Introduction à la programmation linéaire

François Schwarzentruber

7 février 2021

1 Qu'est ce que c'est ?

Exemple 1

$$\textbf{maximiser } -4x_1 + 3x_2 + 2x_3$$
$$\begin{cases} x_1 \leqslant 20 \\ 2x_2 \leqslant 50 \\ x_1 + x_3 \leqslant 10 \\ x_1, x_2 \geq 0 \\ x_1, x_2 \in \mathbb{R}, x_3 \in \mathbb{N} \end{cases}$$

Définition 2 (programme linéaire mixte) Un programme linéaire est la donnée
- d'une fonction objectif linéaire à optimiser (maximiser/minimiser) ;
- de contraintes (égalités/inégalités larges) linéaires sur les variables ;
- contraintes de typage (dire pour toute variable si elle est entière, réelle).

Définition 3 (solution) Une solution est un point qui satisfait les contraintes.

Définition 4 (solution optimale) Une solution est optimale si elle optimise la fonction objectif.

Définition 5 (programmation linéaire mixte) La programmation linéaire est le problème suivant :
- Entrée : un programme linéaire ;
- Sortie : une solution optimale du programme linéaire, "non borné" s'il n'y a pas d'optimum, ou "impossible" si les contraintes sont inconsistantes

2 Solveur `lp_solve`

```
lp_solve monfichier.lp
```

Contenu de `monfichier.lp`

```
max: -4 x1 + 3 x2 + 2 x3;

x1 <= 20;
2 x2 <= 500;
x1 + x3 <= 10;
int x3;
```

- Par défaut les variables sont positives, pour libérer cette contrainte : `free x;`
- Par défaut les variables sont réelles, pour forcer une variable x à être entière : `int x;`, binaire : `bin x;`

Avantages d'utiliser un solveur
- rapide de créer un prototype
- générique : on peut facilement résoudre des variantes d'un problème
- moins de bugs
- efficace car les solveurs le sont
- si pas efficace, permet de justifier l'algorithme dédié

3 Exemples

Exercice 6 (diététique pas cher)

	Carottes crues	Chou blanc	Concombre	Requis par repas
Vitamine A [mg/kg]	35	0.5	0.5	0.5
Vitamine C [mg/kg]	60	300	10	15
Fibre [g/kg]	30	20	10	4
Prix [€/kg]	0.75	0.5	0.15	-

Trouver un repas le moins cher possible qui satisfait les contraintes nutritives. Donner un programme linéaire.

Exercice 7 Une méthode très populaire pour trouver une approximation linéaire s'appelle la méthode des moindres carrés. Etant donné $(x_1, y_1), \dots, (x_n, y_n)$ des points du plan, on souhaite trouver la droite $y = ax + b$ qui minimise $\sum_{i=1}^{n}(ax_i + b - y_i)^2$. Une méthode moins sensible aux erreurs est de minimiser plutôt

$$\sum_{i=1}^{n} |ax_i + b - y_i|.$$

Exprimer ce problème de minimisation comme un problème de programmation linéaire.

Proposition 8 (flot max vers prog linéaire) Le problème du flot maximal se réduit en temps polynomial à la programmation linéaire. A toute réseau de flot (S, A, c, s, t) on associe le programme linéaire suivant :

$$\textbf{maximiser } \sum_{v|(s,v)\in A} f_{sv}$$
$$\begin{cases} f_{uv} \leq c(u,v) \text{ pour tout } (u,v) \in A \\ \sum_{(v,u)\in A} f_{vu} = \sum_{(u,w)\in A} f_{uw} \text{ pour tout } u \in S \setminus \{s,t\} \\ f_{uv} \geq 0, f_{uv} \in \mathbb{R} \text{ pour tout } (u,v) \in A \end{cases}$$

Exercice 9 Montrer comment écrire un programme linéaire où on demande en plus de faire passer au moins 5 unité d'eau dans un certain arc du réseau.

Exercice 10 Montrer comment écrire un programme linéaire où il faut faire un flot f de valeur au moins D (une certaine demande) mais en minimisant une fonction de coût $\sum_{(u,v)\in E} a(u,v) f_{uv}$ où $a(u,v)$ est le coût de faire passer une unité dans l'arc (u,v).

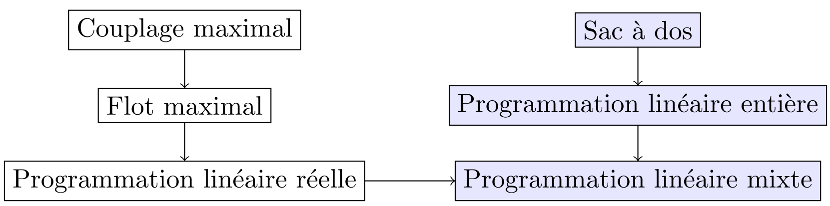
Exercice 11 (Sac à dos) Donner le programme linéaire correspondant à l'instance de sac à dos suivante :

Objet i	Poids p_i	Valeur v_i
1	6	30€
2	3	14€
3	4	16€
4	2	9€

4 Complexité

Définition 12 (programme linéaire réel/entier) = programme linéaire où les variables sont toutes des réels/entiers.

