

Un algorithme de coupe et branchement pour le problème du sous-graphe k -arête connexe

F. Bendali¹, I. Diarrassouba¹, M. Didi Biha², A. Ridha Mahjoub¹ et J. Mailfert¹

¹ Université Blaise Pascal - Clermont Ferrand II, Complexe scientifique des Cèzeaux, 63177 Aubière cedex

(bendali,diarrassouba,mailfert)@isima.fr, ridha.mahjoub@math.univ-bpclermont.fr

² Laboratoire d'Analyse non linéaire et Géométrie, Université d'Avignon, 339 chemin des Meinajaries, 84911 Avignon, France

mohamed.didi-biha@univ-avignon.fr

Résumé Un graphe $G = (V, E)$ est dit k -arête connexe si entre chaque paire de sommets, il existe au moins k chaînes arête-disjointes. Si G est muni d'une fonction coût sur les arêtes, le problème du sous-graphe k -arête connexe consiste à trouver un sous-graphe k -arête connexe de G de coût minimum, couvrant les sommets de G . Dans ce papier, nous considérons ce problème d'un point de vue polyédral. Nous introduisons une nouvelle famille d'inégalités, appelées inégalités de chaîne impaire, valides pour le polytope des solutions, et nous décrivons des conditions suffisantes pour que ces inégalités définissent des facettes. Nous discutons également de procédures de séparation pour ces inégalités ainsi que pour d'autres familles d'inégalités valides. Nous étudions aussi des opérations de réduction de graphes. En utilisant ces résultats, nous développons un algorithme de coupe et branchement pour le problème et présentons certains résultats expérimentaux.

Mots clés : Sous-graphe k -arête connexe, polytope, facette, separation, coupe et branchement, réduction de graphes.

1 Introduction

Avec l'introduction de la technologie optique dans les réseaux de télécommunications, ce domaine a connu un développement considérable. En effet, cette technologie permet d'avoir une grande capacité de transmission. Cependant, un problème qui se pose toujours est celui de la défaillance d'un ou plusieurs liens du réseau, ce qui peut conduire à un arrêt total des communications. Pour éviter ce problème, il faut mettre en place des structures de réseaux suffisamment fiables dans le sens où si une ou plusieurs liaisons sont détruites, le réseau continue à fonctionner. Ces conditions de fiabilité sont généralement exprimées en terme de connectivité. Entre chaque paire de nœuds du réseau, il doit exister un nombre minimum de chaînes arête-disjointes pour assurer le reroutage en cas de panne. Ce nombre de chaînes disjointes est généralement considéré uniforme c'est-à-dire le même pour toutes les paires de sommets. La topologie des réseaux dite 2-arête connexe est souvent utilisée. Ici, entre chaque paire de sommets, il doit exister au moins 2 chaînes arête-disjointes, de telle manière que si une chaîne tombe en panne, le trafic peut être rerouté sur la deuxième chaîne. Cette topologie s'est avérée performante pour plusieurs types de réseaux de télécommunications. Cependant, pour certains réseaux, on exige un degré de fiabilité plus élevé. Dans ce papier, nous considérons ce problème avec forte fiabilité.

Un graphe $G = (V, E)$ est dit k -arête connexe, $k \geq 1$, si entre chaque paire de sommets il existe au moins k chaînes arête-disjointes. Etant donné des coûts sur les arêtes de G , le problème du sous-graphe k -arête connexe (k ECSP) consiste à déterminer un sous-graphe k -arête connexe de G de coût minimum qui couvre tous les sommets de G .

Le problème k ECSP est NP -difficile si $k \geq 2$ [12]. Si $k = 1$, k ECSP n'est rien d'autre que le problème de l'arbre couvrant de poids minimum qui peut être résolu en temps polynomial. k ECSP a été largement étudié quand $k = 2$ [2,10,11,16,17,18,19,20,21]. Il a reçu, par contre, peu d'attention quand $k \geq 3$. Le problème semble plus difficile dans ce cas, en particulier lorsque k est impair. Dans ce papier, nous considérons ce problème d'un point de vue polyédral. Nous introduisons une nouvelle famille d'inégalités, appelées inégalités de chaîne impaire, valides pour le polytope des solutions, et nous décrivons des conditions suffisantes

pour que ces inégalités définissent des facettes. Nous discutons également de procédures de séparation pour ces inégalités ainsi que pour d'autres familles d'inégalités valides. Nous étudions aussi des opérations de réduction de graphes. En utilisant ces résultats, nous développons un algorithme de coupe et branchement pour le problème et présentons certains résultats expérimentaux.

Si $W \subset V$, l'ensemble des arêtes qui ont une extrémité dans W et l'autre dans $\overline{W} = V \setminus W$ est appelé *coupe* et est noté $\delta(W)$. Si $W = \{u\}$, pour un sommet u de V , on écrit $\delta(u)$ pour $\delta(\{u\})$. Etant donné un point $x \in \mathbb{R}^E$ et un sous-ensemble d'arêtes $F \subseteq E$, on définit

$$x(F) = \sum_{e \in F} x(e).$$

Si $(c(e), e \in E)$ est un vecteur coût associé aux arêtes de G , k ECSP est équivalent au programme linéaire en nombre entier suivant :

$$\begin{aligned} \text{Min } & \sum_{e \in E} c(e)x(e) \\ x(\delta(W)) & \geq k \quad \forall W \subset V, W \neq V, W \neq \emptyset, & (1) \\ 0 \leq x(e) & \leq 1 \quad \forall e \in E, & (2) \\ x(e) \in \{0, 1\} & \quad \forall e \in E. & (3) \end{aligned}$$

Les inégalités (1) sont appelées *inégalités de coupe* et les inégalités (2) sont appelées *inégalités triviales*.

L'enveloppe convexe des solutions de k ECSP est appelée *polytope des sous-graphes k -arête connexes* et sera notée k ECSP(G). On désigne par $P(G, k)$ le polytope décrit par les inégalités (1) et les inégalités (2).

Le polytope k ECSP(G) a été étudié en particulier dans le cas où $k = 2$. Mahjoub [19] caractérise le polytope 2 ECSP(G) dans la classe des graphes série-parallèles. Il introduit également une famille d'inégalités valides pour 2 ECSP(G). Barahona et Mahjoub [2] donnent une caractérisation complète de 2 ECSP(G) dans les graphes de Halin (un graphe de Halin est constitué d'un cycle et d'un arbre sans sommet de degré 2 dont les sommets pendants sont les sommets du cycle). Fonlupt et Mahjoub [10] étudient le polytope $P(G, 2)$ et introduisent le concept de point extrême critique. Kerivin et al.[17] proposent un algorithme de coupe et branchement pour 2 ECSP.

Grötschel et al. considèrent dans [15] un modèle plus général de fiabilité de réseaux. Ils décrivent des contraintes valides pour le polytope associé et développent un algorithme de coupe et branchement pour résoudre le problème.

