) = \rho$, for all $e \in [V_i, V_{i+1}]$, $i = 1, \dots, p$ (the indices are modulo p), for some $\rho \in \mathbb{R}$.

Now let $e \in [V_{i_0}, V_{j_0}]$, $i_0, j_0 \in \{1, \dots, p\}$ with $|i_0 - j_0| > 1$. Note that the indices are modulo p and that, for convenience, we will let $T_0 = T_p$, $T_{-1} = T_{p-1}$ and $T'_0 = T'_p$. Consider

$$\begin{aligned} E_3 &= \left(\bigcup_{i=1}^{i_0-2} T_i \right) \cup \left(\bigcup_{i=i_0+1}^p T_i \right) \cup (T'_{i_0-1} \cup T'_{i_0}) \cup \{e\}, \\ E_4 &= E_3 \cup E_0. \\ E'_4 &= (E_4 \setminus \{e\}) \cup \{g_{i_0}\} \end{aligned}$$

One can easily check that E_4 and E'_4 induce k -edge connected subgraphs of G . Since x^{E_4} and $x^{E'_4}$ belong to \mathcal{F} , it follows that $bx^4 = bx^{E'_4} = \beta$ and that $b(e) = b(g_{i_0}) = \rho$. Thus, we obtain

$b(e) = \rho$, for all $e \in [V_{i_0}, V_{j_0}]$, $i_0, j_0 \in \{1, \dots, p\}$ with $|i_0 - j_0| > 1$.

Consider now the edge set

$$E_5 = E_2 \setminus \{e\}, \text{ for some } e \in E_0.$$

Since $G[V_i]$, $i = 1, \dots, p$, are $(k + 1)$ -edge connected, E_5 induces a k -edge connected subgraph of G . As x^{E_2} and x^{E_5} belong to \mathcal{F} , we have that $bx^{E_2} = bx^{E_5} = \beta$, and thus

$$b(e) = 0, \text{ for all } e \in E_0.$$

All together, we obtain

$$b(e) = \begin{cases} \rho & \text{if } e \in E \setminus E_0, \\ 0 & \text{if not.} \end{cases}$$

Thus, we get $b = \rho a$ with $\rho \in \mathbb{R}$. Since $bx \geq \beta$ defines a facet of $k\text{ECSP}(G)$ different from the trivial inequalities, we have that $\rho > 0$, which ends the proof of the Theorem. \square

Chopra [7] described a lifting procedure for inequalities (24). Let $G = (V, E)$ be a graph and $k \geq 3$ an odd integer. Let $G' = (V, E \cup L)$ be a graph obtained from G by adding an edge set L . Let $\pi = (V_1, \dots, V_p)$ be a partition of V such that G_π is series-parallel. Then the following inequality is valid for $k\text{ECSP}(G')$

$$x(\delta_G(V_1, \dots, V_p)) + \sum_{e \in L \cap \delta_{G'}(V_1, \dots, V_p)} a(e)x(e) \geq \left\lceil \frac{k}{2} \right\rceil p - 1, \quad (25)$$

where $a(e)$ is the length (in terms of edges) of a shortest path in G_π between the endnodes of e , for all $e \in L \cap \delta_{G'}(V_1, \dots, V_p)$. We will call inequality (25) *lifted SP-partition inequality*. Chopra [7] also showed, when G is outerplanar, that inequality (25) defines a facet of $k\text{ECSP}(G')$ if G is maximal outerplanar, that is adding one edge e in G lets the resulting graph not outerplanar. This procedure can be easily extended to the case when each edge can be used at most once.

3 Reduction operations

In this section, we are going to describe some graph reduction operations. These are based on the concept of critical extreme points of $P(G, k)$ introduced by Fonlupt and Mahjoub [17] for $k = 2$ and extended by Didi Biha and Mahjoub [13] for $k \geq 3$.

Before giving these operations, we shall first introduce some notations and definitions. Let $G = (V, E)$ be a graph and $k \geq 2$ an integer. If x is a solution of $P(G, k)$, we will denote by $E_0(x)$, $E_1(x)$ and $E_f(x)$ the sets of edges $e \in E$ such that $x(e) = 0$, $x(e) = 1$ and $0 < x(e) < 1$, respectively. We also denote by $C_d(x)$ the set of degree tight cuts $\delta(u)$ such that $\delta(u) \cap E_f(x) \neq \emptyset$, and by $C_p(x)$ the set of proper tight cuts $\delta(W)$ with $\delta(W) \cap E_f(x) \neq \emptyset$. Let \bar{x} be an extreme point of $P(G, k)$. Thus, there is a set of cuts $C_p^*(\bar{x}) \subseteq C_p(\bar{x})$ such that \bar{x} is the unique solution of the system

$$S(\bar{x}) \begin{cases} x(e) = 0 & \text{for all } e \in E_0(\bar{x}); \\ x(e) = 1 & \text{for all } e \in E_1(\bar{x}); \\ x(\delta(u)) = k & \text{for all } \delta(u) \in C_d(\bar{x}); \\ x(\delta(W)) = k & \text{for all } \delta(W) \in C_p^*(\bar{x}). \end{cases}$$

Suppose that \bar{x} is fractional. Let \bar{x}' be a solution obtained by replacing some (but at least one) fractional components of \bar{x} by 0 or 1 (and keeping all the other components of \bar{x} unchanged). If \bar{x}' is a point of $P(G, k)$, then it can be written as a convex combination of extreme points of $P(G, k)$. If \bar{y} is such an extreme point, then \bar{y} is said to be dominated by \bar{x} , and we write $\bar{x} \succ \bar{y}$. Note that if \bar{x} dominates \bar{y} , then $\{e \in E \mid 0 < \bar{y}(e) < 1\} \subset \{e \in E \mid 0 < \bar{x}(e) < 1\}$, $\{e \in E \mid \bar{x}(e) = 0\} \subseteq \{e \in E \mid \bar{y}(e) = 0\}$ and $\{e \in E \mid \bar{x}(e) = 1\} \subseteq \{e \in E \mid \bar{y}(e) = 1\}$. The relation \succ defines a partial ordering on the extreme points of $P(G, k)$. The minimal elements of this ordering (*i.e.* the extreme points x for which there is no extreme point y such that $x \succ y$) correspond to the integer extreme points of $P(G, k)$. The minimal extreme points of $P(G, k)$ are called extreme points of *rank 0*. An extreme point x is said to be of *rank p* , if x only dominates extreme points of rank $\leq p - 1$ and if it dominates at least one extreme point of rank $p - 1$. We notice that if \bar{x} is an extreme point of rank 1 and if we replace one fractional component of \bar{x} by 1, keeping unchanged the other integral components, we obtain a feasible solution \bar{x}' of $P(G, k)$ which can be written as a convex combination of integer extreme points of $P(G, k)$.

