

Programmation linéaire

Heure 4

Optimisation multicritère et dualité

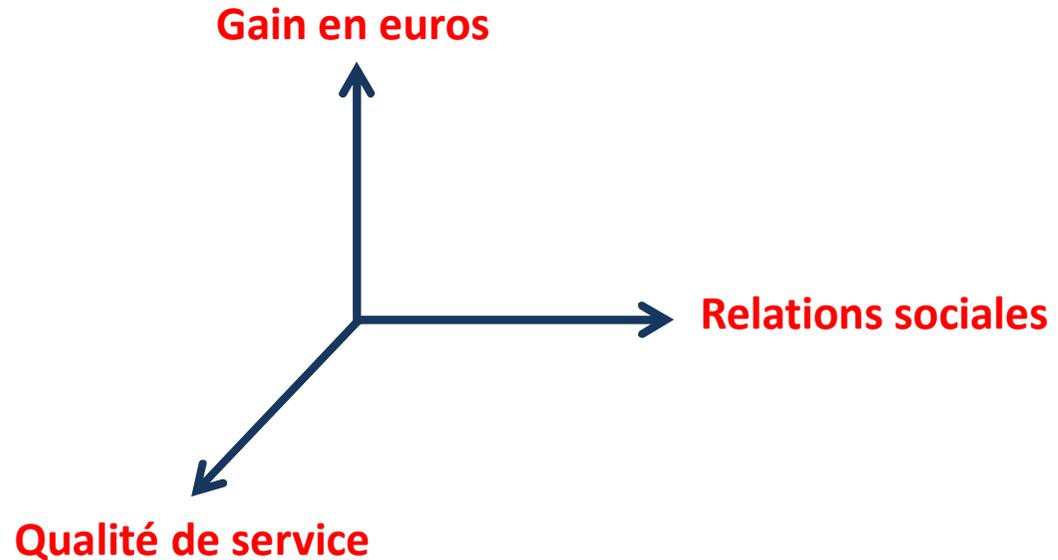
Philippe Morignot

pmorignot@yahoo.fr

OPTIMISATION MULTICRITÈRE

Introduction

- Dans un problème de planification des ressources humaines, on veut maximiser le gain en euros, la qualité de service et les relations sociales.



- Interdépendance des critères : comment trouver la solution optimale ? Et que veut dire « optimale » ?

Formulation

**Programme linéaire
monocritère :**

$$\min (\sum_j c_j x_j)$$

tel que :

$$\forall i , \quad \sum_j a_{i,j} x_j \geq b_i$$

$$\forall j , \quad x_j \geq 0$$

**Programme linéaire
multicritère :**

$$\forall k, \min (F_k = \sum_j c_{j,k} x_j)$$

tel que :

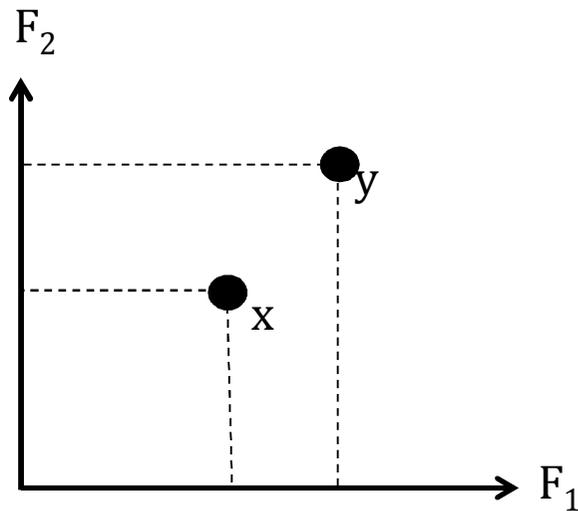
$$\forall i , \quad \sum_j a_{i,j} x_j \geq b_i$$

$$\forall j , \quad x_j \geq 0$$

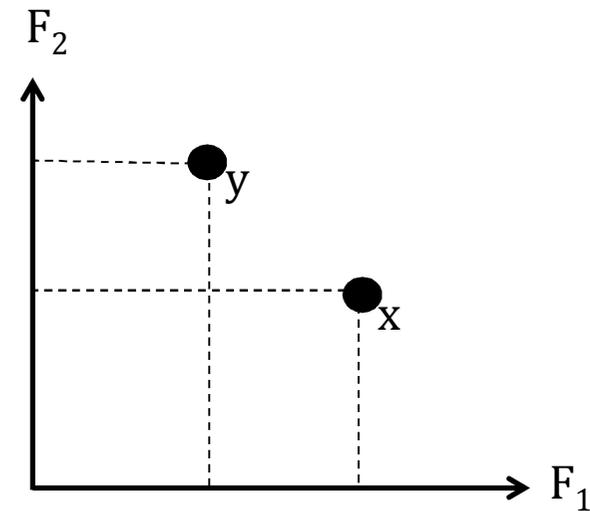
Définition (1/2)