Dans [4], Chopra considère k ECSP lorsque chaque arête du graphe peut être utilisée plus d'une fois. Il caractérise en particulier le polytope lorsque le graphe est outerplanaire (un graphe est dit outerplanaire lorsqu'il est planaire et ses sommets peuvent être disposés sur un cycle). Didi Biha et Mahjoub étendent dans [9] les résultats de Chopra aux graphes série-parallèles (un graphe est dit série-parallèle lorsqu'il n'est pas contractible à K_4). Ils montrent également que, dans cette classe de graphes, k ECSP(G) = $P(G, k)$ lorsque k est pair. Dans [8], Didi Biha et Mahjoub étudient le polytope $P(G, k)$ pour $k \geq 2$. Ils donnent des propriétés structurales des points extrêmes de ce polytope et décrivent des opérations de réduction de graphes.

Dans [3], Bienstock et al. établissent des propriétés structurales des solutions optimales de k ECSP lorsque la fonction coût satisfait les inégalités triangulaires (c'est-à-dire $c(e_1) \leq c(e_2) + c(e_3)$, pour tout triplet d'arêtes (e_1, e_2, e_3) formant un triangle dans le graphe). Leurs résultats généralisent ceux obtenus par Frederickson et Jàjà [11] et Monma et al. [21] pour $k = 2$. Dans [18], Ko et Monma proposent des heuristiques pour k ECSP.

Le papier est organisé comme suit. Dans la section 2, nous introduisons la famille des inégalités de chaîne impaire et nous discutons de propriétés de facettes. Dans la section 3, nous présentons des opérations de réduction de graphes et nous étudions leurs conséquences pour la résolution du problème. Dans la section 4, nous présentons notre algorithme de coupe et branchement pour le problème k ECSP et nous discutons dans la section 5 des résultats expérimentaux. Le reste de cette section est consacré à quelques notations qui seront utilisées tout au long du papier.

Soit $G = (V, E)$ un graphe. Si $W \subset V$, on note par $E(W)$ l'ensemble des arêtes de E dont les deux extrémités sont dans W et par $G[W] = (W, E(W))$ le sous-graphe de G induit par l'ensemble W . Si $\pi = (V_1, \dots, V_p)$, $p \geq 2$, est une partition de V , on note par $E_\pi = \delta(V_1, \dots, V_p)$ l'ensemble des arêtes de G dont les deux extrémités appartiennent à deux ensembles distincts de la partition. On note par $G_\pi = (V_\pi, E_\pi)$ le graphe obtenu à partir de G en contractant les V_i , $i = 1, \dots, p$. Si $U, W \subseteq V$ sont deux sous-ensembles disjoints de sommets, on note par $[U, W]$ l'ensemble des arêtes ayant une extrémité dans U et l'autre dans W .

2 Les Inégalités de chaîne impaire

Dans cette section nous présentons une nouvelle famille d'inégalités valides pour $kECSP(G)$. Nous donnons des conditions suffisantes pour que ces inégalités définissent des facettes de $kECSP(G)$ et nous discutons du problème de séparation associé.

2.1 Configuration de chaîne impaire

Soit $G = (V, E)$ un graphe $(k + 1)$ -arête connexe. Soit $(W_1, W_2, V_1, \dots, V_{2p})$ une partition de V . Soient $I_1 = \{4r, 4r + 1, r = 1, \dots, \lceil \frac{p-1}{2} \rceil\} \setminus \{1, 2p\}$ (les entiers sont pris modulo $2p$) et $I_2 = \{2, \dots, 2p - 1\} \setminus I_1$. On dit que la partition $(W_1, W_2, V_1, \dots, V_{2p})$ induit une configuration de chaîne impaire si :

1. $||[V_i, W_j]|| = k - 1$ pour $(i, j) \in (I_1 \times \{1\}) \cup (I_2 \times \{2\})$,
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. $\delta(V_i) = [V_i, W_2] \cup [V_{i-1}, V_i] \cup [V_i, V_{i+1}]$) si $i \in I_1$ (resp. $i \in I_2$) (voir figure 1 pour $k = 3$ et p pair).

Soit $C = \cup_{i=1}^{2p-1} [V_i, V_{i+1}]$.

A une configuration de chaîne impaire on associe l'inégalité

$$x(C) \geq p. \tag{4}$$

L'inégalité (4) sera appelée *inégalité de chaîne impaire*.

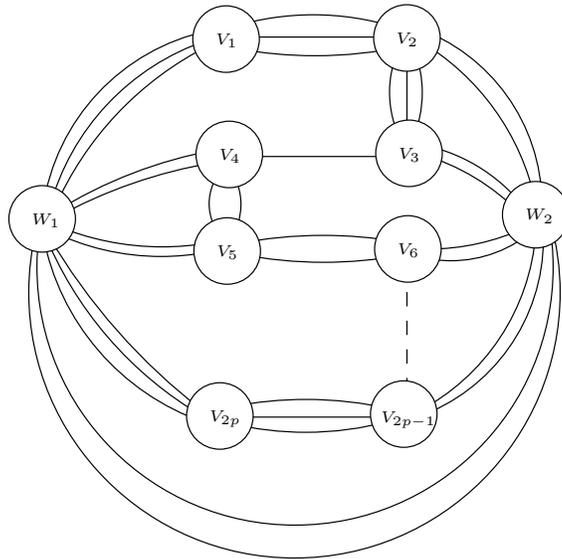


FIG. 1. Configuration de chaîne impaire

Nous avons le résultat suivant.

Theorem 1. *Les inégalités de (4) sont valides pour le polytope $kECSP(G)$.*

Preuve :

Puisque $|[V_{i-1}, V_i]| = k - 1$ et $x(\delta(V_i)) \geq k$ est une contrainte valide pour $kECSP(G)$, $\forall i \in \{2, \dots, 2p - 1\}$, nous avons

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

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

En multipliant les inégalités (5) par $\frac{p-s}{p}$ et les inégalités (6) par $\frac{s}{p}$ et en les sommant, on obtient l'inégalité suivante :

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

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

Puisque $|[W_1, W_2]| \leq k - 1$ et $x([W_1, W_2]) + \sum_{i \in \bar{I}} x([V_i, V_{i+1}]) \geq k$, on a

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

En sommant les inégalités (7) et (8) et en arrondissant le second membre à l'entier supérieur, on obtient l'inégalité (4). □

Dans la suite, nous donnons des conditions suffisantes pour que les inégalités (4) définissent des facettes de $kECSP(G)$.

Theorem 2. *Les inégalités (4) définissent des facettes de $kECSP(G)$ si les conditions suivantes sont satisfaites.*

1. *les sous-graphes $G[W_1]$, $G[W_2]$ et $G[V_i]$ pour $i = 1, \dots, 2p$, sont $(k + 1)$ -arête connexes,*
2. *$|[W_1, W_2]| = k - 1$, $|[V_1, W_1]| = k$ et $|[V_{2p}, W_1]| = k$ (resp. $|[V_{2p}, W_2]| = k$) si p est pair (resp. p impair).*

Preuve : Comme G est $(k + 1)$ -arête connexe, $kECSP(G)$ est donc de pleine dimension [15].

Considérons une configuration de chaîne impaire définie par la partition $(W_1, W_2, V_1, \dots, V_{2p})$, $p \geq 2$, de V . Sans perte de généralité, on peut supposer que p est pair. Soient e_i une arête fixée de $[V_i, V_{i+1}]$, $i = 1, \dots, 2p - 1$, $E_0 = \cup_{s=1}^p [V_{2s-1}, V_{2s}]$ et $E_1 = \cup_{s=1}^{p-1} [V_{2s}, V_{2s+1}]$. Soit $E' = E \setminus (E_0 \cup E_1)$. L'inégalité de chaîne impaire peut s'écrire sous la forme

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

On désigne par \mathcal{F} l'ensemble des points de $kECSP(G)$ qui vérifient (9) à l'égalité.