Didi Biha and Mahjoub [13] introduced the following reduction operations with respect to a solution \bar{x} of $P(G, k)$.

- θ_1 : delete an edge $e \in E$ such that $\bar{x}(e) = 0$;
- θ_2 : contract a node subset $W \subseteq V$ such that $G[W]$ is k -edge connected and $\bar{x}(e) = 1$ for all $e \in E(W)$;
- θ_3 : contract a node subset $W \subseteq V$ such that $|W| \geq 2$, $|\overline{W}| \geq 2$, $|\delta(W)| = k$ and $E(\overline{W})$ contains at least one edge with fractional value;
- θ_4 : contract a node subset $W \subseteq V$ such that $|W| \geq 2$, $|\overline{W}| \geq 2$, $G[W]$ is $\lceil \frac{k}{2} \rceil$ -edge connected, $|\delta(W)| = k + 1$ and $\bar{x}(e) = 1$ for all $e \in E(W)$.

Starting from a graph G and a solution $\bar{x} \in P(G, k)$ and applying $\theta_1, \theta_2, \theta_3, \theta_4$, we obtain a reduced graph G' and a solution $\bar{x}' \in P(G', k)$. It is not hard to see that \bar{x}' is an extreme point of $P(G', k)$ if and only if \bar{x} is an extreme point of $P(G, k)$. Didi Biha and Mahjoub [13] showed the following results.

Lemma 3.1 [13] *\bar{x}' is an extreme point of rank 1 of $P(G', k)$ if and only if \bar{x} is an extreme point of rank 1 of $P(G, k)$.*

Lemma 3.2 [13] *Let $G = (V, E)$ be a graph, $k \geq 2$ an integer and \bar{x} an extreme point of $P(G, k)$ of rank 1. Suppose that $C_p^*(\bar{x}) = \emptyset$. Then the graph induced by $E_f(\bar{x})$ is an odd cycle $C \subseteq E$ such that*

- 1) $\bar{x}(e) = \frac{1}{2}$ for all $e \in C$;
- 2) $\bar{x}(\delta(u)) = k$ for all $u \in V(C)$.

An extreme point \bar{x} of $P(G, k)$ will be said to be *critical* if \bar{x} is of rank 1 and if none of the operations $\theta_1, \theta_2, \theta_3, \theta_4$ can be applied to \bar{x} .

Operations $\theta_1, \theta_2, \theta_3, \theta_4$ can be used in a Branch-and-Cut algorithm for the k ECSP. As it will turn out, they can be used in a preprocessing phase for the separation and may be very effective in solving the problem.

4 A Branch-and-Cut algorithm for the k ECSP

In this section, we describe a Branch-and-Cut algorithm for the k ECSP. Our aim is to address the algorithmic applications of the theoretical results presented in the previous sections and describe some strategic choices made in order to solve that problem. So, let us assume that we are given a graph $G = (V, E)$ and a weight vector $w \in \mathbb{R}^E$ associated with the edges of G . Let $k \geq 3$ be the connectivity requirement for each node of V .

Grötschel et al. [24] introduced a class of valid inequalities for k ECSP(G) called partition inequalities that generalizes the cut inequalities. Let $\pi = (V_1, \dots, V_p)$, $p \geq 3$, be a partition of V . The *partition inequality* induced by π is given by

$$x(\delta(V_1, \dots, V_p)) \geq \left\lceil \frac{kp}{2} \right\rceil. \quad (26)$$

Clearly, if kp is even, then inequality (26) is redundant with respect to the cut inequalities. Grötschel et al. [24] gave sufficient conditions for the partition inequalities (26) to be facet defining.

Given a fractional solution \bar{x} of $P(G, k)$, we let $G' = (V', E')$ and \bar{x}' be obtained by repeated applications of operations $\theta_1, \theta_2, \theta_3, \theta_4$ with respect to \bar{x} . As pointed out above, \bar{x}' is an extreme point of $P(G', k)$ if and only if \bar{x} is an extreme point of $P(G, k)$. Moreover, we have the following lemmas which can be easily seen.

Lemma 4.1 *Let $a'x \geq \alpha$ be an F -partition inequality (resp. partition inequality) valid for k ECSP(G') induced by a partition $\pi' = (V'_0, V'_1, \dots, V'_p)$, $p \geq 2$, (resp. $\pi' = (V'_1, \dots, V'_p)$, $p \geq 3$) of V' . Let $\pi = (V_0, V_1, \dots, V_p)$, $p \geq 2$, (resp. $\pi = (V_1, \dots, V_p)$, $p \geq 3$) be the partition of V obtained by expanding the subsets V'_i of π' . Let $ax \geq \alpha$ be an inequality such that*

$$a(e) = \begin{cases} a'(e) & \text{for all } e \in E', \\ 1 & \text{for all } e \in (E \setminus E') \cap \delta_G(\pi), \\ 0 & \text{otherwise.} \end{cases}$$

Then $ax \geq \alpha$ is valid for k ECSP(G). Moreover, if $a'x \geq \alpha$ is violated by \bar{x}' , then $ax \geq \alpha$ is violated by \bar{x} .

Lemma 4.2 *Let $a'x \geq \alpha$ be an odd path inequality (resp. SP-partition inequality) valid for k ECSP(G') induced by a partition $\pi' = (W'_1, W'_2, V'_1, \dots, V'_{2p})$, $p \geq 2$ (resp. $\pi' = (V'_1, \dots, V'_p)$, $p \geq 3$). Let $\pi = (W_1, W_2, V_1, \dots, V_{2p})$, $p \geq 2$ (resp. $\pi = (V_1, \dots, V_p)$, $p \geq 3$), be the partition of V obtained by expanding the elements of π' . Let $ax \geq \alpha$ be the corresponding lifted odd path inequality (resp. lifted SP-partition inequality) obtained from $a'x \geq \alpha$ by application of the lifting procedure described in Section 2.2 (resp. Section 2.4) for the edges of $E \setminus E'$. Then $ax \geq \alpha$ is violated by \bar{x} if $a'x \geq \alpha$ is violated by \bar{x}' .*

Lemmas 4.1 and 4.2 show that looking for an odd path, F -partition, SP -partition or a partition inequality violated by \bar{x} reduces to looking for such inequality violated by \bar{x}' on G' . Note that this procedure can be applied for any solution of $P(G, k)$ and, in consequence, may permit to separate fractional solutions which are not necessarily extreme points of $P(G, k)$. In consequence, for more efficiency, our separation procedures will be performed on the reduced graph G' . The violated inequalities generated in G' with respect to \bar{x}' are lifted to violated inequalities in G with respect to \bar{x} using Lemmas 4.1 and 4.2.

