

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET
DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE
« HOUARI BOUMEDIENE »
FACULTE DE MATHEMATIQUES



MEMOIRE

Présenté pour l'obtention du diplôme de MAGISTER

EN : MATHEMATIQUES

Spécialité : Recherche Opérationnelle : Génie Mathématiques

Par : LARBI Noura

Sujet :

**OPTIMISATION STOCHASTIQUE DISCRETE A
NIVEAU**

Soutenu le 24/11/2009, devant le jury composé de :

- | | | | |
|--------------------------|------------------------------|----------------------|-----------------------------|
| <i>Mr- A. BERRACHEDI</i> | <i>Professeur</i> | <i>U. S. T. H. B</i> | <i>Président</i> |
| <i>Mr- D. CHAABANE</i> | <i>Maître de conférences</i> | <i>U. S. T. H. B</i> | <i>Directeur de Mémoire</i> |
| <i>Mr - M. MOULAI</i> | <i>Professeur</i> | <i>U. S. T. H. B</i> | <i>Examineur</i> |
| <i>Mme- Y. ZAÏRI</i> | <i>Maître de conférences</i> | <i>U. S. T. H. B</i> | <i>Examinatrice</i> |

À mes parents
À mes frères et soeurs
À mon époux

Remerciement

Je tiens tous d'abord à remercier mon directeur de thèse M.r Chaabane Djamal maître de conférence à l'USTHB qui a suivi ce travail de près, pour ces commentaires, conseils et corrections.

Je remercie vivement M.r Berrachedi Abdlehafidh professeur a l'USTHB pour avoir fait l'honneur de présider le jury.

Je remercie également Mme Ziari Yasmina, maître de conférence ainsi que M.r Moulai Mustapha, professeur a l'USTHB d'avoir accepter de faire partie du jury

Un immense merci pour mes parents, mes frères, mes sœurs et mon époux pour leurs Soutient incontestables et leurs aide.

Je remercie aussi mes amis et mes collègues pour leurs soutiens.

Table des matières

Introduction générale	1
Chapitre I Optimisation multi-objectif dans un environnement certain	
1. Introduction.....	3
1.1 Résolution d'un problème d'optimisation combinatoire.....	3
1.2 Programmation linéaire.....	4
1.2.1 Résultat d'une optimisation linéaire.....	4
1.2.2 Terminologie.....	4
1.2.3 Interprétation géométrique.....	5
1.2.4 Forme équivalente.....	5
1.2.5 Recherche d'un sommet de départ.....	7
1.2.5.1 Forme simpliciale (standard).....	7
1.2.5.2 Cas où la forme simpliciale n'est pas évidente.....	7
1.2.5.3 Passage d'un sommet à un autre.....	8
1.2.5.4 Choix de la variable entrante.....	9
1.2.5.5 Choix de la variable sortante.....	9
1.2.5.6 Calcul du nouveau sommet.....	9
1.2.5.7 Test d'optimalité.....	10
1.2.6 Dualité.....	10
1.3 Optimisation combinatoire multi-objectif.....	11
1.3.1 Problème d'optimisation multi-objectif.....	12
1.4 Notions de dominances et d'optimalité.....	12
1.5 Points particuliers.....	12
1.5.1 Matrice des gains.....	14
1.6 Optimisation linéaire multio-bjective en nombre entiers.....	15
1.6.1 Formulation des problèmes uni-critères en variables discrets.....	15
1.6.2 Complexité.....	16.
1.6.3 Coupe de Dantzig.....	16
1.6.4 Coupe de Gomory.....	16
1.7 Problème multio-bjective en variables discrètes (MOILP).....	17
1.7.1 Structure générale du problème MOILP.....	18
1.7.2 Solutions supportées / non supportée.....	18
1.7.3 Choix de la méthode d'aide à la décision.....	20

1.8	Approches classiques de résolution.....	21
1.8.1	Méthodes exactes pour l'optimisation multi-objective.....	21
1.8.2	Méthodes heuristiques.....	22
1.9	Quelques méthodes de résolution de problèmes d'optimisation multi-objectifs.....	22
1.9.1	L'agrégation linéaire.....	23
1.9.2	La recherche dichotomique.....	23
1.9.3	Méthode ϵ -contrainte.....	23
1.9.4	Goal programming: (Target vector optimisation).....	25
1.9.5	Méthode deux-phases.....	25
1.9.5.1	Première phase.....	25
1.9.5.2	Deuxième phase.....	26
1.10	Quelques méthodes de résolution de MOILP	27
1.10.1	Méthode de résolution dans l'espace des critères.....	27
1.10.1.1	Méthode de Klein et Hannan.....	27
1.10.1.2	Méthode de J. Sylva et A. Crema.....	28
1.10.2	Méthode de résolution dans l'espace des critères.....	31
1.10.2.1	Notations et définitions.....	31
1.10.2.2	Résultats théoriques.....	33
1.10.3	Méthode de M. Abbas & M. Moulaï.....	35
1.10.4	Méthode de M.Abbas & D.Chaabane.....	38
1.10.4.1	L'algorithme de la méthode.....	38
1.11	Conclusion sur l'optimisation multi-objective.....	40
Chapitre II Optimisation multi-objective dans un environnement incertain		
2	Introduction.....	41
2.1	Approches de la programmation linéaire stochastique.....	42
2.1.1	Approche « Wait and see »	42
2.1.2	Approche « Here and now ».....	42
2.2	Critère d'optimisation du problème équivalent.....	43
2.2.1	Contrainte du problème équivalent.....	43
2.2.1.1	Décomposition de benders (L-shaped méthode).....	47
2.3	Programmation linéaire stochastique multi-objective.....	48
2.3.1	Introduction.....	48
2.3.2	Méthode M.Abbas & F.Bellahcene.....	49

2.3.3 Notations et définitions.....	50
2.4 Notations et définitions.....	53
2.4.1 Solution de la méthode.....	53
2.5 Conclusion.....	56
Chapitre III Optimisation sur un ensemble des solutions efficaces d'un problème MOILPS	
3 Introduction.....	57
3.1 Optimisation sur un ensemble des solutions efficaces d'un problème MOLP.....	58
3.1.1 Résultats de base.....	58
3.1.2 Méthode d'Ecker & Song.....	59
3.2 Notations.....	62
3.3 Optimisation d'une fonction linéaire sur un ensemble efficace d'un problème.....	63
3.3.1 Notations et résultat préliminaire.....	63
3.3.2 Présentation de l'algorithme.....	65
3.3.3 Présentation de l'algorithme.....	68
3.4 Optimisation d'une fonction linéaire sur un ensemble de solution efficace d'un problème MOILPS.....	70
3.4.1 Introduction.....	70
3.4.2 Modélisation du problème.....	71
3.4.3 Teste d'efficacité.....	72
3.4.4 Notation et résultats préliminaires.....	73
3.4.5 Solution de la méthode.....	74
Conclusion et perspectives.....	77
Bibliographie.....	78

INTRODUCTION GENERALE

La recherche opérationnelle est en fait née lors de la deuxième guerre mondiale, des efforts conjugués d'un groupe d'éminents mathématiciens (dont Von Newman, Metropolis, Wald, Weiner, Dantzig et Bellman) de contribuer à leurs manières à la victoire alliée. C'est ainsi qu'il a été créé une nouvelle méthodologie quantitative d'aide à la décision. Rapidement la recherche opérationnelle repousse les frontières de son domaine initial pour s'étendre à la modélisation et l'optimisation du système dans des domaines les plus divers. Allant de la conception et la configuration à l'exploitation des systèmes techniques complexes (réseaux de communications, système informatique), de la gestion stratégique d'investissement à celle de la chaîne logistique (transport, production et stock). La recherche opérationnelle fait aussi son apparition dans les domaines (de la santé et l'instruction publique, la voirie, la distribution du courrier,...), apporte ainsi une méthodologie de modélisation et une collection d'outils mathématiques, éventuellement la statistique et la théorie des probabilités afin de créer un modèle du problème. L'approche classique des problèmes de décision, c'est-à-dire l'optimisation d'une unique fonction économique, montre certaines faiblesses auxquelles les méthodes multicritères semblent pallier. Au lieu d'une solution optimale il existe plutôt un ensemble de solution efficaces ou Pareto optimale. Le développement du concept de modèle d'optimisation linéaire et non linéaire présume que toutes les données du problème soient connues avec certitude. Cependant, l'incertitude des données répand à plusieurs aspects de problème de décision. En 1950, Dantzig et Beale commençaient à travailler sur l'optimisation linéaire dans un environnement incertain, plus tard en 1962 Benders propose une méthode de résolution. Plusieurs applications sont faites dans le cas linéaire, non linéaire et discret. Les approches de l'optimisation dans un environnement incertain sont la programmation stochastique (modèles avec recours, modèles probabilistes), programmation floue (programmation flexible) et la programmation stochastique dynamique. Le type de problème stochastique qui peut être défini traite ce qui s'appelle les problèmes de recours. Dans le modèle le plus simple nous avons deux étapes : c'est prendre une décision puis observation de la réalisation des paramètres stochastiques du problème mais somme permis de prendre d'autres décisions pour éviter l'infaisabilité des contraintes du problème. Le concept de recours et appliqué à la programmation linéaire, non linéaire et en nombre entière. L'objectif de ce mémoire est d'optimiser une fonction linéaire sur un ensemble de solutions efficaces d'un problème linéaire multi-objectif stochastique en nombre entière (MOILPS). Ce travail est constitué d'une introduction générale, de trois parties essentielles et d'une conclusion

générale. La première partie porte sur des généralités de la programmation linéaire uni-critère et multicritère dans le cas continue et discret. Dans la deuxième partie nous avons définis la programmation stochastique uni-critère avec recours et présenter une la méthode des coupes planes présenté dans [1]. Dans la troisième partie nous avons proposé une solution a notre méthode ont introduisons un algorithme de résolution. Enfin ce mémoire est clôturé par une conclusion générale et bibliographie.

Chapitre I

Optimisation multi-objective dans un environnement certain

Motivation

Le développement des concepts des modèles d'optimisation linéaire et non linéaire présumés que toutes les données pour le modèle d'optimisation sont connues avec certitude. Cependant l'incertitude des données et des résultats s'infiltrer l'aspect de la plupart des problèmes d'optimisations, et il s'avère que, quand l'incertitude est d'une forme particulière, il est relativement facile d'incorporer l'incertitude au modèle d'optimisation.

1. Introduction

L'optimisation combinatoire regroupe une large classe de problèmes ayant des applications dans de nombreux domaines applicatifs. Un problème d'optimisation combinatoire est défini par un ensemble fini de solutions discrètes est une fonction objectif associant à chaque solution une valeur (La plus part du temps réelle). Ainsi un problème d'optimisation combinatoire consiste en l'optimisation (maximisation où minimisation) d'un certain critère sous différentes contraintes permettant de délimiter l'ensemble des solutions réalisables (où solutions admissibles). La modélisation classique des problèmes de recherche opérationnelle n'a connu que des modèles qui prennent en compte qu'un seul critère d'évolution des solutions possibles, qui ne répond en aucun cas à la réalité, une nouvelle vision nécessite de modéliser un problème en sorte que tous les objectifs soient pris en considérations pour une meilleur approche de la réalité.

Pour un même problème, faire optimiser plusieurs critères mènent à de nouvelles méthodes d'optimisation et faire effondrer toutes les notions classiques de l'optimisation uni-critère, c'est-à-dire on aura plus de solutions optimal, donc plus de convergence d'une méthode. Dans ce chapitre nous proposons de passer en revue quelques notions et définitions de base de la programmation linéaire uni critère et multicritère, dans le cas continue et entier.

1.1 Résolution d'un problème d'optimisation combinatoire

Résoudre un problème d'optimisation combinatoire nécessite l'étude de trois points particuliers :

- La définition de l'ensemble des solutions réalisables,
- L'expression de l'objectif à optimiser,
- Le choix de la méthode d'optimisation à utiliser.

Les deux premiers points relèvent de la modélisation du problème, le troisième de sa résolution.

Afin de définir l'ensemble des solutions réalisables, il est nécessaire d'exprimer l'ensemble des contraintes du problème ceci ne peut être fait qu'avec une bonne connaissance du

problème sous étude et de son domaine d'application. La programmation linéaire peut être utilisée à cet effet.

Le choix de l'objectif à optimiser requiert également une bonne connaissance du problème. La définition de la fonction objectif mérite toute l'attention de l'analyste car rien ne sert de développer de bonnes méthodes d'optimisation si l'objectif à optimiser n'est pas bien défini. Enfin, le choix de la méthode de résolution à mettre en oeuvre dépendra souvent de la complexité du problème. En effet, suivant sa complexité le problème pourra ou non être résolu de façon optimale. Dans le cas de problèmes classés dans la classe P, un algorithme polynomial a été mis en évidence. Il suffit donc de l'utiliser. Dans le cas de problèmes NP-difficiles, deux possibilités sont offertes. Si le problème est de petite taille, alors un algorithme exact permettant de trouver la solution optimale peut être utilisé (procédure de séparation et évaluation (Branch & Bound), programmation dynamique...). Malheureusement, ces algorithmes par nature énumératifs, souffrent de l'explosion combinatoire et ne peuvent s'appliquer à des problèmes de grandes tailles (même si en pratique la taille n'est pas le seul critère limitant). Dans ce cas, il est nécessaire de faire appel à des heuristiques permettant de trouver de bonnes solutions approchées. Parmi ces heuristiques, on trouve les métaheuristiques qui fournissent des schémas de résolution généraux permettant de les appliquer potentiellement à tous les problèmes. En particulier, la définition de l'objectif est cruciale mais peut être difficile à réaliser, surtout lors de l'étude de problèmes réels.

1.2 Programmation linéaire

Un programme linéaire est un problème d'optimisation consiste à maximiser ou minimiser une fonction objective linéaire de n variables de décisions soumises à un ensemble de contraintes exprimées sous formes d'équations ou d'inéquations linéaire.

A l'origine le terme programme a le sens de planification opérationnelle mais il est aujourd'hui employé comme synonyme de problème (d'optimisation). La terminologie est due à G.B Dantzig, inventeur de l'algorithme simplexe (1947).

La forme générale d'un problème linéaire peut s'écrire sous la forme :

$$(PL) \begin{cases} \min(\text{ou max}) Z = cx \\ \text{t.q.} \\ Ax = b \\ x \geq 0 \end{cases} \quad (1.1)$$

Avec $c \in \mathbb{R}^{1 \times n}$, $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^{n \times 1}$ et $b \in \mathbb{R}^{m \times 1}$

1.2.1 Résultat d'une optimisation linéaire :

Le domaine admissible d'un problème linéaire (PL) peut être :

- Vide dans un tel cas le problème est sans solutions admissibles (et ne possède évidemment pas de solutions optimales)
- Borné (et non vide) le problème possède toujours au moins une solution optimale quelque soit la fonction objective.
- Non borné ; dans ce cas la selon la fonction objective choisie

Le problème peut posséder des solutions optimales ;

Il se peut exister des solutions admissibles de valeurs arbitrairement petite (ou grande). Dans un tel cas, le problème linéaire n'admet pas de solutions optimales finies et est dit non borné.

1.2.2 Terminologie

Une solution est une affectation de valeurs aux variables du problème. Une solution est admissible si elle satisfait toutes les contraintes (y compris les contraintes d'intégrités). La valeur d'une solution est la valeur de la fonction objective en cette solution. Le domaine admissible d'un problème linéaire PL est l'ensemble des solutions admissibles du problème.

La solution optimale d'un PL (si elle existe) est formée des valeurs optimales des variables du problème et de la valeur associée de la fonction objective.

1.2.3 Interprétation géométrique

L'ensemble des solutions d'une inéquation (linéaire) correspond à un demi-espace dans \mathbb{R}^n (un demi plan de \mathbb{R}^2). L'ensemble des solutions d'une équation (linéaire) correspond à un hyperplan dans \mathbb{R}^n (une droite de \mathbb{R}^2).

L'ensemble des solutions d'un système d'équations et d'inéquations (linéaires) correspond à l'intersection des demi-espaces et des hyperplans associés à chaque élément du système.

Cette intersection, appelée domaine admissible, est convexe et définit un polyèdre dans \mathbb{R}^n , (une région polygonale dans \mathbb{R}^2).

Reprenons le problème (1.1) défini précédemment :

On suppose que la matrice A est de rang égal à m . Soit B une sous matrice carrée constituée de m colonnes indépendantes de A (constitue une base). Le problème précédent peut être réécrit sous forme :

$$(P_B) \begin{cases} \min Z = c_B x_B + c_N x_N \\ t.q. \\ Bx_B + Nx_N = b \\ x_B, x_N \geq 0 \end{cases} \quad (1.2)$$

Où $B(m \times m)$, $N(m \times n - m)$ et $A(B, N)$.

Les m composants de x_B sont appelées variables de bases. Notons I l'ensemble des indices de bases : $x_B = (x_i, i \in I)$

Les $n - m$ composantes de x_N sont appelées variables hors bases. Notons J l'ensemble des indices hors base : $x_N = (x_j, j \in J)$

La solution du système $Ax = b$ obtenue pour $x_N = 0$ est appelée la solution de base associée à la base B . Cette solution de base est donc $x_B = B^{-1}b, x_N = 0$.

Une solution de base est admissible si $x_B \geq 0$.

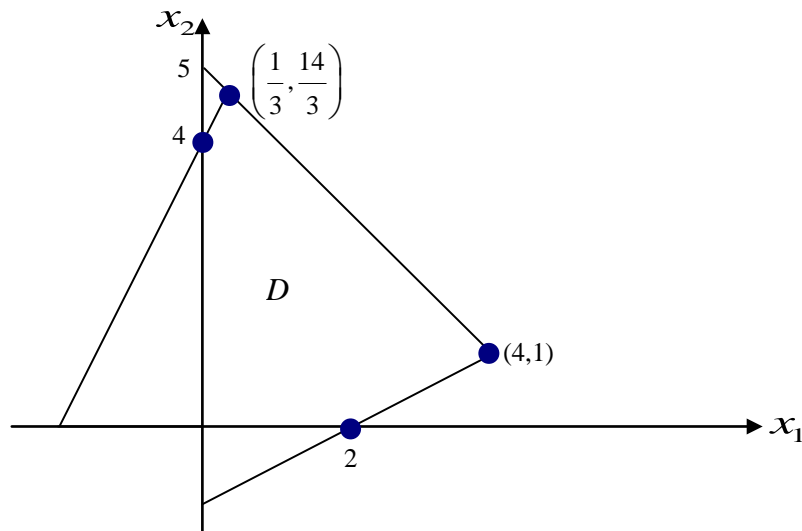
Une solution de base est dite dégénérée si x_B a des composantes nulles.

L'exemple suivant illustre quelques propriétés décrites ci-dessus.

Exemple : soit le problème :

$$\begin{cases} \max Z = 2x_1 + x_2 \\ t.q. \\ x_1 + x_2 \leq 5 \\ -2x_1 + x_2 \leq 4 \\ 2x_1 - 4x_2 \leq 4 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

La représentation géométrique (voir Fig 1.1) est la suivante :



L'intersection des cinq demi-plans représente les contraintes formes un polygone. Les points de la figure correspondent aux solutions de bases.

La solution optimale $x^* = (4,1)$ avec $Z^* = 9$.

1.2.4 Forme équivalente

Le problème peut se mettre sous une forme équivalente en :

- Diagonalisant les contraintes par rapport aux variables de bases.
- Exprimer les variables de base et la fonction économique en fonction de variables hors base. D'après P_B ; il vient par multiplication par B^{-1} :

$$x_B + B^{-1}Nx_N = B^{-1}b$$

Et donc ; en remplaçant x_B

$$Z = C_B B^{-1}b - (C_B B^{-1}N - C_N)x_N$$

Comme x_B est admissible on obtient le problème :

$$\begin{cases} \min Z - c_B B^{-1}b = (c_B B^{-1}N - c_N)x_N \\ t.q. \\ B^{-1}b - B^{-1}Nx_N \geq 0 \\ x_N \geq 0 \end{cases} \quad (1.3)$$

1.2.5 Recherche d'un sommet de départ

1.2.5.1 Forme simpliciale (standard)

Le problème étant sous la forme standard :

$$\begin{cases} \min_x Z = cx \\ Ax = b, x \geq 0 \end{cases} \quad (1.4)$$

$c(n \times 1)$, $A(m \times n)$, $x(n \times 1)$ et $b(m \times 1)$

Il peut se décomposer sous la forme :

$$\begin{bmatrix} B & H \end{bmatrix} \begin{bmatrix} x_B \\ x_H \end{bmatrix} = b$$

Supposons que B est inversible, alors

$$\begin{bmatrix} 1 & B^{-1}H \end{bmatrix} \begin{bmatrix} x_B \\ x_H \end{bmatrix} = B^{-1}b$$

Si $B^{-1}b > 0$ alors la solution $x_B = B^{-1}b$, $x_H = 0$ est une solution de base admissible.

Cette approche nécessite :

- L'inversion d'une matrice ;
- La recherche d'une matrice telle que $B^{-1}b > 0$. La forme simpliciale permet d'éliminer ces inconvénients. En effet, la mise en évidence d'une solution est particulièrement simple si $B = I$ et $b > 0$.

On appellera forme simpliciale une expression des contraintes égalités sous la forme

$$\begin{bmatrix} 1 & H \end{bmatrix} \begin{bmatrix} x_B \\ x_H \end{bmatrix} = b, \text{ avec } b \geq 0 \text{ On a alors une solution directe : } x_B = b, x_H = 0.$$

1.2.5.2 Cas où la forme simpliciale n'est pas évidente

Lorsqu'il n'existe pas de forme simpliciale de départ évidente pour le problème sous forme standard on rajoute des variables, dites artificielles, pour faire apparaître une telle forme, le problème devient :

$$\begin{cases} \min \sum_{i=1}^m y_i \\ t.q. \\ Ax + Y = b \\ x \geq 0, Y \geq 0 \end{cases} \quad (1.5)$$

$A(m \times n), x(n \times 1), b(m \times 1)$, Pour ce problème on a une forme simpliciale évidente $\begin{bmatrix} 1 & A \end{bmatrix} \begin{bmatrix} x_B \\ x_H \end{bmatrix} = b$.

La méthode des deux phases permet alors de déterminer une forme simpliciale du problème de départ. Son principe est le suivant :

- On résout le problème (1.5) par l'algorithme du simplexe.
- Si la solution de ce problème conduit à une solution avec des variables artificielles non nulles ($\sum_{i=1}^m y_i \neq 0$), le problème initial n'a pas de solution.
- Si la solution de ce problème conduit à une solution avec des variables artificielles nulles ($\sum_{i=1}^m y_i = 0$), on utilise la solution ainsi obtenue comme sommet de départ pour le problème initial (on supprime les variables artificielles et on a ainsi une forme simpliciale du problème initial).

1.2.5.3 Passage d'un sommet à un autre

Dès que l'on quitte un sommet, une des variables hors base correspondantes au moins devient non nulle. Comme on veut de diriger vers un sommet voisin, il ne faut en rendre non nulle qu'une, qui va ainsi devenir variable de base pour le prochain sommet, donc entrer dans la base.

1.2.5.4 Choix de la variable entrante

En exprimant le critère Z en fonction des variables hors base. Le critère de sélection de la variable entrante est de prendre la variable hors base qui fournit la plus forte baisse de la fonction objectif. Autrement dit, on choisit celle dont le coût marginal est le plus négatif.

1.2.5.5 Choix de la variable sortante

En considérant le problème sous forme simpliciale, les contraintes lorsqu'une variable hors base x_r rentre dans la base sont :

$$\begin{aligned} x_1 + a_{1r} &= b_1 \\ \vdots & \\ x_s + a_{sr} &= b_s \\ \vdots & \\ x_m + a_{mr} &= b_m \end{aligned}$$

La solution doit être admissible, donc les x_i sont positifs ou nuls. On va donc "pousser" la variable entrante jusqu'à ce qu'une des variables de base s'annule. C'est cette variable qui est choisie comme variable sortante. Pour les i tels que $a_{ir} > 0$, on choisit le plus petit des $\frac{b_i}{a_{ir}}$,

noté $\frac{b_s}{a_{sr}}$ et on donne à x_r la valeur $\frac{b_s}{a_{sr}}$, alors $x_s = 0$.

Si tous les a_{ir} sont inférieurs à zéro, alors la solution est non bornée.

1.2.5.6 Calcul du nouveau sommet

Le calcul du nouveau sommet consiste à remplacer

Les a_{ij} par $a_{ij} - a_{jr} \frac{a_{si}}{a_{sr}}$.

Les b_{ij} par $a_{ij} - a_{jr} \frac{b_{si}}{b_{sr}}$.

Ce calcul est appelé "opération de pivot".

1.2.5.7 Test d'optimalité

Le critère d'arrêt est le suivant :

La solution de base courante est optimale si, lorsque la fonction économique est exprimée avec les variables hors base, aucune des variables ne conduit à l'amélioration du critère. Autrement dit, si tous les coûts réduits sont positifs ou nuls.

1.2.6 Dualité

A tous problème linéaire on associe un autre problème linéaire appelé dual du problème initial. (Appelé primal).