Soit $S_1 = E' \cup \{e_{2s-1}, i = 1, \dots, p\}$. S_1 induit un sous-graphe k -arête connexe et $x^{S_1}(E_0) = p$. Par conséquent, \mathcal{F} est une face propre de $kECSP(G)$.

Supposons qu'il existe une facette \mathcal{F}' de $kECSP(G)$, définie par l'inégalité $bx \geq \beta$, qui contient \mathcal{F} . Nous allons montrer que les inégalités $bx \geq \beta$ et (9) sont équivalentes.

Pour tout $f \in E'$, l'ensemble d'arêtes $S_2 = S_1 \setminus \{f\}$ induit un sous-graphe k -arête connexe. Comme $S_2 \in \mathcal{F}$, on a $b(f) = 0$, pour tout $f \in E'$.

Maintenant, si $f \in [V_{2s-1}, V_{2s}] \setminus \{e_{2s-1}\}$, $s \in \{1, \dots, p\}$, $S_3 = (S_1 \setminus \{e_{2s-1}\}) \cup \{f\}$ induit un sous-graphe k -arête connexe. Comme $x^{S_3}(E_0) = p$, on a $b(f) = b(e_{2s-1})$, pour $f \in [V_{2s-1}, V_{2s}] \setminus \{e_{2s-1}\}$. De façon similaire, en considérant $S_4 = E' \cup (\cup_{s=1}^{p-1} \{e_{2s}\} \cup \{e_1\})$, on voit que x^{S_4} vérifie l'inégalité (9) à l'égalité. Donc pour tout $f \in [V_{2s}, V_{2s+1}] \setminus \{e_{2s}\}$, pour $s \in \{1, \dots, p - 1\}$, $b(f) = b(e_{2s})$.

Soit $S_5 = (S_4 \setminus \{e_1\}) \cup \{e_{2s-1}\}$. Il est clair que x^{S_5} vérifie l'inégalité (9) à l'égalité. En conséquence, nous obtenons $b(e_1) = b(e_{2s-1})$, pour $s \in \{2, \dots, p\}$. Finalement, en considérant $S_6 = (S_4 \setminus \{e_1, e_{2s}\}) \cup \{e_{2s-1}, e_{2s+1}\}$, pour $s \in \{1, \dots, p\}$, il en résulte que $b(e_1) = b(e_{2s})$ pour $s = \{1, \dots, p\}$.

Par conséquent, le vecteur b s'écrit :

$$b(e) = \begin{cases} b(e_1) & \forall e \in E_0 \cup E_1, \\ 0 & \text{sinon,} \end{cases}$$

avec $b(e_1) > 0$.

Ainsi, les inégalités $x(E_0) + x(E_1) \geq p$ et $bx \geq \beta$ sont équivalentes, ce qui termine la preuve du théorème. □

Dans ce qui suit, nous présentons des opérations de lifting pour les inégalités de chaîne impaire. Celles-ci permettent d'obtenir des inégalités plus générales.

2.2 Procédure de lifting

Soit $G = (V, E)$ un graphe et soient $(W_1, W_2, V_1, \dots, V_{2p})$, $p \geq 2$, une partition de V induisant une configuration de chaîne impaire. Soit $G' = (V, E \cup L)$ un graphe obtenu de G en rajoutant un ensemble d'arêtes L .

On définit l'inégalité suivante :

$$x(C) + \sum_{e \in L} a(e)x(e) \geq p \quad (10)$$

avec

$$a(e) = \begin{cases} 2 & \text{si } e \in [V_i, V_j], i, j \in \{1, \dots, 2p\}, \\ 1 & \text{si } e \in [W_1, W_2] \text{ ou } e \in [V_i, W_j], i \in \{1, \dots, 2p\}, j \in \{1, 2\} \\ 0 & \text{sinon.} \end{cases}$$

L'inégalité (10) sera appelée *inégalité de chaîne impaire liftée*. Nous avons le résultat suivant qui sera donné sans preuve.

Theorem 3. *Les inégalités (10) sont valides pour le polytope $kECSP(G')$.*

2.3 Séparation des inégalités de chaîne impaire

Le problème de séparation pour une famille de contrainte consiste à déterminer pour un point x donné, si x vérifie les contraintes de cette famille, et sinon à trouver une contrainte violée par x .

Dans ce qui suit, nous donnons une heuristique pour séparer les contraintes de chaîne impaire. Pour un point $x \in \mathbb{R}^E$ donné, on peut remarquer que si l'on dispose d'une configuration de chaîne impaire et si l'ensemble d'arêtes $[V_1, V_2] \cup \dots \cup [V_{2p-1}, V_{2p}]$ est composé d'arêtes de valeurs fractionnaires telles que

$$x([V_{i-1}, V_i]) + x([V_i, V_{i+1}]) \leq 1, \text{ pour } i = 2, \dots, 2p - 1, \quad (11)$$

alors la contrainte de chaîne impaire est violée. Pour séparer les inégalités de chaîne impaire on peut utiliser la procédure suivante. On recherche dans le graphe support $G(x)$, induit par les arêtes $e \in E$ telles que $x(e) > 0$, une chaîne $\Gamma = \{e_1, \dots, e_{2p-1}\}$, $p \geq 2$, dont les arêtes sont toutes de valeurs fractionnaires et telles que $x(e_{i-1}) + x(e_i) \leq 1$ pour $i \in \{2, \dots, 2p-1\}$. Une fois qu'une telle chaîne est déterminée, on considère chaque sommet de Γ comme un ensemble V_i . On détermine ensuite deux ensembles W_1 et W_2 en calculant une coupe minimum entre l'ensemble de sommets $(\cup_{i \in I_1} V_i) \cup V_1$ (resp. $(\cup_{i \in I_1} V_i) \cup V_1 \cup V_{2p}$, si p est impair (resp. p pair) et l'ensemble de sommets $(\cup_{i \in I_2} V_i) \cup V_{2p}$ (resp. $\cup_{i \in I_2} V_i$), si p est impair (resp. p pair). On vérifie enfin si la partition $(W_1, W_2, V_1, \dots, V_{2p})$ induit bien une configuration de chaîne impaire et si l'inégalité de chaîne impaire correspondante est violée.

3 Opérations de réduction

Dans [10] Fonlupt et Mahjoub ont introduit le concept de *point extrême critique* pour le polytope $P(G, k)$ lorsque $k = 2$. Didi Biha et Mahjoub [8] ont étendu ce concept pour $k \geq 3$. Soit x un point extrême fractionnaire de $P(G, k)$. x est dit de *rang 1* s'il n'existe pas de point extrême fractionnaire \tilde{x} de $P(G, k)$ tel que l'ensemble des arêtes de valeurs fractionnaires pour \tilde{x} est strictement inclus dans l'ensemble des arêtes de valeurs fractionnaires pour x .

Didi Biha et Mahjoub [8] ont introduit les opérations de réductions suivantes qui permettent de préserver le rang 1 d'un point extrême fractionnaire x de $P(G, k)$.

