UNIVERSITÉ DE NEUCHATEL

FACULTÉ DE DROIT ET DES SCIENCES ÉCONOMIQUES

Algorithmes d'optimisation multicritère leur mise au point et implantation sur ordinateur

THÈSE

PRÉSENTÉE À LA FACULTÉ DE DROIT ET DES SCIENCES ÉCONOMIQUES
DE L'UNIVERSITÉ DE NEUCHÂTEL
POUR OBTENIR LE GRADE DE DOCTEUR ÈS SCIENCES ÉCONOMIQUES

PAR

Abdelkader BELKONIENE

Editions DELVAL, Cousset (Fr)

1984

Monsieur Abdelkader BELKONIENE est autorisé à imprimer sa thèse de doctorat ès sciences économiques intitulée :
«**Algorithmes d'optimisation multicritère : leur mise au point et implantation sur ordinateur**».

Il assume seul la responsabilité des opinions énoncées.

Neuchâtel, le 20 juillet 1984

Le doyen
de la Faculté de Droit et
des sciences économiques

François Knoepfler

Impression : Offset Sottaz, Montagny-les-Monts (FR)

Algorithmes d'optimisation multicritère:
leur mise au point et implantation sur ordinateur.

REMERCIEMENTS

Je tiens à remercier le professeur Alfred Strohmeier, directeur de thèse, qui, par ses conseils avisés et ses encouragements, m'a aidé à mener à bien cette étude. Ma reconnaissance s'adresse également à Madame H. Badan qui, par son travail minutieux de dactylographie et sa patience émérite, a réussi à donner forme à cette recherche.

TABLE DES MATIERES

RESUME

Les deux premiers chapitres de ce travail sont consacrés au rappel de certains éléments de la programmation convexe, notamment la convexité dans le premier, l'algorithme du simplexe ainsi que l'algorithme du gradient dans le deuxième.

Dans le troisième chapitre, nous abordons les aspects mathématiques de la programmation multiobjectif. Nous donnons la formulation d'un problème multiobjectif, nous définissons les notions de dominance et d'efficacité et nous énonçons un certain nombre de propositions permettant la caractérisation de solutions et de directions efficaces. La fin du chapitre est réservée à la notion de solutions de compromis qui sont les solutions d'un programme multiobjectif obtenues en minimisant la distance entre le vecteur des fonctions objectifs et un vecteur de goals (ici le vecteur des valeurs idéales). Les distances utilisées constituent une famille définie par $L_p = (\sum_{i=1}^{k} | f_i^* - f_i(x) | ^p)^{1/p}$, $1 \leq p \leq \infty$. Pour chacune d'elle, nous examinons le nombre et l'efficacité des solutions générées. Nous montrons, entre autre, que la solution obtenue lorsqu'on utilise L_∞ n'est pas nécessairement efficace mais que, par contre, celle qu'on obtient en utilisant la combinaison convexe $\varepsilon L_1 + (1-\varepsilon) L_\infty$, où ε est un nombre positif assez petit, est une solution efficace du programme multiobjectif. Ce résultat est utilisé dans la méthode STEM, décrite au chapitre 4, pour la génération d'une solution de compromis efficace.

La description des sept algorithmes d'optimisation multicritère que nous avons implantés sur ordinateur (VAX 11/780) est donnée au chapitre 4. En premier lieu, nous trouvons l'exposé des deux méthodes discrètes (l'ensemble des solutions réalisables est fini) : ELECTRE I et

ELECTRE II, qui sont basées toutes les deux sur la notion de surclassement développée par B. Roy. La première, ELECTRE I, donne une partition de l'ensemble des actions en deux sous-ensembles, l'un constitué des "bonnes" actions et l'autre des moins "bonnes". La deuxième, ELECTRE II, donne un classement des actions des "meilleures" aux "pires". Nous trouvons ensuite la description complète de 5 algorithmes du type continu (l'ensemble des solutions réalisables est infini), dont notre proposition l'algorithme interactif du simplexe. Des modifications majeures ont été apportées aux autres algorithmes, notamment à la méthode de Zionts-Wallenius en ce qui concerne la recherche des directions et solutions efficaces, et à la méthode STEM pour la génération d'une solution de compromis efficace.

Enfin le chapitre 5 est réservé au package qui est un ensemble de programmes permettant l'utilisation sur ordinateur des algorithmes traités au chapitre 4. La première partie de ce chapitre décrit le fonctionnement du package ainsi que l'organisation et la structure des fichiers de données, quant à la deuxième, elle contient le mode d'emploi des programmes illustré par des exemples économiques commentés.

1. RAPPEL MATHEMATIQUE

Dans ce chapitre, nous mettons en place les notations que nous utiliserons tout au long de ce travail et nous rappelons la notion de convexité et les conditions de Kuhn et Tucker nécessaires à la programmation convexe.

1.1 Notations
1.2 Ensembles convexes : définitions et propriétés
1.3 Fonctions convexes, fonctions concaves : définitions et propriétés
1.4 Points et directions extrêmes d'un polyèdre convexe : définitions et caractérisation
1.5 Conditions de Kuhn et Tucker

1.1 Notations

Les matrices et les ensembles sont notés par des lettres majuscules.

Exemples : les matrices A, B, N, E

les ensembles C, D

Une lettre minuscule - simplement indicée ou doublement indicée - représente une ligne, une colonne ou un élément de la matrice de même lettre majuscule.

Exemples : a_i est la ième ligne de la matrice A.

a^j est la jème colonne de la matrice A.

a_i^j est l'élément de la ième ligne et jème colonne de la matrice A.

Toute lettre en minuscule et non indicée représente un vecteur. La même lettre indicée représente une composante du vecteur.

Exemples : x est un vecteur

x_i est la ième composante du vecteur x

Remarque 1 :

Nous nous donnons la possibilité de différencier 2 matrices, 2 ensembles ou 2 vecteurs en rajoutant aux lettres de la convention ci-dessus un chiffre en indice.

Exemples : A_1 et A_2 sont 2 matrices distinctes.

S_1 et S_2 sont 2 ensembles distincts.

a_{1i} est la ième ligne de A_1

a_{2i} est la ième ligne de A_2

Remarque 2 :

Si x_1 et x_2 sont 2 vecteurs distincts, on le précisera dans le texte pour qu'il n'y ait pas de confusion entre les vecteurs x_1 et x_2 et les composantes x_1 et x_2 du vecteur x.

Les ièmes composantes de tels vecteurs sont notées respectivement x_{1i} pour x_1 et x_{2i} pour x_2.

Autres conventions d'écriture

Soient un ensemble D contenu dans R^n et 2 vecteurs y_1, y_2 de R^n par $y_1 \in y_2 + D$ nous entendons que y_1 peut s'écrire comme somme de y_2 et d'un élément de D : $y_1 = y_2 + d$, $d \in D$.

L'ordre suivant sera adopté pour les vecteurs de R^n :

$$x = y \iff x_i = y_i \qquad i=1,\dots,n$$

$$x \geqq y \iff x_i \geqslant y_i \qquad i=1,\dots,n$$

$$x \geqslant y \iff x_i \geqslant y_i \text{ , } i=1,\dots,n \text{ et } x \neq y$$

$$x > y \iff x_i > y_i \qquad i=1,\dots,n$$

Pour alléger la notation, nous noterons le vecteur x de R^n et sa décomposition en 2 vecteurs x_B et x_N respectivement par $x = (x_1, x_2, \dots, x_n)$

et par $x = (x_B\ ,\ x_N)$ au lieu de $x^t = (x_1, x_2, \ldots, x_n)$ et de $x^t = (x_B^t\ ,\ x_N^t)$, où x^t , x_B^t , x_N^t sont les vecteurs lignes x , x_B et x_N. De même, si c et x sont 2 vecteurs de R^n et A une matrice nxn, le produit scalaire $\sum_{i=1}^{n} c_i x_i$ et la multiplication à gauche de A par c seront notés respectivement par cx et par cA au lieu de $c^t x$ et de $c^t A$, où c^t est le vecteur ligne c.

L'abréviation ssi signifie : si et seulement si.

Les nombres entre parenthèses carrées [] qui se trouvent dans le texte renvoient à des références bibliographiques situées à la fin de l'ouvrage.

1.2 Ensembles convexes

Définitions :

1) S, sous-ensemble non vide de R^n, est un ensemble convexe, ssi

$$\left.\begin{array}{l} x_1 \in S \ , \ x_2 \in S \\ \\ \text{et} \quad 0 \leqslant \lambda \leqslant 1 \end{array}\right| \Rightarrow \lambda x_1+(1-\lambda)x_2 \in S$$

2) On appelle combinaison linéaire convexe de k points x_i, $i=1,\ldots,k$ tout point x tel que :

$$x = \sum_{i=1}^{k} \lambda_i x_i \quad , \quad \lambda_i \geqslant 0 \ , \ \sum_{i=1}^{k} \lambda_i = 1$$

Propositions :

Soient S_1 et S_2 deux sous-ensembles convexes de R^n

1) $S_1 \cap S_2$ est convexe

2) $S_1 + S_2 = \{x_1 + x_2 \mid x_1 \in S_1 \ , \ x_2 \in S_2\}$ est convexe

3) $S_1 - S_2 = \{x_1 - x_2 \mid x_1 \in S_1 \ , \ x_2 \in S_2\}$ est convexe

Définition 3 :

Soit S un sous-ensemble non vide de R^n. L'enveloppe convexe $E(S)$ de S est l'ensemble de toutes les combinaisons convexes de S. Donc

$x \in E(S)$ ssi

$$x = \sum_{i=1}^{k} \lambda_i x_i \text{ , avec } \lambda_i \geq 0 \text{ , } \sum_{i=1}^{k} \lambda_i = 1 \text{ et } x_1, x_2, \ldots, x_k \in S$$

Proposition 4 :

E(S) est le plus petit ensemble convexe contenant S.

E(S) est l'intersection de tous les ensembles convexes contenant S.

Définition 4 :

S , *sous-ensemble non vide de* R^n, est *un* polyèdre *convexe* s'il est l'intersection d'un nombre fini de demi-espaces fermés.

Remarque :

Un polyèdre convexe peut être représenté par un nombre fini d'inéquations.

Exemples de polyèdres convexes :

$S_1 = \{x | Ax \leq b\}$

$S_2 = \{x | Ax = b \text{ , } x \geq 0\}$

$S_3 = \{x | Ax > b \text{ , } x \geq 0\}$

A est une matrice mxn, b un vecteur de dimension m et x un vecteur de dimension n.

Définition 5 :

Un ensemble non vide C de R^n est un cône ssi pour $\forall x \in C$, on a : $\alpha x \in C$,

$\alpha \geq 0$. Si en plus C est un ensemble convexe alors C est dit <u>cône convexe</u>.

Définition 6 :

Soit un ensemble C non vide de R^n, l'ensemble

$C^* = \{y \in R^n \mid yx \leq 0 \ , \ x \in C\}$ est le cône polaire de C.

Propositions :

Soient C , C_1 et C_2 3 ensembles de R^n.

5) Le cône polaire C^* de C est un cône convexe fermé.

6) Si l'on désigne par C^{**} le cône polaire de C^*, on a $C \subset C^{**}$.

7) $C_1 \subset C_2$ entraîne $C_2^* \subset C_1^*$.

1.3 Fonctions concaves, fonctions convexes

Définitions :

Soit une fonction $f: S \to R$, où S est un ensemble convexe non vide de R^n.

1) f est appelée fonction concave sur S si

$$\forall x_1, \forall x_2 \in S \quad \text{et} \quad \lambda \in [0,1],$$

$$f(\lambda x_1 + (1-\lambda)x_2) \geq \lambda f(x_1) + (1-\lambda) f(x_2)$$

2) f est appelée fonction strictement concave sur S si

$$\forall x_1, \forall x_2 \in S \ , \ x_1 \neq x_2 \quad \text{et} \ \lambda \in]0,1[$$

$$f(\lambda x_1+(1-\lambda)x_2) > \lambda f(x_1)+(1-\lambda)f(x_2)$$

3) La fonction $f:S\to R$ est <u>convexe</u> (strictement convexe) sur S ssi
   $-f$ est concave (strictement concave) sur S.

<u>Propositions</u> :

Soient S un ensemble non vide convexe et ouvert de R^n et $f:S\to R$ une fonction différentiable sur S.

1) f est une fonction concave ssi :

$$\bar{x} \in S \;;\; f(x) \leqslant f(\bar{x}) + \nabla f(\bar{x})(x-\bar{x}) \text{ pour tout } x \in S$$

2) f est strictement concave ssi :

$$\bar{x} \in S \;,\; \forall x \in S \text{ tels que } x \neq \bar{x}, \text{ on a}$$

$$f(x) < f(\bar{x}) + \nabla f(\bar{x})(x-\bar{x})$$

<u>Propositions</u> :

Soient S un ensemble convexe non vide de R^n, une fonction $f:S\to R$ et $\bar{x}\in S$ une solution optimale locale du problème Max $f(x)$, $x\in S$.

3) Si f est une fonction concave, $\bar{x}$ est une solution optimale globale du problème.

4) Si f est une fonction strictement concave, $\bar{x}$ est l'unique solution optimale globale du problème.

1.4 Points et directions extrêmes d'un polyèdre convexe

Définitions :

A est une matrice mxn et b un vecteur de dimension m.
L'ensemble $P = \{x \in R^n \mid Ax=b \ , \ x \geq 0\}$ est un polyèdre convexe.

1) Un point $x \in P$ est un point extrême de P, si :

$$\forall x_1, x_2 \in P \quad , \quad \forall \alpha \in [0,1]$$

$$x = \alpha x_1 + (1-\alpha)\, x_2 \quad \Rightarrow \quad x = x_1 = x_2$$

ou autrement dit : $x \in P$ est un point extrême de P, s'il ne peut pas s'écrire comme une combinaison convexe de 2 points x_1 et x_2 de P.

2) Un vecteur non nul $d \in R^n$ est une direction réalisable en un point $x \in P$ ssi il existe un scalaire $\bar{\alpha}>0$ tel que

$$x+\alpha d \in P \quad , \quad \alpha \in [0,\bar{\alpha}]$$

3) Une direction d réalisable en un point x de P est une direction extrême de P.
S'il existe des directions réalisables d_1 et d_2 et des scalaires strictement positifs α_1 et α_2 tels que $d = \alpha_1 d_1 + \alpha_2 d_2$, alors d_1 et d_2 sont parallèles et ont même direction, c-à-d il existe $\alpha > 0$ tel que $d_1 = \alpha d_2$.

Proposition 1 :

Caractérisation d'un point extrême

Notations :

Supposons que la matrice A soit de rang m.

Décomposons A et x de la façon suivante :

$$A = (B\ N) \quad \text{et} \quad x = (x_B\ ,\ x_N)$$

où B est une matrice mxm de rang m , donc inversible.

N est une matrice mx(n-m).

x_B est un vecteur de dimension m correspondant à B.

et x_N est un vecteur de dimension n-m correspondant à N.

Les relations $Ax = b$ et $x \geqslant 0$ peuvent alors s'écrire

$$Bx_B + Nx_N = b \quad \text{et} \quad x_B \geqslant 0 \quad , \quad x_N \geqslant 0.$$

Conclusion :

$x \in P$ est un point extrême de P ssi :

A peut être décomposée en $A = (B\ N)$ tel que

$$x_B = B^{-1}b \geqslant 0 \quad \text{et} \quad x_N = 0.$$

où (B^{-1} est la matrice inverse de B).

Définitions :

4) B est appelée base relative (associée) au point x.

5) Les composantes des vecteurs x_B et x_N sont appelées respectivement variables de base et variables hors base.

6) Lorsque la relation $B^{-1}b \geqslant 0$ est vérifiée à l'inégalité stricte, le point x est dit non dégénéré.

Remarque :

Le nombre de points extrêmes de P est fini car il est inférieur ou égal à $\binom{n}{m} = \frac{n!}{m!(n-m)!}$ qui est le nombre maximum de choix possibles de m colonnes parmi n colonnes de A pour former la matrice B.

Proposition 2 :

Un vecteur non nul d est une direction extrême de P ssi A peut être décomposée en $A = (B\ N)$ tel que $B^{-1}a^j \leqslant 0$ et $d = \alpha\begin{pmatrix} -B^{-1}a^j \\ e^j \end{pmatrix}$, $\alpha > 0$ où a^j est la jème colonne de N et e^j la jème colonne de la matrice unité E de dimension (n-m).

1.5 Conditions de Kuhn et Tucker

Dans ce paragraphe, nous allons considérer le programme :

$$(P_r) \quad \begin{aligned} &\text{Min } f(x) \\ &g_i(x) \geqslant 0 \qquad i=1,\dots,m \\ &x \geqslant 0 \end{aligned}$$

où x est un vecteur de dimension n.

Définition : (conditions de Kuhn et Tucker)

Soit $\bar{x}$ une solution réalisable du programme (P_r), c'est-à-dire un vecteur qui vérifie $\bar{x} \geqslant 0$ et $g_i(\bar{x}) \geqslant 0$ pour $i=1,\dots,m$.

Si la fonction f et les fonctions $g_i, i=1,\dots,m$ sont différentiables en $\bar{x}$, on peut définir les conditions de Kuhn et Tucker au point $\bar{x}$ qui sont :

il existe $\lambda \in R^n$, $\lambda \geqslant 0$

et $u_i \geqslant 0$, $i=1,\dots,m$

tel que :

(1) $\nabla f(\bar{x}) - \sum_{i=1}^{m} u_i \nabla g_i(\bar{x}) - \lambda = 0$

(2) $u_i g_i(\bar{x}) = 0 \qquad i=1,\dots,m$

(3) $\lambda\bar{x} = 0$

En tenant compte de (1) et (3), on peut récrire ces conditions sous une forme équivalente :

$$\text{(1 bis)} \quad \nabla f(\bar{x}) - \sum_{i=1}^{m} u_i \nabla g_i(\bar{x}) \geqslant 0$$

$$\text{(2 bis)} \quad u_i g_i(\bar{x}) = 0 \qquad i=1,\dots,m$$

$$\text{(3 bis)} \quad (\nabla f(\bar{x}) - \sum_{i=1}^{m} u_i \nabla g_i(\bar{x}))\bar{x} = 0$$

<u>Proposition</u> (condition nécessaire pour un optimum local) :

Supposons que $\bar{x}$ est un minimum local et que la fonction f et les fonctions g_i , i=1,...,m sont différentiables.

Notons par I l'ensemble des indices correspondant aux contraintes serrées :

$$I = \{i \mid g_i(\bar{x}) = 0\}$$

Alors si les différentielles $\nabla g_i(\bar{x})$, $i \in I$, sont linéairement indépendantes, les conditions de Kuhn et Tucker sont vérifiées au point $\bar{x}$.

<u>Proposition</u> (condition suffisante pour un optimum global) :

Si f est convexe et différentiable en $\bar{x}$, si les fonctions g_i , i=1,...,m sont concaves et différentiables en $\bar{x}$ et si les conditions de Kuhn et Tucker sont vérifiées en $\bar{x}$, alors $\bar{x}$ est un minimum global.

2. PROGRAMMATION LINEAIRE ET NON LINEAIRE

Les algorithmes du simplexe et du gradient réduit sont à la base de certaines méthodes que nous allons décrire dans le chapitre 4. Pour cette raison, nous avons réservé ce chapitre à la description complète de ces deux algorithmes.

2.1 Algorithme du simplexe
2.2 Algorithme du gradient réduit
2.3 Oualité de Lagrange

2.1 Algorithme du simplexe

Notations :

Soit le programme linéaire :

$$(L_p) \qquad \begin{aligned} &\text{Max } cx \\ &Ax = b \\ &x \geqslant 0 \end{aligned}$$

où A est une matrice mxn, x un vecteur de dimension n, b un vecteur de dimension m et c un vecteur de dimension n.

Préliminaires :

$P = \{x \in R^n \mid Ax=b \, , \, x \geqslant 0\}$ est appelé l'ensemble des solutions réalisables. Soit $\bar{x}$ un point extrême de P qui n'est pas dégénéré. Nous savons qu'on peut alors décomposer A tel que :

$$A = (B \; N) \quad ,$$

$$(1) \qquad \bar{x}_B = B^{-1}b > 0 \quad \text{et} \quad \bar{x}_N = 0$$

Appelons J l'ensemble des indices des variables de base et $\bar{J}$ l'ensemble des indices des variables hors base. Si nous décomposons le vecteur c en c_B et c_N de mêmes dimensions respectives que $\bar{x}_B$ et $\bar{x}_N$, nous pouvons écrire la fonction objectif sous la forme :

$$cx = c_B x_B + c_N x_N$$

Considérons maintenant une solution réalisable x.

En tenant compte des relations ci-dessus, nous pouvons écrire :

$$Ax = Bx_B + Nx_N = b \quad \text{avec} \quad x_B \geqslant 0 \quad , \quad x_N \geqslant 0$$

Comme B est inversible, nous pouvons tirer x_B en fonction de x_N

(2) $$x_B = B^{-1}b - B^{-1}Nx_N$$

La valeur de la fonction objectif au point x devient

(3) $$cx = c_Bx_B+c_Nx_N = c_B(B^{-1}b-B^{-1}Nx_N)+c_Nx_N = c_BB^{-1}b+(c_N-c_BB^{-1}N)x_N$$

et au point $\bar{x}$ sa valeur vaut :

(4) $$c\bar{x} = c_BB^{-1}b \quad , \quad \text{car} \quad \bar{x}_B = B^{-1}b \quad \text{et} \quad \bar{x}_N = 0$$

<u>Condition suffisante d'optimalité</u> :

En tenant compte des relations (3) et (4), nous obtenons :

$$cx = c\bar{x} + (c_N-c_BB^{-1}N)x_N$$

Comme $x_N \geqslant 0$, l'inégalité

(5) $$c_N - c_BB^{-1}N \leqslant 0$$

est une condition suffisante pour que

$$c\bar{x} \geqslant cx \qquad \forall\ x \in P$$

ou autrement dit pour que $\bar{x}$ soit une solution optimale.

Principe de l'algorithme

L'algorithme examine itérativement des points extrêmes. Dès que la condition suffisante d'optimalité énoncée ci-dessus est vérifiée, on arrête. Sinon on va chercher un autre point extrême qui a une valeur supérieure pour la fonction objectif.

Amélioration de la valeur de la fonction objectif

Si $\bar{x}$ n'est pas solution optimale, la relation (5) n'est pas vérifiée, c'est-à-dire qu'il existe un indice $j \in \bar{J}$ tel que

$$(6) \qquad c_j - c_B B^{-1} a^j > 0$$

Notons alors par Y la matrice $B^{-1}N$ et par E la matrice unité de dimension (n-m). Nous savons que la matrice D des directions extrêmes de P en $\bar{x}$ peut s'écrire :

$$D = \binom{-Y}{E}$$

Soit d^j la jème colonne de D et considérons le point

$$x = \bar{x} + \alpha d^j \quad \text{avec} \quad \alpha > 0 \quad .$$

La valeur de la fonction objectif en ce point vaut :

$$cx = c(\bar{x}+\alpha d^j) = c\bar{x} + \alpha(c_j - c_B B^{-1} a^j) \quad .$$

Comme α et $c_j - c_B B^{-1} a^j$ sont strictement positifs, nous avons

$$cx > c\bar{x}$$

Il faut maintenant envisager deux cas. Dans le premier, le programme linéaire n'a pas de solution optimale finie - on dit qu'il a une solution optimale infinie - alors que dans le second cas, nous pourrons trouver un point extrême qui améliore la valeur de la fonction objectif.

Premier cas :

$$y^j = B^{-1}a^j \leq 0$$

On va alors prouver que x est solution réalisable pour tout α positif ou nul. Pour ce faire, il suffit de montrer que x est à composantes positives quel que soit α.

Nous avons

$$x = \bar{x} + \alpha d^j$$

ou en décomposant d^j

$$x = \bar{x} + \alpha\begin{pmatrix} -y^j \\ e^j \end{pmatrix}$$

et comme $\bar{x} \geq 0$, $y^j \leq 0$ et $e^j \geq 0$

notre affirmation est prouvée.

La relation suivante montre que la fonction objectif tend vers l'infini quand α tend vers l'infini :

$$cx(\alpha) = c\bar{x} + \alpha(c_j - c_B B^{-1}a^j)$$

car d'aprés (6) $(c_j - c_B B^{-1}a^j) > 0$

Deuxième cas :

$$y^j \nleq 0$$

Notons par $\bar{b}$ le vecteur $B^{-1}b$. Nous avons alors

$$\bar{x}_B = \bar{b}$$

et
$$x = \bar{x} + \alpha\binom{-y^j}{e^j} = \binom{\bar{b}-\alpha y^j}{\alpha e^j}$$

pour que x soit solution réalisable, c'est-à-dire $x \geqslant 0$, il suffit que α soit choisi de façon à ce que les relations suivantes soient vérifiées :

$$\forall i \in J \qquad \bar{b}_i - \alpha y_i^j \geqslant 0$$

En choisissant α par

$$\alpha = \min_{i \in J} \left\{ \frac{\bar{b}_i}{y_i^j} \quad , \quad y_i^j > 0 \right\}$$

et en notant r l'indice où le minimum est atteint, on peut calculer le nouveau point extrême et la base associée :

$$x = \binom{\bar{x}_B}{\bar{x}_N} + \frac{\bar{b}_r}{y_r^j}\binom{-y^j}{e^j} = \binom{\bar{b}}{0} + \frac{\bar{b}_r}{y_r^j}\binom{-y^j}{e^j}$$

ou en explicitant :

$$x_{B_i} = \bar{b}_i - \frac{\bar{b}_r}{y_r^j}\, y_i^j \neq 0 \qquad \text{pour } i \in J \text{ , } i \neq r$$

$$x_{B_r} = \bar{b}_r - \frac{\bar{b}_r}{y_r^j}\, y_r^j = 0$$

$$x_{N_\ell} = 0 \quad \text{pour} \quad \ell \in \bar{J} \;,\quad \ell \neq j$$

$$x_{N_j} = \frac{\bar{b}_r}{y_r^j}$$

En supposant que les éléments de J soient 1,2,...,m , la nouvelle base $\bar{B}$ peut s'écrire

$$\bar{B} = (a^1\ a^2\ \ldots\ a^{r-1}\ a^j\ a^{r+1}\ \ldots\ a^m)$$

La colonne a^r a donc été remplacée par la colonne a^j de N et dans $x_{\bar{B}}$ la variable de base x_r a été remplacée par la variable hors base x_j .
Montrons maintenant comment on peut obtenir $\bar{B}^{-1}$ à partir de B^{-1}.

On a

$$a^j = BB^{-1}a^j = By^j$$

ou en explicitant

$$a_\ell^j = \sum_{i=1}^m a_\ell^i\, y_i^j = y_r^i\, a_\ell^r + \sum_{i\neq r}^m a_\ell^i\, y_i^j \qquad \ell=1,\ldots,m$$

d'où

$$a_\ell^r = \frac{1}{y_r^j}\,(a_\ell^j - \sum_{i\neq r}^m a_\ell^i\, y_i^j) \qquad \ell=1,\ldots,m$$

$$a^r = -\frac{y_1^j}{y_r^j}\,a^1 - \frac{y_2^j}{y_r^j}\,a^2 - \ldots - \frac{y_{r-1}^j}{y_r^j}\,a^{r-1} + \frac{1}{y_r^j}\,a^j - \frac{y_{r+1}^j}{y_r^j}\,a^{r+1} - \ldots - \frac{y_m^j}{y_r^j}\,a^m$$

ou

$$a^r = \bar{B}\begin{pmatrix} -\,y_1^j \,/\, y_r^j \\ \vdots \\ 1 \,/\, y_r^j \\ \vdots \\ -\,y_m^j \,/\, y_r^j \end{pmatrix}$$

Notons E_r la matrice unité de dimension m où on a remplacé la rième colonne par le vecteur colonne :

$$\begin{pmatrix} -y_1^j / y_r^j \\ \vdots \\ 1 / y_r^j \\ \vdots \\ -y_m^j / y_r^j \end{pmatrix}$$

Nous pouvons écrire :

$$B = \bar{B} E_r$$

$$\bar{B}^{-1} B B^{-1} = \bar{B}^{-1} \bar{B} E_r B^{-1}$$

(7) $$\bar{B}^{-1} = E_r B^{-1}$$

La nouvelle solution x, qui est un point extrême de P, est donnée par les relations :

$$x_{\bar{B}} = \bar{B}^{-1} b = E_r B^{-1} b = E_r \bar{x}_B \quad \text{et} \quad x_{\bar{N}} = 0$$

Résumé de l'algorithme

Condition de départ :

$\bar{x}$ est un point extrême non dégénéré de P et B sa base associée.

Tests d'arrêt :

- calcul du vecteur $c_N - c_B B^{-1} N$.

- si $c_N - c_B B^{-1} N \leqslant 0$, on arrête car $\bar{x}$ est solution optimale.

- sinon, on choisit le plus grand élément positif de $c_N - c_B B^{-1} N$ soit $c_j - c_B B^{-1} a^j$ cet élément.

  si $y^j = B^{-1} a^j \leqslant 0$, on arrête car le programme n'a pas de solution optimale finie.

- sinon on effectue une itération.

Itération :

On détermine l'indice r d'aprés la relation :

$$\alpha = \min_{i \varepsilon J} \left\{ \frac{\bar{b}_i}{y_i^j} \,,\; y_i^j > 0 \right\} = \frac{\bar{b}_r}{y_r^j}$$

Le point extrême est donné par :

$$x_{B_i} = \bar{b}_i - \frac{\bar{b}_r}{y_r^j} y_i^j \qquad i=1,\ldots,m$$

$$x_{N_j} = \frac{\bar{b}_r}{y_r^j}$$

Les autres composantes sont nulles.

La nouvelle base associée $\bar{B}$ s'obtiént en remplaçant la colonne a^r de B par la colonne a^j de N.

Remarque :

Pour des raisons d'efficacité, on calcule directement $\bar{B}^{-1}$ à partir de B^{-1} en utilisant la relation (7).

2.2 Algorithme du gradient réduit

Notations :

On se donne le programme :

$$\begin{aligned} &\text{Min } f(x) \\ &Ax = b \\ &x \geqslant 0 \end{aligned}$$

où A est une matrice mxn, x un vecteur de dimension n, b un vecteur de dimension m. On notera

$P = \{x \in R^n \mid Ax = b \quad , \quad x \geqslant 0\}$ l'ensemble des solutions réalisables.

Préliminaires :

Soit $\bar{x}$ un point de P qui n'est pas dégénéré. Nous savons qu'on peut décomposer A tel que :

$$A = (B \; N)$$

$$\bar{x} = (\bar{x}_B, \bar{x}_N) \quad \text{avec} \quad \bar{x}_B > 0$$

Appelons J l'ensemble des variables de base et $\bar{J}$ l'ensemble des variables hors base. Décomposons le gradient de f, vecteur ligne de dimension n, au point $\bar{x}$ de la manière suivante :

$$\nabla f(\bar{x}) = (\nabla_B f(\bar{x}) \quad , \quad \nabla_N f(\bar{x}))$$

où $\nabla_B f(\bar{x})$ est le gradient de f en $\bar{x}$ relatif aux variables de base $\bar{x}_B$

et $\nabla_N f(x)$ le gradient de f en $\bar{x}$ relatif aux variables hors base $\bar{x}_N$.

Proposition :

Une direction d est une direction réalisable qui diminue la valeur de la fonction objectif (sous-entendu en $\bar{x}$), si :

(1) $\nabla f(\bar{x})d < 0$

(2) $Ad = 0$

(3) $d_j \geq 0$ si $\bar{x}_j = 0$

Recherche d'une direction réalisable qui diminue la valeur de la fonction objectif

Décomposons d en deux vecteurs d_B et d_N de même dimensions respectives que x_B et x_N.

La direction d est réalisable en $\bar{x}$ si les relations (2) et (3) sont vérifiées, c'est-à-dire si

$$Ad = Bd_B + Nd_N = 0$$

et $d_j \geq 0$ si $\bar{x}_j = 0$

ou

(4) $d_B = -B^{-1}Nd_N$

et $d_j \geq 0$ si $\bar{x}_j = 0$

En tenant compte de (4) la relation (1) peut maintenant s'écrire :

$$\nabla f(\bar{x})d = \nabla_B f(\bar{x})d_B + \nabla_N f(\bar{x})d_N = (\nabla_N f(\bar{x}) - \nabla_B f(\bar{x})B^{-1}N)d_N < 0$$

Notons par r_N le vecteur $(\nabla_N f(\bar{x}) - \nabla_B f(\bar{x})B^{-1}N)$. r_N est appelé <u>gradient réduit</u> de f au point $\bar{x}$.

En tenant compte de la relation (4), les conditions pour que d soit une direction réalisable qui diminue la valeur de la fonction objectif s'écrivent :

(5) $\qquad r_N d_N < 0$

(6) $\qquad d_j \geq 0$ si $j \in \bar{J}$ et $\bar{x}_j = 0$

Les conditions ci-dessus sont toujours vérifiées si l'on choisit d_N de la façon suivante :

(7) $$d_j = \left| \begin{array}{l} -r_j \quad \text{si} \quad j \in \bar{J} \quad \text{et} \quad r_j \leq 0 \\ \\ -\bar{x}_j r_j \quad \text{si} \quad j \in \bar{J} \quad \text{et} \quad r_j > 0 \end{array} \right.$$

<u>Vérification des conditions de Kuhn et Tucker</u>

$\bar{x}$ vérifie les conditions de Kuhn et Tucker s'il existe des vecteurs $u = (u_B, u_N) \geq 0$ et v tels que :

(8) $\qquad (\nabla_B f(\bar{x}), \nabla_N f(\bar{x})) + (vB, vN) - (u_B, u_N) = 0$

(9) $\qquad u_B \bar{x}_B = 0 \quad , \quad u_N \bar{x}_N = 0$

Or $\bar{x}_B$ est strictement positif, donc d'après (9) $u_B = 0$

En remplaçant u_B par sa valeur dans (8), on tire :

$$v = \nabla_B f(\bar{x})B^{-1}$$

et $\qquad u_N = \nabla_N f(\bar{x}) - \nabla_B f(\bar{x}) B^{-1} N = r_N$

Donc les conditions de Kuhn et Tucker sont vérifiées en $\bar{x}$ si

$$r_N \geqslant 0$$

et $\qquad r_N x_N = 0$

ou, autrement dit, si en calculant d_N selon la relation (7), on a $d_N = 0$.