D'une manière générale on écrit :

Problème linéaire avec inégalité :

$$\text{Primal : } \left(\begin{array}{l} \min c^t x \\ \text{t.q.} \\ Ax \geq b \\ x \geq 0 \end{array} \right) \quad \text{dual : } \left(\begin{array}{l} \max b^t y \\ \text{t.q.} \\ A^t y \leq c \\ y \geq 0 \end{array} \right)$$

Problème linéaire sous forme standard :

$$\text{Primal : } \left(\begin{array}{l} \min c^t x \\ \text{t.q.} \\ Ax = b \\ x \geq 0 \end{array} \right) \quad \text{dual : } \left(\begin{array}{l} \max b^t y \\ \text{t.q.} \\ A^t y \leq c \\ y \text{ qcq} \end{array} \right)$$

Où y est un vecteur $(1 \times m)$ formé de variables duales.

Théorème 1 : (dualité faible)

Si $x \in \{x / Ax = b, x \geq 0\}$ (x est réalisable pour le problème primal) et si $y \in \{y / A^t y \leq c, y \text{ qcq}\}$ (y est réalisable pour le problème dual), Alors $b^t y \leq c^t x$. $y \in \mathbb{R}$

Corollaire 1 :

Si $x^* \in \{x : Ax = b, x \geq 0\}$ et $y^* \in \{y : A^t y \leq c, y \text{ qcq}\}$, et si $b^t y^* = c^t x^*$, alors x^* et y^* sont des solutions optimales respectivement pour le problème primal et pour le problème dual.

Théorème 2 : (dualité forte)

Si un des deux problèmes primal ou dual possède une solution optimale avec valeur finie, alors la même chose est vraie pour l'autre problème, et les valeurs optimales des deux problèmes sont égales. Si un des deux problèmes n'est pas borné, alors le domaine réalisable de l'autre problème est vide.

1.3 Optimisation combinatoire multi-objectif

Les problèmes d'optimisation combinatoire sont la plupart du temps de nature multi-objectif car plusieurs critères sont à considérer simultanément. Optimiser un tel problème relève donc de l'optimisation combinatoire multi-objective.

Les premières études concernant l'optimisation combinatoire multi-objective transformaient les problèmes multi-objectifs en une succession de problèmes mono-objectifs. Pour cela, un ordre d'importance sur les objectifs pouvait être donné, et l'optimisation consistait à optimiser un objectif sans dégrader les valeurs déjà obtenues pour les objectifs plus prioritaires. Une autre approche consistait en l'optimisation d'une agrégation linéaire des objectifs, chacun pouvant avoir un poids représentant son importance. Lorsque l'on se trouve dans un réel contexte multi-objectif, il n'est pas toujours possible de trouver un ordre d'importance sur les critères. Il est alors nécessaire de rechercher les solutions de meilleur compromis entre les objectifs. Il est facile de voir que dans ce contexte la solution recherchée n'est pas une unique solution mais un ensemble de solutions représentant les différents compromis possibles. Ainsi l'optimisation multi-objective s'intéresse aux particularités liées à l'existence de ces différentes solutions optimales. En particulier, les méthodes de résolution devront être dédiées à ce type de problèmes qui sont la plupart du temps NP-difficiles. Résoudre un problème multi-objectif peut être divisé en deux phases :

1. La recherche des solutions de meilleur compromis. Se pose alors la question de savoir si toutes les solutions doivent être produites (elles peuvent être nombreuses) ou seulement un sous-ensemble représentatif. C'est la phase d'optimisation multi-objective.

2. le choix de la solution à retenir. C'est la tâche du décideur qui, parmi l'ensemble des solutions de compromis, doit extraire celle(s) qu'il utilisera. On parle alors ici de décision multi-objective et cela fait appel à la théorie de la décision.

L'optimisation multi-objective possède ses racines dans les travaux en économie d'Edgeworth [23] et Pareto [40]. Elle a ainsi été initialement utilisée en économie et dans les sciences du management, puis graduellement dans les sciences pour l'ingénieur. Pourtant, malgré l'intérêt indéniable de la modélisation et de la résolution multi-objectif des problèmes rencontrés en industrie, dans les télécommunications, etc. peu de travaux ont été réalisés en optimisation combinatoire multi-objectif avant les années 80-90. Mais depuis, un fort intérêt a été montré pour l'aide à la décision multi-objectif qui consiste pour un problème comportant plusieurs objectifs, à déterminer, parmi les solutions de meilleurs compromis entre les objectifs, la solution la plus intéressante pour le problème en question. Ainsi, une phase importante concerne l'optimisation multi-objective qui recherche les solutions de compromis.

1.3.1 Problème d'optimisation multi-objectif

Le problème multi-objectif consiste à optimiser (minimiser ou maximiser) P fonctions objectives ($(P \geq 2)$). Il se présente comme suit :

$$(MO) \begin{cases} \text{"optimiser"}(f_1(x), f_2(x), \dots, f_p(x)) \\ \text{t.q.} \\ x \in S \end{cases} \quad (1.6)$$

$S = \{x \in \mathbb{R}^n / g_i(x) \leq 0, j:1, 2, \dots, m\}, f_k(k:1, 2, \dots, p)$ et $g_i(j:1, 2, \dots, m)$, sont des fonctions à valeurs réelles du vecteur de décision $x \in \mathbb{R}^n$. Le symbole " " indique qu'il ne s'agit pas de l'optimisation au sens classique. (L'optimisation classique uni-critère). La solution de tel problème consiste à déterminer une solution de meilleur compromis (solution optimale au sens de Pareto) autrement des solutions $x \in S$ correspondant à la préférence globale de décideur.

Si les objectifs f_k et les fonctions g_i sont linéaire, on obtient un problème de programmation linéaire multi-objective généralement il s'écrit sous forme :

$$(MOLP) \begin{cases} \text{"optimiser"} Z_k = c^k x ; k:1, 2, \dots, p \\ x \in S \end{cases} \quad (1.7)$$

Où $c^k \in \mathbb{R}^{1 \times n}$ pour $k:1, 2, \dots, p$ et l'ensemble S est déterminé par les contraintes (équations, inéquations) linéaire :

$$S = \{x \in \mathbb{R}^n / Ax \leq b, x \geq 0\}, c \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^n, b \in \mathbb{R}^m, m, n \in \mathbb{N}$$
 est un polyèdre convexe de \mathbb{R}^n .

.Contrairement à l'optimisation mono-objective, la solution d'un problème multi-objectif n'est pas unique, mais est un ensemble de solutions non dominées, connu comme l'ensemble des solutions Pareto Optimales (PO). Sans perte de généralité nous supposons par la suite que nous considérons des problèmes de minimisation.

1.4 Notions de dominances et d'optimalité

Une solution réalisable $x \in S$ est Pareto optimale (ou efficace, ou encore non dominée) si et seulement si il n'existe pas de solution $x^* \in S$ telle que x domine x^* .

Toute solution de l'ensemble Pareto peut être considérée comme optimale puisque aucune amélioration ne peut être faite sur un objectif sans dégrader la valeur relative à un autre objectif. Ces solutions forment le **front Pareto**.

1.5 Points particuliers

En vue d'avoir certains points de références permettant de discuter de l'intérêt des solutions trouvées, des points particuliers ont été définis dans l'espace objectif. Ces points peuvent représenter des solutions réalisables ou non.

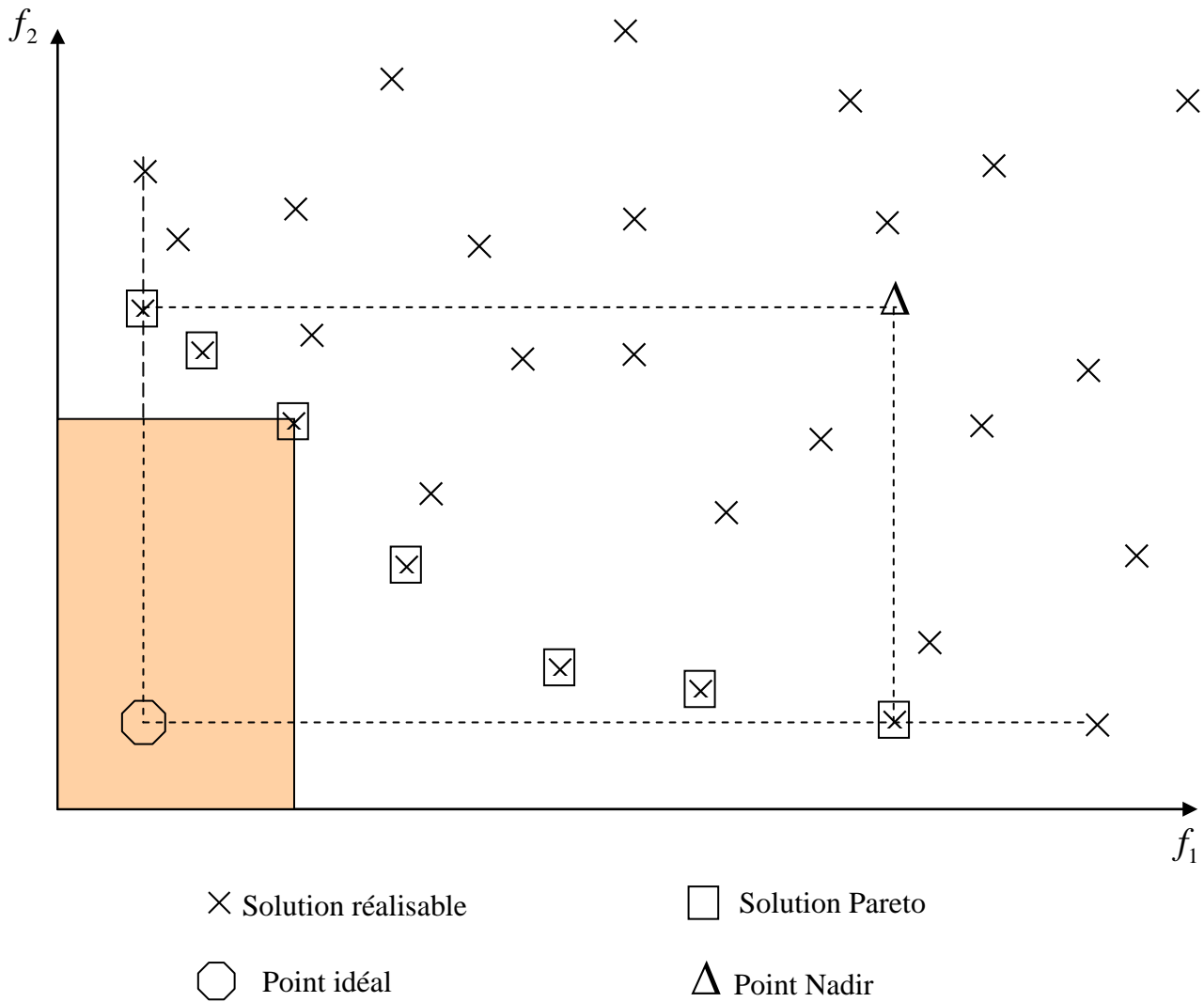


Fig.1.2 Illustration des différentes définitions

Tout d'abord, le **point idéal** est le point \bar{Z}_k de \mathbb{R}^n qui a comme valeur pour chaque objectif la valeur optimale de l'objectif considéré.

$$\bar{Z}_k = \min_{x \in S} Z_k(x) \quad k : 1, 2, \dots, k \quad (1.8)$$

Ce point ne correspond pas à une solution réalisable car si c'était le cas, cela sous entendrait que les objectifs ne sont pas contradictoires et qu'une solution optimisant un objectif, optimise simultanément tous les autres, ce qui ramènerait le problème à un problème ayant une seule solution Pareto optimale.

1.5.1 Matrice des gains

Soit x^{1j} une solution optimale du critère Z_j , la matrice $(p \times p)$ formée des éléments $Z_{kj} = Z_k(x^{1j})$ est dite matrice des gains (*payoff matrix*).

$$\begin{pmatrix} \bar{Z}_1 \dots & Z_{1j} \dots & Z_{1p} \\ Z_{k1} \dots & Z_{kj} \dots & Z_{kp} \\ Z_{p1} \dots & Z_{pj} \dots & \bar{Z}_p \end{pmatrix}$$

Les éléments sur la diagonale représentent les coordonnées du point idéal.

Lorsqu'un critère j à plusieurs solutions optimales et donc pour un problème MOLP une infinité. La colonne j de la matrice des gains dépendra de la solution x^{1j} choisie.

Le point $\eta \in \mathbb{R}^p$ de coordonnées $\eta_k = \max Z_{kj}, j: 1, 2, \dots, p$ où Z_{kj} est un élément de la matrice des gains est appelé le **point nadir**.

Le **point anti-idéal** \underline{Z} est de coordonnées $\underline{Z}_k = \max_{x \in S} (Z_{kj}), k: 1, 2, \dots, p$ (1.9)

Considérons le problème (MOLP), $E(P)$ représente l'ensemble des solutions efficaces et $SE(P)$ l'ensemble des solutions non efficaces.

Soit Λ l'ensemble de tous les vecteur $\lambda = (\lambda_i), i: 1, 2, \dots, p$ définie par :

$$\Lambda = \left\{ \lambda \in \mathbb{R}^p / \sum_{i=1}^p \lambda_i = 1, \lambda_i \geq 0, i: 1, \dots, p \right\}$$

Pour $\lambda \in \Lambda$ on définit le problème P_λ par :

$$(P_\lambda) = \{ \min \lambda_i Z_i(x) \quad (1.10) \}$$

$$Z = (Z_1(x), Z_2(x), \dots, Z_p(x)) \in \psi(S).$$

Définition 1:

L'espace \mathbb{R}^n dans le quel se situe l'ensemble des actions $S(S \subset \mathbb{R}^n)$ est appelé espace de décision.

L'espace \mathbb{R}^p dans le quel se situe $\psi(S)$ (l'image de S par ψ) est appelé espace des critères.

Théorème 3: (Géoffrion 1968) [26]

Etant donné le problème P_λ , alors \tilde{Z} est un vecteur non dominé pour le problème MOLP si et seulement si \tilde{Z} est une solution optimale du problème paramétrique (P_λ) .

1.5.2 Fonctions scalarisantes : Les problèmes de décisions nécessitent de sélectionner une solution de "meilleur compromis", cette sélection exige une information supplémentaire relative à la structure de préférence, les plus courantes sont :

- Les poids λ_k ($k : 1, 2, \dots, p$) qui reflètent l'importance relative de chaque critère ;
- Les taux de substitution qui traduisent l'idée de compensation entre une perte sur un critère et un gain sur un autre ;
- Les points de référence qui représentent des points de \mathbb{R}^p dont les coordonnées sont des valeurs souhaitables (qu'il faut tenter d'atteindre) ou non souhaitables (dont il faut tenter de s'écarter) des différents critères ;
- Les niveaux de réservation qui correspondent à des exigences minimales imposées sur les valeurs de l'un ou l'autre critère ;
- Les poids, les taux de substitution, où les points de référence servent en général à agréger les différents critères en une fonction unique. Ces fonctions d'agrégation sont appelées fonctions scalarisantes les plus couramment utilisées ;

- La somme pondérée : $S_1(Z, \lambda) = \sum_{k=1}^p \lambda_k Z_k$;

- La distance pondérée de Tchebychev : $S_2(Z, \lambda, \bar{Z}) = \max(\lambda_k |Z_k - \bar{Z}_k|), k : 1, \dots, p$

- La distance pondérée augmentée de Tchebychev :

$$S_3(Z, \lambda, \bar{Z}) = \max \lambda_k |Z_k - \bar{Z}_k| + \rho \left(\sum_{k=1}^p Z_k - \bar{Z}_k \right), \rho > 0, k : 1, \dots, p$$

1.6 Optimisation linéaire multi-objective en nombre entiers :

1.6.1 Formulation des problèmes uni-critères en variables discrètes :

Considérons la formule mathématique d'un problème linéaire uni-critère suivante :

$$(P) \begin{cases} \min cx \\ x \in S \end{cases} \quad (1.10)$$

$S = \{x \in \mathbb{R}^n / Ax = b, x \geq 0\}$ Est un polyèdre convexe dans \mathbb{R}^n .

Dans beaucoup d'applications réelles, les variables de décisions peuvent prendre des valeurs entières, ainsi le problème (LP) s'écrit sous forme :

$$(IP) \begin{cases} \min cx \\ x \in D \end{cases} \quad (1.11)$$

$$D = \{x \in \mathbb{R}^n / Ax = b, x \geq 0, x \text{ entier}\},$$

$$c \in \mathbb{R}^{1 \times n}, A \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^{n \times 1} \text{ et } b \in \mathbb{R}^{m \times 1}$$

$D = S \cap \mathbb{Z}^n$, \mathbb{Z} est l'ensemble des nombres entiers relatif.

Soit $F(P)$ l'ensemble des solutions réalisables du problème (P) et soit $F(IP)$ l'ensemble des solutions réalisables du problème (IP) alors $F(P) \subseteq F(IP)$.

Soit x^* une solution de (P) et \bar{x} une solution de (IP) alors on a $C\bar{x} \leq Cx^*$.

1.6.2 Complexité

Il est facile de montrer que la PLNE est un problème NP-complet car de nombreux problèmes NP-complets peuvent être exprimés comme des PLNE. Beaucoup d'algorithmes pour la programmation linéaire ne résolvent pas les problèmes de PLNE. En revanche, la relaxation continue d'un PLNE (PLNE sans les contraintes d'intégrité) est un problème linéaire qui peut être résolu efficacement.

Les algorithmes de résolution de PLNE, tels que les algorithmes par séparation et évaluation, se basent très souvent sur cette relaxation continue

1.6.3 Coupe de Dantzig [50]

Dantzig en 1959 est le premier à avoir proposé une coupe pour résoudre de tel problème linéaire, l'idée en premier est de résoudre le problème linéaire sans conditions entières, alors un nouveau ensemble de valeurs de variables hors base doit être assuré et que leurs somme est supérieur ou égale à un ; si x_j ($j \in J$ indice hors base) sont les variables entières hors base, alors $\sum_{j \in J} x_j \geq 1$ doit être satisfait. Cette coupe actuellement n'est pas utilisée vu sa limite

d'utilisations et son inefficacité

1.6.4 Coupe de Gomory

La coupe de Gomory est une contrainte linéaire employée en même temps que la méthode du simplexe pour avoir des solutions optimales pour un problème linéaire en nombres entiers.

Soit le problème :

$$(LP) \begin{cases} t.q. \\ \min cx \\ Ax = b \\ x \geq 0, x \text{ entier} \end{cases} \quad (1.12)$$

$$(LP) \text{ relaxation} \begin{cases} \text{t.q.} \\ \min cx \\ Ax = b \\ x \geq 0 \end{cases} \quad (1.13)$$

Résoudre le problème relaxé et soit x^* la solution optimale de base obtenue.

Si x^* est entier, alors x^* est solution de PLNE

Sinon, si il existe au moins une variable de base non entière, on génère une inéquation sur la contrainte associée, afin de couper la région de faisabilité courante.

Soit x^* la solution de base optimale et B la base optimale ($B = (I, N)$) du tableau simplexe optimale on a :

$$x_B + B^{-1}Nx_N = B^{-1}b \quad (1)$$

$$\text{Posons } \bar{a}_{ij} = (B^{-1}N)_i, \bar{a}_{i_0} = B^{-1}b$$

$$\text{L'équation (1) se réécrit : } x_i + \sum_{j \in J} \bar{a}_{ij} x_j = \bar{a}_{i_0} \quad (2)$$

Comme $x_j \geq 0$ pour tous j alors :

$$x_i + \sum_{j \in J} \lfloor \bar{a}_{ij} \rfloor x_j \leq x_i + \sum_{j \in J} \bar{a}_{ij} x_j = \bar{a}_{i_0} \quad \text{Avec } \lfloor \bar{a}_{ij} \rfloor \text{ est la partie entière de } (\bar{a}_{ij}) \text{ comme } x_j \text{ sont}$$

$$\text{entières alors } x_i + \sum_{j \in J} \lfloor \bar{a}_{ij} \rfloor x_j \leq \lfloor \bar{a}_{i_0} \rfloor \quad (3)$$

$$(2)-(3) \text{ nous donnons : } \sum_{j \in J} (\bar{a}_{ij} - \lfloor \bar{a}_{ij} \rfloor) x_j \leq (\bar{a}_{i_0} - \lfloor \bar{a}_{i_0} \rfloor)$$

$$\text{Posons } f_{ij} = (\bar{a}_{ij} - \lfloor \bar{a}_{ij} \rfloor), f_{i_0} = (\bar{a}_{i_0} - \lfloor \bar{a}_{i_0} \rfloor)$$

$$f \text{ est la partie fractionnaire } 0 \leq f < 1 \text{ on obtient : } \sum_{j \in J} f_{ij} x_j \leq f_{i_0}$$

On introduit une variable d'écart S_k :

$$S_k = \sum_{j \in J} f_{ij} x_j + f_{i_0} \quad (4) \text{ Est la coupe fractionnaire de Gomory.}$$

1.7 Problème multi-objectif en variables discrètes (MOILP)

La programmation linéaire multi-objective en nombre entiers, est essentiellement appliquée pour résoudre des problèmes réels à titre d'exemple, les problèmes du budget capital, les problèmes de location, gestion de portefeuille, transport, emploi du temps, ordonnancement, etc. A présent il existe un nombre important de méthode pour résoudre le problème MOILP. Les Algorithmes de Sylva & Crema [49], Klein & Hannan [35] et Villareal & Karwan [55]

sont désigné afin de générer l'ensemble de toutes les solutions efficaces. On peut aussi se rappeler de quelques méthodes interactives tel que la méthode de Gonzales & al, Karwan & al,...

1.7.1 Structure générale du problème MOILP

Considérons le problème MOILP suivant :

$$(MOILP) \begin{cases} \min Z(x) = C^k x, k : 1, \dots, k \\ x \in D \end{cases} \quad (1.14)$$

$D = S \cap \mathbb{Z}^n, \mathbb{Z}$ L'ensemble des entiers relatifs.

Notons $IE(P)$ l'ensemble de toutes les solutions efficaces du problème MOILP.

L'ensemble des solutions efficaces du problème MOILP $E(\cdot)$ et bien caractérisé Voir [26], par la solution du problème paramétrique $(P_\lambda) = \{\min \lambda_i Z_i(x)\}$. Ce principe, souvent dit « principe de Geoffrion » (voir le théorème), n'est plus valable lorsqu'on traite les problèmes MOILP car l'ensemble d'admissibilité n'est pas convexe sur cette base.

L'ensemble des solutions efficaces se repartie en deux sous-ensembles. L'ensemble des solutions optimales du problème paramétrique (P_λ) ou S est remplacé par D et noté $SE(\cdot)$; ses solutions sont appelées efficaces supportées (supported efficient solution). L'autre sous ensemble est formé de solutions efficaces non supportées $E(MOILP) \setminus SE(MOILP)$ est noté $SE(\cdot)$ (non supported efficient solutions).

1.7.2 Solutions supportées / non supportées

Sur le front Pareto, deux types de solutions peuvent être différenciées : les solutions supportées et les solutions non supportées. Les premières sont celles situées sur l'enveloppe convexe de l'ensemble des solutions et peuvent donc être trouvées à l'aide d'une agrégation linéaire des objectifs [26]. Elles sont donc plus simples à obtenir que les solutions non supportées. D'ailleurs, les premiers travaux en optimisation combinatoire multi-objectif se sont pour la plupart focalisés sur la recherche de ces solutions supportées en optimisant des combinaisons linéaires des objectifs utilisant différents vecteurs de poids. Alors pourquoi ne pas se satisfaire des solutions supportées ? Tout d'abord parce que ces solutions peuvent ne représenter qu'un petit sous-ensemble des solutions efficaces. De plus, ces solutions supportées ne sont pas forcément bien réparties le long du front et ne représentent pas toujours un bon compromis. Donc, si l'on veut obtenir des solutions de bon compromis entre les objectifs, il est nécessaire de considérer les solutions Pareto non supportées.

Illustrons par un exemple numérique la notion des solutions supportées et non supportées (exemple introduit par Bowman [13]).

Exemple [2] :

$$(P) \begin{cases} \text{t.q.} \\ \max Z_1 = 6x_1 + 3x_2 + x_3 \\ \max Z_2 = x_1 + 3x_2 + 6x_3 \\ x_1 + x_2 + x_3 \leq 1 \\ x_1, x_2, x_3 \in \{0, 1\} \end{cases}$$

Il est clair d'après le théorème de Geoffrion que les solutions optimales du problème :

$$(P_\lambda) \begin{cases} Z_1 = \lambda(6x_1 + 3x_2 + x_3) + (1-\lambda)(x_1 + 3x_2 + 6x_3) \\ \text{t.q.} \\ x_1 + x_2 + x_3 \leq 1 \\ x_1, x_2, x_3 \in \{0, 1\} \quad 0 < \lambda < 1 \end{cases}$$

Sont des solutions efficaces.

