

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA  
RECHERCHESCIENTIFIQUE



UNIVERSITE MOULOU D MAMMERI DE TIZI-OUZOU

ⵍⵓⵎⵓⵝ ⵎⵎⵎⵔⵉ ⵔⵉⵣⵉⵣⵓⵣⵓ



FACULTE DES SCIENCES  
DEPARTEMENT MATHEMATIQUE

# Mémoire de fin d'étude

Pour l'obtention du diplôme de MASTER en Mathématiques  
Appliquées à la Gestion

## Thème:

Programmation multi-objectifs et études des cas  
réels

Présenté par :

MEDJAHED Safia

BASSAID Rebiha

Devant le jury d'examen composé de :

*M<sup>me</sup> F. LESLOUS*

*M<sup>r</sup> M. GOUBI*

Dirigé par :

*M<sup>r</sup> M. CHEBBAH*

Année universitaire : 2020/2021



## *Remerciement*

*Nous remercions le **DIEU** le tout puissant d'avoir guidé nos pas vers les portes de savoir tout en illuminant notre chemin et de nous avoir donné suffisamment de courage et de persévérance pour mener notre travail à terme.*

*Nous tenons à remercier **Mr CHEBAH**, notre promoteur, de nous avoir encadré, nous lui exprimons notre gratitude pour sa disponibilité, ses conseils qui étaient notre guide vers une vision plus étendue et le soutien qu'il nous a apportée tout au long de la préparation de ce mémoire.*

*Nous vifs remerciements s'adressent au membre de jury qui ont aimablement accepté de nous honorer en faisant ce mémoire pour évaluer et apprécier notre travail, nous espérons qu'ils seront satisfaits.*

*Nous apportons aussi nos vifs remerciements aux personnels enseignants de l'Université Mouloud Mammeri de Tizi-Ouzou  
A toute la promotion de 2020/2021.*

## Dédicaces

*Je dédie ce modeste travail :*

*A la mémoire de celui sans qui je ne serais jamais arrivé à ce que je suis aujourd'hui, je cite mon cher **papa MOHAMED** (que dieu l'accueille dans son vaste paradis) qui était derrière moi tout au long de mon parcours, j'aurais aimé qu'il soit là aujourd'hui à mes côtés pour l'honorer mais le destin a décidé autrement, à ma chère **maman MALIKA**, à mes chères frères **FARID, KAMEL, HAKIM** pour leur tendresse, leurs conseils, leurs sacrifices, leur présence à mes côtés et leur soutien, auxquels je dois ce que je suis et à qui je souhaite une longue vie pleine de bonheur, à ma chère sœur **ROSA** à qui je souhaite la réussite, à mes petites amies **LALA, MILOU** et **IMENE**, **AKSEL**, à ma chère belle-sœur **DAHIBIA**.*

*A mes chères tantes, mes chers oncles, mes chers cousins et cousines.*

*A mes chères amies : **Sylia, Fazia, Lydia, Djamila, Samia, Zazi, Sara, Kahina, Nadia, Nawel, Assia, tawes.***

*Safia*



## Dédicaces

*Je rends grâce à dieu de m'avoir donné le courage et la volonté ainsi que la conscience d'avoir pu terminer mes études.*

*Je dédie ce modeste travail :*

*A mes très chères mère et père pour toutes leurs tendresses et pour leurs nombreux sacrifices que Dieu les gardes.*

*A toute la famille spécialement mes frères Achour et sa femme Lynda, Hamid et sa femme Fadhila, Nabil, Smail, Nordine.*

*A mes sœurs Djamila et son mari Rachid, Nora et son mari Mustapha.*

*Au anges de la famille Tarek, Anella, Meriem, Ghiles, Anis, Melina, Amina, Massi, Khadidja, Ahmed.*

*A tous mes cousins et cousines.*

*A mes amies : Fazia, Sylvia, Lydia, Rachida, Nawal.*

*A tous les professeurs et enseignants qui ont collaboré à ma formation depuis mon cycle d'étude.*

## Rebiha

# Table des matières

Introduction générale : .....	1
Chapitre 1: Etude sur la programmation linéaire	
Introduction: .....	4
1.1. Techniques de la programmation linéaire:.....	4
1.2. Formulation d'un programme linéaire:.....	5
1.3. Les différentes formes d'un programme linéaire:.....	7
1.3.1. Forme générale: .....	7
1.3.2. Forme canonique: .....	7
1.3.3. Forme standard: .....	8
1.3.4. Forme mixte:.....	9
1.4. Méthodes de résolutions: .....	9
1.4.1. La méthode de simplexe: .....	9
Critère d'optimalité:.....	11
1.4.2. La M-Méthode: .....	15
1.4.3. La méthode des deux phases:.....	18
1.4.4. Dualité:.....	22
Méthode de construction du dual .....	22
1.4.5. La méthode de dual simplexe: .....	23
Algorithme de dual simplexe .....	24
Chapitre 2: Etude sur la programmation linéaire en nombres entiers	
Introduction: .....	29
2.1. Forme générale d'un programme linéaire en nombres entiers "PLNE": .....	29
2.2. Méthodes de résolution d'un programme linéaire en nombres entiers: .....	29
2.2.1. Méthode « Branch and Bound » (Méthode par séparation et évaluation): .....	29
Principes de la méthode: .....	30
Etapas de résolution : .....	30
Exemple 2.1 .....	32
2.2.2. Méthode de Balas : .....	34
Chapitre 3: Etudes sur la programmation linéaire multi-objectifs	
3.1. Concepts de base:.....	40
3.2. Définitions .....	42
3.3. Méthodes de résolution d'un programme linéaire multi-objectifs.....	51
3.3.1. Méthode de simplexe multi-objectifs: .....	51

Le test d'efficacité: .....	53
Résumé de la méthode: .....	55
Conclusion: .....	60
3.3.2. Méthode de la somme pondérée ou méthode d'agrégation des objectifs (méthode d'agrégation par pondération):.....	60
3.3.3 méthode $\varepsilon$ -contraintes:.....	65
3.3.4 La méthode « but à atteindre » :.....	67
Conclusion: .....	68
Chapitre 4: Etudes des cas réels en programmation linéaire multi-objectifs	
Introduction: .....	70
4.1 Premier cas : Problème de production: .....	70
4.2 Deuxième cas : Problème de mélange alimentaire : .....	80
4.2.1 Formulation du problème: .....	80
4.2.2 Exemple : .....	82
4.3. Troisième cas: Problème de transport multicritères(MOTP) :.....	83
4.3.1 Formulation du problème: .....	83
4.3.2 Exemple: .....	83
Conclusion: .....	89
Chapitre 5: Partie informatique (Logiciel LINGO)	
5.1.1 Introduction: .....	91
5.1.2 Installation de logiciel:.....	91
5.1.3 Présentation de logiciel « LINGO »: .....	91
5.1.4 Interface de logiciel: .....	93
5.1.5 Les étapes de programme: .....	96
5.1.6 Exemples d'application: .....	97
Conclusion générale :.....	107
Bibliographie: .....	108

## Liste des figures :

Figure 1:Principes de la programmation linéaire .....	5
Figure 2: L'organigramme de la méthode de simplexe (maximisation).....	12
Figure 3: L'organigramme de la méthode des deux phases.....	19
Figure 4: Représentation de l'espace de décision et l'espace des objectifs correspondant ....	43
Figure 5: solution efficace et point non dominé .....	45
Figure 6: Le point idéal.....	45
Figure 7: Le point anti-idéal .....	46
Figure 8: Front de pareto.....	47
Figure 9: exemple .....	47
Figure 10:Point Nadir .....	48
Figure 11: Espace convexe et espace non convexe .....	49
Figure 12: Tableau 1 de simplexe multi-objectifs .....	52
Figure 13 :la méthode de $\varepsilon$ -contraintes pour un problème de maximisation.....	66
<i>Figure 14:la barre des menus de logiciel .....</i>	<i>93</i>
<i>Figure 15:la barre des outils de LINGO .....</i>	<i>94</i>
<i>Figure 16:page d'accueil de logiciel .....</i>	<i>96</i>
<i>Figure 17:ouverture d'une nouveau espace de travail.....</i>	<i>96</i>
<i>Figure 18 :l'exécution de modèle linéaire .....</i>	<i>97</i>
<i>Figure 19 : la saisie de notre modèle linéaire .....</i>	<i>98</i>
<i>Figure 20 : l'écran d'affichage en cliquant sur « Solve » .....</i>	<i>99</i>
<i>Figure 21 :la saisie du modèle avec des variables d'écart .....</i>	<i>100</i>
<i>Figure 22 : l'exécution de modèle avec « Solve » .....</i>	<i>101</i>
Figure 23: la saisie de modèle.....	102
Figure 24: l'exécution de premier critère de modèle avec solve.....	103
Figure 25: la saisie de modèle.....	104
Figure 26: l'exécution de deuxième critère avec solve .....	105

# **Introduction générale**

### **Introduction générale :**

L'une des principales missions pour lesquelles la recherche opérationnelle s'est vouée est l'aide à la décision et à la gestion.

Depuis les années 70, les activités de recherche en Recherche Opérationnelle au niveau mondial n'ont cessé de se développer tant au niveau de ses concepts théoriques et de l'amélioration techniques de ses outils d'optimisation qu'au niveau applicatifs où elle intervient de manière cruciale dans des secteurs de plus en plus nombreux et diversifiés comme : « la production industrielle, la planification, le transport, l'informatique, les télécommunications et l'énergie, mais aussi dans les banques et les assurances...etc. Les modèles traditionnels développés dans le cadre des méthodes quantitatives de gestion considéraient en général un critère unique, pour lequel il existe une solution optimale. Les algorithmes mis au point consistaient alors à définir un moyen d'atteindre le plus rapidement possible une telle solution. Cependant, dans de nombreux cas, cette modélisation ne traduit pas exactement la réalité.

La plupart des problèmes réels intervenant en mathématiques de décision sont de nature qui impose la prise en compte de plusieurs critères qui sont souvent antagonistes. Tout décideur est obligé de tenir compte du maximum d'éléments en sa possession pour aboutir à la meilleure décision possible.

Ainsi pour mieux appréhender la réalité, l'approche multicritère devient incontournable. Il est utile dans ce cas de définir un concept d'optimalité, d'étudier les propriétés et les conditions d'existence des solutions et de déterminer les méthodes pratiques de recherche des décisions relatives à ce concept d'optimalité.

Dans un problème multicritère, l'ordre introduit sur l'espace des critères est partiel, ce qui traduit souvent l'impossibilité de comparer les solutions entre elles. L'ensemble des points de recherche tels qu'il n'existe aucun point qui est strictement meilleur que tous les autres simultanément sur tous les critères est appelé « front de Pareto » du problème. Il s'agit de l'ensemble des meilleurs « compromis » réalisables entre les critères. Ainsi, le but de l'optimisation dans ce cas est d'identifier cet ensemble de compromis.

La première notion d'optimalité en multi objectif a été introduite par Edgeworth en 1881. Elle a été utilisée de manière plus formelle par l'économiste italien Pareto[9]. Cette notion est appelée efficacité, optimalité selon Pareto ou encore non dominance.

Le but de ce travail est d'essayer de faire une synthèse sur la programmation mathématique multi-objectif, de voir quelques méthodes de résolution. Pour cela nous avons structuré notre mémoire comme suit:

Le premier chapitre est consacré à la programmation linéaire uni-critère dans le cas général, aux différentes méthodes de résolution d'un problème linéaire.

Au deuxième chapitre on s'intéressera à la programmation linéaire en nombres entiers et on va présenter deux méthodes de résolutions ; la méthode de Branch and Bound et la méthode de Balas.

Au chapitre 3, nous présentons l'optimisation multi-objectif. Nous introduisons des concepts fondamentaux tels que la formulation du problème, le concept de dominance et d'efficacité, point idéal, point Nadir...Etc. Nous décrivons aussi les principales approches de résolution pour ces problèmes comme la méthode du simplexe multi-objectifs, la méthode d'agrégation...etc.

Au quatrième chapitre, nous allons étudier quelques cas réels de l'optimisation multi-objectifs tel que les cas de Problème de production, le problème du mélange alimentaire, le problème de transport bi-objectifs ...etc.

Le chapitre 5 traite la partie implémentation et description générale du logiciel (LINGO) ainsi que la présentation de quelques résultats obtenus.

Enfin, nous terminons notre travail par une conclusion générale et la bibliographie utilisée.

# **CHAPITRE 1:**

## **Etude sur la programmation linéaire**

## **Introduction:**

La programmation linéaire est une technique de recherche opérationnelle qui a pour objet la modélisation et la conception d'algorithmes de résolution des problèmes économiques ou autres, trois types de problèmes relèvent de la programmation mathématique:

- La programmation linéaire, où les fonctions sont linéaires.
- La programmation quadratique.
- La programmation non linéaire, où une partie des fonctions données sont représentées sous la forme de fonctions non linéaires.

Le problème de la programmation linéaire consiste à étudier les méthodes de recherche de la valeur maximale ou minimale d'une fonction linéaire en présence de certaines contraintes linéaires.

La fonction objective dont on cherche la valeur maximale ou minimale est appelée fonction économique, tandis que l'ensemble de valeurs des variables qui optimisent (maximisent, minimisent) cette fonction définit l'ainsi appelé plan (solution) optimale.

Tout autre ensemble de valeurs vérifiant les contraintes imposées constitue un plan de solution admissible.

## **1.1. Techniques de la programmation linéaire:**

La mise en œuvre de la technique de la programmation linéaire peut être subdivisée en Cinq étapes:

### **Etape 1:**

Identification de problème comme étant solvable par la programmation linéaire, cette identification est le fruit d'une expérience dans la modélisation mathématique des problèmes.

### **Etape 2:**

Formulation de problème réel avec utilisation d'un modèle mathématique linéaire, elle se fait en collaboration avec le décideur faisant le problème.

### **Etape 3:**

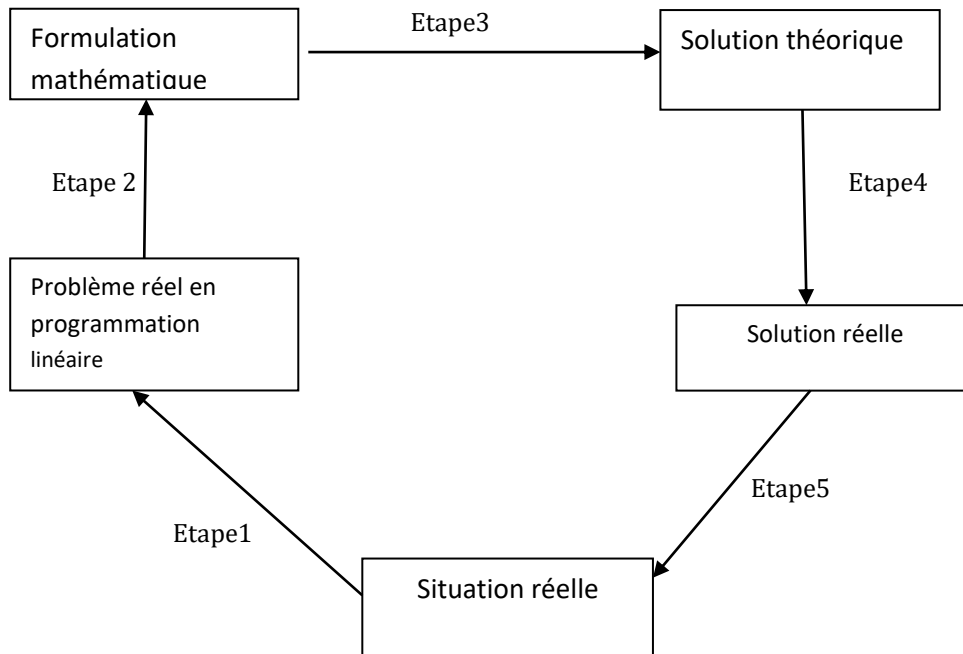
Résolution de problème théorique en utilisant des techniques algorithmiques.

### **Etape 4:**

Détermination d'une solution réelle à partir de la solution théorique.

**Etape 5:**

Vérification de la validité de la solution et modification de la formulation mathématique si nécessaire, cette étape permet d'affirmer le modèle afin d'apporter une solution acceptable pour le décideur.