<u>Recherche d'un nouveau point qui diminue la valeur de f(x)</u>

Soit d une direction définie par la relation (7).
On sait que $f(\bar{x}) \geqslant f(\bar{x}+\alpha d)$ pour des valeurs de α positives et assez petites.
Par ailleurs $\bar{x}+\alpha d$ est réalisable ssi $\alpha \geqslant 0$ et $\bar{x}+\alpha d \geqslant 0$.
En définissant $\bar{\alpha}$ par les relations suivantes :

$$\bar{\alpha} = \begin{cases} \infty \quad \text{si} \quad d \geqslant 0 \\ \min \{\frac{-\bar{x}_j}{d_j} \,,\, d_j<0, j=1,\dots,m\} \quad \text{si } d \not\geqslant 0 \end{cases}$$

$\bar{x}+\alpha d$ sera réalisable pour tout α compris entre 0 et $\bar{\alpha}$.

Posons maintenant $h(\alpha) = f(\bar{x}+\alpha d)$ avec $0 \leqslant \alpha \leqslant \bar{\alpha}$.

Le point $\bar{x}+\alpha d$ qui nous intéresse est celui pour lequel la valeur de la fonction $h(\alpha)$ est la plus petite possible. On obtient ce point en choisissant α comme solution optimale du programme :

$$\text{Min } h(\alpha) = f(\bar{x}+\alpha d)$$

$$0 \leqslant \alpha \leqslant \bar{\alpha}$$

On notera α^* la solution optimale de ce programme et le nouveau point est donc

$$x = \bar{x}+\alpha^* d$$

Résumé de l'algorithme

Conditions de départ :

$\bar{x}$ est un point non dégénéré de P et B sa base associée.

On calcule $r_N = \nabla_N f(\bar{x}) - \nabla_B f(\bar{x}) B^{-1} N$, puis d_N par :

$$d_j = \begin{cases} -r_j & \text{si } j \in \bar{J} \text{ et } r_j \leqslant 0 \\ -x_j r_j & \text{si } j \in \bar{J} \text{ et } r_j > 0 \end{cases}$$

et d_B par $d_B = -B^{-1} N d_N$.

Test d'arrêt :

On arrête dès que $d = (d_B, d_N) = 0$, car les conditions de Kuhn et Tucker sont alors vérifiées en $\bar{x}$.

Itération :

On cherche α^* en résolvant le programme :

$$\text{Min } f(\bar{x}+\alpha d)$$

$$0 \leqslant \alpha \leqslant \bar{\alpha}$$

où $$\bar{\alpha} = \begin{cases} \infty \text{ si } d \geqslant 0 \\ \min \{\frac{-\bar{x}_j}{d_j} , d_j < 0, j=1,\dots,m\} \text{ si } d \ngeqslant 0 \end{cases}$$

Le nouveau point est donné par :

$$x = \bar{x} + \alpha^* d$$

2.3 Dualité de Lagrange

Soit le programme :

(P_r) Min $f(x)$

$g_i(x) \geq 0 \quad , \quad i=1,\ldots,m$

$h_j(x) = 0 \quad , \quad j=1,\ldots,\ell$

où x est un vecteur de dimension n

La fonction de Lagrange associée à (P_r) est :

$$L(x,u,v) = f(x) - \sum_{i=1}^{m} u_i g_i(x) - \sum_{j=1}^{\ell} v_j h_j(x)$$

On appelle programme dual de (P_r) le programme :

(D) Max $\theta(u,v)$

$u \geqq 0$

où $\theta(u,v) = \inf_x L(x,u,v)$

Nous noterons X l'ensemble des solutions réalisables de (P_r), c'est-à-dire l'ensemble $\{x \mid g_i(x) \geq 0 \ , \ i=1,\ldots,m, \ h_j(x) = 0 \ , \ j=1,\ldots,\ell\}$ et U l'ensemble des solutions réalisables de (D), c'est-à-dire l'ensemble $\{u \mid u \geqq 0\}$.

Proposition 1 :

Soient x une solution réalisable de (P_r) et (u,v) une solution réalisable de (D), alors

$$f(x) \geqslant \theta(u,v)$$

Preuve :

$$\theta(u,v) = \inf_z L\,(z,u,v) \leqslant L\,(x,u,v) \leqslant f(x)$$

car $\quad u_i \geqslant 0\ ,\ g_i(x) \geqslant 0$ pour $i=1,\dots,m$ et

$\quad h_j(x) = 0$ pour $j=1,\dots,\ell$

Proposition 2 :

$$\inf\{f(x)\,|\,x\varepsilon X\} \geqslant \sup\,\{\theta(u,v)\ |\ u \geqslant 0\}$$

Proposition 3 :

Si $\bar{x} \varepsilon X$ et $\bar{u} \varepsilon U$ et s'il existe $\bar{v}$ tel que

$f(\bar{x}) \leqslant \theta(\bar{u},\bar{v})$ *alors* $\bar{x}$ et $(\bar{u},\bar{v})$ *sont respectivement solution* optimale de (P_r) et de (D).

Proposition 4 :

Si $\sup\,\{\theta(u,v)\ |\ u\varepsilon U\} = +\infty$, alors le programme primal (P_r) n'a pas de solution réalisable.

Cas particulier de la programmation linéaire

Soit le programme linéaire :

(P_r) $\quad \text{Min } cx$

$\qquad Ax \geqslant b$

$\qquad x \geqslant 0$

Le programme dual est :

(D) $\quad \text{Max } \theta(u)$

$\qquad u \geqslant 0$

En utilisant la définition de la fonction θ :

$$\theta(u) = \inf_x \{cx-u(Ax-b) \mid x\geqslant 0\} = ub+\inf_x \{(c-uA)x \mid x\geqslant 0\}$$

on déduit : $\theta(u) = \begin{cases} ub & \text{si } c-uA \geqslant 0 \\ -\infty & \text{si } c-uA \not\geqslant 0 \end{cases}$

Donc le programme (D) peut s'écrire :

(D) $\quad \text{Max } ub$

$\qquad uA \leqslant c$

$\qquad u \geqslant 0$

Propositions (relation entre le primal et le dual en programmation linéaire) :

5) Le programme dual du programme dual est égal au programme primal.

6) Si le primal a une solution optimale infinie, alors le dual n'a pas de solution réalisable (on dit qu'il est vide).

7) Si le dual a une solution optimale infinie, alors le primal est vide.

8) Si les deux programmes ont des solutions réalisables, alors les deux programmes ont des solutions optimales $\bar{x}$ et $\bar{u}$ telles que :

$$c\bar{x} = \bar{u}b = \bar{u}A\bar{x}$$

Preuves :

5) se démontre très simplement en écrivant le programme dual du dual.

6) découle des propositions 1 et 2 du cas général.

7) se déduit de 5 et 6.

Prouvons maintenant 8.

Soit $\bar{x}$ la solution optimale du primal, alors $\bar{x}$ vérifie les conditions de Kuhn et Tucker, c'est-à-dire qu'il existe un vecteur $\bar{u} \geqslant 0$ tel que

(1) $c-\bar{u}A \geqslant 0$

(2) $\bar{u}(A\bar{x}-b) = 0$

(3) $(c-\bar{u}A)\bar{x} = 0$

(1) signifie que $\bar{u}$ est une solution réalisable du dual.

En utilisant (2) et (3) on prouve la relation :

$$c\bar{x} = \bar{u}A\bar{x} = \bar{u}b$$

qui permet d'affirmer que $\bar{u}$ est solution optimale de (D), ceci en raison de la proposition 3.

3. PROGRAMMATION MULTIOBJECTIF : ASPECTS MATHEMATIQUES

Dans ce chapitre, nous donnons une synthèse des éléments théoriques de la programmation multiobjectif. Nous définissons d'abord les différentes notions utilisées puis nous donnons un certain nombre de propositions avec leur preuve.

3.1 Formulation du programme et solution idéale : sous ce titre, nous décrivons la formulation mathématique d'un programme multiobjectif et définissons la notion de solution et de valeur idéales

3.2 Structure de dominance et solutions non dominées :

- Structure de dominance : définition et exemples
- Solutions non dominées : définition (cas particulier : solution optimale au sens de Pareto)
- Solutions et directions efficaces : définition et caractérisation

3.3 Solutions de compromis (ou notion du point de mire)

- Distances : définition et propriétés
- Solutions de compromis : définition et propriétés

3.1 Formulation du programme et solution idéale

Soient K objectifs $f_1(x), f_2(x), \ldots, f_K(x)$
et $P = \{x \in R^n \mid Ax \leqslant b \ , \ x \geqslant 0\}$ l'ensemble des actions réalisables, où x est un vecteur de dimension n, b un vecteur de dimension m et A une matrice mxn.

Le programme multiobjectif s'écrit :

$$\begin{aligned} \text{Max } F(x) = (&f_1(x), f_2(x), \ldots, f_K(x)) \\ &Ax \leqslant b \\ &x \geqslant 0 \end{aligned}$$

La solution idéale de ce programme est la solution du système :

$$\begin{aligned} f_1(x) &= f_1^* \\ f_2(x) &= f_2^* \\ &\vdots \\ f_i(x) &= f_i^* \\ &\vdots \\ f_K(x) &= f_K^* \end{aligned}$$

où f_i^* , i=1,...,K est la valeur optimale du programme :

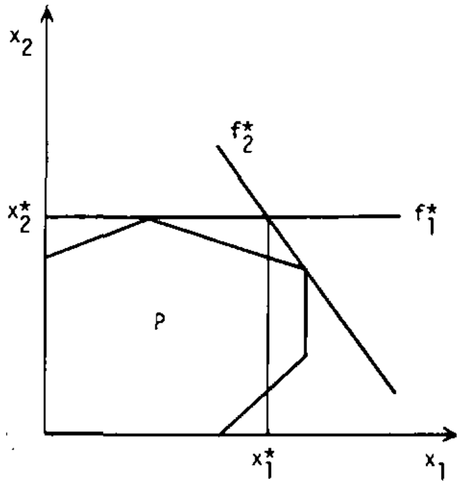
$$\max_{x \in P} \ f_i(x)$$

Généralement la solution idéale n'appartient pas à P, mais si tel est le cas, c'est bien entendu une solution du programme multiobjectif.

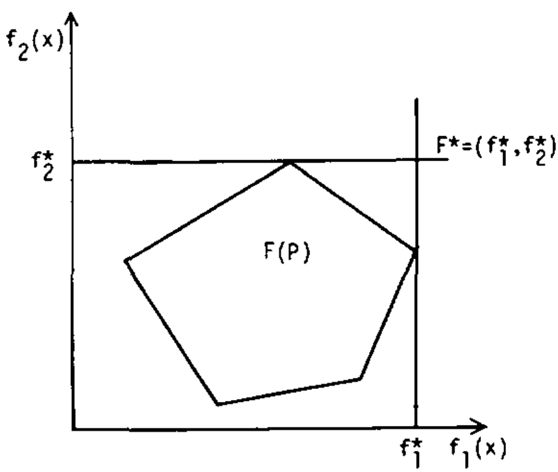
On appelle espace des critères l'image par la fonction F de l'ensemble des actions réalisables de P, et l'on notera F(P).

Représentation graphique de la solution idéale

a) Dans l'espace des actions (ou espace des activités)



b) Dans l'espace des critères (ou espace des objectifs)



3.2 Structure de dominance et solutions non dominées

Structure de dominance (ou notion de cône de dominance)

Soient y_1 et $y_2 \in F(P)$, deux points de l'espace des critères.

Lorsque y_1 est préféré à y_2 on notera $y_1 \succ y_2$.

Définition :

> Un vecteur non nul $d \in R^K$ est appelé facteur de dominance du vecteur $y \in F(P)$ si : $\forall \alpha > 0$, $y \succ y + \alpha d$.

On notera D_y l'ensemble des facteurs de dominance de y.

L'ensemble $D = \{D_y \mid y \in F(P)\}$ est appelé structure de dominance.

Un cas important est celui où pour tout y, D_y est un cône convexe d'origine 0.

L'ensemble D_y est alors appelé cône de dominance.

Dans la suite, on se limitera à ce cas particulier ou, autrement dit, *on supposera toujours que la relation de préférence* $\succ$ est telle que tous les ensembles D_y sont des cônes.

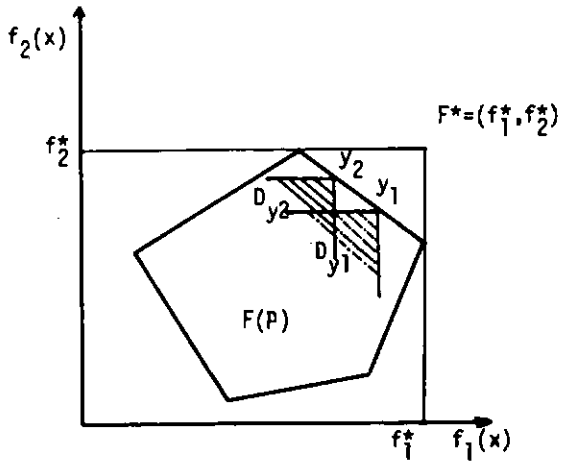
Solutions non dominées

Définition :

> Soient $y_1, y_2 \in F(P)$. On dit que y_1 est dominée par y_2 ssi $y_1 \in y_2 + D_{y_2}$.
>
> Un vecteur $y_0 \in F(P)$ est une valeur non dominée s'il n'existe aucun $y \in F(P)$ différent de y_0, et tel que $y_0 \in y + D_y$.

De même dans l'espace des actions, $x_o \varepsilon P$ est une <u>solution non dominée</u> s'il n'existe pas de $x \varepsilon P$, tel que $F(x_o) \neq F(x)$ et $F(x_o) = F(x) + D_{F(x)}$.

<u>Illustration</u> :



y_1 et y_2 sont 2 valeurs non dominées car il n'existe aucun vecteur $y \varepsilon F(P)$, $y \neq y_i$ et $y_i \varepsilon y + D_y$ (i=1,2).
Sur la figure D_y est l'ensemble $\{d \varepsilon R^2 \mid d < 0\}$

<u>Solution optimale au sens de Pareto ou solution efficace (cas particulier)</u>

Soit $D_o = \{d \varepsilon R^K \mid d \lneq 0\}$

Une solution non dominée pour cette structure de dominance correspond à une solution optimale au sens de Pareto ou solution efficace.

> $y_o \varepsilon F(P)$ est une <u>valeur efficace</u> ssi il n'existe pas de $y \varepsilon F(P)$, $y \neq y_o$ et tel que $y_o \varepsilon y + D_o$.

Dans l'espace des actions, x_o est une solution efficace ssi il n'existe pas de $x \in P$, tel que $F(x) \geqslant F(x_o)$.

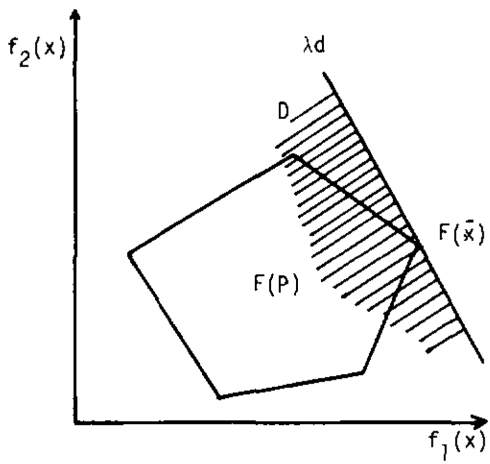
Exemples de structures de dominance :

Exemple 1 :

Lorsque le décideur peut pondérer les critères par un vecteur de poids $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_K)$, $\lambda_i > 0$, $i=1,\ldots,K$, le cône de dominance est le cône constant $D = \{d \in R^K \mid \lambda d \leqslant 0\}$.

Si l'hyperplan $\{d \in R^K \mid \lambda d = 0\}$ n'est pas parallèle à l'une des faces de F(P), il existe une seule solution non dominée $\bar{x}$ qui est la solution optimale du programme

$$\underset{x \in P}{\text{Max}} \quad \sum_{i=1}^{K} \lambda_i \, f_i \, (x)$$



Exemple 2 :

Si on suppose que la fonction d'utilité du décideur $U(y)$, définie sur l'espace des critères, est une fonction pseudo-concave, alors implicitement la structure de dominance du décideur est donnée par

$$D_y = \{d \in R^K \mid \nabla U(y) d \leqslant 0\}.$$

Dans les 2 exemples qu'on vient de voir D_y est un demi-espace. Mais au contraire de l'exemple 1 où D_y est un cône constant, c'est-à-dire indépendant de y, dans l'exemple 2, D_y varie avec chaque point y.

Propositions :

Soit N_D l'ensemble des solutions non dominées correspondant à une structure de dominance D.

1) si $D = \{\emptyset\}$ alors $N_D = F(P)$

2) si $D = R^K$ alors $N_D = \emptyset$

3) Soient D_1 et D_2 deux structures de dominance telles que $D_1 \subset D_2$ alors $N_{D_2} \subset N_{D_1}$

4) Soient λ_1 et λ_2 deux vecteurs : $\lambda_1, \lambda_2 \in R^K$, $\lambda_1 > 0$, $\lambda_2 > 0$.

Considérons les structures de dominance définies par les cônes $i=1,2$ $D_i = \{d \in R^K \mid \lambda_i d \leqslant 0\}$.

On sait que les cônes polaires associés à D_1 respectivement D_2 sont : $i=1,2$ $\Lambda_i = \{\lambda \in R^K \mid \lambda_i > 0\}$.

On a alors le résultat suivant :

Si $\Lambda_2 \subset \Lambda_1$ alors $N_{D_2} \subset N_{D_1}$

Les démonstrations de 1), 2) et 3) découlent immédiatement des définitions. Pour démontrer 4), il suffit de remarquer que si $\Lambda_2 \subset \Lambda_1$ alors $D_1 \subset D_2$ (voir propriétés des cônes polaires) et d'utiliser ensuite 3).

Les propositions énoncées ci-dessus permettent de voir que le "nombre" de valeurs ou solutions non dominées est d'autant plus "grand" que la structure de dominance est "petite".

Solutions efficaces et directions efficaces

Si les K fonctions objectifs $f_1(x), f_2(x), \ldots, f_K(x)$ sont linéaires, le programme multiobjectif peut s'écrire :

(MOLP) $\quad \operatorname*{Max}_{x \in P} Cx$

où $P = \{x \in R^n \mid Ax=b \quad , \quad x \geqslant 0\}$ est l'ensemble des actions réalisables et C est une matrice $K \times n$.

Définitions :

1) $\bar{x} \in P$ est une solution efficace du programme (MOLP), ssi il n'existe pas de $x \in P$, tel que $Cx \geqslant C\bar{x}$.

2) Une base B du programme (MOLP) est appelée base efficace, ssi la solution $\dot{x} \in P$ correspondant est une solution efficace.

3) Une direction $\bar{d}$ est une direction efficace en un point $\bar{x} \in P$ ssi $\bar{d}$ est une direction réalisable au point $\bar{x}$.

   Et s'il existe un scalaire $\hat{\alpha}>0$, tel que pour $\alpha \in [0,\hat{\alpha}]$, la solution $\bar{x}+\alpha\bar{d}$ est une solution efficace.

Propositions :

5) $\bar{x} \varepsilon P$ est une solution efficace du programme (MOLP) ssi il existe un vecteur $\lambda_o \varepsilon R^K$, $\lambda_o > 0$ tel que : $\bar{x}$ est solution optimale du programme multiparamétré

$$(MP) \quad \underset{x \varepsilon P}{\text{Max}} \ \lambda_o Cx$$

6) Soit B une base efficace associée à la solution x_o. Décomposons alors la matrice C en la matrice C_B de dimension Kxm et la matrice C_N de dimension Kx(n-m).

Notons $\Lambda = \{\lambda \varepsilon R^K \mid \lambda(C_N - C_B B^{-1} N) \leqslant 0 \ , \ \lambda > 0\}$, alors pour tout $\lambda \varepsilon \Lambda$, x_o est solution optimale du programme multiparamétré :

$$(MP_\lambda) \quad \underset{x \varepsilon P}{\text{Max}} \ \lambda Cx$$

7) Soient B une base efficace, l'ensemble défini en 6) et

$$\Lambda_o = \{\lambda \varepsilon R^K \mid \sum_{i=1}^{K} \lambda_i = 1, \lambda > 0\}$$

$$\Lambda \neq \emptyset \iff \Lambda_o \cap \Lambda \neq \emptyset$$

8) Soit $\bar{x}$ une solution non dégénérée de P et B la base associée. $\bar{x}$ est une solution efficace ssi il n'existe pas de direction réalisable $d = (d_B, d_N)$ en $\bar{x}$ telle que $(C_N - C_B B^{-1} N) d_N \geqslant 0$, ou ce qui revient au même ssi le programme:

$$\text{Max} \sum_{i=1}^{K} v_i$$

$$(C_N - C_B\ B^{-1}N)d_N - v = 0$$

$$d_N \geqslant 0 \quad , \quad v \geqslant 0$$

a une valeur optimale nulle.

9) Soient $\bar{x} \varepsilon P$ une solution efficace et $\bar{d}$ une direction réalisable en $\bar{x}$. $\bar{d}$ est une direction efficace en $\bar{x}$ ssi il n'existe aucune direction réalisable d en $\bar{x}$ telle que : $C\ d \geqslant C\ \bar{d}$.

10) Soient $\bar{x}$ une solution efficace non dégénérée de P et B la base associée.

$\bar{d} = (\bar{d}_B, \bar{d}_N)$ est une direction efficace ssi il n'existe pas de direction réalisable $d = (d_B, d_N)$ en $\bar{x}$ telle que $(C_N - C_B\ B^{-1}N)d_N \geqslant (C_N - C_B\ B^{-1}N)\bar{d}_N$.

11) Soient $\bar{x}$ une solution efficace non dégénérée de P, B la base associée et $d^j = \binom{-B^{-1}a^j}{e^j}$ la direction extrême en $\bar{x}$ suivant la variable $\bar{x}_j$.

d^j est une direction efficace ssi il n'existe pas de direction réalisable d en $\bar{x}$ telle que $(C_N - C_B B^{-1}N)d_N \geqslant (C_N^j - C_B B^{-1}a^j)$ ou ce qui revient au même ssi le programme :

$$\text{Max} \sum_{i=1}^{K} v_i$$

$$(C_N - C_B B^{-1}N)d_N - v = C_N^j - C_B B^{-1}a^j$$

$$d_N \geqslant 0 \quad , \quad v \geqslant 0$$

a une valeur optimale nulle.

Preuves :

5) Nous donnerons ici la preuve qu'une solution optimale du programme multiparamétré est une solution efficace. Pour une démonstration complète, il faudrait introduire la dualité d'Iserman (voir [32]).

Supposons que $\bar{x}$ est une solution optimale du programme (MP) et qu'il existe un $x \in P$ tel que $Cx \geqslant C\bar{x}$.

Comme λ_0 est strictement positif en toutes ses composantes, on aura $\lambda_0 Cx > \lambda_0 C\bar{x}$, ce qui contredit l'optimalité de $\bar{x}$.

6) La preuve découle de la condition d'optimalité de l'algorithme du simplexe.

7) L'implication "de droite vers la gauche" est évidente. Prouvons l'autre implication.

Soit $\lambda \in \Lambda$ et posons $\psi = 1 / \sum_{i=1}^{K} \lambda_i$.

Par construction même $\psi \lambda \in \Lambda_0$.
D'autre part $\psi \lambda \in \Lambda$, car $\psi > 0$ et Λ est un cône.

8) Pour un α positif assez petit, $x = \bar{x} + \alpha d$ est une solution réalisable. En tenant compte de la définition de d_B :
$d_B = -B^{-1} N\, d_N$, on peut expliciter la valeur de la fonction objectif en x :

$$Cx = C(\bar{x}+\alpha d) = C\bar{x}+\alpha(C_N - C_B B^{-1} N) d_N .$$

$\bar{x}$ ne peut donc être efficace que si le vecteur $(C_N - C_B B^{-1} N) d_N$ a au moins une composante strictement négative.

9) Supposons que $\bar{d}$ soit une direction efficace. Pour des α assez petits, $\bar{x}+\alpha d$ et $\bar{x}+\alpha\bar{d}$ sont des solutions réalisables et comme $\bar{d}$ est une direction efficace, on ne peut pas avoir $C(\bar{x}+\alpha d) \geqslant C(\bar{x}+\alpha\bar{d})$ ce qui prouve qu'il est impossible d'avoir $Cd \geqslant C\bar{d}$.

Montrons maintenant la réciproque. Pour cela, il faut montrer que pour tout α assez petit $x+\alpha d$ est solution efficace. Soit donc y une solution réalisable quelconque et posons $d = \frac{y-\bar{x}}{\alpha}$.

d est une direction réalisable en $\bar{x}$ et par hypothèse on ne peut avoir $Cd \geqslant C\bar{d}$, donc pas non plus $C(\bar{x}+\alpha d) \geqslant C(\bar{x}+\alpha\bar{d})$. Or le premier terme de cette dernière inégalité vaut Cy et nous avons donc prouvé que $\bar{d}$ est une direction efficace.

10) Découle immédiatement de 9).

11) La démonstration est un cas particulier de 10) avec $d_N = e^j$.

3.3 Solutions de compromis (ou notion du point de mire)

Nous avons déjà signalé que la solution idéale x^* n'est généralement pas une solution réalisable. Une stratégie pour trouver une "bonne" solution d'un programme multiobjectif est de trouver la solution pour laquelle les valeurs des K fonctions objectifs sont les plus proches (en terme de distance) des K valeurs idéales $f_1^*, f_2^*, \ldots, f_K^*$. Une telle solution est appelée solution de compromis.

Une des possibilités de mesurer la proximité entre le vecteur des valeurs $F(x)=(f_1(x),\ldots,f_K(x))$ et le vecteur des valeurs idéales $F^*=(f_1^*,\ldots,f_K^*)$ est d'utiliser une distance de Minkowski. Rappelons que ces distances constituent une famille définie par :

$$L_p = \left(\sum_{i=1}^{K} |f_i^* - f_i(x)|^p\right)^{\frac{1}{p}} \quad , \quad 1 \leqslant p \leqslant \infty$$

soit x_p la solution du programme

$$\min_{x \in P} L_p$$

x_p est alors la solution de compromis relative à la distance L_p, $1 \leqslant p \leqslant \infty$.

Malgré le fait que les différentes distances L_p , $1<p<\infty$, sont "topologiquement" équivalentes, elles ne génèrent pas la même solution de compromis. Le problème du choix de la distance L_p reste donc posé.

Les distances L_1 et L_∞ sont généralement retenues pour des raisons de commodité, car les programmes à résoudre sont alors des programmes linéaires.

Posons pour $1 \leqslant p < \infty$ $\bar{L}_p = L_p^p$. Comme la fonction $y \to y^p$ est croissante pour tout $p \geqslant 1$, les programmes

$$\min_{x \in P} L_p \qquad\qquad \min_{x \in P} \bar{L}_p$$

sont équivalents pour $1 \leq p < \infty$, c'est-à-dire ont même solution optimale.

Montrons maintenant comment on peut convertir pour $p=1$, respectivement $p=\infty$, le programme définissant la solution de compromis en un programme linéaire.

Commençons par remarquer que pour tout $x \varepsilon P$ $\quad f_i^* \geq f_i(x)$,

car $f_i^* = \max\limits_{x \varepsilon P} f_i(x)$.

Pour $p=1$ on a $\min\limits_{x \varepsilon P} L_1 = \sum\limits_{i=1}^{K} | f_i^* - f_i(x) |$

et le programme linéaire suivant est équivalent :

$$\min_{x \varepsilon P} \sum_{i=1}^{K} y_i^-$$

$$f_i(x) + y_i^- = f_i^* \qquad i=1,\ldots,K$$

$$y_i^- \geq 0 \qquad i=1,\ldots,K$$

Pour $p=\infty$, on a :

$$\min_{x \varepsilon P} y$$

$$f_i(x) + y \geq f_i^* \qquad , \quad i=1,\ldots,K$$

$$y \geq 0$$

Propositions :

1) Les solutions de compromis générées par L_p , $1<p<\infty$, sont uniques.

2) Les solutions de compromis générées par L_p , $1 \leq p < \infty$, sont des solutions efficaces du programme multiobjectif :

$$\max_{x \varepsilon P} F(x) = (f_1(x), f_2(x), \ldots, f_K(x))$$

3) Au moins une des solutions générées par L_∞ est efficace.

4) Une solution efficace du programme bicritère

$$(BLP) \quad \min_{x \in P} \; (L_1, L_\infty)$$

est une solution efficace du programme multiobjectif

$$(MOLP) \quad \max_{x \in P} \; (f_1(x), f_2(x), \ldots, f_K(x)).$$

<u>Preuves</u> :

1) Il suffit de constater que les fonctions $\bar{L}_p = \sum_{i=1}^{K} |f_i^* - f_i(x)|^p$, $1<p<\infty$, sont strictement convexes.

2) Soit x_p une solution optimale du programme

$$\min_{x \in P} \sum_{i=1}^{K} |f_i^* - f_i(x)|^p, \; 1 \leq p < \infty.$$

Il n'existe pas de $x \in P$ tel que

$$\sum_{i=1}^{K} |f_i^* - f_i(x)|^p < \sum_{i=1}^{K} |f_i^* - f_i(x_p)|^p$$

donc il n'existe pas de $x \in P$ tel que

$$|f_i^* - f_i(x)| \leq |f_i^* - f_i(x_p)|, \; i=1,\ldots,K,$$

avec au moins une inégalité stricte.

Or pour tout $x \in P$, on a $f_i^* \geq f_i(x)$, $i=1,\ldots,K$

Donc il ne peut pas exister de $x \in P$ tel que

$$f_i^* - f_i(x) \leq f_i^* - f_i(x_p) \quad \text{ou} \quad f_i(x) \geq f_i(x_p) \quad , \; i=1,\ldots,K$$

avec au moins une inégalité stricte,

ce qui prouve que x_p est une solution efficace.

3) Soit X_∞ l'ensemble des solutions optimales du programme $\underset{x\varepsilon P}{\text{Min}}\ \underset{i}{\text{Max}}\ \{|f_i^*-f_i(x)|\}$.

L'ensemble X_∞ est contenu dans P, qui est un polyèdre fermé, et est lui-même fermé.

Donc il existe $x_1 \varepsilon X_\infty$ solution du programme

(1) $$\underset{x\varepsilon X_\infty}{\text{Min}} \sum_{i=1}^{K} |f_i^*-f_i(x)|$$

Montrons que x_1 est une solution efficace.

Supposons que x_1 n'est pas une solution efficace. Alors il existe $x\varepsilon P$ tel que

$f_i(x) \geqslant f_i(x_1)$

avec au moins une inégalité stricte.

D'où $f_i^*-f_i(x) \leqslant f_i^*-f_i(x_1)$

avec au moins une inégalité stricte,

soit $L_\infty(x) \leqslant L_\infty(x_1)$ et donc $x \ \varepsilon\ X_\infty$.

De plus on aura, à cause de l'existence d'une inégalité stricte,

$$\sum_{i=1}^{K} |f_i^*-f_i(x)| < \sum_{i=1}^{K} |f_i^*-f_i(x_1)|$$

ce qui contredit le fait que x_1 est solution optimale du programme (1).

4) Soit x* une solution efficace du programme (BLP).

Supposons que x* n'est pas une solution efficace du programme multiobjectif (MOLP).

Alors il existe $x\varepsilon P$ tel que

$f_i(x) \geq f_i(x^*)$, avec au moins une inégalité stricte

d'où $f_i^*-f_i(x) \leq f_i^*-f_i(x^*)$, avec au moins une inégalité stricte.

Donc on a $L_\infty(x) \leq L_\infty(x^*)$

et $\sum_{i=1}^{K} |f_i^*-f_i(x)| < \sum_{i=1}^{K} |f_i^*-f_i(x^*)|$

ce qui contredit le fait que x^* est une solution efficace du programme (BLP).

4. QUELQUES ALGORITHMES D'OPTIMISATION MULTICRITERE

Aprés avoir donné dans les chapitres précédents quelques bases théoriques, nous décrivons ici, de façon détaillée, certains algorithmes d'optimisation multicritère. Ces algorithmes sont de deux types, soit le type discret où l'ensemble P des solutions est fini et le type continu où l'ensemble P des solutions est défini par un ensemble de contraintes linéaires et les critères par des fonctions continues. Pour le type discret, nous avons choisi deux méthodes : ELECTRE I et ELECTRE II basées toutes les deux sur la notion de surclassement développée par B. Roy et, pour le type continu, 5 algorithmes auxquels nous avons apporté des modifications importantes.

4.1 Relations de surclassement : Dans ce paragraphe nous définissons la relation de surclassement et nous décrivons les méthodes basées sur cette relation, soit ELECTRE I et ELECTRE II.