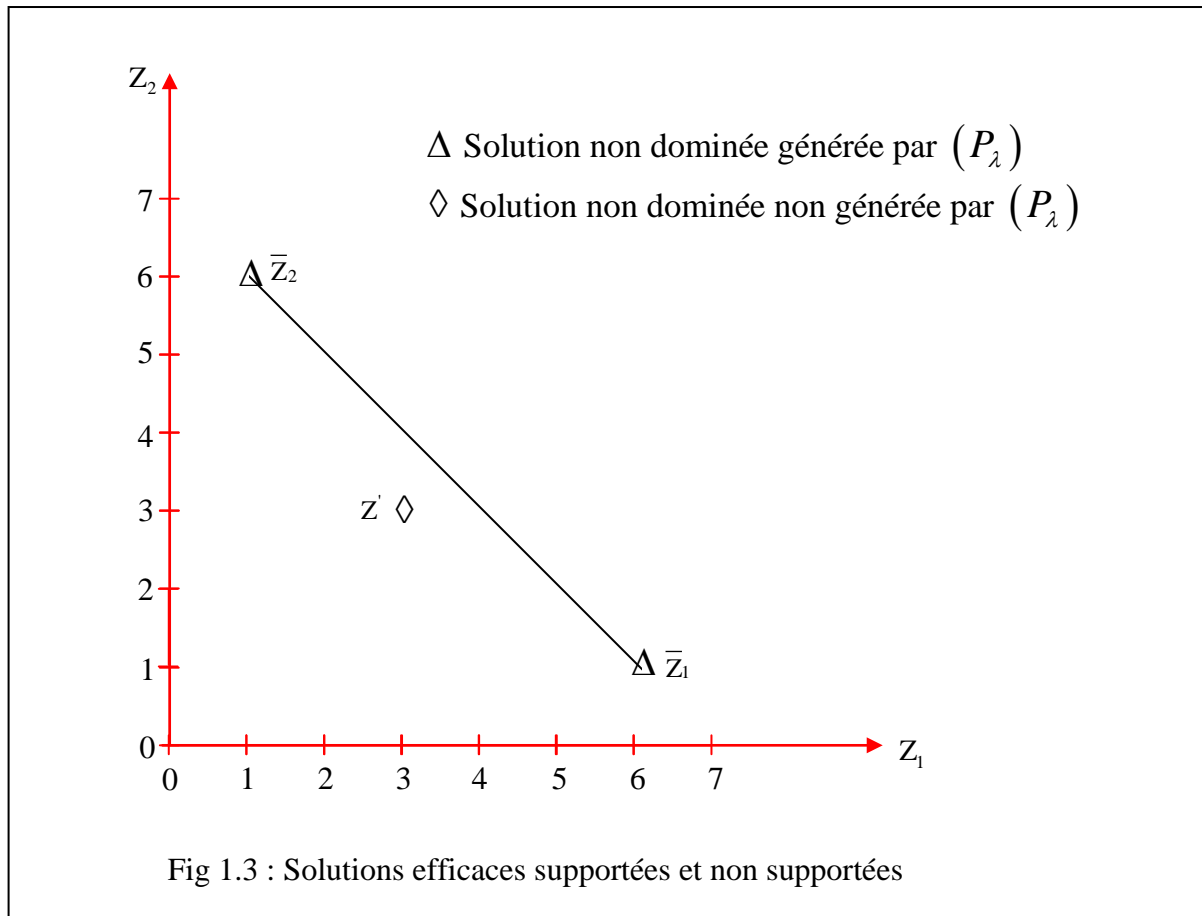
L'ensemble des solutions admissible est formé de 3 solutions :

$$X_1 = (1, 0, 0), X_2 = (0, 1, 0), X_3 = (0, 0, 1)$$

Le tableau ci dessous résume l'énumération de toutes les solutions efficaces et les valeurs correspondantes des objectifs.

	Z_1	Z_2
X_1	6	1
X_2	3	3
X_3	1	6

X_1, X_3 sont des solutions efficaces, solutions de (P_λ) , mais la solution X_2 qui est aussi efficace n'est pas optimale du problème paramétrique (voir la figure 1.3)



1.7.3 Choix de la méthode d'aide à la décision

La résolution d'un problème multi-objectif menant à la détermination d'un ensemble de solutions Pareto, il est nécessaire de faire intervenir un décideur, pour le choix final de la solution à garder. Ainsi, avant de se lancer dans la résolution d'un problème multi-objectif, il faut se poser la question du type de méthode d'optimisation à utiliser. En effet, on peut répartir les méthodes de résolution de problèmes multi-objectifs en trois familles, en fonction du moment où intervient le décideur. Ainsi nous pouvons trouver les familles suivantes :

– **Les méthodes d'optimisation a priori** : dans ce cas, le compromis que l'on désire faire entre les objectifs a été défini avant l'exécution de la méthode. Ainsi une seule exécution permettra d'obtenir la solution recherchée. Cette approche est donc rapide, mais il faut cependant prendre en compte le temps de modélisation du compromis et la possibilité pour le décideur de ne pas être satisfait de la solution trouvée et de relancer la recherche avec un autre compromis.

Dans cette approche le problème linéaire multicritère est remplacé par un problème uni-critère dont la fonction objective est la fonction scalarisante $\max_{z \in \mathcal{P}(S)} S(Z, \lambda)$.

– **Les méthodes d’optimisation progressives** : ici, le décideur intervient dans le processus de recherche de solutions en répondant à différentes questions afin d’orienter la recherche.

Cette approche permet donc de bien prendre en compte les préférences du décideur, mais nécessite sa présence tout au long du processus de recherche.

– **Les méthodes d’optimisation a posteriori** : dans cette troisième famille de méthodes, on cherche à fournir au décideur un ensemble de bonnes solutions bien réparties. Il peut ensuite, au regard de l’ensemble des solutions, sélectionner celle qui lui semble la plus appropriée. Ainsi, il n’est plus nécessaire de modéliser les préférences du décideur (ce qui peut s’avérer être très difficile), mais il faut en contre-partie fournir un ensemble de solutions bien réparties, ce qui peut également être difficile et requérir un temps de calcul important (mais ne nécessite pas la présence du décideur).

Dans ce type de méthode, deux phases importantes sont à considérer : la phase de recherche de l’ensemble des solutions Pareto optimales, que nous appellerons de façon abusive, **Résolution du problème d’optimisation** et la phase de choix parmi ces solutions, qui relève de **l’aide à la décision**.

Dans cette méthode, il s’agit de résoudre le problème paramétrique en $\lambda \left(\max_{z \in \mathcal{P}(s)} S(z, \lambda) \right)$

1.8 Approches classiques de résolution

Sans avoir la prétention de présenter ici toutes les méthodes dédiées à l’optimisation combinatoire multi-objectives, en voici quelques-unes. Pour avoir une vue plus globale, le lecteur peut se référer à [24].

1.8.1 Méthodes exactes pour l’optimisation multi-objective

Concernant les méthodes exactes, plusieurs approches basées sur des procédures de séparation et évaluation (branch and bound) [56], et la programmation dynamique [14] ont été proposées pour résoudre de petits problèmes à deux objectifs (problèmes bi-objectifs). Une approche particulière pour l’optimisation multi-objectif est le ”goal programming” (programmation par buts) [44]. Dans ce type d’approche, le décideur indique une valeur cible (but) et l’objectif est de minimiser l’écart avec cette cible. Souvent la programmation par buts est vue comme une discipline en elle-même, différente de l’optimisation multi-objective.

Une approche intéressante a été proposée par B. Ulungu et J. Teghem pour la recherche du front Pareto de problèmes bi-objectifs [56]. Leur méthode en deux phases consiste dans un premier temps à rechercher l’ensemble des solutions Pareto supportées, puis dans un deuxième temps à rechercher de façon indépendante les solutions non-supportées situées entre tous les couples de solutions supportées adjacentes pour une description précise). Cette

approche a été utilisée efficacement sur des problèmes tels que l'affectation ou le sac à dos bi-objectifs. Cette méthode a ensuite été améliorée afin d'obtenir des fronts complets de façon plus efficace [42]. Pourtant, dès que le nombre d'objectifs ou la taille des problèmes augmentent, les méthodes exactes deviennent inefficaces étant donné la nature NP-difficile des problèmes (déjà en mono-objectif) et l'aspect multi-objectif des problèmes. Ainsi, il est nécessaire afin de résoudre des problèmes de grande taille et/ou des problèmes avec plus de deux objectifs, de faire appel à des méthodes heuristiques. Les méthodes exactes peuvent néanmoins être utiles lorsque des sous-problèmes peuvent être extraits du problème global. Leur résolution permet en effet de contribuer à la recherche de la solution globale,

1.8.2 Méthodes heuristiques

Les méthodes heuristiques ne garantissent pas de trouver de manière exacte tout l'ensemble des solutions Pareto, mais une approximation, aussi bonne que possible, de cet ensemble.

Les approches heuristiques peuvent être classées en trois catégories :

- Approche transformant le problème en un ou plusieurs problème(s) mono-objectif(s) : ces approches transforment le problème initial afin de se ramener à la résolution de un ou plusieurs problèmes mono-objectif. Parmi ces méthodes, on peut citer *les méthodes d'agrégation* [37, 48], *les méthodes ε -contrainte* [29] ou encore *les méthodes utilisant un vecteur cible* [15, 37]. En général ces méthodes nécessitent une connaissance du problème et ne fournissent qu'une seule solution. Elles peuvent alors être classées dans la famille des méthodes d'optimisation a priori, présentée précédemment,
- Approches **Non Pareto** : ces approches transforment le problème d'origine. Elles effectuent leur recherche en traitant indépendamment chacun des objectifs. Ces méthodes ont souvent du mal à trouver les solutions de compromis puisqu'elles se focalisent sur les portions extrêmes du front. Nous pouvons classer dans cette catégorie les méthodes lexicographiques qui donnent un ordre de priorité sur les objectifs à traiter.
- Approches **Pareto** : ces approches utilisent la notion de dominance pour comparer les solutions entre elles. Une seule résolution permet d'approximer l'ensemble de la frontière Pareto.

1.9 Quelques méthodes de résolution de problèmes d'optimisation multi-objectifs

Comme nous l'avons dit précédemment, il existe très peu de méthodes exactes dédiées à la recherche de l'ensemble des solutions Pareto. Après une étude des méthodes existantes, deux méthodes utilisées en bi-objectif ont tout de même retenu notre attention. Il s'agit d'une méthode basée sur l'utilisation de la méthode ε -contrainte et de la méthode deux phases que nous expliquons ci-après.

1.9.1 L'agrégation linéaire

Cette méthode populaire transforme le problème multi-objectif en un problème mono-objectif en combinant linéairement les différents objectifs. Ainsi, le nouveau problème obtenu, car il

s'agit alors d'un problème différent, consiste à $\max_{x \in S} \sum_{k=1}^p \lambda_k Z_k(x)$

Avec $\lambda_k \in \Lambda = \left\{ \lambda_k / \sum_{k=1}^p \lambda_k = 1 \text{ et } \lambda_k > 0, k : 1, 2, \dots, p \right\}$.

Le théorème de Geoffrion [26] indique qu'en utilisant différentes valeurs pour le vecteur λ , il est possible d'obtenir toutes les solutions supportées du problème multi-objectif initial. Par contre, aucune solution non supportée ne peut être trouvée par cette méthode.

La méthode d'agrégation linéaire a donc ses limites. Toutefois, elle est intéressante pour des problèmes ayant de nombreux objectifs et/ou un grand nombre de solutions supportées bien réparties. Dans ce contexte, il peut être suffisant de générer les solutions supportées.

1.9.2 La recherche dichotomique

La recherche dichotomique offre un schéma d'application de l'agrégation linéaire permettant d'obtenir les solutions non supportées [19]. Cette méthode consiste à explorer de façon dichotomique des intervalles de front de plus en plus petits. Tout d'abord les solutions extrêmes sont recherchées. Puis une recherche est menée entre ces solutions x et y suivant une direction perpendiculaire à la droite (x, y) . En interdisant de réobtenir les solutions x et y et en éliminant les solutions dominées par ces solutions, cette recherche trouve la meilleure solution Pareto relativement à cette direction de recherche, solution qui peut alors être non supportée. Cette nouvelle solution crée deux nouveaux intervalles qu'il faut explorer de la même façon.

Cette méthode, dédiée au bi-objectif, est intéressante mais nécessite de l'ordre de 2^n recherches, si n est le nombre de solutions du front Pareto.

1.9.3 Méthode ε -contrainte

Introduite par Haimes, Ladson et Wismer en 1971, cette méthode est basé sur la minimisation d'un objectif Z_l en considérons que les autres objectifs $Z_k, k \neq l$ doivent être inférieur à une valeur ε_k . En général, l'objectif choisi est celui que le décideur souhaite optimiser en priorité :

$$\begin{cases} \min_{x \in S} Z_l(x) \\ t.q. \\ Z_k(x) \leq \varepsilon_k, \forall k \neq l \end{cases} \quad (1.15)$$

Le principe de la méthode ε -contrainte qui consiste, dans le cas bi-objectif, à borner l'un des objectifs (en général le plus difficile à résoudre) et à optimiser l'autre objectif (optimisation mono-objectif) en tenant compte de cette borne [28], est intéressant lorsque l'on cherche à énumérer toutes les solutions d'un front Pareto. En effet, en utilisant cette méthode itérativement, en repartant à chaque fois de la solution trouvée pour définir la borne suivante, il est possible en utilisant une méthode exacte mono-objective de générer, pour des problèmes combinatoires, l'ensemble des solutions Pareto. L'inconvénient principal de cette méthode est qu'elle nécessite une résolution mono-objective pour chacune des solutions du front. Lorsque ce nombre est élevé, cela peut être vu comme une limite, d'autant plus lorsque la méthode de résolution mono-objectif est coûteuse. De plus, lorsqu'il n'existe pas de méthode mono-objective efficace, rechercher une solution particulière (respectant une borne, par exemple) est souvent synonyme d'énumération de nombreuses autres solutions dont certaines peuvent être Pareto optimales. Ainsi certaines solutions seront énumérées plusieurs fois sans que la méthode les repère.

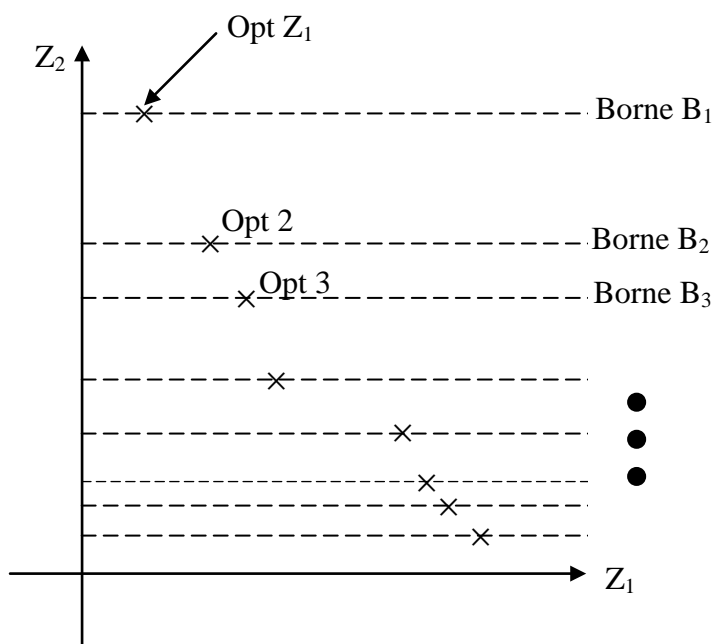


Fig. 1.4 – Illustration de la méthode ε -contrainte.

La figure 1.4 illustre un exemple pour lequel, la solution efficace optimale pour l'objectif Z_1 est d'abord recherchée (solution Opt_{Z_1}). Cette solution détermine la borne B1 sur l'objectif Z_2 en dessous de laquelle l'objectif Z_2 va devoir être optimisé. Cela nous donne la solution Opt_{Z_2} qui elle-même détermine la borne B2, etc.

1.9.4 Goal programming: (Target vector optimisation)

Le décideur fixe un but $Z_k^*, k:1, \dots, p$ à atteindre et minimise la somme des écarts entre les résultats et ces buts

$$\min_{x \in S} \sum_{k=1}^p |Z_k(x) - Z_k^*| \quad (1.16)$$

La définition des poids et des buts à atteindre est une équation délicate qui détermine l'efficacité de la méthode.

1.9.5 Méthode deux-phases

La méthode deux-phases a initialement été proposée par Ulungu et Teghem pour la résolution d'un problème d'affectation bi-objectif [56]. Comme son nom l'indique, cette méthode est décomposée en deux étapes : la première consiste à trouver toutes les solutions supportées du front Pareto, puis la deuxième phase cherche entre ces solutions les solutions Pareto non supportées. Cette méthode travaille donc essentiellement dans l'espace objectif.

1.9.5.1 Première phase

L'objectif de la première phase est d'obtenir l'ensemble des solutions Pareto supportées.

Comme nous l'avons vu précédemment, ces solutions ont l'avantage d'être relativement faciles à trouver puisqu'elles optimisent une certaine combinaison linéaire des objectifs.

Ainsi, durant la première phase de la méthode, les deux solutions extrêmes (solutions optimisant chacune un des deux objectifs) sont recherchées (voir figure 1.5.a). Puis, de façon récursive, dès que deux solutions supportées x et y sont trouvées, la méthode recherche d'éventuelles autres solutions supportées entre x et y , à l'aide de combinaisons linéaires bien choisies des objectifs (voir figure 1.5.b et 1.5.c). A la fin de la première phase l'ensemble des solutions supportées est donc trouvé (voir figure 1.5.d).

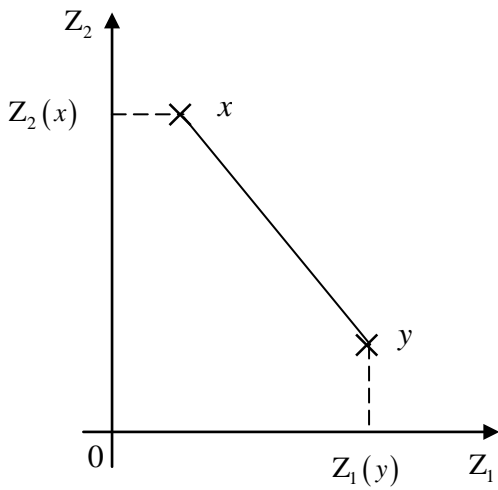
Cette première phase rappelle la méthode dichotomique, mais ici seules les solutions supportées sont recherchées. Pour cela, lors de l'exploration entre deux solutions, on s'autorise à retrouver l'une de ces deux solutions, lorsqu'il n'existe pas d'autres solutions supportées dans l'intervalle.

1.9.5.2 Deuxième phase

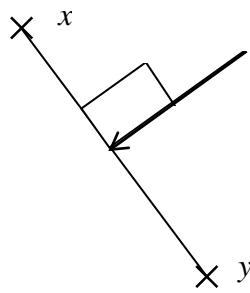
La deuxième phase consiste alors en la recherche des solutions non supportées appartenant au front Pareto. Ces solutions ne peuvent être obtenues par combinaisons d'objectifs. Ulungu et Teghem proposent alors d'utiliser les solutions supportées trouvées pour réduire l'espace de recherche en argumentant que les solutions Pareto non supportées restantes sont forcément dans les triangles rectangles basés sur deux solutions supportées consécutives

(voir figure 1.5.e). Ainsi, une recherche de type deuxième phase est exécutée entre chaque couple de solutions supportées adjacentes (voir figure 1.5.f et 1.5.g). La méthode de recherche au sein de ces triangles dépend du problème étudié. A la fin de la deuxième phase, toutes les solutions

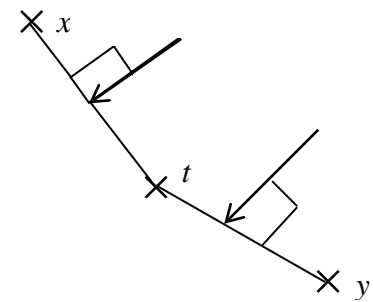
Pareto sont trouvées. Notons, qu'il aura été nécessaire au préalable de préciser si l'on recherche le front minimal ou maximal complet.



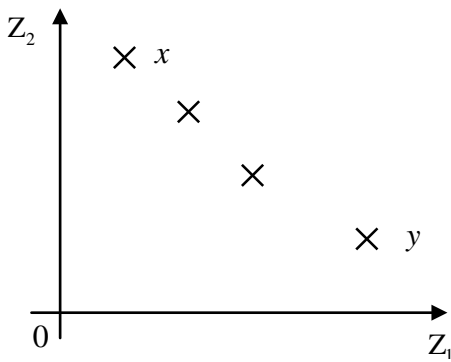
a)



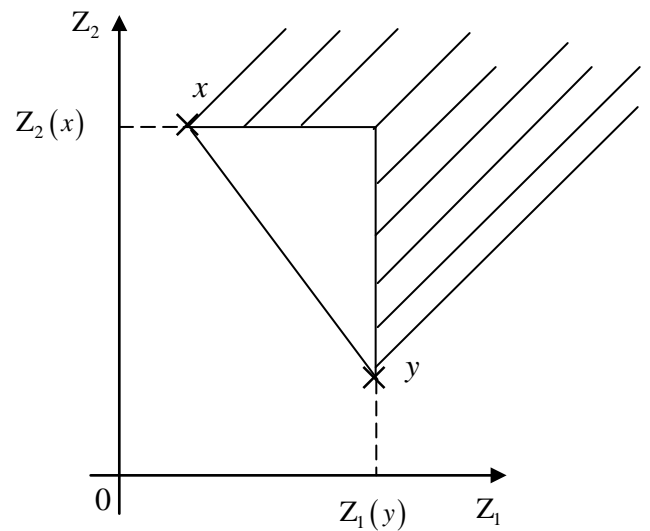
b)



c)



d)



e)

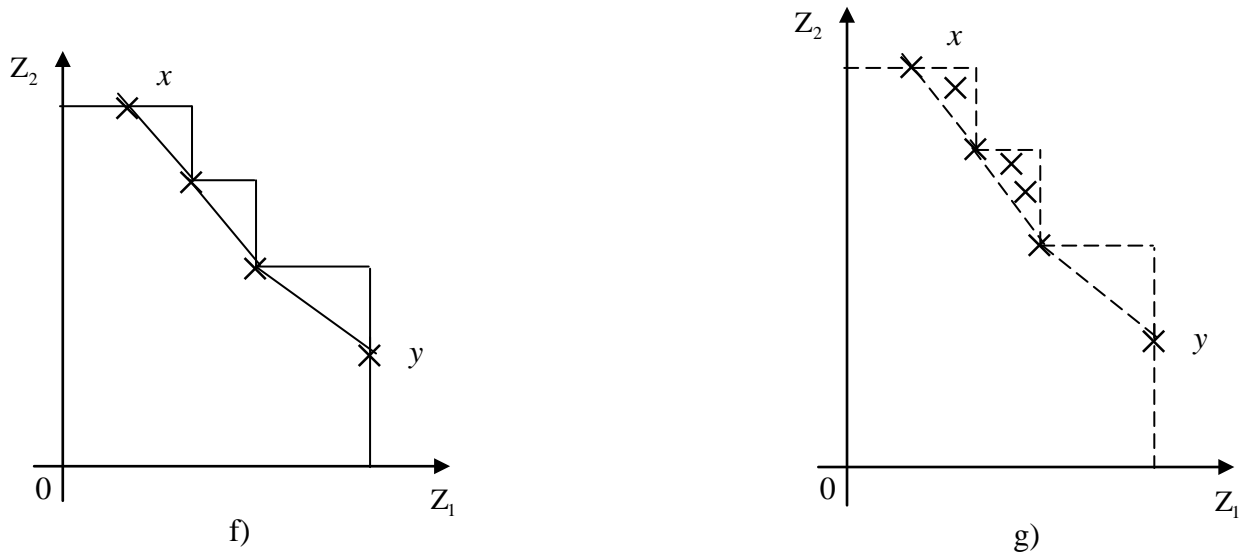


Fig 1.5 : Illustration de la méthode des deux phases

Théorème 3:(soland 1979) :

Un point $x \in S$ est une solution efficace pour le problème linéaire multicritère (1.7) si et seulement si x est une solution optimale du problème (1.15).

1.10 Quelques méthodes de résolution de MOILP :

Dans ce qui suit, on exposera quelques méthodes de résolution dans l'espace des critères et dans l'espace des décisions.

1.10.1 Méthode de résolution dans l'espace des critères :**1.10.1.1 Méthode de Klein et Hannan [35] :**

L'approche de Klein et Hannan consiste à résoudre une suite de programme linéaire en nombre entiers, en rajoutant progressivement des contraintes au problème MOILP. À chaque étape de l'algorithme de nouvelle contrainte sont rajoutées afin d'éliminer les solutions dominées par les solutions non dominées, ce qui permet de générer une suite de solutions efficaces classées par la valeur donnée à la fonction objective choisie. La méthode est résumée par l'algorithme ci-dessous :

L'algorithme :**Etape 1 : (initialisation)**

Choisir arbitrairement un critère $i (i \in \{1, 2, \dots, p\})$ et résoudre le problème uni-critère suivant :

$$P_{ro} \begin{cases} \max c^i x \\ t.p \ x \in D \end{cases} \quad (1.16)$$

Si la solution du problème (P_{ro}) est unique, alors elle est efficace et elle est l'unique élément dans la liste initiale des solutions efficaces $IE_0(P)$, sinon, soit $\xi(P_{ro})$ l'ensemble des solutions optimales de (P_{ro}) et $IE_0(P)$ l'ensemble des solutions efficaces correspondantes à $\xi(P_{ro})$. Etape $j (j \geq 1)$: on résout le problème P_{rj} définie par :

$$(P_{rj}) \begin{cases} \max Z_i = c^i x \\ t.q. \\ x \in D \\ \bigwedge_{k=1}^s \left(\bigvee_{s=1, s \neq i}^p (c^s x \geq c^s \tilde{x}_k + e^s) \right) \end{cases} \quad (1.17)$$

Où \vee est l'opérateur d'addition logique et \wedge est l'opérateur du produit logique.

