République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université A. Mira de Béjaïa

Faculté des Sciences Exactes

Département de Recherche Opérationnelle

Mémoire de Master

en

Mathématiques appliquées

spécialité :Modélisation Mathématique et Evaluation de Performance des Réseaux

Thème

Optimisation sur l'ensemble efficace

présenté par :

DJAOUZI Abdelhak

devant le jury composé de :

Mlle. SAIT Razika	Présidente
M. TOUATI Sofine	Examinateur
M. ZIANI Sofiane	Examinateur
Mlle. AOUDIA Zohra	Encadreur
Mme. AOUDIA Fazia	Encadreur

Béjaia, 2020/2021

Table des matières

Liste des tableaux

Table des figures

Introduction générale

De nombreux secteurs de l'industrie (mécanique, chimie, télécommunications, environnement, transport, etc.) sont concernés par des problèmes complexes de grandes dimensions mettant en jeu des objectifs (critères) très importants (profit, coût, délai, qualité de service, etc.) à atteindre et pour lesquels les décisions doivent être prises de façon rationnelle et logique. La modélisation traditionnelle n'a connu que des modèles d'optimisation à objectif unique jusqu'à l'apparition d'une nouvelle vision plus proche de la réalité et qui consistent à modéliser les problèmes de sorte que tous les objectifs pertinents soient pris en considération. Cette philosophie de multiplicité d'objectifs semble très raisonnable et contribue éfficacement à l'élaboration de nouveaux systèmes d'aide à la décision, qui vise à faciliter la tâche des décideurs devant des choix complexes, où ces derniers s'avèrent souvent être conflictuels. Dans la pratique, une classe très importante de problèmes peuvent être modélisés sous forme de programmes linéaires multiobjectifs (Multiobjective Objective Lineair Programming)(MOLP) où les critères et les contraintes sont tous linéaires et les variables de décision peuvent prendre des valeurs continues, discrètes ou binaires. Cette famille de modèle possède un héritage très riche accumulé par la programmation mathématique monocritére, considérablement développée depuis une cinquantaine d'années, lorsque G. B. Dantzig proposa l'algorithme du simplexe. L'optimisation linéaire multiobjectif possède ses racines aux 19 ièmes siècle dans les travaux en économie d'Edgeworth et Pareto (1896). Elle a été utilisée initialement en économie et dans les sciences de gestion, et graduellement appliquée aux sciences de l'ingénieur. Contrairement à l'optimisation classique mono-objectif, la solution d'un problème multi-objectif n'est pas une solution optimale, mais un ensemble de solutions, connues comme étant ensemble des solutions Pareto Optimales ou bien solutions efficaces, offrant un bon compromis entre les différentes fonctions objectives à optimiser, à partir de lesquelles, il est impossible d'augmenter la valeur d'un objectif sans diminuer au moins celle d'un autre.

Dans de nombreuses situations réelles modélisables par la programmation linéaire multiobjectif, les variables de décision ne peuvent prendre que des valeurs entières, dans ce cas on parle, de programmation linéaire multiobjectif en nombres entiers (MOILP) où le domaine réalisable n'est plus convexe, ce genre de problème appartient à la classe de l'optimisation non convexe. L'importance de cette famille de problèmes d'optimisation a poussé Rokafellar à dire :"The great watershed in optimisation is not between linearity ou non-linearity but between convexity and nonconvexity". Une préoccupation majeure dans la mise en oeuvre des systèmes multicritères d'aide à la décision est la définition d'un formalisme permettant de refléter fidèlement les choix du décideur. Dans beaucoup de cas, les critères sont en conflit, sans compter la taille des calculs informatiques impliqués dans les algorithmes, ainsi que la taille de l'ensemble des solutions efficaces, qui s'averent habituellement considérable ceci tend à saturer le décideur jusqu'au point où le choix de sa solution préférée devient une mission impossible. En effet, l'énumération de tout l'ensemble des solutions efficaces n'est pas appropriée car, les décideurs ne sont pas souvent intéressés par toutes les solutions efficaces mais d'un sous-ensemble de solutions répondant à certaines préférences. Par conséquent, le problème de l'optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces qui est aujourd'hui, l'un des concepts importants et intéressants de la programmation multiobjectifs, Il semble être un moyen fructueux pour éviter de telles préoccupations pour enfin mesurer les préférences des décideurs, et distinguer parmi les nombreuses solutions efficaces. Dans ce travail on s'intéresse à optimiser une fonction linéaire sur un ensemble efficace. Alors, le premier chapitre comportera l'optimisation mono-objectif, programmation linéaire, programmation linéaire en nombre entier, et quelques méthodes de résolution, Le deuxième chapitre on parlera sur l'optimisation multiobjectif et les définitions des points admissibles et quelques méthodes de résolution Le troisième chapitre on va difinire en premier lieu l'optimisation biniveau qui généralise l'optimisation d'une fonction sur un ensemble efficace et en finale en va présenter les deux méthodes de résolution pour l'optimisation d'une fonction sur un ensemble efficace dans le cas discret

Chapitre 1

Optimisation mono objectif

Introduction

Dans le présent chapitre, nous nous intéresserons principalement à des concepts et des théorèmes de base d'optimisation mono-objectif : la programmation linéaire, la programmation linéaire en nombres enties, et l'optimisation Globale ainsi que quelques méthodes de résolution

1.1 Formulaion

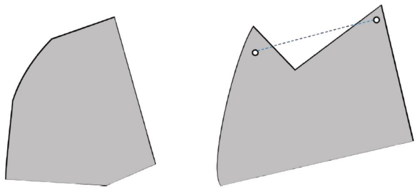
Lorsqu'un seul objectif (critère) est donné, le problème d'optimisation est mono-objectif. On dit que l'on cherche à minimiser ou maximiser une fonction différentiable $f : \mathbb{R}^n \to \mathbb{R}$ sur un ensemble réalisable $C = \{x \in \mathbb{R}^n : h_i(x) = 0,\ \forall i = \overline{1,m}\ ;\ g_k(x) \leq 0,\ \forall k = \overline{1,p}\}$ avec h_i et g_k de classe C^1.[12]
L'optimisation est divisée en sous-disciplines qui se chevauchent, suivant la forme de la fonction objectif et celle des contraintes.
Par exemple lorsque f est une fonction convexe et l'ensemble C convexe, alors on a affaire à L'optimisation convexe qui généralise l'optimisation linéaire.

1.2 Notion de convexité

La convexité joue un rôle majeur dans la théorie de l'optimisation. Elle est un outil indispensable et efficace pour la recherche des conditions à la fois nécessaires et suffisantes d'optimalité, [4]
Définition 1.2.1 (Ensemble convexe) Un ensemble C de $\mathbb{R}^n$ est dit convexe, si

$$\forall x_1, x_2 \in C,\ \forall \lambda \in [0,1],\ [\lambda x_1 + (1-\lambda)x_2] \in C,$$

Cette définition signifie qu'un ensemble C est convexe si le segment joignant deux quelconques de ses points quelconques est contenu dans l'ensemble C.



(a) Ensemble convexe. (b) Ensemble non convexe.

FIGURE 1.1 – Ensembles convexe et non convenxe.

Définition 1.2.2 (Fonction convexe). Une fonction réelle f définie sur un convexe $C \subset \mathbb{R}^n$, est dite convexe, si pour tous les points $x_1, x_2 \in C$, et pour tout nombre réel positif ou nul, tel que $0 \leq \lambda \leq 1$, l'inégalité suivante est vérifiée :

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

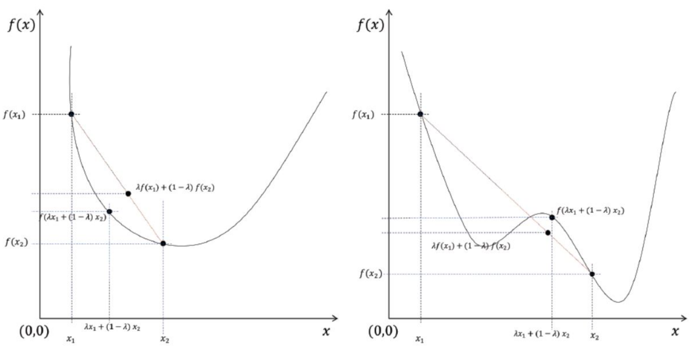


FIGURE 1.2 – Fonctions convexe et non convexe.

Définition 1.2.3 (Fonction strictement convexe) Une fonction réelle f définie sur un convexe $C \subset \mathbb{R}^n$, est dite strictement convexe, si pour tous les points $x_1, x_2 \in C$, avec $x_1 \neq x_2$ et pour tout nombre réel positif ou nul, tel que $0 < \lambda < 1$, l'inégalité stricte suivante est vérifiée :

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

Théorème 1.2.4 (Fonction concave) f est dite concave si $(-f)$ est convexe.

1.3 Programmation linéaire

En mathématiques, les problèmes de programmation linéaire (PL) sont des problèmes d'optimisation où la fonction objectif et les contraintes sont toutes linéaires. La programmation linéaire désigne également la manière de résoudre les problèmes de (PL).
La programmation linéaire est un domaine central de l'optimisation, car les problèmes de (PL) sont les problèmes d'optimisation les plus faciles et que beaucoup de problèmes réels de recherche opérationnelle peuvent être exprimés comme un problème de (PL). De plus un grand nombre d'algorithmes pour la résolution d'autres problèmes d'optimisation sont fondés sur la résolution de problèmes linéaires.
Sa forme générale est donnée par :

$$\begin{aligned} \max\ z = c^T x \\ \text{s.c } Ax = b \\ x \geq 0 \end{aligned}$$

où $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m, c \in \mathbb{R}^n$
La matrice A décrit l'ensemble des contraintes et restrictions du problèmes, son rang est supposé égal à m, le vecteur b représente le second membre des contraintes, le vecteur x désigne l'ensemble des variables de décision, $z = c^T x$ représente la fonction objectif et les composantes du vecteur c sont appelés des coûts ou coefficients économiques. Pour plus de détails nous invitons les lecteurs à consulter la référence [15] et [29]

1.4 Programmation linéaire en nombres entiers

Un problème de programmation linéaire en nombres entiers (PLNE) est un programme linéaire, c'est-à-dire une fonction objectif linéaire à maximiser, sous des contraintes linéaires, où les variables sont entières, donné par :

$$\begin{aligned} \max\ c^T x \\ \text{s.c } Ax \leq b \\ x \geq 0 \\ x \in \mathbb{Z} \end{aligned}$$

Lorsque seul un sous-ensemble de variables doivent être entières et les autres réelles, on parle alors de programme linéaire mixte (PLM) qui s'écrit comme suit :

$$\begin{aligned} \max\ (c^T x + h^T y) \\ \text{s.c } Ax + Gy \leq b \\ x \geq 0 \\ y \in \mathbb{Z}^+ \end{aligned}$$

Il est à noter que les problèmes de PLNE sont des problèmes NP-complets. Les algorithmes de résolution de PLNE, tels que les algorithmes par séparation et évaluation ou les algorithmes de génération de plans sécants, se basent très souvent sur une relaxation continue, voir [23] et [3]

1.5 Optimmisation globale

Le théme de l'optimisation globale étudie la question de trouver les extrema (locaux et globaux) d'une fonction d'une ou plusieurs variables. Il n'y a pas si longtemps, mettre au point des méthodes numériques permettant de déterminer la solution d'un probléme d'optimisation non linéaire, non convexe et non différentiable avec des contraintes, elles aussi, non linéaires et non convexes, paraissait presque impossible, du moins très difficile. Puisque la théorie classique de l'optimisation ne peut pas étre appliquée directement dans les problémes d'optimisation globale, les outils tels que l'analyse convexe, sont largement utilisés dans la construction des méthodes d'optimisation globale.
Cette approche constitue une partie importante de l'optimisation globale déterministe. Par exemple, un remarquable progrès a été accompli dans la construction des algorithmes de minimisation des fonctions concaves dans des régions convexes, et ainsi que la minimisation de fonction, s'écrivant sous forme de différence de deux fonctions convexes.

