TD1 - Modélisation

Arthur Garnier

1 Résolution graphique d'un PL

1.1

- x_1 = Qt de P1 à produire
- x_2 = Qt de P2 à produire
- $\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2$: vecteur inconnu
- Vecteur $= \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1700 \\ 3200 \end{bmatrix}$

1.1.1 Fonction objectif

$F(x) = 1700x_1 + 3200x_2$ (€) $= [c_1, c_2] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = c_1x_1 + c_2x_2$

1.1.2 Contraintes

- machine M1 : $3x_2 \leq 39$
- M2 : $1.5x_1 + 4x_2 \leq 60$
- M3 : $2x_1 + 3x_2 \leq 57$
- M4 : $3x_1 \leq 57$

Forme canonique pure du PL :

$$\begin{cases} max(\mathbf{c}^T\mathbf{X}) \\ A\mathbf{x} \leq \mathbf{b} \\ \mathbf{x} \geq 0 \end{cases}$$

$A = \begin{bmatrix} 0 & 3 \\ 1.5 & 4 \\ 2 & 3 \\ 3 & 0 \end{bmatrix}$

$B = \begin{bmatrix} 39 \\ 60 \\ 57 \\ 57 \end{bmatrix}$

$A = \begin{bmatrix} 0 & 3 \\ 1.5 & 4 \\ 2 & 3 \\ 3 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3x_2 \\ 1.5x_1 + 4x_2 \\ 2x_1 + 3x_2 \\ 3x_1 \end{bmatrix} \leq \begin{bmatrix} 39 \\ 60 \\ 57 \\ 57 \end{bmatrix}$

Forme standard :

On ajoute 4 variables d'écart correpondant aux 4 contraintes inégalité. On les notes $\begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix}$. Les contraintes se réécrivent :

$$\begin{cases} 3x_2 + e_1 = 39 \\ 1.5x_1 + 4x_2 + e_2 = 60 \\ 2x_1 + 3x_2 + e_3 = 57 \\ 3x_1 + e_4 = 57 \end{cases}$$

Avec de plus :

$$\begin{cases} x_1 \geq 0 \\ x_2 \geq 0 \\ e_1 \geq 0 \\ e_2 \geq 0 \\ e_3 \geq 0 \\ e_4 \geq 0 \end{cases}$$

Forme standard :

$\tilde{x} = \begin{bmatrix} x \\ e \end{bmatrix} \in \mathbb{R}^6$

$$\begin{cases} max(\tilde{A}\tilde{x}\tilde{b}) \\ \tilde{x} \geq 0 \end{cases}$$

$\tilde{A} \in \mathbb{M}_{4,6}(\mathbb{R})$ (4 lignes, 6 colonnes)

(2) est $\tilde{A}\tilde{x} = \tilde{b}$

$\tilde{A} = \begin{bmatrix} 0 & 3 & 1 & 0 & 0 & 0 \\ 1.5 & 4 & 0 & 1 & 0 & 0 \\ 2 & 3 & 0 & 0 & 1 & 0 \\ 3 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$

$\tilde{b} = b = \begin{bmatrix} 39 \\ 60 \\ 57 \\ 57 \end{bmatrix}$

Remarque : I_4 sur les variables d'écart !

$\tilde{c} = \begin{bmatrix} 1700 \\ 3200 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

$$\tilde{c}^T\tilde{x} = [170032000000]\begin{bmatrix} x_1 \\ x_2 \\ e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix} = 1700x_1 + 3200x_2$$