$0 < e^s < 1$ (Pour $e^s < 1$, la méthode produit un sous-ensemble de l'ensemble des solutions efficaces).

1.10.1.2 Méthode de J. Sylva et A. Crema [49]

La méthode consiste de maximiser à chaque itération une combinaison positive de toutes les fonctions objective et non pas un seul critère, comme le cas de la méthode de Klein et Hannan. La solution optimale de chaque PLNE est une solution efficace »condition nécessaire de Geoffrion« (i.e. maximisation de la somme pondéré des critères sur l'ensemble réalisable en laissant les poids varies) En affectant des poids à ces fonctions objectives, représentant ainsi les préférences du décideur, alors l'algorithme générera que les solutions efficaces les plus intéressantes de chaque étape de processus de résolution.

Le problème MOILP considéré est :

$$(P) \begin{cases} \text{"max" } cx \\ t.q. \\ Ax = b, x \geq 0, x \in \mathbb{Z}^n \end{cases} \quad (1.18)$$

Avec $C \in \mathbb{Z}^{p \times n}$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ et l'ensemble des solutions réalisables est noté par $F(P)$.

Proposition 1 : [49]

Soit x^1, \dots, x^l des solutions efficaces pour le problème (P), posant $D_s = \{x \in \mathbb{Z}^n / cx \leq cx^s\}$ et soit x^* une solution efficace pour le problème MOILP suivant :

$$(P_1) \begin{cases} \max cx \\ tq. \\ x \in F(P) - \bigcup_{s=1}^l D_s \end{cases} \quad (1.19)$$

Alors x^* est une solution efficace du problème De plus si le problème (P_1) est réalisable, alors

$\{cx^*\}_{s=1}^l$ est l'ensemble de tous les vecteurs critères non dominés pour le problème (P) .

Corollaire 2 : Soit x^1, \dots, x^l des solutions efficaces pour le problème (P) ;

Posant $D_s = \{x \in \mathbb{Z}^n / cx \leq cx^s\}$ si x^* est une solution optimale du problème

$$(P_\lambda) = \max \left\{ \lambda^t Cx : x \in F(P) - \bigcup_{s=1}^l D_s \right\} \text{ alors pour certains } \lambda \in \mathbb{R}^p, \lambda > 0, x^* \text{ est une solution}$$

efficace pour le problème (P)

L'algorithme :

Etape 1 : Après avoir fixé le vecteur poids $\lambda > 0$, la première étape de l'algorithme consiste en la résolution du PLNE

$$(P_0) \begin{cases} \max \lambda^t Cx \\ tq Ax = b \\ x \geq 0, x \in \mathbb{Z}^n \end{cases} \quad (1.20)$$

Si ce problème n'est pas réalisable, alors le problème (P) n'est pas réalisable ; autrement une solution optimale x^1 est trouvée, et du corollaire précédent, x^1 est une solution efficace pour le problème (P) . Vient ensuite la résolution d'une succession de PLNE (P_L) en rajoutant à chaque fois certaines contraintes éliminant les solutions déjà trouvées précédemment. .

Après L étapes du processus de résolution, si le problème (P_{L-1}) n'est pas réalisable alors l'algorithme prend fin ; autrement, une nouvelle solution efficace x^L est trouvée et on définit un nouveau problème (P_L) en enlevant de l'ensemble des solutions réalisable de (P_{L-1}) toutes

les solutions qui vérifient $cx \leq cx^L$ ceci se traduit par le rajout des contraintes suivantes au problème (P_{L-1}) :

$$\begin{aligned} (Cx)_k &\geq \left((Cx^L)_k + 1 \right) Y_k^L - M_k (1 - Y_k^L), \text{ pour } k : 1, \dots, P \\ \sum_{L=1}^P Y_k^L &\geq 1 \quad Y_k^L \in \{0, 1\}, \text{ pour } k : 1, \dots, P \end{aligned}$$

Où $-M_k$ est la borne inférieure pour les valeurs réalisables de la $K^{\text{ième}}$ fonction objective.

Dans le cas où la matrice des critères c est positive, M_k peut être fixé à 0 pour tout k . Le

problème (P_L) s'écrit sous forme :

$$(P_L) \left\{ \begin{array}{l} \max \lambda' Cx \\ tq. \\ Ax = b \\ (Cx)_k \geq \left((Cx^S)_k + 1 \right) Y_k^S - M_k (1 - Y_k^S) \\ \text{pour } S : 1, \dots, L; k = 1, \dots, P \\ \sum_{k=1}^P Y_k^S \geq 1, Y_k^S \in \{0, 1\} \\ \text{pour } S : 1, \dots, L; k : 1, \dots, P \\ x \geq 0, x \in \mathbb{Z}^n \end{array} \right. \quad (1.21)$$

Pour les problèmes à grandes dimensions, le problème à résoudre est :

$$(P_L) \left\{ \begin{array}{l} \max \lambda' Cx \\ tq. \\ Ax = b \\ (Cx)_k \geq \left((Cx^S)_k + f_k \right) - M_k (1 - Y_k^S) \\ \text{pour } S : 1, \dots, L; k = 1, \dots, P \\ \sum_{k=1}^P Y_k^S \geq 1, Y_k^S \in \{0, 1\} \\ \text{pour } S : 1, \dots, L; k : 1, \dots, P \\ x \geq 0, x \in \mathbb{Z}^n \end{array} \right. \quad (1.21)$$

Où f_k est l'amélioration minimale de la $K^{\text{ième}}$ fonction objective.

1.10.2 Méthode de résolution dans l'espace des critères (espace des variables de décision) :

Introduction : Dans cette partie nous présentons deux méthodes de résolutions dans l'espace de décision, celle d'Abbas & Chaabane [2] et Abbas & Moulai [4]. Les deux méthodes sont des corrections de celle de Gupta & Malhotra [27] pour la recherche de toutes les solutions d'un problème MOILP. Les méthodes utilisent des coupes dans le processus de résolution. L'avantage des méthodes est d'éviter les problèmes liés aux solutions non supporté.

1.10.2.1 Notations et définitions :

Nous donnons quelques notations et définitions propres aux auteurs qui aident à comprendre leurs méthodes.

Nous proposons deux méthodes, une d'optimisation discrète linéaire multicritère et une de déterminations des solutions efficaces dans l'espace des variables discrètes voir [2,4].

Considérons le problème MOILP suivant :

$$(P) \begin{cases} \max Z_k(x) = c^k x, & k : 1, 2, \dots, p \\ \text{t.q.} \\ x \in S \\ x \text{ entier} \end{cases} \quad (1.22)$$

$$S = \{x \in \mathbb{R}^n / Ax \leq b, A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m, x \geq 0\}$$

Comme les critères sont conflictuels, donc on ne peut pas les optimiser simultanément. Le problème MOILP sera relaxé, en tenant compte d'un seul critère ;

$$(P_1) \begin{cases} \text{maximiser } Z_1(x) = c^1(x) \\ \text{t.q.} \\ x \in D \end{cases} \quad (1.23)$$

$D = S \cap \mathbb{Z}^n$, \mathbb{Z} est l'ensemble des entiers.

Notons qu'à la place de Z_1 , on pourra maximiser d'une manière analogue $Z_i, i \in \{2, \dots, p\}$.

Introduisons les notations suivantes :

Pour $k \geq 1$

$$S_k = \{x \in \mathbb{R}^{n_k} / A_k x \leq b_k, A_k \in \mathbb{R}^{n_k \times m_k}, b_k \in \mathbb{R}^{m_k}, x \geq 0\}$$

S_k est la région courante tronquée de S par application de la coupe $\sum_{j \in N_{k-1} - \{j_{k-1}\}} x_j \geq 1$ ou $j_{k-1} \in J_{k-1}$ et éventuellement par des coupes de Gomory $x_k^1 = \{x_{k,j}^1\}$ est la $K^{ième}$ solution optimale entière du problème (P_1) obtenue sur S_k à l'étape k .

$x_k^a = \{x_{k,j}^a\}$ Sont les $(P_k - 1)$ solutions entières, lorsqu'elles existent alternatives à x_k^1 où P_k est un nombre entier et $q \in \{2, 3, \dots, P_k\}$.

B_k^1 est une base de S_k .

$$a_{k,j}^1 = (Y_{k,ij}^1) = (B_k^1)^{-1} a_{k,j}^1 \in \mathbb{R}^{m_k \times 1}$$

$$I_k = \{j : a_{k,j}^1 \in B_k^1\}$$

$$N_k = \{j : a_{k,j}^1 \notin B_k^1\}$$

$$Z_{k,1}^1 = \sum_{i \in I_k} C_i^1 Y_{k,ij}^1$$

$J_k = \{j : j \in N_k \text{ et } Z_{k,j}^1 - C_j^1 = 0\}$ Où $Z_{k,j}^1 = C_{B_k^1}^1 Y_{k,j}^1$, C_j^1 est la $j^{ième}$ composante du vecteur c_1 et

$c_{B_k^1}^1$ est le vecteur coût des variables de base associées à B_k^1 dans le vecteur c^1 .

Ainsi, J_k est l'ensemble correspondant à la solution x_k^1 .

$\Gamma_k = \{j \in N_k / Z_{k,j}^1 - C_j^1 > 0 \text{ et } Z_{k,j}^1 - C_j^1 < 0\}$. Pour au moins un $i \in \{2, \dots, P\}$

Remarque 1 : Si l'ensemble J_k correspondant à la solution x_k^1 du problème (P_1) est vide, alors x_k^1 est l'unique $K^{ième}$ meilleure solution réalisable du problème (P_1) .

Définition 2 :

Une arête E_{J_k} incidente à la solution x_k^1 est défini comme étant l'ensemble :

$$E_{J_k} = \left\{ x = (x_i) \in S / \begin{array}{l} x_i = x_{k,i}^1 - \phi_{jk} y_{k,ij}^1, i \in I_k \\ x_{j_k} = \phi_{j_k} \\ S_k = 0 \text{ pour tous } S \in N_k - \{j_k\} \end{array} \right.$$

Où $0 \leq \phi_{j_k} \leq \min_{i \in I_k} \left\{ \frac{x_{k,i}^1}{y_{k,ij}^1}; y_{k,ij}^1 \right\}$, ϕ_{j_k} est un entier positif et $\phi_{jk} y_{k,ij}^1$ sont des entiers pour tous

$i \in I_k$ si de telles valeurs existent.

Définition 3 :

Une arête E_{j_k} incidente à un point entier réalisable x_k^1 est dite dominée si toutes les solutions se trouvant sur cette arête donnent des p-uplets (Z_1, Z_2, \dots, Z_p) dominés.

1.10.2.2 Résultats théoriques :

Théorème 4: Toutes les solutions entières réalisables $x_k^a, a \in \{2, \dots, P_k\}$ du problème (P_1) alternative à la solution x_k^1 sur l'arête E_{j_k} de la région S (ou la région tronquée S_k) émanent d'elle dans la direction d'un vecteur $a_{k,j_k}^1, j \in J_k$, existe dans le demi espace ouvert $\sum_{j \in N_k - \{j_k\}} x_j < 1$.

Preuve : Puisque x_k^1 est une solution réalisable du problème (P_1) alors $Ax_k^1 \leq b$, où $x_k^1 = (x_{k,j}^1)$.

$$Ax_k^1 = \sum a_{k,j}^1 - x_{k,j}^1 \leq b$$

Pour un certain $j_k \in J_k$

$$\sum_{i \in I_k} a_{k,i}^1 - \phi_{j_k} a_{k,j_k}^1 + \phi_{j_k} a_{k,j_k}^1 \leq b, \text{ Où } \phi_{j_k} \text{ est un scalaire positif non nul}$$

Puisque $a_{k,j_k}^1 = \sum_{i \in I_k} a_{k,i}^1 y_{k,ij}^1$, donc

$$\sum_{i \in I_k} a_{k,i}^1 - \phi_{j_k} \left(\sum_{i \in I_k} a_{k,i}^1 y_{k,ij}^1 \right) + \phi_{j_k} a_{k,j_k}^1 \leq b \Rightarrow \sum_{i \in I_k} a_{k,i}^1 (x_{k,i}^1 - \phi_{j_k} y_{k,ij}^1) + \phi_{j_k} a_{k,j_k}^1 \leq b$$

Pour $0 \leq \phi_{j_k} \leq \min_{i \in I_k} \left\{ \frac{x_{k,i}^1}{y_{k,ij}^1}; y_{k,ij}^1 \right\}$, x_k^a est défini comme :

$$x_k^a = \begin{cases} x_{k,i}^a = x_{k,i}^1 - \phi_{j_k} y_{k,ij}^1, i \in I_k \\ x_{k,j_k}^a = \phi_{j_k} \\ x_{k,S}^a = 0 \text{ pour tous } S \in N_k - \{j_k\} \end{cases}$$

Une nouvelle solution entière réalisable du problème (P_1) pourvu que ϕ_{j_k} soit entière positive et $\phi_{j_k} y_{k,ij}^1$ soient entières pour tous $i \in I_k$.

$$\begin{aligned} Z_1(x_k^a) &= c^1 x_k^a = \sum_{i \in I_1} c_i^1 x_{k,i}^a + c_{j_k}^1 x_{k,j_k}^a + \sum_{\alpha \in N_1 - \{j_k\}} c_S^1 x_{k,S}^a \\ &= \sum_{i \in I_1} c_i^1 (x_{k,i}^1 - \phi_{j_k} y_{k,ij}^1) + c_{j_k}^1 x_{k,j_k}^1 \end{aligned}$$

$$\begin{aligned}
 &= \sum_{i \in I_1} c_i^1 x_{k,i}^1 - \phi_{j_k} \sum_{i \in I_1} c_i^1 y_{k,ij_k}^1 + C_{j_k}^1 \phi_{j_k} \\
 &= \sum_{i \in I_1} c_i^1 x_{k,i}^1 - \phi_{j_k} \left(\sum_{i \in I_1} c_i^1 y_{k,ij_k}^1 - c_{j_k}^1 \right) \\
 &= \sum_{i \in I_k} c_i^1 x_{k,i}^1 - \phi_{j_k} \left(z_{k,ij_k}^1 - c_{j_k}^1 \right) \\
 &= Z_1(x_k^1) - \phi_{j_k} \left(Z_{k,ij_k}^1 - c_{j_k}^1 \right)
 \end{aligned}$$

Comme $j_k \in \Gamma_k$, alors $Z_{k,ij_k}^1 - c_{j_k}^1 = 0$ donc $Z_1(x_k^q) = Z_1(x_k^1)$

Par conséquent, x_k^q est une solution réalisable entière du problème (P_1) alternative à x_k^1 sur une arête E_{j_k} de la région tronquée S_k et émanent d'elle, puisque $x_{k,j}^q = 0$ pour tout $j \in N_k - \{j_k\}$, on peut voir que $\sum_{j \in N_k - \{j_k\}} x_j \geq 1$, donc le point x_k^q appartient au demi espace ouvert $\sum_{j \in N_k - \{j_k\}} x_j < 1$.

Théorème 5 :

Une solution réalisable entière du problème (P_1) ne se trouvant pas sur l'arête E_{j_k} , $j_k \in \Gamma_k$ de la région tronquée S_k (ou la région S) à travers un point réalisable entier x_k^1 du problème (P_1) appartient au demi espace fermé $\sum_{j \in N_k - \{j_k\}} x_j \geq 1$ (2).

Preuve :

Soit $(\tilde{x}) = (\tilde{x}_{ij})$ une solution réalisable entière du problème (P_1) ne satisfaisant pas (2) (c'est-à-dire ne se trouvant pas sur l'arête E_{j_k}), alors $\sum_{j \in N_k - \{j_k\}} \tilde{x}_j \geq 1$, (\tilde{x}_j) entier, ceci implique que $\tilde{x}_j = 0$ pour tout $j \in N_k - \{j_k\}$. Si des solutions alternatives à \tilde{x} existent alors $\tilde{x}_{j_k} > 0$; si non $\tilde{x}_{j_k} = 0$.

- a) si $\tilde{x}_{j_k} = 0$, alors l'ensemble des indices hors base dans le tableau optimal correspondant à \tilde{x} est le même que N_k comme l'ensemble $N_k = [N_k - \{j_k\}] \cup \{j_k\}$, ceci implique que les indices des variables de base dans le tableau optimal correspondant à \tilde{x} et B_k^1

sont les mêmes, ce qui implique de plus que $\tilde{x} = x_k^1$, et donc \tilde{x} est sur l'arête E_{J_k} , ce qui contredit l'hypothèse. en conséquence $\tilde{x}_j > 0$ pour au moins $j \in N_k - \{j_k\}$.

b) Si $\tilde{x}_{j_k} > 0$, alors \tilde{x} doit se trouver dans la direction du vecteur $a_{k,j_k}^1, j_k \in J_k$.

Si $\tilde{x}_{j_k} > \min_{i \in I_k} \left\{ \frac{\tilde{x}_{k,i}}{y_{k,ij_k}} / y_{k,ij_k} > 0 \right\} = \frac{x_{k,q}}{y_{k,qk}}$, alors la composante $\tilde{x}_{k,q} = x_{k,q} - \tilde{x}_{j_k} Y_{k,qi} < 0$

implique que la \tilde{x} n'est pas réalisable, ce qui est faux.

Si $\tilde{x}_{j_k} < \min_{i \in I_k} \left\{ \frac{x_{k,i}}{y_{k,ij_k}} / y_{k,ij_k} > 0 \right\}$ alors $\tilde{x}_{j_k} < \frac{x_{k,i}}{y_{k,ij_k}} \forall i \in I_k$ ce qui implique que \tilde{x} est sur

l'arête E_{J_k} . D'où une contradiction avec l'hypothèse et par conséquent $\tilde{x}_j > 0$ pour au moins

un $j \in N_k - \{j_k\}$. Ainsi, dans les deux cas, on a prouvé que pour au moins un

indice $j \in N_k - \{j_k\}$, $\tilde{x}_j > 0$ et par suite $\tilde{x}_j \geq 0$, implique que \tilde{x} appartient au demi-espace

fermé $\sum_{j \in N_k - \{j_k\}} x_j \geq 1$.

1.10.3 Méthode de M. Abbas & M. Moulai [2] :

C'est une méthode exacte appropriée pour résoudre le problème de programmation linéaire à objectifs multiples.

La première itération consiste à déterminer une solution entière réalisable optimale d'un problème à une seule fonction objective sous les contraintes du problème (MOILP).

La deuxième itération permet de trouver les solutions alternatives de la solution entière optimale obtenue à la première itération, si elle existe et déduire les premières solutions entières efficaces du problème principales.

Par la suite il introduit une coupe plane pour dénombrer toutes les autres solutions efficaces restantes en utilisant la méthode duale simplexe.

A la fin, l'algorithme aura exploré toutes les solutions efficaces du problème étudié.

Nous donnons par la suite quelques notations et définitions qui ont été déjà données pour définir les étapes de l'algorithme.

Etape 1 :

Résoudre le problème (P_1) et trouvé la solution optimale entière x_1^1 sur S_1 , construire l'ensemble J_1 .

Etape 2 :

Si $J_1 = \emptyset$, x_1^1 est l'unique solution optimale sur S_1 et Z_1^1 est la valeur optimale de Z_1 . Calculer les valeurs Z_i^1 de Z_i données par $x_i^1, i \in \{2, \dots, p\}$. Enregistrer le premier P-uplet efficace $(Z_1^1, Z_2^1, \dots, Z_p^1)$ pour construire l'ensemble des points efficaces Eff_0 . Tronquer le point x_1^1 par la coupe $\sum_{j \in N_1} x_j \geq 1$ appliquer la méthode du dual simplexe et éventuellement des coupes de Gomory si nécessaire pour avoir une solution réalisable entière $x_2^1 = (x_{2,j}^1)$ dans la région tronquée S_2 . Lire le P-uplet correspondant (Z_1, Z_2, \dots, Z_p) et rajouter le a Eff_0 s'il n'est pas dominé par l'un des P-uplet efficaces précédents. Par conséquent Eff_0 devient Eff_1

1- Si $J_1 \neq \emptyset$, choisir un indice quelconque $j_1 \in J_1$ et calculer le nombre

$$\phi_{j_1} \leq \min_{i \in I_1} \left\{ \frac{x_{1,i}^1}{y_{1,i j_1}} > 0 \right\} \text{ correspond à la solution } x_1^1.$$

a) Si $\phi_{j_1} \geq 1$, déterminer toutes les solutions entières réalisables

$$x_k^q, q \in \{2, \dots, P_k\} \text{ Alternative à } x_1^1$$

Le long de l'arête E_{J_k} . Chaque solution donne naissance à un P-uplet potentiellement efficace de la forme (Z_1, Z_2, \dots, Z_p) , puisque les solutions alternative ont la même valeur que celle de x_1^1 (i.e. Z_1), le premier point efficace est choisi comme le P-uplet ayant la plus grande pour $Z_2(x_1^q)$, si non choisir $\max Z_2(x_1^q)$ et ainsi de suite jusqu'à l'obtention du premier point efficace Eff_0 est l'ensemble des P-uplet non dominés potentielles à l'étape 2. Tronquer le domaine réalisable par l'arête E_{J_1} par la coupe $\sum_{j \in N_1} x_j \geq 1$.

La méthode duale du simplexe et éventuellement des coupes de Gomory, permet d'obtenir une solution réalisable entière $x_2^1 = (x_{2,j}^1)$ dans la région tronquée S_2 .

Lire le P-uplet (Z_1, Z_2, \dots, Z_p) correspondant et le rajouter à l'ensemble Eff_0 s'il n'existe pas dominé par l'un des P-uplet efficaces précédemment déterminés, Eff_0 devient Eff_1 .

b) si $\phi_{j_1} < 1$ pour tous les indices $j_1 \in J_1$, choisir un indice quelconque j_1 et appliquer la

$$\text{coupe } \sum_{j \in N_1} x_j \geq 1.$$

En appliquant la méthode dual simplexe et des coupes successives de Gomory, on obtient une solution réalisable entière $x_2^1 = (x_{2,j}^1)$ dans la région tronquée S_2 lire le p-uplets (Z_1, Z_2, \dots, Z_p) correspondant et le rajouter à l'ensemble Eff_0 s'il n'est pas dominé par l'une des P-uplet efficaces précédent, Eff_0 devient Eff_1 .

Etape 3 :

Choisir un indice $j_2 \in \Gamma_2$ et déterminer toutes les solutions entières réalisables $x_2^q, q \in \{2, \dots, P_2\}$, alternatives à x_2^1 , lorsqu'elles existent, sur l'arête E_{j_2} . Lire les P-uplet obtenus.

Augmenter l'ensemble Eff_1 par les nouveaux P-uplet non dominé pour construire Eff_2 .

Donc Eff_2 est l'ensemble de tous les potentiels P-uplet efficaces à l'étape 3.

Couper l'arête E_{j_2} par l'hyperplan $\sum_{j \in N_2 - \{j_2\}} x_j \geq 1$ et chercher la solution optimale entière dans la région tronquée courante pour entrer dans l'étape 4 de la procédure.

Etape k :

Choisir un indice $j_{k-1} \in \Gamma_{k-1}$ et explorer l'arête correspondante $E_{j_{k-1}}$ pour des possibles solutions entières réalisables $x_{k-1}^q, q \in \{2, \dots, P_k\}$ alternatives à x_{k-1}^1 . Augmenter l'ensemble Eff_{k-2} par les nouveaux P-uplet non dominés correspondants pour construire Eff_{k-1} tronquer l'arête $E_{j_{k-1}}$ par la coupe $\sum_{j \in N_{k-1} - \{j_{k-1}\}} x_j \geq 1$

Après l'application de la méthode duale du simplexe et éventuellement des coupes successives de Gomory, la solution optimale entière obtenue sur la région tronquée S_k sera soit une solution réalisable de (P_1) alternative à x_{k-1}^1 ou la prochaine meilleure solution x_k^1 au un point non entier ceci marque le début de l'étape $k + 1$

Etape finale :

Le processus se termine lorsque l'impossibilité des opérations pivots de la méthode duale du simplexe apparaît, indiquant que la région courante ne contient aucun point réalisable entier et que l'ensemble des points efficaces est complètement déterminé.

1.10.4 Méthode de M.Abbas & D.Chaabane :

Cette méthode détermine toutes les solutions efficaces du problème MOILP sans laisser aucune .La première étape consiste à résoudre le problème (p_1) et ainsi avoir déterminé une solution optimale initiale ; puis une séquence s de coupes $\sum_{j \in N_k - \{j_k\}} x_{j_k} \geq 1$ et de Gomory sont appliquer après avoir explorer l'arête E_{j_k} . La méthode est une forme modifier de celle de Gupta & Malhotra, le test d'arrêt est modifier, ainsi on peut avoir toutes les solutions efficaces de la région d'admissibilité.