θ_1 : supprimer une arête $e \in E$ telle que $x(e) = 0$;

θ_2 : contracter un ensemble $W \subset V$ tel que $G[W]$ soit k -arête connexe et $x(e) = 1$ pour tout $e \in E(W)$;

θ_3 : contracter un ensemble $W \subset V$ tel que $|W| \geq 2$, $|\overline{W}| \geq 2$, $|\delta(W)| = k$ et $E(\overline{W})$ contient au moins une arête de valeur fractionnaire;

θ_4 : contracter un ensemble $W \subset V$ tel que $G[W]$ soit $\lceil \frac{k}{2} \rceil$ -arête connexe, $|\delta(W)| = k + 1$ et $x(e) = 1$ pour tout $e \in E(W)$.

Soient $G' = (V', E')$ le graphe obtenu à partir de G en appliquant les opérations $\theta_1, \theta_2, \theta_3$ et θ_4 et soit x' la restriction de x à E' . Didi Biha et Mahjoub [8] ont montré que x' est un point extrême de $P(G', k)$ de rang 1 si et seulement si x est un point extrême de $P(G, k)$ de rang 1.

On dit qu'un point extrême de $P(G, k)$ est *critique* s'il est de rang 1 et si on ne peut pas lui appliquer une des opérations $\theta_1, \theta_2, \theta_3, \theta_4$.

Ces opérations de réduction peuvent être utilisées dans le cadre d'un algorithme de coupe et branchement pour résoudre $kECSP$. En effet, lors de la phase de coupe, la recherche d'une inégalité violée par un point x peut s'effectuer sur le graphe réduit G' , ce qui permettrait d'accélérer le processus de séparation.

Le lemme suivant, donné sans preuve, permet d'établir une relation entre la réduction d'un graphe G et les inégalités valides de $kECSP(G)$.

Lemme 1. *Supposons que $G = (V, E)$ est $(k + 1)$ -arête connexe. Soient $W \subset V$ et $ax \geq \alpha$, une inégalité valide pour $kECSP(G)$ tels que $G[W]$ soit $(k + 1)$ -arête connexe et $a(e) = 0$, pour tout $e \in E(W)$. Soient $G' = (V', E')$ le graphe obtenu en contractant W et a' la restriction de a à E' . L'inégalité $ax \geq \alpha$ définit une facette de $kECSP(G)$ si l'inégalité $a'x \geq \alpha$ définit une facette de $kECSP(G')$.*

Lemme 2. *Soient $G = (V, E)$ un graphe et \bar{x} un point de $P(G, k)$. Soient $G' = (V', E')$ et \bar{x}' un graphe et une solution obtenus respectivement à partir de G et \bar{x} par application des opérations $\theta_1, \theta_2, \theta_3, \theta_4$.*

Soient $a'x \geq \alpha'$ une inégalité de type (4) définie par une partition $\pi' = (W'_1, W'_2, V'_1, \dots, V'_{2p})$, $p \geq 2$, dans G' et $\pi = (W_1, W_2, V_1, \dots, V_{2p})$ la partition dans G obtenue par expansion des ensembles W'_1, W'_2 et $V'_i, i = 1, \dots, 2p$. Soit $ax \geq \alpha$ l'inégalité de type (10) obtenue à partir de $a'x \geq \alpha'$ par la procédure de lifting.

L'inégalité $ax \geq \alpha$ est violée par \bar{x} si $a'x \geq \alpha'$ est violée par \bar{x}' .

Preuve : Facile. □

4 Algorithme de coupe et branchement

Dans cette section, nous présentons un algorithme de coupe et branchement pour $kECSP$. Pour cela, nous présentons d'abord d'autres familles d'inégalités valides pour le polytope $kECSP(G)$. Pour chaque famille d'inégalités, nous discutons également du problème de séparation. Enfin, nous décrivons notre algorithme.

4.1 Autres Inégalités valides pour $kECSP(G)$

Les inégalités de coupe

Les inégalités de coupe sont introduites dans la section 1. Leur séparation peut se ramener au calcul de $|V| - 1$ flots maximum grâce à l'algorithme de Gomory-Hu [13]. Elles peuvent donc être séparées en temps polynomial.

Les inégalités de SP -partition

Didi Biha et Mahjoub [9] ont introduit une famille d'inégalités valides pour $kECSP(G)$.

Soit $\pi = (V_1, \dots, V_p)$, $p \geq 2$, une partition de V telle que $G[V_i]$ est connexe pour tout $i \in \{1, \dots, p\}$ et $k \geq 3$ un entier impair. Didi Biha et Mahjoub [9] ont montré que si la partition π induit un graphe série-parallèle alors l'inégalité suivante est valide pour $kECSP(G)$:

$$x(\delta(V_1, \dots, V_p)) \geq \left\lceil \frac{k}{2} \right\rceil p - 1. \quad (12)$$

Cette inégalité est appelée *inégalité de SP -partition*.

Dans le cas où k est pair l'inégalité (12) est dominée par les inégalités de coupe.

Le théorème suivant donne des conditions suffisantes pour que l'inégalité (12) définisse une facette de $kECSP(G)$ dans le cas où le graphe G est quelconque.

Theorem 4. [6]

Soient $G = (V, E)$ un graphe, $k \geq 3$ un entier impair et $\pi = (V_1, \dots, V_p)$, $p \geq 2$, une partition de V . L'inégalité (12) définit une facette de $kECSP(G)$ différente des facettes triviales si les conditions suivantes sont vérifiées :

1. $G[V_i]$ est $(k+1)$ -arête connexe, pour $i = 1, \dots, p$,
2. $|[V_i, V_{i+1}]| \geq \left\lceil \frac{k}{2} \right\rceil$, pour $i = 1, \dots, p \pmod{p}$,
3. G_π est outerplanaire.

Dans [7], Diarrassouba et Slama montrent que certaines conditions du théorème précédent sont aussi nécessaires.

Theorem 5. [7]

Soient $G = (V, E)$ un graphe $(k+1)$ -arête connexe et $\pi = (V_1, \dots, V_p)$, $p \geq 2$, une partition de V . Si l'inégalité de SP -partition induite par π définit une facette de $kECSP(G)$ alors :

1. $|[V_i, V_{i+1}]| \geq \left\lceil \frac{k}{2} \right\rceil$, pour $i = 1, \dots, p \pmod{p}$,
2. G_π est outerplanaire.

Dans [4], Chopra a introduit le même type d'inégalités lorsque le graphe G est outerplanaire et lorsqu'une arête peut être utilisée plus d'une fois. Rappelons que la classe des graphes outerplanaires est une sous-classe de la classe des graphes série-parallèles. Chopra a défini aussi une procédure de lifting qui généralise ces inégalités au cas où le graphe G n'est plus outerplanaire. Cette procédure se présente comme suit. Soit $\overline{E} = \{e_1, \dots, e_l\}$, $l \geq 1$, tel que $G'_\pi = (V_\pi, E_\pi \cup \overline{E})$ n'est pas outerplanaire. Alors l'inégalité suivante est valide pour $kECSP(G)$:

$$\sum_{e \in E_\pi} x(e) + \sum_{i=1}^l a(e_i) x(e_i) \geq \left\lceil \frac{k}{2} \right\rceil p - 1, \quad (13)$$

où $a(e_i)$ est la longueur d'un plus court chemin dans G_π entre les extrémités de l'arête e_i (la longueur étant le nombre d'arêtes qui compose le chemin).

