Cours FIIFO 4

PROGRAMMATION LINÉAIRE

Cours de Abdel LISSER
Rédigé par Marc SIBERT
Mise à jour le 7 juin 2004

TABLE DES MATIÈRES

Qu'est ce que la Recherche Opérationnelle

exemple : affectation des 60 fréquences disponibles pour chaque cellule d'un réseau G.S.M. (jusqu'à 3000 cellules dans une ville moyenne). C'est un problème quadratique (produit de variables). Il existe 3 manières de résoudre de tels problèmes :

1. par linéarisation : pour des problèmes booléens $x \in \{0, 1\} \rightarrow x \in [0..1]$ (relaxation linéaire) donne une borne inférieure qui donne une idée de la qualité ;
2. par heuristique ;
3. par relaxation semi-définie (S.D.P.).

Nous utiliserons des logiciels de programmation linéaire :

- CPlex de Ilog

I. Introduction à la programmation linéaire

G.B. Dantzig (1947) découvre la méthode des simplexes pour l'optimisation ou l'affectation des ressources
C'est la plus grande avancée / invention du XX^{ème} siècle.
La taille des programmes linéaires a fortement augmenté (très lié à l'informatique), de quelques dizaines de variables à plusieurs millions aujourd'hui, accompagné par une grande évolution algorithmique : le simplex a une complexité en $O(2^n)$.
Karmarkar (1984) : algorithme polynomial ayant une complexité $O(n^{3,5}L^2)$ dit des « Points Intérieurs ». Il permet l'implémentation de la relaxation semi-définie
Secteurs d'utilisation : télécommunications, chimie (pétrole), finance, agro-alimentaire, transport, etc.
C'est un modèle puissant qui généralise des problèmes classiques :

- plus court chemin ;
- flot max ;
- multiflots ;
- arbres courants de poids minimum ;
- jeux à sommes nulles
- etc.

Exemple : fabrication de bière

	Maïs (C) (livres)	*Houblon (H) (onces)*	*Malt (M) (livres)*	*Bénéfices (€)*
Brune (A)	5	4	35	13
Blonde (B)	15	4	20	23
Quantités	480	160	1190	

L'entrepreneur cherche à minimiser le coût des choses produites (**problème dual**)

Fonction objectif $\min 480\ C + 160\ H + 1190\ M$ critère de l'entrepreneur

sous les contraintes $\begin{cases} 5\ C + 4\ H + 35\ M \geq 13 \rightarrow \textbf{(A)} & \text{critère du brasseur} \\ 15\ C + 4\ H + 20\ M \geq 23 \rightarrow \textbf{(B)} & \text{critère du brasseur} \\ C, H \text{ et } M \geq 0 & \text{critère des cours des matières} \end{cases}$

Le brasseur cherche à maximiser ses ventes / bénéfices (**problème primal**)

fonction objectif $\max 13\ A + 23\ B$

sous les contraintes $\begin{cases} 5\ A + 15\ B \leq 480 \rightarrow \textbf{(C)} & \text{ne pas dépasser les stocks} \\ 4\ A + 4\ B \leq 160 \rightarrow \textbf{(H)} \\ 35\ A + 20\ B \leq 1190 \rightarrow \textbf{(M)} \\ A, B \geq 0 \end{cases}$

La solution du primal = la solution du dual

Comment maximiser le profit du brasseur ?

1. <u>**Tester des possibilités**</u> :

a. Allouer toutes les ressources à la fabrication de la bière blonde :
   32 tonneaux → 736 €
b. tout pour la brune : 34 tonneaux → 442 €
c. 7,5 tonneaux de brune et 29,5 de blonde → 776 €
d. 12 tonneaux de brune et 28 de blonde → 800 €

Par tests aléatoires on peut arriver à la bonne solution, mais ça peut être très long.

$$(P)\begin{cases} A=28 \;\; B=12 \\ OPT=800 \end{cases} \qquad (D)\begin{cases} C=1 \;\; H=2 \;\; M=0 \\ \text{coût mini}=800 \;\; \text{solution optimale} \end{cases}$$

2. Analyse de la sensibilité

Brasseur : quel est le coût marginal pour acheter de nouvelles matières ?

Maïs = 1 € ; Houblon = 2 € ; Malt = 0 €

Si on veut faire une nouvelle bière qui nécessite 2 Maïs, 5 Houblons et 24 Malt. Quel profit obtenir de ce nouveau produit ?

Profit : $1*2\,€+2*5\,€+24*0\,€=12\,€$ le tonneau

3. Écriture matricielle du problème du brasseur[1]

Soient le vecteur $c=\begin{pmatrix}13\\23\end{pmatrix}$ (bénéfices) et le vecteur second membre $b=\begin{pmatrix}480\\160\\1190\end{pmatrix}$ (quantités), la matrice $A=\begin{pmatrix}5 & 15\\4 & 4\\35 & 20\end{pmatrix}$ (composition des produits) et le vecteur de variables de décision $x=\begin{pmatrix}A\\B\end{pmatrix}$ (où A et B sont les quantités de bière brune et blonde)

Le programme linéaire primal du problème s'écrit :

(P) $\begin{cases} \max c^t x \\ \text{s.c.} \quad Ax \le b \\ \qquad\; x \ge 0 \end{cases}$ et sous forme standard (Ps) $\begin{cases} \max c^t x \\ \text{s.c.} \quad Ax+v=b \\ \qquad\; x, v \ge 0 \end{cases}$ avec $v\begin{pmatrix}v_1\\v_2\\v_3\end{pmatrix}$ où $v1$, $v2$ et $v3$ sont les variables d'écart.

1 Petit rappel :

Matrice transposée : on appelle transposée d'une matrice A, la matrice notée A^T, obtenue à partir de A en échangeant les lignes et les colonnes (extrait de http://www.cnam.fr/instrumesure/RsrcTelechageables/MetrologieB1/19008K_matrices.pdf). Dans ce compte-rendu, c'est la notation A^t qui est utilisée.

Exemple :

$$A=\begin{pmatrix}1 & 5 & 3\\6 & 7 & 9\end{pmatrix} \qquad A^T=\begin{pmatrix}1 & 6\\5 & 7\\3 & 9\end{pmatrix}$$

Propriétés :

- Transposée d'une somme :
  $(A+B)^T=A^T+B^T$
- Transposée d'un produit :
  $(AB)^T=B^T A^T$

Soit le vecteur y de variables duales défini par : $y=\begin{pmatrix} C \\ H \\ M \end{pmatrix}$. Le dual de (P) s'écrit :

(D) $\begin{cases} \min b^t y \\ \text{s.c.} \quad A^t y \geq c \\ \qquad\; y \geq 0 \end{cases}$ et sous la forme standard (Ds) $\begin{cases} \min b^t y \\ \text{s.c.} \quad A^t y + w = c \\ \qquad\; y, w \geq 0 \end{cases}$ avec $w \begin{pmatrix} w1 \\ w2 \end{pmatrix}$ où $w1$ et $w2$ sont les variables d'écart.

II. Rappels de la programmation linéaire (PL)

Soit le programme linéaire (PL) suivant sous forme standard :

$$(1.1)\quad \begin{cases} \min z = c^t x & (1.1a) \\ \text{s.c.}\quad Ax = b & (1.1b) \\ \qquad x \geq 0 & (1.1c) \end{cases} \quad c, x \in \mathbb{R}^n, b \in \mathbb{R}^m \text{ et } A \in \mathbb{R}^{m \times n}$$

où n est le nombre de variables et m le nombre de contraintes.

II.1. Idées de base

Soit $P = \{x \in \mathbb{R}^n / Ax = b \text{ et } x \geq 0\}$ le domaine des solutions réalisables. On suppose que P est non vide et que le minimum de z est soit non borné, soit peut être atteint en un point extrême de P.

Soit x^* solution de base réalisable, $x^* = \left[\frac{x_B^*}{x_N^*}\right]$ avec x_B^* variables de base et x_N^* variables hors base. Pour les solutions réalisables $x_B^* \geq 0$; $x_N^* = 0$.

De même, la matrice A peut être partitionnée en fonction de x_B^* et x_N^*

$$A = [B|N], c = \left[\frac{c_B}{c_N}\right] \quad \text{avec } B \in \mathbb{R}^{m \times m} \text{(Régulière donc divisible)}$$

(1.1) peut s'écrire :

$$(1.2)\quad \begin{cases} \min z = c_B^t x_B + c_N^t x_N & (1.2a) \\ \text{s.c.}\quad B x_B + N x_N = b & (1.2b) \\ \qquad x_B, x_N \geq 0 & (1.2c) \end{cases}$$

(1.2b) devient : $x_B = B^{-1} b - B^{-1} N x_N$

En remplaçant x_B dans (1.2a), on obtient :

$$z = c_B^t (B^{-1} b - B^{-1} N x_N) + c_N^t x_N$$
$$\Leftrightarrow z = c_B^t B^{-1} b - c_B^t B^{-1} N x_N + c_N^t x_N$$
$$\Leftrightarrow z = c_B^t B^{-1} b + (c_N^t - c_B^t B^{-1} N) x_N \qquad (1.4)$$

$\boxed{z = c_B^t B^{-1} b + r^t \left[\frac{x_B}{x_N}\right]}$ avec $r = \left[\frac{0 \quad \}\textit{variable de base}}{c_N^t - c_B^t B^{-1} N \quad \}\textit{variable hors base}}\right]$ qui représente les coûts réduits.