1.5.1 Quelques notions générales sur l'optimisation globale

Soit x un vecteur de dimension n dont les composantes x_i vérifient $a_i \leq x_i \leq b_i$, $i = \{1, ..., n\}$, où a_i et b_i sont les composantes de deux vecteurs a et b donnés, définissant un domaine hyperrectangulaire S dans R^n.
$Soit$f : $\mathrm{R}^n \longrightarrow R$, une fonction à valeurs réelles. Le problème considéré est celui de trouver un minimum global x^* de f c'est-à-dire :
$f(x^*) \leq f(x), \ \forall x \in S \subseteq \mathbb{R}^n$
Une des difficultés rencontrées dans la recherche d'un minimum global est la présence éventuelle de minimum locaux $\hat{x}$ vérifiant pour un nombre positif ε assez petit la propriété suivante :

$$\forall x \in S : \|\hat{x} - x\| < \varepsilon \Rightarrow f(\hat{x}) \leq f(x).$$

Noton qu'un problème de maximisation peut se traduire en terme de minimisation par :

$$\max\{f(x)\} = -\min\{-f(x)\}$$

Les méthodes déterministes pour l'optimisation globale s'appuient sur des conditions mathématiques précises afin de garantir une solution approchée assez precise, que les heuristiques donnent moins de certitude aux résultats, mais sont moins restrictives : elles requièrent seulement la capacité d'évaluer la fonction à optimiser en un point quelconque de l'espace des solutions. C'est pourquoi nous présentons deux hypothèses habituellement rencontrées, l'une concernant le domaine de recherche du minimum global et exprimant les contraintes sur les composantes de la solution, l'autre posant des conditions fortes sur la fonction objectif :

Hypothèse (H1) : l'hyperrectangle $S \subseteq R^n$ est convexe et compact.

Hypothèse (H2) :f est continue et possède des dérivées partielles premières $\frac{\partial f}{\partial x_i}$ et secondes $\frac{\partial^2 f}{\partial x_i \partial x_j}$ continues pour tout $x \in R^n$.

L'hypothèse (**H2**) est faite dans la plupart des méthodes déterministes utilisant une procédure de raffinement local. Sous (**H2**), des conditions nécessaires pour que x^* soit un minimum (local ou global) de f sont :

•$\nabla f(x^*) = 0$ (stationnarité) .

• le hessien $\nabla^2 f(x^*) = [\frac{\partial^2 f(x^*)}{\partial x_i \partial x_j}]$ est une matrice semi-définie positive c'est-à-dire

$$y^T \ \nabla^2 f(x^*) \ y \geq 0, \quad \forall y \in \mathbb{R}^n$$

Ces deux conditions reviennent à supposer que f est strictement convexe dans un voisinage de x^*. Dans le cas d'une fonction convexe, la stationnarité à elle seule constitue une condition nécessaire et suffisante d'optimalité globale. Puisqu'une procédure numérique ne peut pas fournir plus qu'une solution approchée. [9]

1.5.2 Minimisation concave

Ont définit un probléme de minimisation concave par :

$$\begin{gathered} \min f(x) \\ g_i(x) \leq 0 \ , \ i = 1, ..., m \\ x \in D \end{gathered}$$

ou $f : F \rightarrow \mathbb{R}$ est une fonction concave sur l'ensemble réalisable F qui est un ensemble non vide, fermé et convexe.

Une proprité importante de la minimisation concave est que la fonction objectif atteint toujours son minimum global en un point extrême de F

Remarque L'hypothèse de convexité apporte simplicité et efficacité à la théorie de l'optimisation. En particulier, les conditions nécessaires d'optimalité deviennent également suffisantes, et tout le résultat acquiert un caractère global.

1.6 Les méthodes de résolution

1.6.1 La méthode d'Approximation intérieure

L'algorithme de base Dans cette section, nous présentons une méthode d'approximation interne pour résoudre le problème de minimisation concave

$$\min f(x) \quad s.c \quad x \in D \tag{1.1}$$

Avec D est un polytope dans $\mathbb{R}^n$, et $D \neq \emptyset$, et f est une fonction concave sur un ensemble suffisamment grand $D \subset A$.
L'idée de base des méthodes d'approximation interne est de construire une séquence de polytopes $P_1, P_2, ...$ tel que :
(i) Pour $k \geq 1, (P_k \cap D) \subset (P_{k+1} \cap D)$.
(ii) une solution optimale w_1 de $\min\{f(x) : x \in P_1 \cap D\}$ est Disponible.
(iii) une solution optimale w_{k+1} de $\min\{f(x) : x \in P_{k+1} \cap D\}$ peut provenir d'une solution optimale w_k de $\min\{f(x) : x \in P_k \cap D\}$.
La procédure s'arrête lorsque $D \subseteq P_k$, c'est-à-dire $P_k \cap D = D$, car, dans ce cas w_k est évidemment une solution optimale de (1.2). La suite $P_1 \cap D, P_2 \cap D, ...$ constitue une approximation interne de D en développant le polytope, Le polytope P_{k+1} est généralement construit à partir de P_k en choisissant un point qui convient $z_k \notin P_k$ et en définissant .

$$P_{k+1} = co(P_k \cup \{z_k\}) \tag{1.2}$$

De toute évidence, la convergence finie de l'approche est assuré lorsque nous garantissons que $P_{k+1} \setminus P_k$ contient un sommet de D. Par conséquent, l'algorithme suivant détecte, à chaque itération, un sommet y_k de D, $y_k \notin P_k$, et détermine z_k en (1.3) à partir de y_k et d'autres informations disponibles de manière à ce que :

$$P_{k+1} = co(P_k \cup \{z_k\}) \subseteq co(P_k \cup \{y_k\}) \tag{1.3}$$

A l'exception de certaines situations particulières qui pourraient se produire, nous aurons $z_k \neq y_k$ de sorte que P_k sera strictement plus grand que $co(P_k \cup \{y_k\})$

Algorithme de la méthode d'Approximation intérieure

Initialisation :
Construire un polytope P satisfaisant $P \cap D \neq \emptyset$;
Calculer une solution optimale w de

$$min\{f(x) : x \in (P \cap D)\};$$

Ont dèfini arrêter $\leftarrow$ faux , $k \leftarrow 1$;

tant que arrêter $=$ faux **faire**

si $D \setminus P = \emptyset$ **alors**

arrêter $\leftarrow$ vrai (w est une solution optimale de (1.2)) ;

sinon

Calculer $y \in V(D) \setminus P$, $z \notin P$ tel que $y \in co(P \cup \{z\})$;

Définir $P \leftarrow co(P \cup \{z\})$;

Calculer une solution optimale w^* de $\min\{f(x) : x \in P \cap D\}$;

Définir $w \leftarrow w^*$;

$k \leftarrow k + 1$

fin si

fin tant que fin

1.6.2 La méthode d'Approximation exterieure

En prenant en considération le problème (1.1), avec les mêmes condtion la seul déffirence par rapport à l'algorithme d'Approximation interne c'est que cette algorithme fait une approche exterieure .

Initialisation

-Construire un polytope $X \subset P$;

-Trouver les sommets de P ;

-Poser arrêter $\leftarrow$ faux , $k \leftarrow 1$;

Traitement

Tant que arrêter $=$ faux **Faire**

Calculer une borne inf

$$\beta \leq f^* = \min\{f(x) : x \in X\}$$

est un point $v \in P$ satisfaisant $f(v) \leq \beta$;

si $I(v) := \{i : a_i^T v > b_i, i = \{1, ..., m\}\} = \emptyset (v \in X)$ **Alors**

arrêter $\leftarrow$ vrai la solution v est optimale $f(v) = \beta$;

Sinon

Choisir $j \in I(v)$;

Poser $P \longleftarrow P \cap \{x : a_i^T x \leq b_j$; Calculer $V(P)$;

$k \leftarrow k + 1$;

Fin si

Fin tant que

Fin

-Le polytope initial est un simplexe contenant l'ensemble D.
-Il existe des techniques pour déterminer le simplexe qui soit le plus près de D.
-La born inf est considérer comme l'envelope convexe f sur P.

1.6.3 la méthode de Séparation et Évaluation (Branch and Bound)

Pour plusieurs problèmes , en particulier les problèmes combinatoires,l'espace de solutions est fini (dénombrable). Il est donc possible en principe d'énumérer toutes les solutions et ensuite de prendre la meilleure .L'inconvénient majeur de cette approche est le temps de calcul qui est en général énorme . La méthode de branch and bound (procédure par évaluation et séparation) est basée sur une méthode arborescente qui consiste à réduire par des découpages l'ensemble des solutions qui ne génèrent pas de meilleures solutions. Toutes les séparations sont permises à condition de ne perdre aucune information. [13]

Algorithme général

X^* : la solution du (P) de problème (PLNE)
X^{opt} :la solution du (PR) de problème (PLNR)
Z^* : la fonction économique du (P) de problème (PLNE)
Z^{opt} :la fonction économique du (PR) de problème (PLNR)
1)Résolution du problème relaxé (PR) par la méthode du simplexe :
-Si X^{opt} est entier : fin
-Sinon, aller à 2).
2)Initialisation :
Soit Z^{opt} obtenu dans (PR) une borne supérieure pour un problème de maximisation respectivement (borne inférieure pour un problème de minimisation)pour (P_i). Soit n_1 le sommet initial de l'arborescence et son ensemble$(S_1 = S)$.
$Z_1 = Z^{opt}$.
3)Séparation :
Choisir une variable non entière x^l,créer deux branches (deux sommets fils n_{i+1} et n_{i+2}),on obtient deux sous-problèmes sous la forme :

$$\begin{cases} P_{i+1,k}: & P_i + \text{ la contrainte } x_l \leq [x_l] \\ P_{i+2,k}: & P_i + \text{ la contrainte } x_l \geq [x_l] + 1 \end{cases}$$

Avec $[x_l]$: la partie entière de x_l
4)Résolution des sous-problèmes :
Résoudre chaque sous-problème en utilisant le simplexe ou le dual simplexe.

5)Évaluation :
Examiner chaque sous-ensemble :
On peux tailler un sommet si :
*) La solution est non réalisable.
*) $Z_1 \leq Z$ pour un problème de maximisation ($Z_1 \geq Z$ pour un problème de minimisation), avec Z la solution du sous-problème.
*) La solution est non entière et son Z inférieur ou égale à une solution entière pour un problème de maximisation (supérieure ou égale pour un problème de minimisation).
Il est inutile de séparé si :
*) La solution est entière.
6)test :
S'il y a plus de sous-ensembles à séparer ,alors :
On compare tous les Z des solutions entières et on prend la plus grande d'entre elles soit Z^* pour un problème de maximisation (la plus petite pour un problème de minimisation). Elle sera la valeur de la fonction économique de la solution optimale X^* de notre problème (P).
Sinon retour à 3).
Remarque : $Z^* \leq Z^{opt}$ pour un problème de maximisation ($Z^* \geq Z^{opt}$ pour un problème de minimisation).
Dans l'itération 3) pour séparer les sous-problèmes on utilise l'une des stratégies cités précédemment.

1.6.4 la méthode de Branch-and-Cut

C'est une méthode hybride entre la méthode Branch-and-bound et la méthode des plans sécants, qui est une méthode d'optimisation combinatoire pour résoudre les PLNEs. Cette méthode implique d'exécuter l'algorithme branche et bound, et en utilisant des plans sécants pour affiner l'espace de recherche [31].
Comme les autres méthodes de résolution de problèmes de programmation linéaire en nombres entiers, cette procédure aussi commence par résoudre le problème relaxé du problème initial. Si une solution de ce problème est obtenue, on effectue un test d'intégrité, c-à-d, si la solution est entière alors l'algorithme s'arrête et renvoie ce point comme solution optimale du problème original. Sinon, l'algorithme des plans sécants est utilisé pour trouver d'autres contraintes linéaires pour réduire l'espace de recherche. Ces inégalités sont ajoutées au programme linéaire. Le processus s'arrête quand une solution entière est trouvée.
Algorithme général
1. Initialisation : Indique le problème initial de programmation d'un entier par ILP^0 et définissez les nœuds actifs sur $L = ILP^0$. Définissez la limite supérieure sur $z = +\infty$ Définir $z_l = -\infty$

pour le seul problème $l \in L$.

2. Résiliation : Si $L = \emptyset$, alors la solution x^* qui a donné lieu au titulaire la valeur de l'objectif z est optimale. S'il n'y a pas de x^* (c.-à-d., $z = +\infty$), alors ILP est impossible.

