Algorithme de génération de coupes pour le problème du sous graphe 2-arête connexe

Sylvie Borne

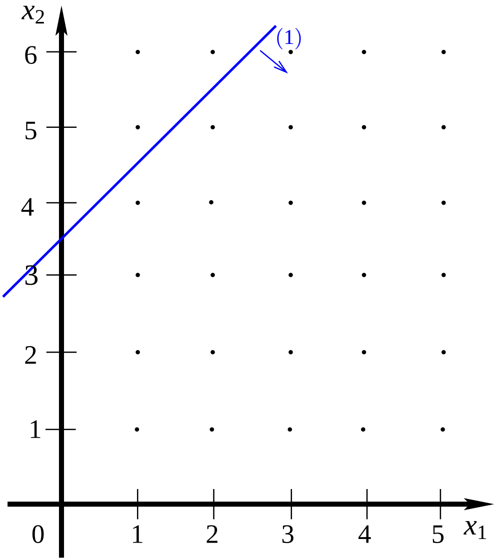
LIPN - UMR CNRS 7030, Université Paris 13.

MES AOC

11 octobre 2011

Outline

Un exemple de programme linéaire

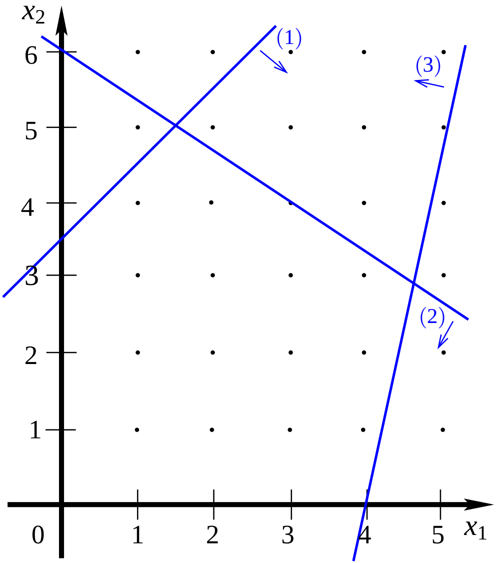


PL

$$\text{Min } 3x_1 + 2x_2$$

$$\begin{aligned} -2x_1+2x_2 &\leq 7 \quad (1) \\ 2x_1+3x_2 &\leq 18 (2) \\ 9x_1-2x_2 &\leq 36 (3) \\ x_1, \quad x_2 &\geq 0 \end{aligned}$$

Un exemple de programme linéaire

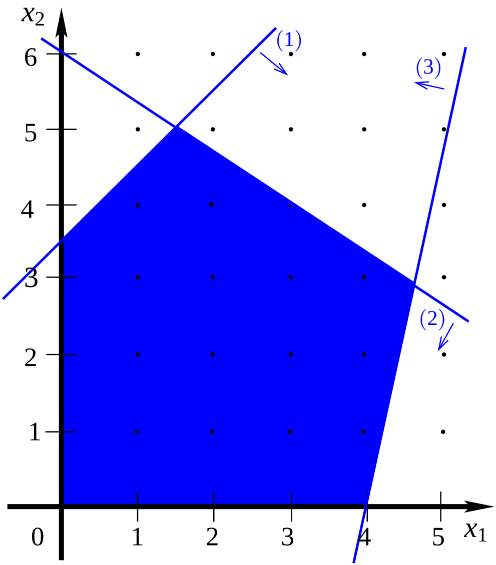


PL

$$\text{Min } 3x_1 + 2x_2$$

$$\begin{aligned} -2x_1 + 2x_2 &\leq 7 \quad (1) \\ 2x_1 + 3x_2 &\leq 18 \quad (2) \\ 9x_1 - 2x_2 &\leq 36 \quad (3) \\ x_1, \quad x_2 &\geq 0 \end{aligned}$$

Un exemple de programme linéaire

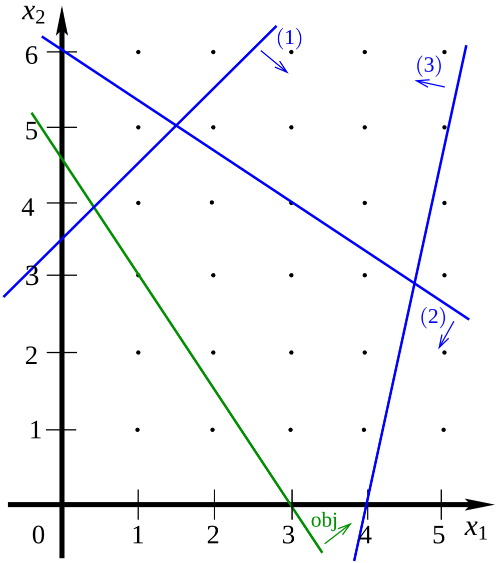


PL

$$\text{Min } 3x_1 + 2x_2$$

$$\begin{aligned} -2x_1 + 2x_2 &\leq 7 \quad (1) \\ 2x_1 + 3x_2 &\leq 18 \, (2) \\ 9x_1 - 2x_2 &\leq 36 \, (3) \\ x_1, \quad x_2 &\geq 0 \end{aligned}$$

Un exemple de programme linéaire

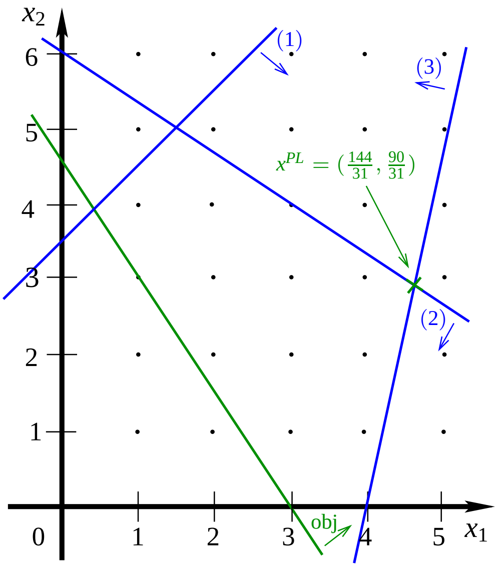


PL

$$\text{Min } 3x_1 + 2x_2$$

$$\begin{aligned} -2x_1 + 2x_2 &\leq 7 \quad (1) \\ 2x_1 + 3x_2 &\leq 18 (2) \\ 9x_1 - 2x_2 &\leq 36 (3) \\ x_1, \quad x_2 &\geq 0 \end{aligned}$$

Un exemple de programme linéaire

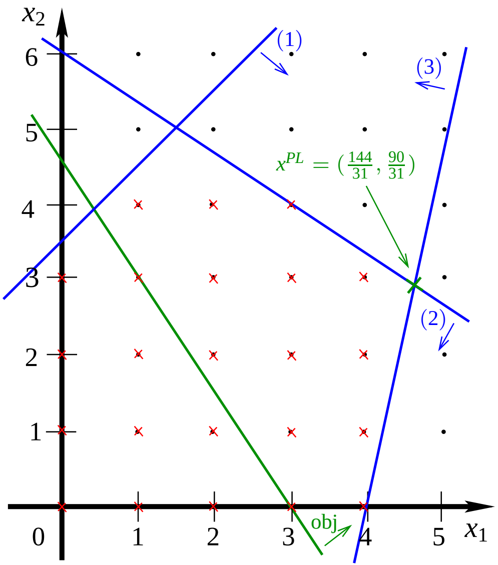


PL

$$\text{Min } 3x_1 + 2x_2$$

$$\begin{aligned} -2x_1 + 2x_2 &\leq 7 \quad (1) \\ 2x_1 + 3x_2 &\leq 18 \, (2) \\ 9x_1 - 2x_2 &\leq 36 \, (3) \\ x_1, \quad x_2 &\geq 0 \end{aligned}$$

Un exemple de programme linéaire en nombres entiers

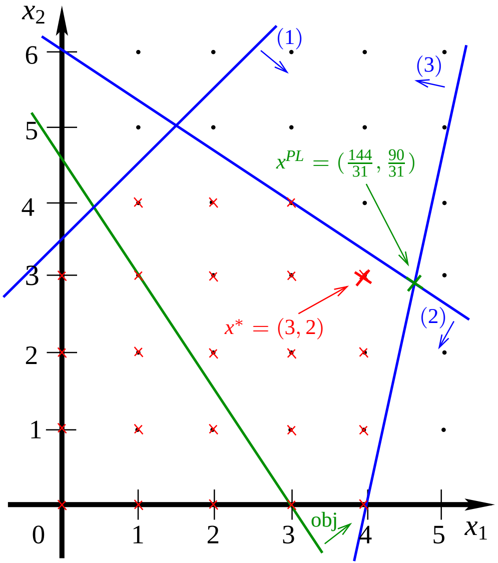


PLNE

$$\text{Min } 3x_1 + 2x_2$$

$$
\begin{aligned}
-2x_1 + 2x_2 &\leq 7 \quad (1)\\
2x_1 + 3x_2 &\leq 18 (2)\\
9x_1 - 2x_2 &\leq 36 (3)\\
x_1, \quad x_2 &\in N
\end{aligned}
$$

Un exemple de programme linéaire en nombres entiers



PLNE

$$\text{Min } 3x_1 + 2x_2$$

$$\begin{aligned} -2x_1 + 2x_2 &\leq 7 \quad (1) \\ 2x_1 + 3x_2 &\leq 18 (2) \\ 9x_1 - 2x_2 &\leq 36 (3) \\ x_1, \quad x_2 &\in N \end{aligned}$$