Pour x^* réalisable, z^* s'écrit : $\boxed{z^* = c_B^t \underbrace{B^{-1} b}_{x_B^*}}$ avec $x_B^* = 0$

Par conséquent, (1.4) s'écrit : $z - z^* = r^t \begin{bmatrix} x_B \\ x_N \end{bmatrix}, \forall x \in P$ (1.6)

Si $r \geq 0$, i.e. toutes les composantes de $c_N^t - c_B^t B^{-1} N$ sont positives et alors $z - z^* \geq 0 \ \forall x \in P$. Dans ce cas, x^* est une solution optimale de (1.1).

Si une composante de $r \leq 0$, l'élément de x_N correspondant peut être augmenté de 0 à une valeur positive pour réduire la valeur de z. r est appelé vecteur des coûts réduits.

Théorème 1

Si $x^* = \begin{bmatrix} x_B^* \\ x_N^* \end{bmatrix} = \begin{bmatrix} B^{-1} b \\ 0 \end{bmatrix} \geq 0$ est une solution de base réalisable dont les coûts réduits (r) sont supérieurs à 0, alors r^* est solution optimale du PL (1.1).

II.2. Théorème de la dualité

II.2.1. Programme linéaire dual

Soit le PL standard :

$$\begin{cases} \min c^t x & (2.1a) \\ A x = b & (2.1b) \\ x \geq 0 & (2.1c) \end{cases} \quad \text{avec } c, x \in \mathbb{R}^n, A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$$

Soit x une solution réalisable non dégénérée. Soient B et N les matrices de base et hors base. Soient $\tilde{B}$ et $\tilde{N}$, les ensembles d'indices des variables de base et hors base respectivement.

D'après le théorème 1, x est optimal si et seulement si $r_q \geq 0, \ \forall q \in \tilde{N}$ ou de manière équivalente :

$$c_B^t B^{-1} A_q \leq c_q, \forall q \in \tilde{N} \quad (1)$$

On a aussi :

$$c_B^t B^{-1} A_p = c_p, \forall p \in \tilde{B} \quad (2)$$

Si l'on note $w^t = c_B^t B^{-1}$, (1) et (2) s'écrivent :

$$w^t A_q \leq c_q, \forall q \in \tilde{N}$$
$$w^t A_p \leq c_p, \forall p \in \tilde{B}$$

Par conséquent, sous forme vectorielle, on a :

$w^t [B|N] \leq c^t$ ou encore $\boxed{A^t w \leq c}$

Pour la fonction objectif ($x_N = 0$), on a :

$$(\underbrace{c_B B^t}_{w} b)^t = (w b)^t = b^t w = w^t b = \underbrace{c_B^t B^{-1}}_{w^t} b = c_B^t \underbrace{B^{-1} b}_{x_B} = c_B^t x_B = c^t x$$

d'où $\boxed{b^t w = c^t x}$

En général, on a seulement : $b^t w = w^t b = w^t A x \leq c^t x$ (pour $A x = b, x \geq 0$)

En fonction de w, on a le PL suivant :

$$(2.2)\quad \begin{cases} \max b^t w & (2.2a) \\ \text{s.c. } A^t w \leq c & (2.2b) \\ \quad w < 0 \text{ ou } w > 0 & \end{cases}$$

II.2.2. Dualité faible en PL

Théorème 2

Si $x°$ est solution primale réalisable,
si $w°$ est solution duale réalisable,
alors $c^t x^0 \geq b^t w^0$

Corollaire 2.1

Si $x°$ est solution primale réalisable,
si w° est solution duale réalisable et $c^t x^0 = b^t w^0$
alors x° et w° sont solutions optimales du PL.

Théorème 3 : dualité forte en PL

1. Si le PL primal (respectivement dual) a une solution optimale finie, alors le dual (respectivement primal) en a une. Cette solution est la même pour les 2 PL.
2. Si l'un des deux est non borné, l'autre n'est pas réalisable.

Conséquences de la dualité forte :

- Pas de saut de dualité entre primal et dual ($c^t x = b^t w$) ;
- Les multiplicateurs du simplexe représentent les variables duales w ;
- A chaque itération du simplexe, on a $c^t x = b^t w$;
- Le simplexe maintient :
  - la réalisibilité du primal,
  - saut de dualité = 0,
  - cherche la réalisibilité du dual.

III. Analyse de la complexité

- Introduite en 1970 pour évaluer les performances des algorithmes
- Analyse du plus mauvais cas pour mesurer la difficulté d'un problème
- Mesurée par le nombre d'opérations élémentaires (+, -, *, etc.)

Soit le PL suivant :

$$\begin{cases} \min c^t x \\ \text{s.c. } A x = b \\ \quad x \geq 0 \end{cases} \quad \text{où } A \in \mathbb{R}^{m*n}, x \in \mathbb{R}^n, b \in \mathbb{R}^m \quad \text{et } m, n \text{ (données entières)} \geq 2$$

Soit L la taille d'une instance[1] qui peut être représentée par le triplet (m, n, L). La complexité sera une fonction $f(m, n, L)$.

Soit $\tau > 0$ tel que le nombre d'opérations élémentaires nécessaires pour l'algorithme, pour une instance donnée est de l'ordre τ $f(m, n, L)$. On dit que la complexité est de l'ordre $O(f(m, n, L))$. Si $f(m, n, L)$ est polynomial, on dira que l'algorithme est polynomial.

III.1. Calcul de la complexité

En système binaire, il faut $(r+1)$ bits pour représenter un entier positif $\varepsilon \in [2^r, 2^{r+1}]$ pour $r \geq 0$. Par conséquent, pour ε, il faut $\lceil \log(1+\varepsilon) \rceil$ bits. Si l'on ajoute un bit pour le signe, il faut $1 + \lceil \log(1+|\varepsilon|) \rceil$ pour coder un entier quelconque.

Pour le PL, la taille des entrées est :

$$L = \lceil 1+\log(1+m) \rceil + \lceil 1+\log(1+n) \rceil + \sum_{j=1}^{n} 1 + \lceil \log(1+|c_i|) \rceil + \sum_{i=1}^{m} \sum_{j=1}^{n} 1 + \lceil \log(1+|a_{ij}|) \rceil + \sum_{i=1}^{m} 1 + \lceil \log(1+|b_i|) \rceil$$

ou encore :

$$L = \sum_{j=1}^{n} \lceil 1+\log(1+|c_j|) \rceil + \sum_{i=1}^{m} \sum_{j=1}^{n} \lceil 1+\log(1+|a_{ij}|) \rceil + \sum_{i=1}^{m} \lceil 1+\log(1+|b_i|) \rceil$$

III.2. Complexité de la méthode du Simplexe

Elle dépend :

a. du nombre d'itérations,

b. du nombre d'opérations élémentaires nécessaires à chaque itération.

Chaque itération de la méthode du simplexe nécessite :

$$\begin{matrix} m(n-m)+(m+1)^2 \text{ multiplications} \\ + \\ m(n+1) \text{ additions} \end{matrix} \qquad (n \gg m)$$

---

1 Une configuration de variables

soit une complexité de l'ordre $O(m\text{-}n)$

Quel est le nombre d'itérations ? Cela correspond au nombre de sommets :

$$C_n^m = \frac{n!}{m!(n-m)!} \geq \left(\frac{n}{m}\right)^m \geq 2^m \text{ quand } n \geq 2m$$

Ceci est un résultat théorique ; en pratique on sait résoudre des problèmes à plusieurs millions de variables.

1970 : le russe Kachian a trouvé un algorithme polynomial en $O(n^6 L^2)$.

1984 : Karmarkar décrit l'algorithme des points intérieurs en $O(n^{3,5} L^2)$.

IV. Programmation semi-définie et problème de sac-à-dos quadratique

Un randonneur veut emporter des aliments dans un sac à dos de contenance limitée. Chaque type d'aliment a une valeur énergétique propre. Quels aliments doit emporter le randonneur pour maximiser l'énergie calorique de ses aliments ?

$$\begin{cases} \max \sum_{i=1}^{n} t_i x_i & \text{(maximiser l'énergie calorique des produits)} \\ \text{s.c.} \sum_{i=1}^{n} p_i x_i \le cst & \text{(capacité limitée du sac)} \\ \quad x_i \in \{0,1\} \end{cases}$$

Soit le problème du sac-à-dos « quadratique » :

$$(\text{QKP}^1) \quad \begin{cases} \max \sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij} \underbrace{x_i x_j}_{z} \\ \text{s.c.} \sum_{j=1}^{n} w_j x_j \le cst \\ \quad x_j \in \{0,1\}_{j=1,n} \rightarrow x_j \in [0,1] \rightarrow \underline{z} : \text{borne inférieure de (Qkp)} \end{cases}$$

où p_{ij} : revenu, utilité... w_j : investissement, poids... cst : budget, capacité...

IV.1. Exemple

Soit $P = [P_{ij}] = \begin{bmatrix} 2 & 0 & 9 \\ 0 & 5 & 3 \\ 9 & 3 & 20 \end{bmatrix}; \quad w^t = [2 \quad 3 \quad 2]; \quad cst = 5$

(QKP) s'écrit :

$$\begin{cases} \max 2x_1^2 + 18x_1x_3 + 5x_2^2 + 6x_2x_3 + 20x_3^2 = z \\ \text{s.c.}\ 2x_1 + 3x_2 + 2x_3 \le 5 \\ \quad x_1, x_2, x_3 \in \{0,1\} \end{cases}$$

Quelques méthodes de résolution :

a. relaxation linéaire

b. linéarisation des termes quadratiques

c. relaxation semi-définie positive

d. heuristiques (réservées à la recherche de la borne inférieure pour une fonction objectif max et de la borne supérieure pour une fonction objectif min)

1 *Quadratic Knapsack Problem* : Problème du Sac-à-Dos Quadratique

$\underbrace{\underline{z}}_{\text{borne inférieure}} \leq \underbrace{z^*}_{\text{solution optimale}} \leq \underbrace{\overline{z}}_{\text{borne supérieure}}$: il faut rechercher les bornes inférieure et supérieure à l'aide des méthodes de résolution ci-dessus.