We now describe the framework of our algorithm. To start the optimization we consider the following linear program given by the degree cuts associated with the vertices of the graph G together with the trivial inequalities, that is

$$\begin{aligned} \text{Min } & \sum_{e \in E} w(e)x(e) \\ x(\delta(u)) & \geq k && \text{for all } u \in V, \\ 0 \leq x(e) & \leq 1 && \text{for all } e \in E. \end{aligned}$$

The optimal solution $\bar{y} \in \mathbb{R}^E$ of this relaxation of the k ECSP is feasible for the problem if \bar{y} is an integer vector that satisfies all the cut inequalities. Usually, the solution \bar{y} is not feasible for the k ECSP, and thus, in each iteration of the Branch-and-Cut algorithm, it is necessary to generate further inequalities that are valid for the k ECSP but violated by the current solution \bar{y} . For this one has to solve the so-called *separation problem*. This consists, given a class of inequalities, in deciding whether the current solution \bar{y} satisfies all the inequalities of this class, and if not, in finding an inequality that is violated by \bar{y} . An algorithm solving this problem is called a *separation algorithm*. The Branch-and-Cut algorithm uses the inequalities previously described and their separations are performed in the following order

1. cut inequalities;
2. SP -partition inequalities;
3. odd path inequalities;
4. F -partition inequalities;
5. partition inequalities.

We remark that all inequalities are global (*i.e.* valid for all the Branch-and-Cut tree) and several inequalities may be added at each iteration. Moreover, we go to the next class of inequalities only if we haven't found any violated inequalities. Our strategy is to try to detect violated inequalities at each node of the Branch-and-Cut tree in order to obtain the best possible lower bound and thus limit the generated nodes. Generated inequalities are added by sets of 200 or less inequalities at a time.

Now we describe the separation procedures used in our Branch-and-Cut algorithm. All these procedures are applied on G' with weights $(\bar{y}'(e), e \in E')$ associated with its edges where \bar{y}' is the restriction on E' to the current LP-solution \bar{y} (G' and \bar{y}' are obtained by repeated applications of operations $\theta_1, \theta_2, \theta_3, \theta_4$).

The separation of the cut inequalities (3) can be performed by computing minimum cuts in G' . This can be done in polynomial time using Gomory-Hu algorithm [22]. This algorithm produces the so-called *Gomory-Hu tree* with the property that for all pairs of nodes $s, t \in V'$, the minimum (s, t) -cut in the tree is also a minimum (s, t) -cut in the graph G' . Actually, we use the algorithm developed by Gusfield [26] which requires $|V'| - 1$ maximum flow computations. The maximum flow computations are handled by the efficient Goldberg and Tarjan algorithm [21] that runs in $O(m'n' \log \frac{n'^2}{m'})$ time where m' and n' are the number of edges and nodes of G' , respectively. Thus, our separation algorithm for the cut inequalities is exact and runs in $O(m'n'^2 \log \frac{n'^2}{m'})$ time.

In what follows, we consider the separation procedure of the odd path inequalities (4). For this, we need the following lemma.

Lemma 4.3 *Let $x \in \mathbb{R}^E$ be a fractional solution of $P(G, k)$ and $\pi = (W_1, W_2, V_1, \dots, V_{2p})$, $p \geq 2$, a partition of V , which induces an odd path configuration. If each edge set $[V_i, V_{i+1}]$, $i = 1, \dots, 2p - 1$, contains an edge with fractional value and*

$$x([V_{i-1}, V_i]) + x([V_i, V_{i+1}]) \leq 1, \text{ for } i = 2, \dots, 2p - 1,$$

then the odd path inequality induced by π is violated by x .

Proof. As $x([V_{i-1}, V_i]) + x([V_i, V_{i+1}]) \leq 1$, $i = 2, \dots, 2p - 1$, we have that

$$x([V_{2s-1}, V_{2s}]) + x([V_{2s}, V_{2s+1}]) \leq 1, \quad s = 1, \dots, p - 1, \quad (27)$$

$$x([V_{2s}, V_{2s+1}]) + x([V_{2s+1}, V_{2s+2}]) \leq 1, \quad s = 1, \dots, p - 1. \quad (28)$$

By multiplying inequality (27) by $\frac{p-s}{p}$ and inequality (28) by $\frac{s}{p}$ and summing the resulting inequalities, we obtain

$$\sum_{i \in I} x([V_i, V_{i+1}]) + \sum_{i \in \bar{I}} \frac{p-1}{p} x([V_i, V_{i+1}]) \leq p - 1, \quad (29)$$

where $I = \{2, 4, 6, \dots, 2p - 2\}$ and $\bar{I} = \{1, 2, \dots, 2p - 1\} \setminus I$. Since each set $[V_i, V_{i+1}]$, $i = 1, \dots, 2p - 1$, contains an edge with fractional value, we have that $x([V_i, V_{i+1}]) < 1$, for $i \in \bar{I}$. Hence, it follows that

$$\sum_{i \in \bar{I}} x([V_i, V_{i+1}]) < p. \quad (30)$$

By multiplying inequality (30) by $\frac{1}{p}$ and summing the resulting inequality and inequality (29), we obtain

$$\sum_{i=1}^{2p-1} x([V_i, V_{i+1}]) < p,$$

and the result follows. □

Using Lemma 4.3 we can devise a separation procedure for inequalities (4). The idea is to find a partition $\pi = (W'_1, W'_2, V'_1, \dots, V'_{2p})$, $p \geq 2$, which induces an odd path configuration that satisfies the conditions of Lemma 4.3. The procedure works as follows. We first look, using a greedy method, for a path $\Gamma = \{e_1, \dots, e_{2p-1}\}$, $p \geq 2$, in G' such that the edges e_1, \dots, e_{2p-1} have fractional values and $\bar{y}'(e_{i-1}) + \bar{y}'(e_i) \leq 1$, for $i = 2, \dots, 2p-1$. If v'_1, \dots, v'_{2p} are the nodes of Γ taken in this order when going through Γ , we let $V'_i = \{v'_i\}$, $i = 1, \dots, 2p$, and $T_1 = (\bigcup_{i \in I_1} V'_i) \cup V'_1$ (resp. $T_1 = (\bigcup_{i \in I_1} V'_i) \cup V'_1 \cup V'_{2p}$) if p is odd (resp. even) and $T_2 = (\bigcup_{i \in I_2} V'_i) \cup V'_{2p}$ (resp. $T_2 = (\bigcup_{i \in I_2} V'_i)$) if p is odd (resp. even) where I_1 and I_2 are as defined in Section 2.1. In order to determine W'_1 and W'_2 , we compute a minimum cut separating T_1 and T_2 . If $\delta(W)$ is such a cut with $T_1 \subseteq W$, we let $W'_1 = W \setminus T_1$ and $W'_2 = V' \setminus (W \cup T_2)$. If the partition $\pi = (W'_1, W'_2, V'_1, \dots, V'_{2p})$ thus obtained induces an odd path configuration, then, by Lemma 4.3, the corresponding odd path inequality is violated by \bar{y}' . If not, we apply again that procedure by looking for an other path. In order to avoid the detection of the same path, we label the edges of the previous paths, so that they won't be considered in the search of a new path. This procedure continue until either a violated odd path inequality is found or all the edges having fractional values are labeled. The routine that permits to look for an odd path runs in $O(m'n')$ time. To compute the minimum cut separating T_1 and T_2 , we use Goldberg and Tarjan algorithm [21]. Since this algorithm runs in $O(m'n' \log \frac{n'^2}{m'})$ time our procedure is implemented to run in $O(m'^2 n' \log \frac{n'^2}{m'})$ time.

