

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



N°Réf :.....

Centre Universitaire
Abdelhafid Boussouf Mila

Institut des sciences et de la technologie

Département de Mathématiques et Informatique

Mémoire préparé en vue de l'obtention du diplôme de Master

en: Mathématiques

Spécialité : Mathématiques fondamentales et appliquées

L'amélioration de la méthode e-contrainte

Préparé par : BENHAMMADA Lokmane
AZZOUNE Ramadane

Soutenue devant le jury

Encadré par : ZAIDI Ali.....M.A.A
Président : BOUFELGHA Ibrahim.....M.A.A
Examineur : AZI Mourad.....M.A.A

Année universitaire : 2015/2016

TABLE DES MATIÈRES

Introduction	4
1 L'optimisation multiobjectif	7
1.1 Les problèmes d'optimisation	7
1.1.1 Fonction objective :	9
1.1.2 La Classification des problèmes d'optimisation	10
1.2 Problèmes d'optimisation multiobjectif	11
1.2.1 La dominance :	12
2 Les méthodes de résolution des problèmes d'optimisation multiobjectif	22
2.1 Les méthodes scalaires	22
2.1.1 La méthode de pondération des fonctions objectifs	22
2.1.2 La méthode du compromis(e-contraint)	27
2.2 Les méthodes interactives	32
2.2.1 La méthode du compromis par substitution	32
2.2.2 La méthode du simplex	37
2.3 Les méthodes floues	40
2.4 Les méthodes exploitant une métaheuristique	41
2.4.1 Qu'est-ce qu'une métaheuristique ?	41

2.4.2 Généralités	42
3 Amélioration de la Méthode de e-Contraint pour La programmation	
Multiobjectif	43
3.1 La méthode de e-Contraint	46
3.2 Obtenir les résultats sur l'efficacité propre	50
3.3 Création des contraintes inflexible	59
3.4 Amélioration de la méthode e-Contraint	65
3.5 Application Numérique	70
Conclusion générale	72
Bibliographie	75

TABLE DES FIGURES

1.1	Les différents Minima [29]	10
1.2	La présentation mathématique d'un problème d'optimisation mono-objectifs.[17]	11
1.3	Déformation d'une poutre subissant une contrainte [29]	12
1.4	L'optimalité locale au sens de Pareto	15
1.5	Le théorème du contact	16
1.6	Les niveaux de préférence dans la relation de dominance [29]	17
1.7	Représentation de la surface de compromis	18
1.8	Formes les plus courantes de la surface de compromis dans le cas de deux objectifs.	19
1.9	Représentation du point idéal et de point « nadir » [29]	20
1.10	La représentation de la surface de compromis. [29]	20
2.1	La méthode de pondération des fonctions objectif.	24
2.2	Une difficulté insurmontable pour la méthode de pondération des fonctions objectif.	25
2.3	La surface de compromis obtenue pour le problème F2 de Schaffer	27
2.4	La surface de compromis totale, avec et sans la contrainte $x \in [0, 2]$	28
2.5	La méthode du compromis.	29

❖ *TABLE DES FIGURES*

2.6	Représentation de la surface de compromis pour le problème Schaffer's F2, en choisissant les valeurs 0, 1, 2 et 3 pour ε	31
2.7	La résolution pas à pas du problème test.	32
2.8	Les zones de satisfaction dans la méthode SWT - Exemple 1.	35
2.9	Les zones de satisfaction dans la méthode SWT - Exemple 2.	37
2.10	Choix d'un nouveau point par symétrie.	38
2.11	Cheminement de la méthode du simplexe.	39
2.12	L'algorithme de la méthode du simplexe.	39
3.1	la méthode ε -contrainte Modifié pour un exemple bicritère	53
3.2	L'ensemble Réalisable et la fonction objective de $P_{\varepsilon-k}^-$	61
3.3	L'ensemble des variables	72
3.4	L'ensemble des critères	73

INTRODUCTION

De nombreux secteurs (mécanique, chimie, télécommunications, environnement, transport, etc.) sont concernés par les problèmes complexes de grande dimension et multicritères, mettant en jeu des coûts financiers très importants et pour lesquels les décisions doivent être prises de façon optimale. Les problèmes d'optimisation rencontrés en pratique sont rarement mono-objectifs. Il y a généralement plusieurs critères contradictoires à satisfaire simultanément. L'optimisation multiobjectif s'intéresse à la résolution de ce type de problèmes.

L'optimisation multiobjectif cherche à optimiser plusieurs composantes d'un vecteur de fonctions objectif. Contrairement au mono-objectif, le problème multiobjectif n'a pas une solution unique, mais un ensemble de solutions, connu comme l'ensemble des solutions Pareto-optimales. Toute solution de cet ensemble est optimale dans le sens qu'aucune amélioration ne peut être faite sur une composante sans dégradation au moins une autre composante du vecteur.

Compte tenu qu'une solution choisie par un décideur peut ne pas être acceptable par un autre, il s'avère utile de prévoir plusieurs alternatives au choix d'une solution Pareto optimale.

La première notion d'optimalité a été introduite par Francis Ysidro Edgeworth en 1881, elle a été utilisée de manière plus formelle par l'économiste italien Vilfredo Pareto, en 1896. Ces solutions sont appelées solutions de Pareto et l'ensemble des solutions que

l'on obtient à la fin de la recherche est la surface de compromis. Par la suite, Kuhan et Tucker en 1951 ont donné des résultats théoriques concernant les problèmes d'optimisation multicritère. C'est après avoir trouvé les solutions du problème multiobjectif que d'autres difficultés surviennent, il faut sélectionner une solution dans cette ensemble. La solution qui sera choisie doit refléter les compromis opérés par le décideur vis-à-vis des différentes fonctions objectif. Le décideur est tant "humain", il va faire des choix et l'un des buts de l'optimisation multiobjectif va être de modéliser les choix du décideur ou plutôt ses préférences. Pour modéliser ces choix, on pourra s'appuyer sur des théories ayant des approches différentes (méthodes de résolution des problèmes multiobjectifs).

Ce présent mémoire contient trois chapitres :

Dans le premier chapitre nous présentons l'optimisation multiobjectif, tout en introduisant des concepts fondamentaux tels que la dominance, la surface de compromis.

Le deuxième chapitre est entièrement consacré aux méthodes de résolution des problèmes multiobjectifs, nous avons présenté quelques méthodes et quelques exemples de résolution utilisant ces méthodes.

Dans le troisième chapitre nous présentons la méthode de résolution des contraintes élastiques qui est une Amélioration de la Méthode de e-Contraint, et puis on présente un exemple en utilisant cette méthode.

CHAPITRE 1

L'OPTIMISATION MULTIOBJECTIF

1.1 Les problèmes d'optimisation

Les problèmes d'optimisation occupent actuellement une place de choix dans la communauté scientifique. Non pas qu'ils aient été un jour considérés comme secondaires mais l'évolution des techniques informatiques a permis de dynamiser les recherches dans ce domaine.

Le monde réel offre un ensemble très divers de problèmes d'optimisation :

- Problème combinatoire ou à variables continues.
- Problèmes à un ou plusieurs objectifs.
- Problèmes statistiques ou dynamiques.
- Problème dans l'incertain.

Cette liste n'est évidemment pas exhaustive, et un problème peut être à la fois multiobjectifs et dynamique.[1]

Un tel problème est caractérisé par :

- **Un problème d'optimisation** est défini par **un espace d'état**, une ou plusieurs **fonction(s) objective(s)** et **un ensemble de contraintes**.

- **L'espace d'état** est défini par l'ensemble de domaines de définition des variables du problème.
- **Les variables** du problème peuvent être de nature diverse (réelle, entière, booléenne, etc.) et sont exprimées de données qualitatives ou quantitatives.
- **Une fonction objective** représente le but à atteindre pour le décideur (minimisation de coût, de durée, d'erreur, ...). Elle définit un espace de solutions potentielles au problème.
- **L'ensemble de contraintes** définit des conditions sur l'espace d'état que les variables doivent satisfaire. Ces contraintes sont souvent des contraintes d'inégalité ou d'égalité et permettent en général de limiter l'espace de recherche.
- **Une méthode d'optimisation** cherche le point ou un ensemble de points de l'espace de état possible qui satisfait au mieux un ou plusieurs critères. Le résultat est appelé valeur optimale ou optimum [1].

On peut dire qu'un problème d'optimisation se définit comme la recherche du minimum ou du maximum (de l'optimum donc) d'une fonction donnée. On peut aussi trouver des problèmes d'optimisation pour lesquels les variables de la fonction à optimiser sont des contraintes à évoluer dans une certaine partie de l'espace de recherche.

De cette définition de problème d'optimisation, il apparaît qu'un problème multiobjectifs ou multicritères peut être défini comme un problème dont on cherche l'action qui satisfait un ensemble de contraintes et optimise un vecteur de fonctions objectives.

La difficulté principale d'un problème multiobjectifs est qu'il n'existe pas de définition de la solution optimale. Le décideur peut simplement exprimer le fait qu'une solution est préférable à une autre mais il n'existe pas une solution meilleure que toutes les autres.

Ce besoin d'optimisation vient de la nécessité de l'ingénieur de fournir à l'utilisateur un système qui répond au mieux au cahier des charges.

Ce système devra être calibré de manière à :

1. Occuper le volume nécessaire à son bon fonctionnement (coût des matières premières),
2. Consommer le minimum d'énergie (coût de fonctionnement),
3. Répondre à la demande de l'utilisateur (cahier des charges).

1.1.1 Fonction objective :

C'est le nom donné à la fonction $f \in \mathbb{R}$ (on l'appelle encore fonction de coût ou critère d'optimisation). C'est cette fonction que l'algorithme d'optimisation va devoir « optimiser » (trouver un optimum).

Mathématiquement parlant, un problème d'optimisation se présentera sous la forme suivante :

$$\begin{cases} \min f(\vec{x}) & (\text{fonction à optimiser}) \\ \text{avec} & \vec{g}(\vec{x}) \leq 0 \quad (m \text{ contraintes d'inégalité}) \\ \text{et} & \vec{h}(\vec{x}) = 0 \quad (p \text{ contraintes d'égalité}) \end{cases}$$

On a $\vec{x} \in \mathbb{R}^n$, $\vec{g}(\vec{x}) \in \mathbb{R}^m$ et $\vec{h}(\vec{x}) \in \mathbb{R}^p$

Ici les vecteurs $\vec{g}(\vec{x}) \in \mathbb{R}^m$ et $\vec{h}(\vec{x}) \in \mathbb{R}^p$ représentent respectivement m contraintes d'inégalité et p contraintes d'égalité [29].

Variables de décision

Elles sont regroupées dans le vecteur \vec{x} . C'est en faisant varier ce vecteur que l'on cherche un optimum de la fonction f .

Minimum global

Un « point » \vec{x}^* est un minimum global de la fonction f si on a :

$$f(\vec{x}^*) < f(\vec{x}) \text{ Quel que soit } \vec{x}^* \text{ tel que } \vec{x}^* \neq \vec{x}.$$

Minimum local fort

Un « point » \vec{x}^* est un minimum local fort de la fonction f si on a :

$f(\vec{x}^*) < f(\vec{x})$ quel que soit $\vec{x} \in V(\vec{x}^*)$ et $\vec{x}^* \neq \vec{x}$, où $V(\vec{x}^*)$ définit un « voisinage » de \vec{x}^* .

Minimum local faible

Un « point » \vec{x}^* est un minimum local faible de la fonction f si on a :

$f(\vec{x}^*) \leq f(\vec{x})$ quel que soit $\vec{x} \in V(\vec{x}^*)$ et $\vec{x}^* \neq \vec{x}$, où $V(\vec{x}^*)$ définit un « voisinage » de \vec{x}^* . [29].

En résumant ces notions mathématiques représentant un problème d'optimisation multiobjectifs par la figure suivante :

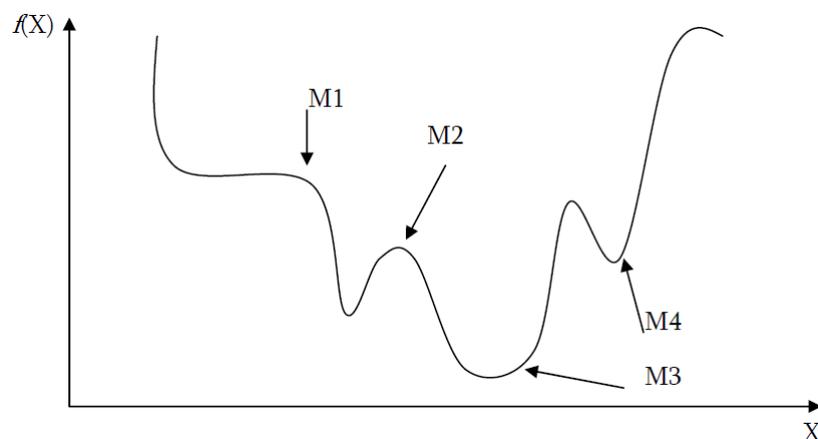


FIGURE 1.1 – Les différents Minima [29]

1.1.2 La Classification des problèmes d'optimisation

On peut classer les différents problèmes d'optimisation que l'on rencontre dans la vie courante en fonction de leurs caractéristiques :

1- Nombre de variables de décision :

✓ Une \Rightarrow mono variable

✓ Plusieurs \Rightarrow multi variable

2- Type de la variable de décision :

✓ Nombre réel continu \Rightarrow continu.

✓ Nombre entier \Rightarrow entier ou discret.

✓ Permutation sur un ensemble fini de nombres \Rightarrow combinatoire.

3- Type de la fonction objectif :

✓ Fonction linéaire des variables de décision \Rightarrow linéaire.

✓ Fonction quadratique des variables de décision \Rightarrow quadratique.

✓ Fonction non linéaire des variables de décision \Rightarrow non linéaire.

4- Formulation du problème :

✓ Avec contraintes \Rightarrow contraint.

✓ Sans contraintes \Rightarrow non contraint [29]

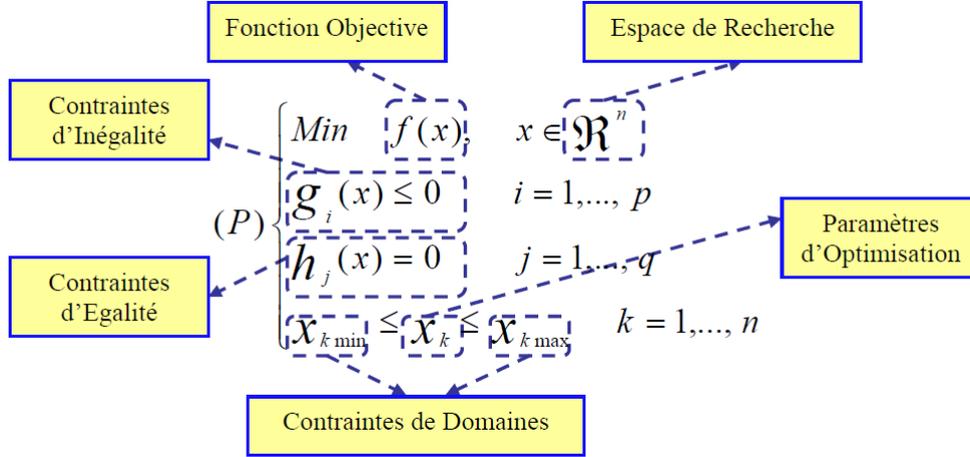


FIGURE 1.2 – La présentation mathématique d’un problème d’optimisation mono-objectifs.[17]

1.2 Problèmes d’optimisation multiobjectif

La formulation précédente était relative à un problème dans lequel on recherchait un optimum pour une fonction objectif (f dans l’expression précédente).

Cependant, lorsque l’on modélise un problème, on cherche souvent à satisfaire plusieurs objectifs. Par exemple, on veut un système performant et on veut aussi que ce système consomme peu. Dans ce cas, on parle de problème d’optimisation multiobjectif (ou problème d’optimisation multicritère). Celui-ci s’écrit de la manière suivante :

$$\left\{ \begin{array}{l} \text{minimiser} \quad \vec{f}(\vec{x}) \\ \text{avec} \quad \vec{g}(\vec{x}) \leq 0 \\ \text{et} \quad \vec{h}(\vec{x}) = 0 \\ \text{où} \quad \vec{x} \in \mathbb{R}^n, \vec{f}(\vec{x}) \in \mathbb{R}^k, \vec{g}(\vec{x}) \in \mathbb{R}^m, \vec{h}(\vec{x}) \in \mathbb{R}^p \end{array} \right.$$

On appellera ce problème **le problème P** dans tout le reste de l’ouvrage.

Comme on peut le voir ici, on n’a plus un seul objectif à atteindre, mais k (le vecteur \vec{f} regroupe k fonctions objectif).

Le but que l’on se fixe dans la résolution d’un problème d’optimisation multiobjectif est de minimiser “au mieux” les différents objectifs. Comme on va le voir dans le paragraphe suivant, dans un problème d’optimisation multicritère, on rencontre souvent des

objectifs contradictoires. Deux objectifs sont contradictoires lorsque la diminution l'un objectif entraîne une augmentation de l'autre objectif.

1.2.1 La dominance :

La plupart des problèmes d'optimisation réels sont décrits à l'aide de plusieurs objectifs ou critères souvent contradictoires devant être optimisés simultanément. Alors que, pour les problèmes n'incluant qu'un seul objectif, l'optimum cherché est clairement défini, celui-ci reste à formaliser pour les problèmes d'optimisation multiobjectifs. En effet, pour un problème à deux objectifs contradictoires, la solution optimale cherchée est un ensemble de points correspondant aux meilleurs compromis possibles pour résoudre notre problème [26] .

Lorsque l'on cherche à obtenir une solution optimale à un problème d'optimisation multiobjectifs donné, on s'attend souvent à trouver une solution et une seule. En effet, on rencontre rarement ce cas de figure. La plupart de temps, on trouve une multitude de solutions, du fait que certaines des objectifs sont contradictoires.

On effet, si l'on prend d'exemple de dimensionnement d'une poutre devant supporter une charge donnée, on va vouloir obtenir une poutre de section la plus petite possible, produisant la plus petite déformation possible, lorsque la charge repose sur le milieu de la poutre. Dans cet exemple (représenté à la figure 1.3) de manière intuitive, on s'aperçoit que répondre à l'objectif « poutre de petite section » ne va pas du tout dans le sens de répondre à l'objectif « petite déformation »

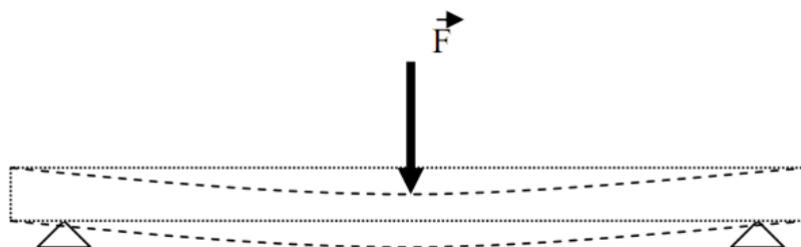


FIGURE 1.3 – Déformation d'une poutre subissant une contrainte [29]

Donc quand on résoudra un problème d'optimisation multiobjectifs, on obtiendra une grande quantité de solutions. Ces solutions comme on peut s'en douter, ne seront pas optimales, au sens où elles ne minimiseront pas tous les objectifs du problème.

Un concept intéressant, qui nous permettra de définir les solutions obtenues, est le **compromis**. En effet, les solutions que l'on obtient lorsqu'on a résolu le problème sont des solutions de compromis. Elles minimisent un certain nombre d'objectifs tout en dégradant les performances sur d'autres objectifs.

La dominance au sens de Pareto :

Comme la solution optimale est une multitude de points de \mathbb{R}^n , il est vital pour identifier ces meilleurs compromis de définir une relation d'ordre entre ces éléments. Dans le cas des problèmes d'optimisation multiobjectifs, ces relations d'ordre sont appelées relations de dominance. Plusieurs relations de dominance ont déjà été présentées : la dominance au sens de Geoffrion [29], la cône-dominance [29],...