IV.2. Linéarisation des termes quadratiques

Posons $y_{ij} = x_i y_j$ tel que $y_{ij} \in \{0,1\}$:

$$
(\text{PNE}^1) \begin{cases} z_{NE} = \max 2\,y_{1,1} + 18\,y_{1,3} + 5\,y_{2,2} + 6\,y_{2,3} + 20\,y_{3,3} \\ \quad 2\,x_1 + 3\,x_2 + 2\,x_3 \leq 5 \\ \quad x_1,\ x_2,\ x_3,\ y_{1,1},\ y_{2,2},\ y_{2,3},\ y_{3,3} \in \{0,1\} \\ \quad \text{mais il n'y a pas de relation entre les } x \text{ et les } y\text{, il faut donc ajouter} \\ \text{s.c.} \left[\begin{array}{l} y_{1,3} \leq x_1 \\ y_{1,3} \leq x_3 \\ x_1 + x_3 - 1 \leq y_{1,3} \end{array} \right. \quad \text{idem pour les autres } y_{ij\, i \neq j} \\ \quad \left[\begin{array}{l} y_{1,1} \leq x_1 \\ 2\,x_1 - 1 \leq y_{1,1} \end{array} \right. \quad \text{idem pour les autres } y_{ii} \end{cases}
$$

IV.3. Programmation Semi-définie Positive (PSD)

En PL, $x \geq 0$; en PSD, $X \succeq 0$ (au sens des matrices).

IV.3.1. Rappels

Une matrice X est semi-définie positive si et seulement si (ssi) :

- on peut lui appliquer la factorisation de Cholesky (définie positive) :
  $$X = L\,L^t \rightarrow L\,\underbrace{L^t\,z}_{x} = b \Rightarrow \begin{cases} L\,x = b \\ L^t\,z = x \end{cases}$$
- les valeurs propres de X $(\lambda_i)_{i=1,n} \geq 0$
- $\forall\, z \in \mathbb{R}^n,\ z^t\,X\,z \geq 0$

Reprenons l'exemple de (QKP) :

$$\sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij}\,x_i\,x_j = x^t\,P\,x = \underbrace{\text{tr}(x^t\,P\,x) = \text{tr}(P\,x\,x^t)}_{\text{Egalité autorisée par trace}}{}^{2}$$

1 Programme en Nombres Entiers

2 Trace de X : $\text{tr}(X) = \sum_{i=1}^{n} x_{ii}$ est un réel

Quelques propriétés de la trace :

- $\forall\, x \in \mathbb{R},\ \text{tr}(x) = x$
- $\forall\, A, B \in \mathbb{R}^{n*n},\ \text{tr}(A\,B) = \text{tr}(B\,A)$

Prenons en fait un autre exemple basé sur une matrice 2x2 plus simple à calculer :

Soient $P=\begin{bmatrix}2 & 1\\ 1 & 3\end{bmatrix}$ et $x=\begin{pmatrix}x_1\\ x_2\end{pmatrix}$

$$\Rightarrow x\,x^t=\begin{pmatrix}x_1\\ x_2\end{pmatrix}\begin{pmatrix}x_1 & x_2\end{pmatrix}=\begin{pmatrix}x_1^2 & x_1\,x_2\\ x_2\,x_1 & x_2^2\end{pmatrix}$$

$$\Rightarrow P\,x\,x^t=\begin{pmatrix}2\,x_1^2+x_2\,x_1 & 2\,x_1\,x_2+x_2^2\\ x_1^2+3\,x_2\,x_1 & x_1\,x_2+3\,x_2^2\end{pmatrix}$$

$$\Rightarrow \mathrm{tr}\,(P\,x\,x^t)=2\,x_1^2+2\;x_2\,x_1+3\;x_2^2$$

Si l'on conduit les mêmes calculs sur $x=\begin{pmatrix}x_1\\ x_2\\ x_3\end{pmatrix}$, on obtient :

$$X=x\,x^t=\begin{pmatrix}x_1^2 & x_1\,x_2 & x_1\,x_3\\ x_2\,x_1 & x_2^2 & x_2\,x_3\\ x_3\,x_1 & x_3\,x_2 & x_3^2\end{pmatrix}\geq 0$$

La fonction objectif du problème de sac-à-dos quadratique s'écrit :

avec $w=\begin{pmatrix}2\\ 3\\ 2\end{pmatrix}$, $W=\begin{pmatrix}2 & 0 & 0\\ 0 & 3 & 0\\ 0 & 0 & 2\end{pmatrix}$ et $\sum_{i=1}^{n} w_i\,x_i=\mathrm{tr}\,(W\,X)$

$$\begin{cases} \mathit{min}\,\mathrm{tr}\,(P\,X) \\ \text{s.c.}\ \begin{matrix} X\geq 0 \\ \mathrm{tr}\,(W\,X)\leq 5 \end{matrix} \end{cases}$$

La PSD permet de prendre en compte les variables en 0-1 :

$$\overline{X}=\left(\begin{array}{ccc|c} & & & x_1\\ & X & & x_2\\ & & & x_3\\ \hline x_1 & x_2 & x_3 & x_4\end{array}\right) \qquad \Delta_1=\begin{vmatrix}x_1^2 & x_1\\ x_1 & 1\end{vmatrix}\Rightarrow 0\leq x_1\leq 1$$

$x^2=x\,(\text{car }x\in\{0,\,1\})$ et $\mathrm{diag}\,(X)=x$ et $x_4=1$

On pose $W=\begin{pmatrix}w_1 & 0 & \cdots & 0\\ 0 & w_2 & \ddots & \vdots\\ \vdots & \ddots & w_3 & 0\\ 0 & \cdots & 0 & 0\end{pmatrix}=\begin{pmatrix}2 & 0 & 0 & 0\\ 0 & 3 & 0 & 0\\ 0 & 0 & 2 & 0\\ 0 & 0 & 0 & 0\end{pmatrix}$

$$\sum_{j=1}^{3} w_j\,x_j\leq 5 \Leftrightarrow \mathrm{tr}\,(W\,\overline{X})\leq 5 \qquad \text{avec} \qquad R=\begin{pmatrix}p & 0\\ 0 & 0\end{pmatrix}$$

Le programme SD (semidéfini) de (Qkp) s'écrit :

$$\begin{cases} \max \operatorname{tr}(R\overline{X}) \\ \text{s.c.} \begin{array}{l} \operatorname{tr}(W\overline{X}) \le 5 \\ \overline{X} \ge 0 \end{array} \end{cases}$$

Résultat du programme :

$$Z^*_{SDP} = 43{,}7 \ (Z^* = 43)$$

$$\overline{X} = \begin{pmatrix} 1 & 0{,}35 & 1 & 1 \\ 0{,}35 & 0{,}33 & 0{,}36 & 0{,}33 \\ 1 & 0{,}36 & 1 & 1 \\ 1 & 0{,}33 & 1 & 1 \end{pmatrix}$$

IV.4. Formulation SD du problème (Qkp)

Le PSD permet de formuler le problème (Qkp) de plusieurs manières. Afin de renforcer la borne inférieure, on peut transformer la contrainte du sac-à-dos de la manière suivante :

(F1) Formulation initiale : $\sum_j w_j x_j \le 5$ avec $\overline{X} = \begin{pmatrix} X & x \\ x^t & 1 \end{pmatrix}$

(F2) 1ère formulation : $\sum_{j=1}^{3} w_j x_j \le 5$

(F3) 2ème formulation : $\left(\sum_j w_j x_j - 5\right)^2 \le 0$

(F4) 3ème formulation : $\left(\sum_j w_j x_j\right)^2 \le 5^2$

(F5) 4ème formulation : $\left(\sum_j w_j x_j\right)^2 \le 5 \left(\sum_j w_j x_j - 5\right)$

(F6) 5ème formulation : $\begin{cases} x_i\left(\sum_{j=1}^{3} w_j x_j\right) \le 5 x_i \\ (1-x_i)\left(\sum_{j=1}^{3} w_j x_j\right) \le 5(1-x_i) \end{cases} \quad \forall i = 1, 2, 3$

V. Dualité et PSD

V.1. Exemple

Soit le PL primal suivant :

$$(\text{PLp}) \quad \begin{cases} \min 2x_1 + 3x_2 & (1) \\ \text{s.c.} \quad 4x_1 + 5x_2 \geq 6 & (2)\ (y) \\ \quad x_{1,}\ x_2 \geq 0 \end{cases}$$

Soit le dual de (PLp) :

$$(\text{PLd}) \quad \begin{cases} \max 6y \quad (3) \\ \quad 4y \leq 2 \quad (4) \\ \text{s.c.} \quad 5y \leq 3 \quad (5) \\ \quad y \geq 0 \end{cases}$$

Soient les matrices F_0, F_1 et F_2 telles que :

$$F_0 = \begin{pmatrix} 6 & 0 \\ 0 & 0 \end{pmatrix}; F_1 = \begin{pmatrix} 4 & 0 \\ 0 & 0 \end{pmatrix}; F_2 = \begin{pmatrix} 5 & 0 \\ 0 & 0 \end{pmatrix}$$

Soit la matrice Y définie par $Y = \begin{pmatrix} y & 0 \\ 0 & 0 \end{pmatrix}$, la relaxation SD primale de (PLp) s'écrit :

$$(\text{SDPp}) \quad \begin{cases} \min 2x_1 + 3x_2 \\ \text{s.c.} \quad x_1 F_1 + x_2 F_2 + F_0 \geq 0 \end{cases}$$

Le dual de (SDPp) s'écrit :

$$(\text{SDPd}) \quad \begin{cases} \max -\operatorname{tr}(F_0 Y) & \text{équivalent à la fonction object du PLd} \\ \quad \operatorname{tr}(F_1 Y) \leq 2 & \text{eq. à la contrainte 4} \\ \text{s.c.} \quad \operatorname{tr}(F_2 Y) \leq 3 & \text{eq. à la contrainte 5} \\ \quad Y \geq 0 \end{cases}$$

V.2. Dualité en PSD

On considère le problème de minimisation d'une fonction linéaire de $x \in \mathbb{R}^m$ sous contraintes d'inégalités matricielles (IML) :

$$\begin{cases} \min c^t x \\ \text{s.c.} \quad \mathrm{F}(x) \geq 0 \end{cases} \quad (1)$$

$$\text{où } \mathrm{F}(x) = -F_0 + \sum_{i=1}^{w} x_i F_i \quad (2)$$

Les données du problème sont le vecteur $c \in \mathbb{R}^m$ et $m+1$ matrices symétriques $F_{0,} \ldots, F_m \in \mathbb{R}^{n*m}$.

