

# HAI820I Recherche Opérationnelle II — TDs

Ivan Lejeune

12 février 2026

## Table des matières

|     |   |    |
|-----|---|----|
| 1   | Programmation linéaire en nombres entiers . . . . . | 2  |
| 1.1 | Matrice totalement unimodulaire . . . . .           | 2  |
| 1.2 | Arbre de branchement et exemples . . . . .          | 3  |
| 1.3 | Résolution effective . . . . .                      | 6  |
| 1.4 | Inégalités valides . . . . .                        | 12 |
| 1.5 | Coupes . . . . .                                    | 15 |
| 1.6 | Enveloppe convexe. . . . .                          | 15 |
| 2   | Génération de colonnes. . . . .                     | 18 |
| 3   | Lagrangien . . . . .                                | 19 |
| 4   | Modélisation . . . . .                              | 20 |

# 1 Programmation linéaire en nombres entiers

## 1.1 Matrice totalement unimodulaire

### Exercice 1 Matrice totalement unimodulaire.

Est-ce que les matrices suivantes sont totalement unimodulaires ?

1. La matrice  $A$  suivante :

$$A = \begin{pmatrix} 1 & -1 & 1 \\ -1 & 1 & 0 \end{pmatrix}$$

2. La matrice  $B$  suivante :

$$B = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ -1 & 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 & 0 \end{pmatrix}$$

3. La matrice  $C$  suivante :

$$C = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$$

4. La matrice  $D$  suivante :

$$D = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

5. Propriétés : est-ce que les matrices suivantes sont totalement unimodulaires sachant que  $A$  est totalement unimodulaire ?

- (a) La matrice  $-A$ ,
- (b) la transposée  $A^T$ ,
- (c) la matrice  $[A, I]$ ,
- (d) la matrice  $[A, -A]$ .

6. Considérons la matrice  $E$  définie de la manière suivante :

$$E = \begin{pmatrix} 1 & -1 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

et soit  $b = \begin{pmatrix} 2 & 1 \end{pmatrix}$ .

Est-ce que  $E$  est totalement unimodulaire ? Trouver les deux solutions à valeurs entières au problème  $EX = B$ .

### Solution.

1. La matrice  $A$  est totalement unimodulaire car toutes ses matrices ont déterminant 1, -1 ou 0.

$$\begin{vmatrix} 1 & -1 \\ -1 & 1 \end{vmatrix} = 0, \quad \begin{vmatrix} 1 & 1 \\ -1 & 0 \end{vmatrix} = 1, \quad \begin{vmatrix} -1 & 1 \\ 1 & 0 \end{vmatrix} = -1$$

2. On peut regarder le critère de suffisance pour qu'une matrice soit TU. On met les lignes 1 et 3 ensemble dans  $A$  et les deux autres dans  $B$ . Alors on a :

$$A = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & -1 \end{pmatrix} B = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{pmatrix}$$

On a bien respecté les critères, les valeurs de signes opposés dont dans le même sous-ensemble et les valeurs de signe identique sont bien dans les sous-ensembles opposés. Alors  $B$  est totalement unimodulaire.

3. La matrice  $C$  n'est pas TU car son déterminant est 2.

4. La matrice  $D$  n'est pas TU car son déterminant est 2.
5. Si  $A$  est une matrice totalement unimodulaire alors :
  - (a) la matrice  $-A$  est aussi TU,
  - (b) la matrice  $A^T$  est TU,
  - (c) en développant le long de la diagonale de  $I$  on retrouve bien que  $[A, I]$  est TU,
  - (d) par les propositions ci-dessus on a bien  $[A, -A]$  qui est TU.
6. La matrice  $E$  n'est pas TU. Le problème s'exprime comme :

$$PL = \begin{cases} x_1 - x_2 + x_3 = 2 \\ x_1 + x_2 = 1 \end{cases}$$

Les solutions sont  $X_1 = (1, 0, 1)$  et  $X_2 = (0, 1, 3)$ .

## 1.2 Arbre de branchement et exemples

### Exercice 2 Solutions entières versus solutions réelles.

Donner les solutions réelles et entières des problèmes suivants.

1. Un premier LP :

$$PL_1 = \begin{cases} 2x_1 - 2x_2 \geq 1 \\ x_1 + x_2 \geq 1 \\ x_2 \leq 2 \\ x_1, x_2 \in \mathbb{R}^+ \\ \min(z) = 3x_1 - x_2 \end{cases}$$

2. Un second LP :

$$PL_2 = \begin{cases} 10x_1 + 7x_2 \leq 29 \\ x_1, x_2 \in \mathbb{R}^+ \\ \max(z) = 300x_1 + 205x_2 \end{cases}$$

3. Un troisième LP :

$$PL_3 = \begin{cases} -3x_1 + 4x_2 \geq 6 \\ 2x_1 + 2x_2 \geq 11 \\ 4x_1 - 5x_2 \leq 10 \\ x_1, x_2 \in \mathbb{R}^+ \\ \min(z) = -3x_1 - 2x_2 \end{cases}$$

### Solution.

1. La meilleure solution réelle ici est  $x_{\mathbb{R}}^* = (0.75, 0.25)$  avec  $z_{\mathbb{R}}^* = 2$ .

La solution entière optimale est  $x_{\mathbb{Z}}^* = (1, 0)$  avec  $z_{\mathbb{Z}}^* = 3$ .

On peut visualiser ces résultats comme suit :

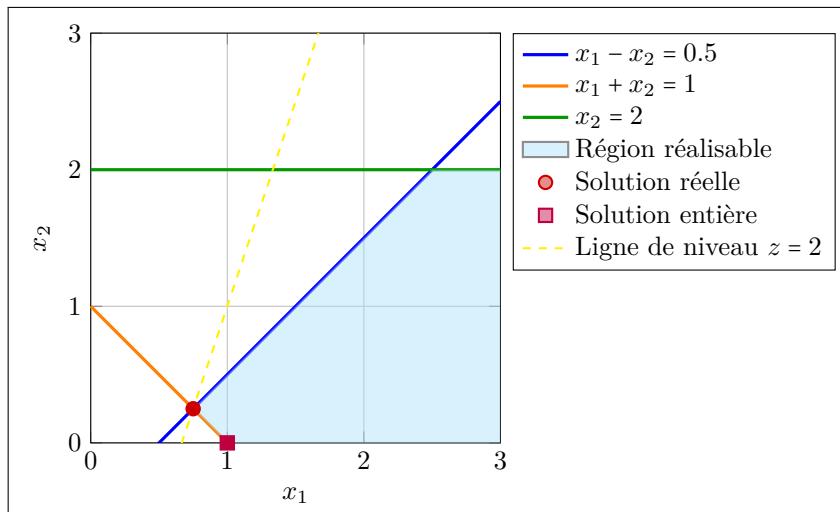


FIGURE 1 – Représentation du problème  $PL_1$  dans  $\mathbb{R}^2$

2. La meilleure solution réelle ici est  $x_{\mathbb{R}}^* = (2.9, 0)$  avec  $z_{\mathbb{R}}^* = 870$ .

La solution entière optimale est  $x_{\mathbb{Z}}^* = (0, 4)$  avec  $z_{\mathbb{Z}}^* = 820$ .