- Pour un programme linéaire de minimisation, la solution x domine la solution y ssi

$$\forall k, F_k(x) \leq F_k(y)$$

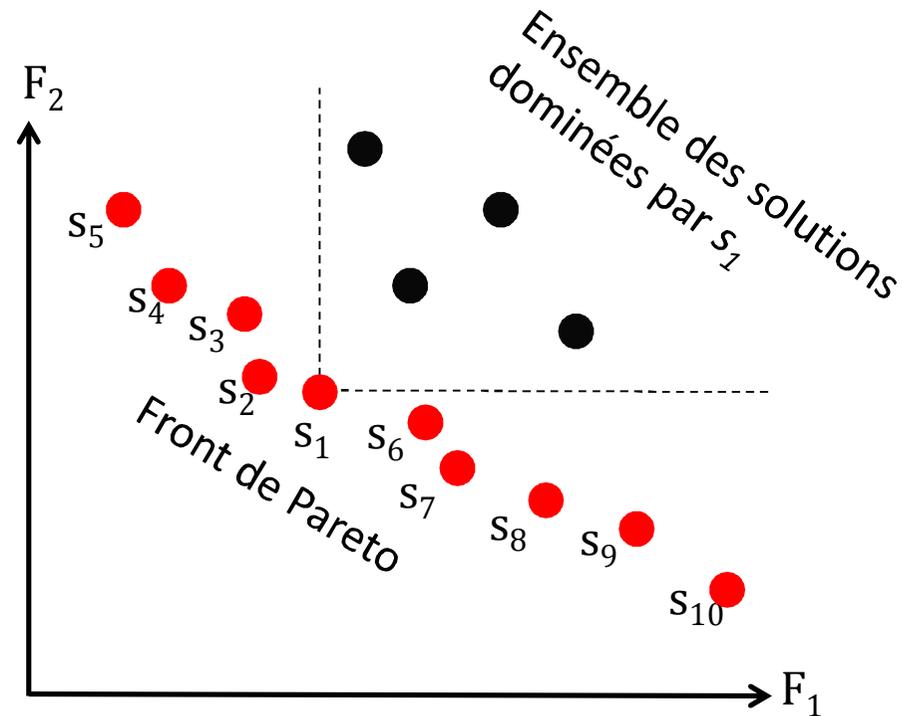


x domine y



x ne domine pas y

Définition (2/2)



- Le **front de Pareto** est l'ensemble des solutions dominantes (ou efficaces) du problème.
- Comment les départager ?

Méthodes de départage (1/3)

Agrégation des critères

- Une nouvelle fonction de coût F est la combinaison linéaire des critères F_i :

$$\min F(x) \quad \text{avec} \quad F(x) = \sum_i p_i F_i(x)$$

et résolution monocritère avec F .

- Avantage : simple.
- Désavantages : on n'atteint pas forcément un optimum. Et comment choisir les coefficients p_i ?

Méthodes de départage (2/3)

Sérialisation

- Algorithme : (1) fixer un ordre sur les critères, et (2) les résoudre un à un dans cet ordre.
- Hypothèse (programme linéaire non compensatoire) : la qualité du critère k ne peut pas être dévalorisée par le critère $k+1$.
- Avantages : efficacité ; décompose le problème en sous-problèmes.
- Désavantages : K programmes linéaires à résoudre ; la solution optimale obtenue varie suivant l'ordre.

Méthodes de départage (3/3)

Tableau des gains

1. Maximiser séparément chacun des K critères F_k :

Critère	Solution 1	Solution 2	...	Solution K
F_1	$F_{1,1}$	$F_{1,2}$...	$F_{1,K}$
F_2	$F_{2,1}$	$F_{2,2}$...	$F_{2,K}$
...
F_K	$F_{K,1}$	$F_{K,2}$...	$F_{K,K}$

Le point idéal $(F_{1,1}, F_{2,2}, \dots, F_{K,K})$ est inaccessible parce que tous les critères $F_{i,i}$ ne peuvent être atteints simultanément.