1.2 Résolution graphique

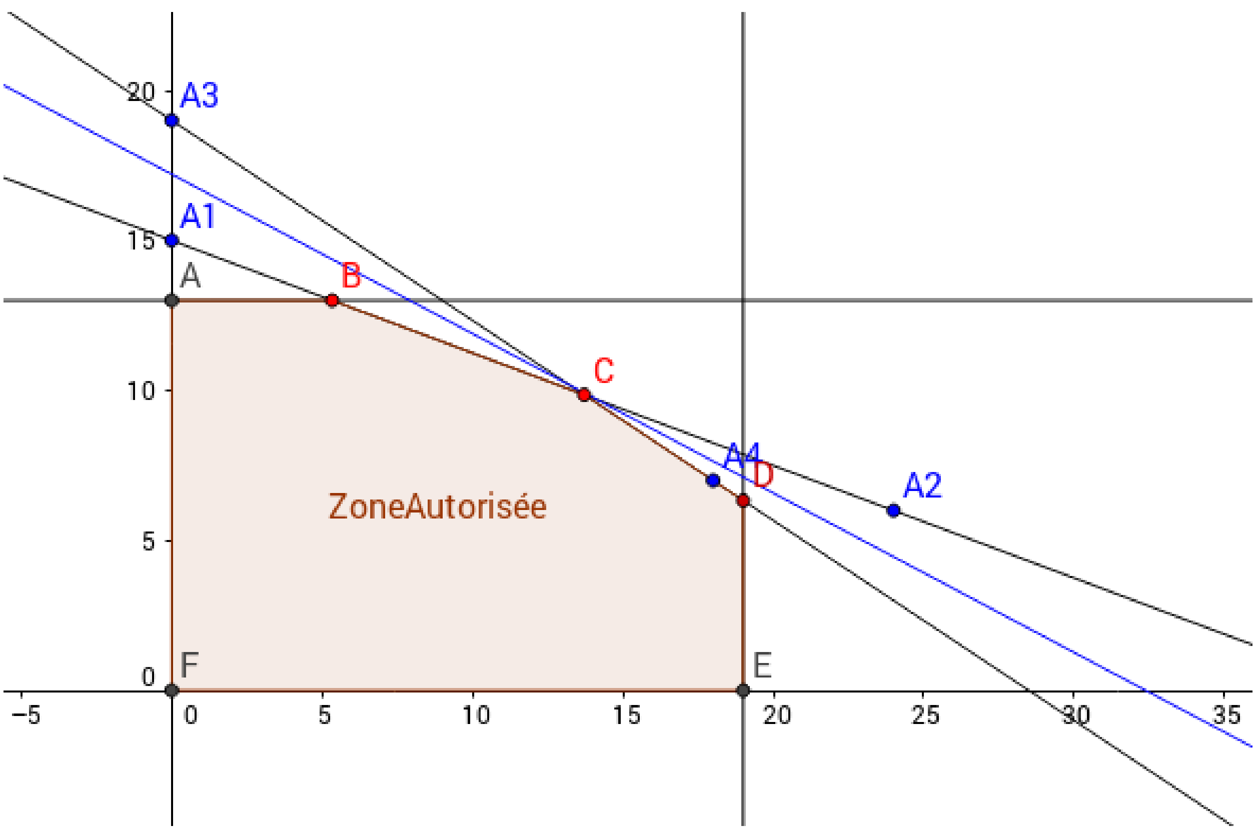
$$\begin{cases} 3x_2 \leq 39 \Leftrightarrow x_2 \leq \frac{39}{3} = 13(1) \\ 1.5x_1 + 4x_2 \leq 60(2) \\ 2x_1 + 3x_2 \leq 57(3) \\ 3x_1 \leq 57 \Leftrightarrow x_1 \leq \frac{57}{3} = 19 \end{cases}$$

- (2) : Demi plan limité par la droite $1.5x_1 + 4x_2 = 60$
- A1(0,15) ; A2(24,6)
- (3) : Demi plan limité par la droite $2x_1 + 3x_2 = 57$
- A3(0,19) ; A4(18,7)

On veut vrendre maximum : $F(x) = 1700x_1 + 3200x_2$ dans la zone autorisée de (x_1, x_2)

Les (x_1, x_2) correspondant à un gain F_0 donné sont sur la droite d'équation :

$1700x_1 + 3200x_2 = F_0 \Leftrightarrow x_2 = -\frac{1700}{3200}x_1 + \frac{F_0}{1700}$



Graphiquement on voit que (droite bleue) que l'optimum est obtenu au point (x_1^*, x_2^*) qui est à l'intersection des droites :

$$\begin{cases} 1.5x_1 + 4x_2 = 60 \\ 2x_1 + 3x_2 = 57 \end{cases}$$

La solution est :

$$\begin{cases} x_1^* = \frac{96}{7} \approx 13.71 \\ x_2^* = \frac{399}{21} \approx 9,85 \end{cases}$$