Mais la plus célèbre et la plus utilisée c'est la dominance au sens de Pareto. Au *XIX^{ème}* siècle, Vilfredo Pareto, un mathématicien italien, formule le concept : « dans un problème multiobjectif, il existe un équilibre tel que l'on ne peut pas améliorer un critère sans détériorer au moins un des autres critères ».

De manière à définir clairement et formellement cette notion, les relations $=, \leq$ et $<$ usuelles sont étendues aux vecteurs.[26]

Soient u et v , deux vecteurs de même dimension.

$$v = u \quad \text{ssi} \quad \forall i \in \{1, 2, \dots, m\}, \quad u_i = v_i$$

$$v \leq u \quad \text{ssi} \quad \forall i \in \{1, 2, \dots, m\}, \quad u_i \leq v_i$$

$$v < u \quad \text{ssi} \quad u \leq v \wedge u \neq v$$

Les relations \leq et $>$ sont définies de manière analogue.

Les relations définies précédemment ne couvrent pas tous les cas possibles. En effet, il est impossible de classer les points $a = (1, 2)$ et $b = (2, 1)$ à l'aide d'une de ces relations.

Contrairement aux problèmes à un seul objectif où les relations usuelles $<$ pour comparer les points, elles sont insuffisantes pour comparer des points issus de problèmes multiobjectifs.

Nous définissons donc maintenant la relation de dominance au sens de Pareto permettant de prendre en compte tous les cas de figures rencontrés lors de la comparaison de deux points (ici des vecteurs).

Lorsque nous avons résolu un problème d'optimisation multiobjectifs, nous allons obtenu une multitude de solutions. Seul un nombre restreint de ces solutions va nous intéresser. Pour qu'une solution soit intéressante, il faut qu'il existe une relation de dominance entre la solution considérée et les autres solutions, dans le sens suivant :

Définition 1.1. *On dit que le vecteur \vec{x}_1 domine le vecteur \vec{x}_2 si :*

✓ \vec{x}_1 est au moins aussi bon que \vec{x}_2 dans tous les objectifs, et,

✓ \vec{x}_1 est strictement meilleur que \vec{x}_2 dans au moins un objectif

$x \succ y$ si

$$\forall i \in \{1, 2, \dots, k\} : f_i(x) \leq f_i(y) \quad \text{et} \quad \exists j \in \{1, 2, \dots, k\} : f_j(x) < f_j(y)$$

Les solutions qui dominent les autres mais ne se dominent pas entre elles sont appelées **solutions optimales au sens de Pareto** (ou solutions non dominées). Dans cet ensemble de solutions il existe deux définitions d'optimalité : l'optimalité locale et l'optimalité globale au sens de Pareto.

Définition 1.2. Optimalité locale au sens de Pareto

Un vecteur $\vec{x} \in \mathbb{R}^n$ est optimal localement au sens de Pareto s'il existe un réel $\delta > 0$ tel qu'il n'y ait pas de vecteur \vec{x}' qui domine le vecteur \vec{x} avec $\vec{x}' \in \mathbb{R}^n \cap B(\vec{x}, \delta)$, où $B(\vec{x}, \delta)$ représente une boule de centre \vec{x} et de rayon δ .

Définition 1.3. Optimalité globale au sens de Pareto

Un vecteur \vec{x} est optimal globalement au sens de Pareto (ou optimal au sens de Pareto) s'il n'existe pas de vecteur \vec{x}' tel que \vec{x}' domine le vecteur \vec{x} .

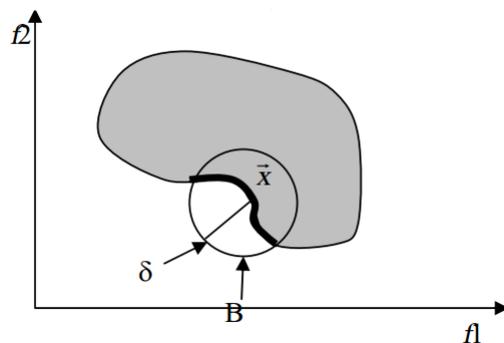


FIGURE 1.4 – L'optimalité locale au sens de Pareto

Définition 1.4. Le cône négatif

Un cône négatif est défini dans \mathbb{R}^k de la manière suivante :

$$C^- = \{\vec{x} | \vec{f}(\vec{x}) \in \mathbb{R}^k \text{ et } \vec{f}(\vec{x}) \leq 0\} \quad (\text{voir la figure 1.5})$$

La différence entre cette définition et celle de l'optimalité locale tient dans le fait que l'on ne considère plus une restriction de l'ensemble \mathbb{R}^n .

On dit aussi qu'un vecteur \vec{x} est optimal au sens de Pareto pour un problème d'optimisation multiobjectifs donné si :

$$(C^- + \vec{x}) \cap F = \{\vec{x}\} \Rightarrow \text{Le théorème de contact (figure 1.5)}$$

Où F désigne l'espace de solutions réalisables.

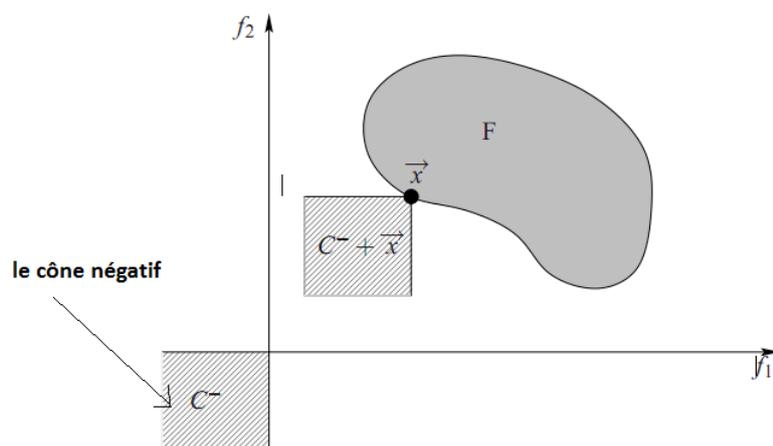


FIGURE 1.5 – Le théorème du contact

Lorsque l'on applique la définition de la dominance, on peut définir quatre régions auxquelles on peut attribuer des niveaux de préférence. Ces régions sont représentées à la figure 1.6. Cette figure reprend le découpage défini par le cône négatif que l'on a introduit précédemment et l'étend à tout l'espace.

Par exemple, si ce graphique est centré sur une solution A et que l'on compare cette solution avec une solution B , on aura les possibilités suivantes :

- Si la solution B se trouve dans le quadrant 1, alors la solution A est préférée à la solution B .
- Si la solution B se trouve dans le quadrant 3, alors la solution A est dominée par la solution B .
- Si la solution B se trouve dans l'un des quadrants 2 où 4, alors on ne peut pas prononcer sur la préférence de A par rapport à B ou B par rapport à A .

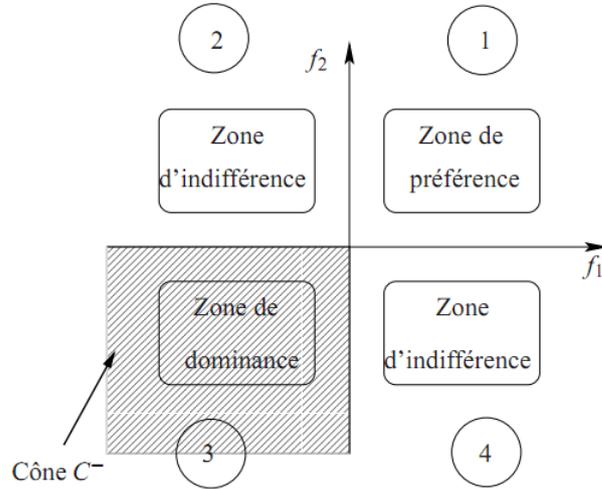


FIGURE 1.6 – Les niveaux de préférence dans la relation de dominance [29]

La dominance au sens de Geoffrion :

Une dernière forme de dominance importante dans le monde de l'optimisation multiobjectif est la dominance au sens de Geoffrion (voir [18] et [13]). Les solutions optimales obtenues par ce type de dominance sont appelées les solutions Pareto optimales propres.

Définition 1.5. Une solution $x^* \in S$ est appelée solution Pareto optimale propre si :

- elle est Pareto optimale,
- il existe un nombre $M > 0$ tel que $\forall i$ et $\forall x \in S$ vérifiant $f_i(x) < f_i(x^*)$, il existe un index j tel que $f_j(x^*) < f_j(x)$ et :

$$\frac{f_i(x^*) - f_i(x)}{f_j(x) - f_j(x^*)} \leq M$$

Cette relation n'est quasiment jamais utilisée telle quelle. En général, on utilise plutôt un résultat qui découle de cette définition. En effet, l'interprétation de ce théorème faite dans 18 est la suivante :

“Les solutions Pareto optimales propres ont des compromis bornés suivant leurs objectifs.”

Un théorème relatif à la méthode de pondération des fonctions objectif utilisant ce résultat est le suivant :

Théorème 1.1. Soit la méthode d'agrégation des fonctions objectif suivantes :

$$f_{eq}(\vec{x}) = \sum_{i=1}^N w_i \cdot f_i(\vec{x})$$

Supposons $\forall i = 1, \dots, N, w_i > 0$ avec $\sum_{i=1}^N w_i = 1$.

Si x^* est une solution optimale obtenue en utilisant la méthode d'agrégation ci-dessus, alors cette solution est aussi Pareto optimale propre.

La méthode de pondération des fonctions objectif, avec des poids qui respectent les relations ci-dessus, permet d'obtenir des solutions avec des compromis bornés.

Remarque 1.1. Ces différents types de relations de dominance permettent d'avoir suffisamment de degrés de liberté pour choisir une relation qui reproduise au mieux le comportement d'un ingénieur ou d'un décideur.

La surface de compromis :

Les solutions obtenues lors du classement basé sur le principe de dominance de Pareto forme ce que l'on appelle la surface de compromis (où la front de Pareto).

Imaginons que nous avons un problème à deux objectifs (minimiser f_1 et minimiser f_2 sous les contraintes $\vec{g}(\vec{x}) \leq 0$ et $\vec{h}(\vec{x}) = 0$:

- On appelle S l'ensemble de valeurs du couple $(f_1(\vec{x}), f_2(\vec{x}))$ quand à \vec{x} respecte les contraintes $\vec{g}(\vec{x})$ et $\vec{h}(\vec{x})$.
- On appelle P la surface de compromis.
- On représente S et P sur la figure suivante :

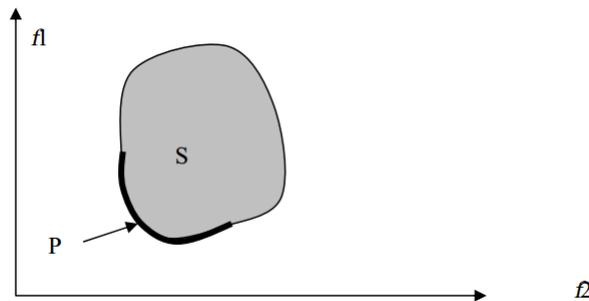


FIGURE 1.7 – Représentation de la surface de compromis

Une propriété est remarquable : en fonction du type du problème que l'on cherche à résoudre, on obtient une forme de surface de compromis. Les formes les plus courantes de surfaces de compromis sont réunies à la figure 1.8. Ces formes de surface de compromis sont typiques pour un problème d'optimisation multiobjectifs sur un ensemble de solutions convexes. C'est se type d'ensemble que l'on rencontre la plupart de temps.

On observe deux points caractéristiques associés à une surface de compromis :

Définition 1.6. Point idéal :

Les coordonnées de ce point sont obtenues en optimisant chaque fonction objective séparément. On dit aussi que les coordonnées du point idéal correspondent aux meilleures valeurs de chaque objectif des points de la frontière Pareto.

Définition 1.7. Point « nadir » :

Les coordonnées de ce point correspondent aux pires valeurs obtenues par chaque fonction objective lorsque l'on restreint l'espace des solutions à la surface de compromis.

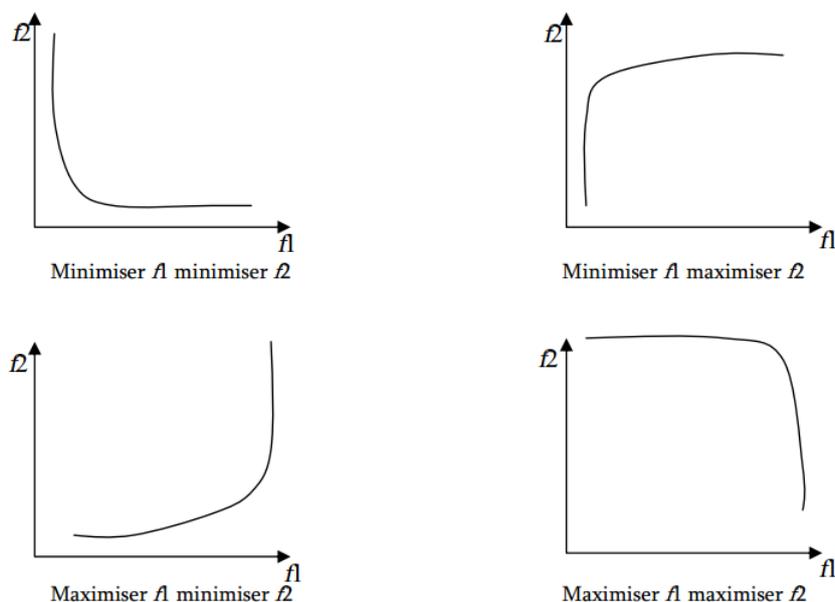


FIGURE 1.8 – Formes les plus courantes de la surface de compromis dans le cas de deux objectifs.

Le point idéal est utilisé dans beaucoup de méthodes d'optimisation comme point de référence. Le point nadir, lui, sert à restreindre l'espace de recherche; il est utilisé dans

certaines méthodes d'optimisation interactives.[29]

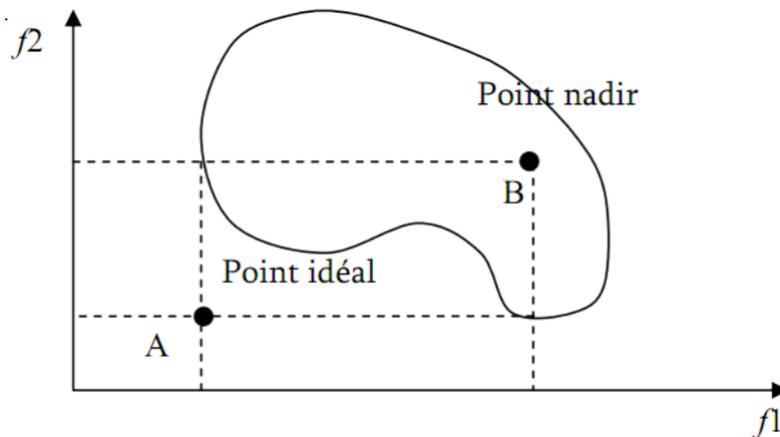


FIGURE 1.9 – Représentation du point idéal et de point « nadir » [29]

Représentation de la surface de compromis :

Toutes les représentations de la surface de compromis, pour un même problème, ne sont pas équivalentes. En effet, la représentation idéale de la surface de compromis devra être constituée de points solution de notre problème répartis de manière uniforme sur la surface de compromis (1.10 la représentation de la surface de compromis).

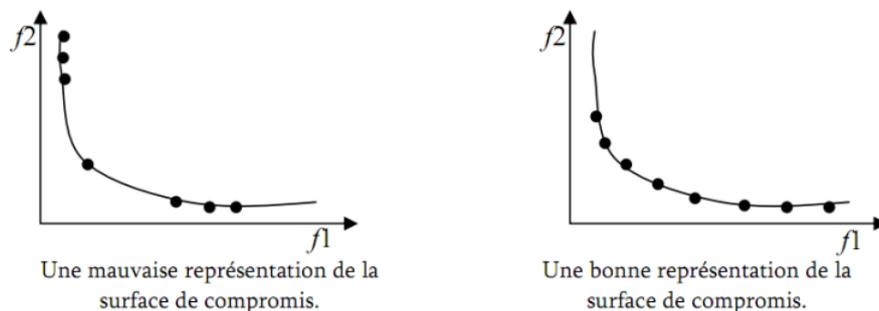


FIGURE 1.10 – La représentation de la surface de compromis. [29]

Dans le premier cas, les points, représentant la surface de compromis ne sont pas répartis de manière uniforme. L'utilisateur n'aura alors pas en sa possession un ensemble de solutions très utiles. En effet, s'il décide que la solution qu'il avait choisie ne lui convient

pas, le choix d'une autre solution risque de se faire varier brusquement tous ses objectifs, et cette nouvelle solution ne lui conviendra pas non plus. Il est alors probable que la solution offrant le « meilleur » compromis se trouve dans une zone qui ne soit pas représentée par des points solution.

CHAPITRE 2

LES MÉTHODES DE RÉOLUTION DES PROBLÈMES D’OPTIMISATION MULTIOBJECTIF

2.1 Les méthodes scalaires

2.1.1 La méthode de pondération des fonctions objectifs

Principe

Cette approche de la résolution d’un problème d’optimisation multiobjectif est la plus évidente. D’ailleurs, on appelle aussi cette méthode l’“approche naïve” de l’optimisation multiobjectif [3]. Le but, ici, est de revenir à un problème d’optimisation mono-objectif, dont il existe de nombreuses méthodes de résolution. La manière la plus simple de procéder consiste à prendre chacune des fonctions objectif, à leur appliquer un coefficient de pondération et à faire la somme des fonctions objectif pondérées. On obtient alors une nouvelle fonction objectif.

Présentation de la méthode

On part du problème P (voir page 11).

Le problème P se transforme de la manière suivante :

$$\begin{cases} \text{minimiser} & f_{eq}(\vec{x}) = \sum_{i=1}^k w_i f_i(\vec{x}) \\ \text{avec} & \vec{g}(\vec{x}) \leq 0 \\ \text{et} & \vec{h}(\vec{x}) = 0 \end{cases}$$

On a $\vec{x} \in \mathbb{R}^n, \vec{g}(\vec{x}) \in \mathbb{R}^m$ et $\vec{h}(\vec{x}) \in \mathbb{R}^p$

Fréquemment, les coefficients de pondération respectent la relation suivante :

$$\sum_{i=1}^k w_i = 1 \quad (2.1)$$

On peut représenter graphiquement le fonctionnement de la méthode sur un problème à deux objectifs. Le problème est le suivant :

$$\begin{cases} \text{minimiser} & f_1(\vec{x}) \\ \text{minimiser} & f_2(\vec{x}) \\ \text{avec} & \vec{g}(\vec{x}) \leq 0 \\ \text{et} & \vec{h}(\vec{x}) = 0 \end{cases}$$

Notre nouvelle fonction objectif aura pour expression :

$$f_{eq}(\vec{x}) = w_1 \cdot f_1(\vec{x}) + w_2 \cdot f_2(\vec{x}) \quad (2.2)$$

Ceci est l'expression d'une droite dans le plan f_1, f_2 .

En effet, si l'on cherche à minimiser $f_{eq}(\vec{x})$, on cherche en fait une constante C de l'équation de droite suivante la plus petite possible :

$$f_2(\vec{x}) = -\frac{w_1}{w_2} \cdot f_1(\vec{x}) + C \quad (2.3)$$

Cette équation de droite correspond à la courbe des isovaleurs de la fonction objectif équivalente.