2. Minimiser l'écart maximum à ce point idéal :

$$\begin{array}{ll} \min \lambda \text{ tel que :} & // \lambda \in [0,1] \\ \left\{ \begin{array}{ll} \forall k, & F_k = \sum c_{j,k} x_j \\ \forall i, & x_i \geq 0 \\ \forall k, & F_{kk} - F_k \leq \lambda (F_{kk} - \min_i (F_{k,i})) \end{array} \right. & // \text{Les K fonctions objectifs.} \\ & // \text{En normant les critères } F_{i,j} \end{array}$$

DUALITÉ

Définitions (1/2)

- Pour un programme linéaire $\min f(x)$ tel que $x \in X$, on a $f(x_{\text{opt}})$ pour un certain $x_{\text{opt}} \in X$.
Que se passe-t-il si on modifie légèrement les données d'entrée ?
- Une contrainte est dite **saturée** ssi sa variable d'écart est nulle.
- La **valeur duale** associée à une contrainte est la quantité dont varie la fonction de coût / objectif quand on fait varier d'une unité la valeur du second membre.

Définitions (2/2)

Programme primal :

$\min C X$ tel que

$$A X \geq B$$

$$X \geq 0$$

$${}^t x_j$$

$$c_j$$

$$a_{i,j}$$

\cong

$$b_i$$

Programme dual :

$\max {}^t B Y$ tel que

$${}^t A Y \leq {}^t C$$

$$Y \geq 0$$

$${}^t y_i$$

$$b_i$$

$$a_{j,i}$$

\cong

$$c_j$$

Propriétés

Le problème dual du problème dual est le problème primal.

<i>Programme primal</i>	<i>Programme dual</i>
Programme sans solution	Programme non borné
Programme non borné	Programme sans solution
Programme optimum	Programme optimum
Coût opt. primal = coût opt. dual	Coût opt. dual = coût opt. primal
Contrainte i saturée	$y_i \geq 0$
Contrainte i non saturée	$y_i = 0$
$x_j \geq 0$	Contrainte j saturée
$x_j = 0$	Contrainte j non saturée

Démonstrations dans le cours IFT2505 de Jacques Ferland.

Conséquences

- Ligne 4 : si on cherche la valeur optimale de la fonction de coût / objectif, résoudre le programme linéaire primal est équivalent à résoudre le programme linéaire dual.
- Pour un programme linéaire primal, l'algorithme du simplexe donne à l'optimum les valeurs des variables duales.
 - Inutile de résoudre le programme linéaire dual pour connaître les variables duales à l'optimum.
 - Si le programme dual est plus facile à résoudre que le programme primal, le résoudre et prendre ses variables duales.
- Modifier une contrainte b_i dans le programme linéaire primal est équivalent à modifier un coût b_i dans la fonction de coût / objectif du programme linéaire dual.
 - Modifier une contrainte peut rendre le programme linéaire primal sans solution (non réalisable), alors que changer un coût est simple.

Exemple (1 / 2)

On a 3h sur une machine 1 et 5h sur une machine 2. Pour produire 1 tonne de produit A, il faut 2h sur la machine 1 et 1h sur la machine 2. Pour produire 1 tonne de produit B, il faut 1h sur la machine 1 et 2h sur la machine 2. Le produit A se vend 600€/tonne et le produit B 700 €/tonne. Quelle est la production optimale ?

Programme primal :

$$\begin{cases} 2x + y \leq 3 \\ x + 2y \leq 5 \end{cases}$$

$$\max(6x + 7y)$$

Programme dual :

$$\begin{cases} 2u + v \geq 6 \\ u + 2v \geq 7 \end{cases}$$

$$\min(3u + 5v)$$

Exemple (2 / 2)