On peut visualiser ces résultats comme suit :

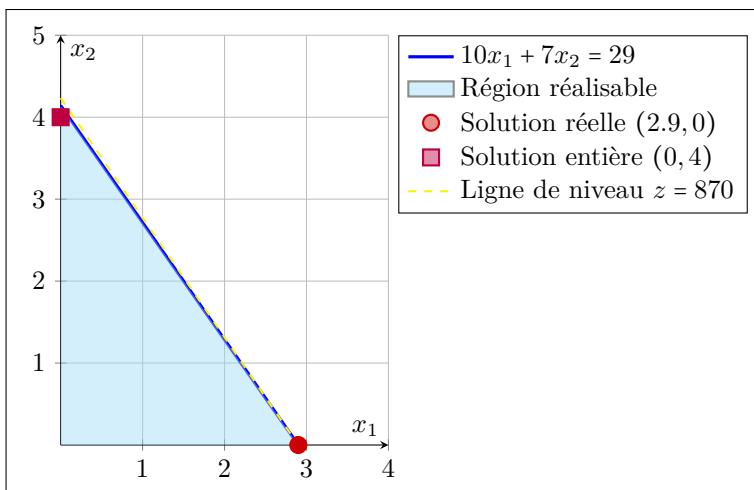


FIGURE 2 – Représentation du problème  $PL_2$  dans  $\mathbb{R}^2$

3. Le dernier problème est non borné.

### Exercice 3 Détermination de l'arbre d'évaluation et de séparation pour un problème donné.

Nous considérons le programme linéaire en nombres entiers suivant :

$$PL_1 = \begin{cases} 2x_1 + 5x_2 \leq 11 \\ x_1, x_2 \in \mathbb{N} \\ \max(z) = 10x_1 + x_2 \end{cases}$$

Nous associons le programme linéaire (en valeurs réelles)  $PL_1$  et la solution est la suivante :

$$x_1^* = 5.5, \quad x_2^* = 0, \quad z^* = 55.$$

Le problème se sépare en deux sous-problèmes :

$$PL_2 = \begin{cases} 2x_1 + 5x_2 \leq 11 \\ x_1 \leq 5 \\ x_1, x_2 \in \mathbb{N} \\ \max(z) = 10x_1 + x_2 \end{cases}$$

et

$$PL_3 = \begin{cases} 2x_1 + 5x_2 \leq 11 \\ x_1 \geq 6 \\ x_1, x_2 \in \mathbb{N} \\ \max(z) = 10x_1 + x_2 \end{cases}$$

Les solutions sont les suivantes pour les deux programmes :

$$\begin{aligned} x_1^* &= 5, & x_2^* &= 0.2, & z^* &= 50.2 \text{ pour } PL_2 \\ &&&&& \emptyset \text{ pour } PL_3 \end{aligned}$$

Ainsi le problème  $PL_2$  associé au programme  $PL_1$  se décompose en sous-problèmes :

$$PL_4 = \begin{cases} 2x_1 + 5x_2 \leq 11 \\ x_1 \leq 5 \\ x_2 \leq 0 \\ x_1, x_2 \in \mathbb{N} \\ \max(z) = 10x_1 + x_2 \end{cases}$$

et

$$PL_5 = \begin{cases} 2x_1 + 5x_2 \leq 11 \\ x_1 \leq 5 \\ x_2 \geq 1 \\ x_1, x_2 \in \mathbb{N} \\ \max(z) = 10x_1 + x_2 \end{cases}$$

Les solutions sont les suivantes pour les deux programmes :

$$\begin{aligned} x_1^* &= 5, & x_2^* &= 0, & z^* &= 50 \text{ pour } PL_4 \\ x_1^* &= 3, & x_2^* &= 1, & z^* &= 31 \text{ pour } PL_5 \end{aligned}$$

Donner l'arbre d'évaluation et de séparation associé à la résolution du problème  $PL_1$ .

**Solution.** L'arbre d'évaluation et de séparation est donné ci-dessous :

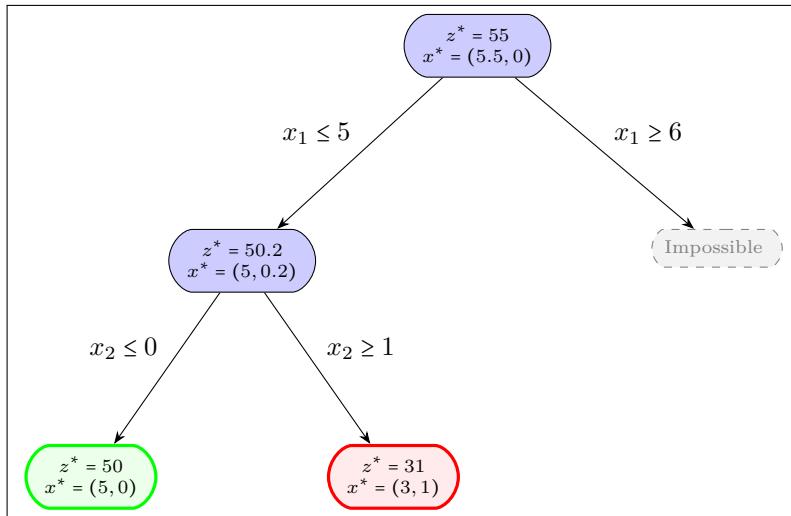


FIGURE 3 – Arbre de branchement du problème  $PL_1$

### 1.3 Résolution effective

#### Exercice 4 Résolution en utilisant la méthode de séparation et d'évaluation.

On considère le problème linéaire en nombres entiers ci-dessous :

$$PL_0 = \begin{cases} 2x_1 + x_2 \leq 6 \\ 2x_1 + 3x_2 \leq 9 \\ x_1, x_2 \in \mathbb{N} \\ \max(z) = 3x_1 + 4x_2 \end{cases}$$

1. Résoudre le programme linéaire en nombres entiers ci-dessus par la méthode d'évaluation et de séparation tel que le premier branchemet se fera sur la variable  $x_2$ .
2. Résoudre le programme linéaire en nombres entiers ci-dessus par la méthode d'évaluation et de séparation tel que le premier branchemet se fera sur la variable  $x_1$ .
3. Donner la signification géométrique du premier branchemet sur la variable  $x_2$ . Quels sont les principes vus en cours que l'on retrouve lors de l'interprétation géométrique ?