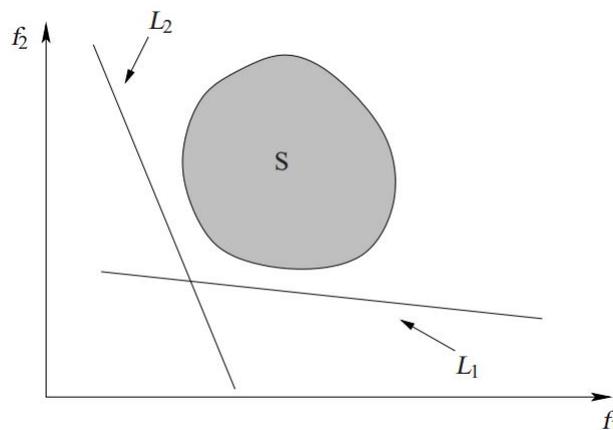


FIGURE 2.1 – La méthode de pondération des fonctions objectif.

Sur la figure 2.1, l'ensemble S correspond à l'ensemble des valeurs du couple $(f_1(\vec{x}), f_2(\vec{x}))$ respectant les contraintes définies par $\vec{g}(\vec{x})$ et $\vec{h}(\vec{x})$. Les droites L_1 et L_2 correspondent à deux couples de coefficients de pondération (w_1, w_2) différents.

Cette méthode consiste à “faire tangenter” la droite L_1 et la droite L_2 avec l'ensemble S . Le point de tangence est alors la solution recherchée.

Si l'on répète ce processus pour plusieurs valeurs des coefficients de pondération, les différentes solutions trouvées forment la surface de compromis.

Cette méthode n'est applicable qu'à des ensembles S convexes. Dans le cas contraire, elle ne permet pas de trouver la totalité de la surface de compromis. Par exemple, la portion en trait gras de la surface de compromis de la figure 2.2 ne peut être obtenue par cette méthode.

Un exemple concret

Exemple : Le problème de “Schaffer's F2”

Considérons le problème suivant à deux fonctions objectif :

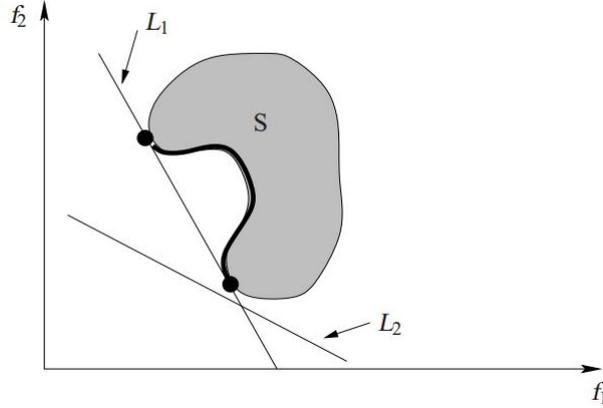


FIGURE 2.2 – Une difficulté insurmontable pour la méthode de pondération des fonctions objectif.

$$\begin{cases} \text{minimiser} & f_1(x) = x^2 \\ \text{minimiser} & f_2(x) = (x - 2)^2 \\ \text{avec} & x \in [0, 2] \end{cases}$$

C'est un problème test appelé "Schaffer's F2".

La représentation de l'ensemble solution de ce problème est différente de celle qui nous a servi à illustrer les différentes définitions de l'optimisation multiobjectif. En effet, si l'on cherche à exprimer f_2 en fonction de f_1 , on obtient une équation d'ellipse :

$$\begin{cases} f_1(\vec{x}) = x^2 \\ f_2(\vec{x}) = (x - 2)^2 = x^2 - 4x + 4 = f_1(\vec{x}) + 4 - 4x \end{cases}$$

On obtient finalement :

$$\begin{aligned} (f_2(\vec{x}) - f_1(\vec{x}) - 4)^2 &= (4x)^2 \\ f_1^2 + f_2^2 - 2f_1f_2 - 8f_1 - 8f_2 + 16 &= 0 \end{aligned}$$

De plus, comme on le verra lors de la résolution du problème d'optimisation, la surface de compromis (qui correspond normalement à une surface) est ici une portion de courbe et l'ensemble des couples (f_1, f_2) est une courbe (une ellipse dans cet exemple).

Nous avons choisi cet exemple atypique car il permet d'illustrer numériquement et simplement le fonctionnement de certaines méthodes d'optimisation multiobjectif. En sachant que le même type de démarche pourra être appliqué à un problème d'optimisation multiobjectif classique.

❖ CHAPITRE 2. LES MÉTHODES DE RÉOLUTION DES PROBLÈMES
D'OPTIMISATION MULTIOBJECTIF

La nouvelle fonction objectif s'écrit :

$$f_{eq}(x) = w_1 f_1(x) + w_2 f_2(x) \quad (2.4)$$

Les coefficients de pondération w_1 et w_2 respectent les conditions suivantes :

$$w_1 \text{ et } w_2 \geq 0,$$

$$w_1 + w_2 = 1$$

On cherche le minimum de la fonction objectif $f_{eq}(x)$. Il vérifie les conditions suivantes nécessaires à l'existence d'un minimum :

$$\frac{df_{eq}(x)}{dx} = 0$$

$$\frac{d^2 f_{eq}(x)}{dx^2} > 0$$

Cherchons les points de $f_{eq}(x)$ qui vérifient ces conditions :

$$\frac{df_{eq}(x)}{dx} = 2 \cdot x \cdot (w_1 + w_2) - 4 \cdot w_2$$

$$\frac{d^2 f_{eq}(x)}{dx^2} = 2(w_1 + w_2) > 0, \text{ car on a choisi } w_1 + w_2 = 1$$

Donc, le minimum se situe en :

$$x^* = \frac{2 \cdot w_2}{w_1 + w_2} = 2 \cdot w_2$$

On va maintenant calculer quatre points de la surface de compromis. On choisit quatre valeurs de w_1 variant de 0.2 à 0.8 par pas de 0.2. On en déduit les valeurs de w_2 , puis on calcule x^* , $f_1(x^*)$ et $f_2(x^*)$. On constate que les valeurs de x^* respectent la contrainte, puis on réunit ces valeurs dans le tableau 2.1 :

w_1	0.2	0.4	0.6	0.8
w_2	0.8	0.6	0.4	0.2
x^*	1.6	1.2	0.8	0.4
$f_1(x^*)$	2.56	1.44	0.64	0.16
$f_2(x^*)$	0.16	0.64	1.44	2.56

TABLE 2.1 – Tableau récapitulatif.

On trace maintenant ces points dans le plan f_1, f_2 et on les relie (voir figure 2.3)

On remarque que, dans cet exemple, on trouve une surface de compromis typique

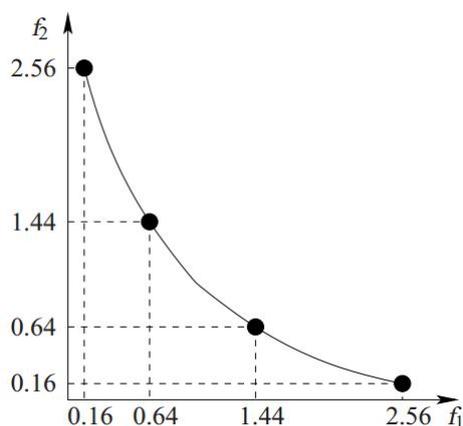


FIGURE 2.3 – La surface de compromis obtenue pour le problème F2 de Schaffer

dans les problèmes de minimisation sur tous les objectifs. A la figure 2.4 , on a représenté l'ensemble des valeurs réalisables du problème test, avec et sans la contrainte $x \in [0, 2]$.

2.1.2 La méthode du compromis(e-contraint)

Principe

Nous allons étudier des méthodes qui permettent de transformer un problème d'optimisation multiobjectif en un problème d'optimisation mono-objectif comportant quelques contraintes supplémentaires.

La démarche est la suivante :

- on choisit un objectif à optimiser prioritairement .

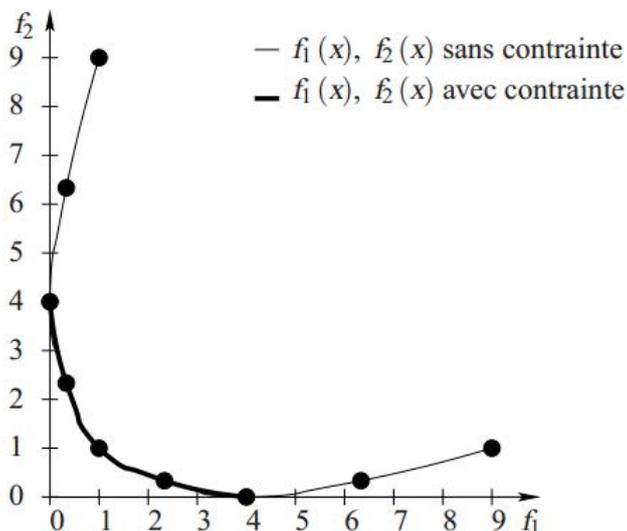


FIGURE 2.4 – La surface de compromis totale, avec et sans la contrainte $x \in [0, 2]$.

- on choisit un vecteur de contraintes initial .
- on transforme le problème en conservant l'objectif prioritaire et en transformant les autres objectifs en contraintes d'inégalité.

On appelle aussi cette méthode la méthode de la e-contrainte[14].

Présentation de la méthode

On part du problème P .

On choisit un vecteur de contraintes $\varepsilon_i, i \in \{2, 3, \dots, k\}, \varepsilon_i \geq 0$

On transforme le problème P de la manière suivante :

$$\left\{ \begin{array}{ll} \text{minimiser} & f_1(\vec{x}) \\ \text{avec} & f_2(\vec{x}) \leq \varepsilon_2 \\ & \vdots \\ & f_k(\vec{x}) \leq \varepsilon_k \\ & g(\vec{x}) \leq 0 \\ \text{et} & h(\vec{x}) = 0 \end{array} \right.$$

A la figure 2.5, on a représenté le fonctionnement de cette méthode pour un problème à deux objectifs.

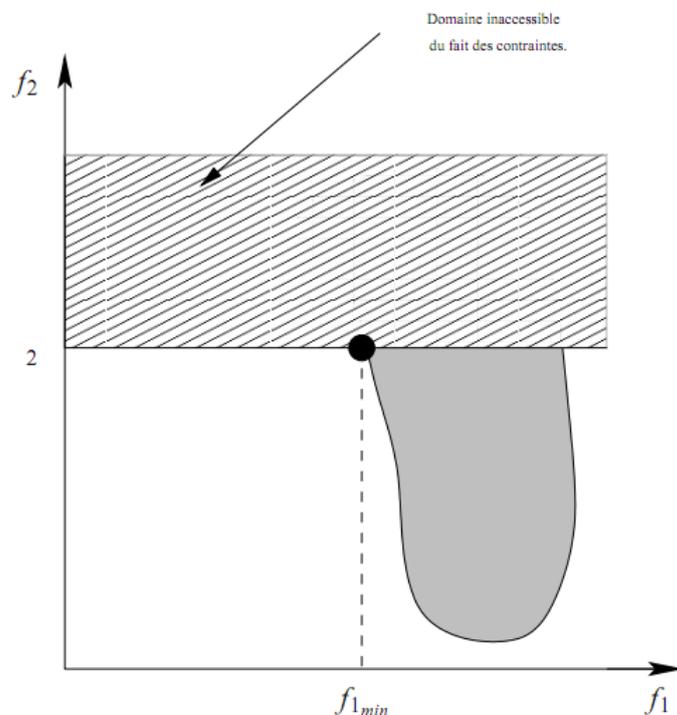


FIGURE 2.5 – La méthode du compromis.

Un exemple concret

On reprend l'exemple du problème "Schaffer's F2".

En appliquant la méthode décrite précédemment, on obtient le problème suivant :

$$\left\{ \begin{array}{ll} \text{minimiser} & f_1(x) = x^2 \\ \text{minimiser} & f_2(x) = (x - 2)^2 \leq \varepsilon \\ \text{avec} & x \in [0, 2] \end{array} \right.$$

On décide maintenant de résoudre ce problème en choisissant quatre valeurs pour la variable de ε : 0, 1, 2, 3.

On commence par réécrire la contrainte :

$$(x - 2)^2 \leq \varepsilon \Leftrightarrow 2 - \sqrt{\varepsilon} \leq x \leq 2 + \sqrt{\varepsilon}$$

Avec les valeurs de ε que nous avons choisies :

❖ CHAPITRE 2. LES MÉTHODES DE RÉOLUTION DES PROBLÈMES
D'OPTIMISATION MULTIOBJECTIF

$$0 < 2 - \sqrt{\varepsilon} \leq 2$$

Lorsque l'on met en relation cette expression avec $x \in [0, 2]$, On obtient la nouvelle contrainte :

$$x \in [2 - \sqrt{\varepsilon}, 2]$$

ε	contrainte	solution x^*	$f_1(x^*)$	$f_2(x^*)$
0	$x \in [2, 2]$	2	4	0
1	$x \in [1, 2]$	1	1	1
2	$x \in [2 - \sqrt{2}, 2]$	$2 - \sqrt{2}$	$6 - 4\sqrt{2}$	2
3	$x \in [2 - \sqrt{3}, 2]$	$2 - \sqrt{3}$	$7 - 4\sqrt{2}$	3

On représente les différentes solutions dans le plan (f_1, f_2) (voir figure 2.6).

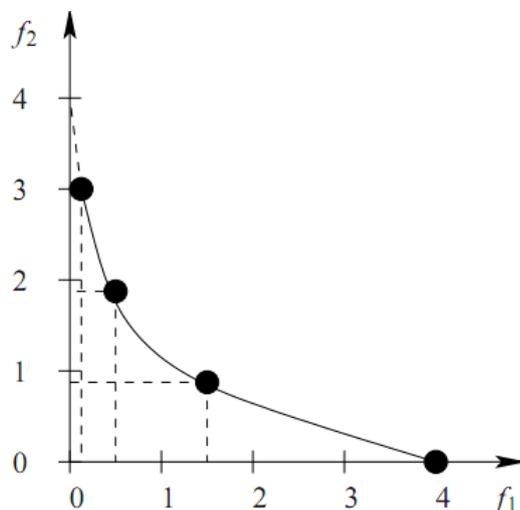


FIGURE 2.6 – Représentation de la surface de compromis pour le problème Schaffer's F2, en choisissant les valeurs 0, 1, 2 et 3 pour ε

A la figure 2.7, on a représenté la démarche qui a été suivie pour résoudre le problème test. On voit, sur cette figure l'effet de la modification de la borne de notre nouvelle contrainte. De plus, on voit apparaître, pas à pas, les solutions de notre problème et, par la même occasion, la surface de compromis.

Pour des problèmes simples, le choix des valeurs de ε_i peut donner une bonne répartition des solutions sur la surface de compromis.

En revanche, dans la plupart des cas concrets, le choix (arbitraire) des valeurs de ε_i ne permet pas d'obtenir une bonne répartition des solutions sur la surface de compromis (c'est le cas pour notre exemple dans lequel il manque un point entre les valeurs 2 et 3 de la coordonnée f_1). Alors, on peut ne pas avoir assez de points et rencontrer de grandes difficultés pour extrapoler l'allure de la surface de compromis.

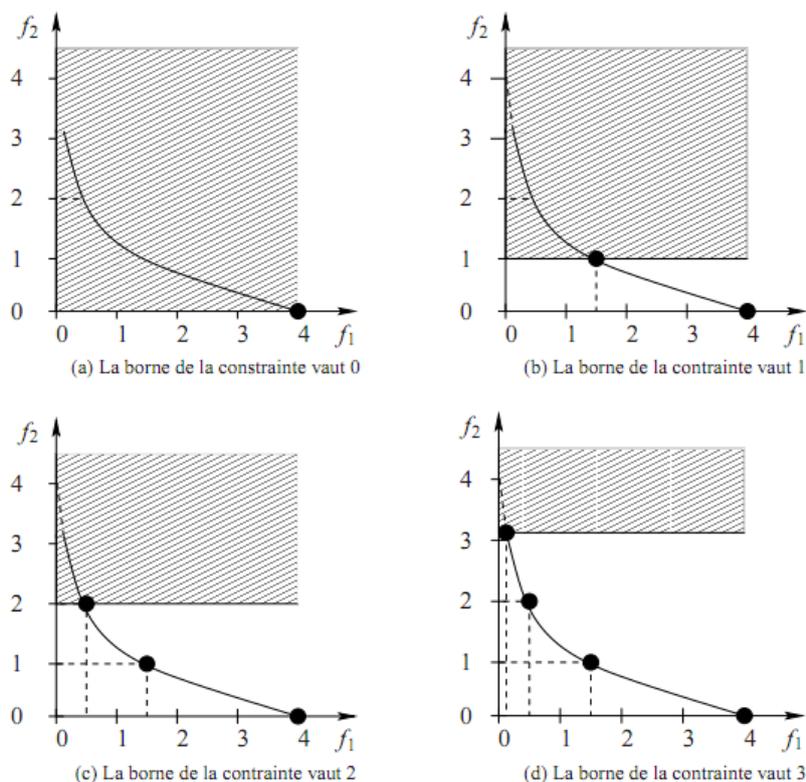


FIGURE 2.7 – La résolution pas à pas du problème test.

2.2 Les méthodes interactives

2.2.1 La méthode du compromis par substitution

Principe

Cette méthode (Surrogate Worth Tradeoff (SWT) ou “méthode du compromis par substitution”) a été très utilisée pour l’optimisation de ressource en eau [30]. Elle est basée sur la méthode du compromis, à laquelle on a ajouté un processus interactif, pour que la méthode converge vers la solution la plus susceptible de satisfaire l’utilisateur.

Présentation de la méthode

On part du problème P

Cette méthode se décompose en sept étapes :

Étape 1 :

Trouver le minimum de la fonction f_j en résolvant

$$\begin{cases} \text{minimiser} & f_j(\vec{x}) \\ \text{avec} & \vec{g}(\vec{x}) \leq 0 \\ \text{et} & \vec{h}(\vec{x}) = 0 \end{cases}$$

La solution à ce problème devient la j^e composante du vecteur \vec{f}_{min} qui regroupe les $k - 1$ valeurs minimal de $f_j, j = \{2, 3, \dots, k\}$. Si possible, on peut chercher à cette étape les composantes du vecteur \vec{f}_{max} , qui regroupe les $k - 1$ valeurs maximales de $f_j, j = \{2, 3, \dots, k\}$.

Étape 2 :

Choisir la valeur de départ de $\varepsilon_j \geq f_{j_{min}}, j = \{2, 3, \dots, k\}$

Étape 3 :

Résoudre le problème suivant :

$$\begin{cases} \text{minimiser} & f_1(\vec{x}) \\ \text{avec} & f_2(\vec{x}) \leq \varepsilon_2, \dots, f_k(\vec{x}) \leq \varepsilon_k \\ & \vec{g}(\vec{x}) \leq 0 \\ \text{et} & \vec{h}(\vec{x}) = 0 \end{cases}$$

Si certaines des contraintes de ce problème ne peuvent pas être respectées, on pose $\varepsilon_j = f_j(\vec{x})$ pour j correspondant aux index des contraintes non respectées (on relâche les contraintes). Il faut ensuite recommencer cette étape. Si les contraintes sont respectées, on appelle \vec{x}^* le vecteur solution et on passe à l'étape suivante.

Étape 4 :

Si les contraintes sont maintenant suffisamment relâchées, on passe à l'étape 5 sinon, on choisit une nouvelle valeur de $\varepsilon_j \geq f_{j_{min}}$ pour tout $j = \{2, 3, \dots, k\}$ et on retourne à l'étape 3.

Une méthode pour sélectionner une nouvelle valeur de ε consiste à commencer par une

grande valeur de ε puis à diminuer chaque j , pour tout $j = \{2, 3, \dots, k\}$ d'un certain $\Delta_j > 0$, chaque fois que les contraintes sont respectées.

Si les contraintes ne sont pas respectées, on pose $\varepsilon_j = f_j(\vec{x})$ avec j index des contraintes non respectées (ici \vec{x} désigne la valeur courante de la solution).

Étape 5 :

On demande maintenant à l'utilisateur de choisir les coefficients $W_{1,j}$, pour tout $j = \{2, 3, \dots, k\}$. Ces coefficients représentent l'avis de l'utilisateur vis-à-vis d'un accroissement de la fonction f_j de λ_{1j} unités (le mode de calcul de λ_{1j} est précisé plus loin). Ce coût est mesuré sur une échelle de -10 à $+10$, où -10 représente un avis défavorable de l'utilisateur vis-à-vis de cet accroissement et $+10$ un avis favorable. 0 représente l'indifférence.

