

Notes de cours de ARO

Yann Miguel

29 septembre 2020

Table des matières

1	Introduction	2
2	Cours 1	3
2.1	La programmation linéaire	3
2.2	La méthode simplex	3
2.2.1	Cas spéciaux	5
3	Cours 2	7
3.1	La méthode simplex à deux phases	7
3.2	L'algorithme du simplex en cas général	7
3.3	Dictionnaires, cas général	8
4	Cours 3	11
4.1	Dualité dans la programmation linéaire	11
5	Informations importantes	13

1 Introduction

Cette UE consiste à modéliser les problèmes en flot, puis à les programmer en linéaire.

2 Cours 1

2.1 La programmation linéaire

Exemple de problème linéaire: maximiser $x_1 + x_2$ avec:

- $x_2 - x_1 \leq 1$
- $x_1 + 6x_2 \leq 15$
- $4x_1 - x_2 \leq 10$
- $x_1 \geq 0$, et $x_2 \geq 0$

Si on dessine ce problème sur un plan, on observe une zone du plan correspondant à ces contraintes, et la solution optimale avec ces contraintes est 5, avec $x_1 = 3$ et $x_2 = 2$.

2.2 La méthode simplex

Il s'agit d'aller de solution de base en solution de base.

Une solution de base est l'ensemble des sommets du polyèdre, et le polyèdre est l'ensemble des solutions admissibles.

La méthode simplex est donc une méthode de voyage entre les sommets du polyèdre solution.

Forme générale du problème linéaire: maximiser $c_1x_1 + c_2x_2 + \dots + c_nx_n$ avec:

- $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1$
- $a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_n$
- $x_1, \dots, x_n \geq 0$

Chaque problème de maximisation a un problème de minimisation associé.

La programmation linéaire date de Fourier, alors que la méthode suplex a été inventée dans les années 50 par Dantzig.

L'algorithme de la méthode simplex est exponentiel dans le pire des cas, et polynomial en moyenne.

L'algorithme de Khachian, créé dans les années 70 est un algorithme polynomial un peu moins efficace que la méthode suplex.

Prenons l'exemple de la diète. Notre corps à besoin de:

- Vitamine A : 0,5mg
- Vitamine C : 15mg
- Fibres : 4g

Chaque nourriture à un prix, et notre corps à des demandes nutritionnelles.

Nom nutriment	Carottes(x_1)	Choux(x_2)	Cornichons(x_3)	Demandes
Vitamine A	35mg	0,5mg	0,5mg	0,5mg
Vitamine C	60mg	30mg	10mg	15mg
Fibres	30g	20g	10g	4g
prix	0,75 euro	0,5 euro	0,15 euro	

Cette exemple demandera donc de MINIMISER la quantité d'aliments à prendre, afin de minimiser le coût, et d'avoir les apports minimaux requis.

Exemple de la méthode suplex:

Objet	Cendrier(x_1)	Bol(x_2)	Cruche(x_3)	Vase(x_4)	maximum
Moulage	2	4	5	7	42
Cuisson	1	1	2	2	17
Peinture	1	2	3	3	24
Bénéfice	7	9	18	17	

On veut donc, dans ce problème, maximiser $7x_1 + 9x_2 + 18x_3 + 17x_4$ avec:

- $2x_1 + 4x_2 + 5x_3 + 7x_4 \leq 42$
- $1x_1 + 1x_2 + 2x_3 + 2x_4 \leq 17$
- $1x_1 + 2x_2 + 3x_3 + 3x_4 \leq 24$
- $x_1, x_2, x_3, x_4 \geq 0$

Dictionnaire D1:

- $x_5 = 42 - 2x_1 - 4x_2 - 5x_3 - 7x_4$
- $x_6 = 17 - 1x_1 - 1x_2 - 2x_3 - 2x_4$
- $x_7 = 24 - 1x_1 - 2x_2 - 3x_3 - 3x_4$
- *traitdedivision*
- $z = 7x_1 + 9x_2 + 18x_3 + 17x_4$

Les variables de base sont les x de 5 à 7, et les variables hors base sont les x de 1 à 4.

La solution de base admissible consiste à affecter à toutes les variables hors base la valeur 0, ce qui crée:

- $x_5 = 42$
- $x_6 = 17$
- $x_7 = 24$
- $z = 0$

La méthode suplex va consister à faire entrer une variable hors base dans la base, et à faire sortir une variable de base de la base.

On va donc faire entrer la variable x_3 , car c'est la variable qui donne le plus de bénéfices (technique glouton). La variable qui va sortir sera celle qui mets le plus de contraintes, donc qui limite le plus la variable entrante. Il s'agit de x_7 dans ce cas.