#### Solution.

1. L'arbre de branchement sur  $x_2$  est le suivant :

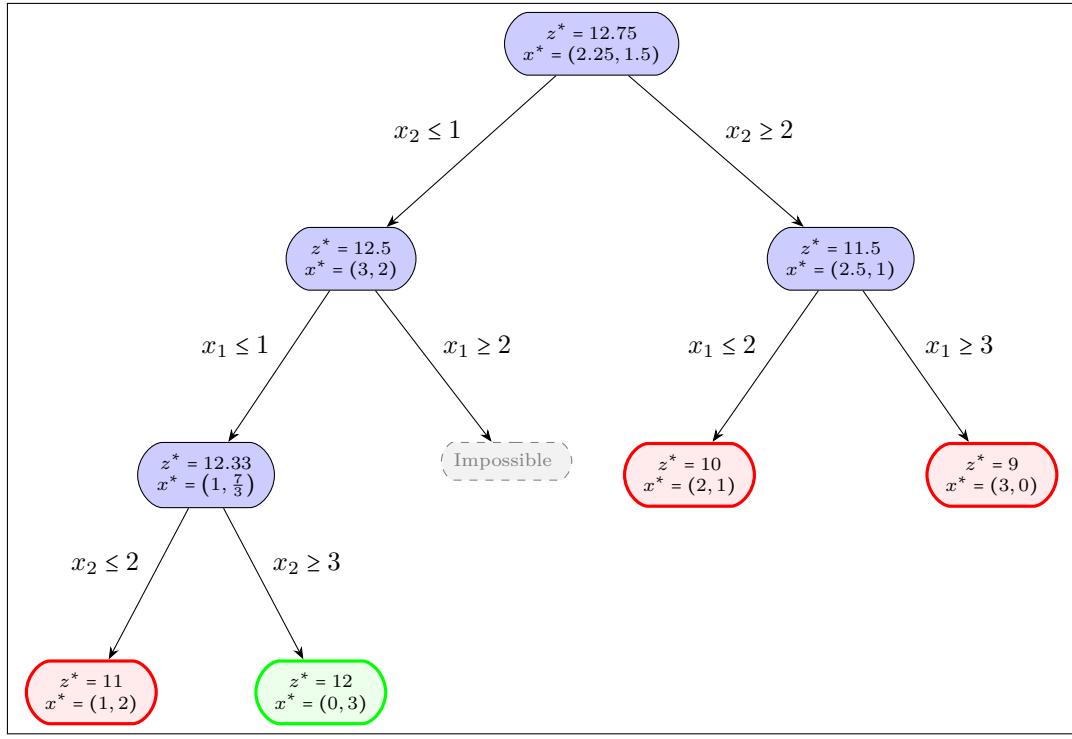


FIGURE 4 – Branch and bound

2. L'arbre de branchement sur  $x_1$  est le suivant :

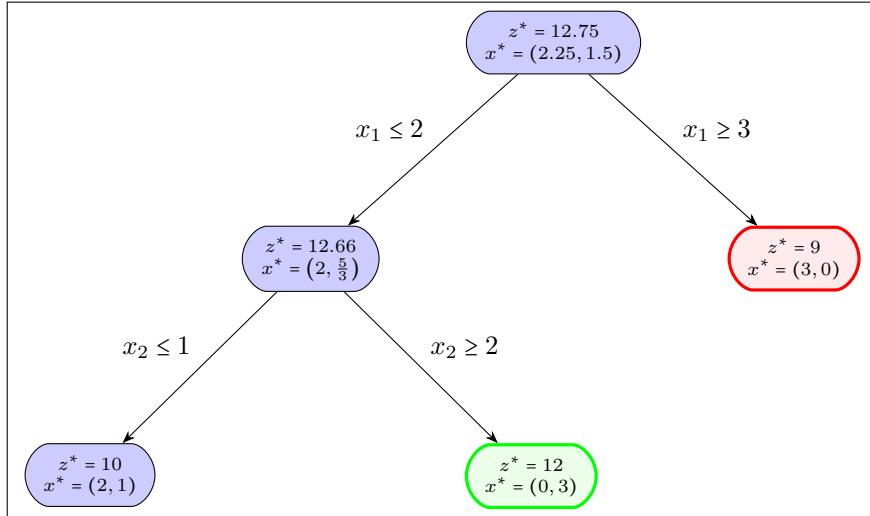


FIGURE 5 – Branch and bound

3. Le premier branchement sur la variable  $x_2$  correspond à la séparation de l'espace des solutions en deux parties par les deux demi-espaces  $x_2 \leq 1$  et  $x_2 \geq 2$ . Si on représente graphiquement le programme linéaire en nombres entiers  $PL_0$ , cela donne :



FIGURE 6 – Représentation du problème  $PL_0$  dans  $\mathbb{R}^2$

### Exercice 5 Branch and bound et plus court chemin.

Nous considérons le graphe donné par la figure 7 suivante :



FIGURE 7 – Recherche d'un plus court chemin par branch and bound

Donner deux stratégies en utilisant le principe du branch and bound pour résoudre le problème d'un plus court chemin. Appliquer vos stratégies sur le graphe de la figure 7.

### Solution.

### Exercice 6 Contraintes avec une seule variable.

On considère un problème de programmation linéaire standard, à une seule contrainte, défini par

$$\max \sum_{j=1}^n u_j x_j$$

avec

$$\sum_{j=1}^n v_j x_j \leq V, \text{ et } x_j \geq 0, \text{ pour } 1 \leq j \leq n$$

Tous les coefficient  $u_j$  et  $v_j$  sont supposés strictement positifs et l'on suppose les variables classées par rapports utilité/volume décroissants ou, pour rester dans un formalisme plus mathématique, suivant les valeurs décroissantes des rapports  $\frac{u_j}{v_j}$ . Montrer que la variable  $x_1$  est entrante et que, en la faisant entrer en base, on atteint l'optimum de l'objectif en une seule étape. Exprimer la valeur de l'objectif en fonction des différents coefficients.

### Solution.

### Exercice 7 Le problème du voyageur de commerce symétrique.

1. Appliquer la méthode par séparation et évaluation décrite en cours pour résoudre le problème du voyageur de commerce dans le graphe  $K_6$  dont la matrice des poids est la suivante :

|     | $x$      | $y$      | $z$      | $t$      | $u$      | $v$      |
|-----|----------|----------|----------|----------|----------|----------|
| $x$ | $\infty$ | 4        | 7        | 2        | 5        | 4        |
| $y$ | 4        | $\infty$ | 3        | 2        | 1        | 2        |
| $z$ | 7        | 3        | $\infty$ | 2        | 6        | 3        |
| $t$ | 2        | 2        | 2        | $\infty$ | 5        | 3        |
| $u$ | 5        | 1        | 6        | 5        | $\infty$ | 2        |
| $v$ | 4        | 2        | 3        | 3        | 2        | $\infty$ |

2. Que faudrait-il faire si on voulait connaître tous les cycles de poids minimum ?

#### Solution.

##### 1. Construction du 1-arbre initial

On considère le graphe complet  $G$  sur  $V = \{x, y, z, t, u, v\}$ . On fixe le sommet  $x$  et on calcule un 1-arbre.