Cette opération est répétée pour j variant de 2 à k

Étape 6 :

On répète l'étape 5 jusqu'à ce que l'on trouve pour les coefficients $W_{1,j}, j = \{2, 3, \dots, k\}$ des valeurs égales à zéro. On note f_j^* , pour tout $j = \{2, 3, \dots, k\}$ les valeurs des fonctions objectif correspondant à ces coefficients.

Étape 7 :

Le vecteur solution préféré \vec{x}^* est déterminé en résolvant le problème suivant :

$$\left\{ \begin{array}{ll} \text{minimiser} & f_1(\vec{x}) \\ \text{avec} & f_2(\vec{x}) = f_2^*, \dots, f_k(\vec{x}) = f_k^* \\ & \vec{g}(\vec{x}) \leq 0 \\ \text{et} & \vec{h}(\vec{x}) = 0 \end{array} \right.$$

Le plus difficile dans cette méthode est de bien comprendre la signification du coefficient $W_{1,j}$. Pour cela, voici une procédure que l'on peut suivre pour évaluer $W_{1,j}$:

Un point possible pour la résolution du problème est le suivant :

- $f_2(\vec{x})$ [unités de f_2],
- $f_3(\vec{x})$ [unités de f_3] et
- $f_1(\vec{x})$ [unités de f_1]

On se pose alors les questions suivantes :

pour $W_{1,2}$:

A ce point de fonctionnement, seriez-vous prêt à augmenter f_2 de $\lambda_{1,2}$ [unités de f_2] pour une réduction de f_1 de 1 [unité de f_1] ? Donnez votre réponse sur une échelle de -10 (pas d'accord) à +10 (d'accord).

Pour $W_{1,3}$:

A ce point de fonctionnement, seriez-vous prêt à augmenter f_3 de $\lambda_{1,3}$ [unités de f_3] pour une réduction de f_1 de 1 [unité de f_1] ? Donnez votre réponse sur une échelle de -10 (pas d'accord) à +10 (d'accord).

Cette méthode va permettre, dans un premier temps, de trouver une zone de satisfaction (entre deux 0 de $W_{1,2}$ ou de $W_{1,3}$, ou entre un 0 de $W_{1,2}$ et un 0 de $W_{1,3}$. (Voir les figures 2.8 et 2.9).

Une fois que l'on a trouvé cette zone de satisfaction, on peut résoudre le problème posé en utilisant la méthode du compromis, en se restreignant aux valeurs de la zone de satisfaction.

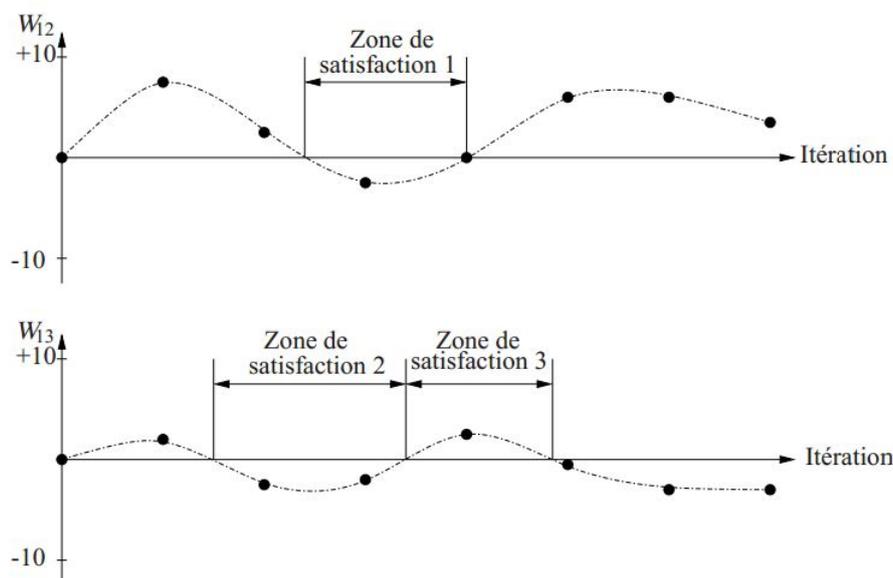


FIGURE 2.8 – Les zones de satisfaction dans la méthode SWT - Exemple 1.

La valeur du coefficient λ_{ij} se calcule de la manière suivante :

$$\lambda_{ij} = -\frac{\partial f_i}{\partial f_j} \quad (2.5)$$

Dans notre cas, on ne calcule que :

$$\lambda_{1j} = -\frac{\partial f_1}{\partial f_j} \quad (2.6)$$

Car on fait toujours notre comparaison en référence à la fonction objectif f_1 .

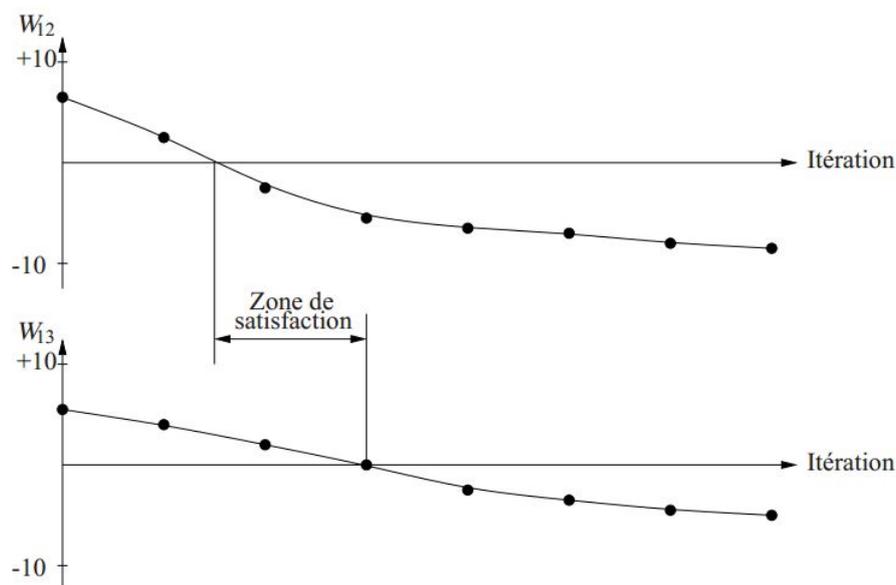


FIGURE 2.9 – Les zones de satisfaction dans la méthode SWT - Exemple 2.

Discussion

Dans cette méthode, il n'est pas toujours évident de savoir quel prix on est prêt à payer un accroissement de la valeur d'une fonction. Dans la plupart des cas, on peut être tenté de répondre un peu au hasard. Ce type de réponse peut alors empêcher la méthode de trouver le ou les bons optima.

De plus, cette méthode n'est applicable qu'à des problèmes dont la fonction objectif de référence est dérivable (à cause du coefficient $\lambda_{1,j}$).

2.2.2 La méthode du simplexe

Principe :

Cette méthode n'a rien à voir avec la méthode du simplexe utilisée en programmation linéaire. Il s'agit là d'une méthode de recherche séquentielle de l'optimum d'un problème d'optimisation. Le logiciel [19] réalise cette optimisation pour un problème d'optimisation multiobjectif de manière interactive.

Cette méthode utilise $k + 1$ essais (où k représente la dimension de la variable de décision \vec{x}) pour définir une direction d'amélioration des fonctions objectif.

Présentation de la méthode :

On commence par choisir au hasard $k + 1$ valeurs pour la variable de décision \vec{x} . L'algorithme évalue alors les $k + 1$ points et supprime le point le moins "efficace". Il crée alors un nouveau point à partir du point supprimé (voir figure 2.10) et recommence l'évaluation. L'algorithme comporte quelques règles, qui permettent de choisir des points qui évitent de tourner autour d'une mauvaise solution. Les deux principales règles sont les suivantes :

Règle 1 : **Rejeter les pires solutions.**

Une nouvelle position de la variable de décision \vec{x} est calculée par réflexion de la position rejetée (voir la figure 2.10). Après cette transformation, on recherche le nouveau pire point. La méthode est alors répétée en éliminant ce point, etc. A chaque étape, on se rapproche de la zone où se trouve l'optimum recherché.

Règle 2 : **Ne jamais revenir sur un point qui vient juste d'être rejeté.**

Sans cette règle, l'algorithme pourrait osciller entre deux "mauvais" points.

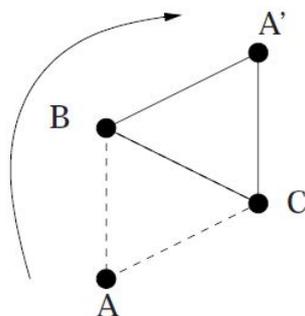


FIGURE 2.10 – Choix d'un nouveau point par symétrie.

Sur la figure 2.11, on peut voir le cheminement de la méthode dans l'espace des variables de décision, dans le cas d'un exemple à deux variables de décision. Sur cette figure sont représentées des courbes de niveau représentant la "direction" d'amélioration de la fonction objectif à optimiser. Dans cet exemple, l'optimisation se déroule de manière à maximiser le pourcentage correspondant à la courbe de niveau.

Présentons maintenant l'algorithme complet de la méthode du simplexe. Nous utiliserons les notations suivantes :

❖ CHAPITRE 2. LES MÉTHODES DE RÉOLUTION DES PROBLÈMES D'OPTIMISATION MULTIOBJECTIF

W : point le moins favorable ou point venant d'être rejeté.

B : point le plus favorable.

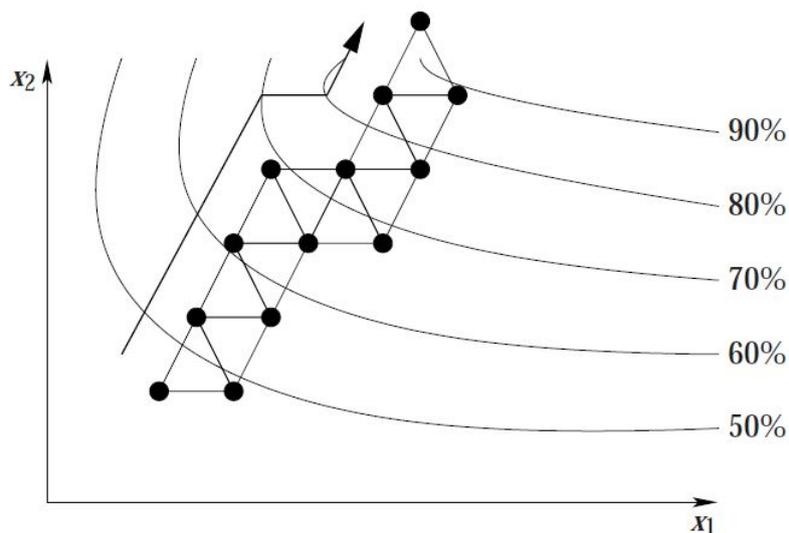


FIGURE 2.11 – Cheminement de la méthode du simplexe.

N : second point le plus favorable.

L'algorithme du simplexe est présenté à la figure 2.12.

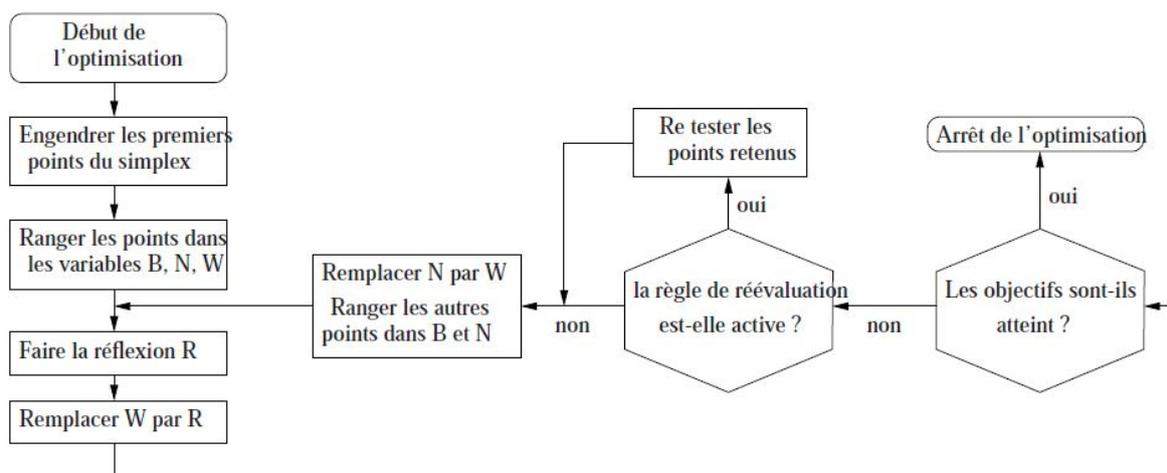


FIGURE 2.12 – L'algorithme de la méthode du simplexe.

A \ B	0	1
0	0	0
1	0	1

TABLE 2.2 – La fonction ET.

2.3 Les méthodes floues

Dans la vie courante, tout ne peut pas être décrit de manière binaire. Par exemple, la transition entre le jour et la nuit se fait progressivement, l'action sur l'embrayage d'un véhicule est, elle aussi, progressive.

Longtemps, le seul outil de description en logique était binaire. Tout en logique a été décrit en termes de VRAI ou FAUX. Le problème de cette description simpliste en logique est qu'elle ne permet pas de traiter l'incertitude et l'imprécision des connaissances humaines.

L'automaticien L. A. Zadeh a élaboré une nouvelle logique basée sur les ensembles flous. Elle permet de traiter l'imprécision et l'incertitude dans la connaissance humaine ainsi que les transitions progressives entre états.

La différence principale entre une logique classique et cette logique floue est l'existence d'une transition progressive entre le VRAI et le FAUX [16].

Parallèle avec la logique classique

Pour mieux comprendre le concept de la logique floue, nous allons transposer les concepts de la logique classique en logique floue.

Tout d'abord, la logique classique possède trois fonctions de base. Ces fonctions travaillent sur des paramètres binaires (0 ou 1, FAUX ou VRAI). Voici ces fonctions de base :

– La fonction ET : $C = A$ et B , ou, en termes plus mathématiques, $C = A.B$. La table de vérité de cette fonction est représentée dans le tableau 2.2.

– La fonction OU : $C = A$ ou B ou, en termes plus mathématiques, $C = A + B$. La table de vérité de cette fonction est représentée dans le tableau 2.3.

– La fonction NON : $C =$ non A ou, en termes plus mathématiques, $C = \bar{A}$. La table de

A \ B	0	1
0	0	1
1	1	1

TABLE 2.3 – La fonction OU.

A	C
0	1
1	0

TABLE 2.4 – La fonction NON.

vérité de cette fonction est représentée dans le tableau 2.4.

Toutes ces fonctions de la logique classique ont leur équivalent en logique floue. La grande différence vient du fait que l'on travaille sur une variable logique qui varie de manière continue de 0 à 1.

Les équivalents en logique floue des fonctions de la logique classique sont représentés dans le tableau 2.5.

Les méthodes de logique floue sont généralement simple à mettre en œuvre. Elles donne de bons résultats sur les problèmes multiobjectif.

2.4 Les méthodes exploitant une métaheuristique

2.4.1 Qu'est-ce qu'une métaheuristique ?

Les métaheuristices sont des méthodes générales de recherche dédiées aux problèmes d'“optimisation difficile” [25]. Ces méthodes sont, en général, présentées sous la forme de

Logique classic	logique floue
C=A et B	C=min(A,B)
C=A ou B	C=max(A,B)
C= non A	C=1-A

TABLE 2.5 – Équivalence entre logique classique et logique floue.

concept. elles reprennent des idées que l'on retrouve parfois dans la vie courante. Les principales métaheuristiques sont le recuit simulé, la recherche tabou et les algorithmes génétiques.

2.4.2 Généralités

Si l'on considère l'ensemble des méthodes de recherche d'optimum, on constate que celles-ci se réunissent autour de quelques concepts généraux. Par exemple, on trouvera beaucoup de méthodes (méthode de Newton, quasi-Newton, etc.) articulées autour de la descente de gradient. Le concept général de cette famille sera la descente de gradient, et ce concept général sera appelé métaheuristique.

Une métaheuristique est donc une méthode très générale, qui nécessite quelques transformations (mineures en général) avant de pouvoir être appliquée à la résolution d'un problème particulier.

Si l'on considère les différentes métaheuristiques, on constate que celles-ci se répartissent en trois familles :

- Les méthodes déterministes de recherche d'optimum local :

Ces méthodes convergent rapidement mais, la plupart du temps, elles ne trouvent pas l'optimum global. Elles se contentent de trouver un optimum local et ne reposent pas sur un processus stochastique pour la recherche de l'optimum .

- Les méthodes déterministes de recherche d'optimum global :

Ces méthodes permettent de trouver un optimum global rapidement et ne reposent pas sur un processus stochastique pour la recherche de l'optimum .

- Les méthodes stochastiques de recherche d'optimum global :

Ces méthodes reposent sur un processus stochastique chargé d'effectuer la recherche de l'optimum. Elles sont moins performantes (du point de vue rapidité) que les méthodes déterministes, mais elles peuvent trouver, en principe, un optimum global difficile à atteindre.

CHAPITRE 3

AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

Résumé

Dans cette partie, nous révisons l'une des plus importantes techniques de scalarisation utilisées dans la programmation multiobjectif, la méthode e-contrainte. Nous résumons cette dernière et nous indiquons quelques faiblesses de cette méthode, à savoir l'absence de conditions faciles pour vérifier les solutions efficaces et propres et l'inflexibilité des contraintes. Nous présentons deux modifications qui répondent à ces faiblesses en incluant les variables d'écart dans la formulation et dans l'élastification des contraintes nous incluons des variables artificielles.

La méthode e-contrainte améliorée que nous proposons dans cette partie combine les deux modifications.

La programmation multiobjectif est une partie de la programmation mathématique qui traite les problèmes de décision caractérisés par des fonctions objectifs multiples et contradictoires optimisés sur un ensemble réalisable de décision. Ces problèmes, appelés

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

les problème multiobjectifs (MOPs). Ils sont couramment rencontrés dans nombreux domaines , comme l'ingénierie, la gestion et d'autres.

Plus précisément, soit \mathbb{R}^n et \mathbb{R}^p des espaces vectoriels euclidiens appelés l'espace de décision et l'espace objectif. Soit $X \subset \mathbb{R}^n$ un ensemble réalisable non vide et f une fonction objective $f : \mathbb{R}^n \rightarrow \mathbb{R}^p$, $f = (f_1, \dots, f_p)$ où $f_k : \mathbb{R}^n \rightarrow \mathbb{R}$ pour $k = 1, \dots, p$. Un problème multiobjectif (MOP) est donnée par :

$$(MOP) \quad \min\{(f_1(x), \dots, f_p(x)) : x \in X\}.$$

Nous supposons généralement que l'ensemble X est donné implicitement sous la forme de contraintes, i.e :

$$X = \{x \in \mathbb{R}^n : g_j(x) \leq 0, j = 1, \dots, l; h_j(x) = 0, j = 1, \dots, m\}.$$

Nous définissons l'ensemble des points réalisables ou vecteurs objectifs pour toutes les solutions possibles $x \in X$ dans l'espace objectif $Y := f(X) \subset \mathbb{R}^p$.

Le symbole "min" dans le problème MOP est généralement introduire comme la recherche optimale ou les points préférée en Y et leurs pré-images dans X , où la préférence entre les vecteurs objectifs résulte d'une relation d'ordre définie sur Y . La notation suivante est utilisé pour définir les relations d'ordre sur \mathbb{R}^p . Pour $y, y' \in \mathbb{R}^p$,

- $y < y'$ noté $y_k < y'_k$ pour tout $k = 1, \dots, p$,
- $y \leq y'$ noté $y_k \leq y'_k$ pour tout $k = 1, \dots, p$,
- $y \leq y'$ noté $y \leq y'$ mais $y \neq y'$.