In the lifting procedure for inequalities (4) given in Section 2.2 we have to compute a coefficient λ for some edges $e \in E \setminus E'$. Since the computation of this coefficient is itself a hard problem, and $\lambda \leq 2$, we consider 2 as lifting coefficient rather than λ for those edges.

Now we discuss our separation procedure for the F -partition inequalities (18). These inequalities can be separated in polynomial time using the algorithm of Baiou et al. [2] when k is even and the edge set F is fixed. For the general case, we devised three heuristics to separate them.

Our first heuristic is based on Lemma 3.2. As pointed out in Lemma 3.2, for some extreme points of $P(G, k)$ of rank one, that is those extreme points $x \in P(G, k)$ such that $C_p^*(x) = \emptyset$, the edges with fractional values form an odd cycle C such that $x(e) = \frac{1}{2}$, for all $e \in C$ and $x(\delta(u)) = k$, for all $u \in V(C)$. The heuristic works as follows. It starts by determining a node set $\{v'_1, \dots, v'_p\}$, $p \geq 3$, that induces an odd cycle in G' . Then we let $V'_i = \{v'_i\}$, for $i = 1, \dots, p$, and $V'_0 = V' \setminus \{v'_1, \dots, v'_p\}$. We choose the edges of F among those of $\delta(V'_0)$ having values greater than $\frac{1}{2}$ and in such a way that $|F|$ and kp have different parities (if such an edge set F is empty then we look for an other partition). The node set $\{v'_1, \dots, v'_p\}$ is obtained by a simple labeling procedure. Hence, the heuristic runs in a linear time.

Before introducing our second heuristic, we first give the following lemma.

Lemma 4.4 *Let $x \in \mathbb{R}^E$ be a fractional solution of $P(G, k)$ and $\pi = (V_0, V_1, \dots, V_p)$, $p \geq 2$, a partition of V such that $x(\delta(V_i)) = k$, $i = 1, \dots, p$. Then an F -partition*

inequality, induced by π and an edge set $F \subseteq \delta(V_0)$ such that $|F|$ and kp have different parities is violated by x if the following inequality holds

$$|F| - x(F) + x(\delta(V_0) \setminus F) < 1. \quad (31)$$

Proof. As $x(\delta(V_i)) = k$, $i = 1, \dots, p$, we have that

$$\sum_{i=1}^p x(\delta(V_i)) = 2x(\delta(V_1, \dots, V_p)) + x(\delta(V_0)) = kp.$$

This together with (31) yield

$$-2x(F) + 2x(\delta(V_0)) + 2x(\delta(V_1, \dots, V_p)) < kp - |F| + 1,$$

and thus, the statement follows. \square

The heuristic is based on Lemma 4.4. It starts by determining all the nodes u of V' such that $\bar{y}'(\delta(u)) = k$ and $\delta(u)$ contains at least one edge with fractional value. In fact, it is this kind of nodes which may lead to violated F -partition inequalities. Let $\{v'_1, \dots, v'_p\}$ be the set of such nodes. We consider the partition $(V'_0, V'_1, \dots, V'_p)$ such that $V'_i = \{v'_i\}$, for $i = 1, \dots, p$, and $V'_0 = V' \setminus \{v'_1, \dots, v'_p\}$ and we choose the edges of F in a similar way as in the first heuristic. If inequality (31) holds with respect to F and V'_0 , then, by Lemma 4.4, the F -partition inequality corresponding to $(V'_0, V'_1, \dots, V'_p)$ and F is violated by \bar{y}' .

Before presenting our last heuristic, let us first remark that a partition $(V'_0, V'_1, \dots, V'_p)$ and an edge set $F \subseteq \delta(V'_0)$ may induces a violated F -partition inequality if $\bar{y}'(\delta(V'_0))$ is high and the edges of F are among those of $\delta(V'_0)$ with high values. Our heuristic tries to find such a partition. For this, we compute a Gomory-Hu tree in G' with the weights $(1 - \bar{y}'(e))$, $e \in E'$ associated with its edges. Then from each proper cut $\delta(W)$ with $V' \setminus W = \{v'_1, \dots, v'_p\}$, $p \geq 2$, obtained from the Gomory-Hu tree, we consider the partition $\pi = (V'_0, V'_1, \dots, V'_p)$ such that $V'_i = \{v'_i\}$, for $i = 1, \dots, p$, and $V'_0 = W$. The edge set F is chosen in a similar way as in the previous heuristics. Since the computation of the Gomory-Hu tree can be done in $O(m'n^2 \log \frac{n'^2}{m'})$ time, the heuristic runs in $O(m'n^2 \log \frac{n'^2}{m'})$.

These three heuristics are applied in the Branch-and-Cut algorithm in that order.

Now we turn our attention to the separation of the SP -partition inequalities (24). These inequalities can be separated in polynomial time using the algorithm of Baiou et al. [2] when G' is series-parallel. This algorithm is based on submodular functions. Recently, Didi Biha et al. [16] devised a pure combinatorial algorithm for the separation of SP -partition inequalities when the graph is series-parallel. For our purpose we devised a heuristic to separate inequalities (24) in the general case. This heuristic is based on Theorems 2.6 and 2.7. The main idea of the heuristic is to determine a partition $\pi = (V'_1, \dots, V'_p)$, $p \geq 3$, of V' which induces an outerplanar graph such that $[[V'_i, V'_{i+1}]] \geq \lceil \frac{k}{2} \rceil$, for $i = 1, \dots, p$ (the indices are modulo p) (see Figure 4), and for every consecutive sets V'_i and V'_j , the edge set $[V'_i, V'_j]$ contains at least one edge with

fractional value. To this end, we look in G' for a path $\Gamma = \{v'_1v'_2, v'_2v'_3, \dots, v'_{p-2}v'_{p-1}\}$, $p \geq 3$ such that $|\{v'_i, v'_{i+1}\}| \geq \lceil \frac{k}{2} \rceil$ and $[v'_i, v'_{i+1}]$ contains one or more edges with fractional value, for $i = 1, \dots, p-2$. We then let $V'_i = \{v'_i\}$, $i = 1, \dots, p-1$, and $V'_p = V' \setminus \{v'_1, \dots, v'_{p-1}\}$. Afterwards, we check by a simple heuristic if the graph G'_π is outerplanar. Finally, we check if the SP -partition inequality induced by π is violated by \bar{y}' or not. If the graph G'_π is not outerplanar or the SP -partition inequality induced by π is not violated by \bar{y}' , we apply again this procedure by looking for an other path. In order to avoid the detection of the same path, we label the nodes we met during the search of the previous paths, so that they won't be considered in the search of a new path. This process continue until either we find a violated SP -partition inequality or all the nodes of V' are labeled. The heuristic can be implemented to run in $O(m'n')$ time.