4.2 Méthode de Geoffrion : Nous relatons cette méthode telle qu'elle a été décrite par plusieurs auteurs (voir [5a], [9]). Nous avons juste apporté une modification au niveau de la solution initiale qui n'était pas nécessairement une solution efficace.

4.3 Méthode de Zionts-Wallenius : Nous nous différencions des auteurs de cette méthode dans la recherche des directions efficaces et des vecteurs de poids nécessaires à la génération des solutions efficaces. Ces différences sont très importantes au niveau de l'implantation de la méthode sur ordinateur.

4.4 Algorithme interactif du simplexe : Zeleny et Philip ont déjà utilisé l'algorithme du simplexe pour générer de façon automatique toutes les solutions efficaces et extrêmes d'un programme multiobjectif (voir [5a], [29] et [30]).

Notre méthode est une variante interactive des méthodes utilisées par ces deux auteurs. Elle permet au décideur de chercher une solution efficace qui l'intéresse plus que toutes les autres et aussi de générer toutes les solutions efficaces et extrêmes d'un programme multiobjectif.

Dans la méthode de Zeleny, on vérifie l'efficacité de la solution après l'avoir trouvée tandis que dans notre méthode, nous n'effectuons un déplacement que si la direction a été trouvée efficace.

4.5 Méthode de STEUER : Cette méthode est décrite de façon beaucoup plus complète que dans tous les ouvrages que nous citons dans la partie bibliographique.

4.6 La méthode STEM : Benayoun, de Montgolfier et Tergny utilisent dans cette méthode la distance L_∞ pour générer une solution de compromis. Une telle solution n'est pas nécessairement efficace et nous avons utilisé la combinaison convexe $\varepsilon L_1+(1-\varepsilon)L_\infty$, où $\varepsilon>0$ est assez petit pour assurer l'efficacité de la solution générée.

4.1 Les méthodes de surclassement

Préliminaires :

Dans les deux méthodes que nous allons décrire par la suite, l'ensemble P des solutions réalisables est supposé fini et peut donc être énuméré au moyen d'une liste d'actions $(a_1, a_2, \ldots, a_n)$.

Les actions seront évaluées par K critères, chacun étant une application, notée f_i, de l'ensemble des actions P à valeur dans un ensemble ordonné fini E_i, appelé l'échelle du critère i . On dira qu'une action a_ℓ est préférée à l'action a_k pour le critère i , si $f_i(a_\ell) > f_i(a_k)$.

La relation de surclassement

Une relation de surclassement S_p est une relation binaire définie sur l'ensemble P des actions et dont la signification peut être décrite par les trois cas exclusifs suivants :

a) Si $a_i \; S_p \; a_j$ et $a_j \; S_p \; a_i$, on dira qu'il y a indifférence entre a_i et a_j

b) Si $a_i \; S_p \; a_j$ et non $(a_j \; S_p \; a_i)$, on dira que a_i est strictement préféré à a_j

c) Si non $(a_i \; S_p \; a_j)$ et non $(a_j \; S_p \; a_i)$, on dira que les deux actions sont incomparables

La relation S_p n'est pas nécessairement transitive à cause de la situation d'incomparabilité. Sa construction dépend de la problématique du choix retenue, dont deux seront décrites par la suite.

La méthode ELECTRE I (élimination et choix traduisant la réalité)

Cette méthode essaie d'apporter une solution à la problématique du choix suivante : sélectionner un sous-ensemble d'actions considérées comme "bonnes". La relation de surclassement S_p sera donc construite de telle manière qu'elle partagera l'ensemble P des actions réalisables en un sous-ensemble N constitué des "bonnes" actions et son complément P-N constitué des "mauvaises" actions.

Construction de la relation de surclassement S_p

On attribue à chaque critère un poids π_i (>0) d'autant plus grand que le critère est important. Si le décideur ne désire pas différencier les critères, il suffit de poser $\pi_i=1$, pour tout i , i=1,...,K.

Pour chaque couple d'actions (a_ℓ,a_k), on construit les ensembles suivants :

$$I^+ = \{i | f_i(a_\ell) > f_i(a_k)\}$$

$$I^= = \{i | f_i(a_\ell) = f_i(a_k)\}$$

et

$$I^- = \{i | f_i(a_\ell) < f_i(a_k)\}$$

On calcule ensuite l'indicateur de concordance :

$$c(a_\ell,a_k) = \frac{\sum_{i\in I^+} \pi_i + \sum_{i\in I^=} \pi_i}{\sum_{i=1}^{K} \pi_i}$$

Cet indicateur présente les trois propriétés suivantes :

. il varie de 0 à 1 de façon croissante avec l'enrichissement de I^+ et $I^=$

. il vaut 1 si et seulement si pour tout

$i, i=1,\dots,K \qquad f_i(a_\ell) \geq f_i(a_k)$

. il conserve sa signification et ne conduit à aucune incohérence lorsqu'on subdivise un critère en plusieurs autres qu'on met à sa place (il faut naturellement que la somme des nouveaux poids soit égale au poids du critère remplacé).

On introduit de plus un <u>indicateur de discordance</u> défini par :

$$d(a_\ell,a_k) = \begin{cases} 0 \ , & \text{si } I^- = \emptyset \\ \frac{1}{\delta} \max\limits_{i\in I^-} \{f_i(a_k)-f_i(a_\ell)\} \end{cases}$$

où δ est l'amplitude de l'échelle E_i associée au critère f_i pour lequel existe le maximum de désaccord.

Ce nouvel indicateur varie lui aussi entre 0 et 1.

Considérons maintenant deux nombres compris entre 0 et 1, l'un p plutôt proche de 1, l'autre q relativement proche de 0. On dit que a_ℓ surclasse a_k ($a_\ell \ S_p \ a_k$) si et seulement si :

$$c(a_\ell,a_k) \geq p \ ,$$

et $$d(a_\ell,a_k) \leq q \ .$$

Cette relation de surclassement peut être représentée par un graphe G :

$G = (P,U)$, où P est l'ensemble des actions et où U est défini par :

$$(a_i,a_j) \in U \Leftrightarrow a_i \ S_p \ a_j \quad .$$

Le sous-ensemble cherché N, constitué des "bonnes" actions de P, est le <u>noyau du graphe</u> G qui est défini par les propriétés suivantes :

a) <u>Propriété de stabilité externe</u>

Tout sommet hors du noyau est surclassé par au moins un sommet du noyau. Ou, autrement dit :

Pour tout $a_j \in$ P-N, il existe $a_i \in$ N, tel que $a_i \, S_p \, a_j$ (ou, ce qui revient au même $(a_i, a_j) \in U$).

b) <u>Propriété de stabilité interne</u>

Aucun sommet du noyau n'est surclassé par un autre sommet du noyau, soit mathématiquement :

Pour tout $a_i \in$ N et tout $a_j \in$ N, on a <u>non</u> $(a_i \, S_p \, a_j)$ <u>et</u> <u>non</u> $(a_j \, S_p \, a_i)$ (ou, ce qui revient au même $(a_i, a_j) \notin U$ et $(a_j, a_i) \notin U$).

<u>Proposition</u> :

Le noyau N existe et est unique, si le graphe G qui représente la relation de surclassement S_p est sans circuit.

<u>Remarque</u> :

Notons par $\sim$ la relation d'indifférence et par $>$ la relation de préférence stricte. On peut distinguer deux cas type pour un circuit :

a) $A \sim B$ <u>et</u> $B \sim C$ <u>et</u> $C \sim A$: Dans ce cas la transitivité n'est pas violée et le remplacement des trois actions A, B et C par un seul groupe d'actions peut être justifié.

b) $A > B$ <u>et</u> $B > C$ <u>et</u> $C \sim A$: Dans un certain sens la transitivité est violée et la substitution d'un groupe d'actions aux trois actions individuelles est un artifice.

Mesures à prendre lorsque le graphe G comporte un ou plusieurs circuits :

a) Remplacer le circuit par un seul sommet v qui représentera le groupe des actions qui font partie du circuit.

b) Répéter a) jusqu'à ce qu'il n'y ait plus de circuit dans le graphe.

c) Pour ce graphe sans circuit, calculer le noyau.

Le noyau ainsi obtenu est appelé un quasi-noyau du graphe initial G. Il jouit des propriétés suivantes :

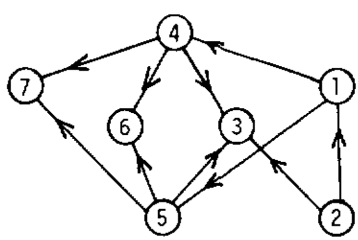
- Aucun sommet du quasi-noyau n'est surclassé par un autre sommet du quasi-noyau.

- Tout sommet hors du quasi-noyau est surclassé (directement) par un sommet du quasi-noyau ou surclassé (indirectement) par un sommet qui lui est surclassé par un sommet du quasi-noyau.

Remarque :

Le quasi-noyau n'est pas unique.

Exemple de calcul du quasi-noyau :



Graphe G de surclassement

Le noyau de ce graphe se compose des sommets 2, 4 et 5.

Si maintenant on introduit un arc orienté du sommet 4 vers le sommet 2, le graphe G comporte un circuit formé des sommets 1, 2 et 4 et nous avons les trois quasi-noyaux $N_1 = \{1\}$, $N_2 = \{2\}$ et $N_3 = \{4\}$.

La méthode ELECTRE II

La méthode ELECTRE II essaie de ranger les actions de P en une suite ordonnée de classes d'indifférence allant des "meilleures" aux "pires". Cette méthode est elle aussi basée, comme la méthode ELECTRE I, sur la notion de surclassement. Cependant cette notion sera affinée ici, et on parlera de relation de surclassement fort S_F et de relation de surclassement faible s_f. On dira qu'une action a_i de P surclasse fortement une autre action a_j de P s'il y a très peu de risques d'erreur lorsqu'on affirme que a_i est préférée à a_j, compte tenu des K critères, et que l'action a_i surclasse faiblement a_j s'il y a un grand risque d'erreur lorsqu'on affirme que a_i est préférée à a_j.

Construction des relations S_F et s_f

Il est supposé, comme dans ELECTRE I, qu'à chaque critère f_i est associé un poids π_i mesurant son importance. Les définitions pour les ensembles I^+, $I^=$ et I^- , ainsi que pour l'indice de concordance c sont celles données pour la méthode ELECTRE I.

Soient trois scalaires c_1,c_2,c_3 tels que $1 > c_1 > c_2 > c_3 > 0$ (les valeurs standards sont $c_1=3/4$, $c_2=2/3$ et $c_3=3/5$) et pour chaque critère f_i , $i=1,\dots,K$ deux valeurs de désaccord d_{1i} et d_{2i} telles que $0 < d_{1i} < d_{2i}$.

On dit que l'action a_ℓ surclasse fortement l'action a_k si

a) $\sum_{i\in I^+} \pi_i \geq \sum_{i\in I^-} \pi_i$

$\underline{et}$

b1) $c(a_\ell, a_k) \geq c_1$ $\underline{et}$ $f_i(a_k) - f_i(a_\ell) \leq d_{2i}$ pour tout $i \in I^-$

$\underline{ou}$

b2) $c(a_\ell, a_k) \geq c_2$ $\underline{et}$ $f_i(a_k) - f_i(a_\ell) \leq d_{1i}$ pour tout $i \in I^-$

De la même manière, on dira que a_ℓ surclasse faiblement a_k

si $\sum_{i \in I^+} \pi_i \geq \sum_{i \in I^-} \pi_i$,

si $c(a_\ell, a_k) \geq c_3$ $\underline{et}$

si pour tout $i \in I^-$: $f_i(a_k) - f_i(a_\ell) \leq d_{2i}$.

Les deux relations de surclassement S_F et s_f étant construites, on passe au classement des actions.

a) <u>Classement direct</u> : Soit C l'ensemble des actions déjà classées. Au départ C=∅. Soient S l'ensemble des sommets qui ne sont surclassés fortement par aucun autre sommet dans P-C et B l'ensemble des sommets qui ne sont surclassés faiblement par aucun autre sommet dans S. Les actions de B forment la classe de rang juste supérieur au rang de la classe précédemment constituée. On pose C = C ∪ B et on refait une itération. On arrête lorsque C = P .

b) <u>Classement indirect</u> : L'algorithme est le même que pour le classement direct sauf que la première classe constituée occupe le dernier rang et les ensembles S et B sont formés respectivement des sommets qui ne surclassent fortement aucun autre dans P-C et de ceux qui ne surclassent faiblement aucun autre dans S. Les actions de B forment la classe de rang juste inférieur au rang de la classe précédemment constituée.

Quand les deux classements sont assez similaires, le classement final est le classement médian. Dans le <u>classement médian</u>, une action a_i a comme rang la moyenne des rangs du classement direct et indirect.

4.2 Méthode de Geoffrion

Notations :

Soit le programme :

$$\max_{x \in P} U(F(x)) = U(f_1(x), f_2(x), \ldots, f_K(x))$$

où $U(F(x))$ est la fonction représentant les préférences du décideur par rapport aux K critères $f_1(x), \ldots, f_K(x)$ (U n'est pas connue explicitement). Les fonctions $f_i(x)$ et l'ensemble P des solutions réalisables sont explicitement donnés.

Préliminaires :

La méthode de Geoffrion est basée sur l'algorithme de Frank-Wolfe (méthode du gradient, voir [2]). Les conditions suivantes doivent être remplies pour l'utilisation de cet algorithme :

- L'ensemble P des solutions réalisables est convexe et compact.
- La fonction $U(f_1(x), \ldots, f_K(x))$ est différentiable et concave.
- Chaque fonction $f_i(x)$ est différentiable et concave sur P.

Si on choisit un point initial $\bar{x}$ de P et si on utilise une approximation linéaire $\bar{U}(x) = U(F(\bar{x})) + \nabla_x U(F(\bar{x}))(x-\bar{x})$ de la fonction $U(F(x))$ au point $\bar{x}$, l'algorithme de Frank-Wolfe se résume aux 2 étapes suivantes :

1. Recherche d'une direction réalisable qui améliore la valeur de la fonction $U(F(x))$

   La direction $(x^*-\bar{x})$, $x^* \varepsilon P$ est une direction réalisable qui améliore la fonction $U(F(x))$

si $\nabla_x U(F(\bar{x}))(x^*-\bar{x}) > 0$

ou si x^* est la solution optimale du programme linéaire

$$(\text{Lpl}) \qquad \underset{x \in P}{\text{Max}}\ \nabla_x U(F(\bar{x}))x$$

Dans la suite on notera $d^* = x^* - \bar{x}$

2. Recherche d'un nouveau point qui augmente la valeur de la fonction $U(F(x))$

Soit $\bar{\alpha}$ la solution optimale du programme

$$(\text{NLp}) \qquad \text{Max}\ U(F(\bar{x}+\alpha d^*)) \qquad 0 < \alpha < 1$$

Le nouveau point cherché est $x_1 = \bar{x} + \bar{\alpha}\, d^*$

si $x_1 = \bar{x}$, $\bar{x}$ est la solution optimale et on arrête, sinon on fait une nouvelle itération.

Méthode pour déterminer une direction qui améliore la valeur de la fonction d'utilité $U(F(x))$

La fonction $U(F(x))$ n'étant pas donnée explicitement, le programme (Lpl) ne peut être résolu sans l'assistance du décideur. L'acquisition de l'information nécessaire au traitement constituera la partie interactive de la méthode.

Quelle information demandée ?

On sait que $\nabla_x U(F(\bar{x})) = \nabla_F U(F(\bar{x})) \cdot F'(\bar{x})$
où $F'(\bar{x})$ est la matrice jacobienne de F au point $\bar{x}$.

Les lignes de cette matrice sont les K vecteurs gradients $\nabla f_i(\bar{x})$, $i=1,\dots,K$, et $\nabla_F U(F(\bar{x})) = (\frac{\partial U}{\partial f_1}(F(\bar{x})), \frac{\partial U}{\partial f_2}(F(\bar{x})), \ldots, \frac{\partial U}{\partial f_K}(F(\bar{x})))$.

Pour alléger les notations, nous écrirons $\frac{\partial U}{\partial f_i}$
en lieu et place de $\frac{\partial U}{\partial f_i}(F(\bar{x}))$;
en particulier nous écrirons $\nabla_x U(F(\bar{x})) = \sum_{i=1}^{K} (\frac{\partial U}{\partial f_i}) \cdot \nabla f_i(\bar{x})$.

Notons que $\frac{\partial U}{\partial f_i}$ est l'utilité marginale du critère f_i au point $F(\bar{x})$.

Le programme (Lp1) peut s'écrire maintenant :

$$\text{(Lp2)} \qquad \underset{x \in P}{\mathrm{Max}} \left[\sum_{i=1}^{K} \left(\frac{\partial U}{\partial f_i}\right) \cdot \nabla f_i(\bar{x}) \right] \cdot x$$

Supposons que l'utilité marginale $\frac{\partial U}{\partial f_1}$ du critère f_1 est positive en tout point de P. Alors la solution optimale du programme (Lp2) ne change pas si on divise la fonction objectif par le scalaire positif $\frac{\partial U}{\partial f_1}$.

D'où le nouveau programme :

$$\underset{x \in P}{\mathrm{Max}} \quad (\Sigma\, w_i \cdot \nabla f_i(\bar{x}))\, x$$

où $\qquad w_i = \frac{\partial U}{\partial f_i} \Big| \frac{\partial U}{\partial f_1} \quad , \quad i=1,\ldots,K$

Par définition, le taux de substitution au point $\bar{x}$ d'un critère f_i par rapport à un critère de référence f_r est :

$$\tau_r^i = \frac{\partial U}{\partial f_r} \Big| \frac{\partial U}{\partial f_i} = - \frac{df_i(\bar{x})}{df_r(\bar{x})}$$

et peut être estimé par $- \frac{\nabla f_i}{\nabla f_r}$

or $\qquad w_i = \frac{1}{\tau_1^i} \qquad i=1,\ldots,K$

Donc en demandant au décideur de nous fournir les taux de substitution de tous les critères $f_i(x)$, $i=2,\dots,K$ par rapport au critère de référence $f_1(x)$, on peut estimer les w_i , $i=2,\dots,K$ et ainsi déterminer une direction réalisable d* qui améliore la valeur de la fonction d'utilité.

Recherche d'un nouveau point de P

La résolution directe du programme (NLp) étant impossible, vu que la fonction U(F(x)) n'est pas donnée explicitement, on tabulera les fonctions $f_i(\bar{x}+\alpha d^*)$, $i=1,\dots,K$ avec $\alpha \in [0,1]$ et le décideur choisira la valeur de α qui lui donnera la plus grande satisfaction globale. Soit $\bar{\alpha}$ cette valeur, le nouveau point est $x_1 = \bar{x} + \bar{\alpha}\, d^*$.

Test d'arrêt pratique

Vu que la fonction U(F(x)) est concave, on peut écrire :

$$U(F(x)) \leq U(F(y)) + \nabla_F U(F(y)) \cdot (F(x)-F(y))$$

Si x_{k-1} et x_k désignent des solutions réalisables rencontrées lors de deux itérations successives, on a donc :

(1) $$U(F(x_k)) - U(F(x_{k-1})) \leq \nabla_F U(F(x_{k-1})) \cdot (F(x_k)-F(x_{k-1})) .$$

Or

(2) $$\nabla_F U(F(x_{k-1})) = \left(\frac{\partial U}{\partial f_1} , \frac{\partial U}{\partial f_2} , \dots , \frac{\partial U}{\partial f_K} \right)$$

est un vecteur non connu.

Seul le vecteur des taux de substitution par rapport à un critère de référence est supposé connu. Si f_1 est le critère de référence, le vecteur des taux de substitution est :

(3) $$\left(\frac{\partial U}{\partial f_1} \Big| \frac{\partial U}{\partial f_1} , \frac{\partial U}{\partial f_2} \Big| \frac{\partial U}{\partial f_1} , \ldots , \frac{\partial U}{\partial f_K} \Big| \frac{\partial U}{\partial f_1} \right)$$

en posant $\frac{\partial U}{\partial f_1} = c$ et $\frac{\partial U}{\partial f_i} \Big| \frac{\partial U}{\partial f_1} = w_i^{k-1}$, $i=2,\ldots,K$,

et en tenant compte des relations (2) et (3), la relation (1) s'écrit :

$$U(F(x_k)) - U(F(x_{k-1})) \leqslant c(1,w_2^{k-1},\ldots,w_K^{k-1})(F(x_k) - F(x_{k-1}))$$

c n'étant pas connu, seule une variation relative peut être majorée. Hwang et Masud dans [9] proposent qu'on arrête à la kième itération si le rapport

$$\frac{(1,w_2^{k-1},\ldots,w_K^{k-1})(F(x_k)-F(x_{k-1}))}{(1,w_2^0,\ldots,w_K^0)(F(x_1)-F(x_0))} \quad , \text{ avec } k > 1$$

est inférieur à un scalaire γ positif assez petit.

Autres propositions :

On pourrait arrêter à la kième itération si le rapport

$$\frac{(1,w_2^{k-1},\ldots,w_K^{k-1}) \cdot (F(x_k)-F(x_{k-1}))}{(1,w_2^{k-1},\ldots,w_K^{k-1}) \cdot (F(x_{k-1})-F(x_{k-2}))}$$

est inférieur à un scalaire positif assez petit ou si le rapport

$$\frac{(1,w_2^0,\ldots,w_K^0)(F(x_k)-F(x_0))}{(1,w_2^0,\ldots,w_K^0)(F(x_{k-1})-F(x_0))} \quad \text{est proche de } 1 .$$

4.3 Méthode de Zionts-Wallenius

Préliminaires :

Cette méthode suppose que les K critères ou objectifs $f_1(x)$, $f_2(x)$,..., $f_K(x)$ sont des fonctions linéaires et que la fonction $U(f_1(x),\dots,f_K(x))$ représentant les préférences du décideur est une fonction linéaire implicitement connue par ce dernier.

Autrement dit, on suppose qu'il existe un vecteur de poids $\lambda^* \in R^K, \lambda_i^* > 0, \sum_{i=1}^{K} \lambda_i^* = 1$ connu par le décideur et tel que

$U(f_1(x),\dots,f_K(x)) = \sum_{i=1}^{k} \lambda_i^* f_i(x) = \lambda^* Cx$ où C est la matrice des coefficients de dimension Kxn.

Par ailleurs l'ensemble des solutions réalisables $P = \{x \in R^n | Ax \leq b, x \geq 0\}$ est un polyèdre convexe.

Caractérisation des directions efficaces extrêmes

Soit $\bar{x}$ une solution efficace et extrême de P et B la base associée. Notons par $\bar{J}$ l'ensemble des indices des variables hors base.

La direction extrême $d^j = \begin{pmatrix} -B^{-1}a^j \\ e^j \end{pmatrix}$ suivant la variable hors base $\bar{x}_j$ est efficace ssi la valeur de la fonction objectif du programme linéaire :

$$\text{Max} \sum_{i=1}^{K} v_i$$

$$(C_N - C_B B^{-1} N)\, d_N - v = C_N^j - C_B B^{-1} a^j$$

$$d_N \in R^{n-m} \;,\; d_N \geq 0 \;,\; v \geq 0$$

est nulle (voir proposition 11 du paragraphe 3.2).

Dans ce cas, on dira que la variable hors base $\bar{x}_j$ est efficace.

Notons par $\bar{J}_{eff}$ l'ensemble des indices des variables hors bases efficaces. On sait que $\bar{J}_{eff} \subset \bar{J}$.

Si $\bar{J}_{eff} = \emptyset$, la solution actuelle $\bar{x}$ est la solution cherchée, sinon on établit le dialogue avec le décideur pour la recherche du vecteur de poids λ^*.

<u>Dialogue avec le décideur</u>

Pour tout vecteur $(C_N^j - C_B B^{-1} a^j) = (w_1^j, w_2^j, \ldots, w_K^j)$, $j \in \bar{J}_{eff}$, on demande au décideur si une variation de w_1^j sur le critère $f_1(x)$, de w_2^j sur le critère $f_2(x)$, de w_K^j sur le critère $f_K(x)$ est globalement une amélioration ou une dégradation.

a) S'il s'agit d'une amélioration, on est sûr que la forme linéaire cherchée $\lambda^* Cx$ augmente de valeur si le déplacement se fait suivant la direction correspondant à la variable hors base $\bar{x}_j$, ou autrement dit, que le vecteur de poids λ^* est tel que

$$\sum_{i=1}^{K} \lambda_i^* w_i^j \geq \varepsilon \text{ , où } \varepsilon \text{ est un scalaire positif assez petit.}$$

b) S'il s'agit d'une dégradation, le vecteur λ^* est tel que

$$\sum_{i=1}^{K} \lambda_i^* w_i^j \leq -\varepsilon \text{ , pour un } \varepsilon \text{ positif assez petit.}$$

c) S'il n'arrive pas à se prononcer pour une amélioration ou une dégradation, on supposera qu'il est indifférent et le vecteur λ^* est tel que $\sum_{i=1}^{K} \lambda_i^* w_i^j = 0$.

Recherche d'un vecteur de poids λ appartenant au même ensemble que λ^*

Notons par $\bar{J}^{+}_{eff}$, $\bar{J}^{-}_{eff}$ et $\bar{J}^{=}_{eff}$ les trois sous-ensembles de $\bar{J}_{eff}$ correspondant (selon les dire du décideur) à une amélioration, une dégradation, respectivement à une situation d'indifférence.

λ^*, le vecteur de poids cherché, est une solution réalisable du système suivant, pourvu que le paramètre ε soit choisi assez petit :

$$
(S_1) \qquad
\begin{aligned}
&\sum_{i=1}^{K} \lambda_i w_i^j \geq \varepsilon \quad \forall j \in \bar{J}^{+}_{eff} \\
&\sum_{i=1}^{K} \lambda_i w_i^\ell \leq -\varepsilon \quad \forall \ell \in \bar{J}^{-}_{eff} \\
&\sum_{i=1}^{K} \lambda_i w_i^h = 0 \quad \forall h \in \bar{J}^{=}_{eff} \\
&\qquad \lambda_i \geq \varepsilon \qquad i=1,\dots,K \\
&\qquad \sum_{i=1}^{K} \lambda_i = 1
\end{aligned}
$$

Si $\bar{J}^{+}_{eff} = \emptyset$, alors la solution actuelle $\bar{x}$ est la solution cherchée, sinon on trouve une solution réalisable $\bar{\lambda}$ du système (S_1) et on résoud le programme linéaire :

$$\underset{x \in P}{\mathrm{Max}} \quad \bar{\lambda} C x$$

pour trouver une nouvelle solution efficace extrême.
On recommence alors une itération.

Considérations pratiques

Pour déterminer simultanément ε et une solution réalisable du système S_1, on résoud le programme linéaire suivant :

$$\text{(Lp1)}\quad \begin{array}{ll} \text{Max } \varepsilon & \\ \sum_{i=1}^{K} \lambda_i w_i^j - \varepsilon \geq 0 & \forall j \in J^+_{eff} \\ \sum_{i=1}^{K} \lambda_i w_i^\ell + \varepsilon \leq 0 & \forall \ell \in J^-_{eff} \\ \sum_{i=1}^{K} \lambda_i w_i^h = 0 & \forall h \in J^=_{eff} \\ \lambda_i - \varepsilon \geq 0 \quad , & i=1,\dots,K \\ \sum_{i=1}^{K} \lambda_i = 1 & \end{array}$$

Pour résoudre l'exemple donné dans [9] :

$$\begin{aligned} &\text{Max } f_1(x) = -0.225x_1 - 2.2x_2 - 0.8x_3 - 0.1x_4 - 0.05x_5 - 0.26x_6 \\ &\text{Max } f_2(x) = -10x_1 - 20x_2 - 120x_3 \\ &\text{Max } f_3(x) = -24x_1 - 27x_2 - 15x_4 - 1.1x_5 - 52x_6 \end{aligned}$$

sous les contraintes :

$$\text{(P)}\quad \begin{aligned} &720x_1 + 107x_2 + 7080x_3 + 134x_5 + 1000x_6 \geq 5000 \\ &0.2x_1 + 10.1x_2 + 13.2x_3 + 0.75x_4 + 0.15x_5 + 1.2x_6 \geq 12.5 \\ &344x_1 + 460x_2 + 1040x_3 + 75x_4 + 17.4x_5 + 240x_6 \geq 2500 \\ &18x_1 + 151x_2 + 78x_3 + 2.5x_4 + 0.2x_5 + 4x_6 \geq 63 \\ &x_1 \leq 6 \text{ , } x_2 \leq 1 \text{ , } x_3 \leq 0.25 \text{ , } x_4 \leq 10 \text{ , } x_5 \leq 10 \text{ , } x_6 \leq 4 \\ &x_i \geq 0 \quad , \quad i=1,\dots,6 \end{aligned}$$

Hwang et Masud choisissent arbitrairement $\varepsilon=0.001$, pour trouver une solution réalisable du système S_1, et aboutissent à une solution finale ($x_1=4.10$, $x_2=0.14$, $x_3=0.10$, $x_4=10$, $x_5=10$, $x_6=0$) différente de la solution du décideur qui devait être la solution optimale du programme suivant :

$$\text{(Lp2)}\quad \begin{array}{c} \text{Max } \quad 0.959\, f_1(x) + 0.029\, f_2(x) + 0.012\, f_3(x) \\ x \in P \end{array}$$

où P est l'ensemble des solutions réalisables.

Pour le même exemple, au lieu de choisir arbitrairement ε, nous avons résolu à chaque itération le programme linéaire (Lp1), et nous avons obtenu comme solution finale la solution optimale du programme (Lp2) qui est ($x_1=4.03$, $x_2=0$, $x_3=0.25$, $x_4=10$, $x_5=5.96$, $x_6=0$).

4.4 Algorithme interactif du simplexe

Notations :

Soit le programme :

$$\text{(MOLP)} \quad \text{Max } Cx \qquad \begin{aligned} Ax &= b \\ x &\geqslant 0 \end{aligned}$$

où C est une matrice Kxn, A une matrice mxn (m<n), b un vecteur de dimension m et x un vecteur de dimension n.

Notons par P l'ensemble des solutions réalisables $\{x \in R^n | Ax=b \ , \ x \geqslant 0\}$.

Soit $\bar{x}$ un point extrême de P, solution efficace du programme (MOLP). La matrice A se décompose en $A=(\bar{B}\ N)$, où $\bar{B}$ est une matrice régulière de dimension mxm et N une matrice mx(n-m) (voir proposition 1 du paragraphe 1.4 : caractérisation d'un point extrême).

Appelons J l'ensemble des indices des variables de base et $\bar{J}$ l'ensemble des indices des variables hors base.

Description d'une itération de l'algorithme

Déterminons la matrice $Z = C_N - C_{\bar{B}}\bar{B}^{-1}N$

où la matrice C_N est la matrice des coefficients des variables hors base et $C_{\bar{B}}$ la matrice des coefficients des variables de base.

Pour tout $j \in \bar{J}$, on résoud le programme linéaire :

$$\begin{aligned} &\text{Max} \sum_{i=1}^{K} v_i \\ &Zd_N - v = z^j \\ &d_N \geq 0 \quad , \quad v \geq 0 \end{aligned}$$

Si la valeur optimale de la fonction objectif est nulle, alors la variable hors base $\bar{x}_{Nj}$ est efficace et son introduction dans la base donne une solution efficace adjacente à $\bar{x}$ (voir proposition 11 du paragraphe 3.2).

Remarques :

1) Pour tout $j \in \bar{J}$, on a $z^j \not\geq 0$, car s'il existait un $j \in \bar{J}$ tel que $z^j \geq 0$, alors l'introduction de la variable hors base $\bar{x}_{Nj}$ dans la base donnerait une solution adjacente x telle que $Cx \geq C\bar{x}$, ce qui contredirait l'efficacité de $\bar{x}$.

2) S'il existe $j \in \bar{J}$ tel que $z^j \leq 0$, alors la variable hors base $\bar{x}_{Nj}$ n'est pas une variable efficace. En effet, si l'on introduit $\bar{x}_{Nj}$ dans la base on obtiendrait une solution x adjacente à $\bar{x}$, telle que $Cx \leq C\bar{x}$.

Notons pas $\bar{J}_{eff}$ l'ensemble des indices des variables efficaces, $\bar{J}_{eff}$ est contenu dans $\bar{J}$.

Posons $Y = \bar{B}^{-1}N$. Pour tout $j \in \bar{J}_{eff}$ on détermine la valeur

$$\alpha_j = \min \left\{ \frac{\bar{x}_{\bar{B}i}}{y_i^j} \quad , \quad y_i^j > 0 \right\}$$

et on présente au décideur les variations $\alpha_j z^j$ sur les K critères. Le décideur doit alors choisir une des variables qui entrera dans la base.

