Recherche opérationnelle: Programmation Linéaire
1. Modélisation et Résolution Graphique en 2D

Sandra U. Ngueveu

INP-ENSEEIHT / LAAS-CNRS
sandra.ngueveu@toulouse-inp.fr - ngueveu@laas.fr

2019/2020

Programmation Linéaire (PL)

Modèle Mathématique Linéaire

$$\min(\text{ou max})\ \ f(X) = c_1x_1 + c_2x_2 + \ldots + c_{n-1}x_{n-1} + c_nx_n$$

sous les contraintes (s.c.)

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1,n-1}x_{n-1} + a_{1n}x_n &\geq b_1 \\ &\vdots \\ a_{j,1}x_1 + a_{j,2}x_2 + \ldots + a_{j,n-1}x_{n-1} + a_{j,n}x_n &= b_j \\ &\vdots \\ a_{m,1}x_1 + a_{m,2}x_2 + \ldots + a_{m,n-1}x_{n-1} + a_{m,n}x_n &\leq b_m \\ x_i \in \mathbb{R}, \qquad i &\in [1 \ldots n] \end{aligned}$$

où les c_i, b_j et a_{ji} sont des coefficients constants.

Programmation Linéaire (PL)

Modèle Mathématique Linéaire

- Les variables :
  - sont en nombre fini n (même s'il est très grand, de l'ordre du million)
  - ne peuvent prendre que des valeurs réelles
- Les contraintes sont linéaires, c'est à dire :
- La fonction objectif est linéaire

Contexte

Programmation Linéaire (PL)

Modèle Mathématique Linéaire

- Les variables :
- Les contraintes sont linéaires, c'est à dire :
  - sont de type égalité (signe $=$) ou inégalité (signe $\leq$ ou $\geq$)
  - ont un terme de gauche correspondant à une combinaison linéaire des variables x_i
  - ont un terme de droite égal à une valeur réelle
- La fonction objectif est linéaire

Exemples de Programmes Linéaires

$$\max x_1 + 2x_2 + 3x_3$$

s.c.
$$\begin{aligned} 7x_1 + x_3 &\leq 6 \\ x_1 + 2x_2 &\leq 20 \\ 3x_2 + 4x_3 &\leq 30 \\ x_1 \geq 0, x_2 \in \mathbb{R}, x_3 &\geq 0 \end{aligned}$$

$$\max x_1 + 2x_2 + 3x_3$$

s.c.
$$\begin{bmatrix} 7 & 0 & 1 \\ 1 & 2 & 0 \\ 0 & 3 & 4 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leq \begin{bmatrix} 6 \\ 20 \\ 30 \\ 0 \\ 0 \end{bmatrix}$$

$$\min -y_1 + y_3$$

s.c.
$$\begin{aligned} y_1 + y_2 &\geq 1 \\ y_1 + 2y_3 &\leq 3 \\ y_1 &\in \mathbb{R} \\ y_2 &\in \mathbb{R} \\ y_3 &\geq 0 \end{aligned}$$

$$\min f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

s.c.
$$\sum_{i=1}^{n} x_{ij} = 1, \quad \forall j \in [1, \ldots, n] \setminus \{i\}$$
$$\sum_{i=1}^{n} x_{ji} = 1, \quad \forall j \in [1, \ldots, n] \setminus \{i\}$$
$$0 \leq x_{ij} \leq 1, \quad \forall i \in [1, \ldots, n], \forall j \in [1, \ldots, n]$$

où n et les c_{ij} sont des constantes prédéfinies.

Un peu d'histoire

La Programmation Linéaire : une branche de la Recherche Opérationnelle

La R.O. apparaît en 1940 en Angleterre puis aux Etats-Unis à des fins de recherche militaire : il s'agit d'utiliser au mieux ses moyens militaires, à l'époques insuffisants (avions, forces antiaériennes (D.C.A.)), moyens maritimes).

Naissance de la Programmation Linéaire

En 1947, D.B. Dantzig, venant de soutenir sa thèse et conseiller à l'US Air Force propose l'algorithme du simplexe pour résoudre les pbs de planification des transports lors d'opérations militaires. Avant cela, ces pbs étaient résolus à la main, sans possibilité de refaire les calculs en cas de changement de dernière minute. De plus, le concept de "fonction-objectif globale" n'existait pas ; les décisions étaient prises par des règles de bon sens sur la base de l'expérience des décideurs.