Outline

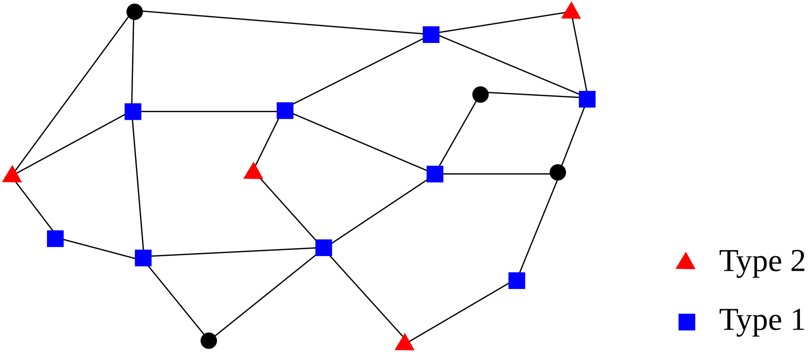
Problème de conception de réseaux fiables

Modélisation : Grötschel, Monma, 1990

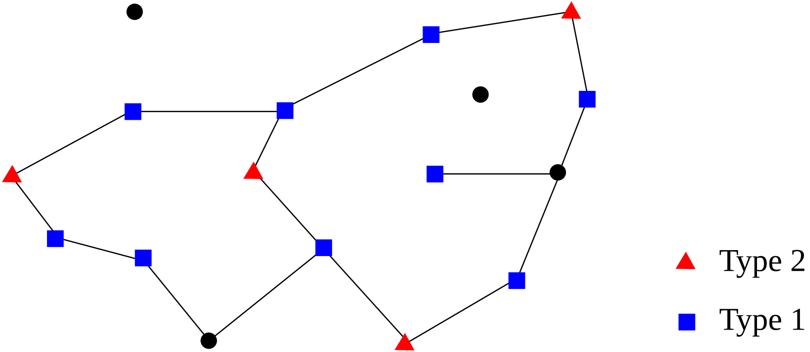
Données :

- Soit $G = (V, E)$ un graphe.
- Chaque arête $e \in E$ est munie d'un coût $c(e)$.
- Type de connexité :
  À chaque sommet $u \in V$ est associé un entier $r(u) \geq 0$,
  appelé *type de connexité*.
  $r(u)$ représente le nombre minimum d'arêtes qui doivent lier u au reste du réseau.

Exemple



Exemple



Problème de conception de réseaux fiables

Conditions d'arête-connexité :

- soient $u, v \in V$, il existe au moins $r(u,v) = min\{r(u), r(v)\}$ chaînes arête-disjointes entre u et v (i.e. n'ayant pas d'arêtes en commun).

Problème

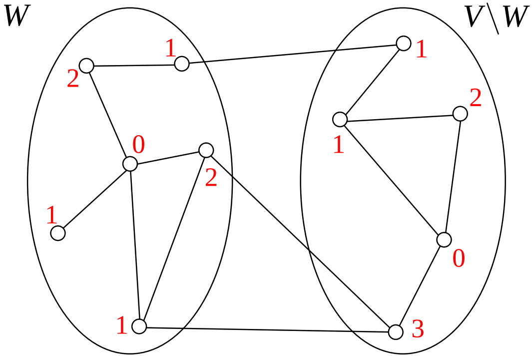
Le problème de conception de réseau fiable consiste à déterminer un sous-graphe $H = (V, F)$, $F \subset E$:

- vérifiant les conditions d'arête-connexité
- de coût minimum.

Les conditions d'arête-connexité assurent l'écoulement du trafic entre u et v s'il y a au max, $min\{r(u), r(v)\} - 1$ arêtes pouvant tomber en panne simultanément.

Notations

$$r(W) = max\{r(u) \mid u \in W\}, \quad \forall W \subset V$$
$$con(W) = min\{r(W), r(V \backslash W)\}, \quad \forall W \subset V, \emptyset \neq W \neq V$$



$$r(W) = 2$$
$$r(V \backslash W) = 3$$
$$con(W) = 2$$

Cas particuliers

$r(u) = 1$ pour tout $u \in V$

problème de l'arbre couvrant de poids minimum

$r(s) = r(t) = 1$ pour deux sommets $s, t \in V$

problème du plus court chemin (chaîne) entre s et t

$r(u) = 1$ pour $u \in S$, où S est un ensemble distingué de sommets
$r(u) = 0$ pour $u \in V \backslash S$

trouver un arbre contenant les sommets de S et éventuellement des sommets de $V \backslash S$ et qui soit de poids minimum
Ce problème est appelé *problème de l'arbre de Steiner*
Les sommets de type $r(u) = 1$ sont dits terminaux et les autres sont dits Steiner.

Outline

Cas où $r(u) = k$ pour tout $u \in V$

Données :

- Soit $G = (V, E)$ un graphe.
- Chaque arête $e \in E$ est munie d'un coût $c(e)$.

Problème kECSP

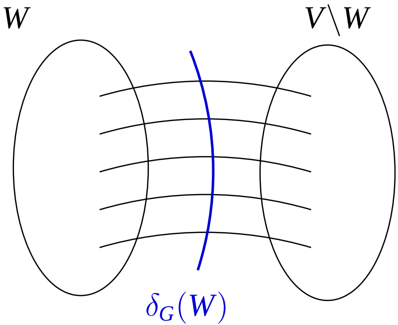
Le problème du sous-graphe k-arête connexe (kECSP, k Edge Connected Spanning Subgraph Problem) consiste à déterminer un sous-graphe $H = (V, F)$, $F \subset E$ tel que :

- entre chaque paire de sommets (u, v) de V, il existe k chaînes arête-disjointes,
- le coût soit minimum.

Les conditions d'arête-connexité assurent l'écoulement du trafic entre toutes paires de sommets, s'il y a au max $k - 1$ arêtes pouvant tomber en panne simultanément.

Coupe

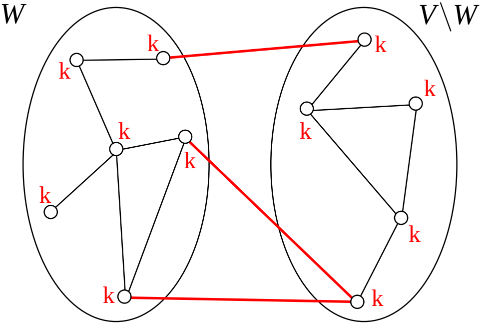
- Soit $G = (V, E)$ un graphe
  Soit $W \subset V$, $W \subset V, \emptyset \neq W \neq V$.
  Une coupe $\delta_G(W)$ est l'ensemble des arêtes dans G entre W et $V \backslash W$.



$$\delta_G(W) = \{uv \in E \,|\, u \in W, v \in V \backslash W\}, \quad W \subset V, \emptyset \neq W \neq V$$

kECSP

$r(W) = max\{r(u) — u \in W\}, \quad \forall W \subset V$
$con(W) = min\{r(W), r(V\backslash W)\}, \quad \forall W \subset V, \emptyset \neq W \neq V$



$r(W) = k$
$r(V\backslash W) = k$
$con(W) = k$

nombre d'arêtes de la coupe $\delta_H(W)$ doit être supérieur à $con(W) = k$
Théorème de Menger $\rightarrow$ cette condition suffit.

Problème kECSP : Formulation

- Soit $x \in \boldsymbol{R}^E$, $x(e) = \begin{cases} 1 \text{ si } e \text{ appartient à la solution,} \\ 0 \text{ sinon.} \end{cases}$

Formulation

Le problème du sous-graphe k-arête connexe (kECSP) peut être formulé comme suit :

$$\text{Minimiser} \sum_{e \in E} c(e)x(e)$$

$$\begin{aligned} x(\delta_G(W)) \geq k & \quad \forall\, W \subset V,\ \emptyset \neq W \neq V, \\ x(e) \in \{0, 1\} & \quad \forall\, e \in E. \end{aligned}$$

Cas où $k = 2$: Le problème du sous-graphe 2-arête connexe

- Soit $x \in \mathbb{R}^E$, $x(e) = \begin{cases} 1 \text{ si } e \text{ appartient à la solution,} \\ 0 \text{ sinon.} \end{cases}$

Formulation

Le problème du sous-graphe 2-arête connexe (2ECSP) peut être formulé comme suit :

$$\text{Minimiser } \sum_{e \in E} c(e)x(e)$$

$$\begin{aligned} & x(\delta_G(W)) \geq 2 \qquad \forall\, W \subset V,\ \emptyset \neq W \neq V, \\ & x(e) \in \{0, 1\} \qquad \forall\, e \in E. \end{aligned}$$

Problèmes

Problème 1

- Nombre de contraintes exponentiel
  - **Cutting plane ou génération de coupes**
    Idée :
    1. On prend un nombre de contraintes limité.
    2. On ajoute peu à peu les contraintes "utiles".