**Figure 1:Principes de la programmation linéaire**

**1.2. Formulation d'un programme linéaire:**

Un problème linéaire PL est un problème d'optimisation qui consiste à maximiser ou minimiser une fonction objectif linéaire à n variables soumises à un ensemble de contraintes exprimées sous forme d'équations ou d'inéquations linéaires.

Il s'agit de chercher les variables  $x = (x_1, x_2, \dots, x_j, \dots, x_n)$  qui optimisent (maximisent /minimisent) une fonction linéaire  $Z$  qui dépend des coûts  $c = (c_1, c_2, \dots, c_j, \dots, c_n)$ .

$$Z = c_1x_1 + c_2x_2 + \dots + c_jx_j + \dots + c_nx_n = \sum_{j=1}^n c_jx_j \rightarrow (\max/ \min) [1]$$



On écrit souvent  $A$  de la manière suivante:

$A = (a_1, a_2, \dots, a_j, \dots, a_n)$ , Où  $a_j$  est une vecteur colonne:

$$a_j = A(I, j) = \begin{pmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{ij} \\ \vdots \\ a_{mj} \end{pmatrix}$$

Avec ces nouvelles notations, notre problème linéaire peut être écrit sous forme matricielle suivante:

$$PL \left\{ \begin{array}{l} Z = Z(x) = c'x \rightarrow \max \\ s.c \\ Ax = (\leq, \geq)b \\ x \geq 0 \end{array} \right.$$

Ici  $c'$  est la transposé de  $c$ , la matrice  $A$  est la matrice de condition du problème et  $b$  vecteur (second membre).

### 1.3. Les différentes formes d'un programme linéaire:

Le programme linéaire apparaît généralement sous l'une des trois formes suivantes:

#### 1.3.1. Forme générale:

$$PL = \left\{ \begin{array}{l} (\min/\max Z(x)) = \sum_{j=1}^n c_j x_j \\ \sum_{j=1}^n a_{ij} x_j \leq (\geq, =) b_i, i = \overline{1.m} \\ x_j \geq 0, \forall j = \overline{1.n} \end{array} \right.$$

#### 1.3.2. Forme canonique:

$$PL \left\{ \begin{array}{l} (\min/\max) Z(x) = c'x \\ s.c \\ Ax \geq (\text{resp } \leq) b \\ x \geq 0 \end{array} \right.$$

Deux propriétés caractérisent la forme canonique:

- Toutes les variables  $x = (x_1, x_2, \dots, x_j, \dots, x_n)'$  sont astreintes à être positives ou nulles.
- Toutes les contraintes sont des inéquations.

**Exemple:**

$$(PL) \begin{cases} \max W = 3x_1 - 4x_2 \\ x_1 - x_2 \leq 3 \\ 3x_1 + x_2 \leq 8 \\ x_1 \geq 0; x_2 \geq 0 \end{cases}$$

Le problème peut s'écrire sous forme matricielle de la manière suivante:

$$(PL) \begin{cases} \max W = c'x \\ Ax \leq b \\ x \geq 0 \end{cases} \quad \text{Ou} \quad c = (3, -4) ; A = \begin{pmatrix} 1 & -1 \\ 3 & 1 \end{pmatrix} ; x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} ; b = \begin{pmatrix} 3 \\ 8 \end{pmatrix}$$

### 1.3.3. Forme standard:

La forme standard est donnée par:

$$(PL) \begin{cases} (\min/\max)Z = c'x \\ s.c \\ Ax = b \\ x \geq 0 \\ b \geq 0 \end{cases}$$

Deux propriétés caractérisent la forme standard:

- Toutes les variables  $x_1, x_2, \dots, x_n$  sont astreintes à être positives ou nulles.
- Toutes les contraintes essentielles sont des équations.

**Exemple:**

$$(PL) \begin{cases} \max W = 3x_1 - 4x_2 \\ x_1 - x_2 + x_3 = 3 \\ 3x_1 + x_2 + x_4 = 8 \\ x_1 \geq 0; x_2 \geq 0; x_3 \geq 0; x_4 \geq 0 \end{cases}$$

**1.3.4. Forme mixte:**

Sous cette forme certaines contraintes sont des équations tandis que d'autres sont des inéquations. On utilise rarement cette forme pour la résolution des problèmes.

**Théorème:**

Tout programme linéaire sous forme canonique peut être écrit sous forme standard, tout programme linéaire sous forme standard peut être écrit sous forme canonique.

**Définition:**

Une solution de base réalisable est dite non-dégénérée si  $x_j > 0, j \in J_B$ .

**Remarque:**

Les problèmes de minimisations sont en fait équivalents puisque  $\min Z = -\max(-Z)$  et  $\max Z = -\min(-Z)$

**1.4. Méthodes de résolutions:**

Dans cette partie, nous allons nous intéresser aux méthodes de résolution d'un programme linéaire à variables continues. Soit (PL) un programme linéaire défini comme suit:

$$(PL) \begin{cases} \max Z = c'x \\ Ax = b \\ x \geq 0 \end{cases}$$

**1.4.1. La méthode de simplexe:**

La méthode du Simplexe est une procédure itérative, elle consiste à se déplacer d'un sommet à un autre tout en améliorant la valeur de la fonction économique, Elle démarre d'un point extrême (sommet de départ) et passe au sommet voisin, et ceci constitue une itération de l'algorithme du simplexe. Pour cela, on doit définir le point extrême de départ et le test d'arrêt.



**Définition 3:**

Une solution réalisable  $x$  de PL est dite de base si  $(n - m)$  de ses composantes sont nulles, et aux autres  $(x_{j_1}, x_{j_2}, \dots, x_{j_m})$  correspondent  $m$  vecteurs  $(a_{j_1}, a_{j_2}, \dots, a_{j_m})$  de la matrice de condition  $A$  linéairement indépendants.

L'ensemble  $J_B = (j_1, j_2, \dots, j_m)$  est appelé ensemble des indices de base,  $J_H = J / J_B$  Ensemble des indices hors base.

Autrement:

Une solution réalisable  $x = x(J)$  est solution de base si  $x_H = x(J_H) = 0$ ,  $\det A_B \neq 0$  où  $A_B = A(I, J_B)$ ,  $x(J_B) \geq 0$  La matrice  $A_B$  est appelée la matrice de base,  $x_j, j \in J_B$  les composantes de base.

$x_j, j \in J_H$  Les composantes hors base.

**Définition 4:**

Une solution réalisable de base  $x$  est dite non dégénérée si  $x_j > 0, j \in J_B$ .

L'accroissement de la fonction objective  $Z$  est égale à:

$$\Delta Z = Z(\bar{x}) - Z(x) = c' \bar{x} - c' x = c' \Delta x.$$

Construisons le  $m$ -vecteur  $y = y(I)$  vecteur de potentiels:

$$y' = c'_B A_B^{-1}$$

Et le vecteur  $\Delta = \Delta(J) = (\Delta_j, j \in J)$ ; le vecteur estimations:

$$\begin{cases} \Delta' = y' A - c' \\ \Delta_j = y' a_j - c_j, j \in J \end{cases}$$

**Critère d'optimalité:**

**Théorème:** soit  $\{x, A_B\}$  une solution réalisable de départ.

L'inégalité  $\Delta_H = \Delta(J_H) \geq 0$  est suffisante et dans le cas de la non dégénérescence elle est nécessaire pour l'optimalité de  $\{x, A_B\}$ .

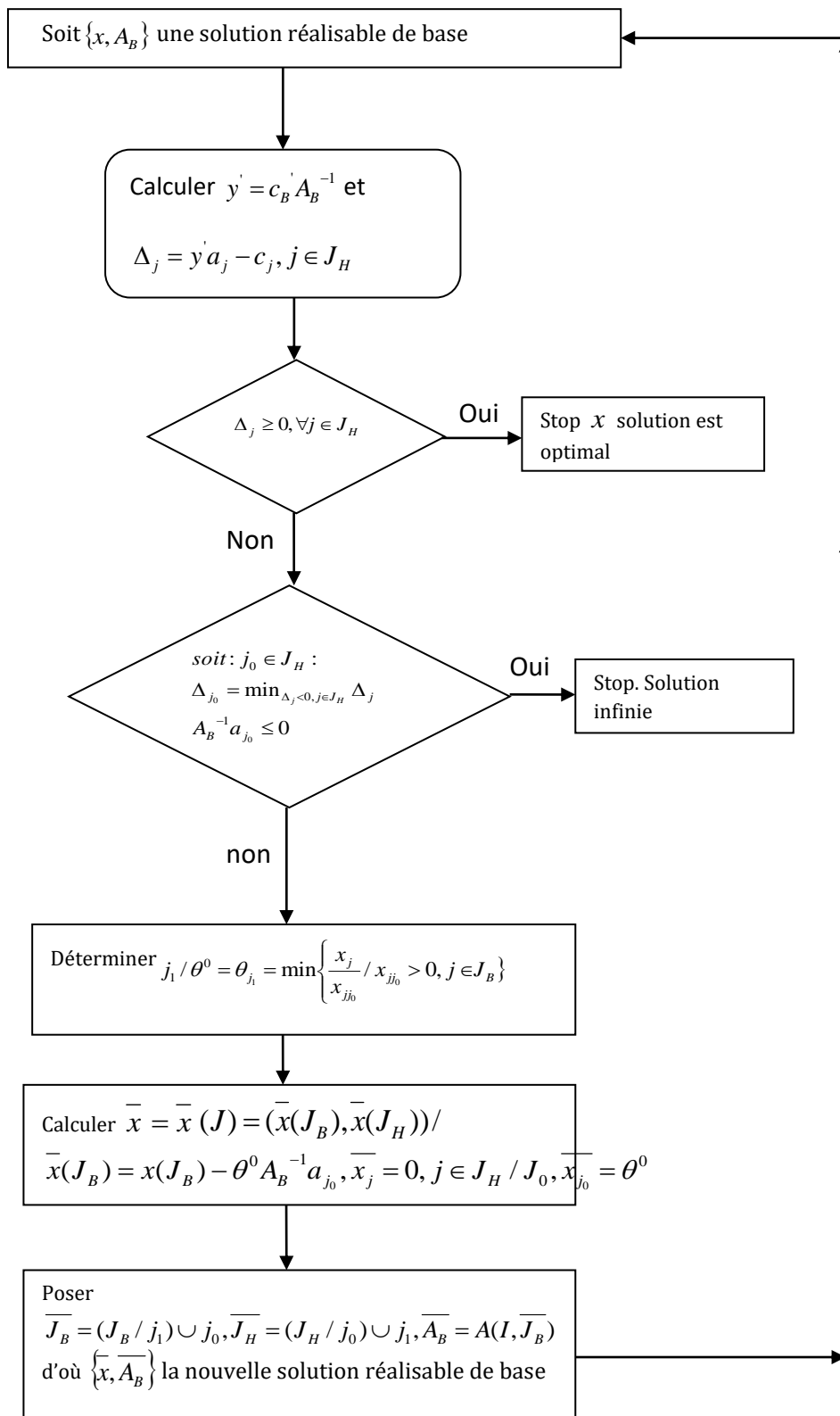


Figure 2: L'organigramme de la méthode de simplexe (maximisation) [1]

**Exemple 1.1:**

Nous allons résoudre le problème linéaire suivant en utilisant la méthode du simplexe:

$$\text{PL} \left\{ \begin{array}{l} \max Z = x_1 + 3x_2 \\ \text{s.c} \\ x_1 + x_2 \leq 14 \\ -2x_1 + 3x_2 \leq 12 \\ 2x_1 - x_2 \leq 12 \\ x_1 \geq 0, x_2 \geq 0 \end{array} \right.$$

On passe d'abord de la forme canonique à la forme standard en ajoutant des variables d'écart:

$$\text{PL} \left\{ \begin{array}{l} \max z = x_1 + 3x_2 + 0x_3 + 0x_4 + 0x_5 \\ \text{s.c} \\ x_1 + x_2 + x_3 = 14 \\ -2x_1 + 3x_2 + x_4 = 12 \\ 2x_1 - x_2 + x_5 = 12 \\ x_j \geq 0, j = \overline{1,5} \end{array} \right.$$

On a  $j = \{1,2,3,4,5\}$  et  $j_B = \{3,4,5\}$ ,  $j_H = \{1,2\}$ , avec  $A_B = I_3$  donc solution réalisable de base est  $x = (0,0,5,1,3)$ , dressons alors le premier tableau du simplexe.

$C$			1	3	0	0	0
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
0	$a_3$	14	1	1	1	0	0
← 0	$a_4$	12	-2	3	0	1	0
0	$a_5$	12	2	-1	0	0	1
$Z=0$		$\Delta_j$	-1	↑ -3	0	0	0

On remarque que  $\Delta_j \geq 0, \forall j \in J_H$  n'est pas vérifiées, donc la solution réalisable de base initiale n'est pas optimale, on doit alors changer la base de la manière suivante :

$$\min_{j \in J_H} \Delta_j = \Delta_2 = -3, \text{ donc } j_0 = 2.$$

De là le vecteur  $a_2$  va entrer dans la nouvelle base.

Et calculons  $\theta^0 = \min_{j \in J_B} \theta_j$

$$\theta_3 = \frac{14}{1}, \theta_4 = \frac{12}{3}. \text{ D'où } \theta^0 = \theta_{j_1} = \min_{j \in J_B} \theta_j = \theta_4 = \frac{12}{3}$$

Donc le vecteur  $a_4$  va sortir de la base pour trouver la nouvelle solution  $\bar{x}$  ; dressons le 2<sup>ème</sup> tableau du simplexe:

C			1	3	0	0	0
$c_B$	Base	b	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
← 0	$a_3$	10	5/3	0	1	-1/3	0
3	$a_2$	4	-2/3	1	0	1/3	0
0	$a_5$	16	4/3	0	0	1/3	1
Z=0		$\Delta_j$	-3 ↑	0	0	1	0

La nouvelle solution de base est donc  $\bar{x} = (0.4.10.0.16)$  de plus elle n'est pas optimale car  $\bar{\Delta}_2 = -4 < 0$ .

On doit alors changer la base une autre fois:

$$\min_{j \in J_H} \bar{\Delta}_j = \bar{\Delta}_1 = -3, \text{ donc Le vecteur } a_1 \text{ va rentrer dans la nouvelle base}$$

Comme  $\theta_{j_1} = \min_{j \in J_B} \theta_j = \theta_3 = 6$  le vecteur  $a_3$  sortira de la base.

D'où on obtient  $\bar{J}_B = \{1,2,5\}, \bar{J}_H = \{3,4\}$ , pour déterminer la nouvelle solution  $\bar{\bar{x}}$  dressons le 3<sup>ème</sup> tableau de simplexe:

$C$			1	3	0	0	0
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
1	$a_1$	6	1	0	3/5	-1/5	0
3	$a_2$	8	0	1	2/5	1/5	0
0	$a_5$	8	0	0	-4/5	1/5	1
$Z=30$		$\Delta_j$	0	0	9/5	2/5	0

La nouvelle solution de base est donc  $\bar{x} = (6.8.0.0.8)$ , comme  $\bar{\Delta}_j \geq 0, \forall j \in \bar{J}_H$

L'algorithme s'arrête et la solution obtenue est optimale. Avec  $\bar{Z} = 30$ .

### 1.4.2. La M-Méthode:

Le mathématicien américain Tcharness a proposé cette méthode pour résoudre les programmes linéaires.

Cette méthode permet de tenir compte des variables artificielles. On les pénalise en leur affectant un coefficient de valeur très élevée dans la fonction économique (-M dans un problème de maximisation et M dans un problème de minimisation).

Soit le problème:

$$(PL) \begin{cases} Z = c'x \rightarrow \max \\ Ax = b \\ x_j \geq 0, j = \overline{1..n} \end{cases}$$

On constitue le problème (PM) de la manière suivante:

$$(PM) \begin{cases} \bar{Z} = c'x - M \sum_{i=1}^m x_{n+i} \rightarrow \max \\ [Ax]_i + x_{n+i} = b_i, i = \overline{1..m} \\ x_j \geq 0, j = \overline{1..n+m} \end{cases}$$

Ou  $M \gg 0$  (un nombre positif très grand).