Le signe de l'inégalité $\mathrm{F}(x) \geq 0$ signifie que F(x) est SDP i.e. $z^t \mathrm{F}(x) z \geq 0, \ \forall z \in \mathbb{R}^n$. F($x$) est appelée Inégalité Matricielle Linéaire (IML) et (1) est appelé PSD. Un PSD est un problème d'optimisation convexe. On peut considérer le PSD (1) comme un programme linéaire semi-infini puisque l'inégalité matricielle $\mathrm{F}(x) \geq 0$ est équivalente à un ensemble infini de contraintes linéaires en x, i.e. $z^t \mathrm{F}(x) z \geq 0, \ \forall z \in \mathbb{R}^n$.

V.3. Exemple de programme non linéaire convexe

$$\begin{cases} \min \dfrac{(c^t x)^2}{d^t x} \quad (4) \\ \text{s.c. } Ax+b \geq 0 \end{cases}$$

avec $d^t x > 0$ quand $Ax+b \geq 0$

On introduit une variable auxiliaire t (borne supérieure pour la fonction objectif)

$$\begin{cases} \min t \\ \text{s.c. } \begin{matrix} Ax+b \geq 0 \\ \dfrac{(c^t x)^2}{d^t x} \leq t \end{matrix} \quad (5) \end{cases} \rightarrow \begin{cases} \min t \\ \text{s.c. } \begin{bmatrix} \mathrm{diag}(Ax+b) & 0 & 0 \\ & t & c^t x \\ & c^t x & d^t x \end{bmatrix} \geq 0 \end{cases}$$

Le dual de (1) s'écrit :

$$\begin{cases} \max -\mathrm{tr}(F_0 Z) \quad (6) \\ \text{s.c. } \begin{matrix} \mathrm{tr}(F_i Z) = c_i, \quad i=1, \ldots, m \\ Z \geq 0 \end{matrix} \end{cases}$$

Dans (6), on a $Z = Z^t \in \mathbb{R}^{n*n}$

Propriété

Le dual donne une borne au primal et *vice versa* :

$$c^t x + \mathrm{tr}(Z F_0) = \sum_{i=1}^{m} \mathrm{tr}(Z F_i) x_i + \mathrm{tr}(Z F_0) = \mathrm{tr}(Z F(x)) \geq 0$$

on déduit (aisément ;-) :

$$-\mathrm{tr}(F_0 Z) \leq c^t x$$

Le saut de dualité η associé à x et Z :

$\boxed{\eta \stackrel{\Delta}{=} c^t x + \mathrm{tr}(F_0 Z) = \mathrm{tr}(F(x) Z)}$ (où $\stackrel{\Delta}{=}$ signifie « s'exprime par »)

V.4. Formulation du problème primal-dual

$$\begin{cases} \min c^t x + \mathrm{tr}(F_0 Z) \\ \text{s.c. } \begin{matrix} F(x) \geq 0, \ Z \geq 0 \\ \mathrm{tr}(F_i Z) = c_i, i=1, \ldots, m \end{matrix} \end{cases}$$

Ce PSD peut être résolu à l'aide des méthodes de points intérieurs.

A. Annexe – Travaux Dirigés 1

Intervenant : M. BENAJAM (benajam@lri.fr)

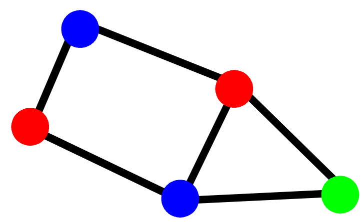
Exercice 1

Le nombre chromatique d'un graphe G=(X,U) est le nombre minimal de couleurs nécessaires pour colorier les sommets du graphe de telle façon que deux sommets adjacents ne soient jamais de la même couleur.
Montrer que le problème de la recherche du nombre chromatique d'un graphe G=(X,U) peut se modéliser comme un programme linéaire en nombres entiers.
Les couleurs que l'on peut utiliser étant numérotées de 0 à k où k est le degré maximal des degrés des sommets du graphe, on prendra comme inconnues x_i le numéro de la couleur associée au sommet i.

Réponses

Exemple :

nombre mini de couleur pour ce graphe : 3



Variables

Afin de simplifier le programme, ce ne sont pas celles proposées par l'énoncé.

$x_i^l \in \{0,1\}$ si le sommet i reçoit la couleur l
$y_l \in \{0,1\}$ si la couleur l est utilisée
$l \in \{1, \cdots, \Delta+1\}$
$i \in X$

avec Δ le maximal du nombre de points voisins de chacun des sommets. Ici $\Delta=4$.

Fonction objectif

$$min \sum_{l=1}^{\Delta+1} y_l$$

Contraintes

- $\forall (i,j) \in U, \forall l \in \{1, \dots, \Delta+1\}$: $x_i^l + x_j^l \leq 1$ (2 sommets adjacents n'ont pas la même couleur)
- $\forall i \in X$: $\sum_{l=1}^{\Delta+1} x_i^l = 1$ (tous les points possèdent une couleur)
- $\forall i \in X, l \in [1 \dots \Delta+1]$: $x_i^l \leq y_l$ (si un sommet a une couleur, alors cette couleur a été sélectionnée)
- $x_i^l \in \{0,1\}$ et $y_l \in \{0,1\}$

Pour faire l'exercice avec les variables de l'énoncé :

Contraintes

$x_i \neq x_j$ (écriture qui doit être transformée)
$\Rightarrow x_i < x_j$ ou $x_i > x_j$

Exercice 2

Soit le graphe G=(X,A) de la figure 1. X correspond à l'ensemble des nœuds et A l'ensemble des arcs. Les valeurs indiquées sur les arcs *ij* correspondent aux coût du routage.

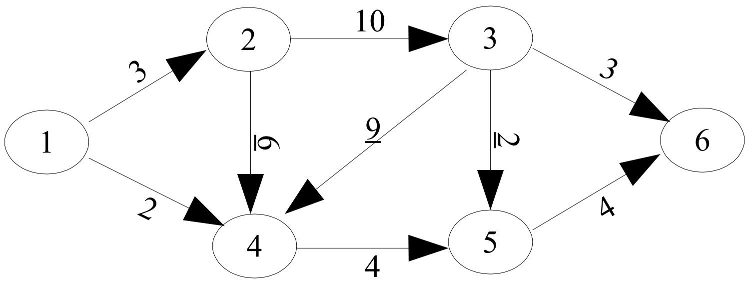


Figure 1

1. Écrire le programme linéaire PL1 correspondant au problème du plus court chemin du nœud 1 au nœud 6.
2. Déduire de PL1, le programme linéaire PL2 correspondant au problème du plus court chemin du sommet 1à tous les autres nœuds.

Les contraintes du problème du plus court chemin doivent respecter la conservation des flots dans les nœuds et la positivité des variables de flots.

1.

Algorithme existant et utilisable pour résoudre ce problème : *Dijkstra*[1] de complexité polynomiale.

Variables

$x_{ij} \in \{0,1\}$ (si un arc *ij* appartient au chemin)

Fonction objectif

$\min z$ avec

$$z = \sum_{i=1}^{6} \sum_{j=1}^{6} \underbrace{c_{ij}}_{\text{coût}} x_{ij} = 3x_{1,2} + 2x_{1,4} + 10x_{2,3} + 6x_{2,4} + 3x_{3,6} + 2x_{3,5} + 9x_{3,4} + 4x_{4,5} + 4x_{5,6}$$

Contraintes

Par convention, on donne un poids positif aux arcs sortant d'un sommet et négatif pour ceux y entrants.

1. $x_{1,2} + x_{1,4} = 1$ (0 entrant, 1 sortant : départ)

1 Edsger Wybe Dijkstra (1930-2002) : a donné son nom à un algorithme qui résout un problème du plus court chemin pour un graphe (http://fr.wikipedia.org/wiki/Algorithme_de_Dijkstra)

2. $-x_{1,2}+(x_{2,3}+x_{2,4})=0$ (1 entrant, 2 sortants)
3. $-x_{2,3}+(x_{3,4}+x_{3,5}+x_{3,6})=0$ (1 entrant, 3 sortants)
4. $-x_{1,4}-x_{2,4}-x_{3,4}+x_{4,5}=0$ (3 entrant, 1 sortant)
5. $-x_{4,5}-x_{3,5}+x_{5,6}=0$ (2 entrants, 1 sortant)
6. $-x_{3,6}-x_{5,6}=-1$ (1 entrant, 0 sortant : arrivée)

Domaine

$x_{ij}\in\{0,1\}$ pour $(i,j)\in A$

2.