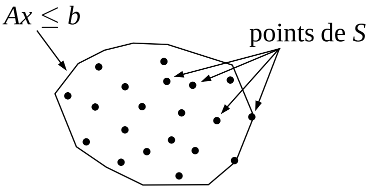
Problème 2

- Variables entières
  - **Branch & Bound**
    Idée :
    1. trouver une solution entière sans effectuer de branchement
    2. limiter le nombre de nœuds dans l'arbre du B&B

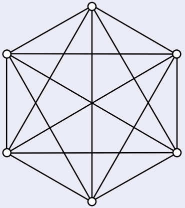
Outline

Nombre exponentiel de contraintes

Max $\sum c_i x_i$
sous les contraintes
$Ax \leq b$,
$x_i \in \{0,1\}, \quad i = 1, \ldots, n$



problème du sous-graphe 2-arête connexe

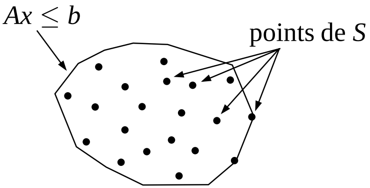


$$\text{Min} \sum_{e \in E} c(e)x(e)$$

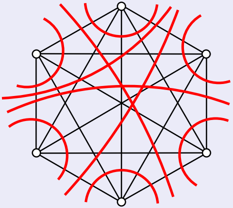
$$x(\delta_G(W)) \geq 2 \ \forall W \subset V, \ \emptyset \neq W \neq V,$$
$$x(e) \in \{0,1\} \quad \forall e \in E.$$

Nombre exponentiel de contraintes

Max $\sum c_i x_i$
sous les contraintes
$Ax \leq b$, <-**Nombre exponentiel**
$x_i \in \{0,1\}, \quad i = 1, \ldots, n$



problème du sous-graphe 2-arête connexe



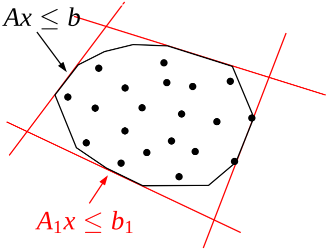
$$\text{Min} \sum_{e \in E} c(e)x(e)$$

$$x(\delta_G(W)) \geq 2 \;\forall\, W \subset V,\; \emptyset \neq W \neq V,$$
$$x(e) \in \{0,1\} \quad \forall\, e \in E.$$

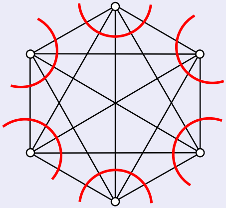
Cutting plane : contraintes initiales

1. On prend un nombre de contraintes limité.

Max $\sum c_i x_i$
sous les contraintes
$A_1 x \leq b_1,$
$x_i \in \{0,1\}, \quad i = 1, \ldots, n$



problème du sous-graphe 2-arête connexe



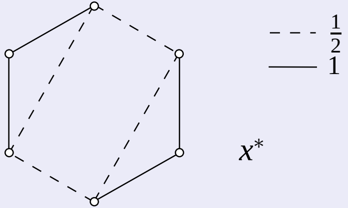
Minimiser $\sum_{e \in E} c(e)x(e)$

$$x(\delta_G(\{u\})) \geq 2 \qquad \forall u \in V,$$
$$0 \leq x(e) \leq 1 \qquad \forall e \in E,$$

Cutting plane : Contraintes utiles ou inutiles ?

1. On prend un nombre de contraintes limité.

problème du sous-graphe 2-arête connexe



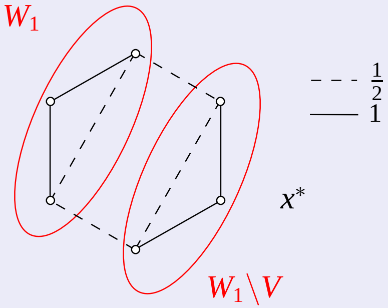
$$\text{Minimiser } \sum_{e \in E} c(e)x(e)$$

$$x(\delta_G(\{u\})) \geq 2 \qquad \forall u \in V,$$
$$0 \leq x(e) \leq 1 \qquad \forall e \in E,$$

Cutting plane : Contraintes utiles ou inutiles ?

1. On prend un nombre de contraintes limité.

problème du sous-graphe 2-arête connexe



Minimiser $\sum_{e \in E} c(e)x(e)$

$$
\begin{array}{ll}
x(\delta_G(\{u\})) \geq 2 & \forall u \in V, \\
0 \leq x(e) \leq 1 & \forall e \in E,
\end{array}
$$

2. On ajoute peu à peu les contraintes "utiles".

problème du sous-graphe 2-arête connexe

$x^*(\delta_G(W_1)) = \frac{3}{2}$

On ajoute la contrainte

$$x(\delta_G(W_1)) \geq 2$$

Séparation des contraintes

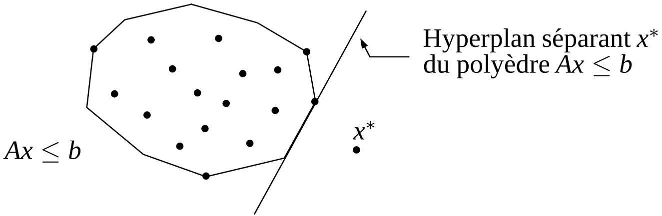
Problème de séparation

À un système linéaire $Ax \leq b$, on peut associer le problème suivant :
Pour une solution donnée x^*, vérifier

- si x^* satisfait $Ax \leq b$, et
- sinon déterminer une contrainte de $Ax \leq b$ qui est violée par x^*.

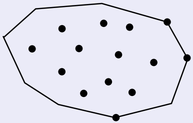
Ce problème est appelé *problème de séparation* associé à $Ax \leq b$.

Si x^* ne vérifie pas le système $Ax \leq b$
alors il existe un hyperplan séparant x^* et le polyèdre $Ax \leq b$.