3. Sélection du problème : Sélectionnez et supprimez un problème ILP^l dans L.

4. Relaxation : Résolvez la relaxation de la programmation linéaire de ILP^l Si la relaxation est impossible, régler $z_l = +\infty$ et passer à l'étape 6. Laisser $z_l dnoterlavaleurobjectiveoptimaledelarelaxatio$
être une solution optimale ; sinon régler $z_l = -\infty$

5. Ajouter des plans de coupe : Si vous le souhaitez, rechercher des plans de coupe qui sont violés par x^{lR} ; si vous en trouvez, ajoutez-les à la relaxation et revenez à l'étape 4.

6. Exploration et élagage :

(a) Si $z_l \geq z$, passer à l'étape 2.

(b) Si $z_l < z$ et x^{lR} font partie intégrante, mettre à jour $z = z_l$, supprimer de L tout problèmes avec $z_l \leq z$ et passer à l'étape 2.

7. Partitionnement : laissé $\{S^{lj}\}_{j=1}^{j=k}$ soit une partition de l'ensemble de contraintes S^l du problème ILP^l Ajouter des problèmes $\{ILP^{lj}\}_{j=1}^{j=k}$ à L, où ILP^{lj} est ILP^l avec possible région limitée à S^{lj} et z_{lj} pour $j = \{1, ..., k\}$ est fixé à la valeur de z_l pour le problème parent l. Passer à l'étape 2.

1.7 Conclusion

Dans ce chapitre, nous avons présenté les définitions et les méthodes que nous trouvons nécessaires pour la suite de ce document. Nous avons commencé par définir les concepts de base de l'optimisation mono objectif de manière générale. Ensuite, nous avons défini les notions de la programmation linéaire, la programmation linéaire en nombres entiers, l'optimisation Global et nous avons présenté les méthodes les plus utile pour ce travail. Ces méthodes ne sont pas exhaustives, il existe plusieurs autres méthodes de résolution que le lecteur pourra trouver dans plusieurs ovrages.

Chapitre 2

Optimisation multicritére

Introduction

Dans ce présent chapitre, nous présentons la définition de la programmation multi-objectifs d'une manière plus général, on citera les concepts de base de l'optimisation linéaire multi-objectifs, ainsie quelques méthodes de résolution.

2.1 Problème d'optimisation multi-objectif

L'optimisation multi-objectif (appelée aussi Programmation multi-objective ou optimisation multi-critére) est une branche de l'optimisation mathématique traitant spécifiquement des problémes d'optimisation ayant plusieurs fonctions objectifs. Elle se distingue de l'optimisation multidisciplinaire par le fait que les objectifs à optimiser portent ici sur un seul critére. Les problémes multiobjectifs ont un intérét grandissant dans l'industrie à les responsables sont contraints de tenter d'optimiser des objectifs contradictoires. Ces problémes sont souvent difficiles, leur résolution en des temps raisonnables devient nécessaire et alimente une partie des travaux de recherche traitant de la recherche operationnelle. Ce chapitre sera consacré à la présentation des concepts fondamentaux de l'optimisation mulicritére. [34]

2.2 Formulation

Dans sa forme la plus générale, un probléme d'optimisation multiobjectif consiste à trouver dans un ensemble de solutions admissibles, un sous-ensemble de solutions minimisant ses objectifs, Le cas de la maximisation peut étre traité comme une minimisation aprés inversion des signes de l'expression à maximiser. Formallement étant donnés un ensemble de solutions admissibles $X \subseteq \mathbb{R}^n$ et $f : \mathbb{R}^n \to \mathbb{R}^p$, les P fonctions objectif, le probléme d'optimisation multiobjectif consiste à déterminer :

$$P : \min\{f(x) : x \in X\} \tag{2.1}$$

L'ensemble des solutions admissibles X est appelé région admissible. Une solution $x \in X$ est également indifféremment dénommée solution réalisable pour le probléme P. On note Y l'image de X par f telle que $\forall y \in Y, \exists x \in X$ tel que $f(x) = y$. Un élément $y \in Y$ est appelé point réalisable ou encore point faisable.

2.3 Notions de dominance et d'optimalité

La résolution d'un problème d'optimisation multi-objectifs donne une multitude de solutions. Seul un nombre restreint de ces solutions va nous intéresser. Pour qu'une solution soit intéressante [14], il faut qu'il existe des relations d'ordre entre la solution considérée et les autres solutions, dans ce cas, ces relations d'ordre appelées relations de dominance, correspondent à différents concepts d'optimalité existant dans la littérature : optimalité de Pareto et optimalité Slater [30].

2.3.1 Dominance faible

On dit que la solution $x = (x_1; x_2; ...; x_n)$ domine faiblement la solution $y = (y_1; y_2; ...; y_n)$ si :

- x est au moins aussi bon que y dans tous les objectifs
- x est strictement meilleur que y dans au moins un objectif.

Mathématiquement , on dit qu'une solution x domine faiblement une solution y (dans le cas de minimisation) ssi :

$$\forall i \in \{1, ..., k\}, f_i(x) \leq f_i(y) \text{ et } \exists i \in \{1, ..., k\} \text{ tel que } f_i(x) < f_i(y)$$

et on note $x \leq y$

2.3.2 Dominance forte

On dit que la solution $x = (x_1; x_2; ...; x_n)$ domine fortement la solution $y = (y_1; y_2; ...; y_n)$, si et seulement si x est meilleure que y sur tous les critères.

Mathématiquement, on dit qu'une solution x domine fortement une solution y (dans le cas de minimisation) ssi :

$$\forall i \in \{1, ..., k\}, f_i(x) < f_i(y)$$

et on note $x < y$.

Propriétés de la relation de dominance

Une relation de dominance a les propriétés suivantes :

- Elle n'est pas réflexive car une solution ne se domine pas elle même.
- Elle n'est pas symétrique, car on n'a jamais $x < y$ et $y < x$.
- Elle est transitive, car si $x < y$ et $y < z$ alors $x < z$.

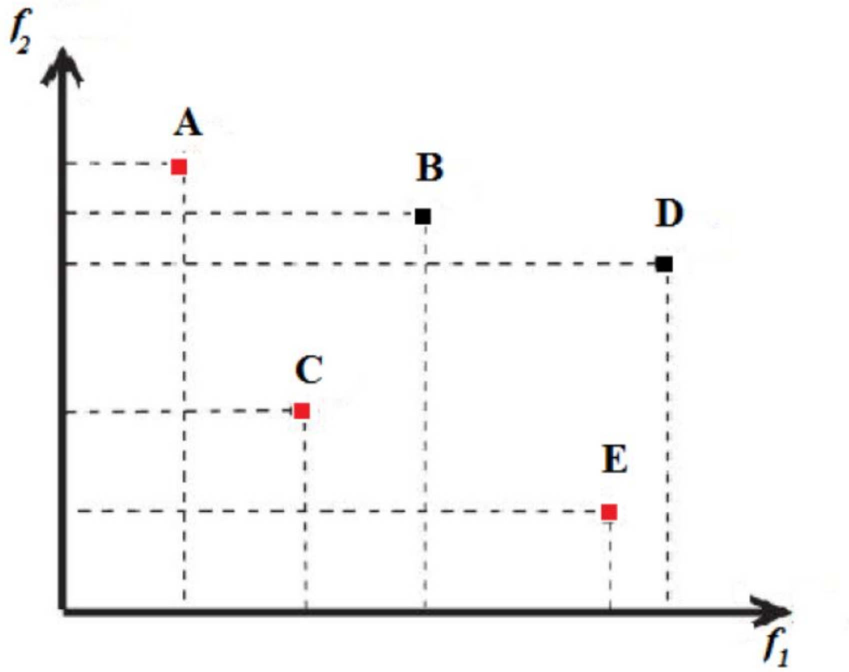


FIGURE 2.1 – Exemple de la relation de dominance

Dans cette exemple, les points A, C, E ne sont pas dominés par d'autres points. Tandis que le point C domine les points B et D, le point E domine le point D.
Les points A, C, E sont au même niveau, c'est-à-dire, ils sont incomparable.

2.3.3 Optimalité de Slater

Une solution réalisable $x^* \in S$ est dite faiblement efficace (ou optimale au sens de Slater), si et seulement si il n'existe pas une autre solution $x \in S$ telle que $f(x) < f(x^*)$, c'est-à-dire :

$$\nexists x \in S, x \neq x^* \; tel \; que \; f_i(x) < f_i(x^*)$$

A partir d'une solution faiblement efficace, il est impossible d'améliorer tous les critères à la foi [14].
L'ensemble de toutes les solutions faiblement efficaces, noté SEF ,l'ensemble des points correspondant dans l'espace des critères est l'ensemble des points non fortement dominés.

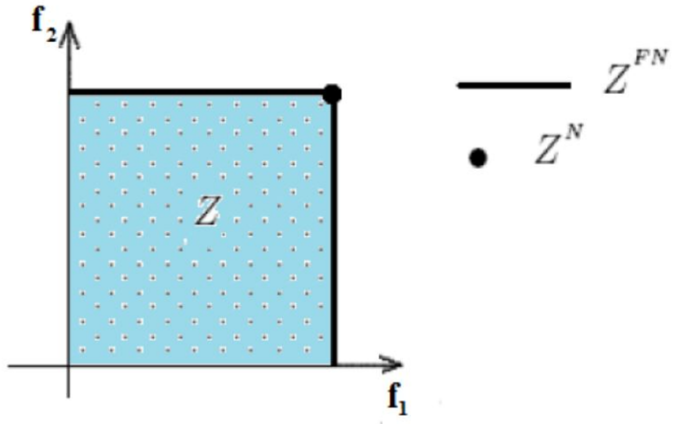


FIGURE 2.2 – L'ensemble des points non fortement dominés et dominés.

2.3.4 Optimalitée de Pareto

Une solution réalisable $x^* \in S$ est efficace (ou optimale au sens de Pareto) si et seulement si il n'existe pas une autre solution $x \in S$ x domine x^* et on la note $x < x^*$:

$$\nexists x \in S \setminus x \neq x^* : \begin{cases} f_i(x) \leq f_i(x^*); \quad \forall i = \overline{1,k} \\ \texttt{et} \\ \exists j \in \{1,...,k\} : \quad f_j(x) < f_j(x^*) \end{cases}$$

La solution d'un problème multi-objectifs est donc un ensemble formé de toutes les solutions efficaces, appelé l'ensemble efficace ou Pareto optimal et noté S^E. L'ensemble de points correspondant dans l'espace des critères est l'ensemble des points non-dominés noté par :

$$z^N = \{z \in \mathbb{Z}, \ tel \ que \ \nexists z' \in \mathbb{Z} : z' \leq z\}$$

Ainsi, toute solution de l'ensemble Pareto peut être considérée comme optimale puisque il est impossible d'augmenter (améliorer) la valeur d'un des critères sans diminuer la valeur d'au moins un autre critère. Ces solutions forment la frontière de Pareto.
Remarque : La relation entre les deux concepts d'optimalité est que l'ensemble des solutions efficaces (points de Pareto) est un sous-ensemble de l'ensemble des solutions faiblement efficaces (points de Slater), c'est-à-dire :

$$S^E \subseteq S^{EF}$$

2.3.5 Points particuliers

Pour certains points de références permettant de discuter de l'intérêt des solutions trouvées, des points particuliers ont été définis dans l'espace objectif. Ces points peuvent représenter des solutions réalisables ou non [17].

Le point idéal Le point idéal z^I est le point ou le vecteur qui a comme valeur pour chaque fonction objectif la valeur optimale.

$$z^I \text{ tel que : } \forall i = \{1, ..., n\};\ f_i(z^I) = opt_{x\in S} f_i(x)\ .$$

Dans la plupart des problèmes multi-objectifs, le point idéal z^I n'est pas une solution réalisable, car si c'était le cas, alors les objectifs ne sont pas contradictoires et une solution qui optimise un objectif, optimise tous les autres objectifs, ce qui ramènerait le problème à un problème ayant une seule solution Pareto optimale.

Le point utopique z^U est un point idéal particulier peut être défini de la façon suivante :

$$z^U = z^I - \varepsilon u.$$

où $\varepsilon > 0$ et $u = \{1; ...; 1\} \in \mathbb{R}^n$ est le vecteur unitaire, à partir se sa définition, il est clair, que ce point n'est pas réalisable.