On applique l'algorithme de Kruskal au graphe  $G \setminus \{x\}$  muni des poids initiaux.

Les arêtes sélectionnées sont :

$$y-u = 1, \quad y-t = 2, \quad y-v = 2, \quad z-t = 2.$$

Poids de l'ACPM :

$$7.$$

On ajoute les deux arêtes incidentes à  $x$  de poids minimal :

$$x-t = 2, \quad x-y = 4.$$

On obtient le 1-arbre initial de poids

$$LB_0 = 7 + 2 + 4 = 13.$$

Les degrés sont :

$$\deg(x) = 2, \deg(y) = 4, \deg(t) = 3, \deg(z) = 1, \deg(u) = 1, \deg(v) = 1.$$

Ce n'est pas un cycle hamiltonien.

On construit une solution réalisable :

$$x \rightarrow t \rightarrow z \rightarrow y \rightarrow u \rightarrow v \rightarrow x$$

de poids

$$14.$$

On fixe donc

$$UB = 14.$$

Comme  $LB_0 < UB$ , on applique la séparation.

#### Règle de branchement

Si le 1-arbre courant n'est pas hamiltonien, on choisit un sommet  $v$  tel que  $\deg(v) > 2$ . On crée une branche pour chacune des arêtes incidentes à  $v$  dans le 1-arbre courant, en les excluant successivement.

Toute exclusion est modélisée par une fonction de poids modifiée  $w^{(k)}$  où l'arête exclue reçoit le poids  $\infty$ . Cette fonction est utilisée :

- pour l'ACPM sur  $G \setminus \{x\}$ ,
- pour le choix des deux arêtes incidentes à  $x$ .

#### **Branchement au niveau racine**

Dans le 1-arbre initial,  $\deg(y) = 4 > 2$ . On branche donc sur  $y$ .

Les arêtes incidentes à  $y$  dans le 1-arbre sont :

$$y-u, \quad y-t, \quad y-v, \quad y-x.$$

On crée donc quatre branches.

#### **Branche A : exclusion de $y-u$**

On impose  $w(y, u) = \infty$ .

ACPM sur  $G \setminus \{x\}$  :

$$y-t = 2, \quad y-v = 2, \quad z-t = 2, \quad u-v = 2.$$

Poids = 8.

On ajoute  $x-t = 2$  et  $x-y = 4$ .

$$LB_A = 8 + 2 + 4 = 14.$$

Comme  $LB_A = UB$ , la branche ne peut être élaguée.

Dans ce 1-arbre,  $\deg(y) = 3 > 2$ . On branche à nouveau sur  $y$ .

Les arêtes incidentes à  $y$  sont

$$y-t, \quad y-v, \quad y-x.$$

#### **Branche A1 : exclusion de $y-t$**

Il reste à  $y$  les arêtes  $y-v = 2$  et  $y-z = 3$ .

Dans tout arbre couvrant,  $y$  doit être incident à au moins une arête. Toute ACPM contient donc une arête de poids au moins 3 incidente à  $y$ .

Les trois autres arêtes ont poids minimal total 6.

Toute ACPM a donc poids  $\geq 9$ .

$$LB_{A1} \geq 9 + 2 + 4 = 15 > UB.$$

Branche élaguée.

#### **Branche A2 : exclusion de $y-v$**

Raisonnement analogue :

Toute ACPM contient une arête incidente à  $y$  de poids au moins 3.

$$LB_{A2} \geq 9 + 2 + 4 = 15 > UB.$$

Branche élaguée.

#### **Branche A3 : exclusion de $y-x$**

ACPM inchangé (poids 8).

Les deux arêtes minimales incidentes à  $x$  deviennent  $x-t = 2$  et  $x-v = 4$ .

$$LB_{A3} = 8 + 2 + 4 = 14.$$

Tous les sommets ont degré 2.

On obtient un cycle hamiltonien de poids 14.

Branche fermée

**Branche B : exclusion de  $y-t$**

On impose  $w(y, t) = \infty$ .

ACPM :

$$y-u = 1, \quad y-v = 2, \quad z-t = 2, \quad u-v = 2.$$

Poids = 7.

Ajout de  $x-t = 2$  et  $x-y = 4$ .

$$LB_B = 13.$$

$\deg(y) = 3$ . On branche sur  $y$ .

**Branche B1 : exclusion de  $y-u$**

Toute ACPM contient alors une arête incidente à  $y$  de poids  $\geq 3$ .

$$LB_{B1} \geq 15.$$

Élaguée.

**Branche B2 : exclusion de  $y-v$**

Même raisonnement :

$$LB_{B2} \geq 15.$$

Élaguée.

**Branche B3 : exclusion de  $y-x$**

Les arêtes incidentes à  $x$  deviennent  $x-t = 2$  et  $x-v = 4$ .

On obtient un cycle hamiltonien de poids 14.

Branche fermée.

**Branche C : exclusion de  $y-v$**

ACPM :

$$y-u = 1, \quad y-t = 2, \quad z-t = 2, \quad u-v = 2.$$

Poids = 7.

$$LB_C = 13.$$

On branche sur  $y$ .

Les deux sous-branches  $y-u$  exclue et  $y-t$  exclue donnent  $LB \geq 15$ , donc élaguées.

Exclusion de  $y-x$  donne un cycle hamiltonien de poids 14.

Branche fermée.

**Branche D : exclusion de  $y-x$**

ACPM initial inchangé (poids 7).

Arêtes incidentes minimales à  $x$  :

$$x-t = 2, \quad x-v = 4.$$

$$LB_D = 13.$$

On branche sur  $y$ .

Les exclusions successives de  $y-u$  et  $y-t$  donnent  $LB \geq 15$  : branches élaguées.

L'exclusion de  $y-v$  produit un cycle hamiltonien de poids 14.

Branche fermée.

### Conclusion globale

Toutes les branches de l'arbre de séparation ont été :

- soit élaguées par une borne strictement supérieure à 14,
- soit explorées complètement et conduisant à un cycle de poids 14.

Il n'existe donc aucun cycle hamiltonien de poids strictement inférieur à 14.

Poids optimal = 14.

L'arbre de séparation est donné à la figure 11 en annexe.

2. Pour obtenir *tous* les cycles optimaux, il faut poursuivre la séparation en n'élaguant que lorsque  $LB > UB$  (et non  $LB \geq UB$ ), et enregistrer toutes les solutions de poids 14.

## 1.4 Inégalités valides

### Exercice 8 Simples inégalités valides.

Considérons l'ensemble suivant :

$$X = \{x_i \in \{0, 1\} : 3x_1 - 4x_2 + 2x_3 + x_5 \leq -2\}$$

Donner deux équations valides.