Le concept de l'optimalité de Pareto ou de l'efficacité est basé sur ces relations d'ordre et peut être définie comme dans la Définition 3.1.

Définition 3.1. *Considérons le problème MOP. Un point $x \in X$ est appelé :*

- (i) *une solution faiblement efficace lorsque $\nexists x' \in X$ tel que $f(x') < f(x)$;*
- (ii) *une solution efficace lorsque $\nexists x' \in X$ tel que $f(x') \leq f(x)$;*
- (iii) *une solution strictement efficace lorsque $\nexists x' \in X, x' \neq x$, telle que $f(x') \leq f(x)$.*

On note les ensembles des solutions faiblement efficace, efficaces et strictement efficaces par $X_{\omega F}$, X_E , X_{sE} , respectivement. Nous appelons leurs images les points faible non

dominée (Pareto faibles) et les points non dominée (Pareto), respectivement. Ils sont noté par $Y_{\omega N}$ et Y_N , respectivement. On appelle les solutions strictement efficaces solutions efficaces uniques et par conséquent, elles ne possèdent pas de contrepartie dans l'espace objectif.

Toute au long de cette partie. Nous supposons que $Y = f(X)$ est bornée et que X_E est non vide [20].

Les résultats principaux dans ce document ont traité à l'efficacité appropriée au sens de Geoffrion [2].

Définition 3.2. *Un point $x \in X$ est dit une solution efficace propre de la MOP si seulement si elle est efficace et s'il existe un scalaire $M > 0$ tel que, pour tout $i, 1 \leq i \leq p$, et chaque $x' \in X$ satisfaisant $f_i(x') < f_i(x)$, il existe au moins un $j, 1 \leq j \leq p$, de telle sorte que $f_j(x') > f_j(x)$ et :*

$$(f_i(x) - f_i(x')) / (f_j(x') - f_j(x)) \leq M.$$

Les ensembles de toutes les solutions efficaces propres et des points (non dominée du sens de Geoffrion) sont désignés par $X_{\rho E}$ et $Y_{\rho N}$, respectivement. Autre définitions (plus générale) d'efficacité propre ont été données par Borwein [11], Hartley [24] et Benson [8]. Cependant, dans la programmation multiobjectif la définition Geoffrion est encore répandue aujourd'hui. Ainsi, a chaque fois que nous référons à l'efficacité propre nous impliquons la définition Geoffrion.

L'approche traditionnelle de résolution MOPs est par la scalarisation, qui implique la formulation d'un programme d'un seul objectif qui lié MOP (SOP) au moyen d'une valeur réelle fonction scalarisation étant typiquement une fonction des fonctions objectives de la MOP, scalaire auxiliaire ou variables de vecteur et / ou des paramètres scalaires ou vectorielles. parfois l'ensemble réalisable de la MOP est en outre limitée par de nouvelles fonctions de contraintes liées aux fonctions objectives de la MOP et / ou les nouvelles variables introduites. Pour une enquête sur la scalarisation (et non scalariser) techniques, le lecteur est renvoyé à [21].

Dans la section suivante, nous examinons une technique de scalarisation bien connu, méthode e-contrainte et la liste des résultats théoriques. Nous soutenons que cette méthode présente quelques inconvénients pratiques et théoriques. Dans les deux sections suivantes nous proposons deux modifications de la méthode e-contrainte pour remédier à la principale deux faiblesses. à la Section 4 une combinaison des modifications proposées à les Sections 2 et 3 est présenté comme l'amélioration de la méthode de e-contrainte .Nous présentons quelques conclusions à la Section 5.

3.1 La méthode de e-Contraint

La méthode e-contrainte est apparue dans[31]. et est discuté en détail dans Changkong et Haimes [27]. Elle est basé sur une scalarisation où l'un des fonctions objectif est minimisée, alors que toutes les autres fonctions objectives sont limitées par le haut au moyen des contraintes supplémentaires :

$$(P_{\varepsilon-k}) \min\{f_k(x) : f_i(x) \leq \varepsilon_i, i \neq k, x \in X\},$$

Où $\varepsilon_{-k} = (\varepsilon_1, \dots, \varepsilon_{k-1}, \varepsilon_{k+1}, \dots, \varepsilon_p)^T \in \mathbb{R}^{p-1}$ et $k \in \{1, \dots, p\}$ On note l'ensemble réalisable du problème e-contrainte $P_{\varepsilon-k}$ par :

$$X_k^\varepsilon = \{x \in X : f_i(x) \leq \varepsilon_i, i \neq k\}.$$

Tout au long de ce travail, nous supposons que ε_{-k} est toujours choisie de telle sorte que $P_{\varepsilon-k}$ (ou ces modifications) sont réalisables savoir $X_k^\varepsilon \neq \phi$.

Ci-dessous, nous indiquons les deux principaux théorèmes sur la méthode e-contrainte (voir [27]).

Les solutions optimales de $P_{\varepsilon-k}$ sont faiblement efficaces, Les solutions optimales de $P_{\varepsilon-k}$ sont des solutions strictement efficaces et efficaces qui peuvent être caractérisés par la méthode e-contrainte.

Théorème 3.1. *Pour tout $\varepsilon - k \in \mathbb{R}^{p-1}$, les déclarations suivantes détiennent.*

- Si $\hat{x} \in X$ est une solution optimale de $P_{\varepsilon-k}$, alors $\hat{x} \in X_{\omega E}$.
- Si $\hat{x} \in X$ est une solution unique de $P_{\varepsilon-k}$, alors $\hat{x} \in X_{sE}$.

Théorème 3.2. *Une solution $\hat{x} \in X$ est efficace si et seulement si elle est une solution optimale de $P_{\varepsilon-k}$ pour tout $k = 1, \dots, p$, où $\varepsilon_i = f_i(\hat{x})$ pour $i = 1, \dots, p, i \neq k$.*

Ainsi, étant donné une solution réalisable de MOP, pour vérifier l'efficacité il suffit de vérifier l'unicité de la solution de $P_{\varepsilon-k}$ pour un k en résolvant le problèmes d'optimisation mono-objectif $P_{\varepsilon-k}$, $k = 1, \dots, P$.

Il y a des résultats aussi sur l'efficacité appropriée. Benson et Morin [9]. considèrent la perturbation des fonctions associées à $P_{\varepsilon-k}$:

$$v(u) = \inf\{f_k(x) : f_i(x) - \varepsilon_i \leq u_i, i \neq k; x \in X\}$$

Le problème $P_{\varepsilon-k}$ est stable si $v(0)$ est fini et il existe un certain $M > 0$ tel que pour tout $0 \neq u \in \mathbb{R}^{p-1}$, le rapport $(v(0) - v(u)) / \|u\| \leq M$. Benson et Morin montrent que pour MOPs convexes $\hat{x} \in X_E$ est bien efficace si et seulement si $P_{\varepsilon-k}$ est stable pour toutes $k = 1, \dots, P$.

Huang et Yang [28]. étendent le résultat de [9]. pour non convexes problèmes. Ils utilisent la scalarisation hybride

$$\min \left\{ \sum_{k=1}^p \lambda_k f_k(x) : f_k(x) \leq f_k(\hat{x}), k = 1, \dots, p; x \in X \right\} \quad (3.1)$$

et de donner deux caractérisations d'efficacité correcte, d'abord en termes d'une pénalité exacte de la fonction et deuxième en termes de la stabilité du problème scalarisé (1). Lee et al. [6] modifier ce résultat pour obtenir une caractérisation d'efficacité appropriée pour MOPs avec des fonctions non convexes et non dérivable.

Enfin Sach [23]. Caractérise les solutions propres efficaces de Benson en termes de points de selle.

D'autre part, Chankong et Haimes [27]. montrent que si $\hat{x} \in X_{\rho E}$ alors \hat{x} est une solution de $P_{\varepsilon-k}$ pour k avec $\varepsilon_i = f_i(\hat{x})$ pour $i \neq k$ et les multiplicateurs positifs de Karush-Kuhn-Tucker pour des contraintes $f_i(x) \leq \varepsilon_i$. de plus, ils montrent que pour les convexes des problèmes MOPs l'inverse est vrai, aussi Ce résultat établit l'équivalence de

problème e-contrainte et la scalarisation lagrangien k -ième

$$\min \left\{ f_k(x) + \sum_{i \neq k} u_i f_i(x) : x \in X \right\}$$

Pour les problèmes convexes. Ceci est également discuté dans Miettinen [15]. Notez que le scalarisation Lagrangien k -ième est essentiellement la même que scalarisation de la somme pondérée .

Pour plus d'informations sur l'efficacité propre dans le problème multiobjectif nous référons le lecteur aux références dans les ouvrages cités au-dessus. Nous omettons une discussion détaillée parce que ces références ne sont pas directement concernée par la méthode e-contrainte.

Les résultats indiqués dans la section précédente montrent clairement l'avantage principal du problème e-contrainte sur les autres scalarisations : Chaque solution efficace peut être trouvé comme solution optimale de $P_{\epsilon-k}$. Ce résultat est indépendant de la structure de X , à avoir , il est vrai pour les problèmes d'optimisation non convexes et discrets. Ce fait est très important, comme il distingue le procist de la méthode de la somme pondérée , de minimiser une combinaison convexe des objectifs de la problème MOP.

Dans cette partie, nous modifions la méthode e-contrainte et nous prouvons une certaine nécessité et des conditions suffisantes pour des solutions optimales de ces problèmes modifiés à être (faiblement, strictement, propre) efficace. Les résultats sur l'efficacité propre, puis que nous prouvons sont plus directes que ceux mentionnés ci-dessus, étant donné que nous ne faisons pas référence à la stabilité de tous les problèmes d'optimisation. Nous notons que la modification donnée en Section 2 est celle qui est utilisée dans la définition de la fonction de perturbation dans [9].

La Programmation multiobjectif est un domaine de l'optimisation guidé par des applications. Dans beaucoup de monde réel des situations de prise de décision, il est nécessaire de considérer les objectifs contradictoires. La bibliographie par White [5]. et Chaps. 20-23

dans [12]. fournissent de nombreuses preuves de ce fait. Par conséquent, il existe un besoin des techniques efficiente et efficace pour résoudre les programmes multiobjectifs dans de nombreux domaines. Ces techniques devraient être intuitifs et ont des propriétés qui sont souhaitables dans l'application. Cela soulève de plusieurs problèmes avec la méthode e-contrainte.

Dans les applications on traite souvent MOPs d'une structure particulière, qui est exploitée dans les algorithmes de la solution. L'ajoute d' e-contraintes peut détruire cette structure et de faire des algorithmes moins efficaces (par exemple le plus court chemin problème polynomialement résolable devient problème NP-difficile avec ressources limitées).

La preuve Numérique de ce comportement a été rapportée par Ehrgott et Ryan [22]. pour un ensemble bicritère de modèle de partitionnement dans la programmation d'équipage aérienne.

Lorsque on utilise $P_{\varepsilon-k}$ pour déterminer une solution (faiblement) efficace \hat{x} de MOP, cette solution satisfait bien sure les contraintes $f_i(\hat{x}) \leq \varepsilon_i$ pour $i \neq k$. Mais, dans les problèmes d'optimisation discrets non-convexe , il est tout à fait possible que $f_i(\hat{x})$ est beaucoup plus petit que ε_i pour un certain i , alors qu'une solution efficace \bar{x} avec $f_i(\bar{x})$ juste un peu plus grand que ε_i existe.

Ce dernier pourrait être une solution préférable pour le décideur. la méthode e-contrainte est inflexible en ce sens, car il impose une rigide contrainte sur des valeurs objectives, alors que nous pourrions plutôt souhaiter d'utiliser les contraintes pour trouver une solution avec $f_i(x)$ "proche a" ε_i .

Un autre problème est lié à des solutions efficaces propres. Dans des situations pratiques, les décideurs sont généralement intéressés par des solutions efficaces propres plutôt que celles efficaces. Ces solutions se caractérisent par des compromis finis (ou taux de substitution) entre les valeurs objectives, à savoir, une augmentation dans un objectif peut

être compensée par une diminution limitée dans l'autre. Alors que toutes les solutions optimales d'un problème de somme pondérée avec des poids positifs sont toujours efficaces propres [2]., la méthode traditionnelle e-contrainte ne permet pas facilement les résultats sur une efficacité propre pour les solutions optimales, comme s'est discuté ci-dessus.

Dans Section.2 nous abordons le troisième problème :

Par l'inclusion des variables d'écart qui mesurent l'écart entre les valeurs objectives $f_i(x)$ et les limites ε_i , $i \neq k$, nous obtenons des déclarations sur l'efficacité propre. Dans Section.3 nous abordons les deux premiers problèmes et nous présentons une modification qui perméte de surmonter l'inflexibilité associée à la méthode e-contrainte en détendant les contraintes $f_i(x) \leq \varepsilon_i$ et une fonction de type pénalité comme une approximation nous montre que les résultats sur les solutions efficaces propres peuvent encore être obtenues.

3.2 Obtenir les résultats sur l'efficacité propre

Une première modification de la méthode de e-contrainte utilise des variables d'écart non négatifs s_i , $i \neq k$, qui sont ajoutés à l'e-contraintes.

La fonction objective correspond à la somme de la fonction objective k -ième et la somme pondérée négative des variables d'écart. Le problème scalarisé $P_{\varepsilon-k}^+$ peuvent être formulés comme :

$$(P_{\varepsilon-k}^+) \min \left\{ f_k(x) - \sum_{i \neq k} \lambda_i s_i : f_i(x) + s_i \leq \varepsilon_i, i \neq k; s_i \geq 0, i \neq k, x \in X \right\},$$

Où $\varepsilon_{-k} = (\varepsilon_1, \dots, \varepsilon_{k-1}, \varepsilon_{k+1}, \dots, \varepsilon_p)^T \in \mathbb{R}^{p-1}$ et $\lambda_i \geq 0$, $i \neq k$, sont des poids non négatif.

Comme mentionné précédemment, à l'exception de la positivité de s , ce problème est utilisée pour définir la fonction de perturbation $v(s)$ en [9], et on a les variables d'écart qui donnent des informations sur l'efficacité propre d'une solution. Ainsi, cette première modification peut être considérée comme une mesure qui remède l'un des deux inconvénients de la méthode e-contraintes mentionnées précédemment. Une première différence à la méthode e-contrainte est que les e-contraintes dans $P_{\varepsilon-k}^+$ sont toujours actif à optimalité.

Lemme 3.1. Soit $\lambda \geq 0$, Si $P_{\varepsilon-k}^+$ admet une solution optimale, alors il admet une solution optimale (\hat{x}, \hat{s}) telle que $f_i(\hat{x}) + \hat{s}_i = \varepsilon_i$ pour tout $i \neq k$. Si $\lambda > 0$, alors chaque solution optimale de $P_{\varepsilon-k}^+$ satisfait $f_i(\hat{x}) + \hat{s}_i = \varepsilon_i$ pour tout $i \neq k$.

Démonstration. Supposons que (\hat{x}, \hat{s}) est une solution optimale de $P_{\varepsilon-k}^+$, Mais il existe $j \in \{1, \dots, p\} \setminus \{k\}$ de telle sorte que $f_j(\hat{x}) + \hat{s}_j < \varepsilon_j$. Alors, pour $\delta = \varepsilon_j - f_j(\hat{x}) > 0$, définir :

$$\tilde{s}_i = \begin{cases} \hat{s}_i, & \text{pour } i \in \{1, \dots, p\} \setminus \{j, k\} \\ \hat{s}_i + \delta, & \text{pour } i = j \end{cases}$$

(\hat{x}, \tilde{s}) est réalisable pour $P_{\varepsilon-k}^+$. Pour $\tilde{s}_j > \hat{s}_j$, nous avons :

$$f_k(\hat{x}) - \sum_{i \neq k} \lambda_i \tilde{s}_i \leq f_k(\hat{x}) - \sum_{i \neq k} \lambda_i \hat{s}_i.$$

Cela signifie que (\hat{x}, \tilde{s}) donne une meilleure valeur de la fonction objectif, meilleur que (\hat{x}, \hat{s}) (si $\lambda_j > 0$), soit la même que (\hat{x}, \hat{s}) (si $\lambda_j = 0$).

□

Avant de commencer l'analyse de cette modification théoriquement, nous voulons acquérir une certaine aperçu de la géométrie de la méthode. En regardant le cas particulier de $p = 2$, Lemme 3.1 fournir les outils d'une visualisation dans l'espace objectif. Notez que l'ensemble réalisable de $P_{\varepsilon-2}^+$ (variables x) est égal à un sous-ensemble de l'ensemble réalisable du BOP, à savoir X_2^ε .

Pour une solution réalisable donnée (\bar{x}, \bar{s}_1) , soit $v = f_2(\bar{x}) - \lambda_1 \bar{s}_1$ désignez l'objectif de la valeur de la fonction de $P_{\varepsilon-2}^+$. L'ensemble des $v = f_2(x) - \lambda_1 s_1$ de niveau peut être interprété comme une ligne dans l'espace $s_1 - f_2$ avec une pente $-\lambda_1$ passant par $(\bar{s}_1, f_2(\bar{x}))$.

Considérons une solution optimale $(\hat{x}, \hat{s}_1) \in \mathbb{R}^2 \times \mathbb{R}$ pour un problème $P_{\varepsilon-2}^+$. Soit $\hat{v} = f_2(\hat{x}) - \lambda_1 \hat{s}_1$ désigne la valeur de la fonction objective optimale de $P_{\varepsilon-2}^+$. D'après le Lemme 3.1, les ε -contraintes tiennent l'égalité à l'optimalité, alors $\hat{s}_1 = \varepsilon_1 - f_1(\hat{x})$.

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

Si $\hat{s}_1 = 0$, on a $f_1(\hat{x}) = \varepsilon_1$ et $f_2(\hat{x}) = \hat{v}$. Supposons maintenant que $\hat{s}_1 \neq 0$.

Substituer \hat{s}_1 dans la fonction objectif de $P_{\varepsilon-2}^+$ donne :

$$-\lambda_1 = (f_2(\hat{x}) - \hat{v}) / (f_1(\hat{x}) - \varepsilon_1).$$

Le poids de scalaire λ_1 est égale à la pente négative de la ligne $(f_1(\hat{x}), f_2(\hat{x}))$ et (ε_1, \hat{v}) . Pour une visualisation voir figure 3.1. L'ensemble réalisable de la MOP est réduite par la force additionnelle de ε -contrainte $f_1(x) \leq \varepsilon_1$. Alors, une ligne avec $-\lambda_1$ est parallèle vers l'origine jusqu'à ce qu'il prend en charge l'ensemble restreint non dominé. Le point de support est le point non dominé $f(\hat{x})$.

Selon le choix des vecteurs du poids de résultats différents pour $P_{\varepsilon-k}^+$ peuvent être dérivés. Considérons d'abord les vecteurs de poids $\lambda \geq 0$. Nous montrons que les solutions de $P_{\varepsilon-k}^+$ sont faiblement efficaces dans ce cas et qu'un résultat plus fort pour $\lambda \geq 0$ ne peut être dérivé.

On notera que $P_{\varepsilon-k}^+$ avec $\lambda = 0$ correspond au problème initial de e-contrainte.

Proposition 3.1. *Soit (\hat{x}, \hat{s}) est une solution optimale de $P_{\varepsilon-k}^+$ avec $\lambda \geq 0$. Alors \hat{x} est un solution faiblement efficace de MOP.*

Démonstration. Soit (\hat{x}, \hat{s}) une solution optimale de $P_{\varepsilon-k}^+$ avec $\lambda \geq 0$. Supposons qu'il existe $x \in X$ tel que $f(x) < f(\hat{x})$. donc, $f_i(x) + s_i < f_i(\hat{x}) + s_i \leq \varepsilon_i$ pour $i \neq k$, à savoir, (x, \hat{s}) est réalisable pour $P_{\varepsilon-k}^+$ et $f_k(x) < f_k(\hat{x})$. Cela implique :

$$f_k(x) - \sum_{i \neq k} \lambda_i \hat{s}_i < f_k(\hat{x}) - \sum_{i \neq k} \lambda_i \hat{s}_i,$$

une contradiction à l'optimalité de (\hat{x}, \hat{s}) pour $P_{\varepsilon-k}^+$.