La séparation des inégalités de SP -partition peut être effectuée de façon polynomiale dans la classe des graphes série-parallèles grâce à l'algorithme de Baïou et al. [1], basé sur les fonctions sous-modulaires. Récemment, Didi Biha et al. [5] ont développé un algorithme purement combinatoire permettant de séparer ces inégalités dans la classe des graphes série-parallèles.

Pour séparer ces contraintes dans le cas général, nous proposons l'heuristique suivante qui se base sur les conditions nécessaires et suffisantes données dans le théorème 5. Cette heuristique fonctionne comme suit. Nous recherchons une chaîne d'arêtes C dont les sommets sont reliés par des arêtes multiples ayant au moins $\lceil \frac{k}{2} \rceil$ arêtes et contenant au moins une arête fractionnaire. Soit $\{u_1, \dots, u_p\}$ les sommets constituant cette chaîne. On construit alors la partition $\pi = (V_1, \dots, V_p, V_{p+1})$, $p \geq 2$, avec $V_i = \{u_i\}$, $i = 1, \dots, p$ et $V_{p+1} = V \setminus \{u_1, \dots, u_p\}$. Nous vérifions ensuite que le graphe G_π est outerplanaire en vérifiant qu'un ensemble V_i , $i = 1, \dots, p$, n'est pas relié à un ensemble V_j tel que $j \notin \{i-1, i+1, p+1\}$. Si le graphe G_π n'est pas outerplanaire, on applique la procédure de lifting de Chopra.

Les inégalités de partition

Soient $k \geq 3$ un entier impair et $\pi = (V_0, \dots, V_p)$, $p \geq 1$ une partition de V . Grötschel et al. [15] ont montré que l'inégalité

$$x(\delta(V_0, \dots, V_p)) \geq \left\lceil \frac{k(p+1)}{2} \right\rceil \quad (14)$$

est valide pour $kECSP(G)$. Cette inégalité est appelée *inégalité de partition*.

Lorsque l'entier k est pair, les inégalités (14) sont redondantes par rapport aux inégalités de coupe. Grötschel et al. [15] donnent des conditions nécessaires et suffisantes pour que les inégalités (14) définissent des facettes de $kECSP(G)$.

Lorsque $k \geq 3$ et impair, nous utilisons l'heuristique suivante pour séparer ces inégalités.

On cherche d'abord dans le graphe support $G(x)$ un cycle constitué d'arêtes de valeurs fractionnaires. Si $C = \{u_1, \dots, u_l\} \subseteq V$, $l \geq 3$, est un tel cycle, on construit la partition $\pi = (V_0, \dots, V_l)$ telle que $V_i = \{u_i\}$, $i = 1, \dots, l$ et $V_0 = V \setminus \{u_1, \dots, u_l\}$.

Si un tel cycle n'existe pas, on construit l'ensemble des sommets qui sont adjacents à au moins une arête de valeur fractionnaire dans $G(x)$. Soit $V_f = \{u_1, \dots, u_l\}$, $l \geq 3$, cet ensemble. La partition $\pi = (V_0, \dots, V_l)$ est alors construite telle que $V_i = \{u_i\}$, $i = 1, \dots, l$, et $V_0 = V \setminus \{u_1, \dots, u_l\}$.

Les inégalités de F -partition

Dans [19], Mahjoub a introduit une famille d'inégalités valides pour le polytope $2ECSP(G)$ comme suit.

Soient $\pi = (V_0, V_1, \dots, V_p)$, $p \geq 2$, une partition de V et $F \subseteq \delta(V_0)$ un ensemble d'arêtes de cardinalité impaire. Mahjoub [19] a montré que l'inégalité suivante est valide pour le polytope $2ECSP(G)$:

$$x(\delta(V_0, V_1, \dots, V_p) \setminus F) \geq p - \frac{|F| - 1}{2}. \quad (15)$$

Ces inégalités sont appelées *inégalités de F -partition*. Dans [6], Didi Biha a généralisé ces inégalités pour tout k . Si $\pi = (V_0, V_1, \dots, V_p)$, $p \geq 2$, est une partition de V , $F \subseteq \delta(V_0)$ et $\Delta = \delta(V_0, V_1, \dots, V_p) \setminus F$ alors la contrainte

$$x(\Delta) \geq \left\lceil \frac{pk - |F|}{2} \right\rceil \quad (16)$$

est valide pour $kECSP(G)$.

Lorsque kp et $|F|$ sont de même parité, la contrainte (16) est redondante par rapport aux contraintes de coupe. Didi Biha [6] donne des conditions suffisantes pour que ces inégalités définissent des facettes de $kECSP(G)$.

Pour séparer les inégalités de F -partition, nous avons développé une heuristique. Celle-ci peut être présentée de la manière suivante. Elle commence par créer une partition $\pi = (V_0, V_1, \dots, V_p)$ en utilisant la méthode employée pour la séparation des inégalités de partition. L'ensemble $F \subseteq \delta(V_0)$ est choisi tel que $F = \{e \in \delta(V_0) | x(e) > 0\}$. Si $|F|$ et kp sont de même parité, on retire de F l'arête e ayant la plus grande valeur $x(e)$.

Dans le cas où l'ensemble V_0 est vide ou si la contrainte de F -partition induite par π et F n'est pas violée, on construit la partition $\pi = (V_0, V_1, \dots, V_p)$, $p \geq 2$, telle que $x(\delta(V_i)) = k$, pour $i = 1, \dots, p$, et $V_0 = V \setminus \{V_1, \dots, V_p\}$. Une telle partition peut être construite en utilisant l'arbre de Gomory-Hu [13] et en associant à chaque arête e du graphe support le poids $1 - x(e)$.

4.2 Opérations de réduction et lifting de contraintes

En utilisant les opérations de réduction $\theta_1, \theta_2, \theta_3, \theta_4$ dans un algorithme de coupe et branchement pour k ECSP, une question qui se pose est comment transformer une contrainte violée dans le graphe réduit en une contrainte violée dans le graphe initial. Dans ce qui suit, nous décrivons une procédure de lifting qui permet d'assurer cette transformation pour les inégalités de type (1), (14) ou (16).

Soient $G = (V, E)$ un graphe et \bar{x} un point de $P(G, k)$. Soient $G' = (V', E')$ et \bar{x}' un graphe et un point obtenus à partir de G et x par application des opérations $\theta_1, \theta_2, \theta_3, \theta_4$.

A une contrainte $a'x \geq \alpha'$ de type (1), (14) ou (16) dans G' , on associe la contrainte $ax \geq \alpha$ dans G telle que

$$a(e) = \begin{cases} a'(e) & \forall e \in E', \\ 1 & \forall e = uv \text{ telle que } u \in V', v \in V' \text{ et } e \text{ a été supprimée par l'opération } \theta_1, \\ 0 & \text{sinon,} \end{cases}$$

et $\alpha = \alpha'$.

Nous avons le lemme suivant.