Le vecteur  $X = (0, b)' = (x = 0, x_{n+i} = b_i, i = \overline{1..m})'$

Est une solution de base réalisable pour (PM).

On résout le problème (PM) par la méthode de simplexe avec une solution réalisable de base de départ  $\{(0.b), A_B\}$ .

**Principe de la M-méthode**

Après la transformation de (P) en (PM), à l'optimum nous avons les cas suivants:

1. si toutes les variables artificielles sont nulles alors la solution est optimale. De plus si  $\Delta_{j_0} = 0 / j_0 \in J_H$  alors on aura une infinité de solutions optimales.
2. Au moins une variable artificielle de la base strictement positive  $\Rightarrow$  contraintes contradictoires.
3. A un certain moment on ne peut pas améliorer, par manque de pivot. On distingue deux cas:
  - Toutes les variables artificielles sont nulles  $\Rightarrow$  solution infinie.
  - Au moins une variable artificielle strictement positive  $\Rightarrow$  contraintes contradictoires.

**Exemple 1.2:**

$$\text{PL} \left\{ \begin{array}{l} \max Z = 5x_1 + 7x_2 \\ \text{s.c} \\ x_1 - x_2 \geq 6 \\ x_1 \geq 4 \\ x_2 \leq 3 \\ x_1 \geq 0, x_2 \geq 0. \end{array} \right.$$

Passons d'abord à la forme standard:

$$\text{PL} \left\{ \begin{array}{l} \max Z = 5x_1 + 7x_2 + 0x_3 + 0x_4 + 0x_5 \\ \text{s.c} \\ x_1 - x_2 - x_3 = 6 \\ x_1 - x_4 = 4 \\ x_2 + x_5 = 3 \\ x_1, x_2, \dots, x_5 \geq 0 \end{array} \right.$$

On n'a pas de base réalisable donc on introduit une variable artificielle dans chacune des contraintes 1 et 2:

$$\text{PL} \left\{ \begin{array}{l} \max Z = 5x_1 + 7x_2 + 0x_3 + 0x_4 + 0x_5 - Mv_1 - Mv_2 \\ \text{s.c} \\ x_1 - x_2 - x_3 + v_1 = 6 \\ x_1 - x_4 + v_2 = 4 \\ x_2 + x_5 = 3 \\ x_1, x_2, \dots, x_5 \geq 0 \quad . v_1, v_2 \geq 0 \end{array} \right.$$

Dressons le premier tableau de simplexe:

$C$			5	7	0	0	0	$-M$	$-M$
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$v_1$	$v_2$
$-M$	$v_1$	6	1	-1	-1	0	0	1	0
$\leftarrow -M$	$v_1$	4	1	0	0	-1	0	0	1
0	$a_5$	3	0	1	0	0	1	0	0
		$\Delta_j$	$\uparrow -2M - 5$	$M - 7$	$M$	$M$	0	0	0

La solution de départ n'est pas optimale car  $\Delta_j < 0$ , donc on doit faire rentrer  $a_1$  dans la base et faire sortir  $v_2$ , ainsi on obtient le nouveau tableau suivant:

$C$			5	7	0	0	0	$-M$	$-M$
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$v_1$	$v_2$
$\leftarrow -M$	$v_1$	2	0	-1	-1	1	0	1	/
5	$a_1$	4	1	0	0	-1	0	0	/
0	$a_5$	3	0	1	0	0	1	0	/
		$\Delta_j$	0	$M - 7$	$M$	$\uparrow -M - 5$	0	0	/

La solution n'est pas optimale car  $\Delta_j < 0$ , donc on doit faire rentrer  $a_4$  dans la base et faire sortir  $v_1$ , ainsi on obtient le nouveau tableau suivant:

$C$			5	7	0	0	0	$-M$	$-M$
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$v_1$	$v_2$
0	$a_4$	2	0	-1	-1	1	0	/	/
5	$a_1$	6	1	-1	-1	0	0	/	/
← 0	$a_5$	3	0	1	0	0	1	/	/
		$\Delta_j$	0	-12 ↑	-5	0	0	/	/

On fait rentrer  $a_2$  dans la base et faire sortir  $a_5$ , ainsi on obtient le tableau suivant:

$C$			5	7	0	0	0	$-M$	$-M$
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$v_1$	$v_2$
0	$a_4$	5	0	0	-1	1	1	/	/
5	$a_1$	9	1	0	-1	0	1	/	/
7	$a_2$	3	0	1	0	0	1	/	/
		$\Delta_j$	0	0	-5 ↑	0	12	/	/

On n'a pas de pivot positif donc la solution est infinie.

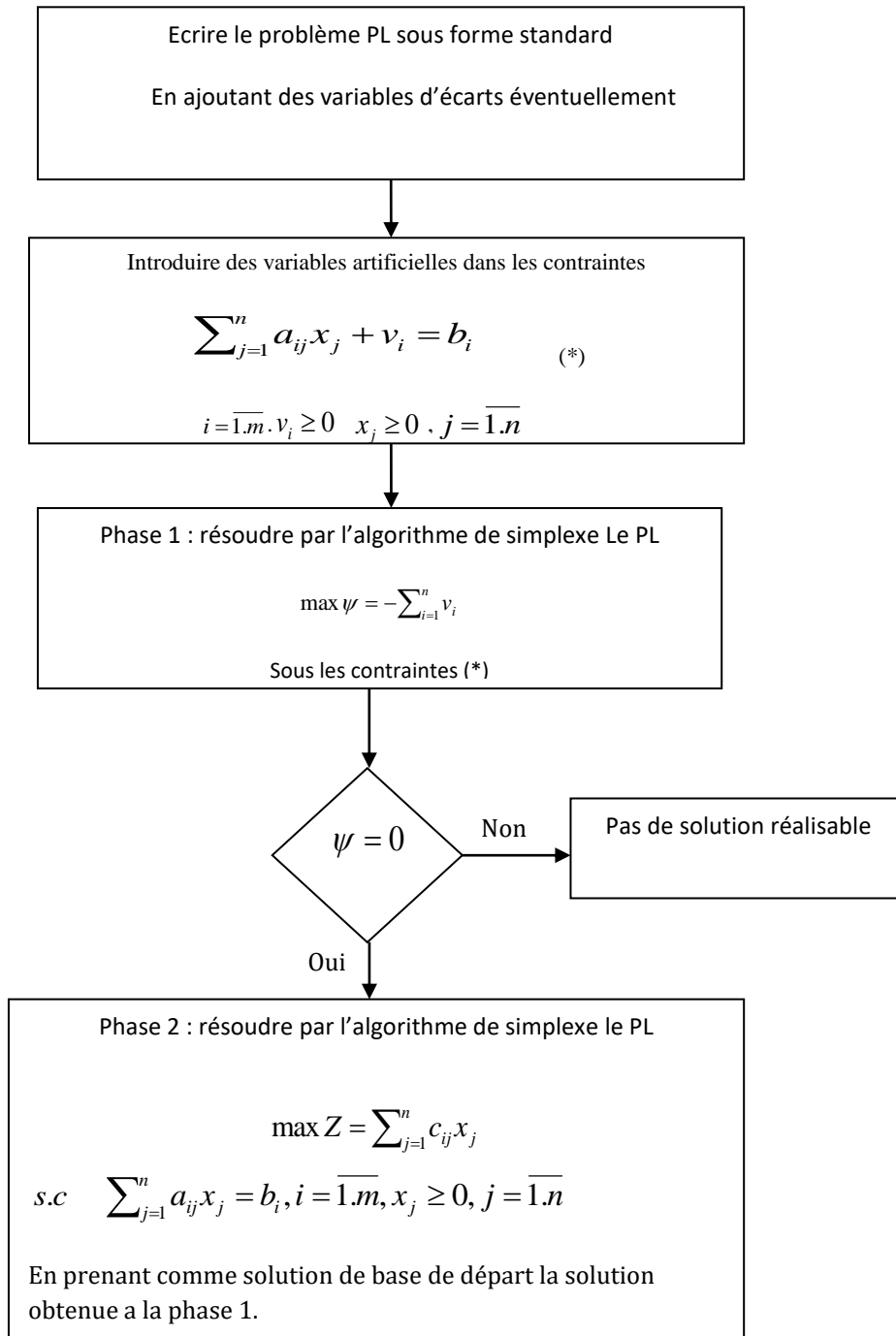
### 1.4.3. La méthode des deux phases:

#### Principe général:

Comme son nom l'indique, la méthode des deux phases admet deux étapes essentielles mais il est nécessaire avant d'entamer la première phase d'écrire le programme linéaire (PL) sous la forme standard en ajoutant des variables artificielles aux contraintes et en générant une fonction objective à ces variables artificielles, une fois la forme standard obtenue, la première phase peut être lancée.

En effet, il s'agit d'appliquer l'algorithme du simplexe au programme linéaire mis sous forme standard dont le but est de trouver une solution de base réalisable. Si cette première phase ne fournit pas une solution de base réalisable, alors cela veut dire que la solution du (PL) est non bornée. Sinon, c'est-à-dire si la solution de base réalisable est obtenue, alors la deuxième phase peut être lancée, elle consiste à résoudre le programme linéaire en appliquant l'algorithme du simplexe en partant de la solution de base réalisable fournie par la première phase.

**Pour un problème de maximisation**



**Figure 3: L'organigramme de la méthode des deux phases**

**Exemple 1.3:**

$$\text{PL} \left\{ \begin{array}{l} \max z = 3x_1 + x_2 - 2x_3 \\ \text{s.c} \\ x_1 + 2x_2 \geq 10 \\ 3x_1 - x_2 + x_3 = 7 \\ x_1 + 3x_3 \leq 8 \\ x_2, x_3 \geq 0. x_1 \in \mathbb{R} \end{array} \right.$$

$x_1 \in \mathbb{R}$  donc on pose  $x_1 = z_1 - z_2$  avec  $z_1 \geq 0$ ,  $z_2 \geq 0$

La forme standard:

$$\text{PL} \left\{ \begin{array}{l} \max Z = 3z_1 - 3z_2 + x_2 - 2x_3 + 0x_4 + 0x_5 + 0x_6 - x_7 - x_8 \\ \text{s.c} \\ z_1 - z_2 + 2x_2 - x_4 + x_7 = 10 \\ 3z_1 - 3z_2 - x_2 + x_3 + x_8 = 7 \\ z_1 - z_2 + 3x_3 + x_5 = 8 \\ x_3 + x_6 = 2 \\ z_1, z_2 \geq 0. \quad x_2 \dots x_8 \geq 0 \end{array} \right.$$

**1<sup>ere</sup> phase:**

C			0	0	0	0	0	0	0	-1	-1
$c_B$	Base	b	$z_1$	$z_2$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$
-1	$a_7$	10	1	-1	2	0	-1	0	0	1	0
← 1	$a_8$	7	3	-3	-1	1	0	0	0	0	1
0	$a_5$	8	1	-1	0	3	0	1	0	0	0
0	$a_6$	2	0	0	0	1	0	0	1	0	0
		$\Delta_j$	-4↑	4	-1	-1	1	0	0	0	0

La solution n'est pas optimale car  $\Delta_j < 0$ , donc on doit faire rentrer  $z_1$  dans la base et faire sortir  $a_8$ , ainsi on obtient le nouveau tableau suivant:

$C$			0	0	0	0	0	0	0	-1	-1
$c_B$	Base	$b$	$z_1$	$z_2$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$
← -1	$a_7$	23/3	0	0	7/3	-1/3	-1	0	0	1	/
0	$z_1$	7/3	1	-1	-1/3	1/3	0	0	0	0	/
0	$a_5$	17/3	0	0	1/3	8/3	0	1	0	0	/
0	$a_6$	2	0	0	0	1	0	0	1	0	/
		$\Delta_j$	0	0	↑ -7/3	1/3	1	0	0	0	/

La solution obtenue n'est pas optimale, donc on doit rentrer  $a_2$  dans la base et faire sortir  $a_7$ , ainsi on obtient le tableau suivant:

$C$			0	0	0	0	0	0	0	-1	-1
$c_B$	Base	$b$	$z_1$	$z_2$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$
0	$a_2$	23/7	0	0	1	-1/7	-3/7	0	0	/	/
0	$z_1$	24/7	1	-1	0	2/7	-1/7	0	0	/	/
0	$a_5$	32/7	0	0	0	19/7	1/7	1	0	/	/
0	$a_6$	2	0	0	0	1	0	0	1	/	/
		$\Delta_j$	0	0	0	0	0	0	0	/	/

**2<sup>eme</sup> phase:**

$C$			3	-3	1	-2	0	0	0		
$c_B$	Base	$b$	$z_1$	$z_2$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$		
1	$a_2$	23/7	0	0	1	-1/7	-3/7	0	0		
3	$z_1$	24/7	1	-1	0	2/7	-1/7	0	0		
← 0	$a_5$	32/7	0	0	0	19/7	1/7	1	0		
0	$a_6$	2	0	0	0	1	0	0	1		
		$\Delta_j$	0	0	0	19/7	-6/7	↑ 0	0		

La solution n'est pas optimale donc on doit rentrer  $a_4$  dans la base et faire sortir  $a_5$  ainsi on obtient le tableau suivant:

$C$			3	-3	1	-2	0	0	0
$c_B$	Base	$b$	$z_1$	$z_2$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$
1	$a_2$	17	0	0	1	8	0	3	0
3	$z_1$	8	1	-1	0	3	0	1	0
0	$a_4$	32	0	0	0	19	1	7	0
0	$a_6$	2	0	0	0	1	0	0	1
		$\Delta_j$	0	0	0	19	0	6	0

La solution optimale est  $\bar{x} = (z_1, z_2, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = (8.0.17.0.32.0.2)$ .

#### 1.4.4. Dualité:

À tout programme linéaire on associe un second programme dual du premier (appelé primal). Ce programme a la propriété d'être en relation étroite avec le premier (la solution optimale de l'un des deux programmes est complètement déterminée par la résolution de l'autre).

Soit le programme linéaire primal écrit sous sa forme standard:

$$(PL) \begin{cases} \max Z = c'x \\ Ax = b \\ x \geq 0 \end{cases}$$

Auquel on associe son programme linéaire dual:

$$(D) \begin{cases} \min W = b'y \\ A'y \geq c \\ y \in \mathbb{R}^m \end{cases}$$

**Méthode de construction du dual [1]:** Pour déterminer le dual d'un programme linéaire, on doit suivre les règles résumées dans le tableau suivant:

primal	dual
$c'x \rightarrow \max$	$b'y \rightarrow \min$
$[Ax]_i = b_i, i = \overline{1.m_1}$	$[A'y]_j \geq C_j, j = \overline{1.n_1}$
$[Ax]_i \leq b_i, i = \overline{m_1+1.m_2}$	$[A'y]_j \leq C_j, j = \overline{n_1+1.n_2}$
$[Ax]_i \geq b_i, i = \overline{m_2+1.m}$	$[A'y]_j = C_j, j = \overline{n_2+1.n}$
$x_j \geq 0, j = \overline{1.n_1}$	$y_i \in \mathbb{R}, i = \overline{1.m_1}$
$x_j \leq 0, j = \overline{n_1+1.n_2}$	$y_j \geq 0, j = \overline{m_1+1.m_2}$
$x_j \in \mathbb{R}, j = \overline{n_2+1.n}$	$y_i \leq 0, i = \overline{m_2+1.m}$

**Exemple 1.4:**

$$\text{primal} \left\{ \begin{array}{l} \max Z = 10x_1 + 6x_2 \\ x_1 + 4x_2 \leq 40 \\ 3x_1 + 2x_2 = 60 \\ 2x_1 + x_2 \geq 25 \\ x_1 \geq 0, x_2 \geq 0 \end{array} \right.$$

$$\text{Dual} \left\{ \begin{array}{l} \min W = 40y_1 + 60y_2 + 25y_3 \\ y_1 + 3y_2 + 2y_3 \geq 10 \\ 4y_1 + 2y_2 + y_3 \geq 6 \\ y_1 \geq 0, y_2 \in \mathbb{R}, y_3 \leq 0 \end{array} \right.$$

**1.4.5. La méthode de dual simplexe:**

Etant donné le problème primal de programmation linéaire:

$$(\text{PL}) \left\{ \begin{array}{l} c'x \rightarrow \max \\ Ax = b \\ x \geq 0 \end{array} \right.$$

Et son dual:

$$(D) \begin{cases} b'y \rightarrow \min \\ A'y \geq c \\ y \in \mathbb{R}^m \end{cases}$$

**Définition 1 [1]:** de l'ensemble  $J$  choisissant un ensemble  $J_B \in J$

Et soit  $A_B = (I, J_B)$  une sous matrice inversible de  $A$ .