Now we discuss the separation of the partition inequalities (26). First observe that if $\pi = (V'_1, \dots, V'_p)$ is a partition of V' , with $p \geq 3$ and odd, such that $\bar{y}'(\delta(V'_i)) = k$, for $i = 1, \dots, p$, then the partition inequality induced by π is violated by \bar{y}' . Thus, one can devise a heuristic to separate inequalities (26) which consists in finding a partition $\pi = (V'_1, \dots, V'_p)$, with $p \geq 3$ and odd, such that $\bar{y}'(\delta(V'_i))$ is as small as possible, for $i = 1, \dots, p$. To do this, we compute a Gomory-Hu tree, say \mathcal{T} , in G' with the weights $(\bar{y}'(e), e \in E')$ associated with its edges. After that, we contract disjoint node subsets that induce proper tight cuts in \mathcal{T} . Let V'_1, \dots, V'_t be these sets and $\{v_{t+1}, \dots, v_p\} = V' \setminus (\bigcup_{i=1}^t V'_i)$. We then consider the partition $(V'_1, \dots, V'_t, \{v_{t+1}\}, \dots, \{v_p\})$ and check whether or not the corresponding partition inequality is violated by \bar{y}' . This algorithm leads to an $O(m'n'^2 \log \frac{n'^2}{m'})$ time complexity.

To store the generated inequalities, we create a pool whose size increases dynamically. All the generated inequalities are put in the pool and are dynamic, *i.e.* they are removed from the current LP when they are not active. We first separate inequalities from the pool. If all the inequalities in the pool are satisfied by the current LP-solution, we separate the classes of inequalities in the order given above.

Another important issue in the effectiveness of the Branch-and-Cut algorithm is the computation of a good upper bound at each node of the Branch-and-Cut tree. To do this, if the separation procedures do not generate any violated inequality and the current solution \bar{y} is still fractional, then we transform \bar{y} into a feasible solution of the k ECSP, say \hat{y} , by rounding up to 1 all the fractional components of \bar{y} . We then try to reduce the weight of the solution thus obtained by removing from the subgraph $H = (V, \hat{E})$ induced by \hat{y} some unnecessary edges that is, edges which do not affect the k -edge connectedness of H . To this end, we remove from \hat{E} each edge $e = uv$ such that $|\delta(u) \cap \hat{E}| \geq k+1$ and $|\delta(v) \cap \hat{E}| \geq k+1$. We then check if the resulting edge set, say \hat{E}' , induces a k -edge connected subgraph of G by computing a Gomory-Hu tree. If there exists in \hat{E}' a cut $\delta(W)$, $W \subseteq V$, containing less than k edges, then we add in \hat{E}' edges of $[W, V \setminus W] \setminus \delta(W)$ that have been previously

removed from \hat{E} as many as necessary in order to satisfy the cut $\delta(W)$. We do this until the graph (V, \hat{E}') induces a k -edge connected subgraph of G . Note that we add to each violated cut the edges having the smallest weights.

5 Computational results

The Branch-and-Cut algorithm described in the previous section has been implemented in C++, using ABACUS 2.4 alpha [1, 35] to manage the Branch-and-Cut tree, and CPLEX 9.0 [11] as LP-solver. It was tested on a Pentium IV 3.4 Ghz with 1 Gb of RAM, running under Linux. We fixed the maximum CPU time to 5 hours. The test problems were obtained by taking TSP test problems from the TSPLIB library [36]. The test set consists in complete graphs whose edge weights are the rounded euclidian distance between the edge's vertices. The tests were performed for $k = 3, 4, 5$. In all our experiments, we have used the reduction operations described in the previous sections, unless otherwise specified. Each instance is given by its name followed by an extension representing the number of nodes of the graph. The other entries of the various tables are :

NCut	:	number of generated cut inequalities ;
NSP	:	number of generated SP -partition inequalities ;
NOP	:	number of generated odd path inequalities ;
NFP	:	number of generated F -partition inequalities ;
NP	:	number of generated partition inequalities ;
COpt	:	weight of the optimal solution obtained ;
Gap	:	the relative error between the best upper bound (the optimal solution if the problem has been solved to optimality) and the lower bound obtained at the root node of the Branch-and-Cut tree ;
NSub	:	number of subproblems in the Branch-and-Cut tree ;
TT	:	total CPU time in hours :min :sec.

The instances indicated with "*" are those whose CPU time exceeded 5 hours. For these instances, the gap is indicated in italic.

Our first series of experiments concerns the k ECSP for $k = 3$. The instances we have considered have graphs with 14 to 318 nodes. The results are summarized in Table 1. It appears from Table 1 that all the instances have been solved to the optimality within the time limit except the last five instances. We have that four instances (burma14, gr21, fri26, brazil58) have been solved in the cutting plane phase (*i.e.* no branching is needed). For most of the other instances, the gap is less than 1%. We also observe that our separation procedures detect a large enough number of SP -partition and F -partition inequalities and seem to be quite efficient.