Soit $\bar{x}_{Nk}$ la variable efficace choisie, et

$$\alpha_k = \min_i \left\{\frac{\bar{x}_{\bar{B}i}}{y_i^k} \quad , \quad y_i^k > 0\right\} = \frac{\bar{x}_{\bar{B}r}}{y_r^k}$$

Donc, la variable de base $\bar{x}_{\bar{B}r}$ quitte la base et la variable hors base $\bar{x}_{Nk}$ rentre dans la base (voir algorithme du Simplexe, paragraphe 2.1). La nouvelle solution sera :

$$\bar{x}_{\bar{B}i} = \bar{x}_{\bar{B}i} - \frac{\bar{x}_{Br}}{y_r^k} \; y_i^k \neq 0 \quad \text{pour} \quad i \in J \quad , \quad i \neq r$$

$$\bar{x}_{\bar{B}r} = \bar{x}_{\bar{B}r} - \frac{\bar{x}_{\bar{B}r}}{y_r^k} \; y_r^k = 0$$

$$\bar{x}_{N\ell} = 0 \quad \text{pour } \ell \in \bar{J} \quad , \ \ell \neq k$$

$$\bar{x}_{Nk} = \frac{\bar{x}_{\bar{B}r}}{y_r^k}$$

Exemple

Enoncé :

Soient les fonctions objectifs suivantes à maximiser :

$$\text{Max} \quad f_1 = 3x_1 + x_2 + 2x_3 + x_4$$

$$\text{Max} \quad f_2 = x_1 - x_2 + 2x_3 + 4x_4$$

$$\text{Max} \quad f_3 = -x_1 + 5x_2 + x_3 + 2x_4$$

sous les contraintes :

$$2x_1 + x_2 + 4x_3 + 3x_4 + x_5 = 60$$

$$3x_1 + 4x_2 + x_3 + 2x_4 + x_6 = 60$$

$$x_i \geq 0 \quad , \quad i=1,\ldots,6$$

On a :

$$C = \begin{bmatrix} 3 & 1 & 2 & 1 & 0 & 0 \\ 1 & -1 & 2 & 4 & 0 & 0 \\ -1 & 5 & 1 & 2 & 0 & 0 \end{bmatrix} \qquad A = \begin{bmatrix} 2 & 1 & 4 & 3 & 1 & 0 \\ 3 & 4 & 1 & 2 & 0 & 1 \end{bmatrix}$$

<u>Solution initiale</u> :

$x_5 = 60 \quad , \quad x_6 = 60$

$$B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad , \quad B^{-1}N = \begin{bmatrix} 2 & 1 & 4 & 3 \\ 3 & 4 & 1 & 2 \end{bmatrix} \quad , \quad C_N - C_B B^{-1} N = \begin{bmatrix} 3 & 1 & 2 & 1 \\ 1 & -1 & 2 & 4 \\ -1 & 5 & 1 & 2 \end{bmatrix}$$

Seules les directions suivant les variables x_3 et x_4 sont intéressantes, car il y a augmentation sur tous les critères.

Choisissons la direction suivant la variable x_4 . La variable x_4 rentre donc dans la base et la variable x_5 sort de la base, car $\alpha = \frac{60}{3}$ est inférieure à $\frac{60}{2}$.

<u>La nouvelle solution est</u> (solution efficace) :

$$x_4 = \underline{20} \quad , \quad x_6 = 60 - (2 \cdot 20) = \underline{20}$$

avec $f_1 = 20 \quad , \quad f_2 = 80 \quad , \quad f_3 = 40$

On calcule ensuite :

$$B = \begin{bmatrix} 3 & 0 \\ 2 & 1 \end{bmatrix} \quad , \quad B^{-1} = \begin{bmatrix} 1/3 & 0 \\ -2/3 & 1 \end{bmatrix} \quad , \quad B^{-1}N = \begin{bmatrix} 1/3 & 0 \\ -2/3 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 4 & 1 \\ 3 & 4 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 2/3 & 1/3 & 4/3 & 1/3 \\ 5/3 & 10/3 & -5/3 & -2/3 \end{bmatrix}$$

$$C_B B^{-1} N = \begin{bmatrix} 1 & 0 \\ 4 & 0 \\ 2 & 0 \end{bmatrix} \begin{bmatrix} 2/3 & 1/3 & 4/3 & 1/3 \\ 5/3 & 10/3 & -5/3 & -2/3 \end{bmatrix} = \begin{bmatrix} 2/3 & 1/3 & 4/3 & 1/3 \\ 8/3 & 4/3 & 16/3 & 4/3 \\ 4/3 & 2/3 & 8/3 & 2/3 \end{bmatrix}$$

$$C_N - C_B B^{-1} N = \begin{bmatrix} 3 & 1 & 2 & 0 \\ 1 & -1 & 2 & 0 \\ -1 & 5 & 1 & 0 \end{bmatrix} - \begin{bmatrix} 2/3 & 1/3 & 4/3 & 1/3 \\ 8/3 & 4/3 & 16/3 & 4/3 \\ 4/3 & 2/3 & 8/3 & 2/3 \end{bmatrix} = \begin{bmatrix} 7/3 & 2/3 & 2/3 & -1/3 \\ -5/3 & -7/3 & -10/3 & -4/3 \\ -7/3 & 13/3 & -5/3 & -2/3 \end{bmatrix}$$

On voit que la direction suivant la variable hors base x_5 n'est pas une direction efficace, car si x_5 rentre dans la base on a une diminution sur tous les critères.

La direction suivant la variable hors base x_3 n'est pas efficace non plus, car la valeur optimale du programme

$$\text{Max } \; v_1 + v_2 + v_3$$

$$\begin{array}{ll} \text{sous :} & 7/3d_1 + 2/3d_2 + 2/3d_3 - 1/3d_4 - v_1 = 2/3 \\ \text{(Lp)} & -5/3d_1 - 7/3d_2 - 10/3d_3 - 4/3d_4 - v_2 = -10/3 \\ & -7/3d_1 + 13/3d_2 - 5/3d_3 - 2/3d_4 - v_3 = -5/3 \\ & d_1 \geq 0 \; , \; d_2 \geq 0 \; , \; d_3 \geq 0 \; , \; d_4 \geq 0 \; , \; v_1 \geq 0 \; , \; v_2 \geq 0 \; , \; v_3 \geq 0 \end{array}$$

est égale à 8.1456 donc strictement supérieure à 0.

On peut par contre montrer que les directions suivant x_1 et x_2 sont des directions efficaces. Pour cela il suffit de remplacer dans le programme linéaire (Lp) le membre de droite par le vecteur (7/3 , -5/3 , -7/3) respectivement (2/3 , -7/3 , 13/3).

Laissons le choix de l'introduction de l'une ou de l'autre variable au décideur. Pour l'aider, nous lui dirons que si on introduit x_1 dans la base, le critère f_1 augmente de 28 ($\frac{20}{5/3} \cdot 7/3 = 28$), le critère f_2 diminue de 20 ($\frac{20}{5/3} \cdot -5/3 = -20$) et le critère f_3 diminue de 28 ($\frac{20}{5/3} \cdot -7/3 = -28$), par contre, si on introduit la variable x_2 dans la base, alors le critère f_1 augmente de 4 ($\frac{20}{10/3} \cdot 2/3 = 4$), le critère f_2 diminue de 14 ($\frac{20}{10/3} \cdot -7/3 = -14$), et le critère f_3 augmente de 26 ($\frac{20}{10/3} \cdot 13/3 = 26$).

Supposons que le décideur choisit la variable x_2. La nouvelle solution est

$$x_2 = \frac{20}{10/3} = \underline{6} \quad , \quad x_4 = 20 - (6.1/3) = \underline{18}$$

avec $f_1 = 24$, $f_2 = 66$ et $f_3 = 66$.

Si le décideur n'est pas satisfait de cette solution, on fait une nouvelle itération en partant de la base associée à cette solution.

4.5. La méthode de STEUER

Préliminaires :

Si le nombre de fonctions objectifs est K, cette méthode génère à chaque itération 2K+1 solutions efficaces et propose au décideur d'en choisir la meilleure.

Soit donc le programme :

$$(MOLP_1) \quad \max_{x \in P} Cx$$

où C est la matrice des coefficients des fonctions objectifs, de dimension Kxn, x un vecteur de R^n et P l'ensemble des solutions réalisables.

Générer toutes les solutions efficaces du programme ($MOLP_1$) reviendrait à résoudre la famille des programmes linéaires, paramétrée par λ :

$$\max_{x \in P} \lambda Cx \quad , \quad \lambda \in \bar{\Lambda}$$

où $\bar{\Lambda} = \{\lambda \in R^K | \lambda_i \in]0,1[, \sum_{i=1}^{K} \lambda_i = 1\}$

Le nombre de programmes à résoudre est très grand, et toutes les solutions générées n'intéressent pas le décideur.

Si le décideur peut donner pour chaque λ_i, i=1,...,K une borne inférieure a_i et une borne supérieure b_i, les solutions qui intéressent le décideur sont les solutions optimales des programmes linéaires suivants :

$$(Lp\lambda) \quad \max_{x \in P} \lambda Cx \quad , \quad \lambda \in \Lambda^*$$

où $\Lambda^* = \{\lambda \in R^K | \lambda_i \in]a_i, b_i[, \sum_{i=1}^{K} \lambda_i = 1\}$

Remarquons que les bornes a_i et b_i doivent être choisies de telle manière que l'ensemble Λ^* ne soit pas vide. Cette condition peut être traduite par les relations :

$$i \; , \; a_i < b_i$$

$$\sum_{i=1}^{K} a_i \leqslant 1 \leqslant \sum_{i=1}^{K} b_i$$

Définissons maintenant ce qu'on appelle un <u>vecteur de poids critiques</u>, noté π. Un tel vecteur vérifie pour un indice m, $1 \leqslant m \leqslant K$, convenablement choisi, les relations :

(1) $i=1,\dots,K \; ; \; i \neq m \qquad \pi_i = a_i \quad \text{ou} \quad \pi_i = b_i$

(2) $\qquad \pi_m = 1 - \sum_{i \neq m} \pi_i \, .$

(3) $\qquad a_m \leqslant \pi_m \leqslant b_m$

On va supposer qu'il existe q vecteurs de ce type qu'on notera $\pi_j, \pi_j = (\pi_{j1}, \pi_{j2}, \dots, \pi_{jK})$.

<u>Proposition 1</u> :

Les vecteurs π_j, $j=1,\dots,q$ sont les vecteurs extrêmes de l'ensemble fermé

$\Lambda = \{\lambda \varepsilon \; R^K \,|\, \lambda_i \varepsilon [a_i, b_i], \Sigma\lambda_i = 1\}$

<u>Preuve</u> :

a) Montrons tout d'abord que tout vecteur π_j obtenu par les relations (1), (2) et (3) est un vecteur extrême de Λ. Supposons qu'il existe deux vecteurs de Λ, λ_1 et λ_2, $\lambda_1 \neq \lambda_2$ et un scalaire $\alpha, 0 < \alpha < 1$ tel que

$\pi_j=\alpha\lambda_1+(1-\alpha)\lambda_2$.

Alors pour tout $i, i\neq m, i=1,\dots,K$ on a :

$$\pi_{ji}=\alpha\lambda_{1i}+(1-\alpha)\lambda_{2i}=a_i \text{ ou } \pi_{ji}=\alpha\lambda_{1i}+(1-\alpha)\lambda_{2i}=b_i$$

et pour $i=m$, on a : $\pi_{jm}=\alpha\lambda_{1m}+(1-\alpha)\lambda_{2m}=1-\sum_{i\neq m}\pi_{ji}$

or $\quad a_i\leq\lambda_{1i}\leq b_i$ et $a_i\leq\lambda_{2i}\leq b_i$

d'où $\quad a_i\leq\alpha\lambda_{1i}+(1-\alpha)\lambda_{2i}\leq b_i$

et par conséquent on ne peut avoir $\alpha\lambda_{1i}+(1-\alpha)\lambda_{2i}=a_i$ ou

$$\alpha\lambda_{1i}+(1-\alpha)\lambda_{2i}=b_i \text{ que si } \lambda_{1i}=\lambda_{2i}=a_i \text{ ou } \lambda_{1i}=\lambda_{2i}=b_i$$

comme $\Sigma\lambda_{1i}=\Sigma\lambda_{2i}=1$, alors $\lambda_{1m}=\lambda_{2m}=1-\sum_{i\neq m}\pi_{ji}$

et donc $\lambda_1=\lambda_2=\pi_j$, ce qui contredit l'hypothèse que $\lambda_1\neq\lambda_2$ et par conséquent π_j est un vecteur extrême.

b) Montrons que tout vecteur $\bar{\lambda}\in\Lambda$ et $\bar{\lambda}\neq\pi_j, j=1,\dots,q$ n'est pas un vecteur extrême de Λ. Supposons que $\bar{\lambda}$, $\bar{\lambda}\neq\pi_j, j=1,\dots,q$ est un vecteur extrême de Λ.

D'après les relations (1), (2) et (3), il existe au moins deux indices h et p, $h\neq p$, et $h,p\in\{1,\dots,K\}$ tel que

$\bar{\lambda}_h\neq a_h$, $\bar{\lambda}_h\neq b_h$ et $\bar{\lambda}_p\neq a_p$, $\bar{\lambda}_p\neq b_p$.

Soit $\quad \delta=\min\{(\bar{\lambda}_h-a_h),(b_h-\bar{\lambda}_h),(\bar{\lambda}_p-a_p),(b_p-\bar{\lambda}_p)\}>0$

et soient deux vecteurs λ_1 et λ_2 tels que

pour $i\neq h, i\neq p, i=1,\dots,K \qquad \lambda_{1i}=\lambda_{2i}=\bar{\lambda}_i$

$\lambda_{1h}=\bar{\lambda}_h+\delta$ et $\lambda_{2h}=\bar{\lambda}_h-\delta$

$\lambda_{1p}=\bar{\lambda}_p-\delta$ et $\lambda_{2p}=\bar{\lambda}_p+\delta$

pour $i\neq h, i\neq p, i=1,\ldots,K$ $\quad a_i\leqslant\lambda_{1i}=\lambda_{2i}=\bar{\lambda}_i\leqslant b_i$ car $\bar{\lambda}\varepsilon\Lambda$

$\lambda_{1h}=\bar{\lambda}_h+\delta>a_h$ car $\bar{\lambda}_h>a_h$ et $\varepsilon>0$

en outre $\varepsilon\leqslant b_h-\bar{\lambda}_h$ donc $\lambda_{1h}=\bar{\lambda}_h+\varepsilon\leqslant b_h$

d'où $a_h<\lambda_{1h}\leqslant b_h$

De même $\lambda_{2h}=\bar{\lambda}_h-\delta\geqslant\bar{\lambda}_h-(\bar{\lambda}_h-a_h)=a_h$ et $\lambda_{2h}=\bar{\lambda}_h-\delta<b_h$ car $\bar{\lambda}_h<b_h$ et $\varepsilon>0$

nous donne $a_h\leqslant\lambda_{2h}<b_h$

On démontre de la même manière que λ_{1p} et λ_{2p} sont telles que

$a_p\leqslant\lambda_{1p}<b_p$ et $a_p<\lambda_{2p}\leqslant b_p$

De plus $\sum_{i=1}^{K}\lambda_{1i} = \sum_{\substack{i\neq h\\ i\neq p}}^{K}\bar{\lambda}_i+(\bar{\lambda}_h+\delta)+(\bar{\lambda}_p-\delta)=\sum_{i=1}^{K}\bar{\lambda}_i=1$

et $\sum_{i=1}^{K}\lambda_{2i}=\sum_{\substack{i\neq h\\ i\neq p}}\bar{\lambda}_i+(\bar{\lambda}_h-\delta)+(\bar{\lambda}_p+\delta)=\sum_{i=1}^{K}\bar{\lambda}_i=1$

Donc λ_1 et λ_2 sont 2 vecteurs différents de Λ.

Or $\bar{\lambda}=0.5\lambda_1 + 0.5\lambda_2$, ce qui contredit l'hypothèse que $\bar{\lambda}$ est un vecteur extrême de Λ.

L'ensemble $\Lambda=\{\lambda\varepsilon\ R^K|\lambda_i\varepsilon[a_i,b_i],\ \sum_{i=1}^{K}\lambda_i=1\}$ est un ensemble convexe et tout vecteur $\lambda\varepsilon\Lambda$ peut s'écrire comme une combinaison convexe des vecteurs extrêmes $\pi_j, j=1,\ldots,q$.

Notons par Π , la matrice qxK dont les lignes sont formées par les vecteurs $\pi_j, j=1,\ldots,q$. L'ensemble Λ peut s'écrire maintenant :

(4) $$\Lambda=\{\lambda\varepsilon\ R^K|\lambda=\alpha\Pi\ ,\ \alpha_i\varepsilon[0,1], i=1,\ldots,q\ ,\ \sum_{i=1}^{q}\alpha_i=1\}$$

$\Lambda^*=\{\lambda\varepsilon\ R^K|\lambda_i\varepsilon]a_i,b_i[,\ \sum_{i=1}^{K}\lambda_i=1\}$ est l'intérieur de Λ et peut s'écrire :

(5) $$\Lambda^*=\{\lambda\varepsilon\ R^K|\lambda=\alpha\Pi\ ,\ \alpha_i\varepsilon]0,1[, i=1,\ldots,q,\ \sum_{i=1}^{q}\alpha_i=1\}$$

Notons par D, la matrice qxn :

$$D = \Pi C$$

Proposition 2 :

Toute solution optimale d'un programme de la famille de programmes ($Lp\lambda$) est solution efficace du programme multi-objectif suivant :

($MOLP_2$) $$\max_{x\varepsilon P} Dx$$

et réciproquement.

Preuve :

Soit x_0 une solution optimale du programme ($Lp\lambda$) pour $\lambda=\lambda_0$.
On a alors pour tout $x\varepsilon P$, $\lambda_0 Cx<\lambda_0 Cx_0$.

λ_0 est un vecteur de Λ^* et d'aprés (5) il existe un vecteur

$\alpha_o, o<\alpha_{oi}<1, i=1,\ldots,q$ et $\sum_{i=1}^{q} \alpha_{oi}=1$ tel que $\lambda_o=\alpha_o\Pi$

D'où pour tout $x\varepsilon P$, $\alpha_o\Pi Cx \leqslant \alpha_o\Pi Cx_o$

Or $\Pi C=D$, donc pour tout $x\varepsilon P$, $\alpha_o Dx \leqslant \alpha_o Dx_o$

Donc x_o est solution optimale du programme linéaire $\underset{x\varepsilon P}{\text{Max}}\ \alpha_o\ Dx$ avec $o<\alpha_{oi}<1$, et $\Sigma\alpha_{oi}=1$ et d'après la proposition 5 du paragraphe 3.2, elle est aussi solution efficace du programme ($MOLP_2$).

La réciproque se démontre de façon analogue.

La matrice D est de dimension qxn (généralement $q > K$) et donc le programme multiobjectif $\underset{x\varepsilon P}{\text{Max}}\ Dx$ est un programme avec q fonctions objectifs non toutes linéairement indépendantes, car q-K lignes de D sont linéairement dépendantes des autres (on suppose que C est de rang K). Pour réduire la taille du problème et se ramener à la résolution d'un programme avec K fonctions objectifs, la proposition suivante est nécessaire :

<u>Proposition 3</u> :

Soient π_j un vecteur donné par les relations (1), (2) et (3) et l'ensemble fermé

$\Lambda = \{\lambda\varepsilon R^K | \lambda_i\varepsilon[a_i,b_i], i=1,\ldots,K, \sum_{i=1}^{K}\lambda_i=1\}$

a) Si pour $i\varepsilon\{1,\ldots,K\}, \pi_{ji}=a_i$ et $a_i<1-\sum_{h\neq i}^{K} b_h$ alors $\pi_j \notin \Lambda$

b) Si pour $i\varepsilon\{1,\ldots,K\}, \pi_{ji}=b_i$ et $b_i>1-\sum_{h\neq i}^{K} a_h$ alors $\pi_j \notin \Lambda$

Preuve :

a) $$\sum_{h=1}^{K} \pi_{jh}=a_i+\sum_{h\neq i}^{K} \pi_{jh}\leq a_i+\sum_{h\neq i}^{K} b_h \text{ car } \sum_{h\neq i}^{K} \pi_{jh}\leq \sum_{h\neq i}^{K} b_h$$

et comme $a_i<1-\sum_{h\neq i}^{K} b_h$ alors on a :

$$\sum_{h=i}^{K} \pi_{jh}<1-\sum_{h\neq i}^{K} b_h+\sum_{h\neq i}^{K} b_h=1 \text{ et donc } \pi_j \notin \Lambda.$$

b) Idem que sous (a)

Soit pour tout $i, i=1,\ldots,K$ $\quad a_j'=\mathrm{Max}\{a_i, 1-\sum_{h\neq i}^{K} b_h\}$ et $b_j'=1$

et soit l'ensemble $\mathcal{X}=\{\lambda\varepsilon R^K|\lambda_i\varepsilon[a_j',b_j'], i=1,\ldots,K, \sum_{i=1}^{K} \lambda_i=1\}$

Montrons que l'ensemble $\mathcal{X}$ a exactement K vecteurs extrêmes. Pour ce faire, considérons les 3 cas suivants :

Cas 1 : (cas général)

S'il existe au moins un $p\varepsilon\{1,\ldots,K\}$ et un $\ell\varepsilon\{1,\ldots,K\}, p\neq\ell$ tel que $a_p'\neq 0$ et $a_\ell'\neq 0$, alors on a pour tout $i\varepsilon\{1,\ldots,K\}$ $b_i'>1-\sum_{h\neq i}^{K} a_h'$.

D'après les relations (1), (2) et (3) et la proposition 3b, les vecteurs extrêmes de $\mathcal{X}$ sont donnés par les relations suivantes : pour m convenablement choisi dans $\{1,\ldots,K\}$,

(6) $\quad$ pour $i=1,\ldots,K; i\neq m$ $\quad \pi_{ji}=a_i'$

(7) $\quad \pi_{jm}=1-\sum_{i\neq m}^{K} \pi_{ji}=1-\sum_{i\neq m}^{K} a_i'$

(8) $\quad a_m'\leq\pi_{jm}\leq b_m'$

Remarque : La relation (8) est toujours vérifiée car si $\mathcal{X}\neq\emptyset$, on a :

$$\sum_{i=1}^{K} a'_i\leq 1 \quad \text{ou} \quad \sum_{i\neq m}^{K} a'_i+a'_m\leq 1$$

et donc $\quad a'_m\leq 1-\sum_{i\neq m}^{K} a'_i=\pi_{jm}\leq b'_m=1$

Donc le nombre de vecteurs extrêmes de $\mathcal{X}$ est égal au choix d'un indice parmi K indices c'est-à-dire $\binom{K}{1}=K$.

Cas 2 :

Si pour $p\in\{1,\ldots,K\}$ on a $a'_p\neq 0$ et pour $i=1,\ldots,K,i\neq p,a'_i=0$ alors pour tout $i\in\{1,\ldots,K\},i\neq p$ on a $b'_i>1-\sum_{h\neq i}^{K} a'_h$.

Et toujours d'après les relations (1), (2) et (3) et la proposition 3b, les vecteurs extrêmes de $\mathcal{X}$ sont donnés par :

pour m convenablement choisi dans $\{1,\ldots,K\}$

Si $m\neq p$, on a :

a) pour $i=1,\ldots,K;i\neq m$ $\quad \pi_{ji}=a'_i$

$$0<\pi_{jm}=1-\sum_{i\neq m}^{K}\pi_{ji}=1-a'_p<1$$

ou

b) pour $i=1,\ldots,K;i\neq m,i\neq p$ $\quad \pi_{ji}=a'_i=0$

$$\pi_{jp}=b'_p=1$$

$$a'_m=\pi_{jm}=1-\sum_{i\neq m}^{K}\pi_{ji}=1-1=0$$

et si $m=p$ on a :

pour $i=1,\ldots,K;i\neq p$ $\quad \pi_{ji}=a'_i=0$ et $\pi_{jp}=1-\sum_{i\neq p}^{K}\pi_{ji}=1$

On constate que le dernier vecteur est égal aux vecteurs calculés sous b qui sont d'ailleurs tous égaux et finalement les vecteurs extrêmes de $\mathcal{X}$ sont donnés par les relations :

pour m convenablement choisi dans {1,...,K} on a :

$$\text{pour } i=1,\dots,K; i\neq m \quad \pi_{ji}=a'_i$$

$$\text{et} \qquad a'_m \leq \pi_{jm} = 1 - \sum_{i\neq m} a'_i \leq b'_m$$

qui sont en fait les relations (6), (7) et (8).

Cas 3 :

Si pour tout $i\in\{1,\dots,K\}$ $a'_i=0$ alors $\mathcal{X}$ a les K vecteurs extrêmes suivants :

$$(1,0,\dots,0,\dots,0),(0,1,\dots,0,\dots,0),\dots,(0,0,\dots,1,\dots,0),$$
$$\dots,(0,0,\dots,0,\dots,1)$$

qu'on obtient aussi par les relations (6), (7) et (8).

Soit Π' la matrice KxK dont les lignes sont les K vecteurs extrêmes de $\mathcal{X}$. Notons par E la matrice Kxn :

$$E = \Pi' C$$

En vertu de la proposition 2, toute solution optimale du programme paramétré

$$\underset{x\in P}{\text{Max}}\ \lambda Cx, \text{ avec } \lambda\in\mathcal{X}\text{int}=\{\lambda\in R^K \mid \lambda_i\in]a'_i,b'_i[, i=1,\dots,K, \sum_{i=1}^{K}\lambda_i=1\}$$

est solution efficace du programme multiobjectif ($MOLP_3$) $\underset{x\in P}{\text{Max}}\ Ex$..

Or, d'après la proposition 3, $\mathcal{X}$int contient Λ^* et donc (voir proposition 4 du paragraphe 3.2) l'ensemble des solutions efficaces du programme ($MOLP_3$) contient l'ensemble des solutions efficaces du programme ($MOLP_2$).

Algorithme

Pour tout $i\varepsilon\{1,\ldots,K\}$ on pose $a_i'=\max\{a_i, 1-\sum_{h\neq i}^{K} b_h\}$ et $b_i'=1$.

On calcule par les relations (6), (7) et (8) les K vecteurs critiques (vecteurs extrêmes de l'ensemble $\Lambda=\{\lambda\varepsilon\ R^K \mid \lambda_i\varepsilon[a_i', b_i'], i=1,\ldots,K, \sum_{i=1}^{K}\lambda_i=1\}$)

On forme la matrice Π' dont les lignes sont les K vecteurs critiques et on calcule $E=\Pi'C$.

A l'aide des 2K+1 vecteurs de poids λ suivants :

$$
\begin{aligned}
&\lambda_1=(1,0,\ldots,0,\ldots,0)\\
&\lambda_2=(0,1,\ldots,0,\ldots,0)\\
&\vdots\\
&\lambda_i=(0,0,\ldots,1,\ldots,0)\\
&\vdots\\
&\lambda_K=(0,0,\ldots,0,\ldots,1)\\
&\lambda_{K+1}=(1/K^2,r,\ldots,r,\ldots,r) \text{ ou } r=K+1/K^2\\
&\lambda_{K+2}=(r,1/K^2,\ldots,r,\ldots,r)\\
&\vdots\\
&\lambda_{K+i}=(r,r,\ldots,1/K^2,\ldots,r)\\
&\vdots\\
&\lambda_{2K}=(r,r,\ldots,r,\ldots,1/K^2)\\
&\lambda_{2K+1}=(1/K,1/K,\ldots,1/K,\ldots,1/K)
\end{aligned}
$$

On génère 2K+1 solutions efficaces du programme multiobjectif $\underset{x\varepsilon P}{\text{Max}}\ Ex$, qu'on présentera au décideur qui doit choisir la meilleure. S'il est satisfait de la solution choisie alors on arrête sinon on contracte l'enveloppe convexe des K lignes de E autour du gradient $\lambda_i E$,

$i\varepsilon\{1,2,\ldots,2K+1\}$, gradient de la fonction objectif dont la solution optimale sur P est la solution choisie.

Remarque : Les K premiers vecteurs de poids λ seront modifiés de la façon suivante :

> $\lambda_i=(\varepsilon,\varepsilon,\ldots,1-(K-1)\varepsilon,\ldots,\varepsilon), i=1,\ldots,K$ et ε un scalaire positif assez petit pour assurer l'efficacité des solutions générées.

Contraction de l'enveloppe convexe des K lignes de E

Si la solution efficace préférée par le décideur est la solution optimale du programme linéaire

$$\max_{x\varepsilon P} \lambda_h Ex$$

où λ_h est un des 2K+1 vecteurs de poids λ, alors
la contraction de l'enveloppe convexe des lignes de E se fera autour du gradient $\lambda_h E$ et l'enveloppe convexe réduite qu'on obtient est l'enveloppe convexe des K lignes de la matrice $E'=\Pi_0 E$, où Π_0 est la matrice des vecteurs de poids critiques construits à partir des K intervalles :

$$i=1,\ldots,K; i\neq h \; [a_i'=0, b_i'=1] \text{ et } [a_h'=1-\bar{\alpha}, b_h'=1] \quad \text{si} \quad h\varepsilon\{1,\ldots,K\}$$

ou

$$i=1,\ldots,K; i\neq h-K \; [a_i'=\frac{1-\bar{\alpha}}{\ell}, b_i'=1] \text{ et } [a_{h-K}'=0, b_{h-K}'=1]$$

si $h\varepsilon\{K+1,K+2,\ldots,2K\}$

ou

$$i=1,\ldots,K \; [a_i'=\frac{1-\bar{\alpha}}{K}, b_i'=1] \quad \text{si} \quad h=2K+1$$

où $\bar{\alpha} = \dfrac{1}{K^{1/K-1}}$ est le facteur de réduction utilisé et $\ell=K-1$.

(pour l'obtention de ces intervalles voir l'exemple numérique de la contraction d'une enveloppe convexe).

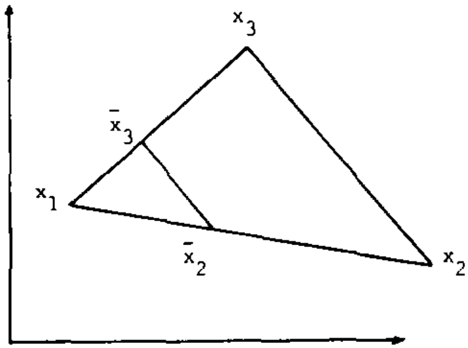
On recommence une nouvelle itération en générant 2K+1 solutions efficaces du programme $\underset{x \varepsilon P}{\mathrm{Max}}\ E'x$ qu'on présente au choix du décideur.

<u>Exemple d'une contraction d'une enveloppe convexe de 3 vecteurs de R^2</u>

Soient x_1, x_2 et x_3 trois points de R^2.

$K=3$, $\ell=2$ et on notera $\bar{\alpha}$ le facteur de réduction ($\bar{\alpha} \geq \frac{1}{2}$).

a) Contraction autour d'un point extrême (par exemple x_1)



<u>Figure 1</u>

Appelons $H(x_1, x_2, x_3)$ l'enveloppe de x_1, x_2 et x_3.

Tout point $x \varepsilon H(x_1, x_2, x_3)$ s'écrit $x=\lambda_1 x_1+\lambda_2 x_2+\lambda_3 x_3$ avec

$\lambda_1,\lambda_2,\lambda_3 \varepsilon [0,1]$ et $\sum_{i=1}^{3} \lambda_i = 1$, pour $\lambda_1=1$, $\lambda_2=0$ et $\lambda_3=0$ on a $x=x_1$.