En utilisant la matrice  $A_B$ , on construit le vecteur  $y$  :

$$y' = c'_B A_B^{-1}$$

Le vecteur  $y$  est dit plan dual basique et  $A_B$  la matrice de base si

$$A'_H y \geq c_H$$

Ou:  $A_H = (I, J_H), J_H = J / J_B$ .

**Définition 2 [1]:** un plan dual basique  $y$  est dit non dégénéré si  $A'_H Y > C_H$

En utilisant un plan dual basique de départ  $y$ , on construit les vecteurs suivants:

$$\delta(J) = A'y - c, x(J) = (x(J_B), x(J_H)), x(J_B) = x_B = A_B^{-1}b, x(J_H) = x_H = 0$$

Appelés coplan et pseudo plan respectivement de problème (P).

**Remarque 1 :** par construction  $\delta(J_B) = 0$  et  $\delta(J_H) \geq 0$ ,

Si  $y$  est un plan dual basique alors  $\delta$  et  $x$  sont dits basiques.

### Algorithme de dual simplexe [1]:

Considérant un plan dual basique  $y$  avec sa matrice de base  $A_B$ .

En utilisant  $A_B$ , on calcule le pseudo plan  $x = (x_B = A_B^{-1}b, x_H = 0)$ .

Si  $x_B \geq 0$  alors  $x$  est optimale pour le problème (P) et  $y$  optimal du dual (D)

Sinon, on calcul  $x_{j_0} = \min x_j, (x_j < 0, j \in J_B)$  de la l'indice  $j_0$  doit sortir de la base et la colonne  $a_{j_0}$  doit sortir de  $A_B$ , c'est-à-dire on change de base ( $A_B \rightarrow \overline{A_B}$ ).

Le changement de base entraine le changement de plan dual  $y$  ( $y \rightarrow \bar{y}$ ) qui entraine aussi le changement de coplan  $\delta$  ( $\delta \rightarrow \bar{\delta}$ ).

Ce changement de coplan se fera de la manière suivante :  $\bar{\delta} = \delta + \Delta\delta$  ou:

$$\Delta\delta_j = \begin{cases} \sigma, j = j_0 \\ 0, j \in J_B / j_0 \end{cases}$$

Ou  $\sigma$  le pas dual positif ou nul.

$$\Delta\delta_j = \sigma x_{j_0 j}, j \in J_H, \text{ ou } x_{j_0 j} \text{ est la } j^{\text{eme}} \text{ composante de vecteur } A_B^{-1} a_j.$$

Pour que  $\bar{\delta}$  soit un coplan, il faut avoir un pas maximal  $\sigma^0$  :

$$\sigma^0 = \min_{x_{j_0 j} < 0, j \in J_H} \left\{ \frac{-\delta_j}{x_{j_0 j}} \right\} = \frac{-\delta_{j_1}}{x_{j_0 j_1}}$$

La nouvelle base sera  $\bar{J}_B = (J_B / j_0) \cup j_1$ , et  $\bar{A}_B = A(I, \bar{J}_B)$  la nouvelle itération débutera avec  $\bar{x} = (\bar{x}_B) = (\bar{A}_B^{-1} b, \bar{x}_H = 0)$ .

**Remarque 2:** Les problèmes de type:

$$PL \begin{cases} Z = c'x \rightarrow \min \\ Ax \geq b \\ x \geq 0 \\ b \geq 0, c \geq 0 \end{cases}$$

Sont résolus dans la plupart des cas par la méthode duale, car en ajoutant des variables d'écart, on obtient facilement de base de départ, par contre si on utilise la méthode de simplexe, on ajoute des variables d'écart et des variables artificielles et ceci augmente la dimension de problème.

**Exemple 1.5:**

$$PL \begin{cases} \min Z = 2x_1 + 3x_2 \\ s.c \\ 4x_1 + x_2 \geq 8 \\ x_1 + 4x_2 \geq 8 \\ 7x_1 + 10x_2 \geq 47 \\ x_1, x_2 \geq 0 \end{cases} \longrightarrow PL \begin{cases} \min Z = 2x_1 + 3x_2 + 0x_3 + 0x_4 + 0x_5 \\ s.c \\ 4x_1 + x_2 - x_3 = 8 \\ x_1 + 4x_2 - x_4 = 8 \\ 7x_1 + 10x_2 - x_5 = 47 \\ x_1 \dots x_5 \geq 0 \end{cases}$$

$$\begin{array}{l} \text{min } Z = 2x_1 + 3x_2 + 0x_3 + 0x_4 + 0x_5 \\ \text{s.c} \\ -4x_1 - x_2 + x_3 = -8 \\ -x_1 - 4x_2 + x_4 = -8 \\ -7x_1 - 10x_2 + x_5 = -47 \\ x_1 \dots x_5 \geq 0 \end{array}$$

Dressons le premier tableau de simplexe:

$C$			2	3	0	0	0
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
0	$a_3$	-8	-4	-1	1	0	0
0	$a_4$	-8	-1	-4	0	1	0
0	$a_5$	-47	-7	-10	0	0	1
			-2	-3	0	0	0

Ici le critère d'optimalité n'est pas vérifié car les composantes de la solution ne sont pas toutes positives.

On a  $a_5 = -47 = \min b_i$  donc  $a_5$  va sortir de la base et  $a_1$  va rentrer dans la base. Passons à la nouvelle itération:

$C$			2	3	0	0	0
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
0	$a_3$	132/7	0	33/7	1	0	-4/7
0	$a_4$	-9/7	0	-18/7	0	1	-1/7
2	$a_1$	47/7	1	10/7	0	0	-1/7
			0	-1/7	0	0	-2/7

On a  $a_4 = -9/7 = \min b_i$  donc  $a_4$  sort de la base et sera remplacé par  $a_2$

Qui correspond au max des  $c_j - z_j$ , par la suite on passe à l'autre itération:

$C$			2	3	0	0	0
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
0	$a_3$	33/2	0	0	1	11/6	-5/6
3	$a_2$	1/2	0	1	0	-7/18	1/18
2	$a_1$	6	1	0	0	5/9	-2/9
			0	0	0	-1/18	-5/18

Tous les  $b_i$  sont positifs donc la solution est optimale.

$$x = (6.1/2.33/2.0.0).$$

## **CHAPITRE 2:**

# **Etude sur la programmation linéaire en nombre entiers**

**Introduction:**

La programmation en nombres entiers concerne les programmes d'optimisation sous contraintes pour lesquels les variables doivent prendre des valeurs entières. Elle est un domaine très riche de la programmation mathématique. Les recherches dans ce domaine sont nombreuses et les premières recherches datent de la période de **Gomory** en 1958 [4].

Dans beaucoup de problèmes d'optimisation une solution à valeurs entières est exigée. Par exemple dans le problème de la production, on cherche le nombre optimal (entier) de pièces à fabriquer.

**2.1. Forme générale d'un programme linéaire en nombres entiers "PLNE":**

La forme générale d'un programme linéaire en nombres entiers [5] se présente comme suit :

$$(PLNE) \left\{ \begin{array}{l} \max/ \min(Z) = \sum_{j=1}^n c_j x_j \\ \forall i = \overline{1.m} : \sum_{j=1}^n a_{ij} x_j \leq, = \text{ou} \geq b_i \\ \forall j = \overline{1.n} : x_j \geq 0 \\ x_j \text{ entier} \end{array} \right.$$

**Remarque:**

Dans le cas d'un programme linéaire avec des variables binaires, on aura, à la suite des contraintes fonctionnelles.

$$x_j = 0 \text{ ou } 1, j = \overline{1.n}$$

**2.2. Méthodes de résolution d'un programme linéaire en nombres entiers:**

**2.2.1. Méthode « Branch and Bound » (Méthode par séparation et évaluation):**

La technique de Branch and Bound [11] est une méthode algorithmique classique pour résoudre un problème d'optimisation linéaire en nombres entiers. Il s'agit de rechercher une solution optimale dans un ensemble linéaire de solutions possibles, la méthode repose d'abord sur la séparation (Branch) de l'ensemble des solutions en sous-ensembles plus petits. L'exploration de ces solutions utilise ensuite une évaluation optimiste pour majorer(minorer). (Bound) les sous-ensembles, ce qui permet de ne plus considérer que ceux susceptibles de contenir une solution potentiellement meilleure que la solution courante.

**Principes de la méthode:**

- On construit un arbre de recherche dont le problème initial est la racine.
- On divise le problème en sous-problèmes : l'optimum peut appartenir à l'un de ces sous-problèmes :
  - ✚ On élimine tout problème infaisable.
  - ✚ Si possible, on calcule la solution du sous-problème.
  - ✚ Sinon on calcule une borne inférieure, si elle est supérieure à la meilleure solution déjà obtenue, on élimine le sous-problème considéré.
  - ✚ Dans le cas restant, on subdivise à nouveau le domaine.

**Programme en nombre entiers:**

$$(P) \begin{cases} \max c'x \\ s.c \quad Ax \leq b \\ x \geq 0, \text{entier} \end{cases}$$

**Relaxation linéaire de problème (P):**

$$(LP) \begin{cases} \max c'x \\ s.c \quad Ax \leq b \\ x \geq 0. \end{cases}$$

**Propriétés de la relaxation linéaire:**

- La valeur de la solution optimale de LP est une borne supérieure sur la valeur de la solution optimale de P.
- La valeur d'une solution admissible de P fournit une borne inférieure sur la valeur de la solution optimale de P.
- Si la solution optimale de LP est entière (donc admissible pour P), elle est également la solution optimale de P.

**Etapes de résolution [12]:**

On présente les étapes nécessaires de résolution comme suit : (on donne ici la démarche pour un problème de maximisation).

**a) Résolution de (LP) avec l'algorithme du simplexe:**

Résoudre le (LP) correspondant (sans contraintes de variables entières), la valeur de Z du tableau optimal donne une borne supérieure au (PLNE) (P) notée par  $Z_{BS}$ .

Arrondir cette solution aux valeurs entières les plus proches et réalisables et calculer à nouveau la valeur de la fonction objective du (PLNE)(P). Appelons cette valeur  $Z_{BI}$

Ainsi  $Z_{BI} \leq Z_{opt}$  du (PLNE)(P)  $\leq Z_{BS}$

**b) Séparation:**

- Construire deux sous-ensembles à l'aide d'une variable non entière, soit  $x_i$  cette variable, soit  $[x_i^*]$  la partie entière de  $x_i^*$ , ( $x_i^*$  optimale).
- Pour éliminer la solution non entière  $x_k$ , on crée deux branches (deux sous-problèmes), on obtient une branche avec  $x_i \leq [x_i^*]$  et l'autre avec  $x_i \geq [x_i^*]+1$ .

Les deux sous-problèmes sont obtenus en ajoutant au programme linéaire (LP) précédent (celui d'où provient la séparation), la contrainte  $x_i \leq [x_i^*]$  pour un des sous-problèmes et la contrainte  $x_i \geq [x_i^*]+1$  pour l'autre sous-problème.

$$(P1) \begin{cases} \max c'x \\ s.c \ Ax \leq b \\ x_i \leq [x_i^*] \\ x \geq 0. \end{cases} \quad (P2) \begin{cases} \max c'x \\ s.c \ Ax \leq b \\ x_i \geq [x_i^*] + 1 \\ x \geq 0. \end{cases}$$

Toutes les solutions entières réalisables sont maintenant contenues dans l'un ou l'autre des sous-problèmes.

**c) Résolution des sous-problèmes et détermination des nouvelles bornes pour Z , s'il y a lieu:**

- Il s'agit de résoudre chaque sous-problème par l'algorithme du simplexe, notons par MBSD, la meilleure borne supérieure disponible de Z , elle correspond à la valeur maximale de la fonction objectif des deux sous-problèmes que nous venons de résoudre.
- Nous révisons également la borne inférieure de Z ,notons cette valeur par MBID, la meilleure borne inférieure disponible, elle correspond à la valeur maximale de la fonction objectif de toutes les solutions entières obtenues jusqu'à présent.

**d) Critère d'arrêt de l'exploitation d'une branche:**

La procédure de branchement est achevée lorsque l'une des conditions suivantes est satisfaite :

- I. Le sous-problème de la branche considérée n'admet pas de solution réalisable.
- II. La solution entière obtenue pour le sous-problème de la branche est réalisable mais la valeur  $Z$  obtenue est plus petite ou égale à MBID ( $Z^* \leq \text{MBID}$ ), ainsi une branche est terminée aussitôt qu'on obtient une solution entière.
- III. Le sous-problème de la branche considérée admet une solution optimale non entière mais sa valeur  $Z^*$  est inférieure ou égale à celle de l'autre sous-problème à solution entière.

**e) Critère d'optimalité:**

Toutes les branches sont saturées, donc on termine avec la procédure de séparation. La solution optimale de (PLNE) (P) est la solution entière qui donne la meilleure borne inférieure pour  $Z$  (MBID), à l'optimum on aura  $\text{MBID} = \text{MBSD} = Z_{\text{opt}}$  de (PLNE) (P).

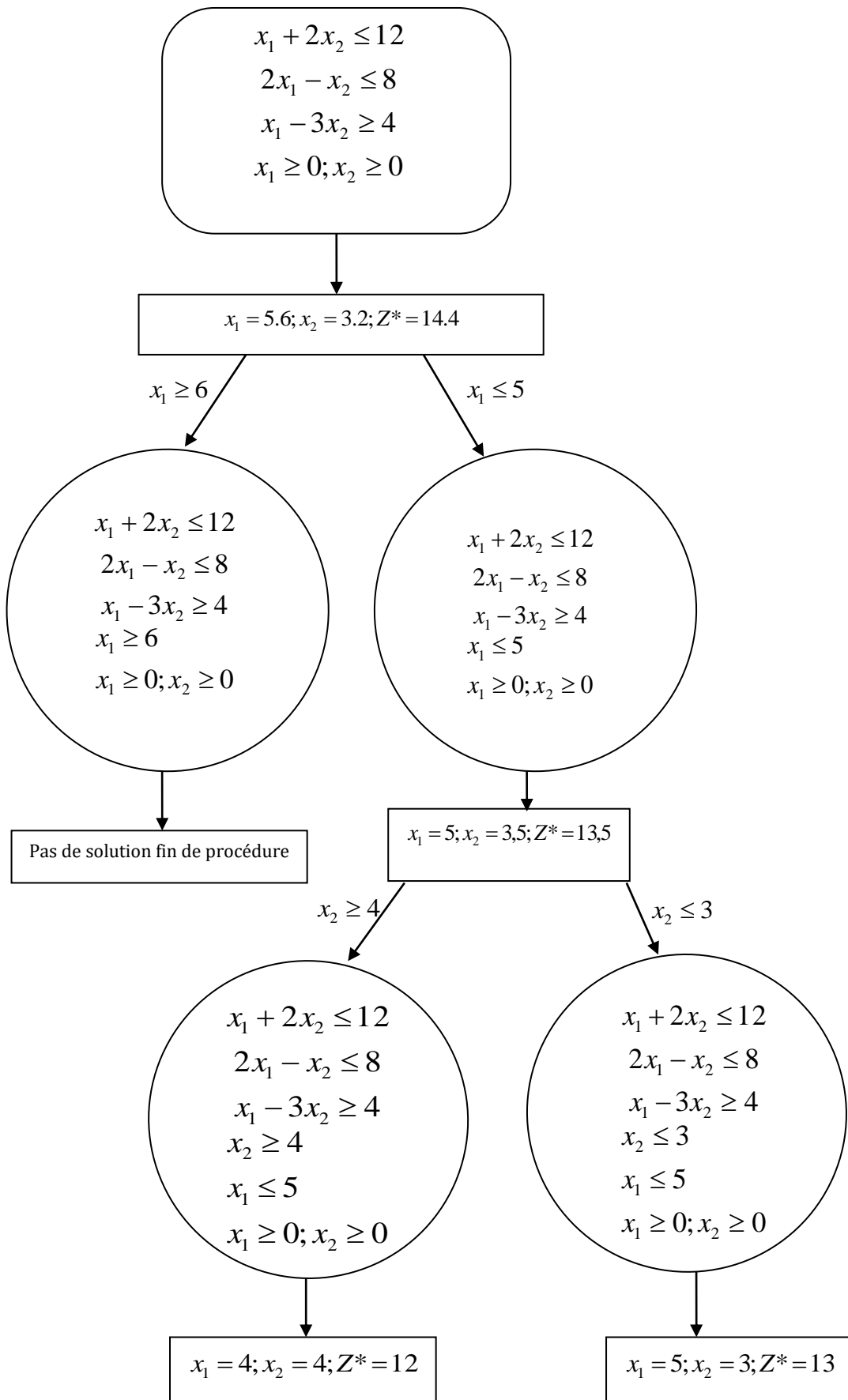
**f) Continuer le processus de séparation:**

Dans le cas où la solution obtenue dans l'un des sous-ensembles de chaque branche est non entière, on poursuit le processus de séparation à partir du sous-ensemble ayant la valeur de  $Z^*$  la plus élevée.