Our next series of experiments concerns the k ECSP with $k = 4, 5$. The results are given in Table 2 for $k = 4$ and Table 3 for $k = 5$. The instances considered have graphs with 52 to 561 nodes. Note that for $k = 4$, the SP -partition and partition

Instance	NCut	NSP	NOP	NFP	NP	COpt	Gap	NSub	TT
burma14	4	3	0	0	4	5530	0.00	1	0 :00 :01
ulysses16	5	7	1	15	7	11412	0.39	3	0 :00 :11
gr21	5	6	1	0	2	4740	0.00	1	0 :00 :01
fri26	9	5	0	0	0	1543	0.00	1	0 :00 :01
bayg29	14	16	2	33	2	2639	0.19	7	0 :00 :01
dantzig42	41	31	6	90	18	1210	0.68	71	0 :00 :07
att48	34	34	5	60	9	17499	0.56	61	0 :00 :06
berlin52	36	31	12	97	6	12601	0.45	33	0 :00 :03
brazil58	46	42	2	36	29	42527	0.00	1	0 :00 :05
eil76	9	12	3	298	2	876	0.06	7	0 :00 :03
pr76	130	207	72	2231	54	187283	1.50	6767	0 :35 :32
rat99	41	26	13	341	23	2029	0.38	41	0 :00 :47
kroA100	170	197	31	1207	57	36337	0.97	4201	0 :54 :06
kroB100	130	114	37	830	47	37179	0.73	723	0 :08 :00
rd100	101	74	11	418	18	13284	0.43	171	0 :03 :37
eil101	86	72	21	3604	15	1016	0.55	1109	0 :17 :41
lin105	179	198	47	829	68	25530	0.69	1031	0 :22 :39
pr107	201	190	34	674	114	70852	0.84	2071	1 :26 :49
gr120	50	45	6	588	17	11442	0.19	99	0 :11 :15
bier127	46	59	4	276	13	198184	0.15	11	0 :01 :55
ch130	121	132	30	1355	40	10400	0.55	1693	1 :05 :05
ch150	92	93	19	588	22	11027	0.41	193	0 :20 :31
kroA150	155	143	41	845	47	44718	0.53	1205	1 :16 :35
kroB150	130	110	16	952	48	43980	0.31	437	0 :38 :43
rat195	24	19	3	514	1	3934	0.06	7	0 :08 :21
d198	171	105	23	617	59	25624	0.21	159	1 :04 :19
gr202	77	69	14	558	22	65729	0.11	69	0 :13 :16
*pr226	364	248	35	162	41	-	<i>9.02</i>	261	5 :00 :00
*gr229	179	245	23	1568	94	-	<i>1.00</i>	1219	5 :00 :00
*pr264	275	181	145	668	62	-	<i>12.29</i>	69	5 :00 :00
*a280	142	84	56	2539	59	-	<i>2.69</i>	459	5 :00 :00
*lin318	189	147	15	610	58	-	<i>4.94</i>	25	5 :00 :00

TAB. 1 – Results for $k = 3$ with reduction operations.

Instance	NCut	NOP	NFP	COpt	Gap	NSub	TT
berlin52	5	0	2	18295	0.00	1	0 :00 :01
pr76	3	0	4	266395	0.00	1	0 :00 :01
kroA100	10	0	11	51221	0.00	1	0 :00 :47
kroB100	9	5	123	53597	0.08	21	0 :00 :09
rd100	10	1	91	19130	0.00	1	0 :00 :05
eil101	0	0	60	1453	0.00	1	0 :00 :02
lin105	20	1	5	36353	0.00	1	0 :00 :01
pr107	29	0	0	98381	0.00	1	0 :00 :01
gr120	6	0	36	16400	0.00	1	0 :00 :02
bier127	16	0	0	282207	0.00	1	0 :00 :01
ch130	12	0	132	14854	0.00	1	0 :00 :05
ch150	12	2	70	15854	0.00	1	0 :00 :02
kroA150	13	0	27	64249	0.00	1	0 :00 :02
kroB150	20	0	4	62710	0.00	1	0 :00 :01
rat195	0	0	37	5750	0.00	1	0 :00 :13
d198	43	0	71	35404	0.01	3	0 :00 :16
gr202	13	3	220	94841	0.02	3	0 :01 :28
pr226	91	0	6	183537	0.00	1	0 :00 :04
gr229	24	2	15	318565	0.00	1	0 :00 :03
pr264	59	1	7	122941	0.00	1	0 :00 :06
a280	3	0	180	6317	0.00	1	0 :01 :00
pr299	30	0	427	117559	0.00	1	0 :00 :20
lin318	28	0	2	105000	0.00	1	0 :00 :06
rd400	21	2	232	36676	0.00	1	0 :07 :39
pr439	78	3	61	264975	0.02	19	0 :02 :52
si535	0	0	4	53604	0.00	1	0 :00 :39
pa561	10	1	306	6724	0.00	1	0 :08 :37

TAB. 2 – Results for $k = 4$.

inequalities are redundant with respect to the cut inequalities (3). Thus, they are not included in the resolution process for $k = 4$ and therefore do not appear in Table 2.

First observe that for $k = 4$, the CPU time for all the instances is relatively small and most of the instances have been solved in less than 1 minute. We can also observe that 23 instances over 27 are solved in the cutting plane phase. Moreover, a few number of odd path inequalities are generated. However a large enough number of F -partition inequalities is detected. Thus, these later inequalities seem to be very effective for solving the k ECSP when k is even. This also shows that the k ECSP is easier to solve when k is even, what is also confirmed by the results of Table 3 for $k = 5$. In fact, the instance pr264 has been solved for $k = 4$ in 1 second, whereas it could not be solved to optimality for $k = 5$ after 5 hours. The same observation can be done for pr439. Also, we can remark that the CPU time for all the instances when $k = 5$ is higher than that when $k = 4$. For instance, the test problem d198 has

Instance	NCut	NSP	NOP	NFP	NP	COpt	Gap	NSub	TT
berlin52	5	2	2	26	2	24845	0.00	1	0 :00 :01
pr76	2	0	0	52	1	372392	0.00	1	0 :00 :01
kroA100	5	1	5	76	6	71422	0.04	11	0 :00 :06
kroB100	6	1	2	83	5	74241	0.01	3	0 :00 :06
rd100	6	2	6	193	5	26168	0.01	5	0 :00 :24
eil101	1	0	0	309	0	1938	0.00	1	0 :01 :10
lin105	9	1	3	119	3	50711	0.00	1	0 :00 :26
pr107	92	40	57	680	33	132870	0.41	381	0 :14 :45
gr120	2	0	3	93	3	22024	0.11	27	0 :00 :17
bier127	22	2	12	450	8	383165	0.09	25	0 :04 :25
ch130	1	0	0	45	0	20508	0.01	3	0 :00 :05
ch150	5	0	7	58	1	21791	0.01	37	0 :00 :50
kroA150	9	0	5	141	3	87950	0.07	11	0 :00 :19
kroB150	14	1	7	462	6	85583	0.02	11	0 :15 :39
rat195	1	0	0	508	0	7773	0.00	1	0 :20 :54
d198	56	9	6	1093	32	47614	0.15	337	1 :50 :40
gr202	0	0	0	64	0	128990	0.00	1	0 :00 :31
pr226	142	34	20	661	50	260878	0.58	103	2 :38 :50
gr229	18	1	11	679	9	434422	0.06	43	0 :31 :58
*pr264	105	12	38	1327	28	-	<i>1.78</i>	43	5 :00 :00
a280	2	0	2	302	0	8643	0.02	7	0 :05 :05
pr299	11	3	2	637	1	161576	0.00	1	0 :05 :12
lin318	24	3	11	1548	11	144341	0.02	7	4 :34 :39
rd400	11	1	15	691	6	49893	0.01	17	1 :29 :09
*pr439	46	2	8	746	0	-	<i>3.46</i>	1	5 :00 :00
si535	0	0	0	0	0	79115	0.00	1	0 :00 :19
pa561	1	0	2	286	1	9161	0.00	1	3 :26 :58

TAB. 3 – Results for $k = 5$.

been solved in 1h 50mn when $k = 5$, whereas only 16 seconds were needed to solve it when $k = 4$.