Soient $\bar{x}_2=(1-\bar{\alpha})x_1+\bar{\alpha}x_2$ et $\bar{x}_3=(1-\bar{\alpha})x_1+\bar{\alpha}x_3$

Essayons de trouver l'expression d'un point $x\varepsilon H(x_1,\bar{x}_2,\bar{x}_3)$ en fonction de x_1,x_2 et x_3

Tout $x\varepsilon H(x_1,\bar{x}_2,\bar{x}_3)$ s'écrit $x= {}_1x_1+\lambda_2\bar{x}_2+\lambda_3\bar{x}_3$ avec $\lambda_1,\lambda_2,\lambda_3 \varepsilon [0,1]$ et $\sum_{i=1}^{3} \lambda_i=1$

ou encore $x=\lambda_1x_1+\lambda_2((1-\bar{\alpha})x_1+\bar{\alpha}x_2)+\lambda_3((1-\bar{\alpha})x_1+\bar{\alpha}x_3)$

$$=(\lambda_1+\lambda_2(1-\bar{\alpha})+\lambda_3(1-\bar{\alpha}))x_1+\lambda_2\bar{\alpha}x_2+\lambda_3\bar{\alpha}x_3$$

En posant $\alpha_1=\lambda_1+\lambda_2(1-\bar{\alpha})+\lambda_3(1-\bar{\alpha})$, $\alpha_2=\lambda_{2\bar{\alpha}}$ et $\alpha_3=\lambda_{3\bar{\alpha}}$

tout $x\varepsilon H(x_1,\bar{x}_2,\bar{x}_3)$ s'écrit :

$$x=\alpha_1x_1+\alpha_2x_2+\alpha_3x_3 \text{ avec } \alpha_1\varepsilon[1-\bar{\alpha},1],\alpha_2\varepsilon[0,\bar{\alpha}] \text{ et } \alpha_3\varepsilon[0,\bar{\alpha}]$$

et $\sum_{i=1}^{3} \alpha_i=1$

En posant $a_i'=\text{Max}\{a_i,1-\sum_{h\neq i}^{K} b_h\}$ et $b_i'=1$, on obtient les 3 intervalles suivants :

$$[1-\bar{\alpha},1] \quad , \quad [o,1] \quad \text{et} \quad [0,1]$$

b) Contraction autour d'un point central

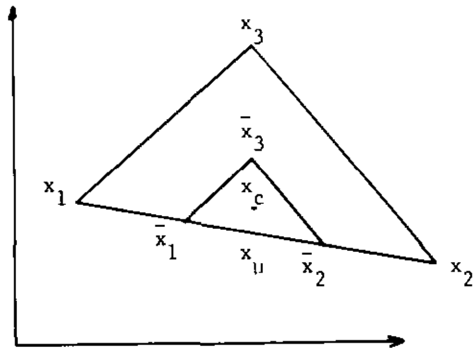


Figure 2

On contracte l'enveloppe $H(x_1,x_2,x_3)$ autour du point

$$x_c = \frac{K+1}{K^2}\, x_1 + \frac{K+1}{K^2}\, x_2 + \frac{1}{K^2}\, x_3 \quad \text{avec } K=3$$

Soient les points $x_\mu = \frac{1}{2}x_1 + \frac{1}{2}x_2 = \frac{1}{K-1}x_1 + \frac{1}{K-1}x_2$, $\bar{x}_1 = \bar{\alpha}x_1 + (1-\bar{\alpha})x_\mu$,

$\bar{x}_2 = \bar{\alpha}x_2 + (1-\bar{\alpha})x_\mu$ et $\bar{x}_3 = \bar{\alpha}x_3 + (1-\bar{\alpha})x_\mu$

Tout point $x \varepsilon H(\bar{x}_1, \bar{x}_2, \bar{x}_3)$ s'écrit :

$$x = \lambda_1\bar{x}_1 + \lambda_2\bar{x}_2 + \lambda_3\bar{x}_3 \text{ avec } \lambda_1, \lambda_2, \lambda_3 \varepsilon [0,1] \text{ et } \sum_{i=1}^{3} \lambda_i = 1$$

ou encore $x = \lambda_1(\bar{\alpha}x_1 + (1-\bar{\alpha})x_\mu) + \lambda_2(\bar{\alpha}x_2 + (1-\bar{\alpha})x_\mu) + \lambda_3(\bar{\alpha}\bar{x}_3 + (1-\bar{\alpha})x_\mu)$

or $\qquad x_\mu = \frac{1}{K-1}x_1 + \frac{1}{K-1}x_2$

d'où $x = \lambda_1(\bar\alpha x_1+(1-\bar\alpha)(\frac{1}{K-1}x_1+\frac{1}{K-1}x_2)) + \lambda_2(\bar\alpha x_2+(1-\bar\alpha)(\frac{1}{K-1}x_1+\frac{1}{K-1}x_2))+$

$\lambda_3(\bar\alpha x_3+(1-\bar\alpha)(\frac{1}{K-1}x_1+\frac{1}{K-1}x_2))$

$= (\lambda_1(\bar\alpha+\frac{1-\bar\alpha}{K-1})+\lambda_2\frac{1-\bar\alpha}{K-1}+\lambda_3\frac{1-\bar\alpha}{K-1})x_1+(\lambda_1\frac{1-\bar\alpha}{K-1}+\lambda_2(\bar\alpha+\frac{1-\bar\alpha}{K-1})+$

$\lambda_3\frac{1-\bar\alpha}{K-1})x_2+\lambda_3\bar\alpha x_3$

en posant $\alpha_1 = \lambda_1(\bar\alpha+\frac{1-\bar\alpha}{K-1})+\lambda_2\frac{1-\bar\alpha}{K-1}+\lambda_3\frac{1-\bar\alpha}{K-1}$, $\alpha_2 = \lambda_1\frac{1-\bar\alpha}{K-1}+\lambda_2(\bar\alpha+\frac{1-\bar\alpha}{K-1})+$

$\lambda_3\frac{1-\bar\alpha}{K-1}$ et $\alpha_3 = \lambda_3\bar\alpha$

x s'écrit :

$$x = \alpha_1x_1 + \alpha_2x_2 + \alpha_3x_3$$

avec $\alpha_1\varepsilon[\frac{1-\bar\alpha}{K-1}, \bar\alpha+\frac{1-\bar\alpha}{K-1}]$, $\alpha_2\varepsilon[\frac{1-\bar\alpha}{K-1}, \bar\alpha+\frac{1-\bar\alpha}{K-1}]$, $\alpha_3\varepsilon[0,\bar\alpha]$

et $\sum_{i=1}^{3}\alpha_i=1$

De même si on pose $a_i'=\text{Max}\{a_i, 1-\sum_{h\neq i}^{K} b_h\}$ et $b_i'=1$ on a les 3 intervalles :

$[\frac{1-\bar\alpha}{K-1},1]$, $[\frac{1-\bar\alpha}{K-1},1]$ et $[0,1]$

c) Contraction autour du point $x_c = \frac{1}{K}x_1 + \frac{1}{K}x_2 + \frac{1}{K}x_3$

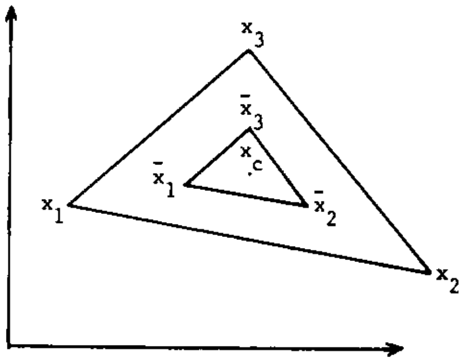


Figure 3

Soient $x_c = \frac{1}{K}x_1 + \frac{1}{K}x_2 + \frac{1}{K}x_3$, $\bar{x}_1 = \bar{\alpha}x_1 + (1-\bar{\alpha})x_c$, $\bar{x}_2 = \bar{\alpha}x_2 + (1-\bar{\alpha})x_c$, $\bar{x}_3 = \alpha\bar{x}_3 + (1-\bar{\alpha})x_c$

Tout $x \varepsilon H(\bar{x}_1, \bar{x}_2, \bar{x}_3)$ s'écrit :

$$x = \lambda_1\bar{x}_1 + \lambda_2\bar{x}_2 + \lambda_3\bar{x}_3 \text{ , avec } \lambda_1, \lambda_2, \lambda_3 \varepsilon [0,1] \text{ et } \sum_{i=1}^{3} \lambda_i = 1$$

ou encore $x = \lambda_1(\bar{\alpha}x_1 + (1-\bar{\alpha})(\frac{1}{K}x_1 + \frac{1}{K}x_2 + \frac{1}{K}x_3)) + \lambda_2(\bar{\alpha}x_2 + (1-\bar{\alpha})(\frac{1}{K}x_1 + \frac{1}{K}x_2 + \frac{1}{K}x_3)) +$

$$\lambda_3(\bar{\alpha}x_3 + (1-\bar{\alpha})(\frac{1}{K}x_1 + \frac{1}{K}x_2 + \frac{1}{K}x_3))$$

$$x = (\lambda_1(\bar{\alpha} + \frac{1-\bar{\alpha}}{K}) + \lambda_2 \frac{1-\bar{\alpha}}{K} + \lambda_3 \frac{1-\bar{\alpha}}{K})x_1 + (\lambda_1 \frac{1-\bar{\alpha}}{K} + \lambda_2 (\bar{\alpha} + \frac{1-\bar{\alpha}}{K}) +$$

$$\lambda_3 \frac{1-\bar{\alpha}}{K}) x_2 + (\lambda_1 \frac{1-\bar{\alpha}}{K} + \lambda_2 \frac{1-\bar{\alpha}}{K} + \lambda_3(\bar{\alpha} + \frac{1-\bar{\alpha}}{K}))x_3$$

Si on pose

$$\alpha_1 = \lambda_1(\bar{\alpha} + \frac{1-\bar{\alpha}}{K}) + \lambda_2 \frac{1-\bar{\alpha}}{K} + \lambda_3 \frac{1-\bar{\alpha}}{K}$$

$$\alpha_2 = \lambda_1 \frac{1-\bar{\alpha}}{K} + \lambda_2(\bar{\alpha} + \frac{1-\bar{\alpha}}{K}) + \lambda_3 \frac{1-\bar{\alpha}}{K}$$

$$\alpha_3 = \lambda_1 \frac{1-\bar{\alpha}}{K} + \lambda_2 \frac{1-\bar{\alpha}}{K} + \lambda_3(\bar{\alpha} + \frac{1-\bar{\alpha}}{K})$$

alors x s'écrit

$$x = \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3$$

avec $\alpha_1 \varepsilon [\frac{1-\bar{\alpha}}{K} , \bar{\alpha}+\frac{1-\bar{\alpha}}{K}]$, $\alpha_2 \varepsilon [\frac{1-\bar{\alpha}}{K} , \bar{\alpha}+\frac{1-\bar{\alpha}}{K}]$, $\alpha_3 \varepsilon [\frac{1-\bar{\alpha}}{K} , \bar{\alpha}+\frac{1-\bar{\alpha}}{K}]$,

$$\sum_{i=1}^{3} \alpha_i = 1$$

D'où les 3 intervalles :

$$[\frac{1-\bar{\alpha}}{K},1] \quad , \quad [\frac{1-\bar{\alpha}}{K},1] \quad , \quad [\frac{1-\bar{\alpha}}{K},1]$$

4.6 La méthode STEM (Step Method)

Préliminaires :

Cette méthode consiste à générer à chaque itération la solution de compromis relative à la distance L_∞ et à la présenter au décideur qui, à l'aide d'un tableau de gains, appelé aussi matrice des paiements, peut juger de la satisfaction obtenue pour chaque critère. Tant que le décideur n'est pas satisfait des valeurs obtenues pour un ou plusieurs critères, on lui demande de désigner des critères pour lesquels il accepte une moins-value. On calcule alors une nouvelle solution sensée représenter un meilleur compromis.

Construction du tableau de gains

Pour tout critère $i, i=1,\dots,K$ on résoud le programme linéaire

$$(\mathrm{Lpi}) \qquad \underset{x \in P}{\mathrm{Max}} \quad f_i(x)$$

soit x_i^* la solution optimale de (Lpi) , nous noterons z_{ji} , $j=1,\dots,K$ les valeurs des K fonctions objectifs au point x_i^* . Remarquons que z_{ii} est la valeur optimale f_i^* du programme (Lpi). On obtient le tableau de gains suivant :

	f_1	$f_2 \dots$	$f_j \dots$	f_K
x_1^*	f_1^*	z_{21}	z_{j1}	z_{x1}
x_2^*	z_{12}	f_2^*	z_{j2}	z_{K2}
$\vdots$				
x_j^*	z_{1j}	z_{2j}	f_j^*	z_{Kj}
$\vdots$				
x_K^*	z_{1K}	z_{2K}	z_{jK}	f_K^*

Principe de la génération d'une solution de compromis :

Les auteurs, R. Benayoun et J. De Montgolfier, proposent la solution de compromis relative à la distance L_∞ , c'est-à-dire la solution optimale du programme $\underset{x \varepsilon P}{\text{Min}}\ \underset{i}{\text{Max}}\ \{|f_i^* - f_i(x)|\}$, $i=1,\ldots,K$.

Or, nous avons vu (proposition 3 du paragraphe 3.3) que cette solution n'est pas nécessairement une solution efficace du programme multiobjectif $\underset{x \varepsilon P}{\text{Max}}\ (f_1(x),\ldots,f_K(x))$.

Pour que la solution de compromis générée soit une solution efficace du programme multiobjectif, nous proposons une solution efficace du programme bicritère $\underset{x \varepsilon P}{\text{Min}}\ (L_1\ ,\ L_\infty)$, c'est-à-dire la solution du programme

$$\underset{x \varepsilon P}{\text{Min}}\ \varepsilon\, L_1 + (1 - \varepsilon)\, L_\infty$$

avec ε positif, suffisamment petit.

En utilisant des poids π_i , $\pi_i > 0$, $i=1,\ldots,K$ et $\sum_{i=1}^{K} \pi_i = 1$, pour normaliser les fonctions objectifs, nous avons le programme

$$(Lp') \qquad \underset{x \varepsilon P}{\text{Min}}\ \varepsilon \sum_i \pi_i |f_i^* - f_i(x)| + (1-\varepsilon)\, \underset{i}{\text{Max}} |f_i^* - f_i(x)|$$

programme qui est équivalent au programme linéaire suivant :

$$(Lp'') \qquad \text{Min} - \varepsilon \sum_{i=1}^{K} \pi_i f_i(x) + (1-\varepsilon)\, d$$

$$f_i(x) + \frac{1}{\pi_i}\, d \geq f_i^* \qquad i=1,\ldots,K$$

$$x \varepsilon P$$

où $\pi_i = \frac{\alpha_i}{\sum_{j=1}^{K} \alpha_j}$, avec $\alpha_i = \begin{cases} \frac{f_i^* - \bar{f}_i}{f_i^*} \left(\frac{1}{(\sum_{j=1}^{n} (c_j^i)^2)^{\frac{1}{2}}} \right) & \text{si } f_i^* > 0 \\ \frac{\bar{f}_i - f_i^*}{\bar{f}_i} \left(\frac{1}{(\sum_{j=1}^{n} (c_j^i)^2)^{\frac{1}{2}}} \right) & \text{si } f_i^* \leqslant 0 \end{cases}$

On a noté par f_i^* la valeur maximum de la iéme colonne du tableau de gains et $\bar{f}_i$ la valeur minimum de cette même colonne.

Description d'une itération de la recherche d'une solution de compromis

Soit x_m la solution optimale du programme (Lp''). On présente au décideur le vecteur des valeurs $(f_1(x_m),\dots,f_x(x_m))$. S'il le trouve satisfaisant, on arrête, sinon on lui demande quels sont les critères dont la valeur peut être diminuée et de combien au maximum. Soient f_j un critère à relâcher et Δf_j la valeur maximale du relâchement. Alors on pose α_j=0 ; on calcule les π_i , $i \neq j$, $i=1,\dots,K$ et la nouvelle solution qui est solution optimale du programme :

$$\text{Min} \quad -\varepsilon \sum_{i \neq j}^{K} \pi_i f_i(x) \qquad +(1-\varepsilon)d$$

$$x \varepsilon P$$

$$f_i(x) + \frac{1}{\pi_i} d \geqslant f_i^* \qquad i \neq j \ , \ i=1,\dots,K$$

$$f_i(x) \geqslant f_i(x_m) \qquad i \neq j \ , \ i=1,\dots,K$$

$$f_j(x) \geqslant f_j(x_m) - \Delta f_j$$

5. PACKAGE ET EXEMPLES ILLUSTRATIFS

Les algorithmes que nous avons décrits dans le chapitre 4 ont été implémentés sur ordinateur (VAX11/780, code FORTRAN 77) sous forme de package.

Nous allons décrire dans ce chapitre l'organisation de ce package et donner pour chaque algorithme (module) un ou deux exemples illustratifs avec les entrées et sorties des programmes.

5.1 L'organisation du package

5.2 Exemples illustratifs

- 5.2.1 ELECTRE I
- 5.2.2 ELECTRE II
- 5.2.3 Méthode de Geoffrion
- 5.2.4 Méthode de Zionts
- 5.2.5 Algorithme interactif du simplexe
- 5.2.6 Méthode de STEUER
- 5.2.7 La méthode STEM

5.1 L'organisation du Package

Le package est composé de deux grands modules, l'un traitant des applications de type discret, que nous avons appelé ANAMULT (pour analyse multicritère) et l'autre traitant des applications de type continu et que nous avons appelé PMO (pour programmation multiobjectif). ANAMULT et PMO permettent ensuite à l'utilisateur d'exécuter le module de son choix, c'est-à-dire pour ANAMULT le module E_1 (pour ELECTRE I) ou le module E_2 (pour ELECTRE II), et pour PMO les modules G (pour Geoffrion) ou Z (pour Zionts) ou M (pour Simplexe interactif) ou T (pour STEM) ou S (pour STEUER).

La procédure OPTMULT (optimisation multicritère), écrite en langage de commande (DCL), opère de la façon suivante :

Phase 0 : L'utilisateur est interrogé sur le type de l'application, sur le nom et l'existence du fichier contenant les données de l'application.

Phase 1 : Si le fichier de données n'existe pas on exécute le programme ENTDDNCNT si l'application est de type continu ou le programme ENTDONDIS si l'application est de type discret.

Ces deux programmes permettent à l'utilisateur d'entrer interactivement, sur terminal, les données du problème. Un fichier de données donnin.CNT (dans le cas continu) ou donnin.DIS (dans le cas discret) est créé où donnin est le nom donné par l'utilisateur. Nous donnerons par la suite les structures de ces fichiers.

Si le fichier de données préexistait à la phase 1, on passe directement à la phase 2.

Phase 2 : Suivant que l'application est de type discret ou continu, on exécute le programme ANAMULT ou le programme PMO. L'utilisateur a la possibilité de faire plusieurs traitements (enchaînement de modules) sur le même fichier de données au niveau de ANAMULT ou PMO.

S'il désire faire le même traitement ou un autre traitement sur un jeu de données différent, on revient à la phase 0.

Le schéma SC1 donne une vue globale de la structure du package.

Description des fichiers de données

A) Le fichier donnin.CNT

enregistrement 1 : 3 champs de 5 colonnes

champs 1 : Col. 1- 5 nombre de fonctions objectifs NOBJ

champs 2 : Col. 6-10 nombre de contraintes NCONT

champs 3 : Col. 11-15 nombre de variables NVAR

enregistrement 2 : titre : fonctions objectifs

enregistrement 3 : Col. 1 indicateur de non linéarité (0 si toutes les fonctions objectifs sont linéaires 1 sinon)

enregistrements suivants (NOBJ) :

Si la fonction objectif est linéaire, l'enregistrement est divisé en NVAR champs de 9 colonnes chacun (F9.3) représentant les coefficients de la fonction objectif; si la fonction objectif est non linéaire, l'enregistrement contient la forme

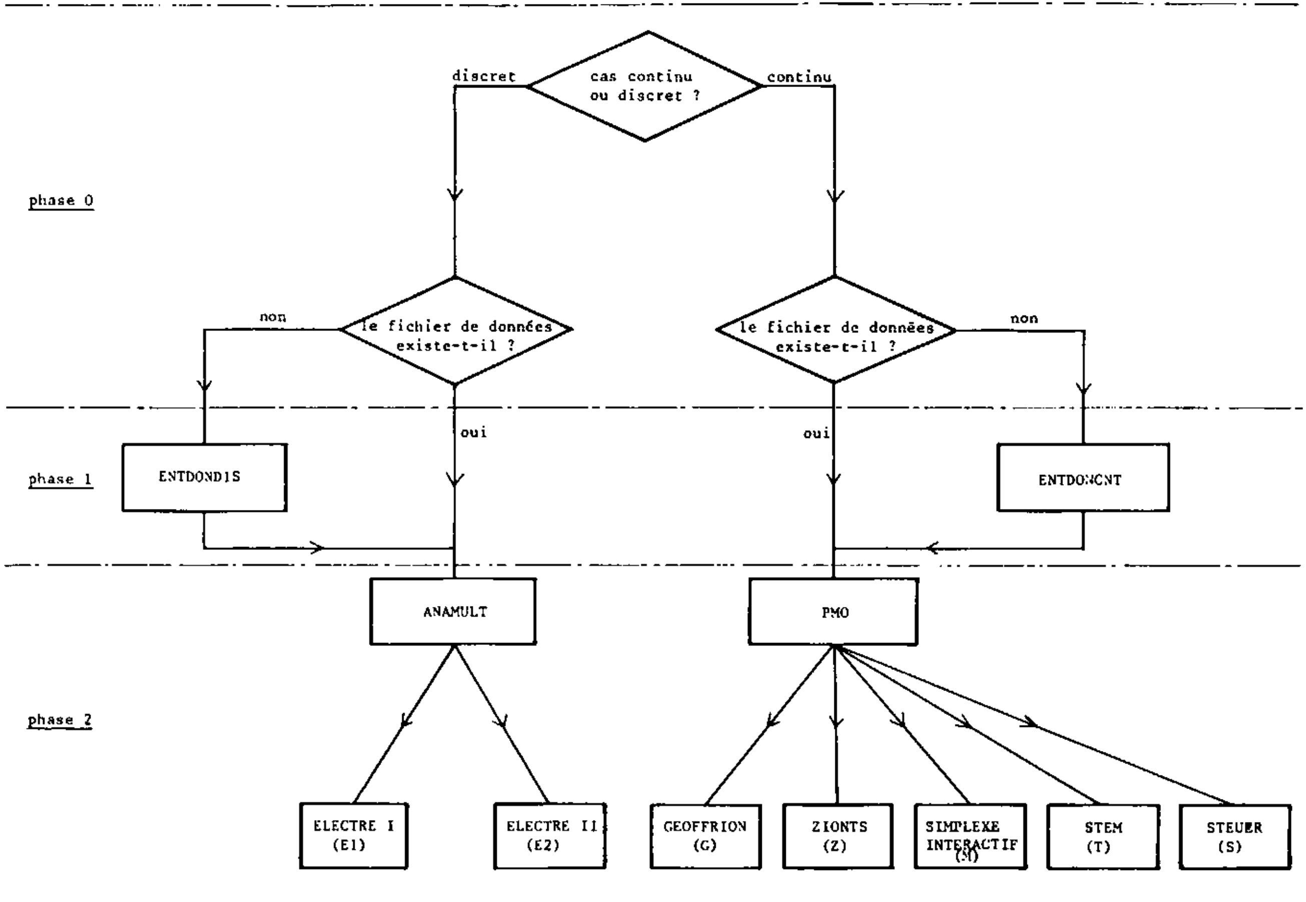

phase 0
phase 1
phase 2
cas continu ou discret ?
discret
continu
le fichier de données existe-t-il ?
non
oui
le fichier de données existe-t-il ?
oui
non
ENTDONDIS
ENTDONCNT
ANAMULT
PMO
ELECTRE I (E1)
ELECTRE II (E2)
GEOFFRION (G)
ZIONTS (Z)
SIMPLEXE INTERACTIF (M)
STEM (T)
STEUER (S)

Schéma SC1

analytique de la fonction objectif (au maximum 132 caract.).

enregistrement suivant : titre : contraintes

enregistrements suivants (NCONT) :

Chaque enregistrement est divisé en NVAR+2 champs. Les premiers NVAR champs sont de 9 colonnes chacun (F9.3) et représentent les coefficients d'une ligne de la matrice des contraintes. Le (NVAR+1)ième champ est de 3 colonnes (F3.0) et représente le type d'inégalité de la contrainte (1 si ≤ , 0 si = , et -1 si ≥). Le (NVAR+2)ième champ est de 9 colonnes (F9.3) et réprésente le membre de droite de la contrainte.

Voir exemples paragraphe 5.2.

B) Le fichier donnin.DI5

enregistrement 1 : 2 champs de 5 colonnes

champ 1 : Col. 1- 5 nombre de critères NCRIT

champ 2 : Col. 6-10 nombre d'actions NACT

enregistrement 2 : NCRIT champs de 4 colonnes (A4) contenant les noms de critères

enregistrement suivant (NACT) : L'enregistrement est divisé en NCRIT+1 champs; le premier de 4 colonnes A(4) contient le nom de l'action et les NCRIT autres sont de 9 colonnes chacun (F9.3) et contiennent les valeurs de l'action sur les NCRIT critères.

enregistrement suivant : titre : poids des critères

enregistrement suivant : NCRIT champs, chacun de 3 colonnes (F3.1)
contenant le poids d'un critère

enregistrement suivant : titre : amplitude des échelles

enregistrement suivant : NCRIT champs, chacun de 4 colonnes (F4.0)
et contenant l'amplitude de l'échelle d'un critère

Voir exemples paragraphe 5.2.

5.2 Exemples illustratifs

Pour les exemples de type continu, nous signalons que la maximisation est l'unique option d'optimisation acceptée par le package. Ceci n'empêche pas l'utilisateur de minimiser des fonctions objectifs s'il le désire. Car du point de vue de la solution optimale Min f(x) est équivalent à Max -f(x), et donc au moment de l'introduction des données, il suffit de considérer la fonction -f(x).

5.2.1 ELECTRE I

Exemple : choix d'une nouvelle voiture

L'exemple qui suit est une adaptation de l'exemple donné par B. Roy et Ph. Vincke dans l'analyse multicritère (cahier de Lamsade, 1980).

Le décideur est un père de famille confronté à l'achat d'une nouvelle voiture et les actions sont sept voitures proposées sur le marché. Pour les juger, le décideur retient quatre critères qui sont : le prix, le confort, la vitesse et la ligne.

Les échelles de ces quatre critères sont :

critères	Niveau	code
prix :	inférieur ou égal à 10.000.-	1
	entre 10.000.- et 11.500.-	2
	entre 11.500.- et 13.500.-	3
	entre 13.500.- et 15.000.-	4
	entre 15.000.- et 16.500.-	5
confort :	Excellent	E
	Moyen	M
	Faible	F
Vitesse :	Rapide	R
	Moyenne	M
Ligne :	Soignée	S
	Ordinaire	O

L'évaluation des sept voitures sur les critères est donnée par le tableau suivant :

	V_1	V_2	V_3	V_4	V_5	V_6	V_7
prix	5	4	4	3	3	3	1
confort	E	E	M	M	M	F	F
vitesse	R	M	R	R	M	R	M
ligne	S	S	S	O	S	S	O

Les poids suivants ont été attribués aux critères :

	prix	confort	vitesse	ligne
poids :	5	3	1	1

Pour obtenir un tableau de valeurs directement utilisable par le programme, nous avons noté les critères sur une échelle ordinale de 100 points. Le premier niveau du prix est le meilleur et nous lui avons attribué la note 100, au niveau 2 la note 80, au niveau 3 la note 60, au niveau 4 la note 40 et enfin au niveau 5 la note 20. Pour le confort, le niveau E aura la note 100 et les niveaux M et F les notes 80 et 60. En ce qui concerne les critères vitesse et ligne, nous avons attribué la note 100 pour les premiers niveaux, c'est-à-dire, R pour la vitesse et S pour la ligne, et la note 80 pour les deuxièmes niveaux, c'est-à-dire, M pour la vitesse et O pour la ligne. Le tableau de valeurs qui sera utilisé est :

	V_1	V_2	V_3	V_4	V_5	V_6	V_7
prix	20	40	40	60	60	60	100
confort	100	100	80	80	80	60	60
vitesse	100	80	100	100	80	100	80
ligne	100	100	100	80	100	100	80

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : VOITURE
S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : DISC
DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : VOIT
LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : N
```

Phase 1

Entrée des données

```
DONNEZ LE NOMBRE DE CRITERES : 4
DONNEZ LE NOMBRE D'ACTIONS : 7
DONNEZ LE NOM DU CRITERE No 1 ( MAX. 4 CARACT.) : PRIX
DONNEZ LE NOM DU CRITERE No 2 ( MAX. 4 CARACT.) : CONF
DONNEZ LE NOM DU CRITERE No 3 ( MAX. 4 CARACT.) : VITE
DONNEZ LE NOM DU CRITERE No 4 ( MAX. 4 CARACT.) : LIGN
DONNEZ LE NOM DE L'ACTION No 1 ( MAX. 4 CARACT.) : V1
DONNEZ LES VALEURS DE CETTE ACTION SUR LES 4 CRITERES : 20 100 100 100
DONNEZ LE NOM DE L'ACTION No 2 ( MAX. 4 CARACT.) : V2
DONNEZ LES VALEURS DE CETTE ACTION SUR LES 4 CRITERES : 40 100 80 100
DONNEZ LE NOM DE L'ACTION No 3 ( MAX. 4 CARACT.) : V3
DONNEZ LES VALEURS DE CETTE ACTION SUR LES 4 CRITERES : 40 80 100 100
DONNEZ LE NOM DE L'ACTION No 4 ( MAX. 4 CARACT.) : V4
DONNEZ LES VALEURS DE CETTE ACTION SUR LES 4 CRITERES : 60 80 100 80
DONNEZ LE NOM DE L'ACTION No 5 ( MAX. 4 CARACT.) : V5
DONNEZ LES VALEURS DE CETTE ACTION SUR LES 4 CRITERES : 60 80 80 100
DONNEZ LE NOM DE L'ACTION No 6 ( MAX. 4 CARACT.) : V6
DONNEZ LES VALEURS DE CETTE ACTION SUR LES 4 CRITERES : 60 60 100 100
DONNEZ LE NOM DE L'ACTION No 7 ( MAX. 4 CARACT.) : V7
DONNEZ LES VALEURS DE CETTE ACTION SUR LES 4 CRITERES : 100 60 80 80
DONNEZ LES POIDS DES CRITERES : 5 3 1 1
DONNEZ LES AMPLITUDES DES ECHELLES DES CRITERES : 100 100 100 100
```

Fichier de données VOIT.OIS

```
    4     7   (* NOMBRE DE CRITERES,NOMBRE D'ACTIONS *)
PRIXCONFVITELIGN
V1    20.0       100.       100.       100.
V2    40.0       100.       80.0       100.
V3    40.0       80.0       100.       100.
V4    60.0       80.0       100.       80.0
V5    60.0       80.0       80.0       100.
V6    60.0       60.0       100.       100.
V7    100.       60.0       80.0       80.0
POIDS DES CRITERES :
 5.0 3.0 1.0 1.0
AMPLITUDES DES ECHELLES :
100.100.100.100.
```

Phase 2

Traitement

```
CHOIX DU MODULE (E1 SI ELECTRE1, E2 SI ELECTRE2): E1
DONNEZ LE NOM DU FICHIER DES SEUILS (MAX. 9 CARACT.) : VOITSL
LE FICHIER DES SEUILS VOITSL.SCD EXISTE T-IL DEJA ? (O/N) : N
DONNEZ LE SEUIL DE CONCORDANCE (0 < P < 1.0) : 0.75
DONNEZ LE SEUIL DE DISCORDANCE (0 < Q < 1.0) : 0.20
VOULEZ VOUS UN AUTRE JEU DE SEUILS ? (O/N) : O
DONNEZ LE SEUIL DE CONCORDANCE (0 < P < 1.0) : 0.85
DONNEZ LE SEUIL DE DISCORDARCE (0 < Q < 1.0) : 0.20
VOULEZ VOUS UN AUTRE JEU DE SEUILS ? (O/N) : N
VOULEZ VOUS CHANGER LA PONDERATION DES CRITERES (O/N): N
VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N
```

Un enregistrement du fichier des seuils nomfich.SCD, dans l'exemple traité nomfich vaut VOITSL, est divisé en deux champs de 5 colonnes (F5.3) chacun. Le premier champ contient le seuil de concordance P et le deuxième le seuil de discordance Q. Il a autant d'enregistrements que le décideur a de jeux de seuils différents. Au cas où ce fichier n'existe pas, ou lorsque le décideur désire introduire un nouveau jeu, les seuils P et Q sont lus de façon interactive et sont écrits au fur et à mesure dans nomfich.SCD.

Lorsqu'on change la pondération des critères, tous les jeux de seuils se trouvant dans nomfich.SCD sont pris en considération.

Résultats :

Les résultats se trouvent dans le fichier nom.RES, où nom est la réponse du décideur à la question NOM DU PASSAGE de la phase 0. Ici nom vaut VOITURE.

VOITURE.RES

METHODE ELECTRE1

MATRICE DES VALEURS DES ACTIONS SUR LES CRITERES :

	PRIX	CONF	VITE	LIGN
V1	20.000	100.000	100.000	100.000
V2	40.000	100.000	80.000	100.000
V3	40.000	80.000	100.000	100.000
V4	60.000	80.000	100.000	80.000
V5	60.000	80.000	80.000	100.000
V6	60.000	60.000	100.000	100.000
V7	100.000	60.000	80.000	80.000

PONDERATION ET AMPLITUDES DES CRITERES :

	PRIX	CONF	VITE	LIGN
POIDS :	5.0	3.0	1.0	1.0
AMPL. :	100.	100.	100.	100.

MATRICE DE DISCORDANCE :

	V1	V2	V3	V4	V5	V6	V7
V1	0.000	0.200	0.200	0.400	0.400	0.400	0.800
V2	0.200	0.000	0.200	0.200	0.200	0.200	0.600
V3	0.200	0.200	0.000	0.200	0.200	0.200	0.600
V4	0.200	0.200	0.200	0.000	0.200	0.200	0.400
V5	0.200	0.200	0.200	0.200	0.000	0.200	0.400
V6	0.400	0.400	0.200	0.200	0.200	0.000	0.400
V7	0.400	0.400	0.200	0.200	0.200	0.200	0.000

MATRICE DE CONCORDANCE :

	V1	V2	V3	V4	V5	V6	V7
V1	1.000	0.500	0.500	0.500	0.500	0.500	0.500
V2	0.900	1.000	0.900	0.400	0.500	0.400	0.500
V3	0.700	0.700	1.000	0.500	0.500	0.500	0.500
V4	0.600	0.600	0.900	1.000	0.900	0.900	0.500
V5	0.600	0.700	0.900	0.900	1.000	0.900	0.500
V6	0.700	0.700	0.700	0.700	0.700	1.000	0.500
V7	0.500	0.600	0.500	0.600	0.600	0.800	1.000

POUR UN SEUIL DE CONCORDANCE P = 0.75
ET UN SEUIL DE DISCORDANCE Q = 0.20

VOITURE.RES (suite)

```
LE SURCLASSEMENT EST :

V2   SURCLASSE  V1    V3
V4   SURCLASSE  V3    V5    V6
V5   SURCLASSE  V3    V4    V6
V7   SURCLASSE  V6
LES ACTIONS  V4    V5   FORMENT UN CIRCUIT

LES QUASI NOYAUX SONT :

V2    V4    V7
V2    V5    V7

POUR UN SEUIL DE CONCORDANCE P = 0.85
ET UN SEUIL DE DISCORDANCE Q = 0.20

LE SURCLASSEMENT EST :

V2   SURCLASSE  V1    V3
V4   SURCLASSE  V3    V5    V6
V5   SURCLASSE  V3    V4    V6
V7   EST UN SOMMET ISOLE DU GRAPHE DE SURCLASSEMENT
LES ACTIONS  V4    V5   FORMENT UN CIRCUIT

LES QUASI NOYAUX SONT :

V2    V4    V7
V2    V5    V7
```

5.2.2 ELECTRE II

Exemple : évaluation des performances des divisions d'une compagnie (voir [5a], page 204)

Pour établir une politique du personnel cadre (ajustement des salaires, promotion, etc ...), une compagnie désire évaluer les performances de ses cinq divisions selon cinq importants critères : la ROI (Return on investment), le taux d'accroissement de la part du marché (TAM), le taux de croissance des ventes (TCV), le taux d'employés non satisfaits (TPNS) et le taux de rotation du personnel (TRP). Les données sont résumées dans le tableau suivant :

	DIV1	DIV2	DIV3	DIV4	DIV5
ROI	0.21	0.20	0.30	0.15	0.18
TAM	0.10	0.12	0.07	0.20	0.11
TCV	0.10	0.08	0.20	0.12	0.25
TPNS	-0.07	-0.10	-0.02	-0.20	-0.05
TRP	-0.10	-0.08	-0.09	-0.12	-0.12

Les deux derniers critères, c'est-à-dire TPNS et TRP, sont à minimiser et, pour cette raison, ils ont été notés négativement.

Pour traiter cet exemple, nous avons attribué aux critères les poids suivants :

	ROI	TAM	TCV	TPNS	TRP
poids :	3	2	2	1.5	1.5

et nous avons choisi pour les seuils de concordance $c_1 > c_2 > c_3$ les valeurs 0.75, 0.65 et 0.60 et pour les seuils de discordance $d_{1i} < d_{2i}$, i=1,5 , les valeurs suivantes :

	i	d_{1i}	d_{2i}
ROI	1	0.02	0.1
TAM	2	0.02	0.09
TCV	3	0.02	0.1
TPNS	4	0.03	0.11
TRP	5	0.02	0.04

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : EVAL
S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : DISC
DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : PERFO
LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : O
```

L'entrée des données est la même que pour la méthode ELECTRE 1 et, pour cette raison, nous avons considéré le cas où le fichier des données PERFO.OIS existe déjà. Nous donnons donc le fichier de données PERFO.OIS et nous passons directement à la phase 2.