Définition 13 (programmation linéaire réel/entière) problème de programmation linéaire où les instances sont des programmes linéaires réels/entiers.



Théorème 14 (admis) La programmation linéaire réelle est dans P.

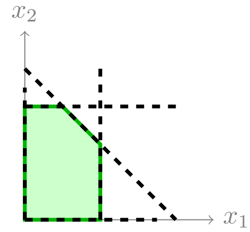
Théorème 15 (admis) La programmation linéaire entière est NP-complète.

5 Programmation linéaire réelle

Exemple 16

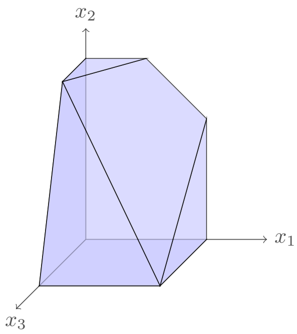
Il vend des chocolats simples (1 euro), des pyramides (6 euros).
Au maximum, il peut vendre 200 chocolats simples, 300 pyramides, pas plus de 400 chocolats en tout.

$$\textbf{maximiser } x_1 + 6x_2$$
$$\begin{cases} x_1 \leqslant 200 \\ x_2 \leqslant 300 \\ x_1 + x_2 \leqslant 400 \\ x_1, x_2 \geq 0 \\ x_1, x_2 \in \mathbb{R} \end{cases}$$



Exemple 17
Il vend des chocolats simples (1 euro), des pyramides (6 euros) et des pyramides de luxe (13 euros).
Au maximum, il peut vendre 200 chocolats simples, 300 pyramides, pas plus de 400 chocolats en tout et le nombre de pyramides plus trois fois le nombre de pyramides de luxe est au plus 600.

$$\textbf{maximiser } x_1 + 6x_2 + 13x_3$$
$$\begin{cases} x_1 \leqslant 200 \\ x_2 \leqslant 300 \\ x_1 + x_2 + x_3 \leqslant 400 \\ x_2 + 3x_3 \leqslant 600 \\ x_1, x_2, x_3 \geq 0 \\ x_1, x_2, x_3 \in \mathbb{R} \end{cases}$$



Proposition 18 L'ensemble des solutions d'un programme linéaire **réel** est **convexe**.

DÉMONSTRATION. C'est l'intersection de convexes. ■

Théorème 19 Considérons un programme linéaire réel.

1. soit l'espace des solutions est vide ;
2. soit l'objectif est non borné ;
3. soit l'objectif est borné et **l'optimum est atteint en l'un sommet du polyèdre.**

Exercice 20 Donner un programme linéaire réel dont l'espace des solutions est vide.

Exercice 21 Donner un programme linéaire réel dont l'espace des solutions est non borné mais dont l'objectif est borné.

Exercice 22 Donner un programme linéaire réel dont l'espace des solutions est non borné et dont l'objectif ne l'est pas non plus.

Notes bibliographiques

L'exemple des chocolats vient de [DPV08]. Le livre [GM07] donne beaucoup d'exemples. A l'IRISA, voici un travail sur l'affectation de machines virtuelles sur des serveurs pour minimiser l'énergie non renouvelables utilisées [CDO17]. Autre travail à l'IRISA sur programmation linéaire pour la bio-informatique : [ADFL19].

Références

[ADFL19] Rumen Andonov, Hristo Djidjev, Sebastien François, and Dominique Lavenier. Complete assembly of circular and chloroplast genomes based on global optimization. *J. Bioinform. Comput. Biol.*, 17(3) :1950014, 2019.

[CDO17] Benjamin Camus, Fanny Dufossé, and Anne-Cécile Orgerie. The SAGITTA approach for optimizing solar energy consumption in distributed clouds with stochastic modeling. In Brian Donnellan, Cornel Klein, Markus Helfert, Oleg Yu. Gusikhin, and António M. Pascoal, editors, *Smart Cities, Green Technologies, and Intelligent Transport Systems - 6th International Conference, SMARTGREENS 2017, and Third International Conference, VEHITS 2017, Porto, Portugal, April 22-24, 2017, Revised Selected Papers*, volume 921 of *Communications in Computer and Information Science*, pages 52–76. Springer, 2017.

[DPV08] Sanjoy Dasgupta, Christos H. Papadimitriou, and Umesh V. Vazirani. *Algorithms.* McGraw-Hill, 2008.

[GM07] Bernd Gärtner and Jirí Matousek. *Understanding and using linear programming.* Universitext. Springer, 2007.