Compared to Table 1, Tables 2 and 3 also show that, for the same parity of k , the k ECSP becomes easier to solve when k increases. In fact, with $k = 3$, we could not solve to optimality instances with more than 202 nodes, whereas for $k = 5$, we could solve larger instances.

The results for $k = 3, 4, 5$ can also be compared to those obtained by Kerivin et al. [29] for the 2ECSP. It turns out that for the same instances, the problem has been easier to solve for $k = 2$ than for $k = 3$. However, for $k = 4$ the problem appeared to be easier to solve than for $k = 2$. This shows again that the case when k is odd is harder to solve than that when k is even and that the problem becomes easier when k increases with the same parity.

In order to evaluate the impact of the reduction operations $\theta_1, \theta_2, \theta_3, \theta_4$ on the separation procedures, we tried to solve the k ECSP, for $k = 3$, without using them. The results are given in Table 4.

Instance	NCut	NSP	NOP	NFP	NP	COpt	Gap	NSub	TT
berlin52	31	28	19	44	4	12601	0.44	15	0 :00 :04
brazil58	50	27	1	28	31	42527	0.22	3	0 :00 :07
eil76	9	6	3	102	2	876	0.00	1	0 :00 :01
pr76	103	168	65	1378	37	187283	1.60	3483	0 :38 :46
rat99	41	19	10	223	17	2029	0.32	61	0 :01 :29
kroA100	193	234	47	1765	70	36337	1.42	7575	4 :13 :38
kroB100	141	142	36	899	38	37179	0.98	1337	0 :45 :34
rd100	103	84	15	445	21	13284	0.40	233	0 :11 :40
eil101	77	58	26	2527	12	1016	0.38	801	0 :18 :50
lin105	161	158	50	569	53	25530	0.61	547	0 :34 :25
*pr107	218	221	136	1101	104	-	0.81	4447	5 :00 :00
gr120	42	38	6	252	15	11442	0.18	93	0 :05 :38
bier127	58	56	9	240	12	198184	0.16	17	0 :04 :43
ch130	141	147	38	1590	45	10400	0.52	2459	4 :10 :31
ch150	90	76	15	391	23	11027	0.39	107	0 :21 :07
kroA150	155	135	23	705	56	44718	0.55	1107	3 :08 :37
kroB150	150	141	22	1006	43	43980	0.31	535	1 :55 :20
rat195	23	18	7	898	1	3934	0.01	19	0 :19 :23
d198	192	118	25	720	50	25624	0.27	585	5 :03 :16
gr202	73	62	13	278	23	65729	0.05	37	0 :37 :31

TAB. 4 – Results for $k = 3$ without reduction operations.

As it appears from Tables 1 and 4, the CPU time increased for the majority of the instances when the reduction operations are not used. In particular, for the instance pr107 we did not reach the optimal solution after 5 hours, whereas it has been solved to optimality after 1h 26mn. Almost, the CPU time for the instances ch130 and d198, for example, increased from 1 hour to more than 4 hours. Moreover, we remark that when using the reduction operations, we generate more SP -partition, F -partition and partition inequalities and fewer nodes in the Branch-and-Cut tree. This implies that our separation heuristics are less efficient without the reduction operations. Thus, it seems that the reduction operations play an important role in the resolution of the problem. They permit to strengthen much more the linear relaxation of the problem and accelerate its resolution.

We also tried to measure the effect of the different non-basic classes of inequalities (*i.e.* inequalities other than cut and trivial inequalities). For this, we have first considered a Branch-and-Cut algorithm for the k ECSP with $k = 3$ using only the cut constraints. In this case, we could not solve any of the instances having more

than 52 nodes. Even more, after less than 10 minutes of CPU time, the Branch-and-Cut tree gets a very big size and the resolution process stops. To illustrate this, we take for example the instance `brazil58`. For this instance, the Branch-and-Cut tree contained 11769 nodes after 10 minutes when the Branch-and-Cut algorithm used only the cut inequalities, whereas it has been solved without branching when using the other classes of inequalities.

Finally, we tried to evaluate separately the efficiency of each class of the non-basic inequalities. For this, we also considered the case when $k = 3$. We have seen that all the classes of inequalities have a big effect on the resolution of the problem. In particular, the *SP*-partition inequalities seem to play a central role. This can be seen by considering the instance `d198`. This instance has been solved in 1h 04mn using all the constraints. However, without the *SP*-partition inequalities, we could not reach the optimal solution after 5 hours. We also remarked that the gap increased when one of these classes of inequalities is not used in the Branch-and-Cut algorithm.

6 Concluding remarks

In this paper, we have studied the k -edge connected subgraph problem with high connectivity requirement that is when $k \geq 3$. We have presented some classes of valid inequalities and have described some conditions for these inequalities to be facet defining for the associated polytope. We also discussed separation heuristics for these inequalities. Using these results, we have devised a Branch-and-Cut algorithm for the problem. This algorithm uses some reduction operations.

Our computational results have shown that the odd path, the *F*-partition, the *SP*-partition and the partition inequalities are very effective for the problem when k is odd. They have also shown the importance of the *F*-partition inequalities for the even case. We could also measure the importance of our separation heuristics. In particular, our heuristics to separate the *SP*-partition and *F*-partition inequalities have appeared to be very efficient. In addition, the reduction operations have been essential for having a good performance of the Branch-and-Cut algorithm. In fact, they permitted to considerably reduce the size of the graph supporting a fractional solution and to accelerate the separation process.

These experiments also showed that the k ECSP is easier to solve when k is even and that, for the same parity of k , the problem becomes easier to solve when k increases.

One of the separation heuristic devised for the *F*-partition inequalities is based on a partial characterization of the critical extreme points of the linear relaxation of the k -edge connected subgraph polytope. It would be very interesting to have a complete characterization of these points. This may yield the identification of new facet defining inequalities for the problem. It may also permit to devise more appropriate separation heuristics for the inequalities given in this paper.

In many real instances, we may consider node-connectivity instead of edge-connectivity. The study presented in this paper may be very useful for the k -node connected subgraph problem for which we require k node-disjoint paths between every pair of nodes.