Le point nadir Le point nadir z^N est le point ou le vecteur correspond à les bornes supérieures sur la surface de Pareto et non pas dans tout l'espace des solutions réalisable.

$$z^N \text{ tel que : } \forall i = \{1, ..., n\};\quad f_i(z^N) = opt_{x\in D\setminus f_j(z^I)} f_i(x) \text{ avec } j \neq i$$

Cela revient donc à affecter pour chaque fonction objectif du point Nadir la meilleure valeur possible parmi les solutions optimisant les autres fonctions.

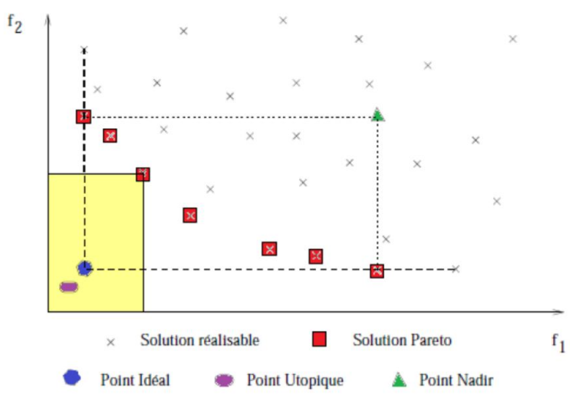


FIGURE 2.3 – Illustration des différents points particuliers.

2.3.6 Teste d'efficacité

Théorème 2.3.1 ([27]) Soit $\hat{x}$ un point arbitraire réalisable de S, $\hat{x}$ est une solution efficace du problème (MOLP) si et seulement si la valeur optimale de la fonction objectif Θ^* est nulle dans le programme linéaire suivant :

$$(P_{\hat{x}}) \begin{cases} \max \quad \Theta = \sum_{i=1}^{p} \psi_i \\ s.c \ \ c^i x - \psi_i = c^i \hat{x}, \quad i = 1, ..., p \\ x \in S \\ \psi_i \in \mathbb{Z}_+^*, \quad i = 1, ..., p \end{cases}$$

Le problème $(P_{\hat{x}})$ est souvent utilisé pour tester l'efficacité d'une solution réalisable donnée. Le théorème suivant, montre que $(P_{\hat{x}})$ peut être aussi utilisé pour générer une solution efficace même si la solution $\hat{x}$ ne l'est pas.

Théorème 2.3.2 ([18]) Si $(P_{\hat{x}})$ possède une valeur maximale finie non nulle atteinte en un point réalisable w alors w est efficace.

Théorème 2.3.3 ([18]) Si $(P_{\hat{x}})$ n'admet pas une solution optimale finie, alors l'ensemble, E des solutions efficaces du problème (MOLP) est vide.

Ces trois théorèmes ont été énoncé dans le cas continu, ils restent valables pour le cas où les variables de décision sont discrétes.

Dans les deux algorithmes décrits dans le chapitre trois, à chaque itération, on est appelé à tester l'efficacité d'une solution réalisable x par la résolution du probléme (P_x), appelé souvent (test d'efficacité) tel qu'il est décrit dans le théorème (2.3.1) dont les paramètres restent inchangés, sauf le domaine de décision qu'il faut remplacé par le domaine discret D.

Théorème 2.3.4 Soit $\hat{x}$ une solution réalisable de D, $\hat{x} \in IE$ si et seulement si la valeur optimale de la fonction objectif Θ^* est nulle dans le programme linéaire suivant :

$$(P_{\hat{x}}) \begin{cases} \max \quad \Theta = \sum_{i=1}^{p} \psi_i \\ s.c \ \ c^i x - \psi_i = c^i \hat{x}, \quad i = 1, ..., p \\ x \in D \\ \psi_i \in \mathbb{Z}_+^*, \quad i = 1, ..., p \end{cases}$$

Preuve

($\Rightarrow$) Par l'absurde, soit $\hat{x}$ est solution efficace de (MOILP), on suppose que $\Theta^* \neq= 0$, alors $\exists i \in \{1, 2, ..., P\}$ tel que $\psi_i > 0$ et $\exists x \in D$ tel que $c^i x > c^i \hat{x}$ donc Cx domine $C\hat{x}$ ce qui est contradictoire avec l'hypothèse que $\hat{x}$ est efficace.

($\Leftarrow$) Soit maintenant $\Theta^* = 0$, on suppose que $\hat{x}$ n'est pas efficace , alors il existe $x \in D$ tel que $Cx \geq C\hat{x}$ et $Cx \neq Cx^*$ alors $\exists i \in \{1, 2, ..., p\}$ tel que $Cx - C\hat{x} > 0$ donc $\psi_i > 0$ ce qui contredit le fait $\Theta^* = 0$.

2.4 La programmation linéaire Multi-objectifs en nombre entiers

Un problème de programmation linéaire multi-objectifs en nombre entiers (MOILP) est constitué d'un espace de décisions discret non convexe définit par un ensemble de contraintes linéaires sur lequel plusieurs critères souvent conflictuels supé sont optimisés. Mathématiquement, ce problème peut être formulé par : Rappelons q'un problème inéaire multi-objectifs en nombres entiers (MOILP) peut être formulé par :

$$(MOILP)\begin{cases} \max \ Cx \\ s.c \ \ x \in D \end{cases} \tag{2.2}$$

avec $D = S \bigcap \mathbb{Z}$, $\mathbb{Z}$ étant l'ensemble des nombres entiers relatifs. S est supposé borné et convexe $S = \{x \in \mathbb{R}^n \setminus Ax \leq b, x \geq 0\}, A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$, C est une matrice de dimension $p \times n$ d'éléments réels, et ses vecteurs lignes $c^i \in \mathbb{R}^n, i = 1, 2, .., p$ définissent les critères.

Définition 2.4.1. Une solution réalisable $\hat{x}$ du problème (MOILP) est une solution éfficace, si et seulement si, il n'existe pas d'autre solution réalisable x telle que $Cx > C\hat{x}$, avec au moins une inégalité stricte. Le vecteur critère correspondant $C\hat{x}$ est dit solution non dominée (Point Pareto).

Notons par IE l'ensemble de toutes les solutions efficaces et par $\mathbb{Z}^{IE}$ l'ensemble de touts points non dominés du problème (MOILP). Notons que les notions de bases concernant, la dominance faible et forte, les points caractéristiques (Points idéal, point nadir,. etc.) énoncées dans la section précédente restent valables pour les problemes (MOILP).

Thorème 2.4.2 [34] si x^* est une solution du problème paramétrique $(P_\lambda) \equiv \max\{\lambda Cx \mid x \in S\}, \lambda \in \mathbb{R}^p$ et $\lambda > 0$ alors x^* est une solution efficace du problème (MOLP)$\equiv \max\{\lambda Cx \mid x \in S\}$

2.5 Les Métodes de résolution

2.5.1 La Méthode ε-contrainte

La méthode de ε-contrainte est probablement la technique la plus connue pour résoudre les problèmes d'optimisation multicritères. Il n'y a pas d'agrégation de critères, mais un seul des objectifs d'origine est minimisé, tandis que les autres sont transformés en contraintes. Il a été introduit par [25], et une discussion approfondie peut être trouvée dans [11]. On remplace le problème d'optimisation multicritère (2.1) par le problème de ε-contrainte.

$$\min_{x\in X}\{f_j(x)\} \tag{2.3}$$

$$soumis\ a\ f_k(x) \leq \varepsilon_k\ avec\ k = \overline{1, n}\ k = j$$

lorsque $\varepsilon \in R^n$, Le composant ε_j n'est pas important pour le problème (2.3), mais la convention pour l'inclure sera pratique plus tard. La figure (2.4) illustre un problème bicritère, où une contrainte de borne supérieure est appliquée à $f_k(x)$. Les valeurs optimales du problème (2.3) avec $j = 2$ pour deux valeurs de ε_1 sont indiquées. Ceux-ci montrent que les contraintes $f_k(x) \leq \varepsilon_k$ peut être actif ou non à une solution optimale de problème (2.3).

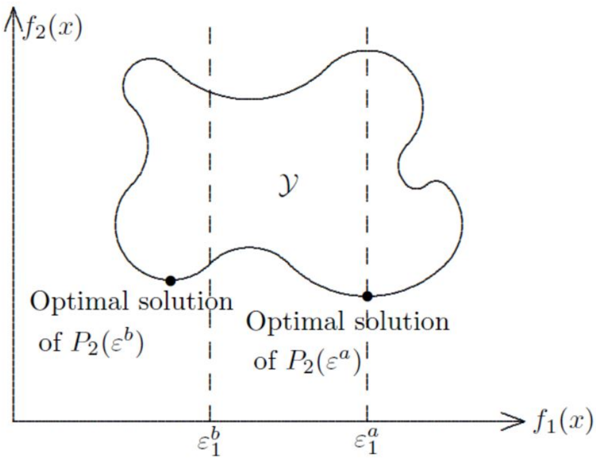


FIGURE 2.4 – La solution optimale de problème ε-contrainte (2.3).

Pour justifier l'approche, nous montrons que les solutions optimales des problèmes (2.3) sont au moins faiblement efficaces. Une condition nécessaire et suffisante d'efficacité montre que cette méthode fonctionne pour des problèmes généraux, aucune hypothèse de convexité n'est nécessaire

2.5.2 La Méthode de Benson

La méthode et les résultats décrits dans cette section sont de [5]. L'idée est de choisir une solution réalisable initiale $x^0 \in \mathcal{X}$ et, si elle n'est pas elle-même efficace, de produire une solution dominante qui l'est. Pour ce faire, des variables d'écart non négatives $l_k = f_k(x^0) - f_k(x)$ sont introduites et leur somme maximisée. Il en résulte un x dominant x^0, s'il existe, et l'objectif s'assure qu'il est efficace, poussant x le plus loin possible de x^0.[20]
Le problème de substitution (2.4) pour x^0 est donné :

$$\max \sum_{k=1}^{p} l_k \tag{2.4}$$

$$sujet\ \grave{a}\ f_k(x^0) - l_k - f_k(x) = 0 \setminus k = 1, ..., p$$

$$l \geqq 0 \; et \; x \in \mathcal{X}$$

Une illustration dans l'espace objectif figure (2.5) démontre l'idée. Le point initial faisable, mais dominé, $f(x^0)$ a des valeurs supérieures au point efficace $f(\hat{x})$. En maximisant l'écart total $\hat{l}_1 + \hat{l}_2$, l'intention est de trouver une solution dominante, efficace.

Tout d'abord, la résolution du problème (2.4) est un contrôle de l'efficacité de la solution initiale x^0

Théorème 2.5.1 La solution réalisable $x^0 \in \mathcal{X}$ est efficace si et seulement si la valeur objective optimale du problème (2.4) est 0.

Démonstration. Soit (x, l) une solution réalisable du problème (2.4). En raison de la contrainte de non-négativité l_k pour $k = 1, ..., p$ et la définition de l_k comme $f_k(x^0) - f_k(x)$ nous avons

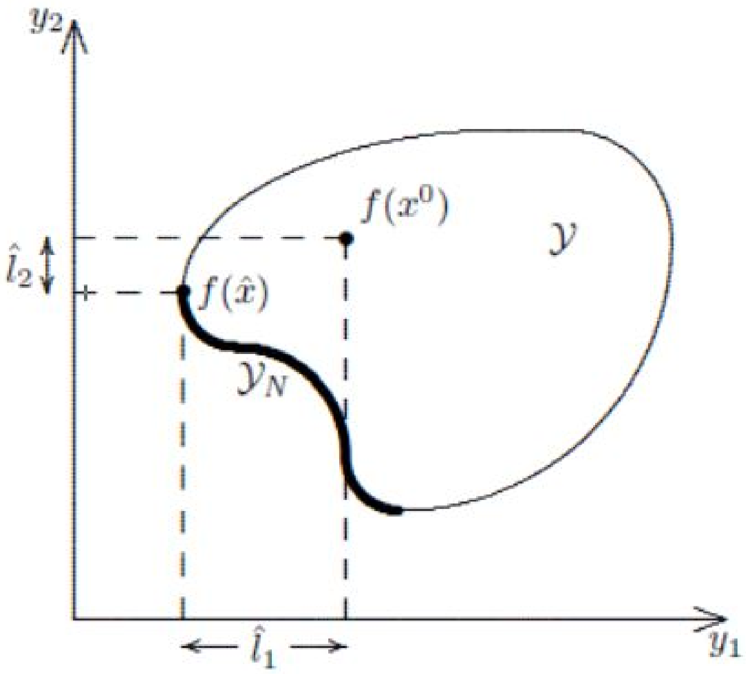


FIGURE 2.5 – Illustration du problème de Benson (2.3).

$$\max \sum_{k=1}^{p} l_k = 0 \Longleftrightarrow l_k = 0 \setminus k = 1, ..., p$$

$$\Longleftrightarrow f_k(x^0) = f_k(x)$$

Ainsi, si la valeur optimale est 0 et que $x \in \mathcal{X}$ est tel que $f(x) \leqq f(x^0)$, il faut que $f(x) = f(x^0)$, c'est-à-dire que x^0 soit efficace. Si, par contre, x^0 est efficace, l'ensemble des possibles de problème (2.4) est constitué de ceux (x, l) pour lesquels $x \in \mathcal{X}$ et $f(x) = f(x^0)$ et donc $l = 0$. On ne peut généralement pas s'attendre à ce que la solution initiale x^0 soit efficace. La force de la méthode réside dans le fait que chaque fois que le problème (2.4) a une valeur de solution optimale finie, la solution optimale est efficace. Sous des hypothèses de convexité, il n'existe aucune solution proprement efficace. D'un point de vue applicatif, cela constitue une situation pathologique : toutes les solutions efficaces auront des compromis illimités. Cependant, cela ne peut se produire que dans des situations où l'existence de solutions efficaces n'est généralement pas garantie.