**Solution.** On a les 5 inégalités suivantes :

$$\begin{aligned} \sum_{i=1}^5 x_i &\leq 4 \\ x_2 + x_4 &\geq 1 \\ x_1 + x_3 + x_5 &\leq 2 \\ x_1 &\leq x_2 \\ x_1 &\leq x_4 \\ x_3 &\leq x_2 \end{aligned}$$

### Exercice 9 Simple inégalité valide.

Considérons l'ensemble suivant :

$$X = \{(x, y), 0 \leq x \leq 5, y \in \{0, 1\} : x \leq 9999y\}$$

Donner une inégalité valide.

**Solution.** On a l'inégalité suivante :

$$x \leq 5y$$

### Exercice 10 Simple inégalité valide.

Considérons l'ensemble suivant :

$$X = \{(x, y), 0 \leq x \leq 14, y \in \{0, 1\} : x \leq 10y\}$$

Donner une inégalité valide.

**Solution.** On n'a pas su trouver d'inégalité valide pour cet ensemble.

**Exercice 11 Simple inégalité valide.** Considérons la région entière  $X = P \cap \mathbb{N}^4$  avec

$$P = \{x \in \mathbb{R}_+^4 : 13x_1 + 20x_2 + 11x_3 + 6x_4 \geq 72\}$$

Donner une inégalité valide.

**Solution.** On sait que les  $x_i$  sont des entiers positifs. Par conséquent, on peut trouver une inégalité valide en arrondissant les coefficients de l'inégalité à l'entier supérieur le plus proche. Par exemple, on peut faire comme suit :

$$\begin{aligned} 13x_1 + 20x_2 + 11x_3 + 6x_4 \geq 72 &\implies \frac{13}{11}x_1 + \frac{20}{11}x_2 + x_3 + \frac{6}{11}x_4 \geq \frac{72}{11} \\ &\implies 2x_1 + 2x_2 + x_3 + x_4 \geq 7 \end{aligned}$$

On peut faire ce procédé pour toute valeur entière pour obtenir des inégalités valides différentes.

### Exercice 12 Inégalités valides.

Considérons une instance de BIN PACKING PROBLEM :

Algorithme 1 – BIN PACKING PROBLEM

**Input:** Soient  $n$  nombres entiers  $a_1, \dots, a_n$  et  $W, k \in \mathbb{N}^*$

**Question:** Existe-t-il une partition en  $k$  boîtes  $B_1, \dots, B_k$  de capacité  $W$  telle que  $\forall j \in \{1, \dots, k\}, \sum_{i \in B_j} a_i \leq W$  ?

On considère l'instance suivante :

- La capacité des boîtes est  $W = 6$ .
  - Les objets à ranger sont  $a_1 = 2, a_2 = 3, a_3 = 3, a_4 = 3, a_5 = 4, a_6 = 4, a_7 = 5$ .
1. Donner un algorithme linéaire en nombres entiers qui modélise le BIN PACKING PROBLEM.
  2. Donner une solution optimale relaxée pour l'instance donnée. Montrer que cette solution est optimale.
  3. Proposer une borne inférieure pour toute solution optimale.
  4. Donner un ensemble de coupes pour l'instance.

### Solution.

1. On peut modéliser le BIN PACKING PROBLEM de deux manières différentes :

- On utilise des variables binaires  $x_{ij}$  qui indiquent si l'objet  $a_i$  est placé dans la boîte  $j$ , et des variables binaires  $y_j$  qui indiquent si la boîte  $j$  est utilisée. L'objectif est de minimiser le nombre de boîtes utilisées, soit :

$$\begin{cases} \min \sum_j y_j \\ \sum_j x_{ij} = 1, \quad \forall i \in \{1, \dots, 7\} \\ \sum_i a_i x_{ij} \leq W y_j, \quad \forall j \\ x_{ij} \in \{0, 1\}, \quad \forall i, j \\ y_j \in \{0, 1\}, \quad \forall j \end{cases}$$

- Une autre modélisation consiste à ne pas considérer les  $y_i$  comme des variables, cela donne alors :

$$\begin{cases} \min k \\ \sum_j x_{ij} = 1, \quad \forall i \in \{1, \dots, 7\} \\ \sum_i a_i x_{ij} \leq W, \quad \forall j \\ x_{ij} \in \{0, 1\}, \quad \forall i, j \end{cases}$$

2. Une solution optimale pour cette instance consiste à placer les éléments comme suit :

- $a_7$  est dans la boîte 1,
- $a_6$  et  $a_1$  sont dans la boîte 2,
- $a_5$  est dans la boîte 3,
- $a_4$  et  $a_3$  sont dans la boîte 4,
- $a_2$  est dans la boîte 5.

Cette solution utilise 5 boîtes. Une borne inférieure pour toute solution optimale est donnée par  $\left\lceil \frac{\sum_i a_i}{W} \right\rceil = \left\lceil \frac{22}{6} \right\rceil = 4$ . Les éléments  $a_4, \dots, a_7$  nécessitent chacun une boîte, et les éléments  $a_2$  et  $a_3$  ne peuvent rentrer tous deux dans ces boîtes. Le nombre minimum de boîtes est alors de 5, et la solution proposée est optimale.

3. Une borne inférieure pour toute solution optimale est donnée par

$$\left\lceil \frac{\sum_i a_i}{W} \right\rceil$$

4. Un ensemble de coupes pour l'instance donnée serait (pour tout  $j$ ) :

$$C_1 = \begin{cases} x_{7j} + x_{1j} \leq y_j \\ x_{7j} + x_{2j} \leq y_j \\ x_{7j} + x_{3j} \leq y_j \\ x_{7j} + x_{4j} \leq y_j \\ x_{7j} + x_{5j} \leq y_j \\ x_{7j} + x_{6j} \leq y_j \end{cases}, \quad C_2 = \begin{cases} x_{6j} + x_{2j} \leq y_j \\ x_{6j} + x_{3j} \leq y_j \\ x_{6j} + x_{4j} \leq y_j \\ x_{6j} + x_{5j} \leq y_j \end{cases},$$

$$C_3 = \begin{cases} x_{5j} + x_{2j} \leq y_j \\ x_{5j} + x_{3j} \leq y_j \\ x_{5j} + x_{4j} \leq y_j \end{cases}, \quad C_4 = \begin{cases} x_{4j} + x_{3j} + x_{2j} \leq 2y_j \\ x_{4j} + x_{3j} + x_{1j} \leq 2y_j \\ x_{4j} + x_{2j} + x_{1j} \leq 2y_j \end{cases},$$

### Exercice 13 Le polytope pour le 0 – 1 Sac à Dos.

Soit

$$S = \{x \in \mathbb{R}^5 \mid 79x_1 + 53x_2 + 53x_3 + 45x_4 + 45x_5 \leq 178\}.$$

avec  $x_i \in \{0, 1\}$  pour  $i = 1, \dots, 5$ .

1. Donner les ensembles minimaux dépendants et donner les inégalités valides.
2. Donner l'extension  $E(C)$  d'un ensemble minimal dépendant et les coupes associées.