Fichier de données PERFO.OIS

```
    5    5   (* NOMBRE DE CRITERES,NOMBRE D'ACTIONS *)
ROI TAM TCV TPNSTRP
DIV10.210    0.100     0.100     -.700E-01-.100
DIV20.200    0.120     0.800E-01-.100     -.800E-01
DIV30.300    0.700E-010.200     -.200E-01-.900E-01
DIV40.150    0.200     0.120     -.200     -.120
DIV50.180    0.110     0.250     -.500E-01-.120
POIDS DES CRITERES :
 3.0 2.0 2.0 1.5 1.5
AMPLITUDES DES ECHELLES :
  1.  1.  1.  1.  1.
```

Phase 2

Traitement

```
CHOIX DU MODULE (E1 SI ELECTRE1, E2 SI ELECTRE2): E2

DONNEZ LE NOM DU FICHIER DES SEUILS (MAX. 9 CARACT.) : PERFOSL

LE FICHIER DES SEUILS PERFOSL.SCD EXISTE T-IL DEJA ? (O/N) : N

DONNEZ LES 3 SEUILS DE CONCORDANCE DANS L'ORDRE SUIVANT :
D'ABORD LE PLUS GRAND ENSUITE LE MOYEN ENSUITE LE PLUS PETIT : 0.75 0.65 0.60

DONNEZ LES 2 SEUILS DE DISCORDANCE POUR LE CRITERE 1 DANS L'ORDRE SUIVANT :
D'ABORD LE PLUS PETIT ENSUITE LE PLUS GRAND : 0.02 0.1

DONNEZ LES 2 SEUILS DE DISCORDANCE POUR LE CRITERE 2 DANS L'ORDRE SUIVANT :
D'ABORD LE PLUS PETIT ENSUITE LE PLUS GRAND : 0.02 0.09

DONNEZ LES 2 SEUILS DE DISCORDANCE POUR LE CRITERE 3 DANS L'ORDRE SUIVANT :
D'ABORD LE PLUS PETIT ENSUITE LE PLUS GRAND : 0.02 0.1

DONNEZ LES 2 SEUILS DE DISCORDANCE POUR LE CRITERE 4 DANS L'ORDRE SUIVANT :
D'ABORD LE PLUS PETIT ENSUITE LE PLUS GRAND : 0.03 0.11

DONNEZ LES 2 SEUILS DE DISCORDANCE POUR LE CRITERE 5 DANS L'ORDRE SUIVANT :
D'ABORD LE PLUS PETIT ENSUITE LE PLUS GRAND : 0.02 0.04

VOULEZ VOUS UN AUTRE JEU DE SEUILS ? (O/N) : N

VOULEZ VOUS CHANGER LA PONDERATION DES CRITERES (O/N): N

VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N
```

Un jeu de seuils est donné sur trois enregistrements du fichier nomfich.SCD, dans l'exemple nomfich vaut PERFOSL. Le premier enregistrement est divisé en 3 champs de 5 colonnes (F5.3) chacun et contient les 3 seuils de concordance $c_1 > c_2 > c_3$. Les 2 autres sont divisés en NCRIT (nombre de critères) champs de 9 colonnes (F9.3) et contiennent, le premier les seuils de discordance D_{1i}, i=1, NCRIT et le second les seuils de discordance D_{2i}, i=1, NCRIT.

Comme pour ELECTRE I, si le fichier nomfich.SCD est absent, ou si le décideur désire un nouveau jeu, les seuils sont entrés à l'aide du terminal et écrits dans le fichier nomfich.SCD.

Résultats :

EVAL.RES

METHODE ELECTRE2

MATRICE DES VALEURS DES ACTIONS SUR LES CRITERES :

	ROI	TAM	TCV	TPMS	TRP
DIV1	0.210	0.100	0.100	-0.070	-0.100
DIV2	0.200	0.120	0.080	-0.100	-0.080
DIV3	0.300	0.070	0.200	-0.020	-0.090
DIV4	0.150	0.200	0.120	-0.200	-0.120
DIV5	0.180	0.110	0.250	-0.050	-0.120

PONDERATION ET AMPLITUDES DES CRITERES :

	ROI	TAM	TCV	TPMS	TRP
POIDS :	3.0	2.0	2.0	1.5	1.5
AMPL. :	1.	1.	1.	1.	1.

MATRICE DE CONCORDANCE :

	DIV1	DIV2	DIV3	DIV4	DIV5
DIV1	1.000	0.650	0.200	0.600	0.450
DIV2	0.350	1.000	0.350	0.600	0.650
DIV3	0.800	0.650	1.000	0.800	0.600
DIV4	0.400	0.400	0.200	1.000	0.350
DIV5	0.550	0.350	0.400	0.600	1.000

SEUILS DE CONCORDANCE :

C1 = 0.75   C2 = 0.65   C3 = 0.60

SEUILS DE DISCORDANCE :

CRITERES	D11	D12
ROI	0.02	0.10
TAM	0.02	0.09
TCV	0.02	0.10
TPMS	0.03	0.11
TRP	0.02	0.04

EVAL.RES (suite)

LE SURCLASSEMENT FORT EST :

DIV2 EST UN SOMMET ISOLE DU GRAPHE DE SURCLASSEMENT

DIV3 SURCLASSE DIV1

DIV5 SURCLASSE DIV4

LE SURCLASSEMENT FAIBLE EST :

DIV1 SURCLASSE DIV2

DIV2 SURCLASSE DIV4

DIV3 SURCLASSE DIV1 DIV2 DIV5

DIV5 SURCLASSE DIV4

CLASSEMENT DIRECT :

DIV1	DIV2	DIV3	DIV4	DIV5
2	3	1	4	2

CLASSEMENT INDIRECT :

DIV1	DIV2	DIV3	DIV4	DIV5
2	3	1	4	3

CLASSEMENT MEDIAN :

DIV1	DIV2	DIV3	DIV4	DIV5
2	4	1	5	3

5.2.3 Méthode de Geoffrion

Exemple 1 : problème de nutrition (voir [9], page 111)

Les informations concernant 6 aliments : lait, boeuf, oeufs, pain, salade-laitue, et jus d'orange, sont résumées dans le tableau T1 de la page suivante.

Il s'agit de trouver les quantités x_1, x_2, x_3, x_4, x_5 et x_6 respectivement de lait, de boeuf, d'oeufs, de pain, de salade-laitue et de jus d'orange tels que les besoins journaliers d'un adulte en vitamine A, en calories, en protéines et en fer soient satisfaits, et tels que :

(1) le coût total soit minimum

et (2) la quantité de cholestérol absorbée quotidiennement soit minimum

et (3) la quantité d'hydrate de carbone absorbée quotidiennement soit minimum.

Les rations journalières peuvent être au maximum de 6 pintes de lait, 1 pound de boeuf, 1/4 de douzaine d'oeufs, 10 onces de pain, 10 onces de salade-laitue et 4 pintes de jus d'orange. Ceci donne le programme multiobjectif suivant :

$$\text{Max } - 0.225x_1 - 2.2x_2 - 0.8x_3 - 0.1x_4 - 0.05x_5 - 0.26x_6$$

$$\text{Max } - 10x_1 - 20x_2 - 120x_3$$

$$\text{Max } - 24x_1 - 27x_2 - 15x_4 - 1.1x_5 - 52x_6$$

sous les contraintes :

	lait (pinte)	boeuf (pound)	oeufs (douzaine)	pain (once)	laitue-salade (once)	jus d'orange (pinte)	besoin journalier pour un adulte
vitamine A (I.U.)	720	107	7080	0	134	1000	5000
énergie (calories)	344	460	1040	75	17.4	240	2500
cholesterol (unité)	10	20	120	0	0	0	
protéine (g)	18	151	78	2.5	0.2	4	63
hydrate de carbone (g)	24	27	0	15	1.1	5.2	
fer (mg)	0.2	10.1	13.2	0.75	0.15	1.2	12.5
coût unitaire ($)	0.225	2.2	0.8	0.1	0.05	0.26	

Tableau T1

$$720x_1 + 107x_2 + 7080x_3 + 134x_5 + 1000x_6 \geq 5000$$

$$0.2x_1 + 10.1x_2 + 13.2x_3 + 0.75x_4 + 0.15x_5 + 1.2x_6 \geq 12.5$$

$$344x_1 + 460x_2 + 1040x_3 + 75x_4 + 17.4x_5 + 240x_6 \geq 2500$$

$$18x_1 + 151x_2 + 78x_3 + 2.5x_4 + 0.2x_5 + 4x_6 \geq 63$$

$x_1 \leq 6$ $\quad$ $x_4 \leq 10$

$x_2 \leq 1$ $\quad$ $x_5 \leq 10$

$x_3 \leq 0.25$ $\quad$ $x_6 \leq 4$

$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$$

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : EXEC1

S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : CONT

DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : NUTRI

LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : N
```

Phase 1

Entrée des données

```
NOMBRE DE FONCTIONS OBJECTIFS ?: 3

NOMBRE DE CONTRAINTES ?: 10

NOMBRE DE VARIABLES ?: 6

LE PROBLEME COMPORTE T'IL DES FONCTIONS  OBJECTIFS NON LINEAIRES (O/N) ? : N

DONNEZ LES COEFFICIENTS DE LA FONCTION OBJECTIF
NUMERO   1 ?: -0.225 -2.2 -0.5 -0.1 -0.05 -0.26

DONNEZ LES COEFFICIENTS DE LA FONCTION OBJECTIF
NUMERO   2 ?: -10 -20 -120 0 0 0

DONNEZ LES COEFFICIENTS DE LA FONCTION OBJECTIF
NUMERO   3 ?: -24 -27  0 -15 -1.1 -52
```

```
DONNEZ LES ELEMENTS DE LA LIGNE  1
DE LA MATRICE DES CONTRAINTES A ?: 720 107 7080 0 134 1000

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : -1

VALEUR DU MEMBRE DE DROITE : 5000

DONNEZ LES ELEMENTS DE LA LIGNE  2
DE LA MATRICE DES CONTRAINTES A ?: 0.2 10.1 13.2 0.75 0.15 1.2

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : -1

VALEUR DU MEMBRE DE DROITE : 12.5

DONNEZ LES ELEMENTS DE LA LIGNE  3
DE LA MATRICE DES CONTRAINTES A ?: 344 460 1040 75 17.4 240

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : -1

VALEUR DU MEMBRE DE DROITE : 2500

DONNEZ LES ELEMENTS DE LA LIGNE  4
DE LA MATRICE DES CONTRAINTES A ?: 18 151 78 2.5 0.2 4

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : -1

VALEUR DU MEMBRE DE DROITE : 63

DONNEZ LES ELEMENTS DE LA LIGNE  5
DE LA MATRICE DES CONTRAINTES A ?: 1 0 0 0 0 0

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 6

DONNEZ LES ELEMENTS DE LA LIGNE  6
DE LA MATRICE DES CONTRAINTES A ?: 0 1 0 0 0 0

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 1

DONNEZ LES ELEMENTS DE LA LIGNE  7
DE LA MATRICE DES CONTRAINTES A ?: 0 0 1 0 0 0

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 0.25

DONNEZ LES ELEMENTS DE LA LIGNE  8
DE LA MATRICE DES CONTRAINTES A ?: 0 0 0 1 0 0

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 10

DONNEZ LES ELEMENTS DE LA LIGNE  9
DE LA MATRICE DES CONTRAINTES A ?: 0 0 0 0 1 0

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 10

DONNEZ LES ELEMENTS DE LA LIGNE 10
DE LA MATRICE DES CONTRAINTES A ?: 0 0 0 0 0 1

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 4
```

Fichier de données NUTRI.CNT

```
   3   10    6  (* NB DE CRITERES,NB DE CONTRAINTES,NB DE VARIABLES *)
FONCTIONS OBJECTIFS :
0 (* 0 = TOUTES LES FONCT. OBJ. SONT LINEAIRES,1 = AU MOINS UNE FONCT. OBJ. EST NON LINEAIRE *)
-.225     -2.20     -.800     -.100     -.500E-01-.260
-10.0     -20.0     -120.     0.000E+000.000E+000.000E+00
-24.0     -27.0     0.000E+00-15.0    -1.10     -52.0
CONTRAINTES :
 720.      107.     0.708E+040.000E+00 134.     0.100E+04-1. 0.50E+04
0.200      10.1      13.2     0.750     0.150     1.20     -1.  12.5
 344.      460.     0.104E+04 75.0      17.4      240.     -1. 0.25E+04
 18.0      151.      78.0     2.50      0.200     4.00     -1.  63.
 1.00     0.000E+000.000E+000.000E+000.000E+000.000E+00 1.  6.0
0.000E+00 1.00      0.000E+000.000E+000.000E+000.000E+00 1.  1.0
0.000E+000.000E+00 1.00     0.000E+000.000E+000.000E+00 1. 0.25
0.000E+000.000E+000.000E+00 1.00    0.000E+000.000E+00 1.  10.
0.000E+000.000E+000.000E+000.000E+00 1.00     0.000E+00 1.  10.
0.000E+000.000E+000.000E+000.000E+000.000E+00 1.00      1.  4.0
```

Phase 2

Dialogue décideur-machine

CHOIX DU MODULE (M.Z.G.S.T OU ?) ? : G

VALEURS DES FONCTIONS AU POINT COURANT : -3.885 -74.120 -168.282

DONNEZ L'INDICE DE LA FONCTION OBJECTIF QUE
VOUS DESIREZ PRENDRE COMME CRITERE DE REFERENCE : 3

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 1 (PAR RAPPORT AU CRIT. DE REF. 3) : 0.013

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 2 (PAR RAPPORT AU CRIT. DE REF. 3) : 0.04

ITERATION No 1

	X1	X2	X3	X4	X5	X6	F1(X)	F2(X)	F3(X)
T = 0.00	5.450	0.981	0.000	0.000	10.000	0.000	-3.885	-74.120	-168.282
T = 0.10	5.064	0.883	0.000	1.000	10.000	0.400	-3.791	-68.498	-192.651
T = 0.20	4.718	0.785	0.000	2.000	10.000	0.800	-3.696	-62.877	-217.021
T = 0.30	4.352	0.687	0.000	3.000	10.000	1.200	-3.602	-57.256	-241.390
T = 0.40	3.986	0.589	0.000	4.000	10.000	1.600	-3.508	-51.635	-265.760
T = 0.50	3.620	0.491	0.000	5.000	10.000	2.000	-3.414	-46.013	-290.129
T = 0.60	3.254	0.392	0.000	6.000	10.000	2.400	-3.320	-40.392	-314.499
T = 0.70	2.888	0.294	0.000	7.000	10.000	2.800	-3.225	-34.771	-338.868
T = 0.80	2.522	0.196	0.000	8.000	10.000	3.200	-3.131	-29.149	-363.238
T = 0.90	2.157	0.098	0.000	9.000	10.000	3.600	-3.037	-23.528	-387.607
T = 1.00	1.791	0.000	0.000	10.000	10.000	4.000	-2.943	-17.907	-411.977

VOULEZ-VOUS LES VALEURS DES FONCTIONS POUR UNE AUTRE VALEUR DE T ? (O/N): N

DONNEZ LA VALEUR DE T QUI VOUS DONNE LA PLUS GRANDE SATISFACTION : 0.4

VALEURS DES FONCTIONS AU POINT COURANT : -3.508 -51.635 -265.760

DONNEZ L'INDICE DE LA FONCTION OBJECTIF QUE
VOUS DESIREZ PRENDRE COMME CRITERE DE REFERENCE : 2

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 1 (PAR RAPPORT AU CRIT. DE REF. 2) : 0.05

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 3 (PAR RAPPORT AU CRIT. DE REF. 2) : 0.5

ITERATION No 2

	X1	X2	X3	X4	X5	X6	F1(X)	F2(X)	F3(X)
T = 0.00	3.986	0.589	0.000	4.000	10.000	1.600	-3.508	-51.635	-265.760
T = 0.10	4.100	0.596	0.025	3.600	10.000	1.440	-3.488	-55.915	-254.361
T = 0.20	4.213	0.603	0.050	3.200	10.000	1.280	-3.468	-60.195	-242.962
T = 0.30	4.327	0.610	0.075	2.800	10.000	1.120	-3.448	-64.476	-231.564
T = 0.40	4.440	0.618	0.100	2.400	10.000	0.960	-3.427	-68.756	-220.165
T = 0.50	4.554	0.625	0.125	2.000	10.000	0.800	-3.407	-73.037	-208.767
T = 0.60	4.668	0.632	0.150	1.600	10.000	0.640	-3.387	-77.317	-197.368
T = 0.70	4.781	0.639	0.175	1.200	10.000	0.480	-3.367	-81.598	-185.969
T = 0.80	4.895	0.647	0.200	0.800	10.000	0.320	-3.347	-85.878	-174.571
T = 0.90	5.008	0.654	0.225	0.400	10.000	0.160	-3.327	-90.158	-163.172
T = 1.00	5.122	0.661	0.250	0.000	10.000	0.000	-3.307	-94.439	-151.773

VOULEZ-VOUS LES VALEURS DES FONCTIONS POUR UNE AUTRE VALEUR DE T ? (O/N): O

DONNEZ LA VALEUR DE T : 0.45

T = 0.45	4.497	0.621	0.113	2.200	10.000	0.880	-3.417	-70.896	-214.466

VOULEZ-VOUS LES VALEURS DES FONCTIONS POUR UNE AUTRE VALEUR DE T ? (O/N): N

DONNEZ LA VALEUR DE T OUI VOUS DONNE LA PLUS GRANDE SATISFACTION : 0.45

VALEURS DES FONCTIONS AU POINT COURANT : -3.417 -70.896 -214.466

DONNEZ L'INDICE DE LA FONCTION OBJECTIF QUE
VOUS DESIREZ PRENDRE COMME CRITERE DE REFERENCE : 3

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 1 (PAR RAPPORT AU CRIT. DE REF. 3) : 0.025

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 2 (PAR RAPPORT AU CRIT. DE REF. 3) : 0.002

ITERATION No 3

	X1	X2	X3	X4	X5	X6	F1(X)	F2(X)	F3(X)
T = 0.00	4.497	0.621	0.113	2.200	10.000	0.880	-3.417	-70.896	-214.466
T = 0.10	4.227	0.559	0.101	2.980	10.000	1.192	-3.370	-65.598	-234.217
T = 0.20	3.956	0.497	0.090	3.760	10.000	1.504	-3.322	-60.299	-253.968
T = 0.30	3.685	0.435	0.079	4.540	10.000	1.816	-3.275	-55.000	-273.719
T = 0.40	3.415	0.373	0.068	5.320	10.000	2.128	-3.228	-49.701	-293.470
T = 0.50	3.144	0.311	0.056	6.100	10.000	2.440	-3.180	-44.402	-313.221
T = 0.60	2.873	0.248	0.045	6.880	10.000	2.752	-3.133	-39.103	-332.972
T = 0.70	2.603	0.186	0.034	7.660	10.000	3.064	-3.085	-33.804	-352.723
T = 0.80	2.332	0.124	0.023	8.440	10.000	3.376	-3.038	-28.505	-372.475
T = 0.90	2.061	0.062	0.011	9.220	10.000	3.688	-2.990	-23.206	-392.226
T = 1.00	1.791	0.000	0.000	10.000	10.000	4.000	-2.943	-17.907	-411.977

VOULEZ-VOUS LES VALEURS DES FONCTIONS POUR UNE AUTRE VALEUR DE T ? (O/N): N

DONNEZ LA VALEUR DE T OUI VOUS DONNE LA PLUS GRANDE SATISFACTION : 0

VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N

Résultats :

Les résultats se trouvent dans deux fichiers nom.RES et nom.TRA où nom est un identificateur donné par l'utilisateur à la phase 0. Pour cet exemple nom vaut EXEG1.

EXEG1.RES

Les résultats de chaque itération sont résumés dans un tableau. Si k-1 et k représentent deux itérations successives, une ligne du tableau résumant les résultats de la kième itération contient les informations suivantes : une valeur de t (t correspond à α dans la description de l'algorithme au paragraphe 4.2) entre 0 et 1, la solution $(x_{k-1}+t(x_k-x_{k-1}))$ et les valeurs des fonctions objectifs $f_i(x_{k-1}+t(x_k-x_{k-1}))$. Pour t=0, nous avons la solution x_{k-1} qui est la solution choisie par le décideur à l'itération k-1 (si k=1, x_0 est la solution initiale trouvée par le programme) et pour t=1, nous avons la solution x_k qui est la solution optimale du programme $\underset{x\epsilon P}{\text{Max}}\ \Sigma w_i \nabla f_i(x_{k-1})x$ (voir paragraphe 4.2).

Le décideur a la possibilité de demander, en cours d'exécution, les valeurs des fonctions objectifs pour d'autres valeurs de t que celles prévues par le programme (voir dialogue décideur-machine). Ces nouvelles informations sont imprimées au bas du tableau à la suite des autres résultats.

METHODE DE GEOFFRION

Max $- 0.225\ X1 - 2.200\ X2 - 0.800\ X3 - 0.100\ X4 - 0.050\ X5 - 0.260\ X6$

Max $- 10.0\ X1 - 20.0\ X2 - 120.0\ X3$

Max $- 24.0\ X1 - 27.0\ X2 - 15.0\ X4 - 1.100\ X5 - 52.0\ X6$

$720.0\ X1 + 107.0\ X2 + 7080.0\ X3 + 134.0\ X5 + 1000.0\ X6 \geq 5000$

$0.200\ X1 + 10.100\ X2 + 13.200\ X3 + 0.750\ X4 + 0.150\ X5 + 1.200\ X6 \geq 12.500$

$344.0\ X1 + 460.0\ X2 + 1040.0\ X3 + 75.0\ X4 + 17.400\ X5 + 240.0\ X6 \geq 2500$

$18.0\ X1 + 151.0\ X2 + 78.0\ X3 + 2.500\ X4 + 0.200\ X5 + 4.0\ X6 \geq 63$

$X1 \leq 6$

$X2 \leq 1$

$X3 \leq 0.250$

$X4 \leq 10$

$X5 \leq 10$

$X6 \leq 4$

ITERATION No 1

	X1	X2	X3	X4	X5	X6	F1(X)	F2(X)	F3(X)
T = 0.00	5.450	0.981	0.000	0.000	10.000	0.000	-3.885	-74.120	-168.282
T = 0.10	5.084	0.883	0.000	1.000	10.000	0.400	-3.791	-68.498	-192.651
T = 0.20	4.718	0.785	0.000	2.000	10.000	0.800	-3.696	-62.877	-217.021
T = 0.30	4.352	0.687	0.000	3.000	10.000	1.200	-3.602	-57.256	-241.390
T = 0.40	3.986	0.589	0.000	4.000	10.000	1.600	-3.508	-51.635	-265.760
T = 0.50	3.620	0.491	0.000	5.000	10.000	2.000	-3.414	-46.013	-290.129
T = 0.60	3.254	0.392	0.000	6.000	10.000	2.400	-3.320	-40.392	-314.499
T = 0.70	2.888	0.294	0.000	7.000	10.000	2.800	-3.225	-34.771	-338.868
T = 0.80	2.522	0.196	0.000	8.000	10.000	3.200	-3.131	-29.149	-363.238
T = 0.90	2.157	0.098	0.000	9.000	10.000	3.600	-3.037	-23.528	-387.607
T = 1.00	1.791	0.000	0.000	10.000	10.000	4.000	-2.943	-17.907	-411.977

ITERATION No 2

	X1	X2	X3	X4	X5	X6	F1(X)	F2(X)	F3(X)
T = 0.00	3.986	0.589	0.000	4.000	10.000	1.600	-3.508	-51.635	-265.760
T = 0.10	4.100	0.596	0.025	3.600	10.000	1.440	-3.488	-55.915	-254.361
T = 0.20	4.213	0.603	0.050	3.200	10.000	1.280	-3.468	-60.195	-242.962
T = 0.30	4.327	0.610	0.075	2.800	10.000	1.120	-3.448	-64.476	-231.564
T = 0.40	4.440	0.618	0.100	2.400	10.000	0.960	-3.427	-68.756	-220.165
T = 0.50	4.554	0.625	0.125	2.000	10.000	0.800	-3.407	-73.037	-208.767
T = 0.60	4.668	0.632	0.150	1.600	10.000	0.640	-3.387	-77.317	-197.368
T = 0.70	4.781	0.639	0.175	1.200	10.000	0.480	-3.367	-81.598	-185.969
T = 0.80	4.895	0.647	0.200	0.800	10.000	0.320	-3.347	-85.878	-174.571
T = 0.90	5.008	0.654	0.225	0.400	10.000	0.160	-3.327	-90.158	-163.172
T = 1.00	5.122	0.661	0.250	0.000	10.000	0.000	-3.307	-94.439	-151.773
T = 0.45	4.497	0.621	0.113	2.200	10.000	0.880	-3.417	-70.896	-214.466

ITERATION No 3

	X1	X2	X3	X4	X5	X6	F1(X)	F2(X)	F3(X)
T = 0.00	4.497	0.621	0.113	2.200	10.000	0.880	-3.417	-70.896	-214.466
T = 0.10	4.227	0.559	0.101	2.980	10.000	1.192	-3.370	-65.598	-234.217
T = 0.20	3.956	0.497	0.090	3.760	10.000	1.504	-3.322	-60.299	-253.968
T = 0.30	3.685	0.435	0.079	4.540	10.000	1.816	-3.275	-55.000	-273.719
T = 0.40	3.415	0.373	0.068	5.320	10.000	2.128	-3.228	-49.701	-293.470
T = 0.50	3.144	0.311	0.056	6.100	10.000	2.440	-3.180	-44.402	-313.221
T = 0.60	2.873	0.248	0.045	6.880	10.000	2.752	-3.133	-39.103	-332.972
T = 0.70	2.603	0.186	0.034	7.660	10.000	3.064	-3.085	-33.804	-352.723
T = 0.80	2.332	0.124	0.023	8.440	10.000	3.376	-3.038	-28.505	-372.475
T = 0.90	2.061	0.062	0.011	9.220	10.000	3.688	-2.990	-23.206	-392.226
T = 1.00	1.791	0.000	0.000	10.000	10.000	4.000	-2.943	-17.907	-411.977

EXEG1.TRA

Ce fichier contient pour chaque itération la solution trouvée x_k et les informations fournies par le décideur pour son obtention, c'est-à-dire pour cette méthode, les taux de substitutions des critères par rapport à un critère de référence dont le taux est 1.

METHODE DE GEOFFRION

SOLUTION INITIALE

X1 = 5.45 X2 = 0.981 X3 = 0.000E+00 X4 = 0.000E+00 X5 = 10.0
X6 = 0.000E+00

VALEURS DES FONCTIONS OBJECTIFS

-3.8848 -74.1196 -168.2818

VECTEUR DES TAUX DE SUBSTITUTION : 0.0130 0.0400 1.0000

NOUVELLE SOLUTION

X1 = 3.99 X2 = 0.589 X3 = 0.000E+00 X4 = 4.00 X5 = 10.0
X6 = 1.60

VALEURS DES FONCTIONS OBJECTIFS

-3.5080 -51.6345 -265.7598

VECTEUR DES TAUX DE SUBSTITUTION : 0.0500 1.0000 0.5000

NOUVELLE SOLUTION

X1 = 4.50 X2 = 0.621 X3 = 0.113 X4 = 2.20 X5 = 10.0
X6 = 0.880

VALEURS DES FONCTIONS OBJECTIFS

-3.4174 -70.8965 -214.4659

VECTEUR DES TAUX DX SUBXTITUTION : 0.0250 0.0020 1.0000

Exemple 2

Malgré le fait que la méthode de Geoffrion n'exige pas que les fonctions objectifs soient linéaires, nous n'avons trouvé dans aucun ouvrage spécialisé un exemple comportant des fonctions objectifs non linéaires. Notre package contient des modules qui permettent l'analyse syntaxique d'une fonction non linéaire et le calcul de son gradient et de sa valeur en un point donné.

Voici un exemple avec deux fonctions objectifs dont une est non linéaire.

$$\text{Max } (x_1+x_2)^2 + x_1x_2$$

$$\text{Max } -3x_1 -2x_x$$

sous les contraintes

$$0.5x_1 + 0.25x_2 \leqslant 8$$

$$0.2x_1 + 0.2x_2 \leqslant 4$$

$$x_1 + 5x_2 \leqslant 72$$

$$x_1 , x_2 \geqslant 0$$

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : EXEG2
S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : CONT
DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : NONLIN
LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : N
```

Phase 1

Entrée des données

Notes concernant l'entrée des fonctions objectifs non linéaires :
Les fonctions logarithmiques et exponentielles ne sont pas admises. Tout autre forme ne comportant qu'un seul niveau de parenthèses est acceptée. Par exemple la forme $((x_1-1)+x_2)(x_1-x_2)$ n'est pas admise, mais la forme $(x_1-1+x_2)(x_1-x_2)$ est acceptée. Un exposant est toujours précédé par le caractère ^. Exemple : $(x_1+x_2)^2$ se donne par (x_1+x_2)^2. Les signes *(multiplié) et /(divisé) ne sont pas admis et les expressions $2*x_1*x_2$ et x_1/x_2 se donnent par $2x_1x_2$ et x_1x_2^-1.

```
NOMBRE DE FONCTIONS OBJECTIFS ?: 2

NOMBRE DE CONTRAINTES ?: 3

NOMBRE DE VARIABLES ?: 2

LE PROBLEME COMPORTE T'IL DES FONCTIONS  OBJECTIFS NON LINEAIRES (O/N) ? : O

LA 1ieme FONCTION OBJECTIF EST-ELLE LINEAIRE ? (O/N): N

DONNEZ ENTIEREMENT LA 1ieme FONCTION OBJECTIF (EX: 2x1^2 + x1x2) : (X1 + 2)^2 + X1X2

LA 2ieme FONCTION OBJECTIF EST-ELLE LINEAIRE ? (O/N): O

DONNEZ LES COEFFICIENTS DE LA 2ieme FONCTION OBJECTIF : -3 -2

DONNEZ LES ELEMENTS DE LA LIGNE  1
DE LA MATRICE DES CONTRAINTES A ?: 0.5 0.25

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 8

DONNEZ LES ELEMENTS DE LA LIGNE  2
DE LA MATRICE DES CONTRAINTES A ?: 0.2 0.2

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 4

DONNEZ LES ELEMENTS DE LA LIGNE  3
DE LA MATRICE DES CONTRAINTES A ?: 1 5

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 72
```

Fichier de données NONLIN.CNT

```
    2     3     2  (* NB DE CRITERES,NB DE CONTRAINTES,NB DE VARIABLES *)
FONCTIONS OBJECTIFS :
1 (* 0 = TOUTES LES FONCT. OBJ. SONT LINEAIRES,1 = AU MOINS UNE FONCT. OBJ. EST NON LINEAIRE *)
(X1+2)^2 + X1X2
-3.00     -2.00
CONTRAINTES :
0.500     0.250      1.  8.0
0.200     0.200      1.  4.0
 1.00      5.00      1.  72.
```

Phase 2

Dialogue décideur-machine

```
CHOIX DU MODULE (M,Z,G,S,T OU ?) ? : G

LA 1ieme FONCTION OBJECTIF EST-ELLE LINEAIRE ? (O/N): N

LA 2ieme FONCTION OBJECTIF EST-ELLE LINEAIRE ? (O/N): O

VALEURS DES FONCTIONS AU POINT COURANT :  4.000  0.000

DONNEZ L'INDICE DE LA FONCTION OBJECTIF QUE
VOUS DESIREZ PRENDRE COMME CRITERE DE REFERENCE : 2

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 1 ( PAR RAPPORT AU CRIT. DE REP. 2 ) : 0.04
```