On garde la même fonction objectif, mais les contraintes changent.

Contraintes

Maintenant, le programme devant permettre de trouver les 5 chemins du sommet 1 vers chacun des autres sommets, le départ doit comporter 5 arcs partants, les autres sommets étant chacun l'arrivée d'un chemin, ils ont tous un arc de plus en entrée qu'en sortie (d'où le total à -1) :

1. $x_{1,2}+x_{1,4}=5$
2. $-x_{1,2}+(x_{2,3}+x_{2,4})=-1$
3. $-x_{2,3}+(x_{3,4}+x_{3,5}+x_{3,6})=-1$
4. $-x_{1,4}-x_{2,4}-x_{3,4}+x_{4,5}=-1$
5. $-x_{4,5}-x_{3,5}+x_{5,6}=-1$
6. $-x_{3,6}-x_{5,6}=-1$

Domaine

$x_{ij}\in\{0,1\}$ pour $(i,j)\in A$

Exercice 3

Un étudiant suit à la faculté d'Orsay les cinq cours suivants : Algèbre, Analyse, Physique, Chimie minérale et Chimie organique. En raison du nombre élevé d'étudiants qui suivent ces cours, quatre groupes de TD ont été organisés dans chaque discipline. Les horaires de ces groupes sont donnés dans le tableau suivant :

	Algèbre	Analyse	Physique	Chimie min.	Chimie org.
Groupe 1	Lundi 13 à 15	Lundi 13 à 15	Mardi 8 à 11	Lundi 8 à 10	Lundi 9 à 10:30
Groupe 2	Mardi 10 à 12	Mardi 10 à 12	Mardi 10 à 13	Lundi 8 à 10	Lundi 10:30 à 12
Groupe 3	Mercredi 10 à 12	Mercredi 11 à 13	Jeudi 15 à 18	Jeudi 13 à 15	Vendredi 11 à 12:30
Groupe 4	Mercredi 11 à 13	Jeudi 8 à 10	Jeudi 17 à 20	Vendredi 13 à 15	Vendredi 13 à 14:30

Tableau 1

Dans chaque discipline, cet étudiant a exprimé sa préférence pour les différents groupes en attribuant à chacun d'eux une note entre 0 et 10. Cette note tient compte de l'horaire, et de la réputation de l'enseignant. Ces préférences sont exprimées dans le tableau 2 ci-dessous :

	Algèbre	Analyse	Physique	Chimie min.	Chimie org.
Groupe 1	5	4	3	10	0
Groupe 2	4	4	5	10	5
Groupe 3	10	5	7	7	3
Groupe 4	5	6	8	5	4

Tableau 2

L'étudiant désire s'inscrire dans les cinq groupes de TD de façon à maximiser la somme de ses préférences. Il désire, en outre, respecter les trois contraintes suivantes :

- ✓ **Ne pas s'inscrire à plus de quatre heures de TD par jour ;**
- ✓ **Disposer chaque jour d'une heure libre située entre 12h et 14h (pause déjeuner) ;**
- ✓ **Pouvoir pratiquer, au moins une fois par semaine, son sport favori qui a lieu le lundi de 13h à 15h, le mercredi de 11h à 13h et le mercredi de 13h à 15h.**

1. **Ecrire le programme mathématique correspond à ce problème d'emploi du temps.**
2. **Existe –t- il un emploi du temps dans lequel les TD sont groupés sur les trois jours, lundi, mardi et jeudi et ou ils correspondent tous à une préférence supérieure ou égale à cinq ?**

1.

Variables

A chaque groupe de chaque matière, on associe une variable x_i telle que $x_i = 1$ si le groupe est retenu par l'étudiant, 0 sinon. Ex. : x_1 pour Algèbre en Groupe 1, x_2 pour Algèbre en Groupe 2, jusqu'à x_{19} pour Chimie Orga. en Groupe 3 et x_{20} pour Chimie Orga. en Groupe 4. Pour le

Sport, viennent s'ajouter x_{21} pour le lundi de 13h à 15h, x_{22} pour le mercredi de 11h à 13h et x_{23} pour le mercredi de 13h à 15h.

Fonction Objectif

$\sum_{i=1}^{20} x_i p_i$ où p_i indique la préférence de l'étudiant (cf. Tableau 2). (Il n'y a pas de coefficient de préférence pour le sport)

Contraintes

a) Un seul groupe sélectionné par matière :

- Algèbre : $x_1 + x_2 + x_3 + x_4 = 1$
- Analyse : $x_5 + x_6 + x_7 + x_8 = 1$
- Physique : $x_9 + x_{10} + x_{11} + x_{12} = 1$
- Chimie min. : $x_{13} + x_{14} + x_{15} + x_{16} = 1$
- Chimie orga. : $x_{17} + x_{18} + x_{19} + x_{20} = 1$

Sport au moins une fois par semaine :
$x_{21} + x_{22} + x_{23} \geq 1$

b) 4h de TD max. par jour (donc hors sport) :

- lundi : $2x_1 + 2x_5 + 2(x_{13} + x_{14}) + 1{,}5(x_{17} + x_{18}) \leq 4$
- mardi : $2x_2 + 2x_6 + 3(x_9 + x_{10}) \leq 4$
- mercredi : $2\ (x_3 + x_4) + 2x_7 \leq 4$
- jeudi : $2x_8 + 3(x_{11} + x_{12}) + 2x_{15} \leq 4$
- vendredi : $2x_{16} + 1{,}5(x_{19} + x_{20}) \leq 4$

c) Une heure de pause minimum entre 12h et 14h :

- lundi : pas de problème car l'énoncé respecte la contrainte et ne nécessite pas d'équation.
- mardi : idem
- mercredi :

  $x_4 + x_{23} \leq 1$ (Algèbre ou Sport)

  $x_7 + x_{23} \leq 1$ (Analyse ou Sport)

  $x_{22} + x_{23} \leq 1$ (2 fois Sport supprime la pause)
- jeudi : pas de problème
- vendredi :

  $x_{16} + x_{19} \leq 1$ (Chimie min. ou orga.)

  $x_{19} + x_{20} \leq 1$ (contrainte redondante avec le point a)

d) L'étudiant ne peut être présent à plusieurs cours en même temps (don d'ubiquité)

- lundi :

$x_1 + x_5 + x_{21} \le 1$

$\underbrace{x_{13} + x_{14}}_{\text{redondante}} + x_{17} \le 1$, peut aussi s'écrire avec les 2 inéquations : $\begin{cases} x_{13} + x_{17} \le 1 \\ x_{14} + x_{17} \le 1 \end{cases}$

- mardi :

$x_2 + x_6 + \underbrace{x_9 + x_{10}}_{\text{redondante}} \le 1$

$x_9 + x_{10} \le 1$, mais là c'est tellement redondant que c'est inutile

- mercredi : $\underbrace{x_3 + x_4}_{\text{redondant}} + x_7 + x_{22} \le 1$
- jeudi : $x_{11} + x_{12} \le 1$, redondant donc inutile
- vendredi :

$x_{16} + x_{20} \le 1$

$x_{19} + x_{20} \le 1$ inutile

Pour résumer, le PL s'écrit donc :

$$\begin{cases} \max \sum_{i=1}^{20} x_i p_i \\ \quad \text{a)} \\ \quad \text{b)} \\ \text{s.c. c)} \\ \quad \text{d)} \\ \quad \forall x_i \in \{0,\ 1\}, i = \{1,\ \ldots,\ 23\} \end{cases}$$

2.

On reprend le programme pour lequel on fixe des variables en fonction des nouvelles contraintes :

a) pas de TD mercredi :

$x_3 = 0\ ;\ x_4 = 0\ ;\ x_7 = 0$

et vendredi :

$x_{16} = 0\ ;\ x_{19} = 0\ ;\ x_{20} = 0$

b) Uniquement les TD dont la préférence est ou dépasse 5 :

$x_2 = 0\ ;\ x_5 = x_6 = 0\ ;\ x_9 = 0\ ;\ x_{17} = x_{19} = x_{20} = 0$

On essaie ensuite de trouver les variables restantes :

$$(a)\quad \begin{cases} x_1 = 1 \ ; x_8 = 1 \ ; x_{18} = 1 \\ x_{10} + x_{11} + x_{12} = 1 \Rightarrow x_{10} = 1 \ \text{ d'après (b)} \\ x_{13} + x_{14} + x_{15} = 1 \Rightarrow x_{15} = 1 \ \text{ d'après (b)} \end{cases}$$

$$(b)\quad \begin{cases} 2 + 2(x_{13} + x_{14}) + 1{,}5 \le 4 \Leftrightarrow 2(x_{13} + x_{14}) \le 0{,}5 \Leftrightarrow x_{13} + x_{14} \le 0{,}25 \Rightarrow x_{13} = x_{14} = 0 \\ 3 \ x_{10} \le 4 \Rightarrow x_{10} \le 4/3 \Rightarrow x_{10} = 1 \ \text{ d'après (a)} \\ 2 + 3(x_{11} + x_{12}) + 2 \le 4 \Leftrightarrow 3(x_{11} + x_{12}) \le 0 \Rightarrow x_{11} = x_{12} = 0 \end{cases}$$

Toutes les variables des TD ont une (et une seule) valeur. Il n'y a donc qu'une unique solution qui satisfasse ces contraintes : $x_1 = x_8 = x_{10} = x_{15} = 1$, les variables des TD restantes sont nulles.