1.10.4.1 L'algorithme de la méthode :

Etape 1 : Résoudre le problème (P_1) . On trouve une solution x_1^* dont les évaluations sont $(z_1^1, z_2^1, \dots, z_p^1)$, .Au lieu de (P_1) , on peut résoudre un des problèmes $(P_i; i = 2, 3, \dots, p)$ qui maximise $z_i = c^i x$.

- Si $J_1 = \{z_j^1 - c_j^1 = 0\} = \emptyset$, alors la solution optimale est unique, enregistrer le premier vecteur non dominé $(z_1^1, z_2^1, \dots, z_p^1)$, et former la liste initiale des solutions non dominées

$opt_1 = \{z_1^1, z_2^1, \dots, z_p^1\}$.Aller à l'étape 2.

- Si $J_1 = \{j \in N_1 / z_j^1 - c_j^1 = 0\} \neq \emptyset$, la solution optimale du problème (IPL_1) peut ne pas

être unique .Pour chaque $j \in J_1$ calculer. $\phi_j = \min_{i \in I_k} \left[\left\{ \frac{x_{1,i}}{y_{1,ij}} / y_{1,ij} \succ 0 \right\} \right]$

- (a) Si pour tout $j \in J_1$ on a $\phi_j < 1$, il n'y a pas de solutions alternatives à la solution x_1 ;

Initialiser la liste des vecteurs potentiellement non dominés $opt_1 = \{(z_1^1, z_2^1, \dots, z_p^1)\}$ et aller à l'étape 2.

- (b) Sinon, tant qu'il existe au moins un $j \in J_1$ tel que $\phi_j \geq 1$ faire :

- Explorer l'arête

$$E_j = \left\{ \begin{array}{l} x'_{1i} = x_{1,i} - \theta \times y_{1,ij} \text{ pour } i \in I_k \\ (x'_i) \in \mathbb{R}^{(|I_1|+|N_1|)} / x'_{1j} = \theta \\ x'_{1,\alpha} = 0, \text{ pour tous } \alpha \in \mathbb{N}_1 - \{j\} \end{array} \right\}$$

Pour θ entier variant entre 1 et ϕ_j , $\theta \times y_{1,j}$ entiers.

- Sur chacune des solutions réalisables trouvées sur une arête, évaluer les critères, et ajouter à la liste des vecteurs critères, et ajouter à la liste les vecteurs critères non dominés (une comparaison deux à deux est effectuée).
- Choisir arbitrairement un $j_1 \in J_1$ et aller à l'étape 2.2 où l'on réduira le domaine à l'aide d'une coupe de type (I) $\sum_{j \in N_k - \{j_k\}} x_j \geq 1$.on peut alternativement choisir j_1 tel que ϕ_{j_1} Soit maximal).

Etape2 : Soit $k = 1$;

2.1 Construire l'ensemble $\Gamma_k = \{j \in N_k / Z_J^1 - C_J^1 > 0 \text{ et } \exists i \in \{2, 3, \dots, p\} \text{ telle que } z_j^i - c_j^i > 0\}$

2.1.1 Si $\Gamma_k = \emptyset$, aller à l'étape 2.2 (la coupe devient une coupe de Dantzig

$$\sum_{j \in N_k} x_j \geq 1.$$

2.1.2 Sinon, soit $\gamma = \Gamma_k$, et aller à (a).

(a) Si $\gamma = \emptyset$, choisir un $j_k \in \Gamma_k$ et aller à l'étape 2.2. si non, soit $j_k \in \gamma$; calculer

$$\theta_{j_k}^0 \text{ la partie entière de } \theta_{k, j_k} = \min_{i \in I_k} \left\{ \frac{x_{k,i}}{y_{k,ij}} ; y_{k,ij} \succ 0 \right\}.$$

- Si $\theta_{j_k}^0 = 0$, il n'y a aucune solution réalisable sur l'arête E_{j_k} , faire $\gamma := \gamma - \{j_k\}$ Aller à (a).
- Sinon aller à 2.1.3.

2.1.3 pour $\theta = 1, 2, \dots, \theta_{j_k}^0$; calculer toutes les solutions réalisables entières sur l'arête E_{j_k} en utilisant

$$\begin{cases} x_i = x_{k,i} - \theta \times y_{k,ij} & \text{pour tous } i \in I_k \\ x_{j_k} = \theta \\ x_\alpha = 0 & \text{pour tous } \alpha \in N_k - \{j_k\} \end{cases}$$

Evaluer les critères sur chaque solution, éliminer les vecteurs critères dominés et introduire les nouveaux vecteurs critères potentiellement non dominés (non dominés par rapport aux solutions sur la liste à l'itération k) dans la liste opt_k . choisir un critère j_k de Γ_k et aller à l'étape 2.2.

2.2 Ajouter la contrainte $\sum_{j \in N_k} x_j \geq 1$ et appliquer la méthode du dual simplexe et Gomory si nécessaire. Soit x_{k+1} une solution optimale du problème augmenté. Évaluer tous les critères en cette solution. Si le vecteur critère correspondant est dominé par des éléments de la liste opt_k , ignorer. Si non, l'ajouter à la liste des vecteurs non dominés pour produire la liste opt_{k+1} . Faire $k := k + 1$ et aller à 2.1.

Étape 3 : la procédure prend fin quand l'opération est impossible, le problème est devenu irréalisable dans la nouvelle région tronquée. L'algorithme se termine et la liste finale opt_{k+1} représente l'ensemble de toutes les solutions non dominées.

1.11 Conclusion sur l'optimisation multi-objective

Ce chapitre avait pour objectif de présenter dans un premier temps les principales définitions nécessaires à la présentation des problèmes d'optimisation combinatoire multi-objectifs. Puis différentes problématiques liées aux spécificités du multi-objectif, comme l'intervention du décideur dans le processus de décision, les choix des méthodes d'optimisation à utiliser ont été évoqués afin de montrer l'étendue du spectre des recherches dans le domaine.

Chapitre II

Optimisation multi-objective dans un environnement incertain

2 Introduction :

L'avenir par nature est une source d'incertitude ce concept est central en matière d'activité humaine, et la prise de décision n'y fait pas exception. Lorsqu'on utilise la programmation linéaire ou entière pour prendre de décision, il faut analyser les données du problème réel de manière à identifier les paramètres qui doivent être pris en compte dans le modèle associé. Dans le cas déterministe, tous les facteurs intervenant dans le problème, en dehors des variables de décision à notre disposition, sont supposés connus avec précision, mais qu'arriverait-il si un ou plusieurs paramètres ne sont pas connus avec certitude ?

Ces questions imposent qu'il faut prendre en compte les effets induits par l'incertitude (événement non observé, erreurs de mesure...) dans les modèles d'optimisation.

Dans beaucoup de problèmes réels, on peut modéliser l'incertitude sur un ou plusieurs paramètres par une distribution de probabilité, par essence chaque paramètre incertain est représenté par une variable aléatoire, ceci quantifie l'incertitude. Une manière courante de modéliser l'incertitude dans les programmes mathématiques est via la programmation stochastique à deux niveaux, où une première décision doit être prise a priori avant la réalisation de l'incertitude et une décision de recours sera prise une fois l'incertitude soit révélée. Le but est de minimiser l'espérance du coût total associé aux deux décisions. Ce qui suppose implicitement qu'on accepte un comportement « optimal en moyenne » le lecteur intéressé pourra se référer à Kall et Wallace (1994), Prékopa (1995), Birge et Louveaux (1997). Le domaine de la programmation stochastique s'est développé rapidement en contribution avec plusieurs disciplines, telle que la recherche opérationnelle, les probabilités, les statistiques et l'économie.

La programmation stochastique désigne de manière générale les modèles mathématiques qui prennent explicitement en compte l'incertitude sous forme de distribution probabiliste de certains paramètres, ces modèles peuvent être classés plus finement en fonctions de la manière dont cette incertitude est exprimée et traitée dans le modèle de programmation linéaire correspondant.

Considérons le problème stochastique suivant

$$\left\{ \begin{array}{l} \text{"min"}_x Z(\omega) = C^t(\omega)x \\ \text{t.q.} \\ T(\omega)x = h(\omega) \\ x \in S \end{array} \right. \quad (2.1) \quad \begin{array}{l} S = \{x \in \mathbb{R}^n / Ax = b, x \geq 0\} \\ \text{Est un polyèdre} \end{array}$$

convexe. La matrice A et le vecteur b sont connus avec certitude de dimension respective $(m \times n), (m \times 1)$.

$C(\omega)$, $T(\omega)$ et le vecteur de droite $h(\omega)$ sont aléatoire de dimension respective $(1 \times n), (m_1 \times n), (m_1 \times 1)$, défini sur un espace de probabilité (Ω, Ξ, P) .

Le symbole " " signifie qu'il s'agit d'une optimisation imprécise (au sens de Pareto), donc "min " ne fourni que l'indication au sens de l'optimisation.

Définition 4 [1] :

$x \in S$ est efficace avec probabilité 1, s'il n'existe aucun $x^* \in S$ tel que $P\{\omega/c^t(\omega)x^* \leq c^t(\omega)x\} = 1$ et $P\{\omega/c^t(\omega)x^* < c^t(\omega)x\} > 0$. En d'autres termes, un point x de S est efficace avec probabilité 1, s'il n'existe aucune autre point x^* de S presque sûrement aussi bon que x , et qui soit meilleur que x avec une probabilité positive.

2.1 Approches de la programmation linéaire stochastique :

2.1.1 Approche « Wait and see »

L'approche « Wait and see » (attendre et voir) suppose que le décideur à la possibilité d'attendre jusqu'à ce que l'incertitude soit levée avant de mettre en œuvre les décisions optimales. Cette approche s'appuie donc sur la disponibilité d'une information parfaite au sujet du futur. Du fait de ces hypothèses une telle solution ne peut être mise en œuvre et est connue sous le nom d'approche passive.

Les modèles attendre et voir sont souvent utilisés pour analyser la distribution probabiliste de la valeur objective et consistent en une famille de modèles de programmation linéaire qui sont associés chacun à un scénario unique.

2.1.2 Approche « Here and now »

Par opposition cette approche est basée sur le principe de prise de décision sans connaître en préalable la réalisation de l'aléatoire $\omega \in \Omega$, et connue sous le nom d'approche active.

Résoudre le problème (1) revient à résoudre un problème déterministe équivalent dont la solution optimale sera considérée comme la solution optimale du problème initiale.

2.2 Critère d'optimisation du problème équivalent :

Plusieurs façons de définir la fonction objective du problème équivalent peuvent être considérés, une interprétation naturelle est de minimiser l'espérance mathématique

$$E(Z(\omega)) \text{ de l'objective } Z(w). \text{ Soit } \min_x E(Z(\omega)) = E(c'(\omega)x) \quad (2.2).$$

Ce qui nous permettons d'adopter le critère de l'espérance mathématique où critère de bayes.

2.2.1 Contrainte du problème équivalent :

Nous supposons que l'objectif est déterministe où qu'il a été rendu déterministe en appliquant critère le précédent. La première façon qui vient à l'esprit pour définir les contraintes du problème équivalent, consiste à remplacer les coefficients aléatoires par leurs espérances mathématiques. L'exemple suivant montre que cette façon de procéder peut ne pas donner de résultat satisfaisant

Soit :

$$\begin{aligned} \min x_1 + x_2 \\ t_{11}x_1 + x_2 &\geq 3 \\ t_{21}x_1 + 2x_2 &\geq 3 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Avec t_{11} , t_{21} deux variables aléatoires discrète de distributions:

t_{11}	1	2	3
$P(t_{11}=t)$	1/3	1/3	1/3

t_{21}	0	1	2
$P(t_{21}=t)$	1/3	1/3	1/3

$$E(t_{11}) = 6/3 \quad \text{et} \quad E(t_{21}) = 3/3$$

Le problème équivalent serait :

$$\begin{aligned} \min x_1 + x_2 \\ 2x_1 + x_2 &\geq 3 \\ x_1 + x_2 &\geq 3 \\ x_1, x_2 &\geq 0 \end{aligned}$$

La solution optimale de ce problème est $x = (1,1)$, cette solution est une solution admissible du problème initial avec la probabilité $P(t_{11} \geq 2, t_{21} \geq 1) = 0,66$. Ainsi la solution optimale obtenue ne satisfait les contraintes que dans 66% des cas. En programmation linéaire stochastique, les contraintes stochastiques seront traitées par deux modèles :

a) Modèle avec recours :

a.1 Recours générales :

Une classe importante de modèles de programmation stochastique est connue par les modèles avec recours. La programmation stochastique avec recours rend explicite la nature dynamique de la programmation stochastique on sépare les variables de décisions du modèle en décision stratégique x , qui doivent être prise lors de la première étape en absence de certitude pour le futur et en action de recours $y(\omega)$ sont obtenues en résolvant le problème du deuxième niveau :

$$\left\{ \begin{array}{l} Q(x, \omega) = \min_y q^i(\omega) y \\ t.q. \\ Wy = h(\omega) - T(\omega)x \\ y \geq 0 \end{array} \right. \quad (2.3)$$

On peut l'expliquer que si la solution x trouvée une fois résout le problème en question sans les contraintes stochastiques viole les contraintes stochastiques, une décision corrective $y(\omega)$ appelée recours sera prise pour compenser l'effet de ses violations. Cette compensation se fait par l'introduction d'une fonction de pénalité, et la minimisation de cette pénalité correspond au problème (2.3).

Où q est un vecteur $(1 \times n)$ de pénalisation de recours. $W(\omega)$ Est une matrice $(m_1 \times n)$ de recours. La formulation de problème de programmation stochastique à deux niveaux avec recours est la suivante :

$$\left\{ \begin{array}{l} \min_x Ec(\omega)x + Q(x) \\ t.q. \\ Ax = b \\ W(\omega)y = h(\omega) - T(\omega)x \\ x \geq 0, y \geq 0 \end{array} \right. \quad (2.4)$$

Avec $Q(x) = E(Q(x, \omega)) = \min_y E q^i(\omega) y$

Remarques 2: le nombre de réalisation de variable aléatoire définit un ensemble de scénarios qui croît exponentiellement avec la dimension de ω . Par exemple si nous avons n composantes aléatoires de ω , chacune a m réalisations possibles, alors le nombre total des scénarios est $S = m^n$.

Notons par K_1 ensemble de faisabilité de la première étape

$$K_1 = \{x \in \mathbb{R}_+^n / Ax = b\}.$$

-Notons par K_2 ensemble de faisabilité de la seconde étape $K_2 = \{x / Q(x) < \infty\}$

Par conséquent nous pouvons réécrire le problème (2.4) comme suit :

$$\min_x \{Ec^t(\omega) + Q(x) / x \in K_1 \cap K_2\}$$

On a :

$$Q(x, \omega) = \min_y \{q(\omega)^t y / Wy = h(\omega) - T(\omega)x, y \geq 0\}.$$

En fait, on devrait écrire $Q(x, \omega) = \inf_y \{q(\omega)^t y / Wy = h(\omega) - T(\omega)x, y \geq 0\}$, car nous n'avons aucune garantie que la fonction $Q(x, \omega)$ soit borné inférieurement sur un ensemble réalisable (définie par rapport à y). On peut donc avoir $Q(x, \omega) = -\infty$.

-Si $Q(x, \omega)$ n'admet aucun point réalisable, on notera $Q(x, \omega) = +\infty$.

a.2 Recours fixe : Fixons $q = q(\omega)$, $W = W(\omega)$, le problème (2) se réduit au problème de programmation avec recours fixe. La première formulation du problème avec recours fixe est donnée par Beale [5] et Dantzig [16] en 1955.

a.3 Recours complet : définissons l'ensemble C par :

$C = \{y / Wy = z, y \geq 0\}$ Nous avons que $Wy = z, y \geq 0$ est réalisable si et seulement si $y \in C$.

Le recours est complet si $\{y / Wy = z, y \geq 0\} = \mathbb{R}^{m_0}$, ceci implique $h(\omega) - T(\omega)x \in C$ pour tous $\omega \in \Omega$ et tous $x \geq 0 / Ax = b$.

Proposition 2 :

L'ensemble C défini par $C = \{y / Wy = z, y \geq 0\}$ est un cône convexe polyédrique

a.4 Recours simple : Le recours simple correspond au cas où la matrice recours est de dimension $m \times 2m$ est égale à $(I, -I)$, où I est la matrice unité. Dans ce cas, le vecteur y est décomposé en deux parties :

$y^+ (m \times 1)$ Variables d'écart par défaut.

$y^- (m \times 1)$ Variables d'écart par excès.

Le vecteur de pénalisation s'écrit $q(\omega) = (q^+(\omega), q^-(\omega))$, avec :

$$q^+(\omega) [h(\omega) - T(\omega)] \quad \text{Si } h(\omega) - T(\omega) \geq 0$$

$$q^-(\omega) [h(\omega) - T(\omega)] \quad \text{Si } h(\omega) - T(\omega) \leq 0$$

Dans ce cas le problème (5) s'écrit :

$$\left\{ \begin{array}{l} \min_x E(C(\omega))x + E\left(\min_{y^+, y^-} q^+ y^+ + q^- y^-\right) \\ T(\omega)x + y^+ - y^- = h(\omega) \\ x \geq 0, y^+ \geq 0, y^- \geq 0 \end{array} \right. \quad (2.5)$$

$$D = \{x \in \mathbb{R}^n / Ax = b, x \geq 0 \text{ et } \forall \omega \in \Omega, \exists y / T(\omega)x + W(\omega)y = h(\omega), y \geq 0\}$$

Théorème 6: (Kall 1976)

$Q(x, \omega)$ Est fini, si est seulement si $q^+(\omega) + q^-(\omega) \geq 0$ avec probabilité 1.

(b) Modèle avec seuil de probabilité sur les contraintes :

L'idée de modélisation «chance constraints» consiste à imposer que la violation des contraintes ne se produise qu'avec une probabilité fixée. Soit l'on impose un seuil de probabilité individuel α_i , avec pour $0 \leq \alpha_i \leq 1$ chaque contrainte $i: 1, \dots, n$. Il représente la probabilité avec laquelle la contrainte i doit être vérifiée. Le problème déterministe équivalent est donc :

$$\left\{ \begin{array}{l} \min EC(\omega)x \\ P(T_i(\omega)x \geq d_i(\omega)) \geq \alpha_i, \quad i: 1, \dots, n \\ x \in D \end{array} \right. \quad (2.6)$$

Pour plus de détaille voir [1]

Considérons maintenant le système linéaire suivant :

$$\begin{cases} Ax = 0 \\ x \geq 0 \end{cases} \quad (2.7)$$

On peut démontrer facilement l'existence d'une solution réalisable pour ce système, mais comment prouver la non-existence de solutions réalisables : l'outil nécessaire est le lemme de Farkas.

Lemme Farkas : (condition nécessaire et suffisante pour la faisabilité d'un système d'équation linéaire).

L'ensemble $\{x / Ax = b, x \geq 0\} \neq \emptyset$ si et seulement si $A^T u \geq 0$ implique $b^T u \geq 0$.

Remarque 3 : Un recours est complet cela implique $\{y / Wy = z, y \geq 0\} \neq \emptyset$.

Du lemme de Farkas on déduit que $(W^T u \geq 0)$ implique $(z^T u \geq 0)$.

En changeant le signe de u , nous obtenons : $(W^T u \leq 0)$ implique $(z^T u \leq 0)$.

Ceci peut être formulé comme : implique $(z^T W \leq 0)$.

2.2.1.1 Décomposition de benders (L-shaped méthode) :

La décomposition de benders (génération des contraintes) est une technique pour résoudre certaines classes de problème difficile comme la programmation stochastique et les problèmes de programmation non linéaire. L'algorithme classique de décomposition de benders est proposé pour les problèmes de la programmation mixte en nombre entière.

Le problème peut s'écrire (voir [7]) sous forme :

$$\begin{cases} \min_{x,y} c^T x + f^T y \\ t.q. \\ Ax + By \\ y \in Y, x \geq 0 \end{cases} \quad (2.8)$$

Dans le cas où y est fixé à \bar{y} , le modèle à résoudre est :

$$\begin{cases} \min_{x,y} c^T x \\ t.q \\ Ax \geq b - B\bar{y} \\ x \geq 0 \end{cases} \quad (2.9)$$

Le concept principale est de partitionner les variables de décisions en deux sous ensembles x et y , dans ce cas la méthode de benders décompose le problème en question en séquence de sous problème linéaire. La programmation stochastique constitue une classe sur laquelle se fonde fortement. Plusieurs algorithmes sont désigner à résoudre les problèmes stochastiques à deux niveaux avec recours qui sont généralement une extension de la méthode L-shaped de Van Slyke & wets (1969). Soit R le nombre de scénarios, L-shaped résout un sous problème pour chacune des R réalisations, si un des problèmes est infaisable, on rajoute une coupe de faisabilité, autrement on génère une coupe d'optimalité.

2.3 Programmation linéaire stochastique multi-objective

2.3.1 Introduction :

Dans le domaine de la recherche opérationnelle, nous rencontrons souvent des problèmes réels où le décideur souhaite optimiser plusieurs objectives à la fois, cependant, et dans la plupart des cas, la valeur de certains paramètres du problème est incertaine au moment de prendre de décision, ces paramètres qu'on peut considérés comme des variables aléatoires et le problème résultant est dit dans la recherche opérationnelle, problème de la programmation stochastique multi-objective.

Il existe dans la littérature peut de méthodes qui sont développées pour résoudre le problème MOLPS. Dans une tentative de prévoir quelques significations pour (P_0) en un problème équivalent déterministe et ainsi avoir des solutions pour tel problème transformé. La plupart des approches utilisent des méthodes interactives pour résoudre le problème déterministe. Par exemple la Méthode PRODRADE (Probabilistic Trade –off Development method) été établie par Goicochea Dukstein et Bulfin en 1976. Elle s'applique à un problème général, avec des objectifs stochastiques et des contraintes non linéaires. La méthode nécessite l'introduction d'une fonction d'utilité, ce qui rend son application limitée.

Autres méthodes, la méthode de STRANGE ([51]) s'adresse au modèle suivant :

$$(P_0) = \begin{cases} \text{"min"} z_k = c_k x & k = 1, \dots, K \\ x \in D = \{x / Tx = d, x \geq 0\} \end{cases} \quad (2.10)$$

Avec $x(n \times 1)$; $c_k(1 \times n)_{k=1, \dots, K}$; $T(m_0 \times n_0)$; $h(m_0 \times 1)$; où c_k, T, h défini des variables aléatoires discrètes sur un espace de probabilité (Ω, Ξ, P) . Chaque critère k dépend d'un ensemble de scénarios $\{s_k / k : 1, \dots, K\}$ et à chacun de ces scénarios sont associés, lors d'application -des niveaux de plausibilité p_{ks_k} .

Notons c_{ks_k} la réalisation du vecteur c_k sous le scénario s_k . de sorte que :

$$P(c_k = c_{ks_k}) = p_{ks_k}; \sum_{s_k=1}^{S_k} p_{ks_k}$$

De même soit $(T_r, d_r) r : 1, \dots, R$ les réalisations envisagées pour les coefficients (T, d) avec des probabilités correspondantes :

$$P(T = T_r, d = d_r) = q_r; \sum_{r=1}^R q_r = 1$$

Dans PROMISE d'Urli & Nadeau[54], ils utilisent une approche similaire à celle de STRANGE. Les deux méthodes sont adaptées à celle de STEM. Dans cette partie, nous intéressons au cas entiers où le rajout des variables entiers entraîne plus de difficulté au problème MOSLP pour déterminer l'ensemble des solutions efficaces. Les méthodes existante traitant ce genre de problème est la méthode interactive de Teghem[53] (STRANGE MOMIX), vient ensuite la méthode des coupes par M.Abbas & F.Bellahcene[1], que nous présentons en détaille dans la suite de ce chapitre.

2.3.2 Méthode M.Abbas & F.Bellahcene:

La première phase de la méthode consiste à transformé le problème original (MOILPS) en un problème déterministe multi objective linéaire en nombre entiers (MOILP). Puis une technique des coupes planes est utilisée pour générer l'ensemble des solutions efficaces du modèle équivalent. L'algorithme combine entre la méthode des coupes de abbas & Moulai[4] et la méthode L-shaped décrite en [7].

2.3.2.1 Le modèle : La méthode s'adresse au modèle suivant :

$$(P) \begin{cases} \min_x Z_k = c_k(\omega)x \quad k : 1, 2, \dots, K \\ t.q \quad Ax = b \\ T(\omega)x = h(\omega) \\ x \geq 0, x \text{ entier} \end{cases} \quad (2.11)$$

Où c_k, T, h sont des matrices aléatoires de dimensions respectives $(1 \times n), (m_0 \times n), (m_0 \times 1)$ définie sur un espace de probabilité (Ω, Ξ, p) . A et b sont déterministes de dimensions.

$(m \times n), (m \times 1)$.

2.3.2.2 Le problème déterministe associé :

Supposons que c_k, T, h dépendent d'un vecteur aléatoire discrète $\omega \in \Omega$, avec un support fini ω^r , $r \in \{1, 2, \dots, R\}$.

Utilisations du concept de la programmation linéaire stochastique avec recours dans le cas uni-critère.