2 Problème du restaurateur

2.1

2.1.1 Disponibles

Variables :

- x_r = nombre de plateaux riches
- x_s = nombre de plateaux simples

2.1.2 Fonction objectif

$F(x_r, x_s) = P_r x_r + P_s x_s$

2.1.3 Contraintes

$$\begin{cases} n_1^r x_r + n_1^s x_s \leq k_1 \text{oursins}(1) \\ n_2^r x_r + n_2^s x_s \leq k_2 \text{palourdes}(2) \\ n_3^r x_r + n_2^s x_s \leq k_3 \text{huitres}(3) \\ x_r \geq 0, x_s \geq 0(4) \\ x_r \text{ et } x_s \text{ entier}(5) \end{cases}$$

(PL) : $max(P_r x_r + P_s x_s)$ sous les contraintes (1),(2),(3),(4),(5)

Forme standard :

On rajoute les variables d'écart

$$\begin{cases} n_1^r x_r + n_1^s x_s + e_1 = k_1 \\ n_2^r x_r + n_2^s x_s + e_2 = k_2 \\ n_3^r x_r + n_3^s x_s + e_3 = k_3 \\ x_r \geq 0, x_s \geq 0, e_1 \geq 0, e_2 \geq 0, e_3 \geq, e_4 \geq 0 \end{cases}$$

2.2

- y_1 = prix d'un oursin
- y_2 = prix d'une palourde
- y_3 = prix d'une huitre

2.2.1 Fonction objectif

$G(y_1, y_2, y_3) = k_1y_1 + k_2y_2 + k_3y_3$ (prix de la totalité de coquillages)

2.2.2 Problème

(PL) : $minG(y_1, y_2, y_3)$

$$\begin{cases} n_1^r y_1 + n_2^r y_2 + n_3^r y_3 \geq P_r \\ n_1^s y_1 + n_2^s y_2 + n_3^s y_3 \geq P_s \\ y_1 \geq 0, y_2 \geq 0, y_3 \geq 0 \end{cases}$$

- Problème (PL) n°1 : 2 inconnues, 3 contraintes
- Problème (PL) n°2 : 3 inconnues, 2 contraintes

Le problème (PL) n°2 est le problème dual du (PL) n°1

- $(n_1^r y_1 + n_2^r y_2 + n_3^r y_3 \geq P_r)x_r \geq P_r, x_r \geq 0$
- $(n_1^s y_1 + n_2^s y_2 + n_3^s y_3 \geq P_s)x_s \geq P_s, x_s \geq 0$

$(n_1x_r + n_1^s x_s)y_1 + (n_2x_r + n_2^s x_s)y_2 + (n_3x_r + n_3^s x_s)y_3 \geq P_r x_r + P_s x_s$

$k_1y_1 + k_2y_2 + k_3y_3 \geq rx_r + P_s x_s$

C'est à dire :

$G(y_1, y2, y_3) \geq F(x_r, x_s)$

Autrement dit : la produit de la vente à l'amie est toujours supérieur ou égal au bénéfice tiré de la vente des plateaux.

En notant

- (x_1^*, x_2^*) = optimum PL n°1
- (y_1^*, y_2^*, y_3^*) =optimum PL n°2

On a en fait : $G(y_1^*, y_2^*, y_3^*) = F(x_1^*, x_2^*)$

3 Problème nutritiel

TO-DO

4 Répartition de tâches

4.1

- n procédures (tâches) à exécuter sur un processeurs.
- procédure i sur un processeur j : coût $c_{ij} > 0$
- Fonction objectif : coût total des n procédures

4.1.1 Modélisation (PL) du problème

$$x_{ij} = \begin{cases} 1 \text{ si la tâche i exécutée par proc j} \\ 0 \text{ sinon} \end{cases}$$

4.1.2 Fonction objectif

Coût total : $\sum_j \sum_i x_{ij} c_{ij}$

$\boxed{F(\mathbf{x}) \triangleq \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij} c_{ij}}$