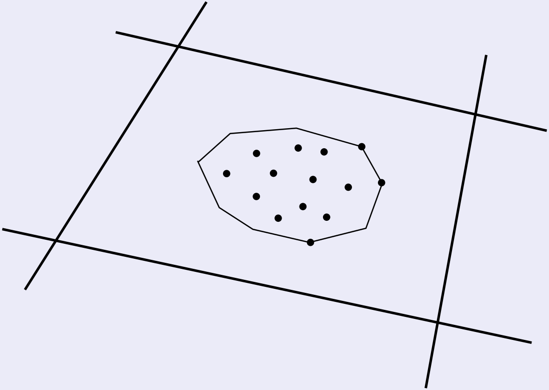
Algorithme de génération de coupes

Phase de coupes



Algorithme de génération de coupes

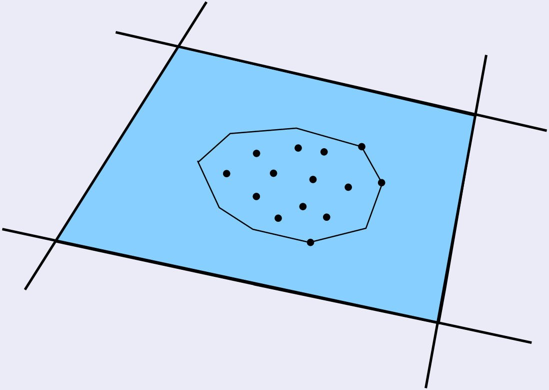
Phase de coupes



$A_1 x \leq b_1$

Algorithme de génération de coupes

Phase de coupes

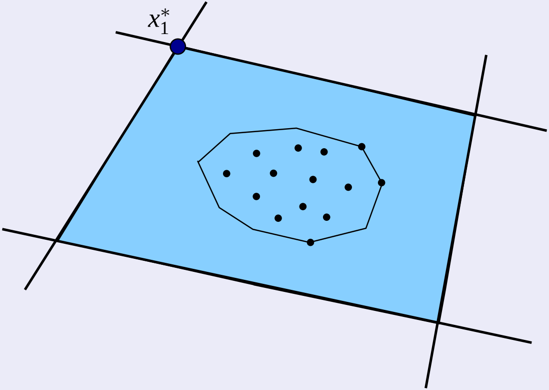


$A_1 x \leq b_1$

$P_1 = max\{cx, A_1 x \leq b_1\}$

Algorithme de génération de coupes

Phase de coupes

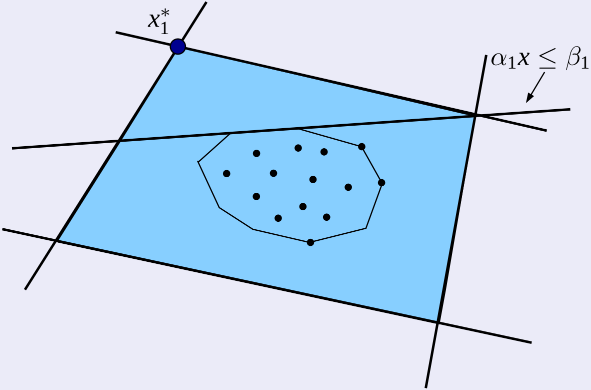


$A_1 x \leq b_1$

$P_1 = max\{cx, A_1 x \leq b_1\}$

Algorithme de génération de coupes

Phase de coupes

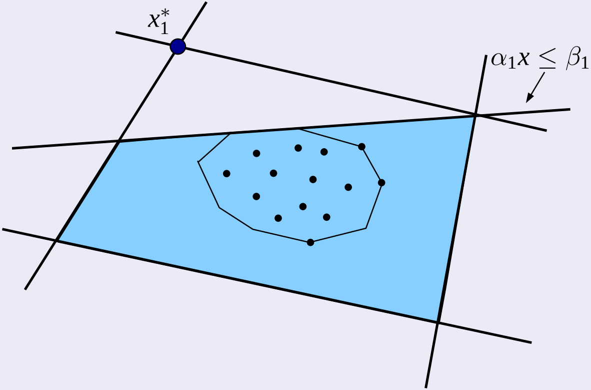


$A_1 x \leq b_1$
$\alpha_1 x \leq \beta_1$

$P_2 = max\{cx, A_1 x \leq b_1, \alpha_1 x \leq \beta_1\}$

Algorithme de génération de coupes

Phase de coupes

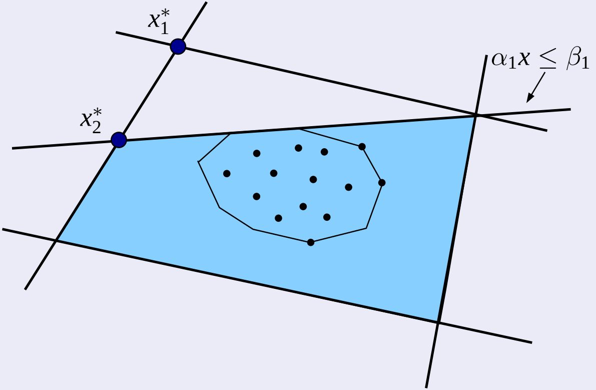


$A_1 x \leq b_1$
$\alpha_1 x \leq \beta_1$

$P_2 = max\{cx, A_1 x \leq b_1, \alpha_1 x \leq \beta_1\}$

Algorithme de génération de coupes

Phase de coupes

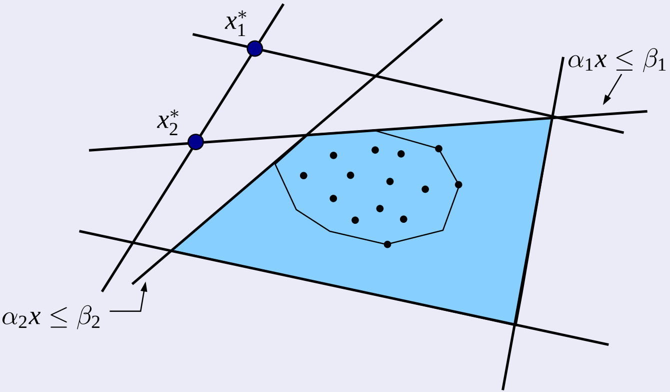


$A_1 x \leq b_1$
$\alpha_1 x \leq \beta_1$

$P_1 = max\{cx, A_1 x \leq b_1, \alpha_1 x \leq \beta_1\}$

Algorithme de génération de coupes

Phase de coupes

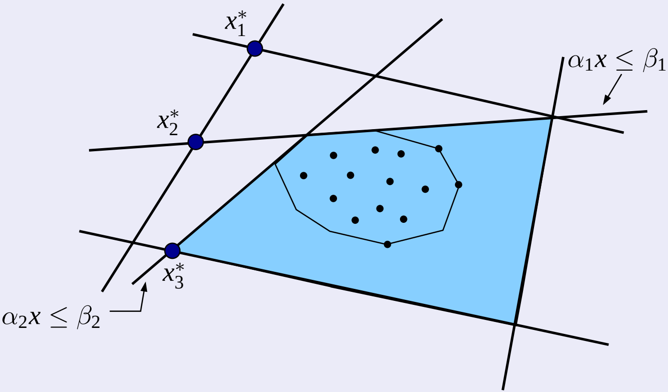


$$A_1 x \leq b_1$$
$$\alpha_1 x \leq \beta_1$$
$$\alpha_2 x \leq \beta_2$$

$$P_3 = max\{cx, A_1 x \leq b_1, \alpha_1 x \leq \beta_1, \alpha_2 x \leq \beta_2\}$$

Algorithme de génération de coupes

Phase de coupes

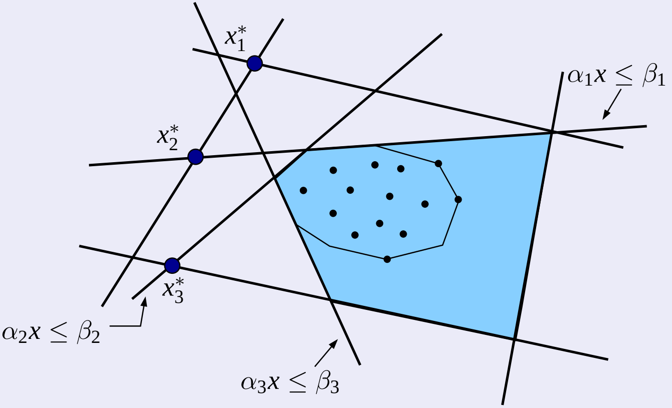


$$A_1 x \leq b_1$$
$$\alpha_1 x \leq \beta_1$$
$$\alpha_2 x \leq \beta_2$$

$$P_3 = max\{cx, A_1 x \leq b_1, \alpha_1 x \leq \beta_1, \alpha_2 x \leq \beta_2\}$$

Algorithme de génération de coupes

Phase de coupes

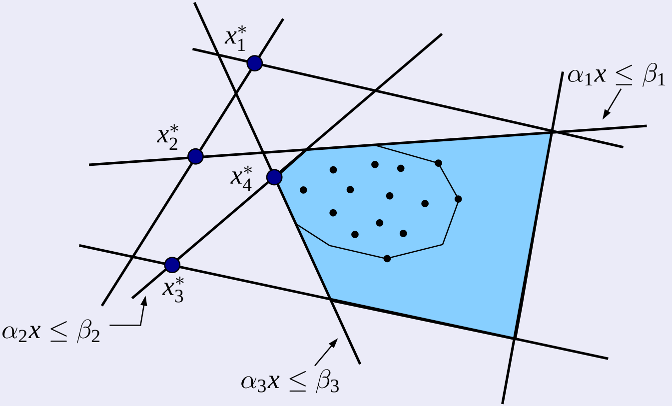


$A_1 x \leq b_1$
$\alpha_1 x \leq \beta_1$
$\alpha_2 x \leq \beta_2$
$\alpha_3 x \leq \beta_3$

$$P_4 = max\{cx, A_1 x \leq b_1, \alpha_1 x \leq \beta_1, \alpha_2 x \leq \beta_2, \alpha_3 \leq \beta_3\}$$

Algorithme de génération de coupes

Phase de coupes



$$\begin{aligned} A_1 x &\leq b_1 \\ \alpha_1 x &\leq \beta_1 \\ \alpha_2 x &\leq \beta_2 \\ \alpha_3 x &\leq \beta_3 \end{aligned}$$

$$P_4 = max\{cx, A_1 x \leq b_1, \alpha_1 x \leq \beta_1, \alpha_2 x \leq \beta_2, \alpha_3 \leq \beta_3\}$$

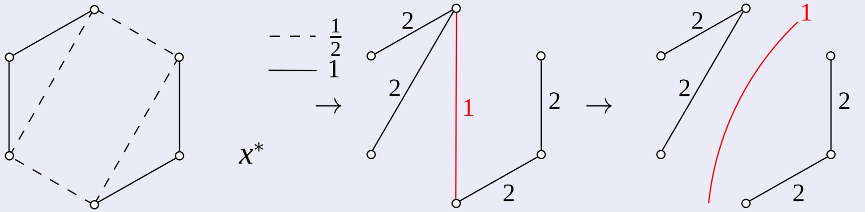
Problème de séparation

- Chaque classe de contraintes a son propre problème de séparation.
- La séparation des contraintes utilisées dans la formulation du problème doit pouvoir être effectuée de manière exacte pour assurer la réalisabilité de la solution.

Séparation des contraintes de coupe

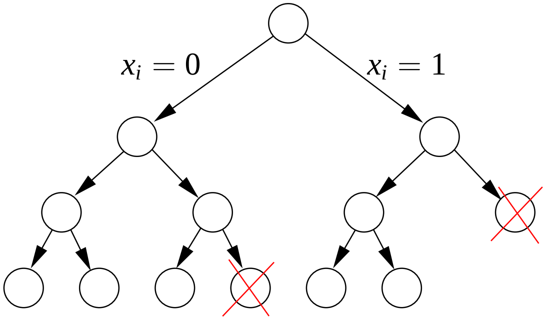
1. On considère le graphe $G = (V, E)$ muni de capacités données par la valeur de x^*.
2. On utilise un algorithme de flot max pour trouver les coupes min.
3. L'arbre de Gomory-Hu est ainsi créé.
4. Toutes les arêtes de cet arbre ayant un poids < 2 donnent une contrainte de coupes violée.

problème du sous-graphe 2-arête connexe

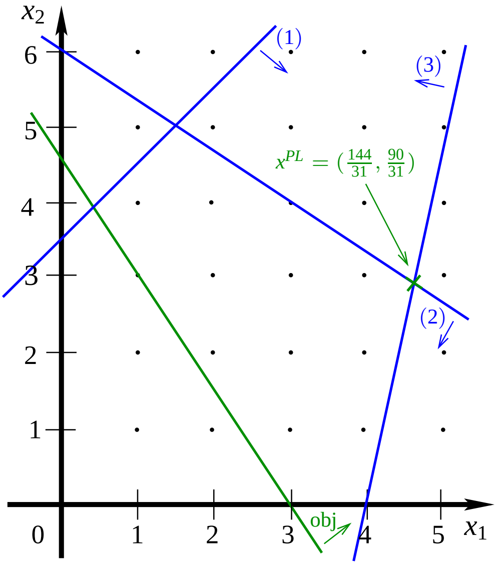


Outline

Algorithme de Branch & Bound



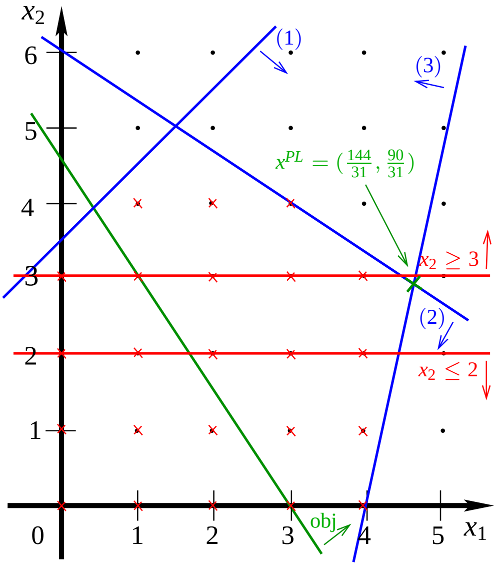
Algorithme de Branch & Bound



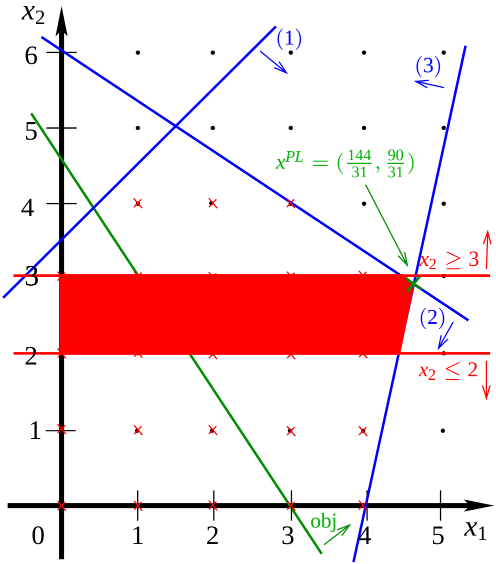
Algorithme de Branch & Bound

$$\textcircled{S} \quad \begin{array}{l} x_{PL} = (4 + \frac{20}{31}, 2 + \frac{28}{31}) \\ z = 19 + \frac{23}{31} \end{array}$$