ITERATION No 1

	X1	X2	F1(X)	F2(X)
T = 0.00	0.000	0.000	4.000	0.000
T = 0.10	1.600	0.000	12.960	-4.800
T = 0.20	3.200	0.000	27.040	-9.600
T = 0.30	4.800	0.000	46.240	-14.400
T = 0.40	6.400	0.000	70.560	-19.200
T = 0.50	8.000	0.000	100.000	-24.000
T = 0.60	9.600	0.000	134.560	-28.800
T = 0.70	11.200	0.000	174.240	-33.600
T = 0.80	12.800	0.000	219.040	-38.400
T = 0.90	14.400	0.000	268.960	-43.200
T = 1.00	16.000	0.000	324.000	-48.000

```
VOULEZ-VOUS LES VALEURS DES FONCTIONS POUR UNE AUTRE VALEUR DE T ? (O/N): O

DONNEZ LA VALEUR DE T : 0.65


T = 0.65   10.400    0.000   153.760     -31.200

VOULEZ-VOUS LES VALEURS DES FONCTIONS POUR UNE AUTRE VALEUR DE T ? (O/N): N

DONNEZ LA VALEUR DE T QUI VOUS DONNE LA PLUS GRANDE SATISFACTION : 0.65

VALEURS DES FONCTIONS AU POINT COURANT :  153.760  -31.200

DONNEZ L'INDICE DE LA FONCTION OBJECTIF QUE
VOUS DESIREZ PRENDRE COMME CRITERE DE REFERENCE : 1

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 2 ( PAR RAPPORT AU CRIT. DE REF. 1 ) : 0.99
```

ITERATION No 2

	X1	X2	F1(X)	F2(X)
T = 0.00	10.400	0.000	153.760	-31.200
T = 0.10	10.960	0.000	167.962	-32.880
T = 0.20	11.520	0.000	182.790	-34.560
T = 0.30	12.080	0.000	198.246	-36.240
T = 0.40	12.640	0.000	214.330	-37.920
T = 0.50	13.200	0.000	231.040	-39.600
T = 0.60	13.760	0.000	248.378	-41.280
T = 0.70	14.320	0.000	266.342	-42.960
T = 0.80	14.880	0.000	284.934	-44.640
T = 0.90	15.440	0.000	304.154	-46.320
T = 1.00	16.000	0.000	324.000	-48.000

VOULEZ-VOUS LES VALEURS DES FONCTIONS POUR UNE AUTRE VALEUR DE T ? (O/N): N

DONNEZ LA VALEUR DE T QUI VOUS DONNE LA PLUS GRANDE SATISFACTION : 0.1

VALEURS DES FONCTIONS AU POINT COURANT : 167.962 -32.880

DONNEZ L'INDICE DE LA FONCTION OBJECTIF QUE
VOUS DESIREZ PRENDRE COMME CRITERE DE REFERENCE : 1

DONNEZ LE TAUX DE SUBSTITUTION DU CRITERE 2 (PAR RAPPORT AU CRIT. DE REF. 1) : 0.8

ITERATION No 3

	X1	X2	F1(X)	F2(X)
T = 0.00	10.960	0.000	167.962	-32.880
T = 0.10	11.464	0.000	181.279	-34.392
T = 0.20	11.968	0.000	195.105	-35.904
T = 0.30	12.472	0.000	209.439	-37.416
T = 0.40	12.976	0.000	224.281	-38.928
T = 0.50	13.480	0.000	239.630	-40.440
T = 0.60	13.984	0.000	255.488	-41.952
T = 0.70	14.488	0.000	271.854	-43.464
T = 0.80	14.992	0.000	288.728	-44.976
T = 0.90	15.496	0.000	306.110	-46.488
T = 1.00	16.000	0.000	324.000	-48.000

VOULEZ-VOUS LES VALEURS DES FONCTIONS POUR UNE AUTRE VALEUR DE T ? (O/N): N

DONNEZ LA VALEUR DE T QUI VOUS DONNE LA PLUS GRANDE SATISFACTION : 0

VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N

Résultats :

EXEG2.RES

METHODE DE GEOFFRION

Max $(X1+2)^2 + X1X2$

Max $- 3.0\ X1 - 2.0\ X2$

$0.500\ X1 + 0.250\ X2 \leq 8$

$0.200\ X1 + 0.200\ X2 \leq 4$

$X1 + 5.0\ X2 \leq 72$

ITERATION No 1

	X1	X2	F1(X)	F2(X)
T = 0.00	0.000	0.000	4.000	0.000
T = 0.10	1.600	0.000	12.960	-4.800
T = 0.20	3.200	0.000	27.040	-9.600
T = 0.30	4.800	0.000	46.240	-14.400
T = 0.40	6.400	0.000	70.560	-19.200
T = 0.50	8.000	0.000	100.000	-24.000
T = 0.60	9.600	0.000	134.560	-28.800
T = 0.70	11.200	0.000	174.240	-33.600
T = 0.80	12.800	0.000	219.040	-38.400
T = 0.90	14.400	0.000	268.960	-43.200
T = 1.00	16.000	0.000	324.000	-48.000
T = 0.65	10.400	0.000	153.760	-31.200

ITERATION No 2

	X1	X2	F1(X)	F2(X)
T = 0.00	10.400	0.000	153.760	-31.200
T = 0.10	10.960	0.000	167.962	-32.880
T = 0.20	11.520	0.000	182.790	-34.560
T = 0.30	12.080	0.000	198.246	-36.240
T = 0.40	12.640	0.000	214.330	-37.920
T = 0.50	13.200	0.000	231.040	-39.600
T = 0.60	13.760	0.000	248.378	-41.280
T = 0.70	14.320	0.000	266.342	-42.960
T = 0.80	14.880	0.000	284.934	-44.640
T = 0.90	15.440	0.000	304.154	-46.320
T = 1.00	16.000	0.000	324.000	-48.000

ITERATION No 3

	X1	X2	F1(X)	F2(X)
T = 0.00	10.960	0.000	167.962	-32.880
T = 0.10	11.464	0.000	181.279	-34.392
T = 0.20	11.968	0.000	195.105	-35.904
T = 0.30	12.472	0.000	209.439	-37.416
T = 0.40	12.976	0.000	224.281	-38.928
T = 0.50	13.480	0.000	239.630	-40.440
T = 0.60	13.984	0.000	255.488	-41.952
T = 0.70	14.488	0.000	271.854	-43.464
T = 0.80	14.992	0.000	288.728	-44.976
T = 0.90	15.496	0.000	306.110	-46.488
T = 1.00	16.000	0.000	324.000	-48.000

EXEG2.TRA

```
                    METHODE DE GEOFFRION

                          SOLUTION INITIALE

                       X1 = 0.000E+00 X2 = 0.000E+00

                       VALEURS DES FONCTIONS OBJECTIFS

                              4.0000     0.0000
VECTEUR DES TAUX DE SUBSTITUTION :  0.0400   1.0000
                          NOUVELLE SOLUTION

                       X1 =  10.4      X2 = 0.000E+00

                       VALEURS DES FONCTIONS OBJECTIFS

                            153.7600  -31.2000
VECTEUR DES TAUX DE SUBSTITUTION :  1.0000   0.9900
                          NOUVELLE SOLUTION

                       X1 =  11.0      X2 = 0.000E+00

                       VALEURS DES FONCTIONS OBJECTIFS

                            167.9616  -32.8800
VECTEUR DES TAUX DE SUBSTITUTION :  1.0000   0.8000
```

5.2.4 Méthode de Zionts-Wallenius

Exemple : problème de nutrition

Pour illustrer cette méthode, nous avons choisi le problème de nutrition que nous avons déjà décrit sous l'exemple 1 de 5.2.3.

Pour simuler les réponses du décideur, nous avons supposé que sa fonction d'utilité (que la méthode suppose connue de façon implicite par le décideur) est $0.959\ f_1(x) + 0.029\ f_2(x) + 0.012\ f_3(x)$. Cette fonction a été utilisée par Hwang et Massud dans [9] dans le même but, et comme nous l'avons déjà signalé dans le paragraphe 4.3, ces deux auteurs n'arrivent pas à une solution optimale du programme

$$\max_{x \in P} 0.959\ f_1(x) + 0.029\ f_2(x) + 0.012\ f_3(x)$$

où P est l'ensemble des solutions réalisables. Avec les modifications que nous avons apportées à cette méthode, nous arrivons à la solution désirée.

Le fichier de données NUTRI.CNT existe et seules la phase 0 et la phase 2 sont exécutées.

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : EXEZ
S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : CONT
DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : NUTRI
LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : O
```

Phase 2

Dialogue décideur-machine

```
CHOIX DU MODULE (M,Z,G,S,T OU ?) ? : Z

ITERATION 1
-----------

UNE VARIATION DE   0.0855 SUR LE CRITERE  1

UNE VARIATION DE   2.6564 SUR LE CRITERE  2

UNE VARIATION DE -10.1333 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELIORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): A

UNE VARIATION DE   2.3130 SUR LE CRITERE  1

UNE VARIATION DE -81.2772 SUR LE CRITERE  2

UNE VARIATION DE  66.0341 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELIORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): A

ITERATION 2
-----------

UNE VARIATION DE  -1.8056 SUR LE CRITERE  1

UNE VARIATION DE  63.4490 SUR LE CRITERE  2

UNE VARIATION DE -51.5495 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELIORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): D

UNE VARIATION DE   0.0373 SUR LE CRITERE  1

UNE VARIATION DE  -1.5028 SUR LE CRITERE  2

UNE VARIATION DE   0.6919 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELIORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): A

UNE VARIATION DE   0.0442 SUR LE CRITERE  1

UNE VARIATION DE  -7.2147 SUR LE CRITERE  2

UNE VARIATION DE  13.8368 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELIORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): D

UNE VARIATION DE  -0.0929 SUR LE CRITERE  1

UNE VARIATION DE  14.5347 SUR LE CRITERE  2

UNE VARIATION DE -41.3648 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELIORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): D
```

```
ITERATION 3
___________

UNE VARIATION DE   0.8155 SUR LE CRITERE  1

UNE VARIATION DE -42.1820 SUR LE CRITERE  2

UNE VARIATION DE  -2.9169 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): D

UNE VARIATION DE  -3.3576 SUR LE CRITERE  1

UNE VARIATION DE 135.3117 SUR LE CRITERE  2

UNE VARIATION DE -62.2976 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): D

UNE VARIATION DE   0.1898 SUR LE CRITERE  1

UNE VARIATION DE   3.1421 SUR LE CRITERE  2

UNE VARIATION DE -36.1197 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): D

UNE VARIATION DE  -0.1441 SUR LE CRITERE  1

UNE VARIATION DE   0.3741 SUR LE CRITERE  2

UNE VARIATION DE  10.3429 SUR LE CRITERE  3

EST CE GLOBALEMENT UNE AMELORATION , UNE DEGRADATION
OU ETES VOUS INDIFFERENT ? (A,D,I): D

VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N
```

A la premiére itération, nous avons répondu deux fois par A car les expressions 0.959 (0.0855) + 0.029 (2.6564) + 0.012 (-10.13333) = 0.0374 et 0.959 (2.313) + 0.029 (-81.2772) + 0.012 (66.0341) = 0.6535 sont positives. Pour les autres itérations, le même procédé a été utilisé pour répondre aux questions posées.

Nous remarquons que lorsque toutes les réponses sont D, c'est-à-dire qu'il y a dégradation suivant toutes les directions indiquées, le programme s'arrête.

Résultats :

EXEZ.RES

Nous avons ici les solutions trouvées lors des différentes itérations. La solution Y_2 est la solution cherchée. On peut vérifier que c'est une solution optimale du programme

$$\underset{x \in P}{\text{Max}}\ 0.959\ f_1(x) + 0.029\ f_2(x) + 0.012\ f_3(x).$$

```
                         METHODE DE ZIONTS-WALLENIUS

Max  - 0.225 X1 - 2.200 X2 - 0.800 X3 - 0.100 X4 - 0.050 X5 - 0.260 X6
Max  - 10.0 X1 - 20.0 X2 - 120.0 X3
Max  - 26.0 X1 - 27.0 X2 - 15.0 X4 - 1.100 X5 - 52.0 X6
     720.0 X1 + 107.0 X2 + 7080.0 X3 + 134.0 X5 + 1000.0 X6 >= 5000
     0.200 X1 + 10.100 X2 + 13.200 X3 + 0.750 X4 + 0.150 X5 + 1.200 X6 >= 12.500
     344.0 X1 + 460.0 X2 + 1040.0 X3 + 75.0 X4 + 17.400 X5 + 240.0 X6 >= 2500
     18.0 X1 + 151.0 X2 + 78.0 X3 + 2.500 X4 + 0.200 X5 + 4.0 X6 >= 63
     X1 <= 6
     X2 <= 1
     X3 <= 0.250
     X4 <= 10
     X5 <= 10
     X6 <= 4
```

	X1	X2	X5	X7	X10	X11	X12	X13	X14	X16 •	F1(X)	F2(X)	F3(X)
Y0	5.450	0.981	10.000	368.673	165.253	0.550	0.019	0.250	10.000	4.000	-3.845	-74.120	-168.282

	X1	X3	X4	X5	X7	X10	X11	X12	X13	X16 •	F1(X)	F2(X)	F3(X)
Y1	3.961	0.205	10.000	10.000	644.424	51.302	2.039	1.000	0.045	4.000	-2.555	-64.278	-256.069

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16 •	F1(X)	F2(X)	F2(X)
Y2	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.255	-253.274

EXEZ.TRA

Au début de ce fichier, nous trouvons les valeurs idéales des fonctions objectifs avec les solutions correspondantes. Puis pour chaque itération, nous avons le vecteur de poids trouvé, la solution efficace générée à l'aide de ce vecteur, les valeurs des fonctions objectifs, la liste des variables efficaces et enfin les décisions de l'utilisateur concernant chacune de ces variables.

```
                          METHODE DE ZIONTS-WALLENIUS

VALEUR IDEALE DE LA FONCTION OBJECTIF NO  1
        F =    -2.256

LA SOLUTION CORESPONDANTE EST:

        X1 =  3.78     X3 = 0.250     X4 =  10.0     X6 = 0.786     X7 =  280.

        X10 =  52.7     X11 =  2.22     X12 =  1.00     X15 =  10.0     X16 =  3.21

VALEUR IDEALE DE LA FONCTION OBJECTIF NO  2
        F =   -17.907

LA SOLUTION CORESPONDANTE EST:

        X1 =  1.79     X4 =  10.0     X5 =  10.0     X6 =  4.00     X7 = 0.163E+04

        X8 =  1.66     X10 =  12.2     X11 =  4.21     X12 =  1.00     X13 = 0.250

VALEUR IDEALE DE LA FONCTION OBJECTIF NO  3
        F =  -150.047

LA SOLUTION CORESPONDANTE EST:

        X1 =  4.67     X2 =  1.00     X3 = 0.250     X5 =  10.0     X7 = 0.158E+04

        X8 =  3.33     X10 =  194.     X11 =  1.33     X14 =  10.0     X16 =  4.00

                              VECTEUR DE POIDS INITIAL

                              0.333    0.333    0.333

                                  SOLUTION INITIALE

        X1 =  5.45     X2 = 0.981     X5 =  10.0     X7 =  369.     X10 =  185.

        X11 = 0.550     X12 = 0.188E-01 X13 = 0.250     X14 =  10.0     X16 =  4.00
```

VALEURS DES FONCTIONS OBJECTIFS

-3.8848 -74.1196 -168.2818

LES VARIABLES EFFICACES SONT : X4 X3

LE VECTEUR DES VARIATIONS : 0.0855 2.6564 -10.1333
RELATIF A LA VARIABLE EFFICACE X4 REPRESENTE UNE AMELIORATION GLOBALE POUR LE DECIDEUR

LE VECTEUR DES VARIATIONS : 2.3130 -81.2772 66.0341
RELATIF A LA VARIABLE EFFICACE X3 REPRESENTE UNE AMELIORATION GLOBALE POUR LE DECIDEUR

NOUVEAU VECTEUR DE POIDS

0.943 0.040 0.017

NOUVELLE SOLUTION

X1 = 3.96 X3 = 0.205 X4 = 10.0 X5 = 10.0 X7 = 644.

X10 = 51.3 X11 = 2.04 X12 = 1.00 X13 = 0.449E-01 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.5554 -64.2282 -256.0694

LES VARIABLES EFFICACES SONT : X2 X15 X14 X6

LE VECTEUR DES VARIATIONS : -1.8056 63.4490 -51.5495
RELATIF A LA VARIABLE EFFICACE X2 REPRESENTE UNE DEGRADATION GLOBALE POUR LE DECIDEUR

LE VECTEUR DES VARIATIONS : 0.0373 -1.5028 0.6919
RELATIF A LA VARIABLE EFFICACE X15 REPRESENTE UNE AMELIORATION GLOBALE POUR LE DECIDEUR

LE VECTEUR DES VARIATIONS : 0.0442 -7.2147 13.8368
RELATIF A LA VARIABLE EFFICACE X14 REPRESENTE UNE DEGRADATION GLOBALE POUR LE DECIDEUR

LE VECTEUR DES VARIATIONS : -0.0929 14.5347 -41.3648
RELATIF A LA VARIABLE EFFICACE X6 REPRESENTE UNE DEGRADATION GLOBALE POUR LE DECIDEUR

NOUVEAU VECTEUR DE POIDS

0.970 0.022 0.008

NOUVELLE SOLUTION

X1 = 4.03 X3 = 0.250 X4 = 10.0 X5 = 5.96 X7 = 470.

X10 = 55.2 X11 = 1.97 X12 = 1.00 X15 = 4.04 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.4047 -70.2993 -253.2743

LES VARIABLES EFFICACES SONT : X2 X13 X6 X14

LE VECTEUR DES VARIATIONS : 0.8155 -42.1820 -2.9169
RELATIF A LA VARIABLE EFFICACE X2 REPRESENTE UNE DEGRADATION GLOBALE POUR LE DECIDEUR

LE VECTEUR DES VARIATIONS : -3.3576 135.3117 -62.2976
RELATIF A LA VARIABLE EFFICACE X13 REPRESENTE UNE DEGRADATION GLOBALE POUR LE DECIDEUR

LE VECTEUR DES VARIATIONS : 0.1898 3.1421 -36.1197
RELATIF A LA VARIABLE EFFICACE X6 REPRESENTE UNE DEGRADATION GLOBALE POUR LE DECIDEUR

LE VECTEUR DES VARIATIONS : -0.1441 0.3741 10.3429
RELATIF A LA VARIABLE EFFICACE X14 REPRESENTE UNE DEGRADATION GLOBALE POUR LE DECIDEUR

5.2.5 Algorithme interactif du simplexe

Nous avons voulu générer toutes les solutions efficaces et extrêmes du problème de nutrition et d'un programme multiobjectif traité par Zeleny dans [29].

Exemple 1 : problème de nutrition

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : EXEM1
S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : CONT
DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : NUTR1
LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : O
```

Phase 2

Le dialogue décideur-machine étant assez long, nous n'en donnerons qu'un extrait.

Dialogue décideur-machine

Première itération

```
CHOIX DU MODULE (M,Z,G,S,T OU ?) ? : M

SI LA VARIABLE HORS BASE X4 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE :
     0.3301  10.2555 -39.1215

SI LA VARIABLE HORS BASE X3 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE :
     0.5782 -20.3193  16.5065

DONNEZ L'INDICE DE LA VARIABLE QUI ENTRE DANS LA BASE : 4

VALEURS DES FONCTIONS OBJECTIFS :
  -3.5547  -63.8641 -207.4033

ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N): N
```

.
.
.
.
.
.
.

Dernière itération

```
SI LA VARIABLE HORS BASE X2 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE :
      0.0497  -2.5709  -0.1778

SI LA VARIABLE HORS BASE X14 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE :
     -0.1153   0.2993   8.2743

SI LA VARIABLE HORS BASE X6 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE :
      0.1492   2.4703 -28.3967

SI LA VARIABLE HORS BASE X13 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE :
     -0.1506   6.0710  -2.7951

DONNEZ L'INDICE DE LA VARIABLE QUI ENTRE DANS LA BASE : 13

VALEURS DES FONCTIONS OBJECTIFS :
  -2.5554  -64.2282 -256.0694

ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N): O

VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N
```

Résultats

EXEM1.RES

Ce fichier contient la liste des solutions efficaces et extrêmes du problème.

```
                    ALGORITHME INTERACTIF DU SIMPLEXE

Max  - 0.225 X1 - 2.200 X2 - 0.800 X3 - 0.100 X4 - 0.050 X5 - 0.260 X6

Max  - 10.0 X1 - 20.0 X2 - 120.0 X3

Max  - 24.0 X1 - 27.0 X2 - 15.0 X4 - 1.100 X5 - 52.0 X6

     720.0 X1 + 107.0 X2 + 7080.0 X3 + 134.0 X5 + 1000.0 X6 >= 5000

     0.200 X1 + 10.100 X2 + 13.200 X3 + 0.750 X4 + 0.150 X5 + 1.200 X6 >= 12.500

     344.0 X1 + 460.0 X2 + 1040.0 X3 + 75.0 X4 + 17.400 X5 + 240.0 X6 >= 2500

     18.0 X1 + 151.0 X2 + 78.0 X3 + 2.500 X4 + 0.200 X5 + 4.0 X6 >= 63

     X1 <= 6

     X2 <= 1

     X3 <= 0.250

     X4 <= 10

     X5 <= 10

     X6 <= 4
```

	X1	X2	X5	X7	X10	X11	X12	X13	X14	X16	* P1(X)	P2(X)	P3(X)
Y0	5.450	0.901	10.000	248.673	185.253	0.550	0.019	0.250	10.000	8.000	-3.885	-74.120	-168.282

	X1	X2	X4	X5	X10	X11	X12	X13	X14	X16	* P1(X)	P2(X)	P3(X)
Y1	4.979	0.704	3.061	10.000	144.540	1.021	0.296	0.250	6.139	4.000	-3.555	-62.066	-207.403

	X1	X2	X3	X4	X5	X10	X11	X12	X13	X16	* P1(X)	P2(X)	P3(X)
Y2	4.102	0.138	0.098	10.000	10.000	66.234	1.898	0.862	0.152	4.000	-2.804	-55.498	-263.162

	X1	X2	X4	X5	X6	X10	X11	X12	X13	X16	* P1(X)	P2(X)	P3(X)
Y3	3.680	0.156	10.000	10.000	0.994	57.710	2.320	0.844	0.250	2.006	-2.929	-39.912	-305.196

	X1	X4	X5	X6	X7	X10	X11	X12	X13	X16	* P1(X)	P2(X)	P3(X)
Y4	2.882	10.000	10.000	2.436	851.141	25.614	2.110	1.000	0.250	1.564	-2.702	-20.016	-356.051

	X1	X4	X5	X6	X7	X10	X11	X12	X13	X14	* P1(X)	P2(X)	P3(X)
Y5	2.302	7.653	10.000	4.000	1997.778	15.576	2.690	1.000	0.250	2.347	-2.022	-23.025	-389.049

	X1	X4	X5	X6	X7	X8	X10	X11	X12	X13	* P1(X)	P2(X)	P3(X)
Y6	1.751	10.000	10.000	4.000	1629.303	1.458	12.222	4.209	1.000	0.250	-2.942	-17.907	-411.977

	X1	X4	X6	X7	X8	X10	X11	X12	X13	X15	* P1(X)	P2(X)	P3(X)
Y7	2.297	10.000	4.000	653.409	0.259	19.337	2.702	1.000	0.250	10.000	-2.557	-22.965	-413.116

	X1	X4	X6	X7	X10	X11	X12	X13	X14	X15	* P1(X)	P2(X)	P3(X)
Y8	2.377	9.433	4.000	711.112	19.860	3.623	1.000	0.250	0.267	10.000	-2.528	-23.765	-409.531

	X1	X4	X6	X7	X10	X11	X12	X13	X15	X16	* P1(X)	P2(X)	P3(X)
Y9	2.467	10.000	3.755	531.799	21.430	3.533	1.000	0.250	10.000	0.245	-2.532	-24.671	-404.496

	X1	X3	X4	X6	X7	X10	X11	X12	X15	X16	* P1(X)	P2(X)	P3(X)
Y10	3.782	0.250	10.000	0.786	279.869	52.727	2.217	1.000	10.000	3.214	-2.286	-67.829	-281.671

	X1	X2	X3	X4	X6	X10	X11	X12	X15	X16	• F1(X)	F2(X)	F2(X)
Y11	4.045	0.051	0.250	10.000	0.312	63.291	1.955	0.949	10.000	3.600	-2.304	-71.470	-264.686

	X1	X2	X3	X4	X5	X10	X11	X12	X15	X16	• F1(X)	F2(X)	F3(X)
Y12	4.165	0.061	0.250	10.000	1.676	66.010	1.035	0.939	8.224	4.000	-2.355	-72.870	-253.452

	X1	X2	X3	X4	X10	X11	X12	X14	X15	X16	• F1(X)	F2(X)	F3(X)
Y13	4.453	0.223	0.250	8.074	90.538	1.547	0.777	1.926	10.000	4.000	-2.500	-78.993	-234.001

	X1	X2	X3	X7	X10	X11	X12	X14	X15	X16	• F1(X)	F2(X)	F3(X)
Y14	5.438	0.603	0.250	770.988	175.662	0.562	0.197	10.000	10.000	4.000	-3.191	-100.440	-152.188

	X1	X2	X3	X5	X7	X10	X11	X12	X14	X16	• F1(X)	F2(X)	F3(X)
Y15	5.122	0.661	0.250	10.000	1060.552	150.499	0.878	0.339	10.000	4.000	-3.307	-94.429	-151.773

	X1	X2	X2	X5	X7	X8	X10	X11	X14	X16	• F1(X)	F2(X)	F3(X)
Y16	4.649	1.000	0.250	10.000	1578.397	3.334	193.535	1.331	10.000	4.000	-3.950	-96.686	-150.047

	X1	X2	X3	X5	X7	X10	X11	X12	X14	X16	• F1(X)	F2(X)	F3(X)
Y17	5.122	0.661	0.250	10.000	1060.552	150.499	0.878	0.339	10.000	4.000	-3.307	-94.439	-151.773

	X1	X3	X4	X5	X7	X10	X11	X12	X14	X16	• F1(X)	F2(X)	F2(X)
Y18	4.000	0.250	9.200	10.000	990.002	53.500	2.000	1.000	0.800	4.000	-2.520	-70.000	-245.000

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16	• F1(X)	F2(X)	F3(X)
Y19	4.030	0.250	10.000	5.960	470.201	55.221	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

	X1	X3	X4	X5	X7	X10	X11	X12	X13	X16	• F1(X)	F2(X)	F3(X)
Y20	3.961	0.205	10.000	10.000	644.428	51.302	2.039	1.000	0.048	4.000	-2.555	-64.228	-256.069

EXEM1.TRA

A chaque itération, nous avons la solution efficace avec les valeurs des fonctions objectifs, la liste des variables efficaces et la variable choisie pour l'entrée dans la base. Le fichier étant trés grand, nous ne donnerons que la premiére itération.

```
                        ALGORITHME INTERACTIF OU SIMPLEXE

                                SOLUTION INITIALE

        X1 =  5.45      X2 = 0.981      X5 =  10.0      X7 =  369.      X10 =  185.
        X11 = 0.550     X12 = 0.188E-01 X13 = 0.250     X14 =  10.0     X16 =  4.00

                            VALEURS DES FONCTIONS OBJECTIFS

                               -3.8848  -74.1196 -168.2818
LES VARIABLES EFFICACES SONT :  X4 X3
LA VARIABLE X4 A ETE CHOISIE POUR L'ENTREE DANS LA BASE
                                NOUVELLE SOLUTION

        X1 =  4.98      X2 = 0.704      X4 =  3.86      X5 =  10.0      X10 =  145.
        X11 =  1.02     X12 = 0.296     X13 = 0.250     X14 =  6.14     X16 =  4.00

                            VALEURS DES FONCTIONS OBJECTIFS

                               -3.5547  -63.8641 -207.4033
```

Exemple 2 : (voir [29] page 499)

Enoncé du problème :

$$\text{Max } x_1 + 2x_2 - x_3 + 3x_4 + 2x_5 + x_7$$

$$\text{Max } x_2 + x_3 + 2x_4 + 3x_5 + x_6$$

$$\text{Max } x_1 + x_3 - x_4 - x_6 - x_7$$

Sous les contraintes :

$$x_1 + 2x_2 + x_3 + x_4 + 2x_5 + x_6 + x_7 \leqslant 16$$

$$-2x_1 - x_2 + x_4 + 2x_5 + x_7 \leqslant 16$$

$$-x_1 + x_3 + 2x_5 - 2x_7 \leqslant 16$$

$$x_2 + 2x_3 - x_4 + x_5 - 2x_6 - x_7 \leqslant 16$$

$$x_i \geqslant 0 \quad , \quad i=1,\dots,7$$

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : EXEM2
S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : CONT
DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : PRODUCT
LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : N
```

Phase 1

Entrée des données

```
NOMBRE DE FONCTIONS OBJECTIFS ?: 3
NOMBRE DE CONTRAINTES ?: 4
NOMBRE DE VARIABLES ?: 7
LE PROBLEME COMPORTE T'IL DES FONCTIONS  OBJECTIFS NON LINEAIRES (O/N) ? : N
```

```
DONNEZ LES COEFFICIENTS DE LA FONCTION OBJECTIF
NUMERO   1 ?: 1 2 -1 3 2 0 1

DONNEZ LES COEFFICIENTS DE LA FONCTION OBJECTIF
NUMERO   2 ?: 0 1 1 2 3 1 0

DONNEZ LES COEFFICIENTS DE LA FONCTION OBJECTIF
NUMERO   3 ?: 1 0 1 -1 0 -1 -1

DONNEZ LES ELEMENTS DE LA LIGNE  1
DE LA MATRICE DES CONTRAINTES A ?: 1 2 1 1 2 1 2

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 16

DONNEZ LES ELEMENTS DE LA LIGNE  2
DE LA MATRICE DES CONTRAINTES A ?: -2 -1 0 1 2 0 1

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 16

DONNEZ LES ELEMENTS DE LA LIGNE  3
DE LA MATRICE DES CONTRAINTES A ?: -1 0 1 0 2 0 -2

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 16

DONNEZ LES ELEMENTS DE LA LIGNE  4
DE LA MATRICE DES CONTRAINTES A ?: 0 1 2 -1 1 -2 -1

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 16
```

Fichier de données PRODUCT.CNT

```
    3    4    7  (* NB DE CRITERES,NB DE CONTRAINTES,NB DE VARIABLES *)
FONCTIONS OBJECTIFS :
0 (* 0 = TOUTES LES FONCT. OBJ. SONT LINEAIRES,1 = AU MOINS UNE FONCT. OBJ. EST NON LINEAIRE *)
 1.00      2.00      -1.00      3.00      2.00      0.000E+00 1.00
0.000E+00 1.00      1.00       2.00      3.00      1.00      0.000E+00
 1.00     0.000E+00 1.00      -1.00      0.000E+00-1.00      -1.00
CONTRAINTES :
 1.00      2.00      1.00      1.00      2.00      1.00      2.00      1.  16.
-2.00     -1.00      0.000E+00 1.00      2.00      0.000E+00 1.00      1.  16.
-1.00      0.000E+00 1.00      0.000E+00 2.00      0.000E+00-2.00      1.  16.
0.000E+00 1.00       2.00     -1.00      1.00     -2.00     -1.00      1.  16.
```

Phase 2

Dialogue décideur-machine

```
CHOIX DU MODULE (M,Z,G,S,T OU ?) ? : M

SI LA VARIABLE HORS BASE X1 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
   -32.0000 -32.0000  32.0000

DONNEZ L'INDICE DE LA VARIABLE QUI ENTRE DANS LA BASE : 1

VALEURS DES FONCTIONS OBJECTIFS :
  16.0000    0.0000   16.0000

ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N): N

SI LA VARIABLE HORS BASE X4 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
    32.0000  32.0000 -32.0000

SI LA VARIABLE HORS BASE X3 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
   -16.0000   8.0000   0.0000

SI LA VARIABLE HORS BASE X5 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
     0.0000  24.0000 -16.0000

DONNEZ L'INDICE DE LA VARIABLE QUI ENTRE DANS LA BASE : 5

VALEURS DES FONCTIONS OBJECTIFS :
  16.0000   24.0000    0.0000

ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N): N

SI LA VARIABLE HORS BASE X4 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
    32.0000   8.0000 -16.0000

SI LA VARIABLE HORS BASE X3 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
   -10.6667  -2.6667   5.3333

SI LA VARIABLE HORS BASE X1 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
     0.0000 -24.0000  16.0000

DONNEZ L'INDICE DE LA VARIABLE QUI ENTRE DANS LA BASE : 3

VALEURS DES FONCTIONS OBJECTIFS :
   5.3333   21.3333    5.3333

ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N): N

SI LA VARIABLE HORS BASE X4 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
     0.0000   0.0000   0.0000

SI LA VARIABLE HORS BASE X11 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRI
    10.6667   2.6667  -5.3333

SI LA VARIABLE HORS BASE X1 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
    -5.3333 -13.3333  10.6667
```

```
DONNEZ L'INDICE DE LA VARIABLE QUI ENTRE DANS LA BASE : 4

VALEURS DES FONCTIONS OBJECTIFS :
   5.3333   21.3333    5.3333

ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N): N

SI LA VARIABLE HORS BASE X5 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
      0.0000   0.0000   0.0000

SI LA VARIABLE HORS BASE X11 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
     42.6667  10.6667 -21.3333

SI LA VARIABLE HORS BASE X1 RENTRE DANS LA BASE LES VARIATIONS SUR LES CRITERES SONT DANS L'ORDRE
     -5.3333 -13.3333  10.6667

DONNEZ L'INDICE DE LA VARIABLE QUI ENTRE DANS LA BASE : 1

VALEURS DES FONCTIONS OBJECTIFS :
   0.0000    8.0000   16.0000

ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N): O

VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N
```

Résultats

EXEM2.RES