Reste le cas du sport pour lequel il reste les inégalités suivantes :

$$x_{23} \le 1 \ ; x_{22} \le 1$$
$$x_{21} + x_{22} + x_{23} \le 1$$
$$x_{22} + x_{23} \le 1$$

Exercice 4

Un administrateur d'un hôpital doit établir l'emploi du temps des infirmières satisfaisant les contraintes données par le tableau 1. Les tours de garde de jour durent huit heures consécutives et ceux de nuits durent quatre heures. L'administrateur cherche à trouver le nombre minimum d'infirmières à embaucher. Formuler ce problème sous forme d'un programme linéaire.

	1	2	3	4	5	6
Intervalle de temps	6 à 10	10 à 14	14 à 18	18 à 22	22 à 2	2 à 6
Nombre minimum d'infirmières	70	80	50	60	40	30

Tableau 1

Réponses

Sûrement perturbé par les infirmières de l'énoncé ou par la température glaciale de l'amphi ce matin là, nous avons hésité dans le choix des variables, pour finalement proposer cette version.

Variables

x_i indique le nombre d'infirmières par intervalle de 8 heures en journée et de 4 heures de nuit. On a donc :

- x_1 de 6h à 14h (8 heures),
- x_2 de 10h à 18h (8 heures),
- x_3 de 14h à 22h (8 heures),
- x_4 de 22h à 2h (4 heures),

- x_5 de 2h à 6h (4 heures).

Fonction objectif

$$\min \sum_{i=1}^{5} x_i$$

Contraintes

$x_1 \geq 70$; $x_1 + x_2 \geq 80$; $x_2 + x_3 \geq 50$; $x_3 \geq 60$; $x_4 \geq 40$; $x_5 \geq 30$

$x_i \geq 0$ pour $i = 1, 2, \dots, 5$

VII. ANNEXE – TRAVAUX DIRIGÉS 2

Exercice 1

Soit le programme quadratique sans contraintes suivant :

$$\min f(x) = x_1^2 + 4\ x_2^2 + 4x_1x_2 \quad x_{1,}\ x_2 \in \{0,\ 1\} \quad \text{(PQuad)}$$

1. Donner le programme quadratique relaxé de (PQuad).
2. Après linéarisation des termes quadratiques, formuler le problème en terme de programme linéaire.
3. Écrire la relaxation semidéfinie positive de (PQuad). On utilisera la notion de trace de matrice pour formuler le programme dual SDP.

1.

$$(\text{PQuad}_{\text{relaxé}}) \quad \begin{cases} \min f(x) = x_1^2 + 4x_2^2 + 4x_1x_2 \\ \text{s.c.}\ \begin{array}{l} 0 \le x_1 \le 1 \\ 0 \le x_2 \le 1 \end{array} \end{cases}$$

2.

On effectue un changement de variables afin de linéariser (PQuad). Posons $y_{ij} = x_i x_j$. Il vient les contraintes suivantes (c'est aussi la table de vérité du ET booléen) :

$$\begin{array}{l} y_{ij} \le x_i \\ y_{ij} \le x_j \\ x_i + x_j - y_{ij} \le 1 \\ 0 \le y_{ij} \le 1 \end{array} \quad \text{d'où le cas particulier si } i=j : \quad \begin{array}{l} y_{ii} \le x_i \\ 2x_i - y_{ii} \le 1 \\ 0 \le y_{ii} \le 1 \end{array}$$

(PQuad$_{\text{relaxé}}$) linéarisé devient alors :

$$(\text{PLin}_{\text{relaxé}}) \quad \begin{cases} \min y_{11} + 4y_{22} + 4y_{12} \\ \text{s.c.}\ \begin{array}{lll} y_{11} \le x_1 & y_{12} \le x_1 & 0 \le x_1 \le 1 \\ 2x_1 - y_{11} \le 1 & y_{12} \le x_2 & 0 \le x_2 \le 1 \\ y_{22} \le x_2 & x_1 + x_2 - y_{12} \le 1 & 0 \le y_{11} \le 1 \\ 2x_2 - y_{22} \le 1 & & 0 \le y_{22} \le 1 \\ & & 0 \le y_{12} \le 1 \end{array} \end{cases}$$

Note : le programme (PLin$_{\text{relaxé}}$) donnera une solution inférieure à celle de (PQuad) ; c'est la borne inférieure.

Exercice 2

Considérons le programme quadratique suivant :

$$\begin{cases} \min f(x) = 9x_1^2 + 4x_2^2 - 18x_1x_2 \\ \text{s/c} \quad 3x_1 + 2x_2 \leq 6 \\ \qquad x_{1,} \; x_2 \in \{0,1\} \end{cases} \qquad \text{(PQuad2)}$$

1. Après linéarisation des termes quadratiques, formuler le problème en terme de programme linéaire.
2. Écrire la relaxation semidéfinie positive de (PQuad2). On utilisera la notion de trace de matrice pour formuler le programme dual SDP.
3. Pour chacune des transformations suivantes de la contrainte linéaire, écrire la nouvelle relaxation SDP :
   a) Remplacer x par x^2,
   b) Élever au carré les deux membres de la contrainte,
   c) Multiplier les deux membres par $3x_1 + 2x_2$,
   d) Remplacer la contrainte par $6 - 3x_1 + 2x_2 \geq 0$ et l'élever au carré,
   e) Multiplier la contrainte respectivement par x_1, x_2, $(1 - x_1)$ et $(1 - x_2)$
4. Comparer les différentes relaxations.

1.

Rappel : pour linéariser un problème discret, il faut appliquer le changement de variables $y_{ij} = x_i x_j$ et ajouter les contraintes suivantes :

$$\begin{aligned} & y_{ij} \leq x_i \\ & y_{ij} \leq x_j \\ & x_i + x_j - y_{ij} \leq 1 \\ & y_{ij} \in \{0,1\} \end{aligned}$$

Ici, si l'on pose $y_{11} = x_1^2$, $y_{22} = x_2^2$ et $y_{12} = x_1 \, x_2$ on obtient ici le PL suivant :

$$\begin{cases} \min 9y_{11} + 4y_{22} - 18y_{12} \\ \qquad 3x_1 + 2x_2 \leq 6 \\ \qquad x_1, x_2 \in \{0,1\} \\ \text{s.c.} \quad y_{11} \leq x_1 \; ; 2x_1 - y_{11} \leq 1 \\ \qquad y_{22} \leq x_2 \; ; 2x_2 - y_{22} \leq 1 \\ \qquad y_{12} \leq x_1 \; ; y_{12} \leq x_2 \; ; x_1 + x_2 - y_{12} \leq 1 \\ \qquad y_{11}, y_{12}, y_{22} \in \{0,1\} \end{cases}$$

TP 1

Dans le cadre du 1er TP, nous avons utilisé CPlex[1] (`c:\ilog\cplex81\bin\msvc6`). Ce programme permet la résolution des :

- PLC : Problèmes Linéaires Continus ($0 \leq x \leq 1$ ou $x > 0$;
- PLE : Problèmes Linéaires Entiers (valeurs discrètes des variables) ;
- PQC : Problèmes Quadratiques Continus ;
- PQE : Problèmes Quadratiques Entiers.

Pour notre exercice, on rédigera un fichier `PLE.pl` ainsi :

```
Minimize
obj: 9 y11 + 4 y22 - 18 y12

Subject To (toutes les variables doivent être dans le membre de gauche de l'inégalité)
c1:  y11 - x1 <= 0
c2:  2 x1 - y11 <= 1
c3:  y22 - x2 <= 0
c4:  2 x2 - y22 <= 1
c5:  y12 - x1 <= 0
c6:  y12 - x2 <= 0
c7:  x1 + x2 - y12 <= 1
c8:  3 x1 + 2 x2 <= 6

Binaries
x1
x2
y11
y22
y12
```

On peut demander la résolution du problème[2] :

```
CPLEX> read PLE.lp
Problem 'PLE.lp' read.
Read time =    0.03 sec.
CPLEX> optimize
Tried aggregator 1 time.
MIP Presolve eliminated 3 rows and 0 columns.
MIP Presolve modified 4 coefficients.
Aggregator did 2 substitutions.
Reduced MIP has 3 rows, 3 columns, and 7 nonzeros.
Presolve time =    0.04 sec.
Clique table members: 2
MIP emphasis: balance optimality and feasibility
Root relaxation solution time =    0.00 sec.

Integer optimal solution:  Objective =  -5.0000000000e+000
Solution time =    0.10 sec.  Iterations = 2  Nodes = 0
```

1 Une présentation rapide du produit est disponible à http://www.ilog.com/products/optimization/presentations/cplexintro.pdf en anglais

2 Merci Laurent pour les copies d'écran.

Et CPlex nous la donne ! La solution optimale est trouvée avec une fonction objectif à -5. Reste à connaître les valeurs des variables :

```
CPLEX> display solution variables -
Variable Name           Solution Value
y11                           1.000000
y22                           1.000000
y12                           1.000000
x1                            1.000000
x2                            1.000000
```

Remarque : notre problème possédant peu de variables, sa résolution par cette méthode est possible. La complexité de la résolution étant exponentielle, elle ne peut pas être utilisée au-delà d'une centaine de variables. Dans ce cas, il faudra commencer par transformer ce problème linéaire entier en problème linéaire continu en relaxant ses variables :

$$\begin{cases} \min 9y_{11} + 4y_{22} - 18y_{12} \\ \quad 3x_1 + 2x_2 \leq 6 \\ \quad 0 \leq x_1 \leq 1,\ 0 \leq x_2 \leq 1 \\ \text{s.c.}\ \ y_{11} \leq x_1\ ;\ 2x_1 - y_{11} \leq 1 \\ \quad y_{22} \leq x_2\ ;\ 2x_2 - y_{22} \leq 1 \\ \quad y_{12} \leq x_1\ ;\ y_{12} \leq x_2\ ;\ x_1 + x_2 - y_{12} \leq 1 \\ \quad 0 \leq y_{11} \leq 1\ ;\ 0 \leq y_{12} \leq 1\ ;\ 0 \leq y_{22} \leq 1 \end{cases}$$