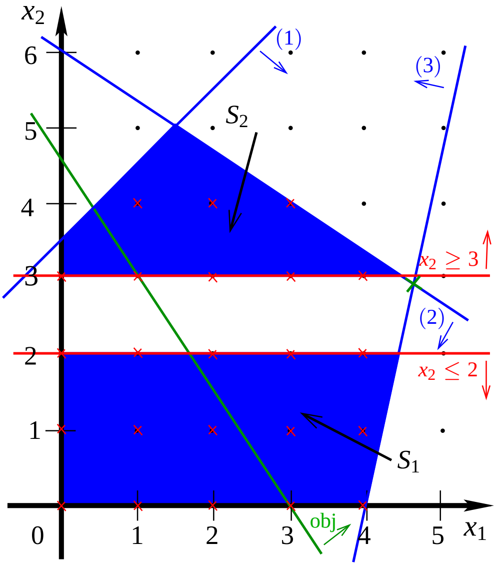
Algorithme de Branch & Bound



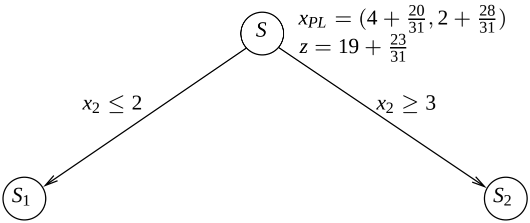
Algorithme de Branch & Bound



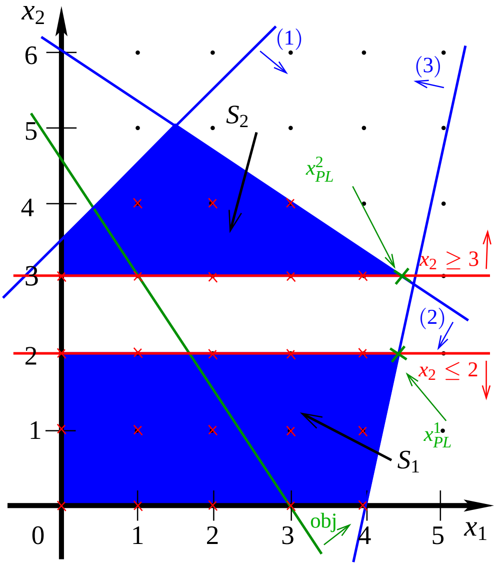
Algorithme de Branch & Bound



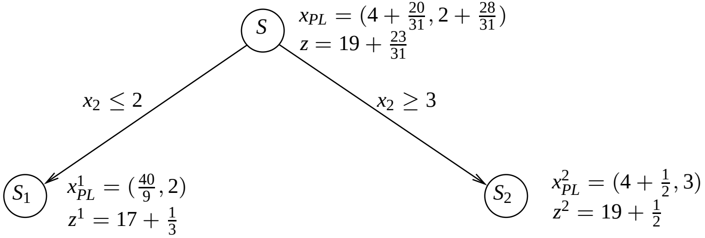
Algorithme de Branch & Bound



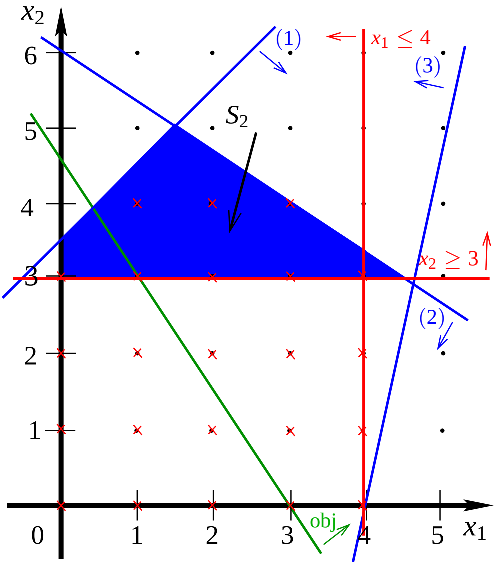
Algorithme de Branch & Bound



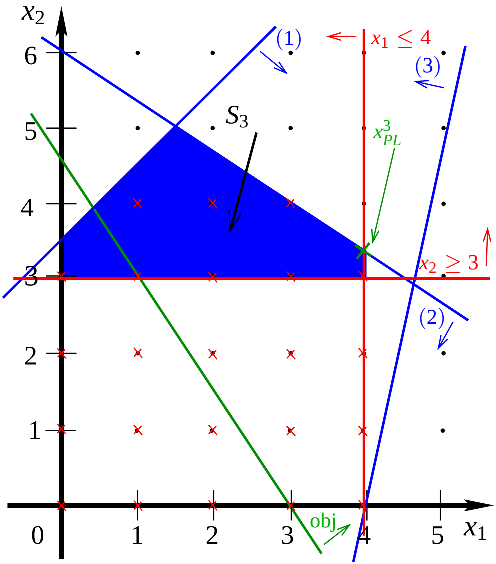
Algorithme de Branch & Bound



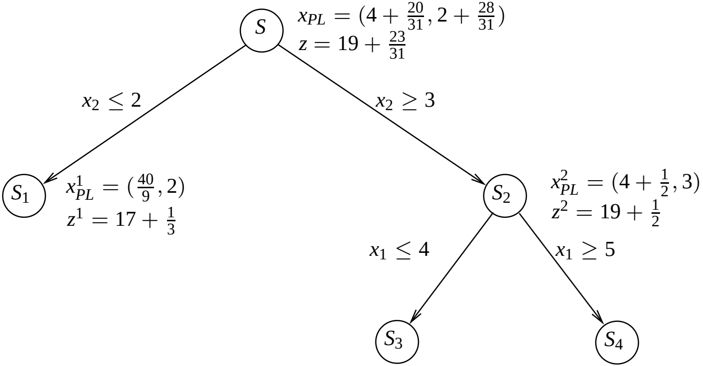
Algorithme de Branch & Bound



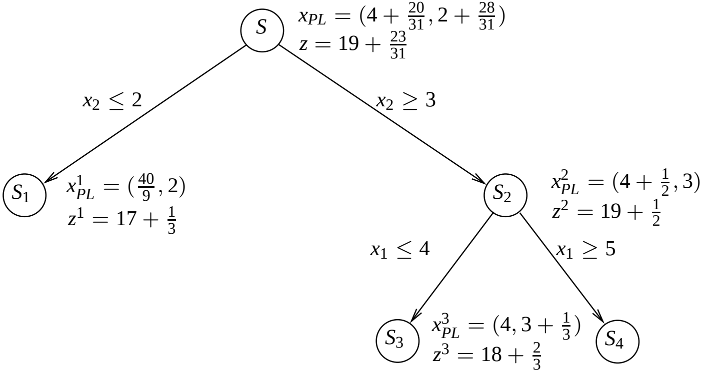
Algorithme de Branch & Bound



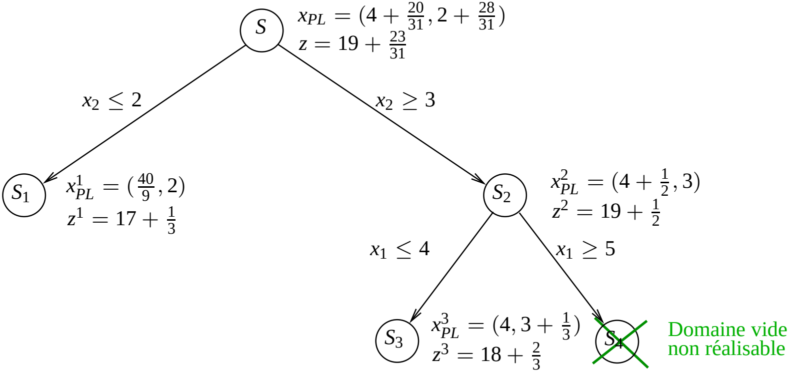
Algorithme de Branch & Bound



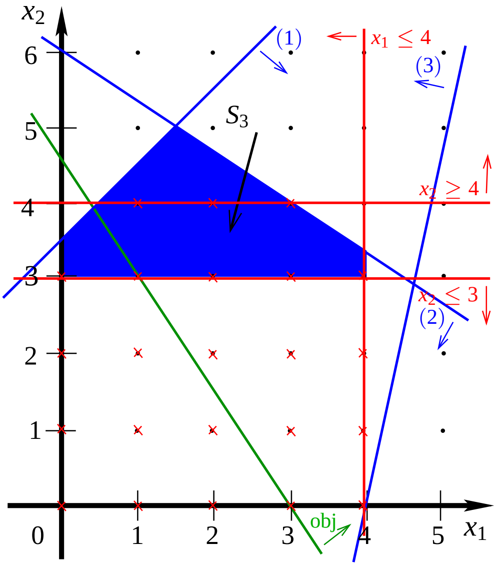
Algorithme de Branch & Bound



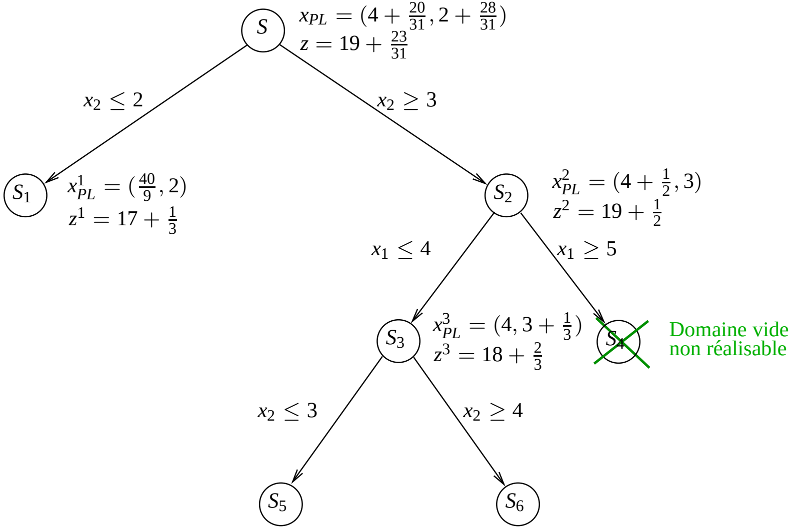
Algorithme de Branch & Bound