- Définir des pénalités $q^r = q(\omega^r) z^r$ $r \in \{1, 2, \dots, R\}$ de contraintes violées.
- Définir une fonction de recours qu'on rajoute à chaque critère

$$z_{kr} = c_k(\omega^r) z^r, r \in \{1, 2, \dots, R\}.$$

- Cette pénalité est défini par :

$$Q(x, \omega^r) = \min_y \left\{ (q^r)^t y^r / W(\omega^r) y = h(\omega^r) - T(\omega^r) x, y \geq 0 \right\} \quad (2.3).$$

Le problème déterministe équivalent a (P) revient à minimiser l'espérance des coûts totaux fixés par le décideur et les coûts de pénalité soit :

$$(P_1) \begin{cases} \tilde{z}_k = z_k' + Q(x), k = 1, 2, \dots, K \\ t.q \\ Ax = b \\ x \geq 0, x \text{ entier} \end{cases} \quad (2.4)$$

Avec :

$$z_k' = E(z_k) = \sum_{r=1}^R P^r z_{kr} = \sum_{r=1}^R P^r c_k(\omega^r) x.$$

$$Q(x) = E(Q(x, \omega)) = \sum_{r=1}^R P^r Q(x, \omega^r) = \sum_{r=1}^R P^r (q^r)^t y^r.$$

2.3.3 Notations et définitions

2.3.3.1 Faisabilité de deuxième niveau : Soit x^0 une solution du problème de premier niveau définit par :

$$(P_2) \begin{cases} \min z_k' = E c_k(\omega^r) x = \sum_{r=1}^R P^r c_k(\omega^r) x \\ t.q. \\ Ax = b \\ x \geq 0 \end{cases} \quad (2.5)$$

Nous vérifions la faisabilité de deuxième niveau pour x^0 et tous ω^r de ω .

Tous d'abord considérons le dual de (1) $\max_{\pi} \left\{ \pi^t h(\omega^r) - T(\omega^r)x / \pi^t \omega \leq (q^r)^T \right\}$ (2.6).

(On suppose que la matrice de recoure est fixé). Soit $P = \left\{ \pi / \pi^t W \leq (q^r)^t \right\}$ (indépendant de x).

Théorème 7 : soit $\{ \pi^t / t \in T \}$ l'ensemble des points extrêmes de P et $\{ \sigma_{\delta} / \delta \in \Delta \}$ l'ensemble de ses arêtes efficaces, il est montré dans [40] que :

- Si $P = \emptyset$, alors $Q(x^0, \omega^r)$ est non borné ($Q(x^0, \omega^r) = -\infty$) ou irréalizable ($Q(x^0, \omega^r) = +\infty$).

- Si $P \neq \emptyset$, $Q(x^0, \omega^r)$ est irréalizable ou possède une solution optimale.

D'autre part, d'après le lemme de Farkas on a : $P = \left\{ y / Wy = h(\omega^r) - T(\omega^r)x^0 \right\} \neq \emptyset$ si est seulement si $\sigma^t W \leq 0$ implique que $\sigma^t (h(\omega^r) - T(\omega^r)x^0) \leq 0$.

Nous pouvons conclure que $Q(x^0, \omega^r)$ est irréalizable si P à une arête extrême σ tel que $\sigma^t (h(\omega^r) - T(\omega^r)x^0) > 0$. Autrement, la valeur optimale de $Q(x^0, \omega^r)$ est donnée par $\pi^t (h(\omega^r) - T(\omega^r)x^0)$. Pour vérifier la faisabilité de deuxième niveau, nous devons trouver un vecteur σ en résolvant le programme suivant :

$$\max_{\sigma} \left\{ \sigma^t (h(\omega^r) - T(\omega^r)x^0) / \sigma^t W \leq 0, \|\sigma\|_1 \leq 1 \right\} \quad (2.7)$$

Si pour un certain $\omega^r, r: 1, 2, \dots, R$, $\sigma^t (h(\omega^r) - T(\omega^r)x^0) > 0$, où σ^r est la solution optimale de (2), alors nous avons trouvé un ω^r pour lequel x^0 rend le problème de deuxième niveau infaisable. Dans ce cas nous créons la coupe de faisabilité

$$\sigma^t (h(\omega^r) - T(\omega^r)x^0) \leq 0 \quad (2.8).$$

2.3.3.2 Optimalité de deuxième niveau : Si le problème de deuxième niveau est réalisable nous pouvons considérer son dual :

$$\max_{\pi} \left\{ \pi^t h(\omega^r) - T(\omega^r)x / \pi^t \omega \leq (q^r)^T \right\} \quad (2.9)$$

Du principe de la dualité, le primale est non borné si est seulement si le dual est irréalizable. Supposons que le problème primal est borné.

Reformulons le problème (P_1) en introduisant une nouvelle variable θ .

$$(P_2) \begin{cases} \min_x \tilde{z}_k = z_k + \theta & k: 1, 2, \dots, K \\ t.q \quad x \in S \\ \theta \geq Q(x) \\ x \text{ entier} \end{cases} \quad (2.10)$$

Où $S = \left\{ x \in \mathbb{R}^n / Ax = b, \sigma^r [h(\omega^r)] \geq \sigma^r [T(\omega^r)], r:1,2,\dots,R, x \geq 0 \right\}$
 $= \left\{ x \in \mathbb{R}^n / \tilde{A}x = \tilde{b}, x \geq 0 \right\}$, est un polyèdre compact non borné de \mathbb{R}^n .

Le problème génère l'ensemble de toutes les solutions efficaces de (P_2) .

Considérons le problème linéaire suivant :

$$(P_3) \begin{cases} \min \tilde{z}_1 = z_1 + \theta \\ x \in S \\ \theta \geq Q(x) \\ x \text{ entier} \end{cases} \quad (2.11)$$

La première étape consiste à trouver une solution réalisable entière pour (P_3) . On ne peut pas directement utiliser $\theta \geq Q(x)$ comme contrainte puisque $Q(x)$ est défini implicitement par un grand nombre de problème d'optimisation. On résout le problème (P_3) sans la contrainte $\theta \geq Q(x)$ et nous obtenons une solution optimale (x_0, θ^0) (avec θ mis à $-\infty$). La structure de l'équivalent déterministe est donc exploitée en minimisant chaque fonction de deuxième niveau séparément et la décomposition se base sur le problème primal. On parlera donc de décomposition primal, ou décomposition de benders. Les solutions optimales du dual (2) sont utilisées pour calculer la valeur de l'espérance de la fonction de recours $Q(x^0)$ donnée par :

$$Q(x^0) = \sum_{r=1}^R P^r Q(x^0, \theta^0) = \sum_{r=1}^R P^r (\pi_r)^T [h(\omega^r) - T(\omega^r)x^0]. \text{ Si } \theta^0 \geq Q(x^0), x^0 \text{ est optimal}$$

pour (P_3) , si non on introduit la coupe d'optimalité :

$$\theta \geq \sum P^r \pi_r^T [h(\omega^r) - T(\omega^r)x], r:1,\dots,R \quad (5), \text{ et le problème } (P_3) \text{ sera réoptimiser à nouveau.}$$

2.4 Notations et définitions :

$S_1 = \{x \in \mathbb{R}^n / \tilde{A}_1 x = \tilde{b}_1, x \geq 0\}$ La région tronquée de S obtenue par une succession de coupes de Gomory,

$x^1 = (x_j^1)$ Solution entière de (P_3) ,

\tilde{z}_k^1 : Valeur de $\tilde{Z}_k, k:1, \dots, K$ correspond à x^1

\tilde{B}^1 : Base de S_1 ,

$(y_j^1) = (\tilde{B}^1)^{-1} a_j^1$,

$I_1 = \{i / \tilde{a}_i^1 \in \tilde{B}^1\}$,

$N_1 = \{j / \tilde{a}_j^1 \notin \tilde{B}^1\}$,

$j_1 = \{j / j \in N_1 \text{ et } \tilde{z}_{1,j} - \tilde{c}_{1,j} = 0\}$ Avec $c_{1,j}$ est la j^{eme} composante hors base,

Pour $\alpha \geq 2$,

$S_\alpha = \{x \in \mathbb{R}^{n_\alpha} / \tilde{A}_\alpha x = \tilde{b}_\alpha, x \geq 0\}$ est la région tronquée obtenue par application de la coupe

$\sum_{j \in N_{\alpha-1} \setminus \{j_{\alpha-1}\}} x_j \geq 1, j_{\alpha-1} \in J_{\alpha-1}$ et éventuellement des coupes de Gomory pour réoptimiser (P_3) ,

$x^\alpha = (x_j^\alpha)$ Est la α^{eme} solution optimale entière de (P_3) obtenue dans S_α à l'étape α ,

$x_l^\alpha = (x_{l,j}^\alpha), l:1, \dots, L_\alpha$ La solution entière alternative à x^α si elle existe,

\tilde{B}^α : La base de S_α ,

$(y_j^1) = (\tilde{B}^\alpha)^{-1} \tilde{a}_i^\alpha$

$I_\alpha = \{i / \tilde{a}_i^\alpha \in \tilde{B}^\alpha\}$

$N_\alpha = \{j / \tilde{a}_j^\alpha \notin \tilde{B}^\alpha\}$

$J_\alpha = \{j / j \in N_\alpha \text{ et } \tilde{Z}_{1,j} - C_{1,j} = 0\}$

2.4.1 Solution de la méthode :

La méthode génère l'ensemble de toutes les solutions efficaces du problème (P_2) .

Etape 1 : Poser $\theta = -\infty$, sans coupe de réalisabilité et d'optimalité minimiser l'objective

$Z_1 = E(C_1(\omega)x)$ sous des contraintes déterministes. Soit x^1 la solution réalisable entière

pour chaque $r \in \{1, 2, \dots, R\}$ résoudre le problème (3) et trouver une direction σ_r .

- Si pour $\sigma_r' [h(\omega^r) - T(\omega^r)x^1] \leq 0, \forall r \in \{1, \dots, R\}$ aller à b.
 - S'il existe $r \in \{1, \dots, R\}$ tel que $\sigma_r' [h(\omega^r) - T(\omega^r)x^1] > 0$ introduire la coupe de réalisabilité (4) réoptimiser le problème (P₃) pour trouver une autre solution x^1 , une coupe de Gomory et rajoutée, si la solution trouvée n'est pas entière, aller à (b).
- b) Résoudre le problème dual et calculer $Q(x^1)$.
- Si $\theta \geq Q(x^1)$ alors x^1 est optimale et θ sa valeur correspondante.
 - Si $\theta < Q(x^1)$ introduire une coupe d'optimisation (5) pour avoir la valeur optimale de x^1 et de θ correspondant dans la région tronquée S_1 (aller à étape 2)

Etape 2 :

1- Construire l'ensemble J_1

Si $J_1 = \emptyset$, x^1 est l'unique solution optimale entière du problème P (3).

- Evaluer chaque objectif en x^1 et pose $Eff_0 = (\tilde{Z}_1^2, \tilde{Z}_2^2, \dots, \tilde{Z}_k^2)$
- Tronquer le point x^1 pour trouver une autre solution réalisable entière x^2 . La coupe de réalisabilité est ajoutée si le problème de deuxième niveau n'est pas réalisable.
- Calcule $Q(x^2)$ et vérifier si la valeur de θ trouvée à la première étape est supérieur à $Q(x^2)$. Si oui alors x^2 est optimale

Si non une coupe d'optimalité (5) est ajoutée pour avoir la valeur optimale de θ . Soit $(\tilde{Z}_1^2, \tilde{Z}_2^2, \dots, \tilde{Z}_k^2)$ le K-uplet correspond à x^2 et l'ajouté à Eff_0 si il n'est pas dominé par la première k-uplet. Remplacer Eff_0 par Eff_1 .

2- Si $J \neq \emptyset$, choisir un indice $j_1 \in J_1$ et calculer le nombre :

$$\Phi = \min \left\{ \frac{x_{1,j}}{y_{1,ij}}, y_{1,ij} > 0 \right\} \text{ correspond à } x^1.$$

- Si $\Phi > 1$, calculer toutes les solutions réalisables entières $x_l^1, l \in \{1, \dots, L_1\}$ alternatives à x^1 le long de l'arête E_{j_1} , tester la réalisabilité du problème du deuxième niveau, ajouter des coupes de réalisabilités si nécessaire.

Calculer $x_1^l, l \in \{1, \dots, L_1\}$ est tester l'optimalité de x_1^l (ajouter des coupe (5) si nécessaire).

Chaque solution x_1^l donnera lieu à un éventuel k-uplet $(\tilde{Z}_1^2, \tilde{Z}_2^2, \dots, \tilde{Z}_k^2)$. Tronquer l'arête E_{j_1} par la coupe $\sum_{j \in N_1 - \{j_1\}} x_j \geq 1$, appliquer du dual simplexe et éventuellement des coupes de

Gomory si nécessaire. Soit x^2 une solution entières réalisable obtenue dans la région tronquée S_2 , lire le K-uplet $(\tilde{Z}_1^2, \tilde{Z}_2^2, \dots, \tilde{Z}_k^2)$ et l'ajouter à Eff_0 si il n'est pas dominé par le premier K-uplet.

- Si $\Phi < 1$ pour tous $j_1 \in J_1$, choisir un $j_1 \in J_1$ et appliquer la coupe $\sum_{j \in N_1 - \{j_1\}} x_j \geq 1$.
 - Appliquer le dual simplexe et éventuellement des coupes de Gomory si nécessaire. Tester la réalisabilité et l'optimalité de la solution admissible x^2 obtenue dans la région tronquée S_2 . Ajouter $(\tilde{Z}_1^2, \tilde{Z}_2^2, \dots, \tilde{Z}_k^2)$ à Eff_0 s'il n'est pas dominé par les k-uplet déjà identifier. Dans ce cas Eff_0 devient Eff_1 .

Etape α : $\alpha \geq 3$

Choisir $j_{\alpha-1} \in J_{\alpha-1}$, scanner l'arête $E_{j_{\alpha-1}}$ correspondante pour chercher les solutions admissibles entières $x_1^{\alpha-1}, l: 1, \dots, L_{\alpha-1}$ alternative à $x^{\alpha-1}$, tester la réalisabilité, ajouter des coupes de réalisabilité (4) si nécessaire. Trouver $Q(x_1^{\alpha-1})$ et tester la valeur de θ trouvée à l'étape précédente si elle est supérieur à $Q(x_1^{\alpha-1})$. Si ce n'est pas le cas ajouter la coupe d'optimalité (5). Lire les nouveaux K-uplet $(\tilde{Z}_{1,l}^{\alpha-1}, \tilde{Z}_{2,l}^{\alpha-1}, \dots, \tilde{Z}_{k,l}^{\alpha-1})$ (s'il y en a). Construire l'ensemble $Eff_{\alpha-1}$. Rejeter ceux qui sont dominés par les K-uplets précédents.

L'arête $E_{j_{\alpha-1}}$ est tronquée par la coupe $\sum_{j \in N_1 - \{j_1\}} x_j \geq 1$.

Etape finale :

La procédure se termine quand une impossibilité des opérations pivots apparaît, indiquant que la région tronquée ne contient aucun point entier réalisable, et que tous les points efficaces sont obtenus.

Définition 5 :

Une arête E_{j_α} incidente à une solution x^α est définie comme l'ensemble :

$$E_{j_\alpha} = \left\{ x = (x_i) \in S / \begin{array}{l} x_i = x_i^\alpha - \phi_{j_\alpha} y_i^\alpha, j_\alpha, \alpha \in I_\alpha \\ x_{j_\alpha} = \phi_{j_\alpha} \\ x_s = 0 \text{ pour tous } s \in N_\alpha - \{j_\alpha\} \end{array} \right\}$$

Définition 6 :

Une arête E_{j_α} incidente à une solution réalisable entière x^α est dite dominée si toutes solutions entières le long de cette arête donnent des p-uplets $(\tilde{Z}_1, \tilde{Z}_2, \dots, \tilde{Z}_p)$ dominées.

Définition 7 :

Toutes solutions réalisables entières $x_l^\alpha, l:1, \dots, l_\alpha$ du problème (P_3) alternative à x^α sur une arête E_{j_α} de la région (S) émanant d'elle dans la direction d'un vecteur $\tilde{a}_{j_\alpha}^\alpha j_\alpha \in \Gamma_\alpha$ existe dans le demi espace $\sum_{j \in N_\alpha - \{j_\alpha\}} x_j < 1$.

Corollaire 3 : Toute solution entière du problème (P_3) qui n'est pas sur une arête E_{j_α} , $j_\alpha \in \Gamma_\alpha$ de la région S à travers un point réalisable x^α se trouve dans le demi espace fermé $\sum_{j \in N_\alpha - \{j_\alpha\}} x_j \geq 1$.

Conclusion :

Nous nous sommes intéressés dans ce chapitre au problème de programmation stochastique linéaire multi-objective, où le rajout des variables entières complique d'avantage le problème. Pour les paramètres incertains intervenants dans le problème sont traités par l'approche du recours pour avoir un problème équivalent déterministe à deux niveaux. Comme nous avons donné un aperçu sur les méthodes existantes dans le cas continu et donné en détaille la méthode des coupes planes présentée dans [1] pour la résolution du problème MOILPS où un algorithme de génération des solutions efficaces a été présenté. Même cet algorithme est approprié seulement pour des problèmes avec un petit nombre de scénarios, il peut être appliqué pour un grand nombre d'objectifs.

Chapitre III

*Optimisation sur un ensemble des solutions
efficaces MOILPS*

3. Introduction :

Un problème de programmation linéaire multi-objective (MOLP) peut s'écrire comme suit :

$$\text{"maximiser "}\{cx / x \in S\} \quad (3.1)$$

$$S = \{x \in \mathbb{R}^n / Ax \leq b, x \geq 0\} \quad (3.2)$$

c est une matrice $(p \times n)$, A est une matrice $(m \times n)$, $b \in \mathbb{R}^m$. Un point (où une solution réalisable) $x^0 \in S$ dite efficace pour le problème (1) et (2) si il n'existe pas $x^1 \in S$ tel que $cx^1 \geq cx^0$ et $cx^1 > cx^0$. Notons par E l'ensemble de toutes les solutions efficaces du problème (1) et (2). Dans le processus de la prise de décision impliquant des modèles de programmations Multi-objectifs, choisir la solution la plus préféré, le décideur souvent analyse le sous ensemble des solutions efficaces (il se peut qu'il existe un grand nombre de solution). Dans ce cas le décideur quantifie sa préférence qui se donne explicitement sous forme d'une fonction $f: \mathbb{R}^n \rightarrow \mathbb{R}$. Donc le problème de recherche de solution la plus préféré devient :

$$\max \{f(x), x \in E\} \quad (3.3)$$

Un cas spécial du problème (3) qu'on peut rencontré souvent :

$$\max \{\phi(x) = d(x), x \in E\} \quad (3.4)$$

$$\max \{c_i x / x \in E\} \quad (3.5)$$

ϕ est une fonction linéaire, c_i est la i^{eme} ligne de la matrice c . Les problèmes (3.4) et (3.5) ont été étudiés par Philipe [41], Fulöp [25], Benson [8,9], Benson et Sayin [10,11], Dessouky et al [20], Isermann et Steuer [33], Reeves et Reid [43], Steuer [45], White [58], Weistrofner [57]. Le problème (3) est considéré par : Dauer [17], Dauer et Fosnaugh [18], Bolintineanu [12], Horst et Thoai [31]. Pour résoudre (3.4) et (3.5), la première méthode a été proposée par Philip [41], sa méthode est basée sur le déplacement sur les sommets efficaces adjacents dans le cas où ϕ est linéaire. Benson à le premier qui suggéré une méthode de programmation non convexe. Philip, Isermann et Steuer, utilise une approche basée sur le simplexe (procédure des coupes planes). Fulöp, propose une méthode de recherche de point extrêmes efficaces adjacentes.

Ces méthodes sont considérées comme combinaison des approches de programmation linéaire de Philip, Isermann et Steuer et l'approche de la programmation non convexe de

Benson. Dans les méthodes Ecker et Kouada [21], Yu et Zeleny [59], Isermann[32], le problème se base sur la recherche de points et arêtes efficaces. La difficulté de résolution de problème d'optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces E est dû à la non convexité de cet ensemble, d'autre part elle nous permis d'éviter le calcul exigé de toutes les solutions efficaces et elle est considéré l'une des plus importante approches dans l'optimisation multi-critère. Nous donnons par la suite quelques résultats de base et définitions de la programmation linéaire multi-objective.

3.1 Optimisation sur un ensemble des solutions efficaces d'un problème MOLP

3.1.1 Résultats de base :

Notons par :

$$\mathbb{R}_+^k = \{x \in \mathbb{R}^k /, x \geq 0\}, \mathbb{R}_{++}^k = \{x \in \mathbb{R}^k /, x > 0\} .$$

$$\mathbb{R}_-^k = \{x \in \mathbb{R}^k /, x \leq 0\}, \mathbb{R}_{--}^k = \{x \in \mathbb{R}^k /, x < 0\} .$$

\mathbb{R}_k Désigne les vecteurs lignes.

e désigne un vecteur dont toutes les composantes sont égales à 1, sa dimension est fixé par le contexte.

Définition8 : l'ensemble $Y = \{y \in \mathbb{R}^p; y = cx \text{ pour } x \in S\}$ est appelé l'ensemble des résultats. L'ensemble $Y^{\leq} = Y + \mathbb{R}_-^p = \{y \in \mathbb{R}^p, y \leq cx, x \in S\}$ est appelé l'ensemble des résultats inférieurs. L'ensemble $Y^{<} = Y + \mathbb{R}_{--}^p = \{y \in \mathbb{R}^p, y < cx, x \in S\}$ est appelé l'ensemble des résultats strictement inférieurs.

Définition9 : Pour $\lambda \in \mathbb{R}_{++}^p$ et $x \in S$,

La fonction $g_\lambda(x) = \max \{\lambda cx' / x' \in S; cx' \geq cx\} - \lambda cx$ est dite fonction lacune.

Quand $\lambda = e = (1, 1, \dots, 1) \in \mathbb{R}_p$, g_λ est noté par g .

Définition10 :

Un point $y \in Y$ est dit solution non dominée du problème MOLP si $Y \cap (y + \mathbb{R}_+^p) = \{y\}$. Un point $y \in Y$ est dit non dominé si $Y \cap (y + \mathbb{R}_{++}^p) = \emptyset$.

Théorème8 : L'ensemble E des solutions efficaces d'un problème linéaire Multi-objectifs continu est connexe. Deux sommets quelconques dans E sont reliés par un chemin formé d'arêtes efficaces (une arête efficace est une arête de S contenu dans E). Définissons le problème d'optimisation d'un critère linéaire sur l'ensemble des solutions efficaces par :

$$(PL_E) = \begin{cases} \max \phi(x) \\ x \in E \end{cases} \quad (3.6)$$

E est l'ensemble des solutions efficaces du problème :

$$MOLP \begin{cases} \text{"max"} cx \\ \text{t.q. } x \in S \end{cases} \quad (3.7)$$

$$S = \{x \in \mathbb{R}^n / Ax \leq b, x \geq 0\}, A \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^n, b \in \mathbb{R}^m, c \in \mathbb{R}^{p \times n}.$$

Le problème relaxé est donné par :

$$(PL_R) \begin{cases} \max \phi(x) \\ \text{t.q. } x \in S \end{cases} \quad (3.8)$$

Nous nous intéressons par la suite à donner les étapes d'une des méthodes de résolution du Problème (PL_E) dans le cas où ϕ est linéaire.

3.1.2 Méthode d'Ecker & Song [22] :

La méthode d'Ecker & Song (1994) est utilisée pour la résolution du problème (PL_E) .

Les deux auteurs ont basés dans leur méthode sur deux techniques.

- Se déplacé d'un sommet efficace à un sommet voisin efficace avec une plus grande valeur de la fonction objective via une arête efficace.
- Ajouté des coupes à la région admissible afin d'échapper au problème d'optimalité local.

En effet, étant donné un point efficace courant \tilde{x} , cherchant les arêtes adjacentes à \tilde{x} , on utilise une technique de pivotement. Si aucune arête efficace adjacente améliorant $\phi(x)$, dans ce cas nous ajoutons une coupe à la région S .

3.1.2.1 Notations :

Le problème (PL_E) est donné par :

$$(PL_E) = \begin{cases} \max \phi(x) \\ x \in E \end{cases} \quad (3.9)$$

Où $d \in \mathbb{R}_p = \mathbb{R}^{1 \times p}$.

Et soit le problème multi-objectif suivant :

$$(\overline{MOLP}) \begin{cases} \text{"max"} z_k = c^k x, \quad k : 1, 2, \dots, p \\ \text{t.q. } x \in \overline{S} \end{cases} \quad (3.10)$$

$\overline{S} = \{x \in S / dx \geq d\tilde{x}\}$. Et soit \overline{E} l'ensemble des solutions efficace de (\overline{MOLP}) .