**Solution.** Les ensembles minimaux dépendants sont les suivants :

- $C_1 = \{1, 2, 3\}$  avec  $a(C_1) = 185 > 178$ ,
- $C_2 = \{1, 2, 4, 5\}$  avec  $a(C_2) = 222 > 178$ ,
- $C_3 = \{1, 3, 4, 5\}$  avec  $a(C_3) = 222 > 178$ ,
- $C_4 = \{2, 3, 4, 5\}$  avec  $a(C_4) = 196 > 178$ .

Les inégalités valides associées à ces ensembles sont les suivantes :

- $x_1 + x_2 + x_3 \leq 2$ ,
- $x_1 + x_2 + x_4 + x_5 \leq 3$ ,
- $x_1 + x_3 + x_4 + x_5 \leq 3$ ,
- $x_2 + x_3 + x_4 + x_5 \leq 3$ .

L'extension  $E(C)$  de ces ensembles est la même pour tous ces ensembles sauf pour  $C_4$  qui devient alors

$$x_1 + x_2 + x_3 + x_4 + x_5 \leq 3.$$

## 1.5 Coupes

### Exercice 14 Sur les coupes de Gomory.

Une coupe  $ax \leq \alpha$  est dite « plus profonde » qu'une coupe  $a'x \leq \alpha'$  si

$$\{x \in \mathcal{D} \mid ax \leq \alpha\} \subset \{x \in \mathcal{D} \mid a'x \leq \alpha'\},$$

1. Expliquer le sens de ce terme.
2. Essayer de déterminer la coupe la plus profonde dans les exercices 15 et 16 suivants.

### Solution.

### Exercice 15 Sur les coupes de Gomory.

On considère le programme linéaire suivant :

$$PL = \begin{cases} 2x_1 + 5x_2 \leq 17 \\ 3x_1 + 2x_2 \leq 10 \\ x_1, x_2 \in \mathbb{N} \\ \max(z) = 2x_1 + x_2 \end{cases}$$

1. Résoudre le programme linéaire en nombres entiers par la méthode des coupes de Gomory.
2. Résoudre le programme linéaire en nombres entiers par la méthode du Branch and Bound.

### Solution.

### Exercice 16 Algorithme de coupe de Gomory.

Donner la solution du programme linéaire suivant :

$$PL = \begin{cases} 2x_1 + x_2 \leq 6 \\ 2x_1 + 3x_2 \leq 9 \\ x_1, x_2 \in \mathbb{N} \\ \max(z) = 3x_1 + 4x_2 \end{cases}$$

### Solution.

## 1.6 Enveloppe convexe

### Exercice 17 Enveloppe convexe de points.

Soit  $S \subseteq \{0, 1\}^3$  qui consiste aux points suivants :

$$S = \{(0, 0, 0), (1, 1, 1), (0, 1, 0), (1, 0, 1)\}$$

Considérons le polytope  $P = \text{conv}(S)$ . Trouver un système d'inéquations linéaires qui caractérise  $P$ , c'est-à-dire tel que :

$$P = \{x \in \mathbb{R}^3 \mid Ax \leq b\}$$

### Solution.

### Exercice 18 Enveloppe convexe et points extrêmes.

Soit  $S \subseteq \{0, 1\}^n$ , c'est-à-dire un ensemble de  $\{0, 1\}$ -vecteurs de dimension  $n$ . Soit le polytope  $P = \text{conv}(S)$ . Montrer que  $x$  est un sommet de points de  $P$  si et seulement si  $x \in S$ .

### Solution.

#### Exercice 19 Enveloppe convexe et points extremes.

Soit  $S$  l'ensemble suivant :

$$S = \{x \in \mathbb{R}^3 \mid x_1 + x_3 \leq 1, x_1 - x_3 \leq 0, x_1 \geq 0\}$$

Donner  $\text{conv}(S)$ .

### Solution.

#### Exercice 20 Enveloppe convexe et points extremes.

Considérons un polytope  $P \subset \mathbb{R}^n$  tel que  $P = \text{conv}(\{x_1, \dots, x_t\})$ . Montrer que si  $x$  est un point extrême de  $P$  alors  $x \in \{x_1, \dots, x_t\}$ . Est-ce que  $x_j$  est forcément un point extrême de  $P$  pour tout  $j \in \{1, \dots, t\}$  ?

### Solution.

#### Exercice 21 Propriétés de l'enveloppe convexe.

Soient  $A, B \subseteq \mathbb{R}^n$  deux ensembles convexes. Montrer les propriétés suivantes :

1.  $\text{conv}(A \cup B) = \text{conv}(\text{conv}(A) \cup \text{conv}(B))$ .
2.  $\text{conv}(A + B) = \text{conv}(A) + \text{conv}(B)$ . Aide : On peut considérer la somme  $\sum_{j,k} \lambda_j \mu_k (a_j + b_k)$  où  $\sum_j \lambda_j = 1$ ,  $\sum_k \mu_k = 1$ ,  $\lambda_j, \mu_k \geq 0$ ,  $a_j \in A$  et  $b_k \in B$ .
3. Soient  $P$  et  $Q$  deux polytopes de  $\mathbb{R}^n$ . Montrer que les ensembles suivants sont aussi des polytopes :
  - $P \times Q = \{(p \times q) \in \mathbb{R}^{2n} \mid p \in P, q \in Q\}$ .
  - $P \cap Q$ .
  - $\text{conv}(P \cup Q)$ .

### Solution.

#### Exercice 22 Propriétés de l'enveloppe convexe.

Soit  $x_0 \in \mathbb{R}^n$  et soit  $C \subseteq \mathbb{R}^n$  un convexe. Montrer que

$$\text{conv}(C \cup \{x_0\}) = \{\lambda x + (1 - \lambda)x_0 \mid x \in C, \lambda \in [0, 1]\}.$$

### Solution.

#### Exercice 23 Enveloppe convexe.

Soit  $C$  le convexe défini comme étant l'enveloppe convexe de  $S$  :  $C = \text{conv}(S)$ . Montrer que tout point extrême de  $C$  appartient à  $S$ .

### Solution.

#### Exercice 24 Faces.

Soit  $P_\varepsilon \in \mathbb{R}^2$  le polyèdre défini par les inégalités linéaires suivantes :

$$P_\varepsilon = \begin{cases} x_2 \geq 3 \\ \varepsilon x_1 + (2 - \varepsilon)x_2 \geq 4 \\ x_i \geq 0, \quad \forall i \in \{1, 2\} \end{cases}$$

1. Illustrer  $P_\varepsilon$  et les inégalités dans le plan pour  $\varepsilon = -1$  et  $\varepsilon = 1$ .
2. Quelles sont les facettes de  $P_1$  ?

3. Soit  $\varepsilon = 3$  et le polyèdre entier  $P_I = \text{conv}(P_\varepsilon \cap \mathbb{Z}^2)$ . Dessiner  $P_I$  et donner les équations qui donnent l'enveloppe convexe.