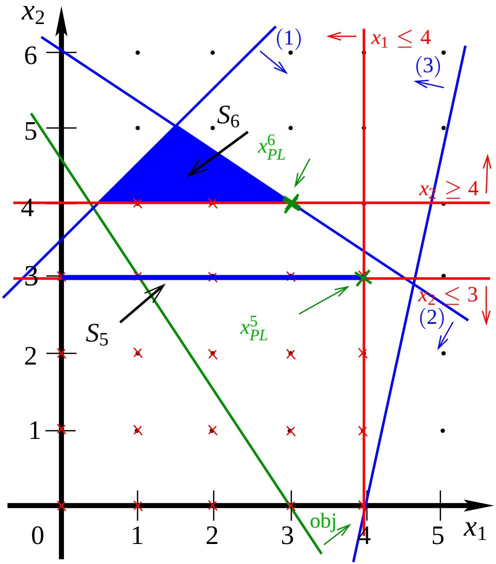
Algorithme de Branch & Bound



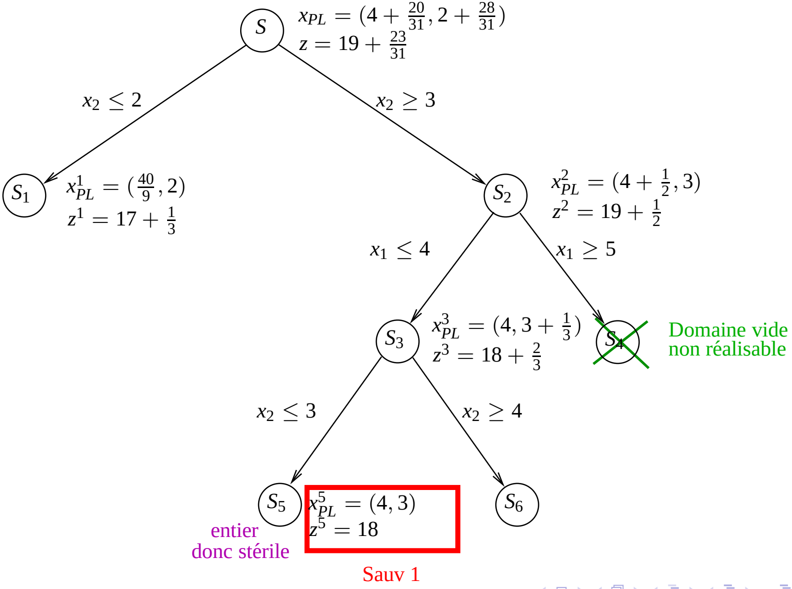
Algorithme de Branch & Bound



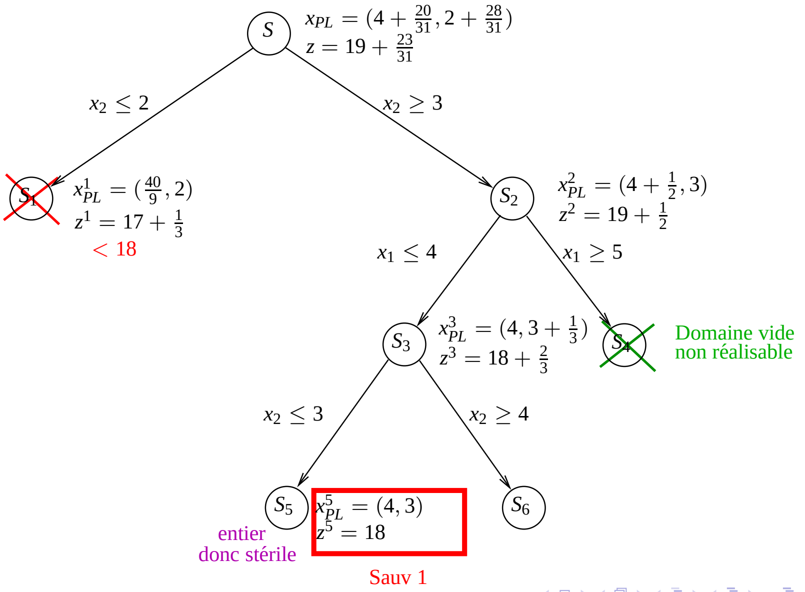
Algorithme de Branch & Bound



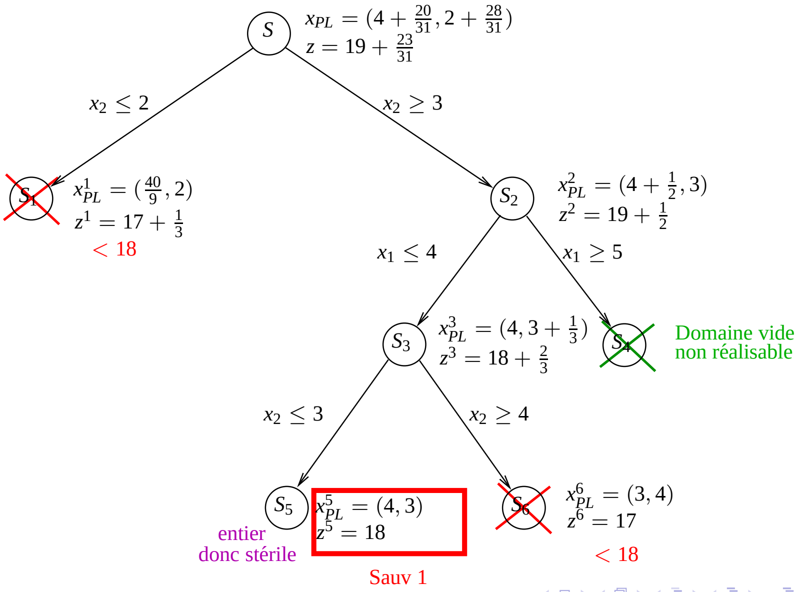
Algorithme de Branch & Bound



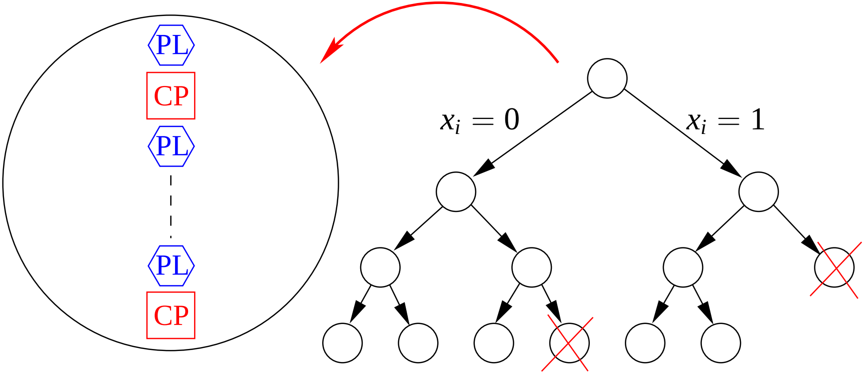
Algorithme de Branch & Bound



Algorithme de Branch & Bound



Algorithme de Branch & Cut

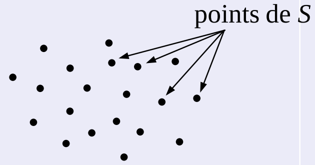


Enveloppe convexe

Principe et idées

Ramener le problème à la résolution d'un programme linéaire continu.

Max $\sum c_i x_i$
sous les contraintes
$Ax \leq b,$
$x_i \in \{0,1\}, \quad i = 1, \ldots, n$

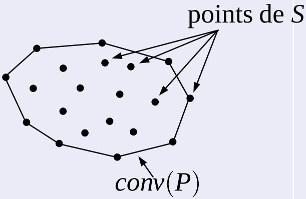


Enveloppe convexe

Principe et idées

Ramener le problème à la résolution d'un programme linéaire continu.