**Exemple 2.1 (Branch and Bound):** Soit le programme linéaire en nombres entiers suivant:

$$\text{PLNE} \left\{ \begin{array}{l} \max Z(x) = 2x_1 + x_2 \\ x_1 + 2x_2 \leq 12 \\ 2x_1 - x_2 \leq 8 \\ 4x_1 - 3x_2 \leq 8 \\ x_1 \geq 0, x_2 \geq 0 \\ x_1, x_2 \text{ entiers} \end{array} \right.$$

Sa solution est donnée dans la figure ci-dessous ou cette méthode est utilisée:



En résolvant le problème relaxé (PLNE) on trouve la valeur de  $Z_{opt(PLNE)}$  tel que:

$$Z_{BI} \leq Z_{opt} \leq Z_{BS} \Rightarrow 11 \leq 14.4 \leq 14.4$$

La solution n'est pas optimale donc on fait un branchement au niveau de  $x_1$ .

Si  $x_1 \geq 6$  pas de solution.

Si  $x_1 \leq 5$  la solution obtenue n'est pas entière donc la solution se trouve au niveau de 2<sup>eme</sup> branchement qui nous donne deux solutions optimales.

On choisit le max des deux qui est :  $(x_1 = 5; x_2 = 3; Z^* = 13)$ .

### 2.2.2. Méthode de Balas [2]:

La méthode de Balas est une méthode arborescente, comme toute autre technique arborescente (Branch and Bound), cette méthode a des limites au-delà d'une certaine dimension.

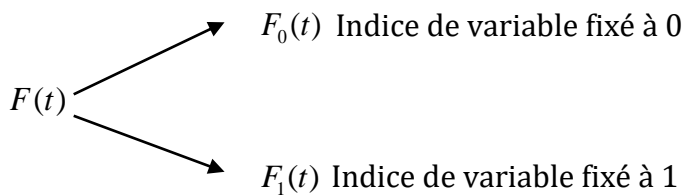
Cette méthode peut être classée dans la classe des problèmes NP-difficile.

#### Nœuds de séparation (branching):

Chaque nœud de (sommet)  $S_i$  correspond à une solution partielle. Ce nœud correspond à deux sous-ensembles d'indices pour les variables.

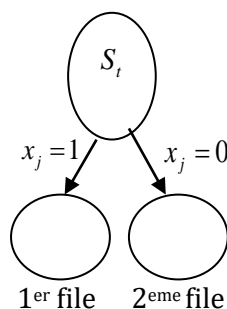
Soient:

$F(t)$  Les sous-ensembles d'indices fixés à 0 ou 1 pour les variables au niveau du nœud  $S_i$



$L(t)$  Les sous-ensembles d'indices de variables non fixés (libres).

Pour faire le branching, on choisira donc une variable telle que  $j \in L(t)$ , à ce moment on aura deux files:



**Remarque 1:**

Le problème posé c'est comment choisir sur lequel on fera le branching (séparation).  
 Pour cela écrivons le problème sur n'importe quel nœud (sommet)  $S_t$ .

$$\begin{aligned} \min Z_t &= \sum_{j \in L(t)} c_j x_j + \sum_{j \in F_1(t)} c_j \\ \sum_{j \in L(t)} a_{ij} x_j &\leq b_i - \sum_{j \in F_1(t)} a_{ij} = S_i \\ i &= \{1, 2, \dots, m\} \end{aligned}$$

Nous appelons l'évaluation  $(S_t) = \sum_{j \in F_1(t)} c_j$  : valeur meilleure pour ce nœud notée  $(ev(S_t))$ .

$(S_t)$  est un nœud quelconque de l'arborescence.

**Les tests sur les nœuds ( les sommets  $S_t$  ):**

Après création d'un nœud  $S_t$  et son évaluation il faut faire les tests suivants :

- 1) On abandonne un nœud si l'évaluation de  $(S_t)$  est supérieure ou égale à  $Z_{trouvé}$  valeur de la solution provisoire trouvée.
- 2)  $x$  solution obtenue en posant  $x_j = 0$  et  $j \in L(t)$  est solution réalisable pour  $Pl_{(0,1)}$  comme avec  $S_i \geq 0 \ i \in \{1 \dots m\}$  alors le nœud est terminal et en  $(S_t)$  son évaluation  $\min Z_t$ .

**Décision de séparation au niveau d'un sommet (nœud):**

Comment séparer au niveau d'un sommet ? (Quelle variable de séparation doit-on choisir ?).

Balas a donné une technique pour minimiser le nombre de branching à effectuer et cela comme suivant:

Soient deux sous-ensembles  $Q(t)$  et  $R(t)$  définis comme suit:

$$\begin{aligned} Q(t) &= \{i / S_i < 0, i = \overline{1..m}\} \\ R(t) &= \{j \in L(t) / \exists i \in Q(t), a_{ij} < 0\} \end{aligned}$$

**Mesure de proximité:**

Balas propose pour ses solutions  $(P(t) < 0)$

$$P(t) = \sum_{i \in Q(t)} S_i = \sum_{i=1..m} \min(0, S_i).$$

La proximité d'une solution si on fixe  $x_j$  à 1 est la suivante:

$$P(t, j) = \sum_{i=1..m} \min(0, S_i - a_{ij}).$$

Le terme  $S_i - a_{ij}$  est le second membre, si on fixe  $x_j$  à 1:

Pour déterminer la variable  $x_{j^*}$  à choisir pour faire le branching (séparation), la méthode propose de choisir l'indice  $j^* \in R(t)$ ,  $P(t, j^*) = \max(P(t, j), j \in R(t))$ .

**Conclusion:**

- ❖  $P(t, j^*) = 0$ , la solution obtenue pour  $x_{j^*} = 1$  et  $\forall j \in L(t)$  est une nouvelle solution réalisable (nœud terminal).
- ❖ En cas d'ex aequo on choisit la variable de coût faible.

Exemple 2.2: soit le problème suivant :

$$PL_{(0-1)} \left\{ \begin{array}{l} \min Z = 4x_1 + 7x_2 + 9x_3 + 3x_4 + x_5 \\ -x_1 + 2x_2 - 5x_3 - x_4 + 3x_5 \leq -3 \\ x_1 - 5x_2 - 3x_3 + x_4 - x_5 \leq 0 \\ x_2 - 3x_3 + x_4 + 2x_5 \leq -2 \\ x_j \in \{0,1\}, j = \overline{1,5} \end{array} \right.$$

$(S_0)$

$F(t) = \emptyset$ .

$L(t) = \{1,2,3,4,5\}$ .

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$S_i$	$x_1$ $S_i - a_{ij}$	$x_3$ $S_i - a_{ij}$	$x_4$ $S_i - a_{ij}$
-1	2	-5	-1	3	-3	-2	2	-2
1	-5	-3	1	-1	0	-1	3	-1
0	1	-3	1	2	-2	-2	1	-3

$P(t) = \{-5,0,-6\}$ .

$Q(t) = \{1,3\}$ .

$R(t) = \{1,3,4\}$ .

$(S_2)$

$x_1$	$x_2$	$x_4$	$x_5$		$S_i$
-1	2	-1	3		-3
1	-5	1	-1		0
0	1	1	2		-2

Par exemple  $0 < -2$  (contradictoire).

$(S_1)$

$x_1$	$x_2$	$x_4$	$x_5$		$S_i$
-1	2	-1	3		2
1	-5	1	-1		3
0	1	1	2		1

On a  $S_i \geq 0$  alors la solution est réalisable (Feuille).

## **CHAPITRE 3:**

# **Etude sur la programmation linéaire multiobjectifs**

## Introduction:

Les ingénieurs se heurtent quotidiennement à des problèmes technologiques de complexité grandissante, qui surgissent dans des secteurs très divers. Le problème à résoudre peut fréquemment être exprimé sous la forme générale d'un problème d'optimisation, dans lequel on définit une fonction objective, ou fonction Coût/profit, que l'on cherche à minimiser (ou maximiser) par rapport à tous les paramètres concernés. On identifie deux types de méthodes d'optimisation, la première est l'optimisation mono-objectif, qui se base sur la minimisation (ou la maximisation) d'une seule fonction objective où le but est de trouver la meilleure solution appelée solution optimale qu'est facilement définie suivant une seule performance du problème étudié (temps de réponse, temps de monté, la robustesse, taux d'erreur, etc). D'autre part, l'optimisation multiobjectif optimise simultanément plusieurs fonctions objectives qui sont souvent contradictoires, on cherche à trouver la meilleure solution suivant un ensemble de performance du problème (temps de réponse plus la robustesse, temps de réponse plus temps de monté plus la robustesse, etc), où le résultat d'un problème d'optimisation multicritère est généralement un assortiment de solutions, qui se distinguent par différents compromis réalisés entre les objectifs. Cet assortiment est connu par Pareto-optimal. Donc le but de l'optimisation multicritères, est d'obtenir les solutions de Pareto, par conséquent, à connaître l'ensemble des compromis possibles entre les objectifs. La résolution d'un problème d'optimisation multicritères, a conduit les chercheurs à proposer des méthodes de résolution de plus en plus performantes. Dans ce chapitre, nous présentons les principes de base de l'optimisation multiobjectifs, et nous donnerons aussi une présentation des différentes méthodes d'optimisation multiobjectifs.

Dans la vie réelle nous nous trouvons face à des objectifs souvent contradictoires, nous pouvons citer:

- Problème de planification de production:
  - Minimiser les couts;
  - Minimiser la consommation;
  - Maximiser la production;
  - Minimiser la taxe ...etc.
- Problème de raffinage de pétrole:
  - Minimiser la matière première;
  - Minimiser la pollution environnementale;

- Minimiser les couts;
- Minimiser les écarts par rapport à la demande...etc

### 3.1. Concepts de base:

Un problème d'optimisation multiobjectifs (Multi objective Optimisation Problem (MOP)) ou multicritères est un problème dans lequel nous optimisons simultanément plusieurs fonctions objectives souvent contradictoires et il est défini comme l'optimisation de plusieurs Fonctions:

$$F(x) = (f_1(x), f_2(x), \dots, f_k(x))'$$

Soumise à:

$$g_i(x) \geq b_i, i = \overline{1.p}.$$

$$h_j(x) = c_j, j = \overline{1.q} \quad x \in \mathbb{R}^n.$$

.Avec:

$$D = \left\{ \begin{array}{l} g_i(x) \geq b_i, i = \overline{1.p} \\ h_j(x) = c_j, j = \overline{1.q} \quad x \in \mathbb{R}^n \end{array} \right\}.$$

Un MOP peut être formulé comme suite:

$$\text{MOP} \left\{ \begin{array}{l} \max \{f_1(x) = Z_1\} \\ \vdots \\ \max \{f_k(x) = Z_k\} \\ \text{s.c } x \in D \end{array} \right.$$

Tel que:

- $f_i$  Sont des fonctions objectives.  $i = \overline{1.k}$ .
- $x \in D$  est une variable de décision.
- $D$  est l'ensemble des solutions réalisables qui est formé par l'intersection des contraintes du problème d'optimisation (MOP).

En particulier dans le cas linéaire, le problème d'optimisation multi objectifs est formulé comme suit:

$$\text{MOP} \begin{cases} \max \{c^i x = Z_i\} \\ \vdots \\ 1 \leq i \leq k \text{ s.c } x \in D \end{cases}$$

Où:

- $c^i$  est un  $n$ -vecteur contenant les coefficients de la  $i^{\text{eme}}$  fonction objectif  $i = \overline{1.k}$ .
- $D = \{x \in \mathbb{R}^n / Ax \leq b, x \geq 0\}$  tel que:
  - ✓  $A$  est une  $m \times n$  matrice réelle.
  - ✓  $b$  est un  $m$ -vecteur réel.

Une solution du (MOP) optimise les composantes d'un vecteur  $F(x)$  où  $x$  est un vecteur de  $n$  variables de décision  $x = (x_1, x_2, \dots, x_n)'$ .

A noter que l'ensemble des solutions réalisables  $D$  est formé par les contraintes en générale.  $(g_i(x) \geq b_i, i = \overline{1.p})$  et  $(h_j(x) = c_j, j = \overline{1.q})$  qui doivent être satisfaites tout en optimisant  $F(x)$ .

On se limite aux problèmes de maximisation, puisque la minimisation d'une fonction  $F(x)$  peut facilement être transformée en un problème de maximisation:

$$\mathbf{min} F(x) = -\mathbf{max} (-F(x))$$

**Fonction objective:**

Une fonction objective est une fonction qui modélise le but à atteindre dans le problème d'optimisation sur l'ensemble des solutions admissibles. Il s'agit de la fonction qui doit être optimisée. Elle est notée  $F(x)$  de manière générale  $F(x)$  est un vecteur :

$$F(x) = (f_1(x), f_2(x), \dots, f_k(x))'$$

Elle est aussi appelée: critère d'optimisation, fonction coût, fonction d'adaptation, ou encore performance.

**Vecteur de décision:**

Un vecteur de décision  $x$  est un vecteur correspondant à l'ensemble des variables du problème, il est noté:

$$x = (x_1, x_2, \dots, x_n)'$$

Avec:  $n$  le nombre de variables ou dimension du problème.

**Critère de décision:**

Est un critère sur lequel sont jugés les vecteurs de décision pour déterminer le meilleur vecteur. Un critère peut être une variable du problème ou une combinaison de variables.

**Contraintes:**

Une contrainte du problème est une condition que doivent respecter les vecteurs de décision du problème. Dans  $D$  Une contrainte est notée:  $g_i(x) \geq b_i$

Avec:  $i = \overline{1, p}$  tel que  $p$  : le nombre des contraintes.

**Espace réalisable:**

Représentant l'ensemble des valeurs des variables satisfaisant les contraintes.

**Espace des objectifs:**

Ensemble image de l'espace réalisable, déterminé par toutes les valeurs possibles des fonctions objectives.

**3.2. Définitions[8]****Définition 1: (espace de décision)**

L'espace  $\mathbb{R}^n$  dans lequel se situe l'ensemble de solutions réalisables  $D \subseteq \mathbb{R}^n$  est appelé espace de décisions.

**Définition 2: (espace de critères)**

L'espace  $\mathbb{R}^k$  dans lequel se situe  $Z_D$  qui est l'image de  $D$  dans  $\mathbb{R}^k$  par l'application  $F(D)$  il est appelé aussi espace objectifs.