Lemme 3. *Soient $a'x \geq \alpha'$ une inégalité de type (1), (14) ou (16) définie par une partition $\pi' = (V'_1, \dots, V'_p)$ dans G' et $\pi = (V_1, \dots, V_p)$ la partition dans G obtenue par expansion des ensembles $V'_i, i = 1, \dots, p$. Soit $ax \geq \alpha$ l'inégalité obtenue à partir de $a'x \geq \alpha'$ par la procédure de lifting. Si, pour tout $i \in \{1, \dots, p\}$, $G[V_i]$ est $(k+1)$ -arête connexe alors nous avons les assertions suivantes.*

1. *L'inégalité $ax \geq \alpha$ est du même type que $a'x \geq \alpha'$ et est induite par la partition π .*
2. *$ax \geq \alpha$ définit une facette de k ECSP(G) si $a'x \geq \alpha'$ définit une facette de k ECSP(G').*
3. *$ax \geq \alpha$ est violée par \bar{x} si $a'x \geq \alpha'$ est violée par le point \bar{x}' .*

Preuve : Facile. □

4.3 Implémentation des opérations de réduction

Afin d'appliquer les opérations de réduction dans un algorithme de coupe et branchement, nous les avons implémenté. Le paragraphe suivant décrit l'implémentation de l'opération θ_4 . L'implémentation des opérations θ_1 et θ_2 est simple et ne sera donc pas présentée ici. Quant à l'implémentation de l'opération θ_3 , elle est similaire à celle de l'opération θ_4 .

Soient $G = (V, E)$ un graphe et $x \in \mathbb{R}^E$ un point qui satisfait les inégalités de coupe et les inégalités triviales. L'algorithme commence par calculer l'arbre de Gomory-Hu associé au graphe support $G(x)$, en mettant 1 comme capacité sur toutes les arêtes de $G(x)$. Une arête f de l'arbre de Gomory-Hu ayant une capacité $k+1$ définit une coupe $\delta(W)$, $W \subset V$, de cardinalité $k+1$ dans $G(x)$. On vérifie alors que le sous-graphe $G[W]$ est $\lceil \frac{k}{2} \rceil$ -arête connexe et qu'il ne contient pas d'arête fractionnaire. Si c'est le cas, on contracte W , sinon on vérifie si \overline{W} satisfait ou non ces conditions. Avant de contracter un ensemble W , on s'assure que $|W| \geq 2$ et $|\overline{W}| \geq 2$ pour éviter de réduire le graphe à deux sommets.

On recommence cette procédure jusqu'à ce que l'arbre de Gomory-Hu calculé ne nous donne plus de coupe de cardinalité $k+1$ ou qu'aucun ensemble $W \subset V$ tel que $|\delta(W)| = k+1$, ne satisfasse les autres conditions de l'opération.

La figure 2 montre un exemple d'application des opérations θ_1 et θ_4 pour $k = 3$ et un point $x \in \mathbb{R}^E$ fractionnaire qui satisfait les inégalités de coupe et les inégalités triviales. Après l'application de ces opérations on contracte l'ensemble $v'_4 = \{v_4, v_5, v_6\}$. Sur le graphe réduit, on peut voir une configuration de chaîne impaire qui induit une inégalité de chaîne impaire définissant une facette du polytope.

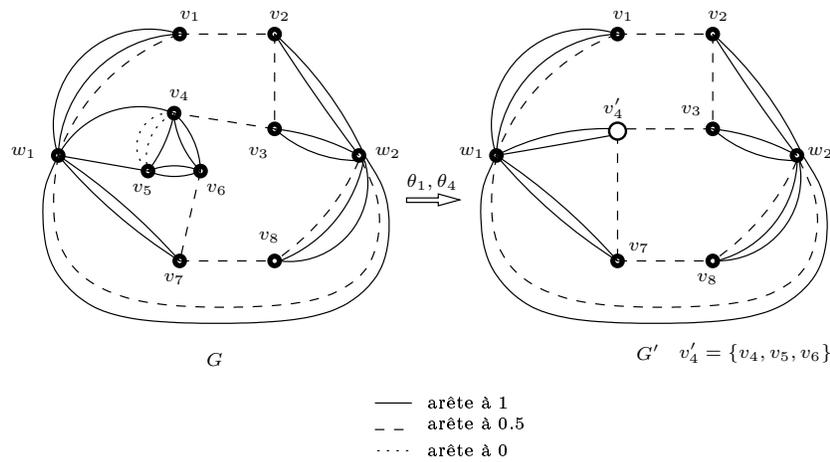


FIG. 2. Operation de réduction conduisant à une inégalité de chaîne impaire définissant une facette

4.4 Description de l'algorithme de coupe et branchement

L'algorithme commence par résoudre la relaxation linéaire suivante de $kECSP$:

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

Une solution optimale $y \in \mathbb{R}^E$ de ce programme linéaire est réalisable pour $kECSP$ si y est entier et toutes les inégalités de coupe sont satisfaites. Ceci n'est pas toujours le cas. A chaque itération de l'algorithme, on génère donc de nouvelles inégalités valides pour le polytope $kECSP(G)$ et violées par y . La séparation de ces inégalités est effectuée dans l'ordre suivant :

- inégalités de coupe,
- inégalités de SP -partition,
- inégalités de chaîne impaire,
- inégalités de F -partition,
- inégalités de partition.

Après la séparation des inégalités de coupe, si la solution est toujours fractionnaire, on applique les opérations de réduction décrites précédemment puis nous séparons les autres familles de contraintes. Ces contraintes sont liftées avant d'être introduites dans le programme linéaire.

5 Résultats expérimentaux

Nous avons utilisé notre algorithme de coupe et branchement pour résoudre $kECSP$ lorsque $k = 3$ et $k = 4$. L'algorithme a été implémenté dans les langages C et C++, en utilisant les logiciels ABACUS 2.4 pour gérer l'arbre de branchements et CPLEX 9.0 comme solveur linéaire. Les expérimentations ont été effectuées sur une machine avec un processeur Pentium IV à 3.4 Ghz et 1024 Mo de mémoire RAM avec Linux comme système d'exploitation.

Les instances qui nous ont servi de tests sont des graphes de la TSPLIB (graphes complets euclidiens). Les différentes entrées des tableaux de résultats qui vont suivre sont les suivantes :

- Nom : le nom de l'instance,
- Taille : le nombre de sommets du graphe,
- NSub : le nombre de sous-problèmes dans l'arbre de branchement,
- NCoupe : nombre total de contraintes de coupes générées par la séparation,
- NSP : nombre total de contraintes de SP -partition introduites,

- NCH : nombre total de contraintes de chaîne impaire obtenues,
- NFP : nombre total de F -partition générées,
- NP : nombre total de contraintes de partition introduites,
- C : coût de la solution avant la phase de branchements,
- COpt : coût de la solution optimale,
- Gap : écart relatif, en pourcent, entre la solution optimale et la solution avant branchements, c'est-à-dire $\frac{COpt-C}{COpt} * 100$,
- TT : temps CPU total en secondes.

Le temps limite de résolution a été fixé à cinq heures. Pour certaines instances dont le temps de résolution dépasse la durée limite, nous avons utilisé une heuristique primale pour obtenir une solution réalisable du problème. Avec la solution avant branchement, on obtient ainsi un encadrement de la solution optimale du problème. Ces instances sont marquées par "*".

Les tableaux 1 et 2 donnent les résultats expérimentaux obtenu pour 3ECSP respectivement avec et sans les opérations de réduction de graphes.