1. Contraintes en variables binaires

$$\begin{cases} \sum_{j=1}^{m} X_{ij} = 1 \\ X_{ij} = \{0, 1\} \end{cases}$$

2. Alternative : on demande seulement que $x_{i,j}$ entier

$\sum_{j=1}^{m} x_{ij} = 1, i = 1...n$

$x_{i,j} \geq 0, i = 1...nj = 1....m$

$X_{i,j}$ entier

3. Contraintes réelles

On veut un système de contraintes réelles sur $x_{i,j}$ qui implique en fait que $x_{i,j}$ est entier

$\sum_{j=1}^{m} x_{ij} = 1, i = 1...n$

$\sum_{i=1}^{n} x_{i,j} \leq n, j = 1..m$

$\forall i, j x_{i,j} \geq 0$

$X_{i,j}$ entier

Alternative : Minimiser le plus ggrand cout d'exécution

La fonction objectif devient :

$F_1(x) \triangleq max_j [\sum_{i=1}^{n} c_{ij} x_{ij}]$

5 Stockage

n fichiers vidéo de taille : $c_1, c_2, ..., c_n$ et m disques.

- Fonction objectif :

Nombre de disques durs.

- Modélisation :

$$y_j = \begin{cases} 1 \text{ si disque j utilisé} \\ 0 \text{ sinon} \end{cases} \qquad x_{i,j} \begin{cases} 1 \text{ si fichier i stocké sur disque j} \\ 0 \text{ sinon} \end{cases}$$

$y_j = 0 \Rightarrow x_{i,j} \forall i = 1...n$ (disque j non utilisé)

Formulation 1 :

1. Fonction objectif :

$F(x, y) = \sum\limits_{j=1}^{m} y_j$ (nombre de disques utilisés)

2. Contraintes :

- $\sum\limits_{i=1}^{n} c_i x_{ij} \leq C, j = 1..m$ [Stockage total sur disque j n'excède pas C]
- $\sum\limits_{j=1}^{n} x_{ij} = 1, i = 1...n$ [un fichier est stocké une seule fois]
- $x_{ij} \leq y_j$ [relation ($\mathbb{R}$)]
- $y_j \in \{0, 1\} \forall j = 1...m$
- $x_{ij} \in \{0, 1\}, \forall i = 1...n; \forall j = 1...m$

Formulation 2 :

On remarque qu'on peut supprimer la 3ème contrainte à condition de remplacer la contrainte 1 par (1'):
$\sum\limits_{i=1}^{n} c_i x_{ij} \leq C_{y_j}$

On a : $1' \Leftrightarrow 1 \oplus 3$

1. Montrons d'abord : $1 \oplus 3 \Rightarrow 1'$

$\sum\limits_{i=1}^{n} c_i x_{ij} \leq C$

$(\sum\limits_{i=1}^{n} c_i x_{ij}) y_j \leq C y_j$

$\sum\limits_{i=1}^{n} c_i (x_{ij} y_j) \leq C y_j$

Mais on constate que : $x_{ij} \leq x_{ij} y_j$!!

En effet :

- Ou bien $y_j = 0$ on a $0 \geq 0$
- Ou bien $y_j = 1$

Donc $\sum_{i=1}^{n} c_i(x_{ij}y_j) \leq Cy_j$ implique $\sum_{i=1}^{n} c_i x_{ij} \leq Cy_j$ c'est 1' !

La Formulation 2 est meilleure (n+m contraintes). Alors qu'il y a $2(n+m)$ contraintes pour la formulation 1.

6 Affectation de bureau

n bureau / n services

- d_{jl} = distance entre bureau j et l
- c_{ik} = nombre de fois où des employés du service i vont au service k

Nombre de répartition possible services / bureaux

- Service 1 : n Choix
- Service 2 : n-1 choix ... service n : 1 choix $\Rightarrow n!$

2. Modélisation 1 du PL

- $x_{ij} = 1$ si service i dans bureau j/0 sinon
- Fonction cout :

$F = \sum_{i=1}^{n} \sum_{j=i+1}^{n} C_{ik} [\sum_{j=1}^{n} \sum_{l=1}^{n} d_j l x_{ij} x_k l]$

Contraintes :

- $\forall i$ service $\sum_{j=1}^{n} x_{ij}$ [le service i est dans un seul bureau]
- $\forall j$ bureau $\sum_{i=1}^{n} x_{ij} = 1$ [le bureau j est occupé par un seul service]

F(x) est quadratique en x $\Rightarrow$ problème !