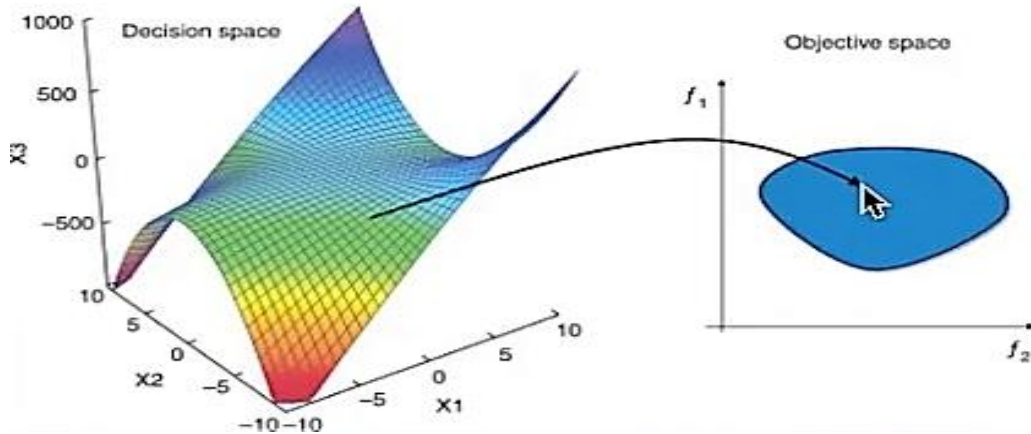


Figure 4: Représentation de l'espace de décision et l'espace des objectifs correspondant

❖ **Dominance**

*Définition 1:*

Soient  $X = (x_1, x_2, \dots, x_n)'$ ,  $Y = (y_1, y_2, \dots, y_n)'$  deux vecteurs de  $\mathbb{R}^n$ , on dit que  $X$  domine  $Y$  si et seulement si  $x_i \geq y_i, \forall i = \overline{1, n}$  avec au moins une inégalité stricte et on note:

$$X \succcurlyeq Y$$

*Définition 2: (dominance forte)*

Soient  $X, Y$  deux vecteurs. On dit que  $X$  domine fortement  $Y$  si et seulement si:  $x_i > y_i, \forall i = \overline{1, n}$  et on note :

$$X \succ Y$$

Si  $X$  domine fortement  $Y$  alors  $X$  est meilleur que  $Y$ .

*Définition 3: (dominance faible)*

Soient deux vecteurs  $X$  et  $Y$ . On dit que  $X$  domine faiblement  $Y$  si et seulement si:

$$x_i \geq y_i, \forall i = \overline{1, n}.$$

*Définition 4: (non dominance)*

Soit  $x^*$  un vecteur. On dit que  $x^*$  est non dominé si et seulement s'il n'existe aucun autre vecteur  $x$  tel que :

$$x_i \geq x_i^*, \forall i = \overline{1, n}$$

et  $x_i > x_i^*$  pour au moins un indice  $i$ . Dans le cas contraire on dit que  $x^*$  est dominé.

**Exemple:**

Pour les vecteurs  $X = (3\ 2\ 1\ 4)'$ ,  $Y = (2\ 2\ 1\ 1)'$ ,  $Z = (0\ 0\ 0\ 0)'$  et  $T = (8\ 5\ 5\ 9)'$ . La comparaison donne  $X \succcurlyeq Y$  et  $T \succ Z$ .

❖ **Efficacité**

**Définition 1 : (Pareto optimale)**

Une solution  $x^* \in D$  d'un MOP est Pareto optimale si et seulement s'il n'existe pas une solution  $x \in D$ , tel que  $Z(x)$  domine  $Z(x^*)$ , i.e  $Z_i(x) > Z_i(x^*)$  pour au moins un indice  $i$ .

La définition de la Pareto optimalité découle directement de la notion de dominance. Elle signifie qu'il est impossible de trouver une solution qui améliore les performances sur un critère sans que cela entraîne une dégradation des performances sur au moins un autre critère. Elles forment le front Pareto. Les solutions Pareto optimales sont aussi connues sous le nom de solutions efficaces, non-dominées ou non inférieures.

**Defintion2: (efficacité forte)**

Une solution  $x^* \in D$  est dite fortement efficace s'il n'existe aucun vecteur  $x \in D$  tel que  $x \neq x^*$  et  $Z_i(x) \geq Z_i(x^*)$ .

Une solution est fortement efficace si son vecteur critère est fortement non dominé.

**Définition 3: (efficacité faible)**

Une solution  $x^* \in D$  est dite faiblement efficace, s'il n'existe aucun vecteur  $x \in D$  Tel que  $Z_i(x) > Z_i(x^*)$ .

Une solution est faiblement efficace si son vecteur critère est faiblement dominé.

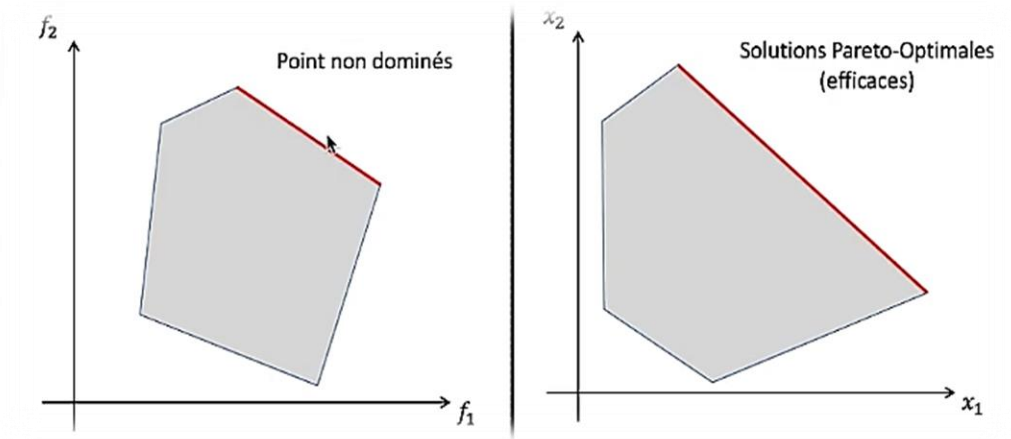


Figure 5: solution efficace et point non dominé

**Définition 4: (le point idéal)**

C'est le point obtenu en optimisant chaque critère individuellement, c'est le vecteur défini par:

$$\bar{Z} = (\max Z_1(x), \dots, \max Z_k(x)), x \in D.$$

Généralement un tel point n'est pas réalisable i.e  $\bar{Z} \notin Z_D$ .

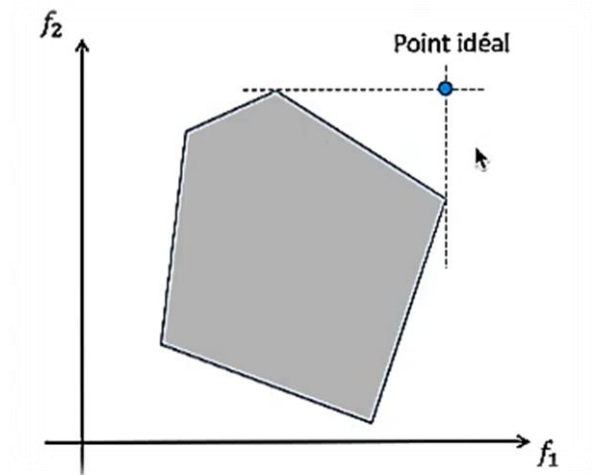


Figure 6: Le point idéal

Dans cet exemple l'intersection des droites perpendiculaire de  $Z_1$  et  $Z_2$  est appelé point idéal .on déduit qu'il n'est pas réalisable.

**Définition 5: (le point anti-idéal)**

C'est l'opposé de point idéal, c'est le vecteur défini par :

$$\underline{Z} = (\min Z_1(x), \dots, \min Z_k(x)), x \in D.$$

Le point anti-idéal n'est pas réalisable en général  $\underline{Z} \notin Z_D$ .

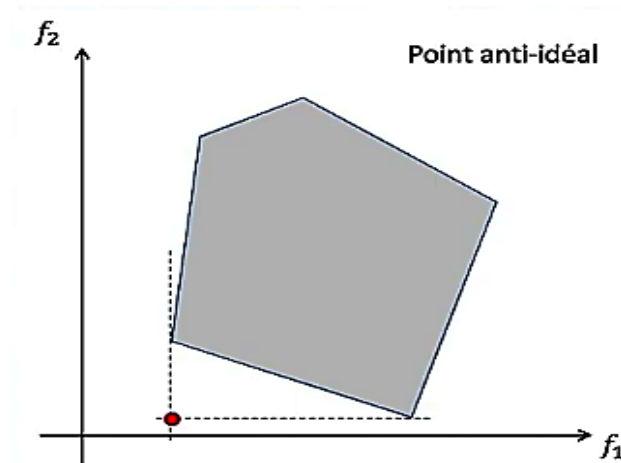


Figure 7: Le point anti-idéal

**Définition 6: (vecteur de référence)**

Un vecteur de référence  $X^* = (x_1^*, x_2^*, \dots, x_k^*)'$  est un vecteur qui définit le but à atteindre par chaque objectif  $Z_i$ .

**Définition 7: (front de Pareto)**

Le front (frontière) de Pareto est l'ensemble des solutions Pareto optimales qui sont composées des points qui ne sont dominés par aucun autre. Le front de Pareto appelé aussi surface de compromis ou l'ensemble des solutions efficaces. La figure ci-dessus (8) représente le Front du Pareto pour un problème de minimisation et maximisation de deux fonctions objectives. C'est l'ensemble des vecteurs de décision qui ne sont pas dominé.

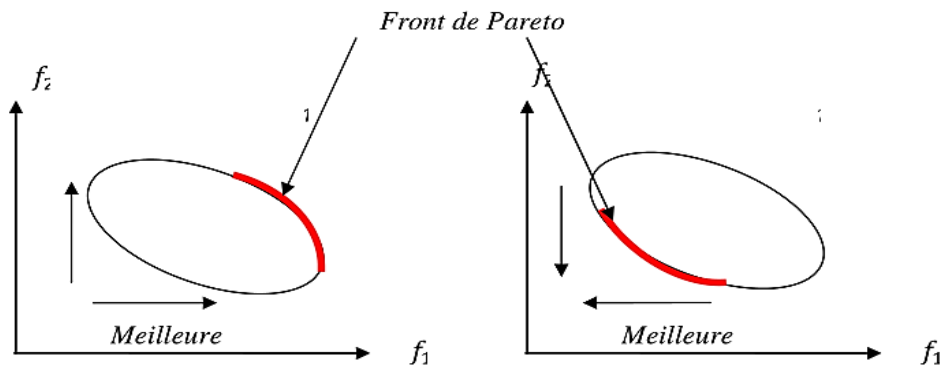


Figure 8: Front de Pareto

Exemple de dominance et d'optimalité au sens de Pareto

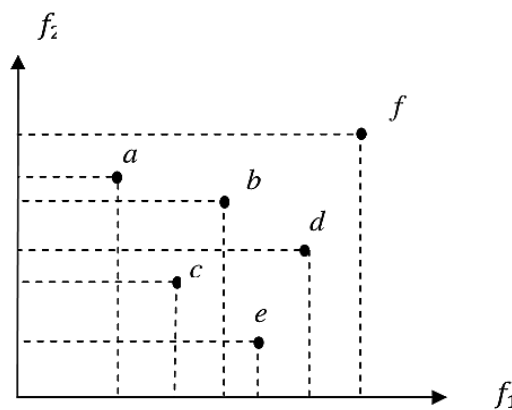


Figure 9: exemple (cas min)

On considère les six vecteurs de décision suivants:  $a = (a_1, a_2)$  ;  
 $b = (b_1, b_2)$ ;  $c = (c_1, c_2)$ ;  $d = (d_1, d_2)$ ;  $e = (e_1, e_2)$ ;  $f = (f_1, f_2)$ , Tels que :

$$a_1 < c_1 < b_1 < e_1 < d_1 < f_1 \quad \text{et} \quad e_2 < c_2 < d_2 < b_2 < a_2 < f_2.$$

Soit la fonction objective  $F(x) = [f_1(x), f_2(x)]'$  La représentation graphique de l'image de ces six vecteurs par la fonction  $F$  est fournie dans la figure 9 . Dans cet exemple on peut écrire :

- $f$  : est dominé par tous les autres vecteurs.
- $b$  : est dominé uniquement par  $c$  .
- $d$  : est dominé par  $c$  et  $e$  .

➤  $a, c$  et  $e$  : ne sont dominés par aucun vecteur.

Par conséquent les vecteurs  $(a, c, e)$  sont Pareto optimaux. Donc le front de Pareto est l'ensemble :  $\{ a, c, e \}$ .

**Définition 8: (Matrice des gains)**

La matrice carrée de dimension  $k$  suivante:

$$\begin{pmatrix} \overline{Z}_{11} & \overline{Z}_{12} & \cdot & \overline{Z}_{1k} \\ \overline{Z}_{21} & \overline{Z}_{22} & \cdot & \overline{Z}_{2k} \\ \cdot & \cdot & \cdot & \cdot \\ \overline{Z}_{k1} & \overline{Z}_{k2} & \cdot & \overline{Z}_{kk} \end{pmatrix} \quad [8]$$

Ou:

$$\overline{Z}_i = \max_{x \in D} Z_i(X) = Z_i(\overline{X}_j) \quad \forall i, j = \overline{1.k}$$

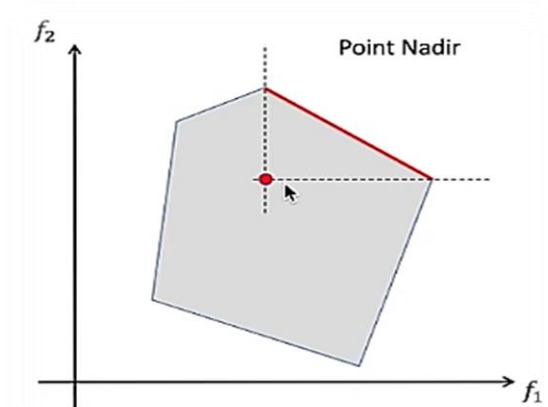
Avec  $Z_{ij} = Z_i(\overline{X}_j) \quad \forall i, j = \overline{1.k}$ .

est appelée matrice des gains (pay-off table).

**Définition 9: (point Nadir)**

Le point Nadir est défini comme la pire solution efficace, il s'écrit comme suit :

$V_j = \min_{x(p)} f_i(x); i, j = \overline{1.k}$  tel que  $p$  est l'ensemble des solutions efficaces.



**Figure 10:Point Nadir**

**Remarque:**

Notons qu'un tel point n'est pas facile à calculer c'est pour cela qu'il est estimé comme le

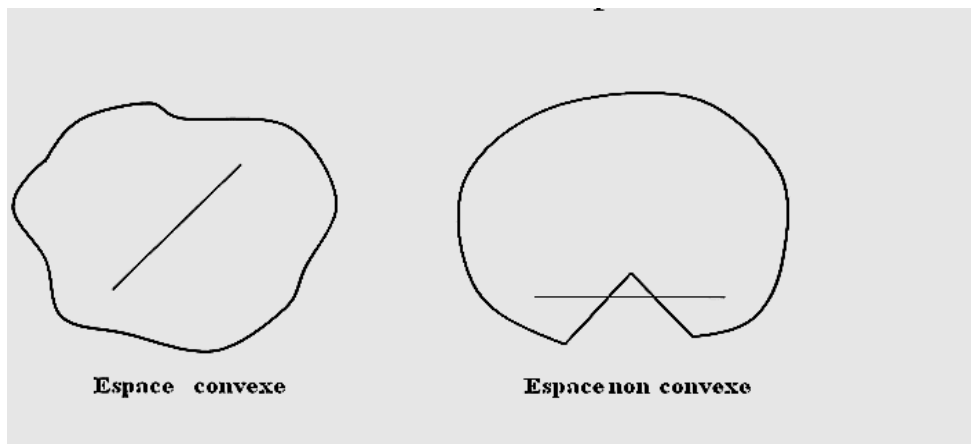
point de coordonnées :  $n_i = \min_{j=\overline{1.k}} \overline{Z}_{ij}, i = \overline{1.k}$

**Définition 10: (la convexité)**

Un ensemble  $A$  est convexe si et seulement si l'équivalence suivante est vérifiée :

$$x \in A \text{ et } y \in A \Leftrightarrow \text{segment}[x, y] \subset A.$$

La convexité est le premier indicateur de la difficulté du problème. En effet plusieurs méthodes d'optimisation sont incapables de résoudre d'une façon optimale des problèmes non convexes, mais il existe d'autres indicateurs tout aussi important, notamment la continuité, la nature des variables de décision (entières ou réelles) ....