Nom	Taille	NSub	NC	NSP	NCH	NFP	NP	C	COpt	Gap	TT
burma14	14	1	4	3	0	0	4	5530	5530	0	0.02
ulyse16	16	3	5	7	1	15	7	11367.7	11412	0.39	0.12
gr21	21	1	5	6	1	0	2	4740	4740	0	0.04
fri26	26	1	9	5	0	0	0	1543	1543	0	0.03
bayg29	29	7	14	16	2	33	2	2634	2639	0.19	0.35
dantzig42	42	71	41	31	6	90	18	1201.83	1210	0.68	6.73
att48	48	61	34	34	5	60	9	17402.2	17499	0.56	5.97
gr48	48	23	26	30	11	51	5	8281.67	8313	0.38	3.84
berlin52	52	35	36	31	12	103	6	12545	12601	0.45	3.25
brazil58	58	1	46	42	2	36	29	42527	42527	0	5.39
eil76	76	7	11	13	3	257	4	875.5	876	0.06	3.18
pr76	76	2891	110	168	37	1716	70	184505	187283	1.51	854.35
rat99	99	61	43	24	15	366	22	2021.33	2029	0.38	47.29
rd100	100	169	100	74	11	419	18	13226.8	13284	0.43	212.48
lin105	105	1045	180	202	46	825	67	25354.4	25530	0.69	1342.73
gr120	120	129	59	55	17	650	14	11415.7	11442	0.23	423.47
ch130	130	975	122	105	17	1348	40	10349.2	10400	0.49	2435.55
ch150	150	733	110	109	33	1045	33	10962.5	11027	0.59	3364.47
d198	198	243	168	93	25	809	97	25585	25624	0.15	3261.78
gr202	202	69	74	68	18	561	20	65659.8	65729	0.11	653.49
*gr229	229	1219	179	245	23	1568	94	221699	223907	1.00	18000
*a280	280	459	142	84	56	2539	59	4343.26	4460	2.69	18000

TAB. 1. Résultats expérimentaux avec les opérations de réduction

Nous remarquons que toutes les instances du tableaux 1 ont été résolues à l'optimum, à l'exception des deux dernières instances. Certaines instances ont même pu être résolues avant la phase de branchements. L'instance fri26 a été résolue uniquement grâce aux contraintes de coupe et de SP -partition. Les contraintes de chaîne impaire, de F -partition et de partition interviennent parfois en nombre élevé pour une grande majorité des instances.

La comparaison du tableau 2 avec le tableau 1 nous montre que les temps de résolution sont meilleurs lorsque les opérations de réduction sont introduites. Par exemple, l'instance pour l'instance d198, la résolution met 10 fois plus de temps sans les opérations de réduction. Seules les instances eil76 et ch150 sont résolues plus rapidement sans les opérations de réduction. Le nombre de nœuds dans l'arbre de branchement devient quant à lui plus élevé pour la majorité des instances lorsque les opérations de réduction sont supprimées. Ces opérations permettent donc d'améliorer la résolution de 3ECSP en réduisant la taille de l'arbre de branchement et le temps de résolution.

Nous avons aussi voulu vérifier l'importance des inégalités de chaîne impaire et de SP -partition. Nous avons donc résolu le problème sans ces inégalités. Les tableaux 3 et 4 pré-

Nom	Taille	NSub	NC	NSP	NCH	NFP	NP	C	COpt	Gap	TT
burma14	14	1	3	4	1	0	3	5530	5530	0	0.12
ulyse16	16	7	7	5	1	6	4	11379.5	11412	0.29	0.17
gr21	21	1	6	6	11	2	2	4740	4740	0	0.11
fri26	26	1	8	9	1	7	0	1543	1543	0	0.12
bayz29	29	17	24	17	5	34	9	2632.5	2639	0.25	0.87
dantzig42	42	67	53	39	20	111	27	1201.81	1210	0.68	10.25
att48	48	75	30	28	7	18	13	17412.9	17499	0.49	8.35
gr48	48	49	40	44	11	85	4	8274.75	8313	0.46	11.49
berlin52	52	15	31	28	19	44	4	12546.3	12601	0.44	4.77
brazil58	58	3	50	27	1	28	31	42435.2	42527	0.22	6.94
eil76	76	1	11	5	6	137	2	876	876	0	1.05
pr76	76	4165	105	267	84	1465	50	184594	187283	1.46	2458.54
rat99	99	37	40	19	12	177	14	2021.56	2029	0.37	66.0
rd100	100	339	110	93	25	568	20	13231.3	13284	0.40	970.90
lin105	105	549	161	160	50	569	53	25375.2	25530	0.61	2047.91
gr120	120	105	40	38	6	187	12	11420.1	11442	0.19	241.53
ch130	130	2655	138	144	46	1616	42	10345.9	10400	0.52	15720.34
ch150	150	91	80	73	22	363	22	10984.7	11027	0.39	1312.05
d198	198	435	160	111	25	499	53	25545.4	25624	0.31	17802.99
gr202	202	37	74	63	13	277	23	65698.3	65729	0.04	2931.21
*gr229	229	239	131	193	33	898	63	221741	223411	0.75	18000
*a280	280	177	138	151	53	983	41	4344.8	4463	2.72	18000

TABLE 2. Résultats expérimentaux sans les opérations de réduction

sentent les résultats obtenus respectivement sans ces inégalités. Ces résultats sont obtenus en appliquant les opérations de réduction.

Nom	Taille	NSub	NC	NSP	NCH	NFP	NP	C	COpt	Gap	TT
gr21	21	1	8	0	10	10	6	4740	4740	0	0.10
fri26	26	3	8	0	0	5	13	1538.25	1543	0.31	0.14
dantzig42	42	69	44	34	0	103	13	1203.13	1210	0.57	4.66
berlin52	52	59	37	39	0	104	8	12538.5	12601	0.50	4.04
rd100	100	283	106	93	0	707	32	13224.7	13284	0.45	310.75
gr120	120	137	55	58	0	471	13	11416.3	11442	0.22	220.03
ch130	130	2321	125	141	0	1444	58	10343.6	10400	0.54	3868.92
d198	198	327	154	94	0	721	56	25588.1	25624	0.14	2922.28
gr202	202	109	83	82	0	898	25	65668.3	65729	0.09	990.63

TABLE 3. Résultats expérimentaux sans les contraintes de chaîne impaire

Pour la plupart des instances du tableau 4, le temps de résolution est moins élevé lorsque les inégalités de chaîne impaire n'interviennent pas. Cependant, pour l'instance ch130, le temps de résolution du problème passe de 2435 seconde avec les inégalités de chaîne impaire à 3868 secondes sans ces inégalités. De plus, pour certaines instances (berlin52, rd100, ch130), le gap entre la solution optimale et la solution avant branchements est plus élevé lorsque les inégalités de chaîne impaire ne sont pas prise en compte.

On peut en conclure qu'elles permettent dans certains cas d'approcher plus rapidement la solution optimale du problème.