Max $\sum c_i x_i$
sous les contraintes
$Ax \leq b$,
$x_i \in \{0, 1\}, \quad i = 1, \ldots, n$

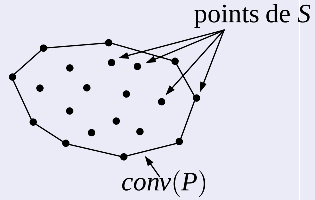


Enveloppe convexe

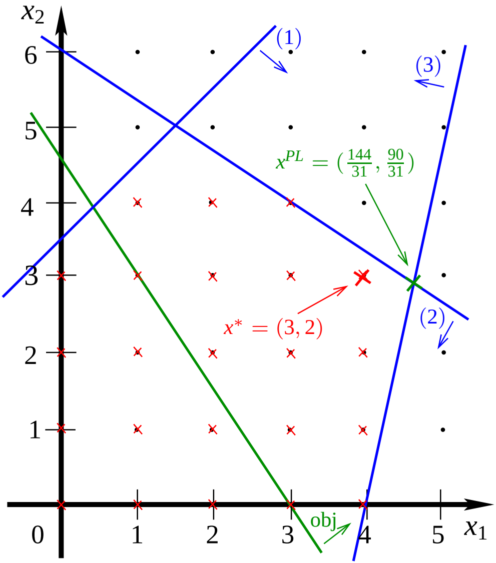
Principe et idées

Ramener le problème à la résolution d'un programme linéaire continu.

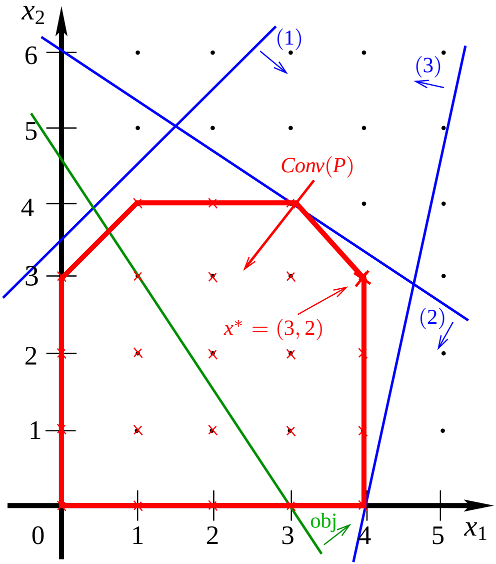
Max $\sum c_i x_i$
sous les contraintes
$Ax \leq b,$
$0 \leq x_i \leq 1, \quad i = 1, \ldots, n$
$\sum \alpha_{ij} x_i \leq \beta_j, \quad j = 1, \ldots, k$



Un exemple d'enveloppe convexe



Un exemple d'enveloppe convexe



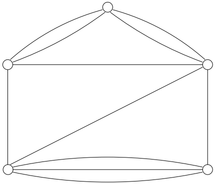
Description complète de l'enveloppe convexe

Le problème 2ECSP est NP-difficile.

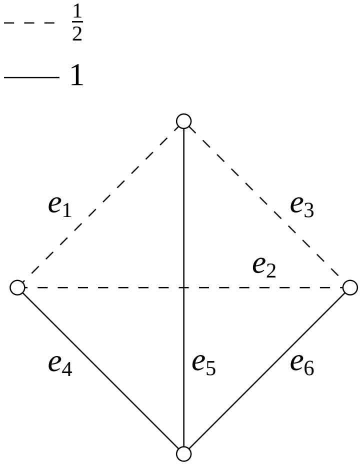
Très lié au problème du voyageur de commerce.
$\rightarrow$ **aucune chance de trouver une description complète de l'enveloppe convexe des solutions.**
Mais on peut trouver une telle description si on restreint le problème à différentes classes de graphes.

Exemple :
Mahjoub (1994) a montré que les contraintes de coupes et les contraintes triviales décrivent entièrement le polytope enveloppe convexe des solutions dans le cas des graphes série-paralleles.

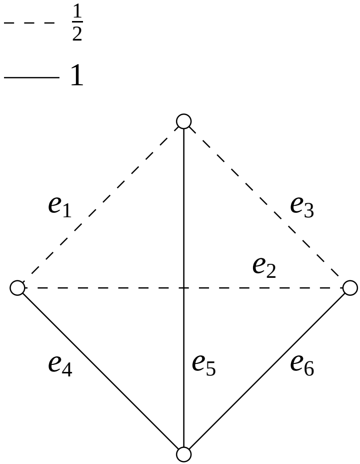
Un graphe série-parallèle est construit à partir d'une arête en appliquant de manière récursive les deux opérations suivantes : dupliquer ou subdiviser une arête.



Problème 2ECSP : point fractionnaire

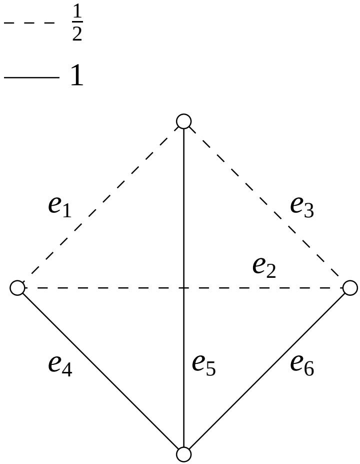


Problème 2ECSP : point fractionnaire



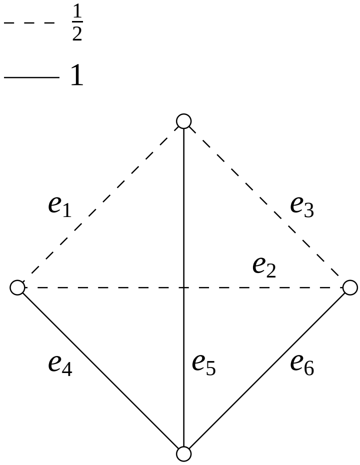
$$x(e_1) + x(e_2) + x(e_4) = 2$$

Problème 2ECSP : point fractionnaire



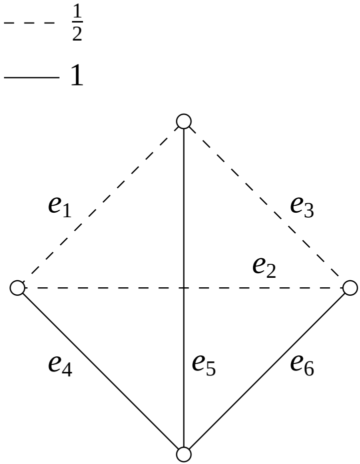
$$x(e_1) + x(e_2) + x(e_4) = 2$$
$$x(e_1) + x(e_3) + x(e_5) = 2$$
$$x(e_3) + x(e_2) + x(e_6) = 2$$

Problème 2ECSP : point fractionnaire



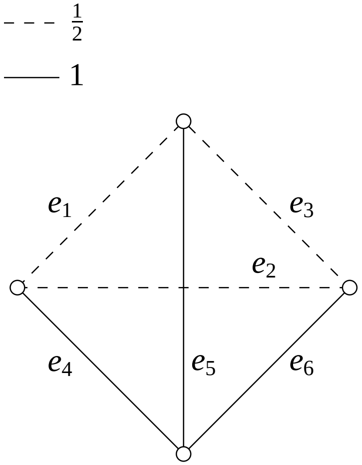
$$x(e_1) + x(e_2) + x(e_4) = 2$$
$$x(e_1) + x(e_3) + x(e_5) = 2$$
$$x(e_3) + x(e_2) + x(e_6) = 2$$
$$x(e_4) = 1$$
$$x(e_5) = 1$$
$$x(e_6) = 1$$

Problème 2ECSP : point fractionnaire



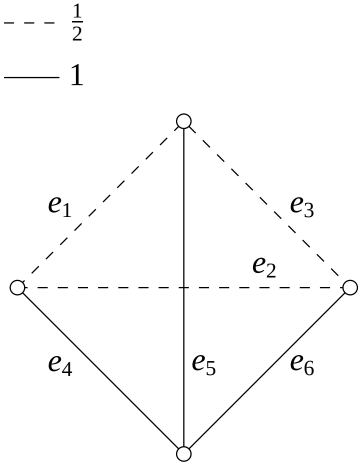
$$x(e_1) + x(e_2) + x(e_4) \geq 2$$
$$x(e_1) + x(e_3) + x(e_5) \geq 2$$
$$x(e_3) + x(e_2) + x(e_6) \geq 2$$

Problème 2ECSP : point fractionnaire



$$
\begin{aligned}
&x(e_1) + x(e_2) + x(e_4) \geq 2 \\
&x(e_1) + x(e_3) + x(e_5) \geq 2 \\
&x(e_3) + x(e_2) + x(e_6) \geq 2 \\
&-x(e_4) \geq -1 \\
&-x(e_5) \geq -1 \\
&-x(e_6) \geq -1
\end{aligned}
$$