**Figure 11: Espace convexe et espace non convexe**

**Caractérisation des solutions efficaces:**

Nous présentons quelques caractéristiques qui permettent de tester l'efficacité d'une solution réalisable:

- **Théorème d'Eker et Wendell:**

Soit  $x^* \in D$  un vecteur de décision donné et  $\tilde{x}$  un vecteur quelconque de  $D$   
 $x^*$  est efficace pour (MOP)  $\Leftrightarrow x^*$  est solution optimale de problème auxiliaire:

$$(P_{\tilde{x}}) \begin{cases} \min \sum_{i=1}^k f_i(x) \\ x \in D \\ f_i(x) \leq f_i(\tilde{x}) \forall i = \overline{1, k} \end{cases}$$

**Exemple 3.1:**

Soit le problème suivant :

$$(P) \begin{cases} \min(-3x_1 - x_2, x_1 + 2x_2) \\ 3x_1 - x_2 \leq 6 \\ x_2 \leq 2 \\ (x_1, x_2) \in \mathbb{R}_+^2 \end{cases}$$

On a pour  $\tilde{x} = (1.1)$  solution réalisable:

$$(P_{\tilde{x}}) \begin{cases} \min \sum_{i=1}^2 f_i(x) \\ 3x_1 - x_2 \leq 6 \\ x_2 \leq 2 \\ (x_1, x_2) \in \mathbb{R}_+^2 \end{cases}$$

$$f_1(x) \leq f_1(\tilde{x}) = f_1(1.1) \Rightarrow -3x_1 - x_2 \leq -4$$

$$f_2(x) \leq f_2(\tilde{x}) = f_2(1.1) \Rightarrow x_1 + 2x_2 \leq 3$$

En résolvant  $(P_{\tilde{x}})$  par la méthode de simplexe, sa solution optimale  $x^* = (2.0)$

Donc  $x^*$  est efficace pour  $P$ .

- **Théorème de Benson:** Soit le problème uni-critère suivant :

$$(P_{\varepsilon}(x^*)) \begin{cases} \max \theta = \sum_{i=1}^k \varepsilon_i \\ x \in D \\ f_i(x) + \varepsilon_i = f_i(x^*); \forall i = \overline{1, k} \\ \varepsilon_i \geq 0; \forall i = \overline{1, k} \end{cases}$$

Soit  $x^*$  une solution réalisable donnée, alors:

1.  $x^*$  est efficace pour le problème (MOP)  $\Leftrightarrow$  la valeur optimale de la fonction objectif est nulle dans  $(P_{\varepsilon}(x^*))$ .
2. Si  $\theta_{opt} \neq 0$  pour la solution  $\tilde{x}$  de  $(P_{\varepsilon}(x^*))$  alors :  $\tilde{x}$  est efficace pour (MOP).

**Exemple 3.2:** soit le problème multiobjectif (MOP) suivant:

$$(MOP) \left\{ \begin{array}{l} \min(x_1 - 2x_2; -x_1 + 4x_2; 2x_1 + x_2) \\ -2x_1 + x_2 \leq 0 \\ x_1 \leq 3 \\ x_2 \leq 2 \\ (x_1, x_2) \in \mathbb{R}_+^2 \end{array} \right.$$

Vérifions si la solution réalisable  $x_1=(0.0)$ est efficace en utilisant le test de Benson:

$$(P_\varepsilon(x^*)) \left\{ \begin{array}{l} \max \theta = \varepsilon_1 + \varepsilon_2 + \varepsilon_3 \\ -2x_1 + x_2 \leq 0 \\ x_1 \leq 3 \\ x_2 \leq 2 \\ f_1(x) + \varepsilon_1 = f_1(x_1) \Rightarrow x_1 - 2x_2 + \varepsilon_1 = 0 \\ f_2(x) + \varepsilon_2 = f_2(x_1) \Rightarrow -x_1 + 4x_2 + \varepsilon_2 = 0 \\ f_3(x) + \varepsilon_3 = f_3(x_3) \Rightarrow 2x_1 + x_2 + \varepsilon_3 = 0 \\ x_1, x_2, \varepsilon_1, \varepsilon_2, \varepsilon_3 \geq 0 \end{array} \right.$$

la solution optimale de  $(P_\varepsilon(x^*))$  est  $x_1=0, x_2=0, \varepsilon_1=0, \varepsilon_2=0, \varepsilon_3 = 0$

$\Rightarrow \theta_{opt}=0 \Rightarrow x_1 = (0.0)$  est efficace pour (MOP).

### 3.3. Méthodes de résolution d'un programme linéaire multi-objectifs

#### 3.3.1. Méthode de simplexe multi-objectifs[13]:

La méthode du simplexe multicritères consiste à générer un premier point efficace à partir d'une solution de base réalisable, puis à énumérer tous les autres points efficaces cependant cette méthode ne teste pas toutes les bases car certaines sont dominées de manière évidente.

Etant donné le problème :

$$\text{MOP} \begin{cases} \max Z_1 = c^1 x \\ \max Z_2 = c^2 x \\ \vdots \\ \max Z_k = c^k x \\ Ax = b, x \geq 0 \end{cases}$$

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

Le tableau du simplexe (en considérant séparément chaque objectif) est donné par le tableau (1).

Dans ce tableau, on suppose sans perte de généralité que les  $m$  première variables sont dans la base, soient:

$J_B$  = Ensemble des indices basiques.

$J_H$  = Ensemble des indices hors base.

					$C^1$	$C_1^1$	.....	$C_m^1$	$C_{m+1}^1$	.....	$C_j^1$	.....
					$C^2$	$C_1^2$	.....	$C_m^1$	$C_{m+1}^2$	.....	$C_j^2$	.....
					$\vdots$				$\vdots$		$\vdots$	
					$C^k$	$C_1^k$	.....	$C_m^k$	$C_{m+1}^k$	.....	$C_j^k$	.....
$C_B^1$	$C_B^2$	$C_B^j$	$C_B^k$	base	B	$a_1$	.....	$a_m$	$a_{m+1}$	.....	$a_j$	.....
$C_1^1$	$C_1^2$	$C_1^j$	$C_1^k$	$a_1$	$b_1$	1		0	$x_{1m+1}$	.....	$x_{1j}$	.....
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$		0	$\vdots$	0	$\vdots$		$\vdots$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$		0	$\vdots$	0	$\vdots$		$\vdots$	
$C_m^1$	$C_m^2$	$C_m^j$	$C_m^k$	$a_m$	$b_m$	0		1	$x_{mm+1}$	.....	$x_{mj}$	.....
						0	0	0	$\Delta_{m+1}^1$		$\Delta_j^1$	
						$\vdots$	0	0	$\Delta$			
						0	0	$\vdots$	$\Delta_{m+1}^i$	.....	$\Delta_j^i$	.....
						$\vdots$	0	$\vdots$	$\vdots$			
						0	0	0	$\Delta_{m+1}^k$	.....	$\Delta_j^k$	.....

Figure 12: Tableau 1 de simplexe multi-objectifs

$$Z_0 = \begin{pmatrix} Z_0^1 = c^1 b \\ Z_0^2 = c^2 b \\ \vdots \\ Z_0^k = c^k b \end{pmatrix}$$

**Le test d’efficacité:**

**Théorème 1:**

Soit  $\bar{x} \in D$  est efficace (respectivement faiblement efficace) pour le (MOP) (linéaire).

Si et seulement s’il existe :  $\lambda \in \Lambda = \{ \lambda \in \mathbb{R}^k / \sum_{j=1}^k \lambda_j = 1, \lambda_i > 0 \}$

Respectivement  $\exists \lambda \in \bar{\Lambda} = \{ \lambda \in \mathbb{R}^k / \sum_{j=1}^k \lambda_j = 1, \lambda_i \geq 0 \}$  tel que  $\bar{x}$  maximise le problème des sommes pondérées donné par:

$$P_\lambda \begin{cases} \max \lambda' c' x \\ s.c \quad x \in D \end{cases}$$

**Théorème 2:**

Si  $D$  possède un point efficace, alors au moins un point extrême de  $D$  est efficace.

**Preuve:**

Soit  $\bar{x}$  un point efficace de  $D$ , d’après le théorème 1 il existe  $\lambda \in \Lambda$  tel que

$$\lambda' c' \bar{x} = \max_{x \in D} \lambda' c' x$$

Comme une fonction linéaire atteint son maximum en un point extrême, donc  $\bar{x}$  est extrême efficace.

**Théorème 3:**

soit  $x \in D$  un point extrême associé à une base efficace  $B$  alors  $x$  est efficace .

**Preuve:**

Puisqu’il existe un  $\lambda \in \Lambda$  pour lequel  $B$  est une base optimale par le théorème 2,  $x$  est efficace.

**Theoreme4:**

Soient  $B$  et  $\bar{B}$  deux bases efficace adjacentes obtenues à partir d'un pivot efficace, et soient  $x$  et  $\bar{x}$  les points extrêmes associés à  $B$  et  $\bar{B}$  respectivement. Alors l'arête  $(x, \bar{x})$  est efficace.

**Théorème 5 [7]:**

Soit  $(x, \nu)$  une arête efficace infinie de  $D$ , alors  $x$  est un point extrême associé à une base efficace  $B$ .

De la théorie de simplexe uni-critères on a:

1.

$$\Delta_j^i = \sum_{r \in J_B} c_r^i x_{rj} - c_j^i, j \in J_H, \forall i = \overline{1.k} \text{ et } Z_0^i = \sum_{r \in J_B} c_r^i b_r, \forall r = \overline{1.k}.$$

Si  $\Delta_j^i \geq 0 \forall j \in J_H$  alors  $x^0 = (b, 0)'$  ( $b \in \mathbb{R}^{m+}, 0 \in \mathbb{R}^{n-m}$ ) est une solution maximale pour le critère  $i$ .

2. si on introduit la  $J^{\text{eme}}$  variable dans la base nous obtenons une nouvelle solution  $x_1$  et un nouveau vecteur  $\hat{Z}_0 = Z_0 - \theta_j \Delta_j$  ou:

$$\hat{Z}_0 = \begin{pmatrix} \hat{Z}_0^1 \\ \hat{Z}_0^2 \\ \vdots \\ \hat{Z}_0^k \end{pmatrix} = \begin{pmatrix} Z_0^1 \\ Z_0^2 \\ \vdots \\ Z_0^k \end{pmatrix} - \theta_j \begin{pmatrix} \Delta_j^1 \\ \Delta_j^2 \\ \vdots \\ \Delta_j^k \end{pmatrix}$$

3.

$$\theta_j = \min \left\{ \frac{b_r}{x_{rj}} / x_{rj} > 0 \right\} \forall j \in J_H, r \in J_B.$$

**Remarques:**

1) Soit  $x^0$  une solution basique réalisable.

a. S'il existe un  $j \in J_H$  tel que tous les  $\Delta_j^i \leq 0$  avec au moins une inégalité stricte et si  $\theta_j > 0$ , alors la solution courante  $x^0$  est dominée.

En effet:

Si on introduit la  $J^{\text{eme}}$  variable a la base ,on obtient un point extrême adjacent  $x^1$  pour lequel  $\hat{Z}_0 \geq Z_0$  avec au moins une inégalité stricte ,car :

$$\theta_j \Delta_j \leq 0 \text{ donc } \hat{Z}_0 = Z_0 - \theta_j \Delta_j \geq Z_0.$$

- b. S'il existe un  $j \in J_H$  tel que  $\Delta_j^i \geq 0$  avec au moins une inégalité stricte et si de plus  $\theta_j > 0$ , alors l'introduction de la  $J^{\text{eme}}$  variable de base mène à une solution dominée.

En effet:

si on introduit la variable  $j$  dans la base, on aura un nouveau point extrême  $x^1$  pour lequel :  $\hat{Z}_0 = Z_0 - \theta_j \Delta_j \leq Z_0$  avec au moins une inégalité stricte.

- 2) soit  $x^\circ$  une solution basique réalisable s'il existe  $j_1, j_2 \in J_H$  tel que:

$\theta_{j_1} \Delta_{j_1} \leq \theta_{j_2} \Delta_{j_2}$  avec au moins une inégalité stricte ,alors l'introduction de la variable d'indice  $j_2$  dans la base conduit à une solution dominée par celle résultant de l'introduction de la variable d'indice  $j_1$ .

En effet:

$$\theta_{j_1} \Delta_{j_1} \leq \theta_{j_2} \Delta_{j_2} \text{ avec au moins une inégalité stricte ,cela implique que}$$

$$Z_0 - \theta_{j_2} \Delta_{j_2} \leq Z_0 - \theta_{j_1} \Delta_{j_1} \text{ Avec au moins une inégalité stricte.}$$

### Résumé de la méthode:

1. S'il existe un indice  $i \in \{1...k\}$  tel que  $\Delta_j^i \geq 0 \forall j \in J_H$  , alors le  $i^{\text{eme}}$  critère à son maximum et la solution basique correspondante est non dominée à condition qu'il n'existe pas de colonne  $k$  avec  $\Delta_k^i = 0$ .
2. D'après les deux remarques précédentes, seules les colonnes (variables) non comparables à zéro et les variables  $j, k \in J_H$  tel que :  $\theta_k \Delta_k$  incomparable à  $\theta_j \Delta_j$  sont admissible pour une introduction dans la base.

Dans ce cas, on ne peut dire si la solution correspondante  $\bar{x}$  est dominée ou pas. Pour cela, on considère le test dit de dominance énoncé par le théorème :

**Théorème (test de dominance):**

soit le problème:

$$\begin{aligned} \max v &= \sum_{i=1}^k \varepsilon_i \\ c'x - \delta &= c'\bar{x} \\ x \in D &= \{ x \in \mathbb{R}^n / Ax = b, x \geq 0 \} \\ \delta \in \mathbb{R}^n \quad \delta &\geq 0, \bar{x} \in D \end{aligned}$$

**Alors:**

- $\bar{x}$  est efficace si et seulement si :  $\max v = 0$ .
- $\bar{x}$  est dominée si et seulement si :  $\max v > 0$ .

**Exemple 3.3:**

Soit le problème:  $\max Z_1 = 0.4x_1 + 0.3x_2$

$\max Z_2 = x_1$

$$\begin{cases} x_1 + x_2 \leq 400 \\ 2x_1 + x_2 \leq 500 \\ x_1 \geq 0; x_2 \geq 0 \end{cases} \quad (1)$$

(1)  $\Leftrightarrow$   $\begin{aligned} \max Z_1 &= 0.4x_1 + 0.3x_2 \\ \max Z_2 &= x_1 \end{aligned}$

$$\begin{cases} x_1 + x_2 + x_3 = 400 \\ 2x_1 + x_2 + x_4 = 500 \\ x_j \geq \forall j = \overline{1,4} \end{cases}$$

$x_1 = (0,0,400,500)'$  est une solution de base réalisable associé a la base

Le tableau de simplexe initial est:

			$C^1$	0.4	0.3	0	0
			$C^2$	1	0	0	0
$C_B^1$	$C_B^2$	Base	$B$	$a_1$	$a_2$	$a_3$	$a_4$
0	0	$a_3$	400	1	1	1	0
0	0	$a_4$	500	2	1	0	1 →
$Z_0 = \begin{pmatrix} Z_0^1 = 0 \\ Z_0^2 = 0 \end{pmatrix}$			$\Delta^1$	-0.4	-0.3	0	0
			$\Delta^2$	-1	0	0	0

Il n'existe pas un  $i \in \{1,2\}$  tel que  $\Delta_j^i \geq 0 \forall j \in J_H = \{1,2\}$  donc aucun objectif n'est à son maximum.

$$\Delta_1 \begin{pmatrix} -0.4 \\ -1 \end{pmatrix} \leq 0 \Rightarrow x_1 = (0,0,400,500) \text{ est dominé.}$$

$$\theta_1 = \min \left\{ \frac{400}{1}, \frac{500}{2} \right\} = 250 \text{ (atteint pour } r=4).$$

$$\theta_2 = \min \left\{ \frac{400}{1}, \frac{500}{1} \right\} = 400 \text{ (atteint pour } r=3).$$

$$\left. \begin{aligned} \theta_1 \Delta_1 &= 250 \begin{pmatrix} -0.4 \\ -1 \end{pmatrix} = \begin{pmatrix} -100 \\ -250 \end{pmatrix} \\ \theta_2 \Delta_2 &= 100 \begin{pmatrix} -0.3 \\ 0 \end{pmatrix} = \begin{pmatrix} -120 \\ 0 \end{pmatrix} \end{aligned} \right\} \text{ donc } \theta_1 \Delta_1 \text{ est incomparable à } \theta_2 \Delta_2.$$

Introduisons  $a_1$  dans la base à la place de  $a_4$ , on aura alors le tableau suivant:

			$C^1$	0.4	0.3	0	0
			$C^2$	1	0	0	0
$C_B^1$	$C_B^2$	Base	$B$	$a_1$	$a_2$	$a_3$	$a_4$
0	0	$a_3$	150	0	1/2	1	-1/2
0.4	1	$a_1$	250	1	1/2	0	1/2
		$Z_0 = \begin{pmatrix} Z_1^1 = 100 \\ Z_1^2 = 250 \end{pmatrix}$	$\Delta^1$	0	-0.1	0	0.2
			$\Delta^2$	0	1/2	0	1/2

Il existe un  $i=2$  tel que tel que  $\Delta_j^i > 0 \forall j \in J_H$  ce qui implique que le deuxième critère est à son maximum.