**Solution.**

**Exercice 25 Preuve sur les enveloppes convexes.**

1. Soit  $X$  l'ensemble suivant

$$X = \{(x, y) \in \mathbb{R}_+^m \times B^1 : \sum_{i=1}^m x_i \leq my, x_i \leq 1, \forall i \in \{1, \dots, m\}\}$$

où  $B^1 = \{0, 1\}$ . Considérons la formulation suivante :

$$P = \{(x, y) \in \mathbb{R}_+^m \times \mathbb{R}^1 : x_i \leq y, \forall i \in \{1, \dots, m\}, y \leq 1\}$$

Montrer que la matrice  $A$  ou la partie  $(A, B)$  admet une structure particulière garantissant que  $P = \text{conv}(X)$ .

2. Montrer que les points  $(x, y) \in P$  (pour l'ensemble  $X$ ) avec  $y$  fractionnaire sont des points extrêmes de  $P$ .

**Solution.**

## 2 Génération de colonnes

Rappelons le principe de l'algorithme de la génération de colonnes :

- Initialisation : on met quelques colonnes de  $PL$ . Soit  $PL_R$ .
- Itérations :
  - Résoudre  $(PL_R)$ .
  - Calculer les couts réduits des variables de  $(PL)$ .
  - Si toutes les variables de  $(PL)$  ont un cout réduit  $\geq 0$  alors STOP ( $(PL)$  est résolu). Sinon ajouter une variable de cout réduit minimum à  $(PL_R)$  et résoudre à nouveau  $(PL_R)$ .

Pour calculer les couts réduits :

- on utilise les variables duales  $\mu \geq 0$  associées aux contraintes
- Couts réduits :  $c - \mu A$ .
- Dimensions :
  - $c$  et  $\mu$  vecteurs lignes,
  - $c$  avec  $n$  colonnes et  $\mu$  avec  $m$  colonnes,
  - $A$  la matrice  $m \times n$
  - $n$  le nombre de variables et  $m$  le nombre de contraintes.

### Exercice 26 Génération de colonnes.

On considère le problème  $PL$  suivant :

$$PL = \begin{cases} \min(z) = 2x_1 + 3x_2 \\ 4x_1 + x_2 - x_3 \geq 5 \\ 3x_1 + 2x_2 - x_4 \geq 6 \\ x_1 + 2x_2 - x_5 \geq 3 \\ x_i \geq 0, \forall i \in \{1, \dots, 5\} \end{cases}$$

Commencer la méthode de génération de colonne avec les variables  $x_1, x_3$  et  $x_4$ . Ecrire le problème restreint  $PL_R$ . Calculer les coûts réduits.

### Solution.

### Exercice 27 Encadrement de la valeur optimale.

Montrer qu'à chaque itération de la génération de colonnes on a

$$\text{valeur}(PL) \leq \text{valeur}(PL_R)$$

et

$$\text{score}(PL_R) + k \cdot \text{cred} \leq \text{score}(PL)$$

où  $k$  est la valeur d'une solution optimale  $\sum_{j=1}^n x_j^* \leq k$  et cred est le cout reduit minimum à une itération donnée.

### Solution.

### Exercice 28 Génération de colonnes : le problème de la tournée de véhicules avec fenêtres temporelles.

### Solution.

### 3 Lagrangien

#### Exercice 29 Lagrangien.

Soit le problème ( $P$ ) suivant :

$$(P) = \begin{cases} \min f(x) = x_1^2 + x_2^2 \\ g(x) = 2x_1 + x_2 + 4 \leq 0 \\ x_i \in \mathbb{R}, \forall i \in \{1, 2\} \end{cases}$$

1. On relache la contrainte et on lui affecte le multiplicateur de Lagrange  $\lambda \geq 0$ .
2. Enoncer ( $D$ ) le problème dual lagrangien de ( $P$ ).
3. Calculer la fonction duale  $\Delta(\lambda)$  en fonction de  $\lambda$ .
4. Résoudre le problème dual ( $D$ ).

#### Solution.

## 4 Modélisation

### Exercice 30 Le problème de la coloration de sommets.

Donner un programme linéaire en nombres entiers qui modélise le problème de la coloration de sommets.

### Solution.

### Exercice 31 Le problème du stable dans un graphe.

On rappelle la définition suivante :

**Définition 4.1.** Un **stable** dans un graphe est un ensemble de sommets qui sont deux à deux non adjacents.

La matrice d'incidence (arêtes-sommets)  $A = (a_{ij})$  est une matrice  $m \times n$  définie de la manière suivante :

$$a_{ij} = \begin{cases} 1 & \text{si l'arête } i \text{ est incidente au sommet } j, \\ 0 & \text{sinon.} \end{cases}$$

1. Donner la matrice d'incidence du graphe donné par la figure 8 :

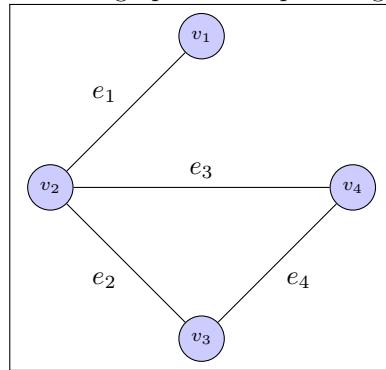


FIGURE 8 – Un graphe non orienté

2. Le problème de trouver un stable maximum dans un graphe ci-dessus peut se formuler comme un programme linéaire. Donner cette formulation.

### Solution.

### Exercice 32 Le problème du transversal dans un graphe.

On rappelle la définition suivante :

**Définition 4.2.** Un **transversal** (couverture) est un ensemble de sommets tel que pour chaque arête du graphe, au moins une de ses deux extrémités appartient au transversal.

On considère le graphe donné par la figure 8. Donner le programme linéaire en nombres entiers qui permet de trouver un transversal de taille minimale dans ce graphe.