Cela se traduit par le fichier PLC.pl suivant :

```
Minimize
obj: 9 y11 + 4 y22 - 18 y12

Subject To
c1:  y11 - x1 <= 0
c2:  2 x1 - y11 <= 1
c3:  y22 - x2 <= 0
c4:  2 x2 - y22 <= 1
c5:  y12 - x1 <= 0
c6:  y12 - x2 <= 0
c7:  x1 + x2 - y12 <= 1
c8:  3 x1 + 2 x2 <= 6

Bounds
0 <= x1 <= 1
0 <= x2 <= 1
0 <= y11 <= 1
0 <= y12 <= 1
0 <= y22 <= 1
```

Et donne le résultat suivant :

```
CPLEX> read PLC.lp
Problem 'PLC.lp' read.
Read time =    0.03 sec.
CPLEX> optimize
```

```
Tried aggregator 1 time.
LP Presolve eliminated 1 rows and 0 columns.
Reduced LP has 7 rows, 5 columns, and 15 nonzeros.
Presolve time =    0.02 sec.

Iteration log . . .
Iteration:     1   Dual objective     =          -18.000000

Dual simplex - Optimal:  Objective =  -9.0000000000e+000
Solution time =    0.05 sec.  Iterations = 3 (0)
```

Et les variables ont les valeurs :

```
CPLEX> display solution variables -
Variable Name           Solution Value
y12                           0.500000
x1                            0.500000
x2                            0.500000
All other variables in the range 1-5 are zero.
```

La fonction objectif a pour valeur -9 et les variables x_1, x_2 et y_{12} valent 0,5, les autres étant nulles. On a ici a faire à une borne inférieure de la solution ($\underline{z}$) qui se trouve à 45% de l'optimum. Cet écart est très mauvais dans le cas d'une application réelle et rend cette borne inférieure peut significative.

Essayons maintenant de faire résoudre notre problème initial, c'est à dire Quadratique et Entier, sachant toujours que cela ne peut fonctionner que pour un petit nombre de variables.

La rédaction du fichier est un peu différente (`PQE.pl`) :

```
Minimize
obj: [ 9 x1 ^ 2 + 4 x2 ^2 - 18 x1 * x2 ] / 2

Subject To:
c1: 3 x1 +2 x2 <= 6

Binaries
x1
x2
```

La fonction objectif s'écrit entre crochets et **doit** être suivi par la division par deux car le système s'attend à résoudre un problème de la forme :

$$\begin{cases} \min \frac{1}{2} x^t Q\, x \\ \text{s.c} \quad \begin{array}{l} Ax \leq b \\ x_i \in \{0,1\} \end{array} \end{cases}$$

Et le résultat est celui attendu ;-) :

```
CPLEX> read PQE.lp
Problem 'PQE.lp' read.
Read time =    0.03 sec.
CPLEX> optimize
Number of nonzeros in lower triangle of Q = 1
```

```
Using Approximate Minimum Degree ordering
Total time for automatic ordering = 0.01 sec.
Summary statistics for factor of Q:
  Rows in Factor            = 2
  Integer space required    = 2
  Total non-zeros in factor = 3
  Total FP ops to factor    = 5
Tried aggregator 1 time.
MIQP Presolve eliminated 1 rows and 0 columns.
MIQP Presolve modified 1 coefficients.
Reduced MIQP has 0 rows, 2 columns, and 0 nonzeros.
Presolve time =    0.11 sec.
Dual steepest-edge pricing selected.
MIP emphasis: balance optimality and feasibility
Root relaxation solution time =    0.00 sec.

Integer optimal solution:  Objective =  -2.5000000000e+000
Solution time =    0.15 sec.  Iterations = 1  Nodes = 0
CPLEX> display solution variables -
Variable Name           Solution Value
x1                            1.000000
x2                            1.000000
```

Évidemment la fonction objectif est inférieure de moitié en valeur absolue à la valeur attendue. Pour éviter cela il aurait fallut doubler les coefficients de l'équation objectif.

Dernier test, vérifier le fonctionnement de CPlex dans le cas du Problème Quadratique Continu. En effet CPlex attend un problème de la forme :

$$\begin{cases} \min \frac{1}{2} x^t Q\, x \\ \text{s.c} \begin{array}{l} Ax \leq b \\ 0 \leq x_i \leq 1 \end{array} \end{cases}$$

mais seulement si Q est semidéfinie positive, c'est à dire ssi $\forall z, z^t Q z \geq 0$. Essayons donc de résoudre notre problème après sa relaxation, ce qui nous donnera une borne inférieure à la solution optimale.

Voici le fichier `PQC.pl`) :

```
Minimize
obj: [ 9 x1 ^ 2 + 4 x2 ^2 - 18 x1 * x2 ] / 2

Subject To:
c1: 3 x1 +2 x2 <= 6

Bounds
0 <= x1 <= 1
0 <= x2 <= 1
```

Et voilà le résultat :

```
CPLEX> read PQC.lp
Problem 'PQC.lp' read.
```

```
Read time =    0.01 sec.
CPLEX> optimize
Number of nonzeros in lower triangle of Q = 1
Using Approximate Minimum Degree ordering
Total time for automatic ordering = 0.02 sec.
Summary statistics for factor of Q:
  Rows in Factor            = 2
  Integer space required    = 2
  Total non-zeros in factor = 3
  Total FP ops to factor    = 5
Presolve time =    0.07 sec.
CPLEX Error  5002: Q is not positive semi-definite.
```

On ne peut pas trouver de solution à ce problème par cette méthode. Mais un peu de patience car le second TP commence ici et nous allons utiliser un autre programme permettant de résoudre ce problème.

TP 3

Pour ce second TP, nous utiliserons un logiciel destiné à la programmation semidéfinie (DSDP4[1]) sur machines LINUX pour résoudre la suite de l'exercice 2.

Commençons par quelques rappels.

En programmation linéaire, les primals et les duals sont de la forme :

(Primal) $\begin{cases} \min c^t x \\ \text{s.c.} \quad Ax \leq b \end{cases}$ (Dual) $\begin{cases} \max b^t y \\ \text{s.c.} \quad A^t y = c \end{cases}$

Alors que pour les problèmes semidéfinis, les primals et les duals sont de la forme :

(Primal) $\begin{cases} \min \operatorname{tr}(A_0\, y) \\ \text{s.c.} \quad \begin{array}{l} \operatorname{tr}(A_i\, y) = b_i \quad \forall i = 1, \dots, m \\ y \geq 0 \end{array} \end{cases}$ (Dual) $\begin{cases} \max b^t x \\ \text{s.c.} \quad A_1 x_1 + \cdots + A_m x_m + A_0 \geq 0 \end{cases}$

Il nous faut donc traduire l'énoncé du programme (PQuad2) pour obtenir un programme SD de la forme de (Primal) :

$$\begin{cases} \min f(x) = 9x_1^2 + 4x_2^2 - 18x_1x_2 \\ \text{s/c} \quad \begin{array}{l} 3x_1 + 2x_2 \leq 6 \\ x_1, x_2 \in \{0,1\} \end{array} \end{cases} \quad \text{(PQuad2)}$$

Posons $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ où x_1 et x_2 sont les variables de (PQuad2). On a alors $X = xx^t = \begin{pmatrix} x_1^2 & x_1x_2 \\ x_2x_1 & x_2^2 \end{pmatrix}$.

Avec $x_{ij} = x_i x_j$, on a $X = \begin{pmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{pmatrix}$.

Posons aussi $y = \begin{pmatrix} X & x \\ x^t & 1 \end{pmatrix}$, on a donc $y = \begin{pmatrix} x_{11} & x_{12} & x_1 \\ x_{12} & x_{22} & x_2 \\ x_1 & x_2 & 1 \end{pmatrix}$. Or on veut $y \geq 0$ (semidéfinie

1 Voici le site d'un des développeurs : http://www-unix.mcs.anl.gov/~benson/dsdp/

positive) ; ce qui est équivalent à dire « déterminant des sous-matrices symétriques ≥ 0 ». D'où les inéquations :

$$y \geq 0 \Leftrightarrow \begin{cases} x_{11} \geq 0 \Rightarrow x_1^2 \geq 0 \Rightarrow x_1 \geq 0 \\ x_{22} \geq 0 \Rightarrow x_2 \geq 0 \\ 1 \geq 0 \\ \det\begin{pmatrix} x_{11} & x_1 \\ x_1 & 1 \end{pmatrix} \geq 0 \Leftrightarrow x_{11} - x_1^2 \geq 0 \Leftrightarrow x_{11} \geq x_1^2 \Leftrightarrow x_1 \leq 1 \quad (???) \\ \det\begin{pmatrix} x_{22} & x_2 \\ x_2 & 1 \end{pmatrix} \geq 0 \Leftrightarrow x_2 \leq 1 \end{cases}$$

$$\Leftrightarrow \begin{cases} 0 \leq x_1 \leq 1 \\ 0 \leq x_2 \leq 1 \end{cases}$$