$$\begin{cases} 2x + y \leq 3 \\ x + 2y \leq 5 \end{cases}$$

$\max(6x + 7y)$

$$\begin{cases} 2u + v \geq 6 \\ u + 2v \geq 7 \end{cases}$$

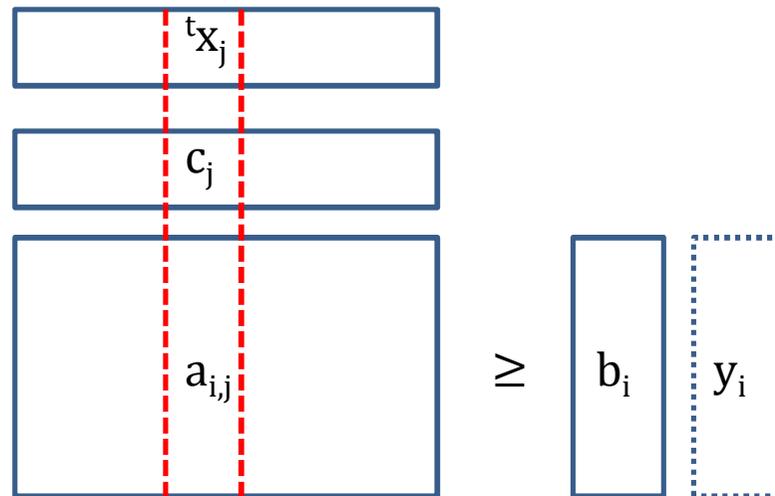
$\min(3u + 5v)$

- u et v sont les coûts horaires de non-exploitation, parce que $3u+5v$ est une quantité qui à l'optimum vaut l'optimum du primal et qui est exprimée en € (puisque 6 et 7 sont des €) et parce que 3 et 5 sont les disponibilités (en heures) des machines.
- $2u+v \geq 6$ signifie que ne pas utiliser la machine 1 pendant 2h et la machine 2 pendant 1h coûte au moins 600€, puisque sinon on pourrait produire 1 tonne de produit A. Idem avec $u + 2v$ pour la machine 2.
- $(u, v)_{\text{opt}} = (5/3, 8/3)$ Chaque heure inutilisée de la machine 1 coûte 166,7€ et 267€ pour la machine 2.
- Le programme dual mesure l'impact de la mauvaise utilisation des ressources.

GÉNÉRATION DE COLONNES

Définition

- Le coût réduit Δ_j d'une variable x_j est : $\Delta_j = c_j - \sum_i a_{i,j} y_i$



- Dans un programme linéaire primal (avec maximisation) :
 - Dans l'algorithme du simplexe, si $\Delta_j > 0$, alors la variable x_j entre en base.
 - Le critère d'arrêt de l'algorithme du simplexe est : $\forall j, \Delta_j \leq 0$

Principe

- Décomposer le problème en un problème maître et un problème esclave.
- Au début, le problème maître ne contient aucune variable (il est le plus simple possible) et utilise la programmation par but (pour qu'il y ait toujours une solution).
- 1 variable de décision = 1 colonne.
- Utilité : quand le nombre de variables est grand et que beaucoup de variables sont nulles.

Algorithme (1/2)

1. Résoudre le programme linéaire maître.
2. TANT QUE $\exists j, \Delta_j \leq 0$, FAIRE reporter les variables duales y_i à l'optimum dans le problème esclave :
 - Si une variable duale y_i est nulle, la contrainte i peut être améliorée.
 - Si une variable duale y_i est positive, la contrainte i est saturée.
3. Résoudre le problème esclave.
4. Reporter les solutions du problème esclave dans le problème maître.

Algorithme (2/2)

Problème maître

< 10 000 variables

t_{x_j}

c_j

$a_{i,j}$

y_i

Problème esclave

Virtuellement > 10 000 variables.
Créées à la demande.

\geq

b_i

x_j

Tant que $\exists j, \Delta_j < 0$

En variables entières (1 / 2)

- Comment résoudre un grand programme linéaire en variables entières ?
- 1^e idée : branch & bound avec génération de colonnes sur chaque nœud pour trouver la solution relâchée.
- Désavantage :
 - À chaque nœud du B&B, un grand nombre d'algorithmes du simplexe doivent être lancés.
 - Beaucoup trop long en général, car le nombre de variables est grand.

En variables entières (2 / 2)

- Algorithme de **branch & cut** :
 - Comme l'algorithme de branch & bound mais, dans un nœud, ne développer qu'un seul nœud-fils, et non considérer les 2 nœuds-fils possibles.
 - A chaque nœud, choisir la variable la plus proche d'une valeur entière et la fixer à cette valeur.
- Algorithme plus rapide que la génération de colonnes en variables entières, mais solution sous-optimale.

Conclusion

- Dans un programme linéaire multicritère, la dominance et le front de Pareto permettent de représenter les solutions efficaces du problème. Trouver un compromis par agrégation, sérialisation et tableau des gains.
- Les variables duales permettent de détecter quand une contrainte ne peut plus être améliorée.
- Pour un programme linéaire contenant beaucoup de variables, le séparer en un problème maître et un problème esclave et itérer (plus efficace qu'avec un algorithme du simplexe).