Considérons aussi le problème

$$(I_j) \begin{cases} \max c^i x, & i : 1, 2, \dots, p \\ \text{t.q. } x \in \bar{S} \end{cases} \quad (3.11)$$

Associons au problème (PL_E) le problème relaxé (PL_R) :

$$(PL_R) \begin{cases} \max \phi(x) \\ \text{t.q. } x \in S \end{cases} \quad (3.8)$$

Soit x^* une solution optimale de (PL_R) , pour tester si x^* est efficace on résout $P(x^*)$:

$$P(x^*) \begin{cases} \max e' \psi \\ \text{t.q. } cx = \psi + cx^* \\ x \in S, \psi \geq 0 \end{cases} \quad (3.12)$$

Où e est un vecteur ligne de composantes égales à 1 et $\psi \in \mathbb{R}$.

Si $\psi=0$, alors x^* est efficace.

Dans leur méthode, Ecker & Song également utilisés une technique de pivotement pour trouver les arêtes efficaces adjacentes à un point extrême efficace \tilde{x} .

Soit \tilde{x} une solution efficace initiale du problème $(MOLP)$, le tableau simplexe correspondant est :

T	x_N	x_B
D	-c	0
B	\tilde{A}	I

Où x_N indiquent les variables hors base et x_B les variables de base.

Dans le tableau T nous avons éliminé les variables de base des objectifs pour avoir des objectifs équivalents de composantes $d + cx_N$.

Dans l'ordre de déterminer une arête incidente à \tilde{x} efficace, nous donnons le théorème suivant :

Théorème 9 [22]: Etant donné le tableau T correspond à la solution \tilde{x} .

Considérons le programme linéaire (Q^j) défini par :

$$(Q^j) \begin{cases} \max z_Q = e's \\ \text{t.q } cx = s + c(j) \\ x \geq 0, s \geq 0 \end{cases} \quad (3.13)$$

Où $c(j)$ désigne la colonne j de c dans T . Soit F^j l'arête incidente à \tilde{x} correspond à l'entrée j . Alors $F^j \subseteq E$ ssi $z_Q = 0$ dans le problème (Q^j) .

Considérons le dual de (Q^j) défini par :

$$(\bar{Q}^j) \begin{cases} \min w'_Q = -c'(j)y \\ \text{t.q } c'y \geq 0 \\ -y \geq e \end{cases} \quad (3.14)$$

Posons $y = -v - e$; on obtient :

$$(\bar{\bar{Q}}^j) \begin{cases} \min -e'c(j) - v'c(j) \\ \text{t.q } c'v \geq -c'e \\ v \geq 0 \end{cases} \quad (3.15)$$

Les auteurs ont montré que le problème (Q^j) admet une solution si et seulement si l'ensemble :

$$\Omega = \{(v, w_N) \geq 0 / c'v + w_N = -c'e, w_{N_j} = 0\} \neq \emptyset . w_N \text{ est l'indice de } x_N \text{ dans } T.$$

Nous utiliserons l'ensemble Ω pour déterminer quelle est l'arête incidente à \tilde{x} est efficace.

Les équations de Ω peuvent être représenté par le tableau ci-dessous :

t^0	v	\tilde{w}_N
$-c'e$	c'	I

Teste d'arrêt

Définition11 : Soit $\Omega_T = \{(v, w_N) \geq 0 / c'v + w_N = -c'e\}$, la variable w_N^j est dite non redantante dans Ω_T si il existe un point $[\hat{v}, \hat{w}_N] \in \Omega_T$, tel que $\hat{w}_N^j = 0$. Si non w_N^j est dit redondant dans Ω_T .

Corollaire 4 :

Si T est efficace, alors $F^j \subseteq E$ Si est seulement si w_N^j est non redondant dans Ω_T .

3.2. Algorithme de la méthode :

Etape 0 : Trouver une solution optimale de (PL_R) Pour tester si x est efficace il suffit de résoudre le problème $P(x^*)$.

- Si $\psi = 0$; alors $x^* \in E$ et x est une solution optimale de (PL_E) .
- Sinon, soit \tilde{x} une solution optimale du problème $P(x^*)$.
 - Si $d\tilde{x} = dx^*$ alors $P(x^*)$ est une solution de (PL_E)
 - Sinon, aller à l'étape 1.

Etape 1 : Ajuster un tableau simplexe initial de (PL_R) à la solution (optimale pour $P(x^*)$).

Au point extrême efficace courant, soit \tilde{x} , trouver si possible une arrête efficace adjacente qui améliore la valeur de la fonction $\phi(x) = dx$.

- S'il n'y a aucune arrête efficace qui rapporte une augmentation en valeur de la fonction Objectif $\phi(x)$ alors passer à l'étape 3.

- Autrement, pivoter au prochain point extrême \hat{x} de l'arrête efficace choisie et faire une autre itération de l'étape 2 en partant de $\tilde{x} = \hat{x}$.

Etape 3 : Ajouter la ligne qui représente la coupe $dx \geq d\tilde{x}$ et une variable d'écart x_{n+1} pour obtenir un tableau simplexe $\overline{T\tilde{x}}$ correspondant à \tilde{x} comme point extrême dans \overline{S} .

Etape 4 : appliquer les étapes 4.1 à 4.4.

4.1 : Soit $j=1$

4.2.a : Trouver une solution optimale x^j de (I_j) qui est une solution extrême efficace du problème (\overline{MOLP}) .

4.2.b : Si $dx^j \geq d\tilde{x}$, soit x^j un nouveau point extrême efficace, faire $\tilde{x} = x^j$. Ajuster le tableau courant pour correspondre à x^j comme point extrême dans S . Aller à l'étape 2.

4.2.c : Si $dx^j = d\tilde{x}$ et s'il y a une arête efficace adjacente $[x^j, \bar{x}]$ rapportant une augmentation de dx , alors utiliser \bar{x} comme nouveau point efficace courant \tilde{x} .

Aller à l'étape 2.

4.2d Si $j \leq k-1$, faire $j = j+1$ et aller à (4.2a).

4.3a Si $dx \leq 0$ pour tout x non nul, alors le sous-ensemble de S , $F = \{x \in S / dx = d\tilde{x}\}$

(Qui est une face de S) est inclus dans \overline{E} .

Pivoter dans le domaine F en cherchant un point y qui a une arête efficace adjacente dans S rapportant une augmentation de dx . Si un tel y n'existe pas alors le point courant x est une solution optimale de (PL_E) .

4.3b Si $dx \geq 0$ pour un certain x non nul, alors ce n'est pas nécessairement que toutes les faces F sont dans \overline{E} . Pivoter sur F suivant les arêtes efficaces en cherchant un point y qui a une arête efficace adjacente et rapportant une augmentation de dx . Si un tel y n'existe pas alors le point courant \tilde{x} est une solution optimale de (PL_E) . S'il y a une telle arête $[y, \tilde{x}]$ alors utilisé \tilde{x} comme un nouveau point efficace courant.

Aller à l'étape2.

3.3 Optimisation d'une fonction linéaire sur un ensemble efficace d'un problème MOILP

3.3.1 Introduction : Le problème d'optimisation sur l'ensemble des solutions efficaces en nombres entier est un problème difficile, pour cela on trouve peut de travaux réalisé dans la littérature, citons la méthode de N.C .Nguyen[38], qui donne seulement une borne supérieure de la valeur optimale de la fonction objective ϕ . Récemment une nouvelle méthode interactive est présenté par M..Abbas & D.Chaabane [3] inspiré du travaille de J.G.Ecker & H.G.Song [22]et H.P.Benson & S.Saying[10,11] pour le cas continue. Le principe est de choisir la meilleure direction qui améliore la fonction objective à chaque itération, utilisant deux types de coupes jusqu'à ce que la région tronquée courante ne contienne aucune solution réalisable.

Considérant le problème MOILP suivant :

$$(P(D)) = \begin{cases} \max Z_i = C^i x, i : 1, \dots, p \\ t.q \\ x \in D \end{cases} \quad (3.16)$$

$$D = S \cap \mathbb{Z}^n, S = \{x \in \mathbb{R}^n / Ax = b, x \geq 0\}$$

$A \in \mathbb{Z}^{m \times n}; b \in \mathbb{Z}^m, p \geq 2, c^1, c^2, \dots, c^p \in \mathbb{Z}^n$, sont des vectrices lignes, \mathbb{Z} est l'ensemble des entiers relatives. Nous supposons que D est non vide et que S est un polyèdre convexe et non borné. L'ensemble de toutes les solutions efficaces est noté par $E(P)$.

Le problème principale étudié est donné par :

$$P_E \{ \max \phi(x) = dx, x \in E(p) \} \quad (3.17)$$

Où d est un vecteur ligne de dimension n qui à pour j^{eme} composante le nombre entiers d_j .

Soit le problème relaxer :

$$P_R \{ \max \phi(x) x, x \in D \} \quad (3.18)$$

Le problème $P_i(D)$ pour $i \in \{1, 2, \dots, p\}$ est défini par :

$$P_i(D) \{ \max z_i = c^i x, x \in D \} \quad (3.19)$$

Définition 12:

Un point x^0 est dit efficace si et seulement si il n'existe pas de $x \in D$ tel que $z_i(x) \geq z_i(x^0)$ Pour tous $i \in \{1, 2, \dots, p\}$ et $z_i(x) > z_i(x^0)$ pour au moins $i \in \mathfrak{I} = \{1, 2, \dots, p\}$

Remarque 4: Si une solution optimale x^0 de $P_i(D)$ et non unique dans ce cas, il existe une autre solution réalisable $x^1 \neq x^0$ avec $z_i(x^1) = z_i(x^0)$. On dit que x^1 est une solution optimale alternative de $P_i(D)$.

Proposition 3: Un point x^0 est l'unique solution du problème de programmation linéaire en nombre entier

$$P_i(D) \{ \max z_i = c^i x, x \in D \}$$

est efficace pour $P(D)$.

3.3.2 Notations et résultat préliminaire :

- On note par Z_1, Z_2, \dots, Z_p les critères du problème et ϕ le critère additionnel ;
- $S_1 = \left\{ \left\{ x \in \mathbb{R}^n / A_1 x = b_1, A_1 \in \mathbb{Q}^{m_1 \times n_1}, b_1 \in \mathbb{Q}^{m_1}, (m_1, n_1) \in \mathbb{N} \times \mathbb{N}, m_1 \neq 0, n_1 \neq 0, x \geq 0 \right\} \right\}$,

S_1 est la région courante de S tronquer par des coupes Gomory introduite on optimisons le problème $P_1(D)$; $D_1 = S_1 \cap \mathbb{Z}^{n_1}$, note que $D_1 = D = S \cap \mathbb{Z}^n$, car les coupes de Gomory n'élimine pas les solution entiers de S .

- $(Z_1^1, Z_2^1, \dots, Z_p^1)$ le premier p-uplet correspondant à la solution entière x_1 dans S_1 ,
où $Z_i^1 = C^i x_1$, pour $i : 1, \dots, p$.

Pour $k \geq 1$, nous avons :

- $x_k \in \mathbb{Z}^{n_k}$ est la solution entière obtenue dans S_k ;
- B_k est la base associée à la solution X_k ;
- $a_{k,j} \in \mathbb{R}^{m_k \times x_1}$ le vecteur $a_{k,j}$ par rapport à la région tronquée S_k ;
- $I_k = \{j / \text{le vecteur } a_{k,j} \text{ est une collone de } B_k\}$ (indices de base) ;
- $N_k = \{j / \text{le vecteur } a_{k,j} \text{ n'est pas une collone de } B_k\}$ (indices hors base) ;
- $y_{k,j} = (y_{k,ij}) = (B_k)^{-1} a_{k,j}$, où $y_{k,j} \in \mathbb{Q}^{m_k \times 1}$;
- $\Gamma_k = \{j \in N_k / Z_{1,j}^k - C_j^1 \geq 0 \text{ et } w_j^k - d_j^k \leq 0\}$. avec $Z_{1,j}^k = C_{B_k}^1 y_{k,j}$, $C_{B_k}^1$ est le vecteur des coefficients coût de variables de bases associés à B_k dans C^1 et C_j^1 est la j^{eme} composante de vecteur C^1 , $Z_{1,j}^k = d_{B_k}^1 y_{k,j}$ ou $d_{B_k}^1$ est le vecteur de coefficient coût de variables de bases associés à B_k .

Pour $k \geq 2$,

$S_k = \left\{ \left\{ x \in \mathbb{R}^k / A_k x = b_k, A_k \in \mathbb{Q}^{m_1 \times n_1}, b_1 \in \mathbb{Q}^{m_1}, (m_1, n_1) \in \mathbb{N} \times \mathbb{N}, m_k \neq 0, n_k \neq 0, x \geq 0 \right\} \right\}$ S_k est la région tronquée obtenue après application de la coupe $\sum_{j \in N_{k-1} - \{j_{k-1}\}} x_j \geq 1$ où la coupe $dx \geq dx_k$ et des successives coupes de Gomory si nécessaire dans chacune de ces cas $D_k = S_k \cap \mathbb{Z}^{n_k}$.

Le tableau simplexe utilisé est donné sous forme :

Tableau	Variable de base	x_1	x_2	...	x_{n_k}
x_{B_k}	$x_{k,1}$	t_{11}	t_{12}	...	t_{1n_k}
	$x_{k,2}$	t_{21}	t_{22}	...	t_{2n_k}

	x_{k,m_k}	$t_{m_k 1}$	$t_{m_k 2}$...	$t_{m_k n_k}$
Coût réduit	z_1^k	$z_{1,1}^k - c_1^1$	$z_{1,2}^k - c_2^1$...	$z_{1,n_k}^k - c_{n_k}^1$

	z_p^k	$z_{p,1}^k - c_1^p$	$z_{p,2}^k - c_2^p$...	$z_{p,n_k}^k - c_{n_k}^p$
	ϕ	$\phi_1^k - d_1$	$\phi_2^k - d_2$...	$\phi_{n_k}^k - d_{n_k}$

Nous avons

$$t_{ij} = \begin{cases} e_j & \text{si } j \in I_k \\ y_{k,ij} & \text{si } j \in N_k \end{cases}$$

e_j est le j^{eme} colone de la matrice identité de dimension $m_k \times m_k$.

Théorème 10: Soit x^* un élément arbitraire de la région D . $x^* \in E(p)$ Si et seulement si la valeur optimale de la fonction \mathcal{G} dans le programme linéaire $P(x^*)$ ci-dessous est nulle.

$$P(x^*) \begin{cases} \max \mathcal{G} = \sum_{i=1}^p \psi_i \\ \text{sujet à :} \\ Cx = I\psi + Cx^* \\ x \in S \\ \psi_i \in \mathbb{R}^+ \forall i \end{cases} \quad (3.20)$$

C est la matrice $p \times n$ dont la i^{eme} ligne correspond à $c^i, i=1,2,\dots,p$. I est la matrice identité ($p \times p$) et $\psi = (\psi_i)_{i=1,2,\dots,p}$.

Définition 12 : Supposons que $j_k \in \mathbb{N}_k$ une arête incidente à une solution x_k de $P_i(D)$ est définie comme l'ensemble :

$$E_{j_k} = \left\{ (x_i) \in D_k / \begin{array}{l} x_i = x_{k,i} - \theta_{j_k} y_{k,ij_k} \text{ pour } i \in I_k \\ x_{j_k} = \theta_{j_k} \\ x_\alpha = 0, \text{ pour tous } \alpha \in \mathbb{N}_k - \{j_k\} \end{array} \right\}$$

Où $0 < \theta_{j_k} \leq \min_{i \in I_k} \left\{ \frac{x_{k,i}}{y_{k,ij_k}} > 0 \right\}$

θ_{j_k} est un entier positive et $\theta_{j_k} y_{k,ij_k}$ sont entiers pour tous $i \in I_k$ si de tel valeurs de θ_{j_k} existe.

Théorème 11: (condition suffisante d'unicité).

Si pour une solution optimale x_1 de $P_1(D)$, on a $J_1 = \{j \in \mathbb{N}_1 / Z_{1,j}^1 - C_j^1\} = \emptyset$, alors il n'existe pas d'autres solutions $x \in D$ tel que $Z_1(x) = Z_1(x_1)$ c'est-à-dire que x_1 est unique.

Théorème 12 : Une solution réalisable entière du problème $P_i(D_k)$, qui n'est pas sur l'arête E_{j_k} ($j_k \in \Gamma_k$) est incidente à x_k , de la région tronqué D_k se trouve dans le demi espace fermé $\sum_{j \in \mathbb{N}_1 - \{j_1\}} x_j \geq 1$.

Remarque 5 :

La coupe $\sum_{j \in \mathbb{N}_1 - \{j_1\}} x_j \geq 1$ est appelé coupe de type (I) et elle est considéré comme une généralisation de la coupe de Dantzig $\sum_{j \in \mathbb{N}_k} x_j \geq 1$. Si Γ_k est vide, alors toutes les solutions entières de $P_i(D_k)$ qui ne sont pas sur E_{j_k} est incidente à x_k se trouve dans le demi espace fermé défini par la coupe de Dantzig.

Calculons la valeur de la fonction linéaire $\phi(x)$ pour chaque solution $x_k^1 = (x_1^1, x_2^1, \dots, x_n^1)$ se trouvant sur arête E_{j_k} :

$$\begin{aligned} \phi(x_k^1) &= \sum_{j=1}^n d_j x_{k,j}^1 = \sum_{i \in I_k} d_i (x_{k,i} - \theta_{j_k} y_{k,ij_k}) + d_{j_k} \times \theta_{j_k} \\ &= \left(d_{j_k} - \sum_{i \in I_k} d_i \times y_{k,ij_k} \right) \theta_{j_k} + \sum_{i \in I_k} d_i x_{k,i} \end{aligned}$$

Où θ est un entier positif vérifions $0 < \theta \leq \theta_{j_k}^0$ et $\theta_{j_k}^0$ est la partie entière de

$$\min_{i \in I_k} \left\{ \frac{x_{k,i}}{y_{k,ij_k}} > 0 \right\}.$$

$$\text{On pose } \beta_k = \left(d_{j_k} - \sum_{i \in I_k} d_i \times y_{k,ij_k} \right)$$

Le long de l'arête E_{j_k} , nous avons $\beta_k \geq 0$, cependant la valeur de $\phi(x_k)$ croit et atteint son maximum pour $\theta = \theta_{j_k}^0$.

Proposition 4 :

Un point x^0 l'unique solution pour le problème linéaire en nombre entier suivant :

$$P_i(D) \begin{cases} \max Z_i = C^i x \\ t.q \ x \in D \end{cases}$$

Où Z_i est l'une des fonctions objectives du problème (P) est efficace pour $P(D)$.

Coupe de type (II) :

L'inégalité $dx \geq w_{opt}$ est dite coupe de type(II), de telles coupes sont imposées juste après avoir trouvé une nouvelle solution efficace et la mise à jour de la variable x_{opt} et la valeur $\phi(x_{opt}) = \phi_{opt}$.

3.3.3 Présentation de l'algorithme :

Etape1 :

Résoudre le problème relaxé(P_R). Soit x^* une solution optimale, cette solution est testée pour l'efficacité en résolvons le problème $P(x^*)$ (voir le théorème). Si elle est efficace, alors elle est aussi une solution de (P_E)et l'algorithme se termine. Autrement, aller à l'étape 2.

Etape 2 :

Soit $\phi_{opt} = -\infty$, on résout le problème ($P_1(D)$) (on peut considérer alternativement n'importe quel de ces problème ($P_i(D), i : 1, 2, \dots, p$) au lieu de ($P_1(D)$)).

2.1. Si $J_1 = \{j \in N_1 / Z_{1,j}^1 - C_j^1 = 0\} = \emptyset$, alors la solution optimale trouver x_1 est unique et elle est efficace (voir la proposition 1) soit $\phi^1 = dx_1$, poser $\phi_{opt} = \phi^1, x_{opt} = x_1$, aller à l'étape 3.

2.2. Si $J_1 \neq \emptyset$, alors la solution optimale x_1 du problème $(P_1(D))$ n'est pas unique, tester l'efficacité de x_1 (théorème 1) ; si elle n'est pas efficace aller à 3 ; autrement soit $\phi^1 = dx_1$ poser $\phi_{opt} = \phi^1, x_{opt} = x_1$, aller à 3.

Etape 3 : $k := 1$

3.1. Construire l'ensemble $\Gamma_k = \{j \in N_k / Z_{1,j}^k - C_j \geq 0 \text{ et } \phi_j^k - d_j^k \leq 0\}$.

- Si $\Gamma_k = \emptyset$ alors aller à 3.3 et la coupe dans ce cas et celle de Dantzig $\sum_{j \in N_k} x_j \geq 1$.
- Autrement poser $\gamma = \Gamma_k$, aller à (a).
-

(a) Si $\gamma = \emptyset$, alors soit $j_k \in \Gamma_k$ et aller à 3.3, autrement, sélectionné $j_k \in \gamma$ et calculé

$$\theta_{j_k}^0 \text{ la partie entière de } \min_{i \in I_k} \left\{ \frac{x_{k,i}}{y_{k,ij_k}} > 0 \right\}$$

- Si $\theta_{j_k}^0 = 0$, alors il n'existe pas de solution réalisable entière sur l'arête E_{j_k} mettre $\gamma := \gamma - \{j_k\}$ et aller à (a).
- Autrement, si $\theta_{j_k}^0 \geq 1$ aller à (b).

(b) Si x_k est efficace et $dx_k \geq \phi_{opt}$ alors calculer la valeur du paramètre B_k . Si cette valeur est différente de zéro, alors aller à (c), autrement, mettre $\gamma := \gamma - \{j_k\}$ et aller à (c), si x_k n'est pas efficace où $dx_k < \phi_{opt}$, alors aller à (c) (l'arête E_{j_k} est explorée quelque soit la valeur de β_k).

(c) Explorer l'arête E_{j_k} , cherchant une solution réalisable de $(P_1(D))$ correspondant à θ et tester pour l'efficacité à partir de $\theta = \theta_{j_k}^0$ jusqu'à $\theta = 1$. (θ Est un entier positif) dès que une solution efficace x_k' vérifiant $dx_k' > \phi_{opt}$, remplacer x_{opt} par x_k' et ϕ_{opt} par dx_k' , aller à 3.2. S'il y a aucune solution efficace entière sur l'arête, mettre $\gamma := \gamma - \{j_k\}$ aller à (a).

3.2. Soit $k := k + 1$ défini la nouvelle région tronquée D_k est obtenue comme sous ensemble D_{k-1} en appliquant la coupe $dx \geq \phi_{opt}$ (coupe de type II). utiliser le dual simplexe et coupe de Gomory si nécessaire pour trouver une nouvelle solution x_k faire $x_{opt} := x_k, \phi_{opt} := dx_k$, aller à 3.1.

3.3. Soit $k := k + 1$

La nouvelle région tronquée D_k est obtenue comme sous ensemble D_{k-1} (ou D si $k = 1$) en appliquant la coupe de Dantzig ou coupe de type (I), utiliser le dual simplexe et coupes de Gomory si nécessaire pour trouver une nouvelle solution x_k . Soit $\phi_k := dx_k$, si la solution x_k est efficace et $dx_k > \phi_{opt}$ Faire $x_{opt} := x_k, \phi_{opt} := dx_k$, aller à 3.1. Si non aller à 3 sans la mise à jour de x_{opt} et ϕ_{opt} .

Etape finale : La procédure prend fin, ou bien à la première étape si la solution x^0 est efficace, ou lorsque l'impossibilité des opérations pivots, indiquant que la région courante ne contient aucun point réalisable, la solution optimale est x_{opt} et ϕ_{opt} la valeur du critère ϕ .

3.4 Optimisation d'une fonction linéaire sur un ensemble de solution efficace d'un problème MOILPS

3.4.1 Introduction :

Dans beaucoup de problèmes de décision multicritères quelques paramètres prennent des valeurs incertaines qui ne sont pas connues avec précision au moment de prendre de décision. Cette incertitude dépend du problème d'observation de ses paramètres où leurs valeurs dépendent des facteurs naturels. Si les paramètres sont des variables aléatoires le problème est dit problème de programmation linéaire multi-objective stochastique MOLPS.

Il existe beaucoup de recherche qui sont réalisées dans la littérature, nous mentionnons les articles de Stancu_Minisian et Tigan [47], Stancu_Minisian [46], Urli et Nadeau [54], Ben Abdelaziz, Lang et Nadeau [6]. La solution de tel problème toujours nécessite une transformation du problème initial à un problème déterministe. Cette transformation porte l'utilisation de quelques caractéristiques de variables aléatoires, ce qui mène à de différents concepts de solutions efficaces.

Le rajout de variables entières au problème MOLPS nous donne le problème MOILP et ainsi la notion d'efficacité et la recherche de solutions efficaces deviennent plus difficiles. Mais cela n'est pas notre intérêt. D'autre part, le problème d'optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces du problème MOLP joue un rôle important dans la

décision multicritère, qui est en lui-même un problème d'optimisation globale difficile dont plusieurs optimums locaux ne sont pas globaux, cela est dû à la région réalisable qui est généralement non convexe.

Si on rajoute des variables entières au problème MOLP on aura un problème d'optimisation sur un ensemble de solution efficace de MOILP.

Dans cette partie nous présentons une nouvelle méthode d'optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces de MOILPS. La méthode résout une suite de programmes linéaires en nombre entiers.