Choisissons maintenant les coefficients de $A_0 = \begin{pmatrix} 9 & -9 & 0 \\ -9 & 4 & 0 \\ 0 & 0 & 0 \end{pmatrix}$, et vérifions que $\mathrm{tr}(A_0\, y) = 9x_1^2 + 4x_2^2 - 18\, x_1 x_2$:

$$\mathrm{tr}(A_0\, y) = \mathrm{tr}\left(\begin{bmatrix} 9 & -9 & 0 \\ -9 & 4 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} & x_1 \\ x_{12} & x_{22} & x_2 \\ x_1 & x_2 & 1 \end{bmatrix} \right) = \mathrm{tr}\begin{pmatrix} 9x_{11} - 9x_{12} & & \\ & -9x_{12} + 4x_{22} & \\ & & 0 \end{pmatrix}$$

$$\mathrm{tr}(A_0\, y) = 9x_{11} + 4x_{22} - 18\, x_{12}$$

Pour l'inéquation $3x_1 + 2x_2 \leq 6$, nous devons commencer par la traduire en une équation de la forme : $3x_1 + 2x_2 + z = 6$ avec $z \geq 0$, ce qui nous oblige à rajouter cette variable z dans la matrice y sous la forme, par exemple, d'un élément sur la diagonale :

$$y = \begin{pmatrix} x_{11} & x_{12} & x_1 & 0 \\ x_{12} & x_{22} & x_2 & 0 \\ x_1 & x_2 & 1 & 0 \\ 0 & 0 & 0 & z \end{pmatrix}$$

Ainsi, il faut maintenant modifier A_0 et sa trace qui deviendra :

$$\mathrm{tr}(A_0\, y) = \mathrm{tr}\left(\begin{bmatrix} 9 & -9 & 0 & 0 \\ -9 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} y \right)$$

L'inéquation transformée en équation aura la trace :

$$\mathrm{tr}\left(\begin{bmatrix} 0 & 0 & 1{,}5 & 0 \\ 0 & 0 & 1 & 0 \\ 1{,}5 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} y \right) = 6$$

A noter que la constante peut être placée dans la matrice, mais attention au signe :
$3x_1 + 2x_2 + z = 6 \Leftrightarrow 3x_1 + 2x_2 + z - 6 = 0$, d'où la trace :

$$\mathrm{tr}\left(\begin{bmatrix} 0 & 0 & 1{,}5 & 0 \\ 0 & 0 & 1 & 0 \\ 1{,}5 & 1 & -6 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} y\right) = 0$$

Le résultat sera le même.

Pour $x_{11} = x_1 \Leftrightarrow x_{11} - x_1 = 0$ (à noter que toutes les variables doivent être à gauche pour se trouver dans la trace des matrices), on a :

$$\mathrm{tr}\left(\begin{bmatrix} 1 & 0 & -0{,}5 & 0 \\ 0 & 0 & 0 & 0 \\ -0{,}5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} y\right) = 0$$

De même pour $x_{22} = x_2 \Leftrightarrow x_{22} - x_2 = 0$, on a :

$$\mathrm{tr}\left(\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & -0{,}5 & 0 \\ 0 & -0{,}5 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} y\right) = 0$$

Enfin il faut rajouter des contraintes pour les valeurs constantes de la matrice y.

Pour *cst* = 1 (???) :

$$\mathrm{tr}\left(\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} y\right) = 1$$

De même pour les valeurs nulles liées à l'ajout de z :

$$\mathrm{tr}\left(\begin{bmatrix} 0 & 0 & 0 & 0{,}5 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0{,}5 & 0 & 0 & 0 \end{bmatrix} y\right) = 0$$

$$\mathrm{tr}\left(\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0{,}5 \\ 0 & 0 & 0 & 0 \\ 0 & 0{,}5 & 0 & 0 \end{bmatrix} y\right) = 0$$

$$\mathrm{tr}\left(\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0{,}5\\0&0&0{,}5&0\end{bmatrix}y\right)=0$$

3.

Ici on ne va modifier que la contrainte pour chacun des cas suivants :

a) $3x_1+2x_2\leq 6 \rightarrow 3x_1^2+2x_2^2\leq 6 \Leftrightarrow 3x_{11}+2x_{22}+z=6$ d'où la trace :

$$\mathrm{tr}\left(\begin{bmatrix}3&0&0&0\\0&2&0&0\\0&0&0&0\\0&0&0&1\end{bmatrix}y\right)=6$$

b) $3x_1+2x_2\leq 6 \rightarrow (3x_1+2x_2)^2\leq 6^2 \Leftrightarrow 9x_{11}+4x_{22}+12x_{12}+z=36$:

$$\mathrm{tr}\left(\begin{bmatrix}9&6&0&0\\6&4&0&0\\0&0&0&0\\0&0&0&1\end{bmatrix}y\right)=36$$

c) $3x_1+2x_2\leq 6 \rightarrow (3x_1+2x_2)^2\leq 6(3x_1+2x_2) \Leftrightarrow 9x_{11}+4x_{22}-12x_{12}-18x_1-12x_2+z=0$:

$$\mathrm{tr}\left(\begin{bmatrix}9&-6&-9&0\\-6&4&-6&0\\-9&-6&0&0\\0&0&0&1\end{bmatrix}y\right)=0$$

d) $3x_1+2x_2\leq 6 \rightarrow (6-(3x_1+2x_2))^2\geq 0$

$\Leftrightarrow 36-18x_1-12x_2-18x_1+9x_{11}+6x_{12}-12x_2+6x_{12}+4x_{22}-z=0$

$\Leftrightarrow 9x_{11}+4x_{22}-36x_1-24x_2+12x_{12}-z=-36$:

$$\mathrm{tr}\left(\begin{bmatrix}9&6&-18&0\\6&4&-6&0\\-18&-6&0&0\\0&0&0&1\end{bmatrix}y\right)=-36$$

e) Ici on établit 4 contraintes simultanées à la place de l'unique contrainte existante :

$$3x_1+2x_2\leq 6 \rightarrow \begin{cases}(3x_1+2x_2)x_1\leq 6x_1 & (\mathrm{A}_1)\\(3x_1+2x_2)x_2\leq 6x_2 & (\mathrm{A}_2)\\(3x_1+2x_2)(1-x_1)\leq 6(1-x_1) & (\mathrm{A}_3)\\(3x_1+2x_2)(1-x_2)\leq 6(1-x_2) & (\mathrm{A}_4)\end{cases}$$

Mais toutes les variables z sont maintenant distinctes (z_1, z_2, z_3 et z_4). On a donc une nouvelle matrice y :

$$y=\begin{bmatrix} x_{11} & x_{12} & x_1 & 0 & \cdots & \cdots & 0 \\ x_{12} & x_{22} & x_2 & \vdots & & & \vdots \\ x_1 & x_2 & 1 & 0 & & & \vdots \\ 0 & \cdots & 0 & z_1 & \ddots & & \vdots \\ \vdots & & & \ddots & z_2 & \ddots & \vdots \\ \vdots & & & & \ddots & z_3 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & z_4 \end{bmatrix}$$

Revenons à nos contraintes :

(A1) $\quad (3x_1+2x_2)x_1 \leq 6x_1 \Leftrightarrow 3x_{11}+2x_{12}-6x_1+z_1=0$:

$$\operatorname{tr}\left(\begin{bmatrix} 3 & 1 & -3 & 0 & \cdots & \cdots & 0 \\ 1 & 0 & 0 & \vdots & & & \vdots \\ -3 & 0 & 0 & 0 & & & \vdots \\ 0 & \cdots & 0 & 1 & \ddots & & \vdots \\ \vdots & & & \ddots & 0 & \ddots & \vdots \\ \vdots & & & & \ddots & 0 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 \end{bmatrix} y\right)=0$$

(A2) $\quad (3x_1+2x_2)x_2 \leq 6x_2 \Leftrightarrow 3x_{12}+2x_{22}-6x_2+z_2=0$:

$$\operatorname{tr}\left(\begin{bmatrix} 0 & 1{,}5 & 0 & 0 & \cdots & \cdots & 0 \\ 1{,}5 & 2 & -3 & \vdots & & & \vdots \\ 0 & -3 & 0 & 0 & & & \vdots \\ 0 & \cdots & 0 & 0 & \ddots & & \vdots \\ \vdots & & & \ddots & 1 & \ddots & \vdots \\ \vdots & & & & \ddots & 0 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 \end{bmatrix} y\right)=0$$

(A3) $\quad (3x_1+2x_2)(1-x_1) \leq 6(1-x_1) \Leftrightarrow 9x_1+2x_2-3x_{11}-2x_{12}+z_3=6$:

$$\operatorname{tr}\left(\begin{bmatrix} -3 & -1 & 4{,}5 & 0 & \cdots & \cdots & 0 \\ -1 & 0 & 1 & \vdots & & & \vdots \\ 4{,}5 & 1 & 0 & 0 & & & \vdots \\ 0 & \cdots & 0 & 0 & \ddots & & \vdots \\ \vdots & & & \ddots & 0 & \ddots & \vdots \\ \vdots & & & & \ddots & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 \end{bmatrix} y\right)=6$$

(A4) $\quad (3x_1+2x_2)(1-x_2) \leq 6(1-x_2) \Leftrightarrow 3x_1+8x_2-3x_{12}-2x_{22}+z_4=6$:

$$\operatorname{tr}\left(\begin{bmatrix} 0 & -1{,}5 & 1{,}5 & 0 & \cdots & \cdots & 0 \\ -1{,}5 & -2 & 4 & \vdots & & & \vdots \\ 1{,}5 & 4 & 0 & 0 & & & \vdots \\ 0 & \cdots & 0 & 0 & \ddots & & \vdots \\ \vdots & & & \ddots & 0 & \ddots & \vdots \\ \vdots & & & & \ddots & 0 & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 1 \end{bmatrix} y\right) = 6$$

Autre exercice de TP qui n'a rien à voir avec les énoncés

Euh ! Je crois qu'il fait trop beau pour un passage de variables du primal au dual : oui c'est une impasse.