Proposition 2.5.2. Si le problème (2.4) a une solution optimale $(\hat{x}, \hat{l})$ (et la valeur objective optimale est finie) alors $\hat{x} \in \mathcal{X}$

Théorème 2.5.3 [5]. Supposons que les fonctions f_k, $k = 1, ..., p$ sont convexes et que $\mathcal{X} \subset R^n$ est un ensemble convexe. Si le problème (2.4) n'a pas de valeur objective optimale finie $\mathcal{X}_{pE} = \emptyset$.

2.6 Conclusion

Dans ce chapitre, Nous avons commencé par définir les concepts de base de l'optimisation multi objectif de manière générale. Ensuite, nous avons défini les Notions de dominance et d'optimalité, et les points (nadir, ideal, efficce, parito, slater).tout on focalisons sur la programation linéare muli-objectifs, et nous avons présenté les méthodes les plus utile pour ce travail. Ces méthodes ne sont pas exhaustives, il existe plusieurs autres méthodes de résolution que le lecteur pourra trouver dans plusieurs ouvrage.

Chapitre 3

L'optimisation sur l'ensemble efficace

3.1 Introduction

Dans ce chapitre on va commencer par la présentation d'un problème bi-niveaux (PB) en général, en suite le problème linéaire bi-niveaux (PBL), ce qui va nous aider à présenter l'optimisation d'une fonction sur un enssemble efficace.

3.2 Programmation bi-niveaux

Les problèmes de programmation à deux niveaux sont des problèmes d'optimisation dont les contraintes sont déterminées, en partie, par un autre problème d'optimisation. En d'autres termes, ce sont des programmes mathématiques hiérarchiques avec deux niveaux de décision, les toutes premières formulations liées au problème de programmation bi-niveaux sont apparues dans l'œuvre des auteurs [10]

3.2.1 Formulation mathématique d'un problème de programmation bi-niveaux

Dans sa forme générale, un problème de programmation bi-niveaux (PBN) est formulé comme suit :

$$\begin{cases} \min\limits_{x} F(x,y) \\ s.c \begin{cases} G(x,y) \leq 0 \\ \min\limits_{y} f(x,y) \\ s.c \quad g(x,y) \leq 0 \end{cases} \end{cases} \tag{3.1}$$

où l'ensemble de toute les variables est partitionnée entre un vecteur $x \in \mathbb{R}^{n_1}$ représentant le premier niveau de décision (niveau supérieur), et un vecteur $y \in \mathbb{R}^{n_1}$ pour le second niveau de décision (niveau inférieur). De même, les fonctions $F : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}$ et $f : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}$ sont les fonctions objectifs du premier niveau de décision et de second niveau de décision respectivement, et les fonctions vectorielles $G : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}^{m_1}$ et $g : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}^{m_1}$ sont respectivement les contraintes du niveau supérieur et niveau inférieur.

3.2.2 Programmation linéaire bi-niveaux

Le problème de programmation linéaire bi-niveaux (PBL) est un cas particulier des problèmes de programmation bi-niveaux, où les fonctions objectifs et les contraintes du Leader et du Suiveur sont linéaires. pour plus de détails vous pouvez voir [2]

$$\begin{cases} \min\limits_{x} F(x,y) = c_1^t x + d_1^t y \\ s.c \ A_1 x + B_1 y \leq b_1, \\ x \geq 0, \\ \\ \min\limits_{y} f(x,y) = c_2^t x + d_2^t y \\ s.c \begin{cases} A_2 x + B_2 y \leq b_2, \\ y \geq 0, \end{cases} \end{cases} \tag{3.2}$$

où $F : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}$, $f : \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}$ sont les fonctions objectifs du leader et du suiveur (follower) respectivement ; $c_i \in \mathbb{R}^{n_1}, d_i \in \mathbb{R}^{n_2}, b_i \in \mathbb{R}^{m_i}, A_i \in \mathbb{R}^{m_i} \times \mathbb{R}^{n_1}, B_i \in \mathbb{R}^{m_i} \times \mathbb{R}^{n_2}, i = 1, 2$. Les contraintes $A_1x+B_1y \leq d_1$ (respectivement $A_2x+B_2y \leq d_2$), sont les contraintes du premier (respectivement du second) niveau. La fonction linéaire $F(x, y)$ (respectivement $f(x, y)$) représente la fonction économique du leader (respectivement du suiveur), alors que x (respectivement y) désigne le vecteur des variables du leader (respectivement du suiveur).

3.3 L'optimisation d'une fonction sur l'ensemble efficace

Le problème de l'optimisation d'une fonction sur l'ensemble des solutions efficaces du problème (linéaire multi-objectifs) a été étudié pour la premiére fois par [33], qui a décrit un algorithme basé sur le déplacement sur les sommets efficaces adjacents dans le cas où la fonction à optimiser est linéaire. Depuis, notamment dans le cas continue, un bon nombre de chercheurs, citons par exemple, [6], [7], [8], [26], [35], [22] et [19] ont suivi cette voie, plusieurs méthodes ont été développées où plusieurs formulations équivalentes au probléme en question ont été proposées. [35] à classifier ces méthodes en différentes catégories à savoir :

-les algorithmes de recherche de sommets adjacents ;

-les algorithmes de recherche de sommets non adjacents ;
-les algorithmes basés sur la méthode de branch and bound ;
-les algorithmes basés sur la méthode de relaxation Lagrangienne ;
-les algorithmes basés sur la méthode duale ;
-les algorithmes basés sur la bisection.
Contrairement au cas continue qui a été complétement étudié, le cas discret n'a pas vu beaucoup de développements semblable, la premiére tentative d'étudier le problème à été réalisée par [32] où seulement une borne supérieure de la valeur optimale du critére principal à été donnée. La premiére méthode proposée pour l'optimisation sur l'ensemble éfficace d'un problème (linéaire multi-objectifs en nombres entiers) en évitant l'énumération explicite de tous les points éfficaces est celle de [1], où différents types de coupes sont imposées de telle maniére que l'amélioration de la valeur optimale du critére principal à chaque itération soit garantie, suivie par l'algorithme de [28] qui est basé sur l'analyse d'un ordre approprié des problèmes linéaires en nombres entiers pour éliminer successivement les solutions moins bonnes sur le critère principal.

3.3.1 Formulation du problème

Rappelons q'un problème inéaire multi-objectifs en nombres entiers (MOILP) peut être formulé par :

$$(MOILP) \begin{cases} \max \ Cx \\ s.c \ \ x \in D \end{cases} \tag{3.3}$$

avec $D = S \bigcap \mathbb{Z}$, $\mathbb{Z}$ étant l'ensemble des nombres entiers relatifs. S est supposé borné et convexe $S = \{x \in \mathbb{R}^n \setminus Ax \leq b, x \geq 0\}, A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$, C est une matrice de dimension $p \times n$ d'éléments réels, et ses vecteurs lignes $c^i \in \mathbb{R}^n, i = 1, 2, .., p$.
Le problème de l'optimisation d'une fonction linéaire appelée (critère principal) qu'on note par P_E sur l'ensemble des solutions efficaces IE du probléme (MOILP) est formulé comme suit :

$$(P_E) \begin{cases} \max \ \phi = dx \\ s.c \ x \in IE \end{cases} \tag{3.4}$$

Son problème relaxé noté par (P_R) est définit comme suit :

$$(P_R) \begin{cases} \max \ \phi = dx \\ s.c \ x \in D \end{cases} \tag{3.5}$$

Le problème (P_E) est un problème d'aide à la décision. Ce problème est doté de deux classes de problémes : les problèmes de l'optimisation linéaire multi-objectifs en nombres entiers dont l'étude est en plein essor et les problèmes de l'optimisation unicritère discrète (ILP) qui constitue une extension assez large de la programmation linéaire (LP). Compte tenu de cette caractérisation, Il n'est pas surprenant que peu de travaux aient été réalisés et que seulement deux méthodes ont été proposée.

Naïvement, on peut résoudre le problème (P_E) en formant la liste de toutes les solutions efficaces IE et en optimisant $\phi(x) = dx$ sur cette liste. Cette approche n'est pas appropriée pour des raisons d'ordre pratique liées à la dificulté de déterminer IE qui peut être un ensemble de taille exponentielle en le nombre de variables. Les sources de motivation pour étudier de tel problème vise à éviter cependant l'énumération explicite de toutes les solutions efficaces. On peut se demander pourquoi optimiser sur l'ensemble des solutions Pareto- optimales alors qu'il suffit d'optimiser sur la frontière efficace, l'ensemble des points non dominés dans l'espace des critères. Ceci est possible si la fonction à optimiser peut être exprimée en fonction des critères initiaux (variables dans l'espace des critères), mais la difficulté s'installe lorsque cette fonction est exprimée en fonction des variables de décision.

Comme connu, le problème en variables continues est déjà difficile à traiter ; il devient plus difficile encore en variables discrètes. Les difficultés particulières rencontrées dans sa résolution sont dues à :

- L'optimisation sur un ensemble IE qui ne peut pas être déterminé à priori ;
- Le cadre non convexité et discret du domaine de décision D ;
- Sur le front de Pareto, non convexe, deux types de solutions peuvent être différenciées : les solutions supportées et les solutions non supportées.

3.3.2 Les Méthodes de résolution

Méthode de Abbas et Chaabane

Les auteurs ont développé une méthode exacte pour résoudre le problème (P_E) dans l'espace des critères, dont la fonction objectif est donnée par une somme pondérée des critères du (MOILP). Le principe de la méthode est le suivant :

Dans une première étape la condition nécessaire du théorème suivant :

Théorème 3.5 [24] Soit $\hat{x} \in S$, $\hat{x}$ est une solution efficace si et seulement si $\hat{x} \in S$ est une solution optimale du problème paramétrique (P_λ) pour un certain $\lambda \geq 0$.

est utilisée pour déterminer une solution efficace initiale en résolvant le problème paramétrique (P_λ), en suite, dans chaque itération k, la region d'admissibilité est réduite en ajoutant des contraintes qui se traduit par deux types de coupes éliminant les solutions dominées par la solution efficace courante, pour éliminer toutes les solutions moins bonnes sur le critère principal en résolvant un problème paramétrique (P_λ^k).

Notations et résultats préliminaires :

Dans cette méthode nous adopterons les notations suivantes :

- $z_1, z_2, ..., z_p$ dénotent les critères initiaux du (MOILP).

- x_{opt} est la solution optimale du problème (PE).
- ϕ_{opt} est la valeur optimale de ϕ.
- $z_i(x_{opt})$ est l'évaluation du critère i en la solution x_{opt}.
- $H^0 = D = S \bigcap \mathbb{Z}^n$.

Pour $k \geq 1$ on a :
- I_k : l'ensemble des indices de base du tableau optimale de l'itération k.
- N_k : l'ensemble des indices hors-base du tableau optimale de l'itération k.
- D_k la région tronquée à l'itération k.
- E_{jk} l'arête incidente à x_j à l'étape k.