3. Modélisation 2

On introduit une autre variable binaire :

$x_{ij}^{kl} = x_{ij} \times kl$

On a :

$x_{ij}^{kl} = 1$ si le couple de services (i,k) occupe le couple de bureau (j,l) / 0 sinon

La fonction F devient :

$F = \sum_{i=1}^{n} \sum_{k=1+1}^{n} \sum_{j=1}^{n} \sum_{l=1}^{n} c_{ik} d_{jl} x_{ij}^{kl}$

Les contraintes deviennes :

- $\forall i, \forall k \sum_{j=1}^{n} \sum_{l=1}^{n} x_{ij}^{kl} = 1$ [chaque couple de service occupe un couple de bureaux]
- $\forall j, l \sum_{i=1}^{n} \sum_{k=1}^{n} x_{ij}^{kl} = 1$ [chaque couple de bureaux est affecté à un seul couple de services]

7 Exercice 10

(1) $min_{x \in \mathbb{R}^n} max|f(t_i) - y_i|$

Montrons que (1) est en fait un problème (PL)

$||x||_\infty = max|x_i|$

$||x||_1 = \sum_{i=1}^{n} |x_i|$

1. (1) s'écrit $minmax|(Ax)_i - b_i|$ où A est une matrice $m \times n$ à déterminer et $b \in \mathbb{R}^m$ = vecteur à déterminer

On pose :

(2) $b = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$ = vecteur des m mesures

On veut :

$b_i = y_i = f(t_i) = \sum_{j=1}^{n} x_j \varphi_j(t_i) = (Ax)_i$

Ca donne : $A = [a_{ij}]$ avec $a_{ij} = \varphi_j(t_i)$

Avec (2) et (3) on a bien :

(1) $\Leftrightarrow minmax|(Ax)_i - b_i| = min||Ax - b||$ (4)

2. Mise de (4) sous forme (PL) canonique pure.

On pose $e = max|(Ax)_i|$=nouvelle variable

On a $|(Ax)_i - b_i| \leq e$

Variable $\tilde{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ e \end{bmatrix} \in \mathbb{R}^{n+1}$

Fonction objectif : On minimise $F(\tilde{x}) = e$

$|(Ax)_i - b_i| \leq e \Leftrightarrow -e \leq (Ax)_i - b_i \leq e$

Conclusion : (4) équivaut à (PL)

$min[F \triangleq e]$

8 Exercice 9 : Voyageur de commerce en PL

- Variables $u_i, 1 \leq i \leq n$

u_i = position ville i dans la tournée

$u_1 = 1; i_i = \{2, 3, ..., n\}; i \geq 2$

8.1

$x_{ij} = 1 \Rightarrow u_i < u_j(1)$

Montrons que (1) équivaut à $u_i - u_j + 1 \leq \alpha(1 - x_{ij})(2)$ avec α bien choisi.

En effet :

1. Si x_{ij} = 1 (2)$ devient $u_i - u_j + 1 \leq 0 \Leftrightarrow u_i < u_j$
2. Si $x_{ij} = 0(2)$ devient $u_i - u_j + 1 \leq \alpha$ (3)

Pour quelle valeur de α (3) est elle vraie ?

On a : $u_i \in \{2...n\}$ pour $i = 2, ...n$ donc

$2 - n + 1 \leq u_i - u_j + 1 \leq n - 2 + 1 \Leftrightarrow 3 - n \leq u_i - u_j + 1 \leq n - 1$

Donc avec le choix $\alpha = n - 1$, la relation (3) est VRAIE.

Donc on traduit $u_i < u_j$ par : $u_i - u_j + 1 \leq (n - 1)(1 - X_{ij})$