L'étape de pivot fait donc entrer x_3 et sortir x_7 . Donc, on a :

$$3x_3 = 24 - 1x_1 - 2x_2 - 3x_4 - x_7$$

Ce qui donne :

$$x_3 = 8 - \frac{1}{3}x_1 - \frac{2}{3}x_2 - x_4 - \frac{1}{3}x_7$$

Le dictionnaire D2 est crée en remplaçant x_3 par l'expression en haut sur toutes les lignes de D1. En le simplifiant, ça donne :

- $x_3 = 8 - \frac{1}{3}x_1 - \frac{2}{3}x_2 - x_4 - \frac{1}{3}x_7$
- $x_5 = 2 - \frac{1}{3}x_1 - \frac{2}{3}x_2 - 2x_4 - \frac{5}{3}x_7$
- $x_6 = 1 - \frac{1}{3}x_1 - \frac{1}{3}x_2 - \frac{2}{3}x_7$
- *traitdedivision*
- $z = 144 + x_1 - 3x_2 - x_4 - 6x_7$

La solution de base admissible dans ce cas est 144 euros gagnés.

La variable entrante de D2 est x_1 , car c'est la seule encore positif dans l'équation avec z. La variable sortante est x_6 . Ce qui donne :

$$\frac{1}{3}x_1 = 1 + \frac{1}{3}x_2 - x_6 + \frac{2}{3}x_7$$

Ce qui donne, après simplification :

$$x_1 = 3 + x_2 - 3x_6 + 2x_7$$

On obtient D3 en remplaçant x_1 par cette équation dans tout D2. Après simplification, ça donne :

- $x_1 = 3 + x_2 - 3x_6 + 2x_7$
- $x_3 = 7 - x_2 - x_4 + x_6 - x_7$
- $x_5 = 1 - x_2 - 2x_4 + x_6 + x_7$
- *traitdedivision*
- $z = 147 - 2x_2 - x_4 - 3x_6 - 4x_7$

Étant donné qu'il n'y a plus de variable positive dans l'équation avec z, la solution de base admissible de D3 est la solution optimale. Donc, optimalement, on fera 3 cendriers, 0 bols, 7 cruches et 0 vases, et on fera un bénéfice de 147 euros.

2.2.1 Cas spéciaux

Cas spécial 1: maximum non borné

Si le maximum n'est pas borné, alors maximiser retourne $+\infty$.

Si il n'y a pas de contrainte sur x_2 , alors le maximum va vers l'infini.

Cas spécial 2: Dégénérescence et cyclage

La dégénérescence, c'est quand la borne supérieure de la variable

entrante est 0. Dans ce cas, on fait entrer dans la base cette variable, même si on ne gagne rien.

Le cyclage, c'est quand des étapes dégénèrent et nous ramènent toujours aux mêmes dictionnaires. Il faudra appliquer la règle de Blund pour s'en sortir.

La règle de Blund consiste à toujours choisir les variables entrantes et sortantes de plus petit indice, si jamais il y a plus d'une variable possible.

Théorème La méthode suplex avec la règle de Blund n'a pas de cyclage.

Cas spécial 3: Recherche d'un dictionnaire et infaillibilité

Il faut utiliser la méthode suplex à deux phases, ce qui permet, via l'introduction d'un programme linéaire auxiliaire, de gérer le cas où créer la solution de base admissible du dictionnaire donne des x négatifs.

3 Cours 2

3.1 La méthode simplex à deux phases

La méthode simplex à deux phases consiste à résoudre un programme linéaire auxiliaire avant de résoudre le premier. Si le résultat est 0, alors il n'y a aucune réparation à faire dans le programme principal. Sinon, il faut réparer le programme principal, et il n'a pas de solution admissible.

Remarque:

Un programme linéaire possède une solution admissible, et donc un dictionnaire réalisable, si et seulement si son programme linéaire auxiliaire a $x_0=0$ comme solution optimale. Par conséquent, x_0 est une variable dite de réparation pour le programme linéaire.

3.2 L'algorithme du simplex en cas général

Forme canonique:

maximiser $c^t x$ avec:

- $Ax \leq b$
- $x \geq 0$

Forme equationnelle:

maximiser $c^t x$ avec:

- $Ax = b$
- $x \geq 0$

On passe de la forme canonique à la forme équationnelle en ajoutant les variables d'écart x_{n+1}, \dots, x_{n+m} . Notons que A devient une matrice $(n+m) \times m$ et que c devient une matrice $(n+m)$ alors que b reste comme avant.

Convention:

On suppose par la suite le programme linéaire en forme equationnelle avec les lignes de la matrice A linéairement indépendantes.

Définition:

Une solution de base admissible du programme linéaire $\max c^t x, Ax=b, x \geq 0$ est une solution admissible x pour laquelle il existe $B \subseteq \{1, 2, \dots, n+m\}$ tel que:

1. La matrice A_B est non singulière.
2. $x_j=0$ pour tout $j \notin B$.

Observations:

1. A_B est une matrice carrée $m \times m$.

2. Une matrice carrée est non-singulière si et seulement si:

- elle est inversible
- son déterminant est $\neq 0$

Par la suite, si $B \subseteq \{1, 2, \dots, n+m\}$ est associé à une solution de base admissible, alors on appelle B une base admissible.