Soit le problème unicritère $(P_i(D))$, $i \in \{1, ..., p\}$

$$(P_i(D)) \begin{cases} \max c^i \\ s.c \;\; x \in D \end{cases} \tag{3.6}$$

Le problème $(P_i(D))$ peut avoir plusieurs solutions optimales, nous rappelons la notion de solution alternative dans la définition suivante.

Définition 3.1 : Soit x^* une solution optimale du problème $(P_i(D))$, une solution réalisable $\bar{x} \in D$ est dite solution alternative à x^* si $C^i x^* = C^i \bar{x}$.

Considérant le problème paramétrique (P_λ) :

$$P_\lambda \begin{cases} \max z_\lambda = \sum_{i=1}^{p} \lambda_i c^i x, \quad i = 1, 2, ..., p \\ s.c \;\; x \in D \end{cases} \tag{3.7}$$

Où λ est une valeur de :

$$\Lambda = \{\lambda \in R^p, \sum_{k=1}^{p} \lambda_i = 0, \lambda_i \geq 0, i = 1, 2, ..., p$$

Théorème 3.6 si $\hat{x}$ est une solution optimale de problème pramétrique P_λ pour un certain $\lambda > 0$, alors $\hat{x}$ est efficace pou le problème (MOILP).

Soit le problème (P_λ^k)

$$P_\lambda^k \begin{cases} \max z_\lambda = \sum_{i=1}^{p} \lambda_k z^k x, \quad k = 1, 2, ..., p \\ s.c \;\; x \in D^k \end{cases} \tag{3.8}$$

Où $\lambda > 0$ et D^k est le domaine réduit définit par :

$$D^k = (D^k \backslash \bigcup_{r=0}^{k-1} \Delta_r) \bigcap D_{x_{opt}}$$

Où $D_{x_{opt}} = \{x | x \in D; dx \geq \phi_{opt} + 1\}$ et $\Delta_r = \{x | x \in \mathbb{Z}; Cx \leq Cx_r\}$ avec $x^0, x^1, ..., x^{k-1}$ des solutions efficaces obtenues en résolvant les problèmes (P_λ^0),(P_λ^1),...,(P_λ^{k-1}) Ceci peut être traduit mathématiquement par les contraintes additionnelles suivantes :

$$H^k = H^{k-1} \bigcap \begin{cases} x \in D | z_i(x) \geq (z_i(x_{opt}) + 1)y_i^k + M_i(1 + y_i^k), \quad \forall i = 1, 2, ..., p \\ \sum_{i=1}^{p} y_i^k \geq 1, \quad \forall i = 1, 2, ..., p \\ y_i^k \in \{0, 1\}, \quad \forall i = 1, 2, ..., p \end{cases}$$

Où M_i est la borne inférieure de la ième fonction objectif dans le domaine D et à chaque critère z_i nous associons une variable binaire y_i^k définie par :

$$y_i^k \begin{cases} 1 & \textit{si le critere } z_i \textit{ est strictement ameliorer par rapport a } z_i(x_{opt}); \\ 0 & \textit{sinon} \end{cases}$$

Coupes de type I

L'algorithme proposé par les auteurs est basé principalement sur l'exploration des arêtes incidentes à une solution trouvée et l'utilisation de coupes éliminant une arête contenant des solutions admissibles au lieu d'un seul point admissible.

Proposition 2. Soit x_k une solution optimale du problème unicritère $(P_1(D_k))$. Supposons que $j_k \in \mathbb{N}_k$. Une arête E_{j_k} incidente à la solution x_k est définit par l'ensemble

$$E_{j_k} = \begin{cases} & |x_i = x_{j_k} - \theta_{j_k} y_{k,i_{j_k}}, i \in I_k \\ x_i \in D_k & |x_{j_k} = \theta_{j_k} \\ & |x_\alpha = \alpha \quad \forall \alpha \in \mathbb{N}_k \setminus \{j_k\} \end{cases}$$

Où D_k est la région tronquée à l'itération k, $x_{k,i}$ est la i^{me} composante de la solution x_k, $y_{k,i_{j_k}}$ est l'élément de la ligne i de la colonne j_k dans le tableau optimale de x_k et θ_{j_k} est un entier positif tel que $0 < \theta_{j_k} \leq \min_{i \in I_k} \{ \frac{x_{k,i}}{y_{k,i_{j_k}}}, y_{k;i_{j_k}} > 0\}$ et $\theta_{jk} \times y_{k;i_{j_k}}$ est un vecteur entier $\forall i \in I_k$ si de tels vecteurs existent.

Théorème 3.7. Une solution réalisable entière du problème $(P_1(D_k))$ qui n'est pas sur l'arête E_{j_k} et incident à x_k de la région tronquée D_k, se situe dans le demi espace fermé.

$$\sum_{j \in \mathbb{N}_k \setminus j_k} x_j \geq 1, \quad k \geq 1$$

Ce théorème montre que la coupe de type I peut être considérée comme une généralisation de la coupe de Dantzig. L'avantage d'utiliser cette coupe est de tronquer toutes les solutions réalisables entières du problème $(P_1(D_k))$ qui se trouvent sur l'arête E_{jk} issue de la solution optimale x_k, tandis que, la coupe de Dantzig ne tronque qu'un seul point du domaine d'admissibilité à savoir le point x_k.

Coupes de type II

Après avoir trouvé une nouvelle solution efficace de (MOILP), qui améliore la meilleure valeur de ϕ obtenue jusqu'à present notée ϕ_{opt}, nous imposons une coupe dite coupe de type II pour chercher une nouvelle solution efficace dont la valeur ϕ est meilleure ou égale à ϕ_{opt}.

$$dx \geq \phi_{opt}$$

L'algorithme de la méthode

étape 1 :

- Résoudre le problème relaxé (P_R) définit par :

$$(MOILP) \equiv \max\{Cx;\ x \in D\}$$

et soit x^* sa solution optimale.

- Si x^* est efficace, terminer x^* est optimale pour le problème (P_E) définit par :

$$P_E \equiv \max\{dx;\ x \in IE\}$$

- Sinon, aller à l'étape (2)

étape 2 :

-Poser $\phi_{opt} = -\infty$ et résoudre un problème unicritère $(P_i(D))$,$i \in \{1,...,p\}$, par exemple, prendre $i = 1$ est poser sa solution x^1.

2.1 : Si $J_1 = \{j \in \mathbb{N}_1 | \mathbb{Z}^1_{1,j} - C^1 - j = 0\} = \emptyset$ alors la solution optimale trouvée est unique et elle est efficace, poser $\phi_{opt} = dx^1, x_{opt} = x^1$ et aller à l'étape (3).

2.2 : Si $J_1 \neq \emptyset$, alors la solution optimale x^1 peut ne pas être unique, tester son efficacité, si'elle n'est pas efficace, poser $\phi_{opt} = dx^1, x_{opt} = x^1$ et aller à l'étape (3)

étape 3 :

-Poser $k = 1$ et executer les sous étapes suivantes

3.1 : Construire l'ensemble $\Gamma_k = \{j \in \mathbb{N}_k | \mathbb{Z}^k_{1,k} - C^1_j \geq 0 \text{ et } \phi^k_j - d_j \leq 0\}$

-si $\Gamma_k = \emptyset$ ajouter la coupe de Dantzig $\displaystyle\sum_{j \in \mathbb{N}_k} \geq 1$ et aller à l'étape (3.2) :

-Sinon, soit $\gamma = \Gamma_k$ et aller à l'étape (a)

(a) : Si $\gamma = \emptyset$, alors, soit $j_k \in \Gamma_k$, appliquer la coupe de type I $\displaystyle\sum_{j \in \mathbb{N}_k \setminus \{j_k\}} x_j, \quad k \geq 1$ et aller à l'étape (3.2). Sinon, sélectionner $j_k \in \gamma$ et calculer $\theta^0_{j_k}$ la partie entiére de

$$\min_{i \in I_k} \{\frac{x_{k,i}}{Z_{k,ij_k}} | Z_{k,ij_k} > 0\}$$

- Si $\theta^0_{j_k}$ alors il n'y a aucune solution réalisable entière sur l'arête E_{j_k} , poser $\gamma := \gamma \setminus \{j_k\}$ et aller à l'étape (a) - Sinon, si $\theta^0_{jk} > 1$ aller à l'étape (b)

(b) : Si x^k est efficace et $dx^k \geq \phi_{opt}$ calculer β_k tel que

$$\beta_k = (d_{j_k} - \sum_{i \in I_k} d_i z_{k,ij_k})$$

-Si $\beta_k \neq 0$ aller à l'étape (C)
-Sinon, poser $\gamma := \gamma \setminus \{j_k\}$ et aller à l'étape (a)
-Si x^k n'est pas efficace ou $dx^k < \phi_{opt}$ aller à l'étape (c) (L'arête E_{j_k} doit être explorée quelle que soit la valeur de β_k)
(c) : Explorer l'arête E_{j_k} , en cherchant des solutions réalisables de $(P(D_k))$ correspondant à θ et tester pour l'efficacité à partir de $\theta = \theta^0_{j_k}$ jusqu'à $\theta = 1$ (tel que $\theta \in \mathbb{Z}^*_+$). Dès qu'une solution efficace $\bar{x}^k$ vérifie $d\bar{x}^k > \phi_{opt}$ est trouvée pour une valeur de θ, remplacer x_{opt} par $\bar{x}^k$ et ϕ_{opt} par $d\bar{x}^k >$ et aller à sous-étape (3.2), s'il n'y a aucune solution efficace entière sur l'arête E_{j_k}, poser $\gamma := \gamma \setminus \{j_k\}$ et aller à l'étape (a)
3.2 : Soit $k = k+1$. La nouvelle région tronquée D_k est obtenue comme sous-ensemble de D_{k-1} an appliquant la coupe de type II ($dx > \phi_{opt}$) puis en utilisant la méthode dual de simplex et les coupes Gomory autant de fois nécessaire pour trouver une nouvelle solution optimale x^k, poser $x_{opt} = x^k$ et $\phi_{opt} = dx^k$ et aller à la sous-étape (3.1)
3.3 : Soit $k = k+1$. La nouvelle région tronquée D_k est obtenue comme sous-ensemble de D_{k-1} (ou D si $k = 1$) an appliquant la coupe de Dantzig ou la coupe de type I puis en utilisant la méthode dual de simplex et les coupes Gomory autant de fois nécessaire pour trouver une nouvelle solution optimale $x^k, \phi^k = dx^k$
-Si x^k est efficace et $\phi_{opt} < dx^k$ poser $x_{opt} = x^k$ et $\phi_{opt} = dx^k$ et aller à la sous-étape (3.1)
-Sinon, aller à l'étape (3.1) sans mettre rien à jour.
étape terminale : La procédure prend fin, ou bien à la premiére étape si la solution x^0 est efficace, ou lorsque l'impossibilité des opérations de pivot indique que la région courante ne contient aucun point entier réalisable. La solution optimale est alors x_{opt} et sa valeur sur le critère ϕ est ϕ_{opt}.

Méthode de Jesus

L'algorithme proposé consiste à produire une solution optimale globale de (P_E) sans devoir énumérer l'ensemble de toutes les solutions efficaces IE, la procédure commence à résoudre le problème relaxé (P_R), sa solution est testée pour l'efficacité, évidemment, seulement dans un nombre limité de cas spéciaux la solution optimale de (P_R) fournit une solution optimale de (P_E). Ainsi, si ce n'était pas le cas, une solution efficace qui domine la solution optimale de (P_R) est alors générer par le programme linéaire de test d'efficacité définit dans théorème (3.4). Par suite, dans chaque itération, le critère principal est optimisé sur le domaine restreint $D - \bigcup_{s=1}^{l} D_s$ $D_s = \{x; x \in Z^n, Cx \leq C\hat{x}^s\}$ tel que $\hat{x}^1, \hat{x}^2, ..., \hat{x}^l$ sont des solutions efficaces obtenues à l'itérations l, en incluant progressivement des contraintes pour éliminer les solutions dominées par la solution efficace courante, afin de fournir une solution non dominée par les solutions détectées antérieurement jusqu'à ce qu'une solution optimale et efficace soit finalement trouvée.

L'algorithme de la méthode :

Initialisation :

poser $\Phi_{inf} = -\infty, \Phi_{sup} = +\infty, l = 1$, et résoudre le problème relaxé (P_R)

- Si (P_R) n'est pas réalisable. Alors terminer, le problème (P_R) n'a pas de solution.
- Autrement, soit x^l solution optimale de (P_R).