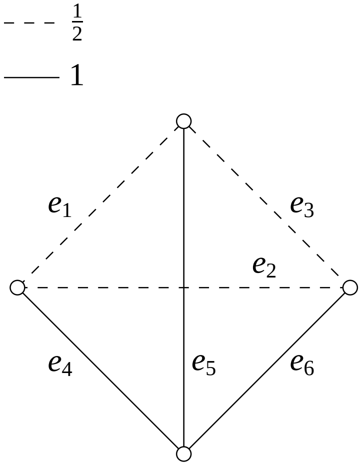
Problème 2ECSP : point fractionnaire



$$
\begin{aligned}
x(e_1) + x(e_2) + x(e_4) &\geq 2\\
x(e_1) + x(e_3) + x(e_5) &\geq 2\\
x(e_3) + x(e_2) + x(e_6) &\geq 2\\
-x(e_4) &\geq -1\\
-x(e_5) &\geq -1\\
-x(e_6) &\geq -1
\end{aligned}
$$

$$2\sum_{i=1}^{3} x(e_i) \geq 3$$

Problème 2ECSP : point fractionnaire

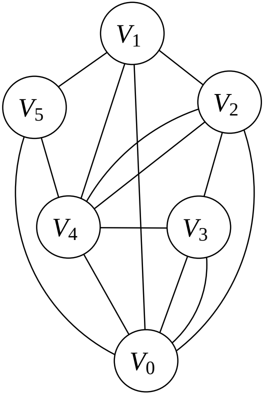


$$
\begin{aligned}
x(e_1) + x(e_2) + x(e_4) &\geq 2 \\
x(e_1) + x(e_3) + x(e_5) &\geq 2 \\
x(e_3) + x(e_2) + x(e_6) &\geq 2 \\
-x(e_4) &\geq -1 \\
-x(e_5) &\geq -1 \\
-x(e_6) &\geq -1
\end{aligned}
$$

$$2 \sum_{i=1}^{3} x(e_i) \geq 3$$

$$\Longrightarrow \sum_{i=1}^{3} x(e_i) \geq \left\lceil \frac{3}{2} \right\rceil = 2$$

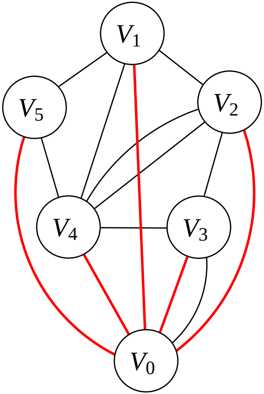
Problème 2-ECSP : Contraintes de F-partition



- Soit $G = V, E)$ un graphe.
- Soit $V_0, V_1, V_2, \ldots, V_p$ une partition de V,
- Soit $F \subset \delta_G(V_0)$ un ensemble d'arêtes contenu dans $\delta_G(V_0)$ et $|F|$ est impair.
- $x(\delta(V_i)) \geq 2 \;\forall i = 1, \ldots, p$
  $-x(e) \geq -1 \;\forall e \in F$
  $x(e) \geq 0 \;\forall e \in \delta(V_0) \backslash F$
- En sommant : $\displaystyle\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - \frac{|F|}{2}$
- $|F|$ impair ($|F| = 2q + 1$), $\displaystyle\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q - \frac{1}{2}$
- $\displaystyle\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q$ valide pour le problème 2ECSP.

Problème 2-ECSP : Contraintes de F-partition

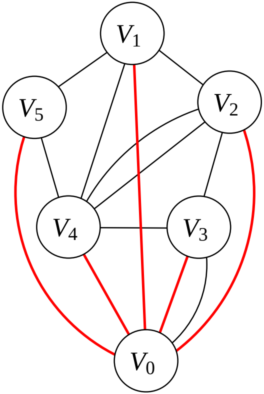
- Soit $G = V, E)$ un graphe.
- Soit $V_0, V_1, V_2, \ldots, V_p$ une partition de V,
- Soit $F \subset \delta_G(V_0)$ un ensemble d'arêtes contenu dans $\delta_G(V_0)$ et $|F|$ est impair.



- $x(\delta(V_i)) \geq 2 \;\forall i = 1, \ldots, p$
  $-x(e) \geq -1 \;\forall e \in F$
  $x(e) \geq 0 \;\forall e \in \delta(V_0) \backslash F$
- En sommant : $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - \frac{|F|}{2}$$
- $|F|$ impair ($|F| = 2q + 1$), $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q - \frac{1}{2}$$
- $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q$$ valide pour le problème 2ECSP.

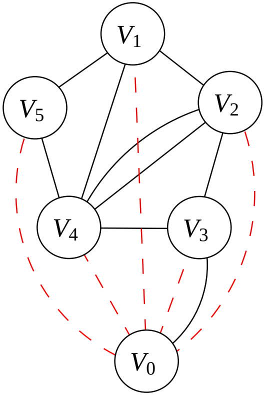
Problème 2-ECSP : Contraintes de F-partition

- Soit $G = V, E)$ un graphe.
- Soit $V_0, V_1, V_2, \ldots, V_p$ une partition de V,
- Soit $F \subset \delta_G(V_0)$ un ensemble d'arêtes contenu dans $\delta_G(V_0)$ et $|F|$ est impair.
- $x(\delta(V_i)) \geq 2 \ \forall i = 1, \ldots, p$
  $-x(e) \geq -1 \ \forall e \in F$
  $x(e) \geq 0 \ \forall e \in \delta(V_0) \backslash F$



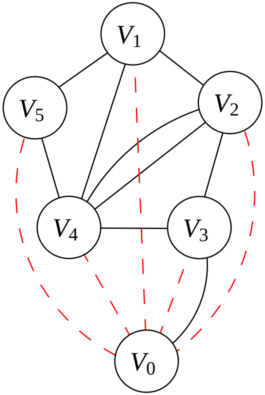
- En sommant : $\displaystyle\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - \frac{|F|}{2}$
- $|F|$ impair ($|F| = 2q + 1$), $\displaystyle\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q - \frac{1}{2}$
- $\displaystyle\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q$ valide pour le problème 2ECSP.

Problème 2-ECSP : Contraintes de F-partition



- Soit $G = V, E)$ un graphe.
- Soit $V_0, V_1, V_2, \ldots, V_p$ une partition de V,
- Soit $F \subset \delta_G(V_0)$ un ensemble d'arêtes contenu dans $\delta_G(V_0)$ et $|F|$ est impair.
- $x(\delta(V_i)) \geq 2 \ \forall i = 1, \ldots, p$
  $-x(e) \geq -1 \ \forall e \in F$
  $x(e) \geq 0 \ \forall e \in \delta(V_0) \backslash F$
- En sommant : $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - \frac{|F|}{2}$$
- $|F|$ impair ($|F| = 2q + 1$), $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q - \frac{1}{2}$$
- $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q$$ valide pour le problème 2ECSP.

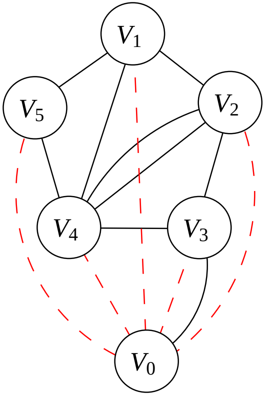
Problème 2-ECSP : Contraintes de F-partition



- Soit $G = V, E)$ un graphe.
- Soit $V_0, V_1, V_2, \ldots, V_p$ une partition de V,
- Soit $F \subset \delta_G(V_0)$ un ensemble d'arêtes contenu dans $\delta_G(V_0)$ et $|F|$ est impair.
- $x(\delta(V_i)) \geq 2 \ \forall i = 1, \ldots, p$
  $-x(e) \geq -1 \ \forall e \in F$
  $x(e) \geq 0 \ \forall e \in \delta(V_0) \backslash F$
- En sommant : $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - \frac{|F|}{2}$$
- $|F|$ impair ($|F| = 2q + 1$), $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q - \frac{1}{2}$$
- $$\sum_{e \in \delta(V_0, \ldots, V_p) \backslash F} x(e) \geq p - q$$ valide pour le problème 2ECSP.

Problème 2-ECSP : Contraintes de F-partition

- Soit $G = V, E)$ un graphe.
- Soit $V_0, V_1, V_2, \dots, V_p$ une partition de V,
- Soit $F \subset \delta_G(V_0)$ un ensemble d'arêtes contenu dans $\delta_G(V_0)$ et $|F|$ est impair.
- $x(\delta(V_i)) \geq 2 \;\forall i = 1, \dots, p$
  $-x(e) \geq -1 \;\forall e \in F$
  $x(e) \geq 0 \;\forall e \in \delta(V_0) \backslash F$



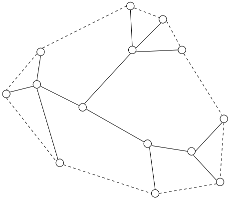
- En sommant : $$\sum_{e \in \delta(V_0, \dots, V_p) \backslash F} x(e) \geq p - \frac{|F|}{2}$$
- $|F|$ impair ($|F| = 2q + 1$), $$\sum_{e \in \delta(V_0, \dots, V_p) \backslash F} x(e) \geq p - q - \frac{1}{2}$$
- $$\sum_{e \in \delta(V_0, \dots, V_p) \backslash F} x(e) \geq p - q$$ valide pour le problème 2ECSP.

Problème 2-ECSP : Contraintes de F-partition

Contraintes de F-partition

- Introduites par Mahjoub (1994).
- Problème de séparation NP-complet.
- Peuvent être généralisées au problème général quand $r(u)$ est quelconque.
- Barahona et Mahjoub (1996) ont montré que les contraintes de coupe, les contraintes triviales et les contraintes de F-partition suffisent pour donner une desription complète du polytope des solutions du problème 2ECSP dans les graphes de Halin.

Un graphe de Halin est un graphe $G = (V, T \cup C)$ où T est un arbre sans sommet de degré 2 et C est un cycle dont les sommets sont les sommets pendants de T.



Approche polyédrale

Beaucoup de questions encore ...

- Dimension du polytope enveloppe convexe des solutions du problème ?
- Description complète du polytope ?
- Une contrainte est-elle ou non une facette du polytope ?
- Qualité des algorithmes de séparation ?
- Existence d'opérations de réductions conservant les contraintes violées ?
- ...

pour une prochaine fois ... peut-être !