### Solution.

### Exercice 33 Modéliser les problèmes suivants par un programme linéaire en nombre entiers.

1. L'arbre couvrant de poins minimum. Quel est le problème sur le nombre de contraintes ?
2. Le problème du plus court chemin entre un sommet  $s$  et un sommet  $t$ . Soit  $G = (V, E)$  un graphe orienté avec des coûts  $c_{ij} \geq 0$  sur les arcs  $(i, j) \in E$ . Soit  $F = \{P = (e_{j_1}, e_{j_2}, \dots, e_{j_k})\}$  une séquence d'arcs d'un plus court chemin entre  $s$  et  $t$  dans le graphe  $G$ . On considère

le graphe donné par la figure 9 ci-dessous :

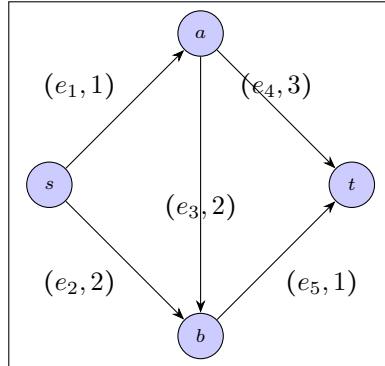


FIGURE 9 – Un graphe orienté pondéré

- Donner sa matrice d'incidence  $A$ .
- Donner le dual.
- Résoudre le problème par l'algorithme du simplexe.
- Considérons un vecteur  $f = (f_1, \dots, f_n)$  avec  $f_i$  associé à l'arc  $e_i$  tel que  $f_i = 1$  si l'arc  $e_i$  appartient à un plus court chemin  $P$  entre  $s$  et  $t$ , et  $f_i = 0$  sinon.
  - Que signifie  $\sum_{j=1}^m a_{ij} f_j = 1$  ?
  - Que signifie  $\sum_{j=1}^m a_{ij} f_j = -1$  ?
  - Que signifie  $\sum_{j=1}^m a_{ij} f_j = 0$  ?
- Utiliser l'algorithme du dual simplexe.

### Solution.

#### Exercice 34 Sur le problème du voyageur de commerce.

Le problème du voyageur de commerce consiste à effectuer un circuit Hamiltonien de cout minimum dans un graphe complet non orienté. Pour cela nous considérons  $n + 1$  villes tel que le cout entre deux villes est donné par la matrice  $C = (c_{ij})$  où  $c_{ij}$  est le cout pour aller de la ville  $i$  à la ville  $j$ .

- Pourquoi le problème du voyageur de commerce est étudié dans le cadre d'un graphe complet valué et non dans un graphe quelconque ?
- Rappeler la définition d'un circuit Hamiltonien. Quelles conséquences sur les degrés des sommets du circuit Hamiltonien ? Modéliser ce problème par un programme linéaire en nombres entiers en justifiant les contraintes.
- Donner un exemple qui satisfait les contraintes du programme linéaire en nombres entiers mais qui n'est pas une solution réalisable du voyageur de commerce. Quel est l'inconvénient des nouvelles contraintes ? (Penser aux problèmes des sous-tours.)
- Nous ajoutons au programme linéaire en nombres entiers les contraintes suivantes :

$$u_i - u_j + nx_{ij} \leq n - 1, \quad \forall 1 \leq i \neq j \leq n$$

Montrer que les contraintes qu'on vient d'ajouter définissent bien le problème du voyageur de commerce.

- Comment appelle-t-on ce type de programme linéaire ?

### Solution.

**Exercice 35 Le problème du flot maximum versus la coupe minimum.**

1. Questions préliminaires
  - (a) Rappeler le principe des algorithmes principaux pour résoudre le problème du flot maximum.
  - (b) Dans toute solution optimale, que dire des arcs retours ? Et des arcs avant appartenant à une coupe minimale ?
2. Modélisation et propriétés. On considère dans la partie qui suit le graphe orienté pondéré donné par la figure 10 suivante :

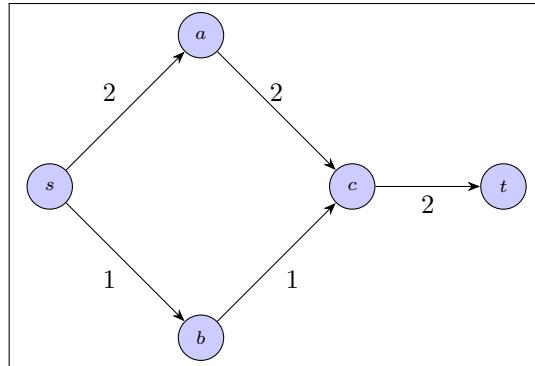


FIGURE 10 – Un réseau de flot

- (a) Formuler le problème du flot maximum par un programme linéaire noté  $P_0$  pour le réseau donné par la figure 10. Formuler le problème du flot maximum par un programme linéaire  $\mathcal{P}$ .
- (b) Donner le programme dual de  $P_0$  et de  $\mathcal{P}$ . Quelle est la structure de la matrice des contraintes pour le dual ?
- (c) Utiliser les écarts complémentaires pour retrouver un théorème de MIN-MAX.

**Solution.**

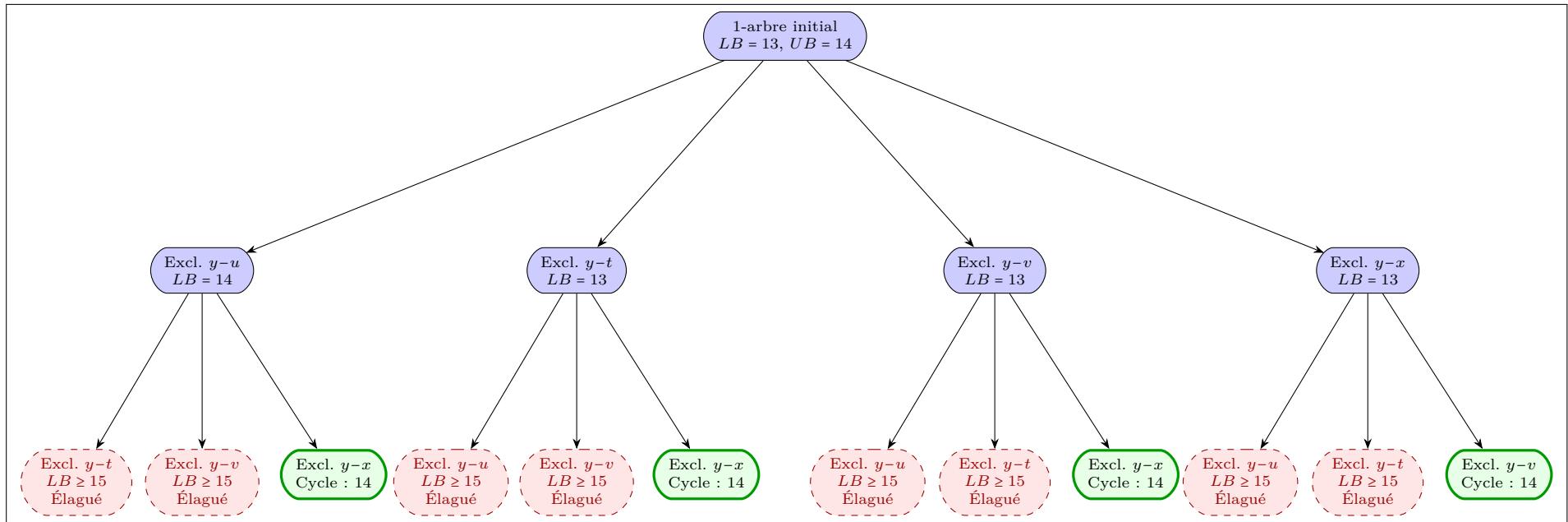


FIGURE 11 – Arbre de séparation et évaluation — TSP