Proposition:

Toute solution de base admissible est déterminée de façon unique par B. En d'autres termes, si $B \subseteq \{1, 2, \dots, n+m\}$ est A_B est non-singulière, alors il existe au plus une solution admissible x avec $x_j=0, \forall j \notin B$.

3.3 Dictionnaires, cas général

Définition:

Un dictionnaire $D(B)$, associé à une base admissible B est un système de $m+1$ équations linéaires en variables $N=\{1, \dots, n+m\} \setminus B$ qui possède le même ensemble des solutions que le système $Ax=b, z=c^t x$ et à la forme $\frac{x_B=p+Qx_N}{z=z_0+d^t x_N}$, avec:

- x_B : vecteur de variables de base
- x_N : vecteur de variables hors-base
- p : vecteur avec m coordonnées
- d : vecteur avec n coordonnées
- Q : matrice $n*m$
- z_0 : réel

Solution de base associée à $D(B)$: $x_N=0, x_B=p, z=z_0$.

Proposition:

Si B est une base admissible, alors $D(B)$ est défini de façon unique pour les formules:

- $Q = -A_B^{-1}A_N$
- $p = A_B^{-1}b$
- $z_0 = c_B^T A_B^{-1}b$
- $d = c_N - (c_B^T A_B^{-1}A_N)^T$

L'algorithme du simplexe

0. Calculer une base admissible et une solution base admissible (en utilisant le prog. lin. auxiliaire). Si une telle base et solution de base existe pas, alors stop. Sinon,
1. Soit B la base admissible courante et soit $D(B)$ le dictionnaire associé.
2. Si $d \leq 0$, alors la solution base admissible $x_N=0, x_B=p, z=z_0$ est optimale. Stop.
3. Sinon, choisir $x_r, r \in N$ avec $d_r > 0$ comme variable entrante. Si il y a plusieurs possibilités, utiliser la règle du pivot.
4. Choisir $x_u, u \in B$ la plus contraignante sur l'augmentation de x_r comme variable sortante.
5. Faire $B \leftarrow B \setminus \{u\} \cup \{r\}$, calculer $D(B)$ et aller à l'étape 1.

Optimalité:

Comme $z = z_0 + dx$ de $D(B)$ est égale à $c^t x$, si on choisit une autre solution admissible $x' = (x'_1, \dots, x'_{n+m})$, alors, comme $x'_j \geq 0, \forall j \in N$ et $d_j \leq 0, \forall j \in N$, on obtiendras que $c^t x' = z_0 + dx' \leq z_0$.

Règle du pivot:

1. (Dantzig) plus grand coefficient: choisir comme variable entrante la variable avec le plus grand coefficient positif.
2. plus grande croissance: la variable avec le coefficient positif qui augmente le plus la fonction objectif

3. (Bland) la variable avec coefficient positif et plus petit indice
4. steepest edge: maximiser le produit scalaire avec le vecteur c : $\max \frac{c^T(x_{NEW}-x_{OLD})}{||x_{new}-x_{OLD}||}$

4 Cours 3

4.1 Dualité dans la programmation linéaire

maximiser $2x_1+3x_2$ avec :

- $4x_1+8x_2 \leq 12$
- $2x_1+x_2 \leq 3$
- $3x_1+2x_2 \leq 4$
- $x_1, x_2 \geq 0$

L'objectif est de calculer les bornes supérieures pour l'optimum sans résoudre le programme linéaire.

1. Notons que $2x_1+3x_2 \leq 4x_1+8x_2 \leq 12 \Rightarrow \text{OPT} \leq 12$
2. Notons aussi $2x_1+3x_2 \leq \frac{1}{2}(4x_1+8x_2) \leq 6 \Rightarrow \text{OPT} \leq 6$
3. Aussi $2x_1+3x_2 = \frac{1}{3}(4x_1+8x_2) + \frac{1}{3}(2x_1+x_2) \leq \frac{12}{3} + \frac{3}{3} = 5 \Rightarrow \text{OPT} \leq 5$

Cas général:

$$\left\{ \begin{array}{l} \text{maximiser } c_1 x_1 + \dots + c_n x_n \\ a_{11} x_1 + \dots + a_{1n} x_n \leq b_1 \quad (y_1) \\ \vdots \\ a_{m1} x_1 + \dots + a_{mn} x_n \leq b_m \quad (y_m) \end{array} \right. \quad (PL)$$

$$\left\{ \begin{array}{l} \text{minimiser } b_1 y_1 + \dots + b_m y_m \\ a_{11} y_1 + \dots + a_{m1} y_m \geq c_1 \\ \vdots \\ a_{1n} y_1 + \dots + a_{mn} y_m \geq c_n \end{array} \right. \quad (D)$$

③

$$\left\{ \begin{array}{l} \text{maximiser } c^T x \\ Ax \leq b \\ x \geq 0 \end{array} \right. \quad (PL) \qquad \left\{ \begin{array}{l} \text{minimiser } b^T y \\ A^T y \geq c \\ y \geq 0 \end{array} \right. \quad (D)$$

5 Informations importantes

Il n'y aura que deux séances de tds et tps, il faut donc bosser chez soi.