□

Des solutions optimales de $P_{\varepsilon-k}^+$ sont en effet pas nécessairement efficace, même si $\lambda \neq 0$, comme l'exemple suivant montre.

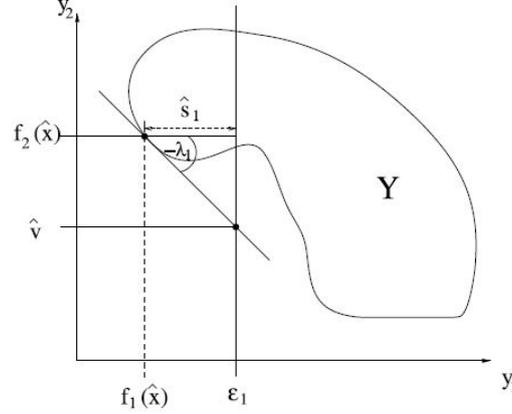


FIGURE 3.1 – la méthode ϵ -contrainte Modifié pour un exemple bicritère

Exemple 3.1. *Soit :*

$$Y = \{y^1 = (16, 1, 2)^T, \quad y^2 = (1, 5, 4)^T, \quad y^3 = (1, 5, 5)^T, \quad y^4 = (17, 3, 1)^T\}$$

*soit l'ensemble des vecteurs objectifs réalisables d'une MOP discrète. Notez que $Y_{\omega N} = Y$.
soit*

$$k = 1, \quad \lambda = (\lambda_2, \lambda_3)^T = (1, 0)^T, \quad \epsilon_1 = (10, 10)^T.$$

Soit $z(y)$ la valeur optimal de la fonction objectif de $P_{\epsilon-k}^+$ pour $y \in Y$. Alors,

$$z(y^1) = 16 - 9 = 7$$

$$z(y^2) = 1 - 5 = -4$$

$$z(y^3) = 1 - 5 = -4$$

$$z(y^4) = 17 - 7 = 10$$

*Les deux points $(1, 5, 4)^T$ et $(1, 5, 5)^T$ valeurs objectives rendement optimales de $P_{\epsilon-1}^+$.
cependant, $(1, 5, 5)^T$ est faiblement non dominée mais pas non dominée.*

Proposition 3.2. *Soit (\hat{x}, \hat{s}) est une solution optimale de $P_{\epsilon-k}^+$ avec $\lambda \geq 0$. Si \hat{x} est unique, alors \hat{x} est une solution strictement efficace de MOP.*

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

Démonstration. Pour x , On suppose que $f(x) \leq f(\hat{x})$. alors (x, \hat{s}) est aussi une solution réalisable $P_{\varepsilon-k}^+$. Étant donné que la valeur de la fonction objective (x, \hat{s}) ne soit pas pire que celle du (\hat{x}, \hat{s}) ; unicité de \hat{x} implique que $x = \hat{x}$.

□

Propositions 3.1 et 3.2 et l'exemple 3.1 montrent que les résultats de la méthode originale de e-contrainte ne peuvent être renforcés que par l'inclusion des variables d'écart s_i si $\lambda \geq 0$, il est permis d'avoir Des solutions optimales du problème e-contrainte qui ne sont pas faiblement efficace en général. Cependant, des solutions optimales du problème e-contrainte modifiées avec poids positifs sont toujours efficaces.

Théorème 3.3. *Soit (\hat{x}, \hat{s}) est une solution optimale de $P_{\varepsilon-k}^+$ avec $\lambda > 0$. Donc, \hat{x} est un solution efficace de MOP.*

Démonstration. Soit (\hat{x}, \hat{s}) une solution optimale de $P_{\varepsilon-k}^+$. D'après le lemme 3.1 il résulte que \hat{x} est une solution optimale du problème $\min \sum_{i=1}^p \lambda_i f_i(x)$ sur l'ensemble réalisable X_k^ε avec $\lambda_k = 1$.

Par la théorème de Geoffrion [2]. il résulte que \hat{x} est une solution efficace propre :

$$\min\{(f_1(x), \dots, f_p(x)) : x \in X_k^\varepsilon\}$$

En particulier, \hat{x} est une solution efficace à ce problème. Il est facile de voir que \hat{x} est également une solution efficace du problème original MOP. □

Donc, nous déclarons un outil facile pour vérifier les conditions suffisantes pour identifier correctement l'efficacité Parmi les solutions de $P_{\varepsilon-k}^+$ avec des poids positifs. Pour prouver que nous avons besoin d'une technique lemme relative à des solutions propres efficaces de la MOP avec une possibilité de définir X_k^ε et X , respectivement.

Lemme 3.2. Soit \hat{x} une solution efficace propre de MOP avec un ensemble réalisable X_k^ε et $f_i(\hat{x}) < \varepsilon_i$ pour tout $i \neq k$. Donc, \hat{x} est une solution efficace propre de la MOP sur l'ensemble réalisable X .

Démonstration. Supposons que \hat{x} n'est pas efficacement propre pour MOP pour l'ensemble réalisable X . Ainsi, pour chaque $M > 0$, il existe un $x_M \in X$ et un indice i avec $f_i(x_M) < f_i(\hat{x})$ qui satisfait :

$$(f_i(\hat{x}) - f_i(x_M))/(f_j(x_M) - f_j(\hat{x})) > M$$

pour tous les indices j à $f_j(x_M) > f_j(\hat{x})$. Considérons une suite (M_α) avec, $\lim_{\alpha \rightarrow \infty} M_\alpha = \infty$. Sans perte de généralité, il existe un indice (fixe) i tel que, pour tous les M_α , il existe $x_\alpha \in X$ avec $f_i(M_\alpha) < f_i(\hat{x})$ et

$$(f_i(\hat{x}) - f_i(x_\alpha))/(f_j(x_\alpha) - f_j(\hat{x})) > M_\alpha,$$

pour tous les $j \neq i$ avec $f_j(x_\alpha) > f_j(\hat{x})$. On peut supposer que :

$$Q = \{j : f_j(x_\alpha) > f_j(\hat{x})\}$$

est constant pour tous α .

Cas 1 : Supposons $f_i(\hat{x}) - f_i(x_\alpha) \rightarrow \infty$ pour , $\alpha \rightarrow \infty$. Cela clairement contredit les bornes de $f(X)$.

Cas 2 : Supposons $f_j(x_\alpha) - f_j(\hat{x}) \rightarrow 0$ pour $\alpha \rightarrow \infty$.

Considérons l'ensemble $\tilde{Q} = \{j : f_j(x_\alpha) \rightarrow f_j(\hat{x}), \text{ pour } \alpha \rightarrow \infty\}$. De toute évidence, $\tilde{Q} \neq \emptyset$. On notera que $f_j(x_\alpha) > f_j(\hat{x})$. suppose qu'il existe un indice $j \notin \tilde{Q}$ et $f_j(x_\alpha) > \varepsilon_j$ pour une infinité de α . Depuis

$$(f_i(\hat{x}) - f_i(x_\alpha))/(f_j(x_\alpha) - f_j(\hat{x})) > M_\alpha,$$

les deux cas applique ou $j \in \tilde{Q}$. Puis $f_j(x_\alpha) > \varepsilon_j$ pour seulement un nombre fini α . Donc, il y a un α_0 tel que $f_j(x_\alpha) \leq \varepsilon_j$ pour tout $j \notin \tilde{Q}$ et $\alpha > \alpha_0$. Cela implique que $(x_\alpha)_{\alpha > \alpha_0} \subseteq X_k^\varepsilon$, Ainsi, $f_j(\hat{x}) < f_j(x_\alpha) \leq \varepsilon_j$. Cela contredit l'efficacité propre de \hat{x} pour le MOP sur l'ensemble réalisable X_k^ε . \square

Théorème 3.4. *Si (\hat{x}, \hat{s}) est une solution optimale de $P_{\varepsilon-k}^+$ avec $\lambda > 0$ et $\hat{s} > 0$, alors \hat{x} est une solution efficace propre de MOP.*

Démonstration. Soit (\hat{x}, \hat{s}) une solution optimale de $P_{\varepsilon-k}^+$ et $\hat{s} > 0$. D'après le théorème 3.3 nous savons que \hat{x} est efficace.

Depuis, à optimalité, les ε -contraintes tiennent à l'égalité (Lemme 3.1), on peut réécrire la fonction objective :

$$f_k(\hat{x}) - \sum_{i \neq k} \lambda_i (\varepsilon_i - f_i(\hat{x})) = \sum_{i=1}^p \lambda_i f_i(\hat{x}) - \sum_{i \neq k} \lambda_i \varepsilon_i,$$

Où nous définissons $\lambda_k := 1$. Puisque le terme $\sum_{i \neq k} \lambda_i \varepsilon_i$ est constante, nous pouvons interpréter \hat{x} comme une solution d'un problème de somme pondérée avec le poids des coefficients strictement supérieur à zéro, ce qui a un ensemble réalisable restreint, à savoir que l'ensemble réalisable X_k^ε de $P_{\varepsilon-k}^+$. Par Le théorème de Geoffrion [2]. \hat{x} est efficace propre pour la MOP avec ensemble réalisable X_k^ε .

D'après le lemme 3.2, une solution x efficace propre de MOP restreinte est aussi une solution efficace propre de l'original MOP . □

L'exemple suivant montre que, si $f(X)$ est n'est pas bornée, le résultat du théorème 3.4 n'est plus vrai.(n'est pas vrai de ce point).

Exemple 3.2. *Considérons l'ensemble*

$$X = \{x \in \mathbb{R}^2 : x_2 = 1/x_1, -1 < x_1 < 0\}$$

et les $\min_{x \in X} x$ de MOP. clairement, $X_E = X$, mais $X_{\rho E} = \emptyset$. Par conséquent, une solution optimale de $P_{\varepsilon-k}^+$ ne peut être jamais une solution efficace propre de la MOP.

Soit $k = 2$, $\varepsilon_1 = -(1/2)$ et $\lambda = 4$. Alors, $x = (-1, -1)$ est la solution optimale de $P_{\varepsilon-2}^+$. Notez que $\hat{s} = 1/2 > 0$.

Jusqu'à présent, les résultats des solutions optimales de $P_{\varepsilon-k}^+$ ont été trouvé. Nous

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

allons maintenant analyser comment obtenir les solutions efficaces (faiblement / propres) par des choix appropriés des paramètres.

En utilisons la méthode e-contrainte des solutions faiblement efficaces peuvent être obtenues avec $\lambda = 0$.

Lemme 3.3. Soit \hat{x} efficace. Alors, il existe ε , \hat{s} et $\lambda \geq 0$ tel que (\hat{x}, \hat{s}) est une solution optimale de $P_{\varepsilon-k}^+$ pour tout $k \in \{1, \dots, p\}$.

Ce résultat découle immédiatement du théorème choisissant $\varepsilon = f(\hat{x})$, $\hat{s} = 0$ et $\lambda = 0$. Ensuite, nous allons montrer que toute solution efficace propre peut être obtenue comme solution optimale de $P_{\varepsilon-k}^+$ avec des poids positifs.

Théorème 3.5. Soit \hat{x} efficace propre. Alors, pour tout $k \in \{1, \dots, P\}$, il existe ε , \hat{s} et $\lambda > 0$ tel que (\hat{x}, \hat{s}) est une solution optimale de $P_{\varepsilon-k}^+$.

Démonstration. Soit $\hat{x} \in X_{\rho E}$ et $k \in \{1, \dots, P\}$. on définir $\varepsilon_i := f_i(\hat{x})$ pour $i \neq k$. Ainsi, nous pouvons choisisse $\hat{s} = 0$.

Supposons qu'il n'existe pas $x \in X$ et $i \neq k$ tel que $f_i(x) < f_i(\hat{x})$. Considérer tous les solution possible pour $P_{\varepsilon-k}^+$, à savoir, tout $x \in X_k^\varepsilon$.

Alors pour tout $i \neq k$ nous avons $f_i(x) = f_i(\hat{x})$ et, puisque $\hat{x} \in X_E$, $f_k(x) \geq f_k(\hat{x})$. En conséquence, (x, s) ne donnée pas une meilleur valeur de la fonction objective pour $P_{\varepsilon-k}^+$ pour tout choix de $\lambda > 0$.

Supposons donc qu'il existe $x \in X$ et $i \neq k$ tel que $f_i(x) < f_i(\hat{x})$.

Puisque $\hat{x} \in X_{\rho E}$ il existe $M > 0$ tel que pour $\forall i$, et tout $x \in X$ avec $f_i(x) < f_i(\hat{x})$ il existe au moins un j , avec $f_j(x) > f_j(\hat{x})$ de telle sorte que :

$$(f_i(\hat{x}) - f_i(x)) / (f_j(x) - f_j(\hat{x})) \leq M.$$

De plus, si on ne considère que les points réalisable x de $P_{\varepsilon-k}^+$, à savoir, $x \in X_k^\varepsilon$, dans la définition de l'efficacité propre, il en résulte qu'il n'y a qu'un seul j , soit $j = k$, de telle

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

sorte que $f_j(x) > f_j(\hat{x})$ et

$$(f_i(\hat{x}) - f_i(x))/(f_j(x) - f_j(\hat{x})) \leq M.$$

Ainsi, puisque $x \in X_{\rho E}$, il y a $M > 0$ tel que, pour tout i et tout $x \in X_k^\varepsilon$ avec $f_i(x) < f_i(\hat{x})$, il soutient que $f_k(x) > f_k(\hat{x})$ et

$$(f_k(x) - f_k(\hat{x})) / (f_i(\hat{x}) - f_i(x)) \geq 1/M.$$

Définir les poids $\lambda_i := 1/(M(p-1))$. Notez que $\lambda > 0$. Nous affirmons que (\hat{x}, \hat{s}) est optimal de $P_{\varepsilon-k}^+$. Supposons qu'il existe un point réalisable (x, s) avec

$$f_k(x) - \sum_{i \neq k} \lambda_i s_i < f_k(\hat{x}) - \sum_{i \neq k} \lambda_i \hat{s}_i = f_k(\hat{x}). \quad (3.2)$$

De plus, supposons sans perte de généralité de (x, s) est optimale pour $P_{\varepsilon-k}^+$, c'est à dire., $f_i(x) + s_i = \varepsilon_i = f_i(\hat{x})$ pour chaque $i \neq k$. Substituer par $s_i, i \neq K$, en 3.2 on obtienne :

$$f_k(x) - f_k(\hat{x}) < \sum_{i \neq k} \lambda_i (f_i(\hat{x}) - f_i(x)). \quad (3.3)$$

L'inégalité 3.3 est une contradiction que nous allons montrer dans ce qui suit. Par le choix des poids et l'examen ci-dessus, pour tout $x \in X_k^\varepsilon$ et tout indice i , avec $f_i(x) < f_i(\hat{x})$, il estime que :

$$\lambda_i = \frac{1}{M(p-1)} \leq \frac{1}{p-1} \cdot \frac{f_k(x) - f_k(\hat{x})}{f_i(\hat{x}) - f_i(x)}, \quad i \neq k.$$

Par conséquent, l'inégalité

$$\lambda_i (f_i(\hat{x}) - f_i(x)) \leq \frac{1}{p-1} (f_k(x) - f_k(\hat{x}))$$

est valide pour tout $x \in X_k^\varepsilon$ et tout indice $i \neq k$. en suppose sur tous $i \neq k$ on a :

$$\sum_{i \neq k} \lambda_i (f_i(\hat{x}) - f_i(x)) \leq f_k(x) - f_k(\hat{x}).$$

et la contradiction est claire.

□

L'exemple suivant montre que $x \in X_E \setminus X_{\rho E}$ ne peut pas nécessairement être obtenu avec des poids positifs.

Exemple 3.3. Soit $p = 2$ et soit

$$X = \{x \in \mathbb{R}^2 : (x_1 - 1)^2 + (x_2 - 1)^2 \leq 1\}$$

avec $f(x) = x$. tel que, $(1, 0)$ et $(0, 1)$ sont efficaces, mais pas proprement efficaces.

La scalarisation :

$$\min\{x_2 - \lambda s : x_1 + s \leq 1; \quad x \in X\}$$

est équivalente à (a la même solution optimale x) :

$$\min\{x_2 - \lambda(1 - x_1) : (x_1 - 1)^2 + (x_2 - 1)^2 = 1\}.$$

Il est facile de voir que pour obtenir le point $(1, 0)$, le paramètre de poids λ doit être égal à zéro.

Notez que, pour obtenir $(1, 0)$, on peut également envisager

$$\min\{x_1 - \lambda s : x_2 + s \leq 0; \quad x \in X\}.$$

Il est clair que $x_1 = 1, x_2 = 0, s = 0$ est l'unique solution de ce problème pour tout $\lambda \geq 0$.

Insertion des variables d'écart fournit des informations permettant des déclarations sur une Efficacité propre. Néanmoins, l'inflexibilité des contraintes est pas abordée dans $P_{\varepsilon-k}^+$.

Dans la section suivante les ε -contraintes sont autorisés à être violés. Nous montrons que cette modification permet des résultats similaires que dans cette section à obtenir.

3.3 Création des contraintes inflexible

Dans cette section, nous abordons l'inflexibilité des contraintes dans $P_{\varepsilon-k}$. Pour ce faire, nous "Relaxons" les contraintes $f_i(x) \leq \varepsilon_i$ en leur permettant d'être violés et péna-

liser toute violation dans la fonction objectif. Ainsi, nous considérons le problème suivant :

$$(P_{\varepsilon-k}^-) \quad \min \left\{ f_k(x) + \sum_{i \neq k} \mu_i s_i : f_i(x) - s_i \leq \varepsilon_i \quad i \neq k, s_i \geq 0 \quad i \neq k, x \in X \right\}$$

Où $\mu_i \geq 0$, $i \neq k$, L'ensemble réalisable de $P_{\varepsilon-k}^-$ dans les variables $x \in X$, est l'ensemble réalisable de la MOP. Notez que si (\hat{x}, \hat{s}) est une solution optimale, alors nous pouvons suppose que

$$\hat{s}_i = \max\{0, \varepsilon_i - f_i(\hat{x})\}$$

Dans la figure 3.2, $P_{\varepsilon-2}^-$ illustre un problème bicritère. La ligne verticale marque la valeur de ε_1 . La ligne pointillée montre la fonction objectif de $P_{\varepsilon-2}^-$ en fonction du composant y_1 de la point non dominés Y_N . L'idée de la méthode est que, en pénalisant violations de la contrainte $f_1(x) \leq \varepsilon_1$, un minimum de l'objectif de $P_{\varepsilon-k}^-$ est atteinte avec ε -contrainte est active lorsque μ est choisi de manière propre.

On obtient les résultats suivants.

Proposition 3.3. *Soit (\hat{x}, \hat{s}) une solution optimale de $P_{\varepsilon-k}^-$ avec $\mu \geq 0$. Alors, \hat{x} est une solution faiblement efficace de la MOP.*

Démonstration. Supposons que \hat{x} n'est pas une solution faiblement efficace, $\exists x \in X$ tel que tel que $f(x)_i \leq f_i(\hat{x})$, $i = 1, \dots, p$, Puis, (x, \hat{s}) est réalisable pour $P_{\varepsilon-k}^-$ avec une valeur objective qui est plus petite que celle de (\hat{x}, \hat{s}) □

Selon des hypothèses supplémentaires, nous obtenons des résultats plus solides.