On ajoute progressivement des contraintes au problème après transformation du problème original à un problème déterministe équivalent (MOILP), en utilisant la programmation stochastique avec recours à deux niveaux. La fonction linéaire est améliorée progressivement on utilisant des coupes planes jusqu'à ce que le domaine réalisable soit vide. Cette technique fourni une solution optimal et un sous ensemble de solutions efficaces qui améliore itérativement la fonction linéaire.

3.4.2 Modélisation du problème :

Considérons le problème linéaire multio-bjective stochastique en nombres entiers suivant :

$$(P) \left\{ \begin{array}{l} \min Z_k = C_k(\omega)x, \quad k : 1, \dots, p \\ \text{t.q} \quad Ax = b \\ T(\omega)x = h(\omega) \\ x \geq 0, x \text{ entier} \end{array} \right. \quad (3.21)$$

Où C_k, T, h sont des matrices aléatoires de dimension $(1 \times n), (m_0 \times n)$ et $(m_0 \times 1)$, défini sur un espace de probabilité $(\Omega, E, P), \omega \in \Omega$ évènement aléatoire A, b sont des matrices déterministes de dimension respectives $(m \times n), (m \times 1)$.

Le problème déterministe associe comme défini dans le chapitre précédant est :

$$(P_2) \left\{ \begin{array}{l} \min_x \tilde{Z}_k = Z_k + \theta \quad k : 1, 2, \dots, K \\ \text{t.q} \quad x \in D \\ \theta \geq Q(x) \\ x \text{ entier} \end{array} \right. \quad (3.22)$$

Considérons le problème restreint :

$$(P_3) \left\{ \begin{array}{l} \min_x \tilde{Z}_1 = Z_1' + \theta \quad k:1,2,\dots,K \\ t.q \quad x \in D \\ \theta \geq Q(x) \\ x \text{ entier} \end{array} \right. \quad (3.23)$$

$$D = S \cap \mathbb{Z}^n, S = \left\{ x \in \mathbb{R}^n / Ax = b, \sigma_r^T T(\omega^r) x \geq h(\omega^r), r:1,\dots,R; x \geq 0 \right\}$$

$$= \left\{ x \in \mathbb{R}^n / \tilde{A}x = \tilde{b}, x \geq 0 \right\}$$

$$Z_k' = E(Z_k) = \sum_{r=1}^R p^r Z_{k,r}$$

$$Q(x) = E(Q(x, \omega)) = \sum_{r=1}^R p^r Q(x, \omega^r) = \sum_{r=1}^R p^r (q^r)^T y^r$$

$$Q(x, \omega^r) = \min_y \left\{ (q^r)^T y^r / W(\omega^r) y = h(\omega^r) - T(\omega^r) x, x \geq 0 \right\} \quad (3.24)$$

Avec P est la probabilité de réalisation de la variable aléatoire, y est une action de recours et W est la matrice de recours.

Rappelons la définition de solutions efficaces pour le problème (P_2) .

Définition 13:

Un point $x^0 \in S$ est dit efficace pour (P_2) si et seulement s'il n'existe pas un autre point $x^1 \in S$ tel que $\tilde{Z}_k(x^1) \leq \tilde{Z}_k(x^0)$ pour tout $k \in \{1, 2, \dots, p\}$ et $\tilde{Z}_k(x^1) < \tilde{Z}_k(x^0)$ pour au moins un $k \in \{1, 2, \dots, p\}$ et pour toute réalisation $\omega^r, r:1, 2, \dots, R$.

On définit le problème principale par :

$$P(E(P)) \left\{ \begin{array}{l} \min f(x) = dx \\ x \in E(p) \end{array} \right. \quad (3.25)$$

$E(p)$ est l'ensemble des solutions efficaces du problème MOILPS.

3.4.3 Teste d'efficacité :

Le théorème si dessous est utilisé pour tester l'efficacité d'une solution réalisable du problème (P) .

Théorème 13 :

Soit x^* une solution arbitraire de la région D , $x^* \in E(p)$ si et seulement si la valeur optimale de la fonction objective v est nul. Dans le problème linéaire en nombre entier suivant :

$$P(x^*) \left\{ \begin{array}{l} \min v = -\sum_{i=1}^k \psi_i \\ t.q. \\ EC(\omega)x = I\psi + E(\omega)x^* \\ Ax = b \\ x \geq 0, x \text{ entier} \\ \psi_i \in \mathbb{R}^+ \quad \forall i \end{array} \right. \quad (3.26)$$

3.4.4 Notation et résultats préliminaires :

$\tilde{Z}_1, \tilde{Z}_2, \dots, \tilde{Z}_p$ Sont les critères de (P) et f le critère additionnel.

$S_1 = \{x \in \mathbb{R}^{n_1} / \tilde{A}x = \tilde{b}_1, x \geq 0\}$ Est la région tronquée obtenue par des coupes successives de

Gomory $(\tilde{Z}_1^1, \tilde{Z}_2^1, \dots, \tilde{Z}_p^1)$ est le premier p-uplet non dominé correspondant à la solution x_1 .

Pour $k \geq 1$:

$x_k \in \mathbb{Z}^{n_k}$, la solution optimale entière dans D_k .

\tilde{B}_k est la base associée à x_k .

$\tilde{a}_{k,j}$ est le vecteur d'activité de $x_{k,j}$

$$I_k = \{j / \tilde{a}_{k,j} \in B_k\}$$

$$N_k = \{j / \tilde{a}_{k,j} \notin B_k\}$$

$$J_k = \{j \in N_k / \tilde{Z}_{1,j}^k - \tilde{C}_1^1 = 0\} \text{ Avec } \tilde{C}_1 = (EC_1(\omega), 1)$$

$$y_{k,j} = (B_k^{-1})a_{k,j}, y_{k,j} \in \mathbb{R}^{m_k \times 1} \quad \Gamma_k = \{j \in N_k / \tilde{Z}_{1,j}^k - \tilde{C}_j^1 \leq 0 \text{ et } f_j^k - d_j^k \geq 0\},$$

$\tilde{Z}_{1,j}^k = \tilde{C}_{B_k}^1 y_{k,j}$, $\tilde{C}_{B_k}^1$ est le vecteur de coefficient coût des variables de base associées à B_k dans

C^1 et C_j^1 représente la $j^{\text{ème}}$ composante du vecteur C^1 , $f_j^k = d_{B_k} y_{k,j}$; d_{B_k} est le vecteur coût des variables de base associées à B_k dans d .

$$S_k = \left\{ x \in \mathbb{R}^{n_k} / \tilde{A}_k x = \tilde{b}_k, x \geq 0, A \in \mathbb{R}^{m_k \times n_k}, b_k \in \mathbb{R}^{m_k}, (m_k \times n_k) \in \mathbb{N} \times \mathbb{N}, m_k \neq 0, n_k \neq 0 \right\}$$

S_k est la région tronquée obtenue après avoir appliqué la coupe $\sum_{j \in N_{k-1} - \{j_{k-1}\}} x_j \geq 1$,

avec $j_{k-1} \in \Gamma_{k-1}$, ou bien la coupe $dx \leq dx_k$ et les coupes de Gomory si nécessaire, $S_k = D_k \cap Z^{n_k}$.

Le tableau simplexe utilisé à la forme suivante :

Tableau	Variable de base	x_1	x_2	...	x_{n_k}
x_{B_k}	$x_{k,1}$	t_{11}	t_{12}	...	t_{1n_k}
	$x_{k,2}$	t_{21}	t_{22}	...	t_{2n_k}

	x_{k,m_k}	$t_{m_k,1}$	$t_{m_k,2}$...	t_{m_k,n_k}
Coût réduit	\tilde{Z}_p^1	$\tilde{Z}_{1,1}^k - \tilde{C}_1^1$	$\tilde{Z}_{1,2}^k - \tilde{C}_2^1$...	$\tilde{Z}_{1,n_k}^k - \tilde{C}_{n_k}^1$

	\tilde{Z}_p^k	$\tilde{Z}_k^k - \tilde{C}_1^p$	$\tilde{Z}_{p,2}^k - \tilde{C}_2^p$...	$\tilde{Z}_{p,n_k}^k - \tilde{C}_{n_k}^p$
	f^k	$f_1^k - d_1^k$	$f_2^k - d_2^k$...	$f_{n_k}^k - d_{n_k}^k$

Tableau simplexe générique pour notre problème

$\tilde{Z}_{1,j}^k - \tilde{C}_j^1, f_j^k - d_j^k$ Sont des coûts réduits

Dans le tableau on a aussi :

$$t_{ij} = \begin{cases} y_{k,ij} & \text{si } j \in N_k \\ (e_j)_i & \text{si } j \in I_k \end{cases} ;$$

e_j est la $j^{\text{ème}}$ colonne de la matrice identité de dimension $m_k \times m_k$

3.4.5 Solution de la méthode :

Etape 1. Poser $\theta = -\infty, f = +\infty$, puis résoudre le problème linéaire

$$(1) \begin{cases} \min EC_1(\omega)x \\ t.q \\ Ax = b \\ x \geq 0, x \text{ entier} \end{cases}$$

Sans coupe de faisabilité et d'optimalité : Soit x_1 une solution optimale entière.

1) tester la faisabilité pour le deuxième niveau, pour chaque réalisation $r : 1, \dots, R$

Résoudre le problème :

$$(2) \begin{cases} \max \sigma^T (h(\omega^r) - T(\omega^r)x_1) \\ \text{t.q.} \\ \sigma^T W \leq 0 \\ \|\sigma\|_1 \leq 1 \end{cases}$$

Si $\exists r : 1, 2, \dots, R$ tel que $\sigma^T h(\omega^r) - T(\omega^r)x_1 > 0$, introduire la coupe de faisabilité $\sigma^T [h(\omega^r) - T(\omega^r)x] \leq 0$, (vérifier dans ce cas la faisabilité), aller à b

2) tester l'optimalité de x_1 , soit à calculé $Q(x_1) = \sum_{r=1}^R P^r [\pi^T [h(\omega^r) - T(\omega^r)x_1]]$

Si $\alpha \geq Q(x_1)$, alors x_1 est optimale. Autrement si $\alpha < Q(x_1)$, introduire la coupe

d'optimalité $\alpha \geq \sum_{r=1}^R P^r [\pi^T [h(\omega^r) - T(\omega^r)x]]$

Optimiser le problème (1) pour avoir une solution optimale entière, ajouté éventuellement des coupes de Gomory si nécessaire. Aller à a.

a. construire $J_1 = \{j \in N_1 / \tilde{Z}_{1,j}^1 - \tilde{C}_j^1 = 0\} = \emptyset$.

a1) si $J_1 = \emptyset$ alors x^1 est unique est efficace.

a2) Si $J_1 \neq \emptyset$ alors x^1 peut ne pas être unique. Tester son efficacité par le teste d'efficacité. Si x^1 est non efficace, alors aller a l'étape 2, si non, soit $f^1 = x_1, f_{opt} = f^1, x_{opt} = x_1$, aller à l'étape 2.

Etape 2 : Soit $k = 1$

2.1 construire l'ensemble $\Gamma_k = \{j \in N_k / \tilde{Z}_{1,j}^k - \tilde{C}_j^1 \leq 0 \text{ et } f_j^k - d_j^k \geq 0\}$.

Si $\Gamma_k = \emptyset$, aller à (2.3). Autrement, soit $\gamma = \Gamma_k$ aller à (A)

A. Si $\gamma = \emptyset$, soit $j_k \in \Gamma_k$ aller à (2.3), autrement sélectionné $j_k \in \gamma_k$ et calculer $\alpha_{j_k}^0$ la

partie entier de $\min_{i \in I_k} \left\{ \frac{x_{k,i}}{y_{k,ij_k}} > 0 \right\}$.

- Si $\alpha_{j_k}^0 = 0$ pas de solutions réalisables sur l'arête.

$$E_{j_k} = \left\{ x = (x_i) \in S / \begin{array}{l} x_i = x_i - \alpha_{j_k} y_{k,ijk}, i \in I_k \\ x_{j_k} = \alpha_{j_k} \\ x_s = 0 \text{ pour tous } s \in N_k - \{j_k\} \end{array} \right\}$$

Avec $0 < \alpha_{j_k} \leq \min_{i \in I_k} \left\{ \frac{x_{k,i}}{y_{k,ijk}} > 0 \right\}$, Mettre $\gamma := \gamma - \{j_k\}$, aller à (A).

- Autrement, si $\alpha_{j_k}^0 \geq 1$, alors aller à (B).

B. si x_k est efficace et $dx_k \leq f_{opt}$; calculer la valeur de $B_k = \left(d_{j_k} - \sum_{i \in I_k} d_j \times y_{k,ijk} \right)$.

Si cette valeur est différente de zéro, aller à (C) Autrement, mettre $\gamma := \gamma - \{j_k\}$, aller à (A). Si x_k n'est pas efficace où $dx_k > f_{opt}$ aller à (C).

C. Explorer l'arête E_{j_k} est cherché une première solution réalisable entière. Correspondante a α . Pour $\alpha = 1$ jusqu'à $\alpha = \alpha_{j_k}$. Tester la faisabilité de deuxième niveau, tester son optimalité et tester l'efficacité de la première solution trouvée.

- Soit x'_k la solution efficace, alors soit $x_{opt} = x'_k, f_{opt} = dx'_k$ aller à (2.2).
- Si pas de solution efficace sur l'arête E_{j_k} , mettre $\gamma := \gamma - \{j_k\}$ aller à (A).

2.2 : $k = k + 1$; ajouté la coupe $dx \leq dx'_{k-1}$ pour définir la nouvelle région tronqué, appliqué le dual simplexe, ajouté des coupe de Gomory si nécessaire, pour avoir une nouvelle solution réalisable, testé la faisabilité et l'optimalité de x_k et testé son efficacité.

Faire $f^k = dx_{opt}$, Si $f^k < f_{opt}$ faire $x_{opt} = x_k$. Aller à 2.1.

2.3 $k = k + 1$: appliquer la coupe de Dantzig où coupe de type 1, appliquer Dual simplexe et ajouté éventuellement des coupes de Gomory si nécessaire, soit x_k la nouvelle solution réalisable.

Tester la faisabilité et l'optimalité pour le deuxième niveau.

Tester son efficacité, Faire $f^k = dx_k$, si $f^k < f_{opt}$ faire $f^k x_k, x_{opt} = x_k$ aller à (2.1)

Etape finale :

La procédure se termine lorsque, les opérations pivots seront impossibles, indiquant que la région de faisabilité est vide.

Conclusion et perspectives

Nous avons considéré un problème d'optimisation d'une fonction linéaire sur l'ensemble des solutions efficaces d'un problème stochastique linéaire multi-objectif en nombre entier (MOILPS). Après avoir transformé le problème stochastique en un problème déterministe on utilise la programmation stochastique avec recours, le problème devient un problème d'optimisation d'une fonction linéaire sur un ensemble des solutions efficaces d'un problème linéaire multi-objectif en nombre entier (MOILP). Utilisons la programmation linéaire classique, la méthode évite l'énumération de toutes les solutions efficaces. Nous fixons comme perspectives une création de base de données de Benchmark qui nous permis de tester l'efficacité de notre proposition et définir des domaines stochastiques qui peuvent être utile pour éventuelle résolution lorsque le critère principale n'est pas linéaire.

Bibliographie:

- [1] M. Abbas, F. Bellahcene Cutting plane method for multiple objective stochastic integer linear programming, *European Journal of Operational Research* 168 (2006) 967–984.
- [2] M. Abbas, D. Chaabane, An algorithm for solving multiple objective integers linear Programming problems, *RAIRO* 36 (2002) 351-364.
- [3] M. Abbas, D. Djamel, optimizing a linear function over the efficient set, *Journal of Operational Reachearch* 174(2006) 1140-1161.
- [4] M. Abbas, M. Moulai, Solving multiple objective integer programming problems, of the Italian Operation Research Society *Recera Operative* 29(89) (1999) 15-39.
- [5] Beale, E. M. L. (1955). On minimizing a convex function subject to linear inequalities.
Journal of the Royal Statistical Society, 17B, 173–184.
- [6] F. Ben Abdelaziz, P. Lang, R. Nadeau 1999, Dominance and efficiency in multicriteria decision under uncertainty. *Theory and decision* 47,191-211.
- [7] J.F. Benders, Partitioning procedures for solving mixed-variables programming problems, *Numerische Mathematik* 4 (1962) 238–252.
- [8] H.P. Benson, Optimization over the efficient set, *Journal of Mathematical Analysis and Applications* 98 (1984) 562-580.
- [9] H.P. Benson, A finite, nonadjacent extreme-point search algorithm for optimizing over the efficient set *Journal of Optimization Theory and Applications* 73(1992) 47-64.
- [10] H.P. Benson, S. Sayin, A face search heuristic algorithm for optimizing over the efficient set, *Naval Research Logistics* 40 (1993) 103-116.
- [11] H.P. Benson, S. Sayin, Optimization over the efficient set: Four special cases, *Journal of Optimization Theory and Applications* 80 (1994) 3-18.
- [12] S. Bolintineanu, Minimization of a quasi-concave function over an efficient set *Mathematical Programming* 61 (1993) 89-110.
- [13] Bowman V.J., On the relationship of the Tchebycheff Norm an the Efficient Frontier of Multiple-Criteria Objectives, in Thirez H. Zionts S. (eds), *MCDM*, Springer-Verlag, Berlin (1976), 76-85.

- [14] R.L. Carraway, T.L. Morin, and H. Moskowitz. Generalized dynamic programming for multicriteria optimization. *European Journal of Operational Research*, 44 :95–104, 1990.
- [15] C.A. Coello Coello. Using a min-max method to solve multiobjective optimization problems with genetic algorithms. In *IBERAMIA'98*, pages 303–314, 1998. LNCS vol 1993, Springer Verlag.
- [16] Dantzig, G. B. (1955). Linear programming under uncertainty. *Management Science*, 1, 197–206.
- [17] J.P. Dauer, Optimization over the efficient set using an active constraint approach, *ZOR-Methods and Models of Operations Research* 35 (1991) 185-195.
- [18] J.P. Dauer, T.A. Fosnaugh, Optimization over the efficient set, *Journal of Global Optimization* 7 (1995) 261-277.
- [19] F. Degoutin and X. Gandibleux. Un retour d'expérience sur la résolution de problèmes combinatoires bi-objectifs. *Programmation Mathématique MultiObjectif (PM2O)*, mai 2002.
- [20] M.I. Dessouky, M. Ghiassi, W.J. Davis, Estimates of the minimum nondominated criterion values in multiplecriteria decision making, *Engineering Costs and Production Economics* 10 (1986) 95-104 [16] J.G. Ecker, I.A. Kouada, Finding efficient extreme points for multiple objective linear programming, *Mathematical Programming* 14 (1978) 249-261.
- [21] J.G. Ecker, I.A. Kouada, Finding efficient points for linear multiple objective programs, *Mathematical Programming* 8 (1975) 375-377
- [22] J.G.Ecker, H.G.Song , optimizing a linear function over the efficient set ,*Journal of optimization Theory and Applications* 83 (3) (1994) 541-563.
- [23] F.Y. Edgeworth. *Mathematical Physics*. P. Keagan, London, 1881.
- [24]M. Ehrgott and X. Gandibleux. A survey and annotated bibliography of multiobjective combinatorial optimization. *OR specktrum*, 22 :425–460, 2000.
- [25] J. Fulöp, Cutting plane method for linear optimization over the efficient set, S. Komlosi, T. Rapcsak, S. Schaible (Eds.), *Generalized Convexity*, Berlin, 1994, pp. 374-385.
- [26] Geoffrion A.M., Solving Bicriterion Mathematical Programs, *Operations research* 15, (1967), pp. 39-54.
- [27] R.Gupta, R. Malhotra, Multi-criteria Integer linear programming problem, *Cahier de CERO* 34 (1992), 51-68.

- [28] Y. Haimes, L. Ladson, and D. Wismer. On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Transactions on System, Man and Cybernetics*, 1 :296–297, 1971
- [29] A. Hertz, B. Jaumard, C.C. Ribeiro, and W.P. Formosinho Filho. A multi-criteria tabu search approach to cell formation problems in group technology with multiple objectives. *RAIRO Operations Research*, 28(3) :303–328, 1994.
- [30] T.P. Hong, H. Wang, and W. Chen. Simultaneously applying multiple mutation operators in genetic algorithms. *Journal of heuristics*, 6 :439–455, 2000.
- [31] R. Horst, N.V. Thoai, Utility function programs and optimization over the efficient set in multiple objective decision making, *Journal of Optimization Theory and Optimizations* 92 (1997) 605-631.
- [32] H. Isermann, The enumeration of the set of all efficient solutions for a linear multiple objective program, *Operational Research Quarterly* 28 (1977) 711-725.
- [33] H. Isermann, R.E. Steuer, Computational experience concerning payoff tables and minimum criterion values over the efficient set, *European Journal of Operational Research* 33 (1987) 91-97.
- [34] P.Kall, W.Wallace, *Stochastique Programming*. Willy Interscience Series in Systems and Optimization, 1994.
- [35] D. Klein , E. Hannan ,an algorithm for the multiple objective integer linear programming problem , *Europeen journal of operational research* 9(4) (1982) 378-385.
- [36] S. Mardle, S. Pascoes, and M. Tamiz. An investigation of genetic algorithm for the optimization of multi-objective fisheries bioeconomic models. *International Transaction of Operation research*, 7 :33–49, 2000.
- [37] T. Murata and H. Ishibuchi. A multi-objectives genetic loacal search algorithm and its application flow-shop scheduling. *IEEE Transaction System*, 28(3) : 392–403, 1998.
- [38] N.C. Nguyen, An algorithm for optimizing a linear function over the efficient set, *Konrad -Zuse–Zentrum fur informationtechnik Berlin*, November 1992.
- [39] G.L. Nemhauser, L.A. wolsey, *Integer Combinatorial optimisation*. Wiley Interscience Series in discrete Mathematics and Optimisation, 1988.
- [40] V. Pareto. *Cours d'économie politique*. Rouge, Lausanne, 1896.

- [41] J. Philip, Algorithms for the vector maximization problem, *Mathematical Programming 2* (1972) 207-229.
- [42] A. Przybylski, X. Gandibleux, and M. Ehrgott. Seek and cut algorithm computing minimal and maximal complete efficient solution sets for the biobjective assignment problem. In 6th Int. Multi-Objective Programming and Goal Programming conf (MOPGP'04), 2004.
- [43] G.R. Reeves, R.C. Reid, Minimum values over the efficient set in multiple objective decision making, *European Journal of Operational Research* 36 (1988) 334-338.
- [44] E. Sandgren. *Advances in design optimization*. Chapman and Hall, 1994. chapter Multicriteria design optimization by goal programming.
- [45] R.E. Steuer, *Multiple Criteria Optimization, Theory, Computation and Application*, Wiley, New York, 1986.
- [46] I.M .Stancu –Minisian., 1984,*Stochastic programming with multiple objective functions* ,D.Reidel Publishing Company,Dordrecht.
- [47] I.M..Stancu- Minisian, S.Tigans . 1984. The vectorial minimum risk problem, In *proceeding of the Colloquim of Approximation and Optimisation* .Cluj-Napoca,pp.321-328.
- [48] R.E. Steuer, *Multiple Criteria Optimization, Theory, Computation and Application*, Wiley, New York, 1986.
- [49] N. Srinivas and K. Deb. Multiobjective optimization using nondominated sorting in genetics algorithms. *Evolutionary Computation*, 2(3) :221–248, 1994.
- [50] J.Sylva, A.Crema . A method for finding the set of non dominated vectors for Multiple objective integer linear programs, *European Journal of Operational research* , 2003),in press.
- [51] H.A.TAHA, *integer programming theory, Application and Computation* Academy Press, New York (1975).
- [52] Teghem J ,Dufrane D.,Thauvoye M. et Kunsch P.L.*STRANGE:an interactive method for Multiobjective linear programming under uncertainty*, *European Journal of Operational Research*,26(1),65-82.
- [53] J.Thegem , P.L. Kunsh, A survey of techniques to determine the efficient solutions to multi-objective integer linear programming , *Asia passific Journal of Operational Research* 3 (1986) 95-108

- [54] Teghem J.(1990), STRANGE MOMIX an interactive method for mixed integer linear programming in R. Slowinski and J.Teghem (eds), stochastic versus Fuzzy Approches to Multiobjective Mathematical programming Under Uncertainty. Dordrecht:Kluwer Academic Publisheres ,101-115.
- [55] B. urli, R.Nadeau PROMISE/ scenarios: An interactive method for multiobjective stochastic programming under partial uncertainty. European journal of operational research 155(2004) 361-372.
- [56] B.Villareal, M.H. Karwan, Multicriteria integer programming: A (hybrid) dynamic programming recursive approach .mathematical programming 21(1981) 204-223.
- [57] E.L. Ulungu and J. Teghem. The two phases method : An efficient procedure to solve biobjective combinatorial optimization problems. Foundation of computing and decision science, 20 :149–156, 1995.
- [58] H.R. Weistroflerr, Careful usage of pessimistic values is needed in multiple objective optimizations, Operations Research Letters 4 (1985) 23-25.
- [59] D.J. White, The maximization of a function over the efficient set via a penalty function approach, European Journal of Operational Research 94 (1996) 143-153.
- [60] P.L. Yu, M. Zeleny, The set of all non dominated solutions in linear cases and a multicriteria simplex method, Journal of Mathematical Analysis and Applications 49 (1975) 430-468.