Méthodes de résolution

Le premier algorithme polynomial pour la programmation linéaire est dérivé de la méthode générale de l'ellipsoïde défini par A. Nemirovski (Prix John von Neumann 2003), D. B. Yudin et N. Shor en 1970. L. Khachiyan a ainsi construit un algorithme de l'ellipsoide adapté à la programmation linéaire en 1979 dont le mérite tient plus à la contribution en théorie de la complexité et à l'ouverture ainsi réalisée vers les méthodes polynomiales plutôt qu'en son efficacité pratique jugée médiocre. Une nouvelle avancée décisive a été réalisée en 1984 par N. Karmarkar [14], chercheur à IBM qui a proposé, pour la première fois, une méthode des points intérieurs dont il a démontré la complexité polynomiale dans le pire des cas.

La majorité des solutions logicielles actuelles sont construites autour de l'algorithme du simplexe et d'algorithmes des points intérieurs.

Exemple

Modéliser le problème P_1 de la fiche jointe.

Variables

Fonction-objectif

Contraintes

Domaine de définition

Préciser un cas de figure où P_1 se modélise par PL et un cas de figure où P_1 se modélise par PLNE.

Exemple

Modèle Mathématique Résultant

Mises en forme particulières

Forme matricielle : On peut noter

- $x = (x_1, x_2, ...x_n)^T$ le vecteur des variables
- $b = (b_1, b_2, ..., b_m)^T$ celui des seconds membres des contraintes,
- $c = (c_1, c_2, ..., c_n)^T$ les coûts ou profits affectés aux variables
- et A la matrice $m \times n$ des a_{ij}.

Dans ce cas, les deux formes suivantes sont les plus courantes :

Forme Canonique

$$\begin{aligned} \max c.x \\ A.x \leq b \\ x \geq 0 \end{aligned} \quad \text{ou} \quad \begin{aligned} \min c.x \\ A.x \geq b \\ x \geq 0 \end{aligned}$$

Forme Standard

$$\begin{aligned} \max c.x \\ A.x = b \\ x \geq 0 \end{aligned}$$

Quelques transformations possibles

Appliquer à l'exemple précédent

Exemple : Représentation

Démarche

- Tracer la droite (D1) correspondant à l'égalité dérivée de la 1ère contrainte
- Que représentent les points sur la droite ?
- Quels sont les points qui satisfont la contrainte (1) ?
- Faire de même avec la contraintes (2)
- Quels sont les points qui satisfont à la fois les contraintes (1) et (2) ?
- Faire de même avec la contrainte (3)
- Faire de même avec le domaine de définition des variables
- Quels sont les points qui satisfont TOUTES les contraintes du PL ?
  Ces points constituent l'**ensemble des solutions réalisables du PL**.

Exemple : Interprétation

Ensemble des solutions réalisables

Polyèdre

Polyèdre convexe

Points extremum vs points intérieurs

Exemple : Représentation (2)

- Tracer la droite correspond à une valeur de fonction-objectif de 0 ($z = 0$, ou $f(X) = 0$). Conclure.
- Idem pour $z = 1$.
- Idem pour $z = 3$.
- Identifier le gradient de la fonction-objectif. Conclure.
- Jusqu'où pousser la démarche ?

Exemple : Interprétation (2)

Théorème

Récapitulatif de l'approche graphique

Approche Graphique 2D

- Tracer les axes d'abscisses et ordonnées en tenant compte du domaine de définition
- Tracer toutes les contraintes pour obtenir le polyèdre
- Tracer le vecteur-gradient correspondant à la fonction-objectif
- Trouver l'optimum s'il existe

Cas à n variables/dimensions

- Même idée à n dimensions (méthodes de gradient)
- Reste "dessinable" en 3D mais plus compliqué pour $n \geq 4$

Pour conclure ...

Intérêt/Force de la PL : Optimum local = Optimum global !

Limites : Pour être directement modélisable par PL, les actions/décisions modélisées par les variables doivent être :

- additives
- proportionnelles
- divisibles

Possibilité d'utiliser des techniques de linéarisation pour modéliser quand même en PL des problèmes qui ne respectent pas a priori les conditions ci-dessus.

Résumé de ce qui a été vu

- Spécificités de la PL
- Modélisation mathématique
- Tracé, Interprétation et Résolution Graphique du cas 2D

Aperçu du prochain cours

- Programmes linéaires à $n > 2$ variables : Résolution par Simplexe [1]

1. pré-requis : calculs matriciels, résolution de systèmes linéaires par pivot de Gauss