Proposition 3.4. *Si x est unique dans une solution optimale de $P_{\varepsilon-k}^-$ avec $\mu \geq 0$, alors \hat{x} est un solution strictement efficace de la MOP.*

Démonstration. Soient x tel que $f(x) \leq f(\hat{x})$. tel que, (x, \hat{s}) est une solution réalisable de $P_{\varepsilon-k}^-$, puisque la valeur de la fonction objective (x, \hat{s}) ne soit pas pire que celle du (\hat{x}, \hat{s}) , unicité de x implique que $\hat{x} = x$. □

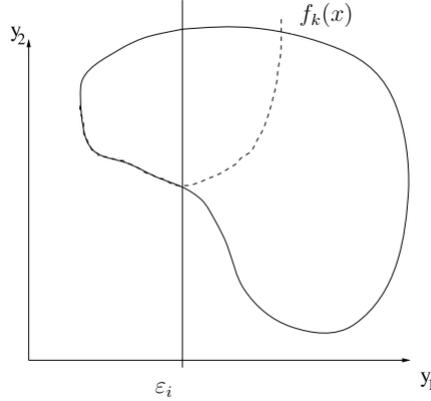


FIGURE 3.2 – L'ensemble Réalisable et la fonction objective de $P_{\varepsilon-k}^-$

On remarque qu'un résultat similaire au théorème 3.3 est non impossible. Même si $\mu > 0$, un solution optimale de $P_{\varepsilon-k}^-$ peut être juste faiblement efficace, mais pas efficace

Exemple 3.4. *Considérons un MOP avec*

$$X = \{(x_1, x_2) : (x_1 - 1)^2 + (x_2 - 1)^2 \leq 1\} + \mathbb{R}_{\geq}^2$$

Et $f(x) = x$ (où \mathbb{R}_{\geq}^2 est le quadrant non négatif dans \mathbb{R}^2). Soit $\varepsilon_1 > 1$. Tel que, $(\hat{x}_1, \hat{x}_2, \hat{s}_1) = (\hat{x}_1, 0, 0)$ est une solution optimale de $P_{\varepsilon-2}^-$ pour tout $1 \leq \hat{x}_1 \leq \varepsilon_1$. Si $\hat{x}_1 > 1$, cette solution est faiblement efficace, mais pas efficace, indépendante du choix de μ .

L'exemple montre que le problème ici est la possibilité d'existence des solution faiblement efficace qui satisfont les contraintes $f_i(x) \leq \varepsilon_i$, $i \neq k$. Cependant, si tous les ε_i sont choisies d'une manière telle qu'aucune solution faiblement efficace satisfait aux ε -contraintes, une solution optimale de $P_{\varepsilon-k}^-$, avec $\mu > 0$ donnera une solution efficace de MOP.

Aussi, si $\mu > 0$, de plus $P_{\varepsilon-k}^-$ a une solution optimale (\hat{x}, \hat{s}) avec $\hat{s} > 0$ même d'obtenir une efficacité propre de \hat{x}

Théorème 3.6. *Si (\hat{x}, \hat{s}) est une solution optimale de $P_{\varepsilon-k}$, avec $\mu > 0$ et $\hat{s} > 0$ alors \hat{x} est une solution efficace propre de MOP.*

Démonstration. Soient $\mu > 0$ et $\hat{s} > 0$, nous savons que $\hat{s}_i = f_i(\hat{x}) - \varepsilon_i > 0$, $i \neq k$. Nous

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

montrons d'abord que \hat{x} est efficace. Supposons qu'il existe $x \in X$ tel que $f(x) \leq f(\hat{x})$ alors avec s définis par $s_i := \max\{0, f_i(x) - \varepsilon_i\}$ et on a $s \leq \hat{s}$. Mais à partir de $f(x) \leq f(\hat{x})$, $f_k(x) < f_k(\hat{x})$ ou $s_k(x) < s_k(\hat{x})$ pour $i \neq k$ sont des contradiction d'optimalité de (\hat{x}, \hat{s}) .

De plus, $\hat{s}_i = \varepsilon_i - f_i(\hat{x})$, que l'on peut remplacer dans la fonction objectif de $P_{\varepsilon-k}^-$. Cela implique que \hat{x} est une solution optimale du problème :

$$\min \left\{ f_k(x) + \sum_{i \neq k} \mu_i (f_i(x) - \varepsilon_i) \mid f_i(x) \geq \varepsilon_i, i \neq k, x \in X \right\}.$$

Ainsi, \hat{x} est la solution d'un problème de la somme pondérée avec des poids positifs (Et des contraintes supplémentaires). Par le théorème de Geoffrion (voir [2].), \hat{x} est une solution efficace propre pour MOP avec des contraintes ajoutés $f_i(x) \geq \varepsilon_i, i \neq k$. Toutefois, puisque aucune de ces les contraintes sont actifs à la solution optimale \hat{x} , on peut employer un résultat analogue du Lemme 3.2 pour montrer que \hat{x} est une solution efficace propre de MOP.

□

Ce résultat est pas vrai si l'ensemble réalisable dans l'espace objectif est non bornée comme on va montrer dans l'exemple suivant.

Exemple 3.5. *Considérons un MOP*

$$X = \{(x_1, x_2) : x_2 = \frac{1}{x_1}, x_1 < 0, x_2 \geq -1\}$$

et $f(x) = x$. Clairement, $X_E = X$ et $X_{\rho E} = \emptyset$ soit $\varepsilon_1 = -2$ et $\mu = \frac{1}{4}$. Alors $(\hat{x}_1, \hat{x}_2, \hat{s}_1) = (-1, -1, 1)$ est une solution optimale de $P_{\varepsilon-2}^-$.

Nous passons maintenant au problème de montrer que les solutions (proprement) efficaces de MOP sont des solutions optimales de $P_{\varepsilon-k}^-$ pour des choix appropriés de k, ε et μ .

Lemme 3.4. *Soit \hat{x} un solution efficace. tel que , $\exists \varepsilon, \mu \geq 0$ et \hat{s} tel que (\hat{x}, \hat{s}) est une solution optimale de $P_{\varepsilon-k}^-$ pour tout $k \in \{1, \dots, p\}$*

Le lemme résulte immédiatement du théorème 3.3 en choisissant $\varepsilon = f(\hat{x})$, $\hat{s} = 0$ et

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

$\mu_i = \infty$ pour tout $i = 1, \dots, p$. Une analyse plus attentive montre que, pour des solutions efficaces propres, nous pouvons faire sans les pénalité infinie.

Théorème 3.7. Soit \hat{x} une solution efficace propre de MOP. Alors, pour tout $k \in \{1, \dots, p\}$, $\exists \varepsilon, \hat{s}, \mu_k$ avec $\mu_i^k < \infty$ pour tout $i \neq k$ tel que (\hat{x}, \hat{s}) est une solution optimal de $P_{\varepsilon-k}^-$ pour tout $\mu \in \mathbb{R}^{p-1}, \mu \geq \mu^k$.

Démonstration. Nous choisissons $\varepsilon_i := f_i(\hat{x}), i = 1, \dots, p$. Ainsi que nous pouvons choisir $\hat{s} = 0$ Soit $k \in \{1, \dots, p\}$ Parce que \hat{x} est une solution efficace propre, $\exists M > 0$ tel que, pour tout $x \in X$ avec $f_k(x) < f_k(x^*) \exists i \neq k$, tel que $f_i(x^*) < f_i(x)$ et

$$\frac{f_k(x^*) - f_k(x)}{f_i(x) - f_i(x^*)} < M$$

Nous définissons μ^k par $\mu_i^k := \text{Max}(M, 0)$ pour tout $i \neq k$.

Soit $x \in X$ et s telles que :

$$s_i = \max\{0, f_i(x) - \varepsilon_i\} = \max\{0, f_i(x) - f_i(x^*)\}, i \neq k$$

i.e la plus petite valeur possible qu'il peut prendre et on va montrer que :

$$f_k(x) + \sum_{i \neq k} \mu_i s_i \geq f_k(\hat{x}) + \sum_{i \neq k} \mu_i \hat{s}_i = f_k(\hat{x}) \quad (3.4)$$

Tout d'abord, nous prouvons que nous pouvons supposer que $x \in X_E$ dans l'équation (3.4). D'autre part, $\exists x' \in X_E$, avec $f(x') \leq f(x)$, (à noter que, en raison de nos hypothèses, le MOP est extérieurement stable [32]. de sorte que l'existence de x' est garanti) et s' avec

$$s'_i = \max\{0, f_i(x') - \varepsilon_i\}$$

Depuis $s' \leq s$, nous obtenons que :

$$f_k(x') + \sum_{i \neq k} \mu_i s'_i \leq f_k(x) + \sum_{i \neq k} \mu_i s_i$$

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

pour chaque $\mu \geq 0$

Maintenant, Soit $x \in X_E$. Considérons le cas $f_k(x) \geq f_k(\hat{x})$. Alors,

$$f_k(x) + \sum_{i \neq k} \mu_i s_i > f_k(\hat{x}) + 0 = f_k(\hat{x}) + \sum_{i \neq k} \mu_i^k \hat{s}_i$$

pour tout $\mu \geq 0$.

Maintenant, considérons le cas $f_k(x) < f_k(\hat{x})$ et soit

$$I(x) := \{i \neq k : f_i(x) > f_i(\hat{x})\}$$

Comme \hat{x} et x sont efficaces, $I(x) \neq \emptyset$. De plus, on peut supposer que $s_i = 0$ pour tout $i \notin I(x)$, $i \neq k$. Soit $i' \in I(x)$. Alors,

$$\begin{aligned} f_k(x) + \sum_{i \neq k} \mu_i s_i &\geq f_k(x) + \sum_{i \neq k} \mu_i^k s_i \\ &\geq f_k(x) + \sum_{i \in I(x)} \frac{f_k(\hat{x}) - f_k(x)}{f_i(x) - f_i(\hat{x})} s_i \\ &\geq f_k(x) + \frac{f_k(\hat{x}) - f_k(x)}{f_{i'}(x) - f_{i'}(\hat{x})} s_{i'} \\ &= f_k(x) + \frac{f_k(\hat{x}) - f_k(x)}{f_{i'}(x) - f_{i'}(\hat{x})} (f_{i'}(x) - f_{i'}(\hat{x})) \\ &= f_k(\hat{x}) = f_k(\hat{x}) + \sum_{i \neq k} \mu_i \hat{s}_i \end{aligned}$$

A partir de $\mu_i \geq \mu_i^k$, la définition de μ_i^k , la non négativité de tous les termes, $s_i = f_i(x) - f_i(\hat{x})$ pour $i \in I(x)$ et $\hat{s} = 0$. \square

Nous pouvons également voir que, pour $x \in X_E \setminus X_{pE}$, des valeurs finies de μ ne sont pas suffisantes.

Exemple 3.6. Soit $p = 2$ et

$$X = \{x \in \mathbb{R}^2 : (x_1 - 1)^2 + (x_2 - 1)^2 \leq 1\}$$

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

avec $f(x) = x$. tel que, $(1, 0)$ et $(0, 1)$ sont efficaces, mais pas propre la scalarisation

$$\min\{x_2 + \mu s : x_1 - s \leq 0; x \in X\}$$

est équivalente à (a la même solution optimale x)

$$\min\{x_2 + \mu x_1 : (x_1 - 1)^2 + (x_2 - 1)^2 = 1\}$$

Il est facile de voir que la solution optimale unique est donnée par $x_1 = 1 - \sqrt{1 - \frac{1}{\mu+1}}$ et il est nécessaire que $\mu \rightarrow \infty$ pour obtenir $x_1 \rightarrow 0$

Notez cependant que, pour obtenir $(0, 1)$, on peut aussi considérer que :

$$\min\{x_1 + \mu s : x_2 - s \leq 1; x \in X\}$$

Il est clair que $x_1 = 0, x_2 = 1, s = 0$ est une solution optimale de ce problème pour tout $0 \leq \mu \leq \infty$

Jusqu'à présent, nous avons obtenu des résultats sur (stricte, faible, propre) l'efficacité des solutions de deux modifications de la méthode e-contrainte. Dans la section suivante, nous combinons ces modifications dans la méthode améliorée de e-contrainte et indiquent les résultats pour cette méthode. Nous décrivons également comment utiliser cette méthode dans un algorithme pour résoudre MOPs.

3.4 Amélioration de la méthode e-Constraint

Considérons maintenant une combinaison des deux modifications de la section. 2 et 3, qui est :

$$(P_{\varepsilon-k}^{comb}) \min \left\{ f_k(x) - \sum_{i \neq k} \lambda_i s_i^+ + \sum_{i \neq k} \mu_i s_i^- : f_i(x) + s_i^+ - s_i^- \leq \varepsilon_i, i \neq k, x \in X : s_i^+, s_i^- \geq 0, i \neq k \right\}$$

Où $\lambda_i, \mu_i \geq 0, i \neq k$.

Étant donné une solution réalisable $(x, s^+, s^-) \in \mathbb{R}^p \times \mathbb{R}^{p-1} \times \mathbb{R}^{p-1}$ de $P_{\varepsilon-k}^{comb}$ les variables superficielle et les variables d'écart s_i^+ et s_i^- , $i \neq k$, peut être changé simultanément par

une quantité $\alpha_i \in \mathbb{R}$

$$s_i^+ := s_i^+ + \alpha_i, i \neq k, \text{ et } s_i^- := s_i^- + \alpha_i, i \neq k \quad (3.5)$$

Sans affecter la réalisabilité du i -ème e-contrainte, $i \neq k$. Nous voulons étudier comment choisie la largeur de a_i sans affecter la réalisabilité globale de la solution.

Pour simplifié, nous limitons notre examen à un seul $i \in \{1, \dots, p\}$. Lorsque on changer les variables superficielle comme dans l'équation 3.5, la valeur de la fonction objectif est modifiée par $(\mu_i - \lambda_i)\alpha_i$. Si $\mu_i - \lambda_i < 0$, α_i peut être choisi arbitrairement grand sans affecter la restriction non négative sur s_i^+ ou s_i^- respectivement. En conséquence, la valeur de la fonction objectif diminue de façon arbitraire et donc $P_{\varepsilon-k}^{comb}$ est non bornées. Donc considérons le cas où $\mu_i - \lambda_i \geq 0$. Alors, α_i peut être choisi par :

$$\alpha_i := -\min\{s_i^+, s_i^-\}$$

Pour d'améliorer la valeur de la fonction objective, sans violer les contraintes non négative. Dans ce cas, s_i^+ ou s_i^- est égal à zéro après la mise à jour. Si $\mu_i - \lambda_i = 0$, changer s_i^+ ou s_i^- n'affecte la valeur de la fonction objectif. Ainsi, nous obtenons le lemme suivants.

Lemme 3.5. (i) Si $\exists i \neq k$ tel que $\mu_i - \lambda_i < 0$, $P_{\varepsilon-k}^{comb}$ est non bornée.

(ii) Si $\mu - \lambda \geq 0$, alors \exists toujours une solution optimale de $P_{\varepsilon-k}^{comb}$ tel que est $s_i^+ \cdot s_i^- = 0$, $i \neq k$, où en d'autres termes, \exists une partition $I \dot{\cup} \bar{I}$ de $\{1, \dots, p\} \setminus k$ tel que $s_i^+ = 0$ pour tout $i \in I$ et $s_i^- = 0$ pour tout $i \in \bar{I}$.

Notez que depuis $s_i^+ = s_i^- = 0$ si $f_i(x) = \varepsilon_i$ cette partition n'est pas nécessairement unique. A partir de maintenant on supposer que $\mu_i - \lambda_i \geq 0$ De manière analogue à la section. 2 et 3, nous obtenons les résultats suivants dont la plupart suivent directement des résultats des sections précédentes.

Proposition 3.5. Soit $(\lambda, \mu) \geq 0$. Si $(\hat{x}, \hat{s}^+, \hat{s}^-)$ est une solution optimale du $P_{\varepsilon-k}^{comb}$, alors \hat{x} est une solution faiblement efficace de la MOP.

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

Le résultat découle de Sect. 2, Proposition 3.1 ou Sect. 3, Proposition 3.3, respectivement. Avec l'hypothèse supplémentaire de l'unicité de la solution, \hat{x} est encore strictement efficace.

Proposition 3.6. Soit $(\lambda, \mu) \geq 0$. et $(\hat{x}, \hat{s}^+, \hat{s}^-)$ une solution optimale de $P_{\varepsilon-k}^{comb}$, si \hat{x} est unique, alors \hat{x} est une solution strictement efficace de la MOP

Dans Sect. 2, nous avons montré que les vecteurs positifs de pondération λ donnent des solutions efficaces. Malgré l'absence d'une analogie dans la Sect. 3, un résultat similaire peut être obtenu pour $P_{\varepsilon-k}^{comb}$

Théorème 3.8. Soit $\lambda > 0$ et $\mu > 0$. Soit $(\hat{x}, \hat{s}^+, \hat{s}^-)$ est une solution optimale de $P_{\varepsilon-k}^{comb}$. Alors, \hat{x} est une solution efficace de la MOP.

Démonstration. Supposons que $\hat{x} \notin X_E$. Tel que, $\exists x \in X$ avec $f(x) \leq f(\hat{x})$ et \exists au moins un indice $j \in \{1, \dots, \}$ telle que $f_j(x) < f_j(\hat{x})$.

Cas 1 : $j = k$. Talque, $(\hat{x}, \hat{s}^+, \hat{s}^-)$ est réalisable pour $P_{\varepsilon-k}^{comb}$ et donne la meilleure valeur de la fonction objectif qui contredit l'optimalité de \hat{x} .

Cas 2 : $j \neq k$. Plusieurs sous-classes peuvent se produire :

sous-cas 2.1 : $f_j(x) < f_j(\hat{x}) \leq \varepsilon_j$. Puis, une contradiction découle de la section. 2 théorème 3.3

sous-cas 2.2 : $\varepsilon_j \leq f_j(x) < f_j(\hat{x})$ Notez que $\hat{s}_j^- > 0$. Il est $\delta > 0$ tel que $f_j(x) + \delta = f_j(\hat{x})$ alors :

$$f_j(x) - (\hat{s}_j^- - \delta) = f_j(\hat{x}) - \hat{s}_j^- \leq \varepsilon_j$$

puis :

$$\delta = f_j(\hat{x}) - f_j(x) \leq f_j(\hat{x}) - \varepsilon_j = \hat{s}_j^-$$

nous pouvons définir

$$s_i^- = \begin{cases} \hat{s}_i^- - \delta, & \text{pour } i = j \\ \hat{s}_i^-, & \text{pour } i \neq j \end{cases}$$

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

et (x, \hat{s}^+, s^-) est réalisable pour $P_{\varepsilon-k}^{comb}$ et donne la meilleure valeur de la fonction objectif que $(\hat{x}, \hat{s}^+, \hat{s}^-)$

sous-cas 2.3 : $f_j(x) < \varepsilon_j < f(\hat{x})$. Puisque \hat{x} est optimale, nous supposons sans perte de généralité que $\hat{s}_j^+ = 0$ et $\hat{s}_j^- > 0$. Définir

$$s_i^+ = \begin{cases} \varepsilon_i - f_i(x), & \text{pour } i = j \\ s_i^+, & \text{pour } i \neq j \end{cases}$$

et

$$s_i^- := \begin{cases} 0, & \text{pour } i = j \\ \hat{s}_i^-, & \text{pour } i \neq j \end{cases}$$

Cette définition de (x, s^+, s^-) Donne la meilleure valeur de la fonction objectif, ce qui contredit l'optimalité de $(\hat{x}, \hat{s}^+, \hat{s}^-)$.