In addition to the survivability aspect, one can consider the capacity dimensioning of the network. These issues have been mostly treated separately in the literature. It would be interesting to extend the study developed in this paper to the more general capacitated survivable network design model.

Références

- [1] ABACUS - A Branch-And CUt System, "http ://www.informatik.uni-koeln.de/abacus".
- [2] M. Baïou, F. Barahona and A. R. Mahjoub, "Separation of partition inequalities", *Mathematics of Operational Research* 25, 2000, pp. 243-254.
- [3] M. Baïou and A. R. Mahjoub, "Steiner 2-edge connected subgraph polytopes on series-parallel graphs", *SIAM Journal on Discrete Mathematics* 10, 1997, pp. 505-514.
- [4] F. Barahona and A. R. Mahjoub, "On two-connected subgraph polytopes", *Discrete Mathematics* 147, 1995, pp. 19-34.
- [5] D. Bienstock, E. F. Brickel and C. L. Monma, "On the structure of minimum weight k -connected networks", *SIAM Journal on Discrete Mathematics* 3, 1990, pp. 320-329.
- [6] S. C. Boyd and T. Hao, "An integer polytope related to the design of survivable communication network", *SIAM Journal on Discrete Mathematics* 6, 1993, pp. 612-630.
- [7] S. Chopra, "The k -edge connected spanning subgraph polyhedron", *SIAM Journal on Discrete Mathematics* 7, 1994, pp. 245-259.
- [8] G. Cornuéjols, J. Fonlupt and D. Nadeff, "The traveling salesman problem on a graph and some related polyhedra", *Mathematical Programming* 33, 1985, pp. 1-27.
- [9] R. Coullard, A. Rais, R. L. Radin, D. K. Wagner, "Linear time algorithm for the 2-connected Steiner subgraph problem on special classes of graphs", *Networks* 23, 1993, pp. 195-206.
- [10] R. Coullard, A. Rais, R. L. Radin, D. K. Wagner, "The Dominant of the 2-connected Steiner subgraph polytope for W_4 -free graphs", *Discrete Applied Mathematics* 66, 1996, pp. 33-43.
- [11] Cplex, "http ://www.ilog.com".
- [12] I. Diarrassouba and L. Slama, "Etude polyédrale du problème du sous-graphe k -arête connexe dans les graphes outerplanaires et série-parallèles", *Preprint*, 2005.

- [13] M. Didi Biha and A. R. Mahjoub, "The k -edge connected subgraph problem I : Polytopes and critical extreme points", *Linear Algebra and Its applications* 381, 2004, pp. 117-139.
- [14] M. Didi Biha and A. R. Mahjoub, " k -edge connected polyhedra on series-parallel graphs", *Operations Research Letters* 19, 1996, pp. 71-78.
- [15] M. Didi Biha and A. R. Mahjoub, "Steiner k -edge connected subgraph polyhedra", *Journal of Combinatorial Optimization* 4, 2000, pp. 131-134.
- [16] M. Didi Biha, A. R. Mahjoub and L. Slama, "On the separation of partition inequalities", *Proceedings INOC 2005*, pp. B2.500-B2.505.
- [17] J. Fonlupt and A. R. Mahjoub, "Critical extreme points of the 2-edge connected spanning subgraph polytope", *Mathematical Programming* 105, 2006, pp. 289-310.
- [18] J. Fonlupt and D. Naddef, "The traveling salesman problem in graphs with some excluded minors", *Mathematical Programming* 53, 1992, pp. 147-172.
- [19] G. N. Frederickson and J. Jàjà, "On the relationship between the biconnectivity augmentations and traveling salesman problem", *Theoretical Computer Science* 13, 1982, pp. 189-201.
- [20] M. R. Garey and D. J. Johnson, "Computer and Intractability : A Guide to the Theory of NP -completeness", *Freeman, San Francisco*, 1979.
- [21] A. Goldberg and R. E. Tarjan, "A New Approach to the Maximum Flow Problem", *Journal of the Association for Computing Machinery* 35, 1988, pp. 921-940.
- [22] R. E. Gomory and T. C. Hu, "Multi-Terminal Network Flows", *Journal of the Society for Industrial and Applied Mathematics* 9, 1961, pp. 551-570.
- [23] M. Grötschel and C. L. Monma, "Integer polyhedra arising from certain network design problems with connectivity constraints", *SIAM Journal on Discrete Mathematics* 3, 1990, pp. 502-523.
- [24] M. Grötschel, C. L. Monma and M. Stoer, "Polyhedral approaches to network survivability", In F. Roberts, F. Hwang and C. L. Monma, eds, *Reliability of computer and Communication networks, Vol 5, Series Discrete Mathematics and Computer Science, AMS/ACM*, 1991, pp. 121-141.
- [25] M. Grötschel, C. L. Monma and M. Stoer, "Polyhedral and computational investigations arising for designing communication networks with high survivability requirements", *Operation Research* 43, 1995, pp. 1012-1024.
- [26] D. Gusfield, "Very simple method for all pairs network flow analysis", *Society of Industrial and Applied Mathematics* 009, 1990, pp. 143-155.
- [27] J. Hao and J. B. Orlin, "A faster algorithm for finding the minimum cut in a directed graph", *Journal of Algorithms*, 1992, pp. 424-446.
- [28] H. Kerivin and A. R. Mahjoub, "Design of Survivable Networks : A Survey", *Networks* 46, 2005, pp. 1-21.
- [29] H. Kerivin, A. R. Mahjoub and C. Nocq, "(1,2)-survivable networks : Facets and Branch-and-Cut", *The Sharpest Cut, MPS-SIAM Series in Optimization*, M. Grötschel (Editor), 2004, pp. 121-152.

- [30] C.-W. Ko and C. L. Monma, "Heuristics for designing highly survivable communication networks", *Technical Report, Bellcore, Morristown, New Jersey*, 1989.
- [31] A. R. Mahjoub, "Two-edge Connected spanning subgraphs and polyhedra", *Mathematical Programming* 64, 1994, pp. 199-208.
- [32] A. R. Mahjoub, "On perfectly Two-edge connected subgraphs and polyhedra", *Discrete Mathematics* 170, 1997, pp. 153-172.
- [33] C. L. Monma, B. S. Munson and W. R. Pulleyblank, "Minimum weight two-connected spanning networks", *Operations Research* 37, 1989, pp. 153-171.
- [34] G. L. Nemhauser and L. A. Wolsey, "Integer and Combinatorial Optimization", Wiley Editions, 1988, pp. 259-295.
- [35] S. Thienel, "ABACUS - A Branch-And-CUt System", PhD thesis, Universität zu Köln, 1995.
- [36] TSPLIB, "<http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>".