```
                    ALGORITHME INTERACTIF DU SIMPLEXE

Max  X1 + 2.0 X2 - X3 + 3.0 X4 + 2.0 X5 + X7

Max  X2 + X3 + 2.0 X4 + 3.0 X5 + X6

Max  X1 + X3 - X4 - X6 - X7

     X1 + 2.0 X2 + X3 + X4 + 2.0 X5 + X6 + 2.0 X7 <= 16

     - 2.0 X3 - X2 + X4 + 2.0 X5 + X7 <= 16

     - X1 + X3 + 2.0 X5 - 2.0 X7 <= 16

     X2 + 2.0 X3 - X4 + X5 - 2.0 X6 - X7 <= 16
```

	X4	X9	X10	X11 * F1(X)	F2(X)	F3(X)	
Y0	16.000	0.000	16.000	32.000	48.000	32.000	-16.000

	X1	X9	X10	X11 * F1(X)	F2(X)	F3(X)	
Y1	16.000	48.000	32.000	16.000	16.000	0.000	16.000

	X5	X9	X10	X11 * F1(X)	F2(X)	F3(X)	
Y2	8.000	0.000	0.000	8.000	16.000	24.000	0.000

	X3	X5	X9	X10* F1(X)	F2(X)	F3(X)	
Y3	5.333	5.333	5.333	0.000	5.333	21.333	5.333

	X3	X4	X9	X10* F1(X)	F2(X)	F3(X)	
Y4	10.667	5.333	10.667	5.333	5.333	21.333	5.333

	X1	X3	X9	X10 * F1(X)	F2(X)	F3(X)	
Y5	8.000	8.000	32.000	16.000	0.000	8.000	16.000

Remarque : En traitant cet exemple, Zeleny trouve en plus des 6 solutions efficaces ci-dessus 2 solutions non efficaces $Y_6 = (0,0,12,0,0,4,0)$ et $Y_7 = (0,0,48/5,0,0,0,16/5)$ qu'il rejette après avoir testé leur non efficacité. Ce cas ne peut pas se présenter dans notre méthode car nous ne faisons entrer dans la base que les variables hors base efficaces.

EXEM2.TRA

ALGORITHME INTERACTIF DU SIMPLEXE

SOLUTION INITIALE

X4 = 16.0 X9 = 0.000E+00 X10 = 16.0 X11 = 32.0

VALEURS DES FONCTIONS OBJECTIFS

48.0000   32.0000   -16.0000

LES VARIABLES EFFICACES SONT : X1

LA VARIABLE X1 A ETE CHOISIE POUR L'ENTREE DANS LA BASE

NOUVELLE SOLUTION

X1 = 16.0   X9 = 48.0   X10 = 32.0   X11 = 16.0

VALEURS DES FONCTIONS OBJECTIFS

16.0000   0.0000   16.0000

LES VARIABLES EFFICACES SONT : X4 X3 X5

LA VARIABLE X5 A ETE CHOISIE POUR L'ENTREE DANS LA BASE

NOUVELLE SOLUTION

X5 = 8.00   X9 = 0.000E+00 X10 = 0.000E+00 X11 = 8.00

VALEURS DES FONCTIONS OBJECTIFS

16.0000   24.0000   0.0000

LES VARIABLES EFFICACES SONT : X4 X3 X1

LA VARIABLE X3 A ETE CHOISIE POUR L'ENTREE DANS LA BASE

NOUVELLE SOLUTION

X3 = 5.33   X5 = 5.33   X9 = 5.33   X10 = 0.000E+00

VALEURS DES FONCTIONS OBJECTIFS

5.3333   21.3333   5.3333

LES VARIABLES EFFICACES SONT : X4 X11 X1

LA VARIABLE X4 A ETE CHOISIE POUR L'ENTREE DANS LA BASE

NOUVELLE SOLUTION

X3 = 10.7   X4 = 5.33   X9 = 10.7   X10 = 5.33

VALEURS DES FONCTIONS OBJECTIFS

5.3333   21.3333   5.3333

LES VARIABLES EFFICACES SONT : X5 X11 X1

LA VARIABLE X1 A ETE CHOISIE POUR L'ENTREE DANS LA BASE

NOUVELLE SOLUTION

X1 = 8.00   X3 = 8.00   X9 = 32.0   X10 = 16.0

VALEURS DES FONCTIONS OBJECTIFS

0.0000   8.0000   16.0000

5.2.6 Méthode de STEUER

En supposant que le décideur puisse donner dans le cas du problème de nutrition les intervalles de poids suivants :]0.9 1[,]0.02 0.03[,]0.001 0.015[, nous avons voulu voir si nous arrivions à une solution efficace dont le vecteur de poids générateur appartient à l'ensemble formé par les 3 intervalles ci-dessus, par exemple la solution trouvée avec la méthode de Zionts-Wallenius.

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : EXES
S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : CONT
DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : NUTRI
LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : O
```

Phase 2

Dialogue décideur-machine

```
CHOIX DU MODULE (N,Z,G,S,T OU ?) ? : S
 DONNEZ LA BORNE INF. ET LA BORNE SUP. DE L'INTERVALLE NO 1 : 0.9 1.0
 DONNEZ LA BORNE INF. ET LA BORNE SUP. DE L'INTERVALLE NO 2 : 0.02 0.03
 DONNEZ LA BORNE INF. ET LA BORNE SUP. DE L'INTERVALLE NO 3 : 0.001 0.015
```

****
ITERATION No 1
****

	X1	X4	X6	X7	X10	X11	X12	X13	X14	X15 *	F1(X)	F2(X)	F3(X)
Y1	2.377	9.633	4.000	711.111	19.860	2.623	1.000	0.250	0.367	10.000	-2.538	-23.745	-409.531

	X1	X4	X6	X7	X8	X10	X11	X12	X13	X15 *	F1(X)	F2(X)	F3(X)
Y2	2.297	10.000	4.000	653.488	0.259	19.337	3.703	1.000	0.250	10.000	-2.557	-22.965	-413.116

	X1	X2	X3	X5	X7	X10	X11	X12	X14	X16 *	F1(X)	F2(X)	F3(X)
Y3	5.122	0.661	0.250	10.000	1868.551	150.499	0.878	0.339	10.000	4.000	-3.307	-94.439	-151.773

	X1	X3	X4	X5	X7	X10	X11	X12	X13	X16 *	F1(X)	F2(X)	F3(X)
Y4	3.961	0.205	10.000	10.000	644.424	51.302	2.039	1.000	0.045	4.000	-2.555	-64.228	-256.069

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16 *	F1(X)	F2(X)	F3(X)
Y5	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

VOULEZ VOUS CONTINUER ? (O/N): O

DONNEZ L'INDICE DE LA SOLUTION CHOISIE ? : 5

****
ITERATION No 2
****

	X1	X3	X4	X6	X7	X10	X11	X12	X15	X16 *	F1(X)	F2(X)	F3(X)
Y1	3.783	0.250	10.000	0.786	279.848	52.737	2.217	1.000	10.000	3.214	-2.256	-67.829	-281.471

	X1	X4	X6	X7	X10	X11	X12	X13	X15	X16 *	F1(X)	F2(X)	F3(X)
Y2	2.467	10.000	3.755	531.298	21.430	3.933	1.000	0.250	10.000	0.285	-2.532	-24.671	-408.896

	X1	X2	X3	X5	X7	X10	X11	X12	X14	X16 *	F1(X)	F2(X)	F3(X)
Y3	5.122	0.661	0.250	10.000	1868.551	150.499	0.878	0.239	10.000	4.000	-3.307	-94.439	-151.773

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16 *	F1(X)	F2(X)	F3(X)
Y4	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

VOULEZ VOUS CONTINUER ? (O/N): O

DONNEZ L'INDICE DE LA SOLUTION CHOISIE ? : 4

****
ITERATION No 3
****

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16 *	F1(X)	F2(X)	F3(X)
Y1	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

	X1	X3	X4	X5	X7	X10	X11	X12	X13	X16 *	F1(X)	F2(X)	F3(X)
Y2	3.961	0.305	10.000	10.000	644.424	51.302	2.039	1.000	0.045	4.000	-2.555	-64.228	-256.069

	X1	X2	X3	X5	X7	X10	X11	X12	X14	X16 *	F1(X)	F2(X)	F3(X)
Y3	5.122	0.661	0.250	10.000	1868.551	150.499	0.878	0.239	10.000	4.000	-3.307	-94.439	-151.773

	X1	X3	X4	X5	X7	X10	X11	X12	X14	X16 *	F1(X)	F2(X)	F3(X)
Y4	4.000	0.250	9.200	10.000	990.000	53.500	2.000	1.000	0.800	4.000	-2.520	-70.000	-245.000

VOULEZ VOUS CONTINUER ? (O/N): O

DONNEZ L'INDICE DE LA SOLUTION CHOISIE ? : 1

****
ITERATION No 4
****

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16 *	F1(X)	F2(X)	F2(X)
Y1	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

	X1	X3	X4	X5	X7	X10	X11	X12	X14	X16 *	F1(X)	F2(X)	F3(X)
Y3	4.000	0.250	9.200	10.000	990.000	93.500	2.000	1.000	0.800	4.000	-2.520	-70.000	-245.000

VOULEZ VOUS CONTINUER ? (O/N): O

DONNEZ L'INDICE DE LA SOLUTION CHOISIE ? : 1

****
ITERATION No 5
****

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16 *	F1(X)	F2(X)	F2(X)
Y1	4.020	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

VOULEZ VOUS CONTINUER ? (O/N): N

VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N

Résultats

EXES.RES

A chaque itération nous générons 7 (2K+1) solutions efficaces, mais ne sont imprimées que les solutions différentes.

METHODE DE STEUER

```
INTERVALLES DE POIDS  ]0.9 , 1[   ]0.02 , 0.03[   ]0.001 , 0.015[

Max  - 0.225 X1 - 2.200 X2 - 0.800 X3 - 0.100 X4 - 0.050 X5 - 0.260 X6
Max  - 10.0 X1 - 20.0 X2 - 120.0 X3
Max  - 24.0 X1 - 27.0 X2 - 15.0 X4 - 1.100 X5 - 52.0 X6
     720.0 X1 + 107.0 X2 + 7080.0 X3 + 134.0 X5 + 1000.0 X6 >= 5000
     0.200 X1 + 10.100 X2 + 13.200 X3 + 0.750 X4 + 0.150 X5 + 1.200 X6 >= 12.500
     344.0 X1 + 460.0 X2 + 1040.0 X3 + 75.0 X4 + 17.400 X5 + 240.0 X6 >= 2500
     18.0 X1 + 151.0 X2 + 78.0 X3 + 2.500 X4 + 0.200 X5 + 4.0 X6 >= 63
     X1 <= 6
     X2 <= 1
     X3 <= 0.250
     X4 <= 10
     X5 <= 10
     X6 <= 4

**************
ITERATION No 1
**************
```

	X1	X4	X6	X7	X10	X11	X12	X13	X14	X15 *	F1(X)	F2(X)	F3(X)
Y1	2.377	9.633	4.000	711.111	19.860	3.433	1.000	0.250	0.367	10.000	-2.538	-23.765	-409.531

	X1	X4	X6	X7	X8	X10	X11	X12	X13	X15 *	F1(X)	F2(X)	F3(X)
Y2	2.297	10.000	4.000	653.488	0.259	19.337	3.703	1.000	0.250	10.000	-2.557	-22.965	-413.116

	X1	X2	X3	X5	X7	X10	X11	X12	X14	X16 *	F1(X)	F2(X)	F3(X)
Y3	5.122	0.661	0.250	10.000	1868.551	150.499	0.878	0.339	10.000	4.000	-3.307	-94.439	-151.773

	X1	X3	X4	X5	X7	X10	X11	X12	X13	X16 *	F1(X)	F2(X)	F3(X)
Y4	3.961	0.305	10.000	10.000	644.424	51.303	2.039	1.000	0.045	4.000	-2.555	-64.228	-256.069

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16	F1(X)	F2(X)	F3(X)
Y5	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

****
ITERATION No 2
****

	X1	X2	X4	X6	X7	X10	X11	X12	X15	X16	F1(X)	F2(X)	F3(X)
Y1	3.783	0.250	10.000	0.766	279.660	52.737	2.217	1.000	10.000	3.214	-3.256	-47.829	-241.471

	X1	X4	X6	X7	X10	X11	X12	X13	X15	X16	F1(X)	F2(X)	F3(X)
Y2	2.467	10.000	3.755	531.798	31.430	3.533	1.000	0.250	10.000	0.345	-2.532	-24.671	-404.496

	X1	X2	X3	X5	X7	X10	X11	X12	X14	X16	F1(X)	F2(X)	F3(X)
Y3	5.122	0.461	0.350	10.000	1040.551	150.499	0.478	0.339	10.000	4.000	-3.307	-94.439	-151.773

	X1	X3	X4	X5	X7	X10	X11	X13	X15	X16	F1(X)	F2(X)	F3(X)
Y4	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-3.405	-70.299	-253.274

****
ITERATION No 3
****

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16	F1(X)	F2(X)	F3(X)
Y1	4.030	0.350	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

	X1	X3	X4	X5	X7	X10	X11	X12	X13	X16	F1(X)	F2(X)	F3(X)
Y2	3.961	0.305	10.000	10.000	444.434	51.303	2.039	1.000	0.045	4.000	-2.555	-64.234	-356.069

	X1	X2	X3	X5	X7	X10	X11	X12	X14	X16	F1(X)	F2(X)	F3(X)
Y2	5.122	0.641	0.250	10.000	1046.551	150.499	0.478	0.339	10.000	4.000	-3.307	-94.439	-151.773

	X1	X2	X4	X5	X7	X10	X11	X12	X14	X16	F1(X)	F2(X)	F3(X)
Y4	4.000	0.250	9.300	10.000	990.000	53.500	2.000	1.000	0.400	4.000	-2.520	-70.000	-245.000

****
ITERATION No 4
****

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16 =	F1(X)	F2(X)	F3(X)
Y1	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

	X1	X3	X4	X5	X7	X10	X11	X12	X14	X16 =	F1(X)	F2(X)	F3(X)
Y3	4.000	0.250	9.200	10.000	990.000	53.500	2.000	1.000	0.800	4.000	-2.520	-70.000	-245.000

****
ITERATION No 5
****

	X1	X3	X4	X5	X7	X10	X11	X12	X15	X16 =	F1(X)	F2(X)	F3(X)
Y1	4.030	0.250	10.000	5.960	470.199	55.231	1.970	1.000	4.040	4.000	-2.405	-70.299	-253.274

EXES.TRA

Pour chaque itération, nous avons les différentes solutions efficaces avec les valeurs des fonctions objectifs et la solution qui a été choisie par le décideur à cette itération.

METHODE DE STEUER

****
ITERATION No 1
****

NOUVELLE SOLUTION

X1 = 2.36 X4 = 9.63 X6 = 4.00 X7 = 711. X10 = 19.9

X11 = 3.62 X12 = 1.00 X13 = 0.250 X14 = 0.367 X15 = 10.0

VALEURS DES FONCTIONS OBJECTIFS

-2.5380 -23.7654 -409.5308

NOUVELLE SOLUTION

X1 = 2.30 X4 = 10.0 X6 = 4.00 X7 = 653. X8 = 0.259
X10 = 19.3 X11 = 3.70 X12 = 1.00 X13 = 0.250 X15 = 10.0

VALEURS DES FONCTIONS OBJECTIFS

-2.5567 -22.9651 -413.1163

NOUVELLE SOLUTION

X1 = 5.12 X2 = 0.661 X3 = 0.250 X5 = 10.0 X7 = 0.187B+04
X10 = 150. X11 = 0.876 X12 = 0.339 X14 = 10.0 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-3.3065 -94.4389 -151.7733

NOUVELLE SOLUTION

X1 = 3.96 X3 = 0.205 X4 = 10.0 X5 = 10.0 X7 = 644.
X10 = 51.3 X11 = 2.04 X12 = 1.00 X13 = 0.449E-01 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.5554 -64.2282 -256.0694

NOUVELLE SOLUTION

X1 = 4.03 X3 = 0.250 X4 = 10.0 X5 = 5.96 X7 = 470.
X10 = 55.2 X11 = 1.97 X12 = 1.00 X15 = 4.04 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.4047 -70.2993 -253.2743

LA SOLUTION Y5 A ETE CHOISIE PAR LE DECIDEUR

****
ITERATION No 2
****

NOUVELLE SOLUTION

X1 = 3.78 X3 = 0.250 X4 = 10.0 X6 = 0.786 X7 = 280.
X10 = 52.7 X11 = 2.22 X12 = 1.00 X15 = 10.0 X16 = 3.21

VALEURS DES FONCTIONS OBJECTIFS

-2.2556 -67.8290 -281.6710

NOUVELLE SOLUTION

X1 = 2.47 X4 = 10.0 X6 = 3.76 X7 = 532. X10 = 21.4
X11 = 3.53 X12 = 1.00 X13 = 0.250 X15 = 10.0 X16 = 0.245

VALEURS DES FONCTIONS OBJECTIFS

-2.5315 -24.6711 -404.4956

NOUVELLE SOLUTION

X1 = 5.12 X2 = 0.661 X3 = 0.250 X5 = 10.0 X7 = 0.187E+04
X10 = 150. X11 = 0.878 X12 = 0.339 X14 = 10.0 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-3.3065 -94.4389 -151.7733

NOUVELLE SOLUTION

X1 = 4.03 X3 = 0.250 X4 = 10.0 X5 = 5.96 X7 = 470.
X10 = 55.2 X11 = 1.97 X12 = 1.00 X15 = 4.04 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.4047 -70.2993 -253.2743

LA SOLUTION Y4 A ETE CHOISIE PAR LE DECIDEUR

****
ITERATION No 3
****

NOUVELLE SOLUTION

X1 = 4.03 X3 = 0.250 X4 = 10.0 X5 = 5.96 X7 = 470.
X10 = 55.2 X11 = 1.97 X12 = 1.00 X15 = 4.04 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.4047 -70.2993 -253.2743

NOUVELLE SOLUTION

X1 = 3.96 X3 = 0.205 X4 = 10.0 X5 = 10.0 X7 = 644.

VALEURS DES FONCTIONS OBJECTIFS

-2.5554 -64.2282 -256.0694

NOUVELLE SOLUTION

X1 = 5.12 X2 = 0.661 X3 = 0.250 X5 = 10.0 X7 = 0.187E+04
X10 = 150. X11 = 0.878 X12 = 0.339 X14 = 10.0 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-3.3065 -94.4389 -151.7733

NOUVELLE SOLUTION

X1 = 4.00 X3 = 0.250 X4 = 9.20 X5 = 10.0 X7 = 990.

X10 = 53.5 X11 = 2.00 X12 = 1.00 X14 = 0.800 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.5200 -70.0000 -245.0000

LA SOLUTION Y1 A ETE CHOISIE PAR LE DECIDEUR

****
ITERATION No 4
****

NOUVELLE SOLUTION

X1 = 4.03 X3 = 0.250 X4 = 10.0 X5 = 5.96 X7 = 470.

X10 = 55.2 X11 = 1.97 X12 = 1.00 X15 = 4.04 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.4047 -70.2993 -253.2743

NOUVELLE SOLUTION

X1 = 4.00 X3 = 0.250 X4 = 9.20 X5 = 10.0 X7 = 990.

X10 = 53.5 X11 = 2.00 X12 = 1.00 X14 = 0.800 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.5200 -70.0000 -245.0000

LA SOLUTION Y1 A ETE CHOISIE PAR LE DECIDEUR

****
ITERATION No 5
****

NOUVELLE SOLUTION

X1 = 4.03 X3 = 0.250 X4 = 10.0 X5 = 5.96 X7 = 470.

X10 = 55.2 X11 = 1.97 X12 = 1.00 X15 = 4.04 X16 = 4.00

VALEURS DES FONCTIONS OBJECTIFS

-2.4047 -70.2993 -253.2743

5.2.7 La méthode STEM

Pour cette méthode, nous donnerons un exemple de production où 2 objectifs sont considérés.

Exemple : Plan de production de 2 types de poupées (voir [9] page 23).

La Hardee Toy Company fabrique 2 types de poupées : A et B.
Le type A est de meilleure qualité que le type B. Le bénéfice net est \$ 0.40 pour les poupées de type A et \$ 0.30 pour les poupées de type B. Le temps de fabrication d'une poupée de type A est le double du temps de fabrication d'une poupée de type B et si toutes les poupées étaient du type B, l'entreprise pourrait en fabriquer 500 par jour. L'approvisionnement en matériel est suffisant pour 400 poupées par jour (type A ou B). On aimerait trouver les quantités respectives de poupées des deux types à fabriquer par jour de manière à maximiser le bénéfice et à maximiser la production des poupées de type A. Ce programme se formule de la façon suivante :

$$\text{Max } 0.4x_1 + 0.3x_2$$

$$\text{Max } x_1$$

Sous les contraintes

$$x_1 + x_2 \leqslant 400$$

$$2x_1 + x_2 \leqslant 500$$

$$x_1 \, , \, x_2 \geqslant 0$$

Phase 0

```
NOM DU PASSAGE (SERVIRA DE NOM AUX FICHIERS DE RESULTATS) : EXET
S'AGIT-IL D'UN CAS DISCRET OU CONTINU (DISC/CONT) ? : CONT
DONNEZ LE NOM DU FICHIER DES DONNEES (MAX. 6 CARACT.) : PROD
LE FICHIER DES DONNEES EXISTE T-IL DEJA (O/N) ? : N
```

Phase 1

Entrée des données

```
NOMBRE DE FONCTIONS OBJECTIFS ?: 2

NOMBRE DE CONTRAINTES ?: 2

NOMBRE DE VARIABLES ?: 2

LE PROBLEME COMPORTE T'IL DES FONCTIONS  OBJECTIFS NON LINEAIRES (O/N) ? : N

DONNEZ LES COEFFICIENTS DE LA FONCTION OBJECTIF
NUMERO   1 ?: 0.4 0.3

DONNEZ LES COEFFICIENTS DE LA FONCTION OBJECTIF
NUMERO   2 ?: 1 0

DONNEZ LES ELEMENTS DE LA LIGNE  1
DE LA MATRICE DES CONTRAINTES A ?: 1 1

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 400

DONNEZ LES ELEMENTS DE LA LIGNE  2
DE LA MATRICE DES CONTRAINTES A ?: 1 2

TYPE DE L'INEGALITE 1 SI <=, -1 SI >=, 0 SI = : 1

VALEUR DU MEMBRE DE DROITE : 500
```

Fichier de données PROD.CNT

```
    2     2     2  (* NB DE CRITERES,NB DE CONTRAINTES,NB DE VARIABLES *)
FONCTIONS OBJECTIFS :
0 (* 0 = TOUTES LES FONCT. OBJ. SONT LINEAIRES,1 = AU MOINS UNE FONCT. OBJ. EST NON LINEAIRE *)
0.400      0.300
 1.00      0.000E+00
CONTRAINTES :
 1.00       1.00       1.  0.40E+03
 2.00       1.00       1.  0.50E+03
```

Phase 2

Dialogue décideur-machine

```
CHOIX DU MODULE (N,Z,G,S,T OU ?) ? : T
 VALEURS DES FONCTIONS OBJECTIFS :
        104.000  230.000
VOULEZ VOUS LE TABLEAU DE GAIN ? (O/N) : O
           TABLEAU DE GAIN
           ***************

           130.000  100.000
           100.000  250.000
ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N) : N
DONNEZ LE NOMBRE DE CRITERES A RELACHER : 1
DONNEZ L'INDICE DU 1e CRITERE A RELACHER : 2
DONNEZ LA VALEUR DU RELACHEMENT : 30
 VALEURS DES FONCTIONS OBJECTIFS :
        110.000  200.000
VOULEZ VOUS LE TABLEAU DE GAIN ? (O/N) : N
ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N) : N
DONNEZ LE NOMBRE DE CRITERES A RELACHER : 1
DONNEZ L'INDICE DU 1e CRITERE A RELACHER : 2
DONNEZ LA VALEUR DU RELACHEMENT : 10
 VALEURS DES FONCTIONS OBJECTIFS :
        112.000  190.000
VOULEZ VOUS LE TABLEAU DE GAIN ? (O/N) : N
ETES VOUS SATISFAIT DE CETTE SOLUTION ? (O/N) : O
VOULEZ VOUS UN AUTRE TRAITEMENT SUR LES MEMES DONNEES (O/N) ? : N
```

Résultats

EXET.RES

METHODE S.T.E.M

Max 0.400 X1 + 0.300 X2

Max X1

X1 + X2 <= 400

2.0 X1 + X2 <= 500

	X1	X2	X3	X4	* F1(X)	F2(X)
Y0	230.000	40.000	11.304	130.000	104.000	230.000

	X1	X2	X3	X4	X8	* F1(X)	F2(X)
Y1	200.000	100.000	20.000	100.000	6.000	110.000	200.000

	X1	X2	X3	X4	X8	* F1(X)	F2(X)
Y2	190.000	120.000	18.000	90.000	2.000	112.000	190.000

EXET.TRA

METHODE S.T.E.M

SOLUTION INITIALE

X1 = 230. X2 = 40.0 X3 = 11.3 X4 = 130.

VALEURS DES FONCTIONS OBJECTIFS

104.0000 230.0000

LE CRITERE 2 A ETE RELACHE DE : 30.0

NOUVELLE SOLUTION

X1 = 200. X2 = 100. X3 = 20.0 X4 = 100. X8 = 6.00

VALEURS DES FONCTIONS OBJECTIFS

110.0000 200.0000

LE CRITERE 2 A ETE RELACHE DE : 10.0

NOUVELLE SOLUTION

X1 = 190. X2 = 120. X3 = 18.0 X4 = 90.0 X8 = 2.00

VALEURS DES FONCTIONS OBJECTIFS

112.0000 190.0000

REFERENCES BIBLIOGRAPHIQUES

LIVRES ET CONFERENCES PROCEEDINGS

[1] ARROW,K.J., HURWICZ,L., et H.,UZAWA, Studies in Linear and Non-Linear Programming, Univ. Press, 1972.

[2] BAZARAA,M.S., et C.M.,SHETTY, Nonlinear Programming : Theory and Algorithms, J. Wiley, New York, 1979.

[3] BELL,D.E., KEENEY,R.L., et H.,RAIFFA (Eds.), Conflicting Objectives in Decision, J. Wiley, New York, 1977.

[4] COCHRANE,J.L., et M.,ZELENY (Eds.), Multiple Criteria Decision Making, University of South Carolina Press, Columbia, South Carolina, 1973.

[5] FANDEL,G., et T.,GAL (Eds.), Multiple Criteria Decision Making : Theory and Application, Springer-Verlag, Berlin, 1979.

[5a]GOICOECHEA,A.,HANSEN,D.R., et DUCKSTEIN,L., Multiobjective decision analysis with engineering and business applications, John Wiley and Sons, New York, 1982.

[6] GROS,C., Approche de la dualité en optimisation multicritère, C.C.E.R.O., 1980, 22, 73-79.

[7] HARVEY,C., Operation Research : An Introduction to Linear Optimization and Decision Analysis, North Holland, New York, 1979.

[8] HIRSCH,G., Aide à la Dècision Multicritère, Ecole Sup. de Commerce, Paris, 1980.

[9] HWANG,C.L., et A.S.M.,MASUD (Eds.), Multiple Objective Decision Making : Methods and Applications, Springer-Verlag, Berlin, 1979.

[10] HWANG,C.L., et K.,YOON, Multiple Attribute Decision Making : Methods and Applications, Springer-Verlag, Berlin, 1981.

[11] IGNIZIO,J.P., Linear Programming in Single and Multiple Objective Systems, Prentice-Hall, London, 1982.

[12] IGNIZIO,J.P., Goal Programming and Extensions, Lexington Books, Massachusetts, 1976.

[13] KEENEY,R.L., et H., RAIFFA, Decision with Multiple Objectives : Preferences and Value Tradeoffs, Wiley, New York, 1976.

[14] LAND,A., et S., POWELL, Fortran Codes for Mathematical Programming, J. Wiley, New York, 1979.

[15] LEITMANN,G. (Eds.), Multicriteria Decision Making and Differential Games, Plenum Press, New York, 1976.

[16] LEITMANN,G., et A., MARZOLLO (Eds.), Multicriteria Decision Making, Springer-Verlag, Wien, 1975.

[17] MACMILLAN,C., Mathematical Programming, J. Wiley, New York, 1975.

[18] MENUET,J., Quasi-Ordre et Modelisation des Preferences, Sema, Montrouge, France, 1974.

[19] J.DE MONTGOLFIER, et P., BERTIER, Approche Multicritère des Problèmes de Décision, Hommes et Techniques, Suresnes, France, 1978.

[20] RAIFFA,H., Analyse de la Décision, Dunod, Paris, 1973.

[21] RIETVELD,P., Multiple Objective Decision and Regional Planning, North Holland, Amsterdam, 1980.

[22] ROY,B., Critères Multiples et Modélisation des Préférences, Sema, Montrouge, France, 1973.

[23] ROY,B., Vers une Méthodologie Générale d'Aide à la Décision, Sema, Montrouge, France, 1975.

[24] ROY,B., Management Scientifique et Aide à la Décision, Sema, Montrouge, France, 1974.

[25] SCHITTKOWSKI,K., Nonlinear Programming Codes, Springer-Verlag, Berlin, 1980.

[26] SHAPIRO,J., Mathematical Programming : Structures and Algorithms, J. Wiley, New York, 1979.

[27] STARR,M.K., et M.,ZELENY (Eds.), Multiple Criteria Decision Kaking, North Holland, New York, 1977.

[28] THIRIEZ,H., et S.,ZIONTS (Eds.), Multiple Criteria Decision Making, Jouy-en-Josas, France, Springer-Verlag, New York, 1976.

[29] ZELENY,M., Multiple Criteria Decision Making, Mcgraw-Hill, New-York, 1982.

[30] ZELENY,M., Linear Multiobjective Programming, Springer-Verlag, New York, 1974.

[31] ZELENY,M. (Eds.), Multiple Criteria Decision Making, Kyoto, 1975, Springer-Verlag, New York, 1976.

[32] ZIONTS,S. (Eds.), Multiple Criteria Problem Solving : Proceedings, Buffalo,New York, Springer-Verlag, New York, 1978.

ARTICLES

[33] APPA,G., et C.,SMITH, On L1 and Chebyshev Estimation, Mathematical Programming, Vol. 5, 1973.

[34] BELENSON,S.M., et K.C.,KAPUR, An Algorithm for Solving Multicriterion Linear Programming Problems with Examples, Operational Research Quarterly, 1973, 24, 65-77.

[35] BENAYOUN,R., et J.,TERGNY, Critères Multiples en Programmation Mathématique : une solution dans le cas linéaire, Revue Française d'Informatique et de Recherche Opérationnelle, 1969, 2, 31-56.

[36] BITRAN,G.R., Linear Multiple Objective Problems with Interval Coefficients, Management Science, 1980, 7, 694-706.

[37] CHARNES,A., et W.W.,COOPER, Goal Programming and Multiple Objective Optimizations, Europan Journal of Operational Research, 1977, 1, 39-54.

[38] ECKER,J.G., et I.A.,KOUADA, Finding all efficient extreme points for multiple objective linear programs, Mathematical Programming, vol. 14, 1978.

[39] EVANS,J.P., et R.E.,STEUER, A revised simplex method for linear multiple objective programs, Mathematical Programming, Vol. 5, 1973.

[40] FICHEFET,J., Some insight into the nature of computer selection, Revue Belge de Statistique d'Informatique et de Recherche Opérationnelle, Vol. 21, No 3.

[41] FREIMER,M., et P.L.,YU, Some new results on compromise solutions for group decision problems, Management Science, Vol. 22, 1976.

[42] GAL,T., A General Method for Determining the Set of all Efficient Solutions to Linear Vectormaximum Problem, European Journal of Operational Research, 1977, 1, 307-322.

[43] GAL,T., Rim Multiparametric Linear Programming, Management Science, 1975, 21, 567-575.

[44] GAL,T., et J.,NEDOMA, Multiparametric Programming, Management Science, 1972, 18, 406-422.

[45] GEOFFRION,A.M., J.S.,DYER, and A.,FEINBERG, An Interactive pproach for Multi-Criterion Optimization with An Application to the Operation of Academic Department. Management Science, 1972, 4, 357-368.

[46] HAIMES,Y.Y., et W.A.,HALL, Multiobjectives in Water Resource Systems Analysis : the Surrogate Worth Trade off Method, Water Resources Research, 1974, 615-624.

[47] ISERMANN,H., The Enumeration of all Efficient Solutions for a linear Multiple-Objective Transportation Problem, Nav. Res. Logist. Q., 1979, 1, 123-239.

[48] ISERMANN,H., On Some Relations between a Dual Pair of Multiple Objective Linear Programs, Z. Oper. Res. Ser. A., 1978, 1, 33-41.

[49] KORNBLUTH,J.S.H., Duality, Indifference and Sensitivity Analysis in Multiple Objective Linear Programming, Operational Research Quarterly, 1974, 25, 599-614.

[50] MUKAI,H., Algorithms for Multicriterion Optimization, IEEE Trans. Autom. Control, 1980, 2, 177-186.

[51] NAKAYAMA,H., T.,TANINO, and Y.,SAWARAGI, An Interactive Optimization Method in Multicriteria Decision Making, IEEE Transactions on Systems, Man and Cybernetics, 1980, 10, 163-169.

[52] RODDER,W., A Generalized Saddle Point Theory : Its Application to Duality Theory for Linear Vector Optimum Problems, European Journal of Operational Research, 1977, 1, 55-59.

[53] SAKAWA,M., Multiobjective Optimization by the Surrogate Worth Trade off Method, IEEE Trans. Reliab., 1978, 5, 311-314.