□

La preuve montre que, en fait, pas nécessairement tous λ_j et μ_j à être positive. \hat{x} sera efficace si $\lambda_j > 0$ pour tout j tel que $f_j(\hat{x}) \leq \varepsilon_j$ et $\mu_j > 0$

Théorème 3.9. *soit $\lambda > 0$ et $\mu > 0$, si $(\hat{x}, \hat{s}^+, \hat{s}^-)$ est une solution optimale de $P_{\varepsilon-k}$ et s'il y a une partition $I \cup \bar{I}$ de $\{1, \dots, p\} \setminus k$ telle que $\hat{s}_i^+ > 0$ pour $i \in I$ et $\hat{s}_i^- > 0$, pour $i \in \bar{I}$, alors \hat{x} est une solution efficace propre de la MOP.*

Démonstration. \hat{x} est efficace suivent du théorème 3.8. Parce que $(\lambda, \mu) > 0$, nous savons que :

$$\hat{s}_i^+ = \begin{cases} \varepsilon_i - f_i(\hat{x}), & \text{pour } i \in I \\ 0, & \text{pour } i \in \bar{I} \end{cases}$$

$$\hat{s}_i^- := \begin{cases} f_i(\hat{x}) - \varepsilon_i, & \text{pour } i \in \bar{I} \\ 0, & \text{pour } i \in I \end{cases}$$

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

Par conséquent, la valeur objective de $P_{\varepsilon-k}^{comb}$ est

$$f_k(\hat{x}) + \sum_{i \in I} \lambda_i (f_i(\hat{x}) - \varepsilon_i) + \sum_{i \in \bar{I}} \mu_i (f_i(\hat{x}) - \varepsilon_i)$$

et \hat{x} est en fait une solution optimale du problème de la somme pondérée

$$\min \left\{ f_k(x) + \sum_{i \in I} \lambda_i f_i(x) + \sum_{i \in \bar{I}} \mu_i f_i(x) : f_i(x) \leq \varepsilon_i, \quad i \in I; f_i(x) \geq \varepsilon_i, \quad i \in \bar{I}, x \in X \right\} \quad (3.6)$$

D'après le théorème Geoffrion [2]., On a \hat{x} est une solution efficace propre du MOP $\min_{x \in X} (f_1(x), \dots, f_p(x))$ avec des contraintes artificiel comme en 3.6. Puisque aucun des ces contraintes sont actifs à la solution optimale \hat{x} , alors nous pouvons utiliser une autre fois un lemme analogue à lemme 3.2 pour montrer que \hat{x} est aussi efficace propre pour le problème original de MOP. \square

Lemme 3.6. *Soit \hat{x} une solution efficace de la MOP. Tel que, $\exists \lambda \geq 0, \mu \geq 0, \varepsilon$ et s^-, s^+ De telle sorte que $(\hat{x}, \hat{s}^+, \hat{s}^-)$ est une solution optimale du $P_{\varepsilon-k}^{comb}$ pour tous $k = \{1, \dots, p\}$. Le résultat suivent immédiatement des résultats analogues dans la section. 2 et 3.*

Théorème 3.10. *Soit \hat{x} efficace propre pour la MOP. Alors, pour tout $k \in \{1, \dots, p\}$, $\exists \lambda > 0, \mu > 0$, avec $\mu_i < \infty$ pour tout i , ε et \hat{s}^+, \hat{s}^- de telle sorte que $(\hat{x}, \hat{s}^+, \hat{s}^-)$ est une solution optimale de $P_{\varepsilon-k}^{comb}$.*

Le théorème résulte immédiatement de la secte.(3), théorème 3.5 et section 3, théorème 3.7.

Pour conclure cette section, nous décrivons comment les résultats peuvent être utilisés dans un réglage algorithmique.

Lorsque on utilisez la méthode améliorée de ε -contrainte pour résoudre un MOP, nous avons choisir et contrôler les trois paramètres $\varepsilon_{-k}, \lambda$, et μ . Choisissons $k = 1$ et Pour ε_{-k} , nous cherchons de trouver un ensemble fini des solutions efficaces.

Pour ε_{-1} , Nous pouvons considérer une ensemble fini des vecteurs $\{\varepsilon_{-1}^1, \dots, \varepsilon_{-1}^m\}$ Couvrant la range des valeurs des objectifs f_2, \dots, f_p .

Pour chacun de ceux dont nous avons besoin pour résoudre un problème $P_{\varepsilon-1}^{comb}$ avec

l'intention d'obtenir une solution optimale qui a $f_i(x)$ approximativement égale ε_i^j , $i > 1$. En supposant que cet algorithme est itératif, nous intégrons le contrôle de λ et μ dans les itérations. Nous pourrions commencer avec tout λ et $\mu \geq \lambda$ non nul (car du lemme 3.5). Cependant, la preuve du théorème 3.7 suggère que la valeur de μ dépend sur le compromis entre les objectifs, ainsi μ_i doit être plus grand, les plus petits ε_i solutions sont efficaces. Au cours des itérations on observe les valeurs de $f_i(x)$. Si $f_i(x)$ est « beaucoup plus » que ε_i alors μ_i devrait être augmenter. Si $f_i(x)$ est "beaucoup moins" que ε_i alors μ_i devrait être diminué.

Une fois une solution optimale est trouvée Théorème 3.8 garantit qu'il est efficace. Si en plus les variables d'écart et excédentaires remplissent les conditions du théorème 3.9 alors il est efficace propre.

Ehrgott et Ryan [22]. ont mis en œuvre $P_{\varepsilon-2}^-$ pour résoudre un ensemble bicritère partitionnement problème dans la programmation aérienne de l'équipage. Ici, les valeurs de μ sont réglées pour être égales à le compromis entre les deux critères obtenus à partir de la résolution de deux relaxations LP avec légèrement différentes valeurs droite de la contrainte sur $f_1(x)$. D'autres ajustements au cours de l'algorithme étaient inutiles et de très bons résultats de calcul ont été obtenus.

3.5 Application Numérique

On a le problème suivante :

$$\text{Min } Z_1 = -x_1 + 2x_2$$

$$\text{Min } Z_2 = x_1 - 3x_2$$

$$D. \quad \begin{cases} 6x_1 + 8x_2 & \leq 41 \\ -2x_1 - x_2 & \leq -4 \\ x_1, x_2 & \in \mathbb{Z}_+ \end{cases}$$

Pour résoudre ce problème nous utilisons la méthode contrainte amélioré.

Nous choisissons Z_2 à optimisé donc nous résolvons Z_2 sous les contrainte D en utilisons

❖ CHAPITRE 3. AMÉLIORATION DE LA MÉTHODE DE E-CONSTRAINT POUR LA PROGRAMMATION MULTIOBJECTIF

Cplex

$$\text{Min } Z_2 = x_1 - 3x_2$$

$$D. \quad \begin{cases} 6x_1 + 8x_2 & \leq 41 \\ -2x_1 - x_2 & \leq -4 \\ x_1, x_2 & \in \mathbb{Z}_+ \end{cases}$$

1^{ère} Itération :

$$x^1(0, 5)' \text{ et } Z_1 = 10, Z_2 = -15$$

Supposons que $\lambda = 1$, $\mu = 1$ et $\varepsilon = Z_1 = 10$

Nous résolvons le problème :

$$\text{Min } P_{\varepsilon-2} = x_1 - 3x_2 + s_1^+ - s_1^-$$

2^{ème} Itération :

$$x^2(1, 4)' \text{ et } Z_1 = 7, Z_2 = -11$$

3^{ème} Itération :

$$x^3(2, 3)' \text{ et } Z_1 = 4, Z_2 = -7$$

4^{ème} Itération :

$$x^4(4, 2)' \text{ et } Z_1 = 0, Z_2 = -2$$

5^{ème} Itération :

$$x^5(5, 1)' \text{ et } Z_1 = -3, Z_2 = 2$$

6^{ème} Itération :

$$x^6(4, 0)' \text{ et } Z_1 = -4, Z_2 = 4$$

7^{ème} Itération :

$$x^7(5, 0)' \text{ et } Z_1 = -5, Z_2 = 5$$

8^{ème} Itération :

$$x^8(6, 0)' \text{ et } Z_1 = -6, Z_2 = 6$$

9^{ème} Itération : $x^9 = \Phi$ pas de solution l'algorithme s'arrête

L'ensemble des solutions efficaces :

$$E = \{(0, 5)', (1, 4)', (2, 3)', (4, 2)', (5, 1)', (4, 0)', (5, 0)', (6, 0)'\}$$

En représente respectivement dans les figures 3.3 et 3.4 l'ensemble des variables et l'ensemble des critères :

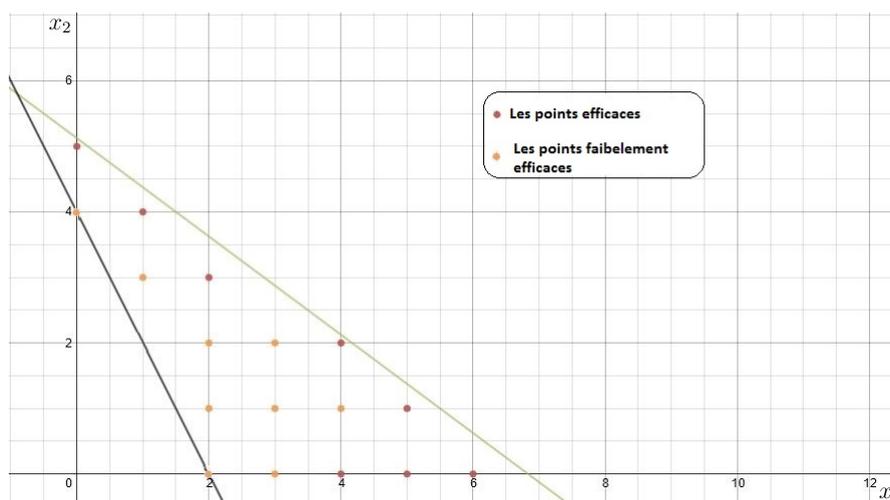


FIGURE 3.3 – L'ensemble des variables

Dans cet chapitre, nous avons proposé des modifications de la méthode ε -contrainte de programmation multiobjectif. Avec ces modifications, nous sommes en mesure de prouver les résultats sur une bonne efficacité des solutions optimales. En outre, la nouvelle formulation est plus flexible en termes de trouver des solutions efficaces du problème multiobjectif à proximité à ε_{-k} . Cela peut à son tour conduire à des avantages de calcul, en particulier pour les problèmes de programmation nombres entiers qui ayant un structure spécifique.

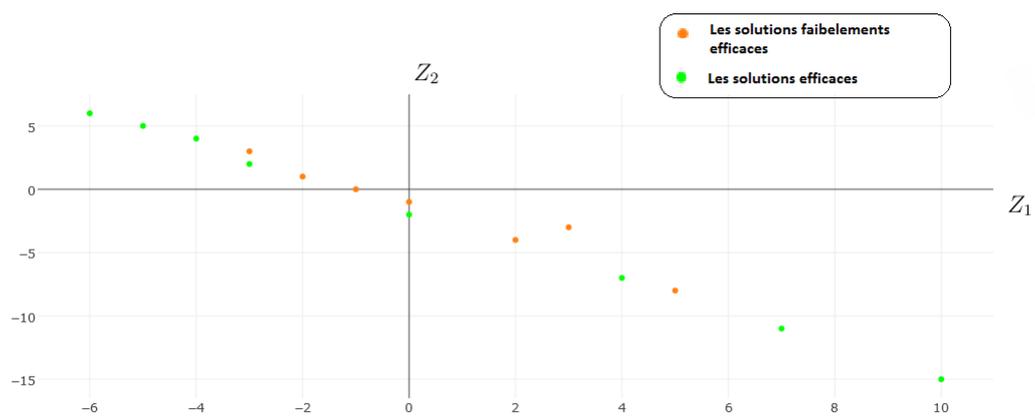


FIGURE 3.4 – L'ensemble des critères

CONCLUSION GÉNÉRALE

Les problèmes d'optimisation multiobjectif sont des plus en plus étudiés car ils ont un grand intérêt dans l'industrie où plusieurs objectifs contradictoires doivent être satisfaits. Ils sont la plupart du temps NP-Difficiles même si leur version mono-objectif ne l'est pas. Contrairement aux problèmes d'optimisation mono-objectif, il existe généralement plusieurs solutions pareto optimales ou efficaces en optimisation multiobjectif c.-à-d. des solutions qu'on ne peut améliorer sur un objectif sans le détériorer sur un autre. Comme il n'existe aucune solution meilleure en tous points qu'une autre, un compromis différent selon les personnes doit être choisi.

Le choix est donc subjectif et il est indispensable de proposer l'ensemble des choix possibles afin de ne pas exclure une possibilité. L'optimisation multiobjectif est donc avant tout un outil d'aide à la décision et c'est une personne qui prendra la décision finale.

Dans cet mémoire, nous avons présenté quelques méthodes de résolution des problèmes multiobjectif comme la méthode de pondération des fonctions objectif, la méthode de compromis, la méthode de salarisation....etc. Finalement nous avons présenté la méthode des contraintes élastiques qui est le noyau de notre mémoire.

La méthode des contraintes élastiques est généralement le résultat d'une modification sur la méthode d' ϵ -contraintes à l'aide de la méthode de pondération de fonctions objectives. Des exemples illustratifs sont résolus avec matlab et cplex pour montrer l'avantage de cette méthode.

BIBLIOGRAPHIE

- [1] A. Berro ; Optimisation Multiobjectif et Stratégie d'évolution en environnement Dynamique (thèse de Doctorat) ; 18 Décembre 2001 ; Université des sciences sociales Toulouse I.
- [2] A. Geoffrion : Proper efficiency and the theory of vector maximization *J.Math.Anal.Appl.*22,618–630 (1968).
- [3] C. A. Coello Coello, An updated survey of G.A. based multiobjective optimization techniques, Rapport technique Lania-RD-98-08, Xalapa, Veracruz, Mexico, décembre 1998.
- [4] D. A. Van Veldhuizen, Multiobjective evolutionary algorithms : classifications, analyses and new innovations, Ph. D., Graduate School of Engineering. Air Force Institute of Technology, Wright Patterson AFB, Ohio, USA, janvier 1999.
- [5] D. J. White : A bibliography on the application of mathematical programming multiple-objective methods. *J. Oper. Res. Soc.* 41, 669–691 (1990).
- [6] G. M. Lee, D. S. Kim, P. H. Sach : Characterizations of Hartley proper efficiency in nonconvex vector optimization. *J. Glob. Optim.* 33(2), 273–298 (2005).
- [7] H. Eschenauer, J. Koski, A. Osyczka, Multicriteria design optimization : procedures and applications, Springer-Verlag, 1990.

- [8] H. P. Benson : An improved definition of proper efficiency for vector maximization with respect to cones. *J. Math. Anal. Appl.* 71, 232–241 (1979).
- [9] H. P. Benson, T. L. Morin : The vector maximization problem proper efficiency and stability. *SIAM J. Appl. Math.* 32, 64–72 (1977).
- [10] J. Koski, Defectiveness of weighting method in multicriterion optimization of structures, *Comm. in Appl. Num. Math*, volume 1, pages 333-337, 1985.
- [11] J. Borwein : Proper efficient points for maximizations with respect to cones. *SIAM J. Control Optim.* 15, 57–63 (1977).
- [12] J. Figueira, S. Greco, M. Ehrgott : Multiple Criteria Decision Analysis : State of the Art Surveys. *International Series in Operations Research Management Science*, vol. 78. Springer, New York (2005).
- [13] K. M. Miettinen, *Nonlinear multiobjective optimization*, éditions Kluwer academic publisher, 1999.
- [14] K. M. Miettinen, *Nonlinear multiobjective optimization*, éditions Kluwer academic publisher, 1999.
- [15] K. M. Miettinen : *Nonlinear Multiobjective Optimization*. Kluwer Academic, Dordrecht (1999).
- [16] L. A. Zadeh, *Fuzzy Sets, Information and Control*, volume 8, pages 338-353, 1965.
- [17] M. Caldora Costa ; *Optimisation de Diapositifs Electromagnétiques dans un Contexte d'analyse par la Méthode des Eléments Finis* ; 17 Juillet 2001 ; L.E.G- Laboratoire d'électrotechnique de Grenoble.
- [18] M. Ehrgott ; *Multicriteria optimization* ; In *Lecture Notes in Economics and Mathematical Systems*, volume 491 ; Springer ; 2000.
- [19] Manuel d'utilisation du logiciel MultiSimplex .
- [20] M. Ehrgott : *Multicriteria Optimization*, 2nd edn. Springer, Berlin (2005).
- [21] M. Ehrgott, M. Wiecek : *Multiobjective programming*. In : Figueira, J., Greco, S., Ehrgott, M. (eds.) *Multicriteria Decision Analysis : State of the Art Surveys*, pp. 667–722. Springer, New York (2005).

- [22] M. Ehrgott, D. M. Ryan : Constructing robust crew schedules with bicriteria optimization. *J. MultiCriteria Decis. Anal.* 11, 139–150 (2002).
- [23] P. H. Sach : Nearly subconvex like set-valued maps and vector optimization problems. *J. Optim. TheoryAppl.* 119(2), 335–356 (2003).
- [24] R. Hartley : On cone-efficiency, cone-convexity and cone-compactness. *SIAM J. Appl. Math.* 34,211–222 (1978).
- [25] S. M. Sait, H. Youssef, *Iterative Computer Algorithms with Applications in Engineering : Solving Combinatorial Optimization Problems*, IEEE Computer Society, 1999.
- [26] V. Barichard ; *Approches Hybrides pour les Problèmes Multiobjectifs* (thèse de Doctorat) ; 24 Novembre 2003 ; Université d’Angers.
- [27] V. Chankong, Y. Haimes : *Multiobjective Decision Making Theory and Methodology*.Elsevier, New.York (1983).
- [28] X. X. Huang, X. Q. Yang : On characterizations of proper efficiency for nonconvex multiobjective optimization. *J. Glob. Optim.* 23(3–4), 213–231 (2002).
- [29] Yann Collette, Patrick Siarry ; *Optimisation Multiobjectif* ; Septembre 2002 ; Edition Eyrolles, 75240 Paris Cedex 05, France.
- [30] Y. Y. Haimes, W. A. Hall, H. T. Freedman, *Multiobjective optimization in water resources systems*, éditions Elsevier scientific publication company, 1975.
- [31] Y. Y. Haimes, L. S. Lasdon, D. A. Wismer : On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Trans. Syst. Man. Cybern.* 1, 296–297 (1971).
- [32] Y. Sawaragi, H. Nakayama, T. Tanino : *Theory of Multiobjective Optimization*. Academic Press San Diego (1985).

Résumé :

De nombreux problèmes réels impliquent l'optimisation simultanée de plusieurs critères souvent contradictoires. La solution d'un problème multiobjectif n'est pas une solution unique, mais un ensemble de solutions, connu comme l'ensemble des solutions Pareto optimales.

Dans ce mémoire, nous avons présenté l'optimisation combinatoire multiobjectif qui est le cadre de travail. Nous avons introduit les concepts fondamentaux au premier chapitre et quelques principales approches de résolution au deuxième chapitre, au troisième chapitre nous avons introduit la méthode des contraintes élastique qui a permis de donner un ensemble des solutions plus approfondie et précise.

Mots clés : Optimisation Multiobjectif combinatoire, Pareto Optimale, contraintes élastique.

Abstract :

Many real problems involve the simultaneous optimization of several and often contradictory criteria. The solution of a multiobjective problem is not a single solution, but a set of solutions, known as the set of Pareto optimal solutions .

In this work, we presented multi-objective combinatorial optimization which is the framework. We introduced the basic concepts in the first chapter and some main approaches to solving in the second chapter, in the third chapter we introduced the method of elastic constraints which allowed to give a set of more detailed and accurate solutions.

Key words : Combinatory Multiobjective Optimization, Pareto Optimal, Elastic Constraints

Remerciement

*Nous remercierons dieu tout puissant pour nous avoir offert la force
et la patience durant toutes ces années.*

*Nous tenons à exprimer un remerciement particulier à notre
encadreur **Zaidi Ali** pour avoir dirigé ce travail.*

Pour sa présence, son aide et surtout pour ses précieux conseils.

Mille merci nos trop chères familles, que dieu vous garde pour nous.

*Nous tenons à exprimer, nos sincères remerciements à tout le
personnel de l'institut des sciences et de la technologie surtout les
enseignants qui nous ont formé durant les années d'étude, et tous
ceux qui nous ont apporté une aide pour la réalisation de ce projet.*

*Sans oublier bien-sûre tous les amis et collègue d'études pour leur
enjouement et soutien moral.*

Ramadan, Lokmane