étape 1 : Tester l'efficacité de x^l

- Si x^l est efficace, l'algorithme prend fin et $X_{opt} = x^l, \Phi_{opt} = dx^l$.
- Sinon, poser $\Phi_{sup} = dx^l$ et aller à l'étape 2.

étape 2 : Soit $\hat{x}^l \in IE$ une solution optimale du test d'efficacité dont le vecteur critères domine celui de x^l.

Dans l'espace des critères, plusieurs solutions efficaces peuvent avoir le même vecteur critères, pour cela le problème (T_l) est résolus pour optimiser le critère principal sur toutes les solutions équivalentes à $\hat{x}^l$.

Le problème (T_l) est défini par

$$T_l : \max\{dx|Cx = C\hat{x}^l, x \in D\} \tag{3.9}$$

soit $\tilde{x}^l$ une solution optimale du (3.9).

- Si $d\tilde{x}^l > \Phi_{inf}$, poser $\Phi_{inf} = d\tilde{x}^l$ et $X_{opt} = \tilde{x}^l$. Aller à l'étape 3.
- Sinon, si $\Phi_{inf} = \Phi_{sup}$. Terminer, X_{opt} est la solution optimale de (P_E).

étape 3 : Résoudre le problème (R_l) défini par

$$R_l : \max\{dx|x \in D - \bigcup_{s=l}^{l} D_s\} \tag{3.10}$$

où $D_s = \{x; x \in Z^n, Cx \leq C\tilde{x}^s\}$ avec $\tilde{x}^1, \tilde{x}^2, ..., \tilde{x}^l$ sont des problèmes $(T_1), (T_2), ..., (T_l)$ respectivement.

- Si (3.10) n'est pas réalisable. Terminer, X_{opt} est une solution optimale de (P_E).
- Autrement, soit x^{l+1} solution optimale de (3.10).

*) Si $dx^{l+1} \leq \Phi_{inf}$. Terminer, X_{opt} est une solution optimale de (P_E).

*) Sinon, poser $l = l + 1$ et aller à l'étape 1.

Proposition 3. Soit x^{l+1} une solution optimale du problème (R_l) telle que $\Phi(x^{l+1}) > \max_{s \in 1,...,l}\{\Phi(\tilde{x}^s)\}$. Si $x^{l+1} \in IE$ alors x^{l+1} est une solution optimale de (P_E).

Proposition 4. Soient $\tilde{x}^1, ..., \tilde{x}^l \in IE$, si (R^l) est irréalisable, alors l'ensemble de toutes les solutions non dominées est $Z(IE) = \{C\tilde{x}^1, ..., C\tilde{x}^l\}$.

Exemple Illustratif pour la Méthode de Jesus

Considérons l'exemple à deux objectifs proposé par l'auteur :

$$(P(D))\begin{cases} \max f_1 = z_1 = x_1 - 2x_2 \\ \max f_2 = z_2 = -x_1 + 4x_2 \\ -2x_1 + 4x_2 \leq 0 \\ x_1 \leq 3 \\ x_2 \leq 2 \\ x_1, x_2 \in \mathbb{Z}_+^* \end{cases}$$

Soit l'ensemble de décisions
$D = \{x| - 2x_1 + 4x_2 \leq 0; x_1 \leq 3; x_2 \leq 2; x_1, x_2 \in \mathbb{Z}_+^*\}$.
L'ensembles de toutes les solutions efficaces est : $IE = \{(2;0), (3;0), (2;1), (3;1), (1;2), (2;2), (3;2)\}$.
Notons que les solutions $(2;1)$ et $(2;0)$ sont non-supportées.
On considère le problème principal :

$$(P_E)\begin{cases} \max \Phi = -x_1 - 2x_2 \\ x \in IE \end{cases}$$

On note par $d = [-1 - 2]$ le vecteur ligne de la fonction objectif Φ. Soit le problème relaxé du (P_E) :

$$(P_R)\begin{cases} \max \Phi = -x_1 - 2x_2 \\ x \in D \end{cases}$$

Les figures (3.1),(3.2) montrent respectivement l'espace de décisions et l'espace des critères du problème considéré.

x	(0,0)	(1,0)	(1,1)	(2,0)	(3,0)	(2,1)	(3,1)	(1,2)	(2,2)	(3,2)
$Z(x)$	(0,0)	(1,-1)	(-1,3)	(2,-2)	(3,-3)	(0,2)	(1,1)	(-3,7)	(-2,6)	(-1,5)
$\Phi(x)$	0	-1	-3	-2	-3	-4	-5	-5	-6	-7

TABLE 3.1 – Tableau de résultats de l'exemple $P(D)$

Les bornes inférieures des deux fonctions objectifs sont $-M_1 = -3, -M_2 = -3$.
Itération 1 :
Initialisation : poser $\Phi_{inf} = -\infty, \Phi_{sup} = +\infty, l = 1$.
Résoudre le problème relaxé :

$$(P_R)\begin{cases} \max \Phi = -x_1 - 2x_2 \\ x_1 \leq 3, \\ x_2 \leq 2, \\ x_1, x_2 \in \mathbb{N} \end{cases} \tag{3.11}$$

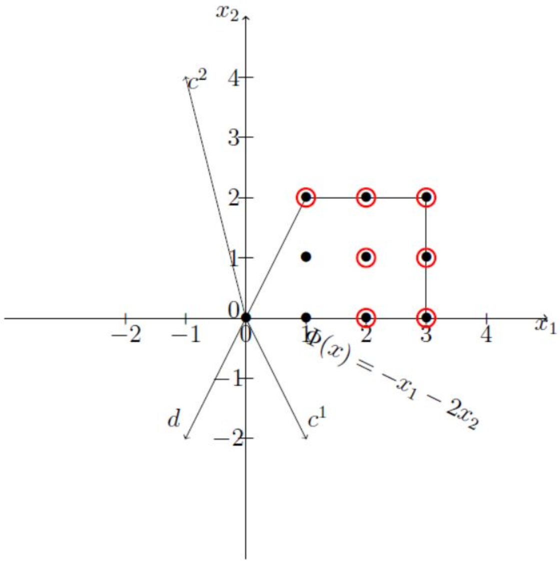


FIGURE 3.1 – L'espace des décisions.

Soit $x^1 = (0,0)$ une solution optimale de (P_R) dont le vecteur critères $Z(x^1) = (0,0)$.
étape 1 : Tester l'efficacité de x^1 en résolvant le problème suivant :

$$(P_{x^1}) \begin{cases} \max \Theta = \Psi_1 + \Psi_2 \\ x_1, x_2 \in D \\ x_1 - 2x_2 - \Psi_1 = 0 \\ -x_1 + 4x_2 - \Psi_2 = 0 \\ \Psi_i \geq 0, i = 1, 2. \end{cases} \tag{3.12}$$

à l'optimum $\Theta^* = 2$ avec les deux vecteurs suivants $\hat{x}^{1_1} = (2,1)$; $\Psi^{1_1} = (0,2)$ et $\hat{x}^{1_2} = (3,1)$; $\Psi^{1_2} = (1,1)$ donc $\Theta^* \neq 0$ c'est-à-dire que $x^1 = (0,0)$ n'est pas efficace, poser $\Phi_{sup} = \Phi(x^1) = 0$, et aller à l'étape 2.
étape 2 : Soit $\hat{x}^{1_1} = (2,1)$ une solution optimale de (P_{x^1}), qui est efficace, le vecteur critères correspondant est $Z(\hat{x}^1) = (0,2)$.
Résoudre le problème (T_1) tel que $(T_1) \equiv \max\{dx | Cx = C\hat{x}^1\}$ pour trouver les solutions efficaces ayant le même vecteur critère :

$$(T_1) \begin{cases} \max \Phi = -x_1 - 2x_2 \\ x_1, x_2 \in D \\ x_1 - 2x_2 = 0 \\ -x_1 + 4x_2 = 2 \end{cases}$$

La solution optimale de (T_1) est $\tilde{x}^1 = \hat{x}^1 = (2,1)$ et comme $\Phi(\tilde{x}^1) = -4 > \Phi_{inf} = -\infty$, poser $\Phi_{inf} = -4$ et $X_{opt} = \tilde{x}^1$.

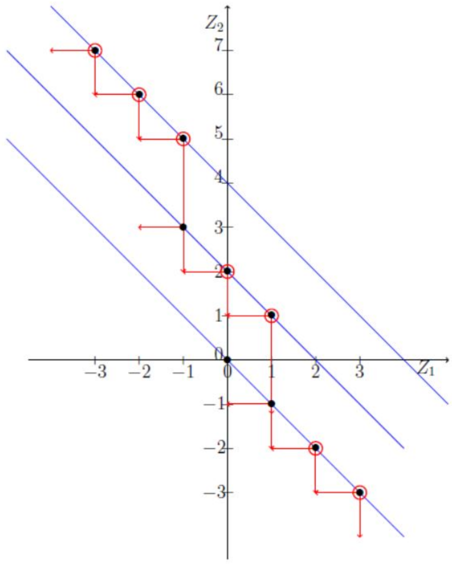


FIGURE 3.2 – Espace des critères.

$\Phi_{inf} \neq \Phi_{sup}$, aller à l'étape 3.

étape 3 : Résoudre le problème (R_1) tel que $(R_1) \equiv \max\{dx | x \in D - \bigcup_{s=1}^{l} D_s\}$ où $D_s = \{x \in \mathbb{Z}^n | Cx \leq C\tilde{x}^1\}$ avec $\tilde{x}^1$ est la solution optimale de problème (R_1) défini par :

$$(R_1) \begin{cases} \max \Phi = -x_1 - 2x_2 \\ x_1, x_2 \in D \\ C^i x \geq (C^i X_{opt} + 1) y_i^1 + M_i(1 - y_i^1), \quad i = 1, 2, ..., p \\ \sum_{j=1}^{2} y_i^1 \geq 1; \\ y_i^1 \in \{0, 1\}, \quad i = 1, 2 \end{cases}$$

qui est équivalent au problème suivant :

$$(R_1) \begin{cases} \max \Phi = -x_1 - 2x_2 \\ x_1, x_2 \in D \\ x_1 - 2x_2 \geq y_1^1 - 3(1 - y_1^1) \quad (1) \\ -x_1 + 4x_2 \geq 3y_2^1 - 3(1 - y_2^1) \quad (2) \\ y_1^1 + y_2^1 \geq 1, \quad y_1^1, y_2^1 \in \{0, 1\} \end{cases}$$

Soit $x^2 = (1,0), y = (1,0)$ une solution optimale de (R_1) et $Z(x^2) = (1, -1)$. Comme $\Phi(x^2) = -1 > \Phi_{inf}$, poser $l = l + 1 = 2$ et aller à l'étape 1.

Itération 2 : étape 1 : La solution x^2 est testée pour l'efficacité en résolvant le problème (P_{x^2})

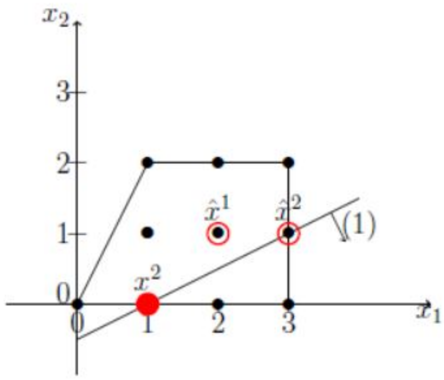


FIGURE 3.3 – La région réduite D^1.

$$(P_{x^2}) \begin{cases} \max \Theta = -x_1 - 2x_2 \\ x_1, x_2 \in D \\ x_1 - 2x_2 - \Psi_1 = 1 \\ -x_1 + 4x_2 - \Psi_2 = -1 \\ \Psi_i \geq 0, i = 1, 2. \end{cases}$$

A l'optimum $\Theta^* = 2$ avec le vecteur $\hat{x}^2 = (3, 1)$; $\Psi^3 = (0, 2)$ donc $\Theta^* \neq 0$ alors $x^2 = (1, 0)$ n'est pas efficace, poser $\Phi s_{sup} = \Phi(x^1) = -1$, et aller à l'étape 2.