En comparant le tableau 4 avec le tableau 1, on voit que les temps de résolution deviennent plus élevés lorsque les inégalités de SP -partition ne sont pas prises en compte. L'instance d198 est particulièrement significative car elle n'a pas pu être résolue au bout de cinq heures de temps lorsque les inégalités de SP -partition ne sont pas utilisées alors qu'elle a été résolue lorsque celles-ci sont présentes. En outre, les gaps obtenus entre les solutions optimales et

Nom	Taille	NSub	NC	NSP	NCH	NFP	NP	C	COpt	Gap	TT
gr21	21	1	8	0	10	10	6	4740	4740	0	0.10
fri26	26	3	8	0	0	5	13	1538.25	1543	0.31	0.14
dantzig42	42	123	47	0	4	72	46	1199.77	1210	0.85	10.59
berlin52	52	41	42	0	22	240	35	12526.3	12601	0.60	7.29
rd100	100	997	155	0	84	1246	146	13205.3	13284	0.60	1145.29
gr120	120	99	58	0	16	555	65	11420.3	11442	0.19	244.16
ch130	130	2303	158	0	87	2112	164	10318.1	10400	0.79	5587.79
d198	198	899	225	0	79	1818	244	25378.2	-	-	18000
gr202	202	161	104	0	39	1388	101	65649.3	65729	0.12	3788.37

TAB. 4. Résultats expérimentaux sans les contraintes de SP -partition

les solutions avant branchements sont, dans l'ensemble, plus grands lorsque les inégalités de SP -partition n'interviennent pas. Cela montre l'importance de ces contraintes dans la résolution de $3ECSP$. Toutefois, certaines instances (gr21, fri26) ont quand même été résolues sans branchements et ce, malgré l'absence des contraintes de SP -partition. Cela signifie que les inégalités de F -partition, de chaîne impaire et de partition peuvent aussi jouer un rôle important dans la résolution de $kECSP$ lorsque k est impair.

Une question soulevée dans la résolution de $kECSP$ est de savoir si le cas où k est pair est plus facile à résoudre que le cas où k est impair. Pour cela nous avons résolu des instances de $kECSP$ avec $k = 4$ et nous avons comparé les résultats avec ceux obtenus pour $k = 3$. Le tableau 5 donne les résultats obtenus pour $k = 4$ pour quelques une des instances. Rappelons que pour des valeurs paires de k , les inégalités de partition et de SP -partition sont redondantes et dominées par rapport aux inégalités de coupe. Notons aussi que les opérations de réduction de graphes sont appliquées lors des phases de séparation.

Nom	Taille	NSub	NC	NCH	NFP	C	COpt	Gap	TT
berlin52	52	1	5	0	2	18295	18295	0	0.10
pr76	76	1	3	0	4	266395	266395	0	0.21
rd100	100	1	10	1	91	19130	19130	0	5.26
gr120	120	1	6	0	36	16400	16400	0	2.73
ch130	130	1	12	0	132	14854	14854	0	5.92
d198	198	3	43	0	71	35401.5	35404	0.01	16.17
gr202	202	3	13	3	220	94826.5	94841	0.02	88.10
gr229	229	1	24	2	15	318565	318565	0	3.33
a280	280	1	3	0	180	6317	6317	0	60.45

TAB. 5. Résultats expérimentaux pour $k = 4$

On voit que la quasi-totalité des instances a été résolue avant la phase de branchements. Seules les instances d198 et gr202 ont nécessité chacune la génération de 2 sous-problèmes dans l'arbre de branchement. Aussi, les instances sont toutes résolues en un temps inférieur à 2 minutes. Nous remarquerons, en particulier, que les instances gr229 et a280 ont été résolues à l'optimum alors que pour $k = 3$ nous ne disposons que d'une solution approchée au bout de cinq heures.

Les résultats que nous avons obtenus pour $k = 3$ et $k = 4$ peuvent aussi être comparés avec les résultats obtenus par Kerivin et al. [17] pour $k = 2$. Ces observations laissent pressentir que $kECSP$ est plus simple à résoudre lorsque l'entier k est pair.

6 Conclusion

Dans ce papier, nous avons étudié le problème de conception de réseaux fiables avec forte connectivité. Nous avons développé un algorithme de coupe et branchement pour ce problème.

Les résultats expérimentaux ont bien mis en évidence l'utilité des contraintes de chaîne impaire que nous avons introduites et des contraintes de SP -partition. Ils montrent également l'efficacité des opérations de réduction.

Il serait intéressant de décrire des conditions pour que les inégalités de chaîne impaire liftées définissent des facettes. Ces conditions peuvent être utilisées pour améliorer la procédure de séparation de ces contraintes.

Une question qui peut se poser également est de caractériser les points extrêmes critiques du polytope $P(G, k)$. Cela peut donner lieu à de nouvelles facettes pour le polytope associé et aussi améliorer l'algorithme de coupe et branchement.

Références

1. M. Baïou, F. Barahona and A.R. Mahjoub, "Separation of partition inequalities", *Mathematics of Operational Research* 25, 2000, pp.243-254.
2. F. Barahona and A. R. Mahjoub, "On two-connected subgraph polytopes", *Discrete Mathematics* 147, 1995, pp. 19-34.
3. 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, 320-329.
4. S. Chopra, "The k -edge connected spanning subgraph polyhedron", *SIAM Journal on Discrete Mathematics* 7, 1994, 245-259.
5. M. Didi Biha, A.R. Mahjoub and L. Slama, "On the separation of partition inequalities", *INOC 2005, Proceedings*, pp B2.500-B2.505.
6. M. Didi Biha, "Graphes k -arêtes connexes et polyèdres", *Thèse de doctorat*, Université de Bretagne Occidentale, Brest, 1998.
7. I. Diarrassouba and L. Slama, "Etude polyédrale du problème du sous-graphe k -arête connexe dans les graphes outerplanaires et série-parallèles", Preprint, 2005.
8. 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.
9. M. Didi Biha and A.R. Mahjoub, " k -edge connected polyhedra on series-parallel graphs", *Operations Research Letters* 19, 1996, pp. 71-78.
10. J. Fonlupt and A.R. Mahjoub, "Critical extreme points of the 2-edge connected spanning subgraph polytope", à paraître dans *Mathematical Programming*, 1996.
11. 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.
12. M.R. Garey and D.J. Johnson, "Computer and Intractability : A Guide to the Theory of NP-completeness", *Freeman, San Francisco*, 1979.
13. R.E. Gomory and T.C. Hu, "Multi-Terminal Network Flows", *Science for Industrial and Applied Mathematics*, 1961, pp. 551-570.
14. M. Grötschel and C. Monma, "Integer polyhedra arising from certain network design problems with connectivity constraints", *SIAM Journal on Discrete Mathematics*, Vol. 3, No. 4, 1990, pp. 502-523.

15. M. Grötschel, C. Monma and 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, pp. 121-141. AMS/ACM, 1991.
16. H. Kerivin and A.R. Mahjoub, "Design of Survivable Networks : A Survey", *Networks*, Vol. 46(1), 2005, pp. 1-21.
17. 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.
18. C.-W. Ko and C.L. Monma, "Heuristics for designing highly survivable communication networks", Technical Report, Bellcore, Morristown, New Jersey, 1989.
19. A.R. Mahjoub, "Two-edge Connected spanning subgraphs and polyhedra", *Mathematical Programming* 64, 1994, pp.199-208.
20. A.R. Mahjoub, "On perfectly Two-edge connected subgraphs and polyhedra", *Discrete Mathematics* 170, 1997, pp.153-172.
21. C.L. Monma, B.S. Munson and W.R. Pulleyblank, "Minimum weight two-connected spanning networks", *Operations Research* 37, 1989, pp. 153-171.