étape 2 :

$\hat{x}^2 = (3, 1)$ une solution optimale de (P_{x^2}) qui est efficace, $Z(\hat{x}^2) = (1, 1)$

Résoudre le problème suivant :

$$(T_2) \begin{cases} \max \Phi = -x_1 - 2x_2 \\ x_1, x_2 \in D \\ x_1 - 2x_2 = 1 \\ -x_1 + 4x_2 = 1 \end{cases}$$

Sa solution est $\tilde{x}^2 = \hat{x}^2 = (3, 1)$.

Comme $\Phi(\hat{x}^2) = -5 < \Phi_{inf} = -4$, aller à l'étape 3 sans faire la mise à jour.

étape 3 : Résoudre le problème R_2

$$(R_2)\begin{cases} \max \Phi = -x_1 - 2x_2 \\ x_1, x_2 \in D \\ x_1 - 2x_2 \geq y_1^1 - 3(1 - y_1^1) \\ -x_1 + 4x_2 \geq 3y_2^1 - 3(1 - y_2^1) \\ y_1^1 + y_2^1 \geq 1, \quad y_1^1, y_2^1 \in \{0,1\} \\ x_1 - 2x_2 \geq 2y_1^2 - 3(1 - y_1^2) \quad (3) \\ -x_1 + 4x_2 \geq 2y_2^2 - 3(1 - y_2^2) \quad (4) \\ y_1^2 + y_2^2 \geq 1, \quad y_1^2, y_2^2 \in \{0,1\} \end{cases}$$

$x^3 = (2,0), y = (1,0,1,0)$ une solution optimale de (R_2) et $Z(x^3) = (2;-2)$. Comme $\Phi(\hat{x}^3 = -2 > \Phi_{inf} = -4$, poser $l = l + 1 = 3$ et aller à l'étape 1.

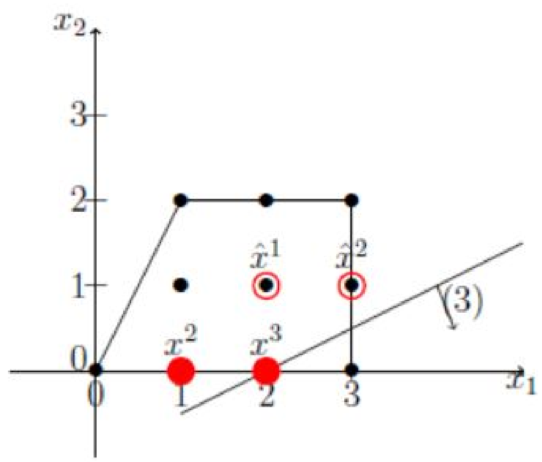


FIGURE 3.4 – La région réduite D^2.

Itération 3 :

étape 1 : La solution $x^3 = (2,0)$ est testée pour l'efficacité en résolvant le problème (P_x^3), on trouve qu'elle est efficace $(\Theta^* = 0)$,donc l'algorithme prend fin. $X_{opt} = (2,0)$ solution optimale de (P_E) et $\Phi_{opt} = -2$ est la valeur optimale du critère principal.

3.4 Conclusion

Dans ce chapitre on a présenté le problème, et nous sommes intéressés au problème de l'optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces d'un problème linéaire multi-objectifs en nombres entiers (MOILP), et les deux méthodes de résolution

Conclusion générale

En général, il y a toujours plusieurs objectifs à vouloir atteindre simultanément ce qui est impossible d'où l'idée de solution de compromis qui est propre aux problèmes d'optimisation multi-objectifs. L'optimisation multi-objectifs n'a pas encore fait toutes ses preuves que se soit du point de vue Applications ou du point de vue théorique et recherche, de sorte que beaucoup de pistes de recherche sont encore ouvertes et les champs d'applications sont très fertiles et ne demandent qu'à être exploités. Dans certaines situations le décideur préfère optimiser un critère tout à fait diffèrent des autres critères du problème multi-objectif et ce sur l'ensemble des solutions efficaces de ce dernier. C'est ce qu'on appelle l'optimisation sur l'ensemble des solutions efficaces et c'est sur cette classe de problèmes, que nous avons axé notre travail. Dans ce mémoire, on a procédé de la manière suivante : d'abord rappeler les notions de bases de l'optimisation mono-objectif et la programmation linéaire en nombres entiers, avec les méthodes de résolution comme les méthodes d'aproximation qui consiste a résoudre un problème mono-objectif sur un ensemble D en divisant ce domaine en polytope P, la méthode de branch and bound est une méthode qui divise l'ensemble admissible on sous ensemble pour chercher les soulution admissible en final de ce chapitre la méthode de branch and cut qui utilise le meme principe de branche and bound mais celle la cherche les solution admissible entiere puis dans le deuxième chapitre on à commencer par la programmation multi-objectifs où, nous avons donnée la terminologie et quelques notions de bases, ainsi quelques méthodes de résolution. la méthode de Bonson qui consiste à chercher des solutions dominantes sur l'ensemble des solutions réalisables et la méthode ϵ-contrainte, elle est la technique la plus connue pour résoudre les problèmes d'optimisation multicritères.Dans le troisième chapitre nous avons présenter le problème bi-niveau qui généralise le problème d'optimisation sur l'ensemble des solutions efficaces et les deux méthodes de résolution du problème dans le cas discret la méthode de Jesus M Jorg qui consiste à l'analyse d'un ordre appropri e des probl'emes lin eaires ennombres entiers pour eliminer successivement les solutions moins bonnes sur le crit'ere principal

Bibliographie

[1] Moncef Abbas and Djamal Chaabane. Optimizing a linear function over an integer efficient set. *European Journal of Operational Research*, 174(2) :1140–1161, 2006.

[2] Aicha Anzi, Mohammed Said Radjef, et al. *Resolution d'un probleme de programmation bi-niveaux lineaire par la methode DC.* PhD thesis, Universié de bejaia, 2009.

[3] Sanjeev Arora and Boaz Barak. *Computational complexity : a modern approach.* Cambridge University Press, 2009.

[4] Daniel Axehill. *Applications of integer quadratic programming in control and communication.* PhD thesis, Institutionen för systemteknik, 2005.

[5] Harold P Benson. Existence of efficient solutions for vector maximization problems. *Journal of Optimization Theory and Applications*, 26(4) :569–580, 1978.

[6] Harold P Benson. An all-linear programming relaxation algorithm for optimizing over the efficient set. *Journal of Global Optimization*, 1(1) :83–104, 1991.

[7] Harold P Benson. A finite, nonadjacent extreme-point search algorithm for optimization over the efficient set. *Journal of Optimization Theory and Applications*, 73(1) :47–64, 1992.

[8] HP Benson and S Sayin. Optimization over the efficient set : Four special cases. *Journal of Optimization Theory and Applications*, 80(1) :3–18, 1994.

[9] Gérard Berthiau and Patrick Siarry. État de l'art des méthodes "d'optimisation globale". *RAIRO-Operations Research*, 35(3) :329–365, 2001.

[10] Jerome Bracken and James T McGill. Mathematical programs with optimization problems in the constraints. *Operations Research*, 21(1) :37–44, 1973.

[11] Vira Chankong and Yacov Y Haimes. Optimization-based methods for multiobjective decision-making-an overview. *Large Scale Systems In Information And Decision Technologies*, 5(1) :1–33, 1983.

[12] Ionel Sorin CIUPERCA. Cours optimisation. *Cours à l'ISFA, en M1SAF, PP [16-17].*

[13] Jens Clausen. Branch and bound algorithms-principles and examples. *Department of Computer Science, University of Copenhagen*, pages 1–30, 1999.

[14] Yann Collette and Patrick Siarry. *Optimisation multiobjectif*. Editions Eyrolles, 2002.

[15] George B Dantzig. On the significance of solving linear programming problems with some integer variables. *Econometrica, Journal of the Econometric Society*, pages 30–44, 1960.

[16] Jerald P Dauer and Timothy A Fosnaugh. Optimization over the efficient set. *Journal of Global Optimization*, 7(3) :261–277, 1995.

[17] Clarisse Dhaenens. *Optimisation Combinatoire Multi-Objectif : Apport des méthodes coopératives et contribution à l'extraction de connaissances*. PhD thesis, Université des Sciences et Technologie de Lille-Lille I, 2005.

[18] Joseph G Ecker and IA Kouada. Finding all efficient extreme points for multiple objective linear programs. *Mathematical Programming*, 14(1) :249–261, 1978.

[19] Joseph G Ecker and Jung Hwan Song. Optimizing a linear function over an efficient set. *Journal of Optimization Theory and Applications*, 83(3) :541–563, 1994.

[20] Matthias Ehrgott. *Multicriteria optimization*, volume 491. Springer Science & Business Media, 2005.

[21] Mohamed Zeriab Es-Sadek. *Contribution à l'optimisation globale : approche déterministe et stochastique et application*. PhD thesis, Rouen, INSA, 2009.

[22] János Fülöp. A cutting plane algorithm for linear optimization over the efficient set. In *Generalized Convexity*, pages 374–385. Springer, 1994.

[23] Michael R Garey and David S Johnson. Crossing number is np-complete. *SIAM Journal on Algebraic Discrete Methods*, 4(3) :312–316, 1983.

[24] Arthur M Geoffrion. Proper efficiency and the theory of vector maximization. *Journal of mathematical analysis and applications*, 22(3) :618–630, 1968.

[25] Yacov Haimes. On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE transactions on systems, man, and cybernetics*, 1(3) :296–297, 1971.

[26] R Horst, NV Thoai, Yoshitsugu Yamamoto, and D Zenke. On optimization over the efficient set in linear multicriteria programming. *Journal of optimization theory and applications*, 134(3) :433–443, 2007.

[27] Heinz Isermann. Proper efficiency and the linear vector maximum problem. *Operations Research*, 22(1) :189–191, 1974.

[28] Jesús M Jorge. An algorithm for optimizing a linear function over an integer efficient set. *European Journal of Operational Research*, 195(1) :98–103, 2009.

[29] Narendra Karmarkar. A new polynomial-time algorithm for linear programming. In *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, pages 302–311, 1984.

[30] Taous Mernache. *Méthode adaptée pour la résolution des problèmes de programmation linéaire multiobjectif.* PhD thesis, Béjaia, Université Abderrahmane Mira. Faculté des Sciences et sciences des l . . ., 2007.

[31] John E Mitchell. Branch-and-cut algorithms for combinatorial optimization problems. *Handbook of applied optimization*, 1 :65–77, 2002.

[32] Nguyen. A review of some exchange algorithms for constructing discrete d-optimal designs. *Computational Statistics & Data Analysis*, 14(4) :489–498, 1992.

[33] Johan Philip. Algorithms for the vector maximization problem. *Mathematical programming*, 2(1) :207–229, 1972.

[34] Ralph E Steuer. Multiple criteria optimization. *Theory, computation and applications*, 1986.

[35] Yoshitsugu Yamamoto. Optimization over the efficient set : overview. *Journal of Global Optimization*, 22(1) :285–317, 2002.

Résumé

Dans le cadre de l'optimisation vectorielle post-optimale, nous nous sommes particulièrement intéressés à l'état de l'art de l'optimisation d'une fonction linéaire sur l'ensemble des solutions d'un problème multiobjectifs Eff dits efficaces. Notre premier objectif fut le rappel des notions fondamentales sur la programmation linéaire, et à objectifs multiples en passant en revue l'important de la littérature existant dans ce domaine. Ensuite, notre attention ses focalisée sur la maîtrise des concepts de l'optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces Eff en présentant la méthode célèbre existant dans la littérature due à Benson. Une étude qualitative sur les deux méthodes les plus récentes sur le problème de l'optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces d'un problème de programmation multi objectif dans le cas entier dû à jésus M. Jorge et Djamal Chaabane donnant ainsi beaucoup de perspectives quant aux d´développement d'autres approches de résolution de ce problème réputé difficile.

.

Abstract

In the context of post-optimal vector optimization, we are particularly interested in the state of the art of optimizing a linear function over the set of solutions of a multi-objective problem Eff called efficient. Our first goal was a reminder of the basics on continuous, discrete linear programming, and multiple objectives by reviewing the important of literature in this field. Then our attention was focused on mastering the concepts of optimizing a linear function over the set of solutions efficient Eff by presenting the existing method known in the literature due to Benson. A qualitative study on the two newest methods on the problem of optimizing a linear function over the efficient solutions of a problem of multi-objective programming in the world where due to jesus M JORRGE and Djamal CHAABANE giving many opportunities for the development of alternative approaches to solving this problem Known difficult.