Le point extrême  $x_2 = (250, 0, 150, 0)$  est non dominé.

$$\theta_2 = \min \left\{ \frac{150}{0.5}, \frac{250}{0.5} \right\} = 300 \text{ (atteint pour } r=3).$$

$$\theta_4 = \min \left\{ \frac{250}{0.5} \right\} = 500 \text{ (atteint pour } r=1).$$

$$\theta_2 \Delta_2 = 300 \begin{pmatrix} -0.1 \\ 0.5 \end{pmatrix} = \begin{pmatrix} -30 \\ 150 \end{pmatrix} .$$

$$\theta_4 \Delta_4 = 500 \begin{pmatrix} 0.2 \\ 0.5 \end{pmatrix} = \begin{pmatrix} 100 \\ 250 \end{pmatrix} .$$

$$\theta_2 \Delta_2 < \theta_4 \Delta_4$$

Donc  $a_2$  rentre dans la base et  $a_3$  sort de la base d'après la remarque, dressons alors le 3<sup>ème</sup> tableau de simplexe:

			$C^1$	0.4	0.3	0	0
			$C^2$	1	0	0	0
$C_B^1$	$C_B^2$	Base	$B$	$a_1$	$a_2$	$a_3$	$a_4$
0.3	0	$a_2$	300	0	1	2	-1
0.4	1	$a_1$	100	1	0	-1	1 →
		$Z_0 = \begin{pmatrix} Z_1^1 = 130 \\ Z_1^2 = 100 \end{pmatrix}$	$\Delta^1$	0	0	0.2	0.1
			$\Delta^2$	0	0	-1	1 ↑

$\Delta^1 \geq 0$ , alors le premier objectif est à son maximum et puisqu'il n'existe pas de colonne  $k$  telle que  $\Delta_k^1 = 0$  avec  $k \in J_H$  alors  $x_3 = (100, 300, 0, 0)$  est non dominée.

$$\theta_3 = \min \left\{ \frac{300}{2} \right\} = 150 \text{ (atteint pour } r = 2).$$

$$\theta_4 = \min \left\{ \frac{100}{1} \right\} = 100 \text{ (atteint pour } r = 1).$$

$$\theta_3 \Delta_3 = 150 \begin{pmatrix} 0.2 \\ -1 \end{pmatrix} = \begin{pmatrix} 30 \\ -150 \end{pmatrix} .$$

$$\theta_4 \Delta_4 = 100 \begin{pmatrix} 0.1 \\ 1 \end{pmatrix} = \begin{pmatrix} 10 \\ 100 \end{pmatrix} .$$

On remarque que  $\theta_3 \Delta_3$  est incomparable à  $\theta_4 \Delta_4$  .

Puisque l'introduction de  $a_3$  dans la base conduit à une base déjà explorée, donc on introduira  $a_4$ .

Dressons alors le 4<sup>ème</sup> tableau de simplexe:

			$C^1$	0.4	0.3	0	0
			$C^2$	1	0	0	0
$C_B^1$	$C_B^2$	Base	$B$	$a_1$	$a_2$	$a_3$	$a_4$
0.3	0	$a_2$	400	1	1	1	0
0	0	$a_4$	100	1	0	-1	1
		$Z_0 = \begin{pmatrix} Z_1^1 = 120 \\ Z_1^2 = 0 \end{pmatrix}$	$\Delta^1$	-0.1	0	0.3	0
			$\Delta^2$	-1	0	0	0

Il n'existe pas un  $i \in \{1,3\}$  tel que  $\Delta_j^i \geq 0 \forall j \in J_H = \{1,3\}$  donc aucun objectif n'est à son maximum.

$$\Delta_1 \begin{pmatrix} -0.1 \\ -1 \end{pmatrix} < 0 \Rightarrow x_4 = (0,400,0,100) \text{ est dominé.}$$

Finalement, En considérant le problème initial (1) seuls les points extrêmes

$x_2 = (250,0)$  . Et  $x_3 = (100,300)$  sont non dominés avec:

$$Z(x_2^*) = \begin{pmatrix} 100 \\ 250 \end{pmatrix} \text{ et } Z(x_3^*) = \begin{pmatrix} 130 \\ 100 \end{pmatrix}$$

On a recensé deux solutions efficaces.

### Conclusion:

La méthode du simplexe multicritères nous apporte clarté et précision dans toutes ses étapes donc facilité d'application, cependant en pratique, elle peut être non efficace, car la solution désirée peut être sur arête et non un point extrême efficace.

### 3.3.2. Méthode de la somme pondérée ou méthode d'agrégation des objectifs (méthode d'agrégation par pondération):

Dans cette approche, le but consiste à ramener le problème multicritère à un problème monocritère plus simple à traiter.

Cette méthode est la plus simple des méthodes d'optimisation multiobjectif. La transformation que l'on effectue est la suivante:

$$(P_\lambda) \left\{ \begin{array}{l} \max \sum_{i=1}^k \lambda_i f_i(x) \\ x \in D \end{array} \right.$$

$D$ : représente le domaine réalisable.

$$\lambda_i \geq 0; i = \overline{1,k} ; \sum_{i=1}^k \lambda_i = 1.$$

$\lambda_i$ : appelé le poids, est une pondération associée au critère, cette pondération permet d'exprimer des préférences sur les critères de décisions.

Cette méthode consiste à additionner tous les objectifs en affectant à chacun d'eux un coefficient poids  $\lambda_i$ . Les poids  $\lambda_i$  reflètent l'importance relative de chaque critère  $i$  et servent en général à agréger les différents critères en une fonction unique.

**Théorème 1:**

Toute solution optimale de problème  $(P_\lambda)$  est une solution pareto-optimale pour le problème (MOP) si tous les  $\lambda_i$  sont strictement positifs  $i = \overline{1,k}$ .

**Théorème 2:**

Si le problème (MOP) est concave (en particulier linéaire) si  $x^*$  est solution pareto-optimale de (MOP) alors :  $\exists \lambda = (\lambda_1, \lambda_2, \dots, \lambda_k)'$  avec  $(\lambda_i > 0, i = \overline{1,k})$  chaque  $x^*$  est solution optimale de  $(P_\lambda)$ .

**Remarque:**

Les résultats obtenus dans la résolution du problème mono-objectif paramétrique  $(P_\lambda)$  dépendent largement des paramètres choisis pour le vecteur de poids  $\lambda$ . Aussi, les différents poids  $\lambda_i$  doivent être choisis en fonction des préférences associées aux différents objectifs.

Le principal avantage de cette approche réside dans le fait qu'elle garantit l'optimalité Pareto de la solution trouvée.

**Exemple 3.4:** Soit le problème bi-objectifs suivant:

$$(P) \left\{ \begin{array}{l} \max(Z_1) = 4x_1 + 6x_2 \\ \max(Z_2) = 3x_3 \\ 2x_1 + 3x_2 \geq 12 \\ 5x_1 + 2x_2 \leq 15 \\ x_1 + x_3 = 10 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0 \end{array} \right.$$

La transformation de problème bi-objectifs (P) en un problème mono-objectifs par la méthode d'agrégation linéaire donne le problème ( $P_\lambda$ ):

$$(P_\lambda) \left\{ \begin{array}{l} \max(Z) = \lambda_1 Z_1 + \lambda_2 Z_2 \\ 2x_1 + 3x_2 \geq 12 \\ 5x_1 + 2x_2 \leq 15 \\ x_1 + x_3 = 10 \\ x_1 \geq 0; x_2 \geq 0; x_3 \geq 0 \end{array} \right.$$

On fixe  $\lambda_1 = 0.8$  et  $\lambda_2 = 0.2$  ( par priorité des objectifs).

On obtient:

$$(P_\lambda) \left\{ \begin{array}{l} \max(Z) = 0.8(4x_1 + 6x_2) + 0.2(3x_3) \\ 2x_1 + 3x_2 - x_4 + x_6 = 12 \\ 5x_1 + 2x_2 + x_5 = 15 \\ x_1 + x_3 = 10 \\ x_j \geq 0, j = \overline{1,6} \end{array} \right.$$

Dressons le premier tableau de simplexe:

**1<sup>ère</sup> phase:**

	C	0	0	0	0	0	-1	
Base	$b_i$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$\theta_j$
$a_6$	12	2	3	0	-1	0	1	12/3
$a_5$	15	5	2	0	0	1	0	15/2
$a_3$	10	1	0	1	0	0	0	/
	$\Delta_j$	-2	-3	0	1	0	0	

$a_2$  va rentrer dans la base et  $a_6$  va sortir de la base

Passons au deuxième tableau de simplexe:

	C	0	0	0	0	0	-1	
Base	$b_i$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$\theta_j$
$a_2$	4	2/3	1	0	-1/3	0	1/3	
$a_5$	7	11/3	0	0	2/3	1	-2/3	
$a_3$	10	1	0	1	0	0	0	
	$\Delta_j$	0	0	0	0	0	0	

**La 2<sup>eme</sup> phase:**

Dressons le troisième tableau:

	C	3.2	4.8	0.6	0	0	
Base	$b_i$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$\theta_j$
$a_2$	4	2/3	1	0	-1/3	0	/
← $a_5$	7	11/3	0	0	2/3	1	21/2
$a_3$	10	1	0	1	0	0	/
	$\Delta_j$	0.6	0	0	-1/6	0	

Passons au 4<sup>eme</sup> tableau:

	C	3.2	4.8	0.6	0	0
Base	$b_i$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
$a_2$	7.5	15/6	1	0	0	1/2
$a_4$	10.5	11/2	0	0	1	3/2
$a_3$	10	1	0	1	0	0
	$\Delta_j$	9.4	0	0	0	2.4

On remarque que les  $\Delta_j$  sont tous positifs ou nuls, Alors la solution

$x^*=(x_2,x_3,x_4)=(7.5,10.5,10)$  est optimale pour  $(P_\lambda)$  avec  $Z=42$  et elle est efficace pour (P) avec  $Z_1=45$  et  $Z_2=30$ .

### 3.3.3 méthode $\varepsilon$ -contraintes:

Elle est aussi dite méthode de compromis, elle transforme un problème d'optimisation multiobjectif en un problème d'optimisation mono-objectif de la façon suivante:

- Choisir un objectif à optimiser prioritairement.
- Choisir un vecteur de contraintes initial  $\varepsilon$ .
- Transformer le problème en gardant l'objectif prioritaire et en transformant les autres objectifs en contraintes d'inégalités comme suite:

$$(p_\varepsilon) \begin{cases} \max f_i(x) \\ x \in D \\ f_j(x) \geq \varepsilon_j, \forall i \neq j, (j = \overline{1..k}) \end{cases}$$

Les  $\varepsilon_j \in \mathbb{R}$  sont appelés « seuil de satisfaction »

Pour un problème (MOP) a minimum:

$$(P_\varepsilon) \begin{cases} \min f_i(x) \\ x \in D \\ f_j(x) \leq \varepsilon_j, \forall i \neq j, (j = \overline{1..k}) \end{cases}$$

#### **Théorème 3:**

Pour tout vecteur  $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{j-1}, \varepsilon_{j+1}, \dots, \varepsilon_k)'$  tel que le problème  $(p_\varepsilon)$  a une solution optimale alors cette solution est pareto-optimale pour le problème (MOP).

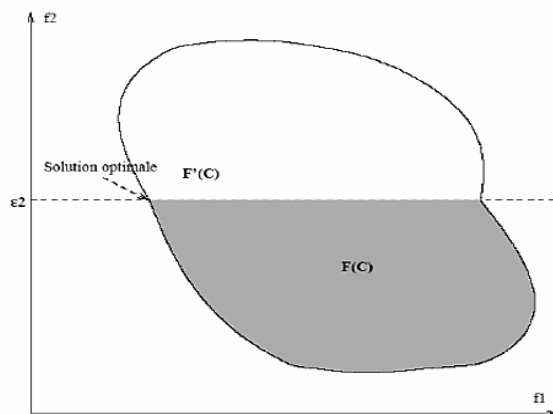
#### **Remarques:**

1. les paramètres  $\varepsilon_j$  sont choisis par le décideur.
2. en faisant varier les  $\varepsilon_j$  dans les deux méthodes, on peut générer un sous-ensemble de l'ensemble pareto du (MOP).

Cette approche a l'avantage par rapport à la précédente dans les problèmes non convexes, mais présente plusieurs inconvénients à savoir:

- La formulation des préférences utilisateurs est délicate et nécessite une connaissance approfondie du problème de départ.

- Les contraintes rajoutées compliquent la résolution de problème.



**Figure 13 :la méthode de  $\epsilon$ -contraintes pour un problème de maximisation**

La figure 13 illustre la méthode de  $\epsilon$ -contraintes pour un problème de maximisation bi-objectifs ou  $F(c)$  est l'espace de problème original , restreint à  $F'(C)$  par la transformation du problème en maximisant la fonction  $f_1$  et en rajoutant la contrainte  $f_2(x) \geq \epsilon$  .

**Exemple 3.5:**

Résolution de problème (MOP) suivant avec la méthode  $\epsilon$ -contraintes en considérant le critère 3 comme prioritaire et  $\epsilon = (\epsilon_1, \epsilon_2) = (1, 3)$ .

$$\text{MOP} \left\{ \begin{array}{l} \min f(x) = (f_1(x) = -3x_1 + x_2; f_2(x) = x_1 + 2x_2; f_3(x) = 2x_1 + x_2) \\ 3x_1 - x_2 \leq 6 \\ x_2 \leq 2 \\ x = (x_1, x_2) \in \mathbb{R}_+^2 \end{array} \right.$$

$$(P_\epsilon) \left\{ \begin{array}{l} \min f_{i=3}(x) \\ x \in D \\ f_j(x) \leq \epsilon_j, \forall j \neq 3, j = \overline{1,3} \end{array} \right.$$

$$(P_\varepsilon) \left\{ \begin{array}{l} \min 2x_1 + x_2 \\ 3x_1 - x_2 \leq 6 \\ x_2 \leq 2 \\ f_1(x) \leq \varepsilon_1 \rightarrow -3x_1 + x_2 \leq 1 \\ f_2(x) \leq \varepsilon_2 \rightarrow x_1 + 2x_2 \leq 3 \\ x = (x_1, x_2) \in \mathbb{R}_+^2 \end{array} \right.$$

En résolvant  $(P_\varepsilon)$  par le simplexe ,sa résolution optimale  $x^* = (x_1, x_2) = (0,0)$ ,donc d'après le théorème 3,  $x^* = (0,0)$  est pareto-optimale pour le problème (MOP).

### 3.3.4 La méthode « but à atteindre » [14]:

Contrairement à la méthode de pondération de fonctions objectifs, cette méthode n'impose pas de travailler sur un domaine réalisable convexe, la démarche suivie est la suivante:

- On choisit un vecteur de fonctions objectifs initial  $F$  .
- On choisit une direction de recherche (on fournit en quelque sorte des Coefficients de pondération, comme pour la méthode de pondération des fonctions objectifs)  $w$  .
- On cherche ensuite à minimiser un coefficient scalaire  $\lambda$  qui représente l'écart par rapport à l'objectif initial  $F$  que l'on s'était fixé.

On part du problème(P), on choisit un vecteur de fonction objectif initial  $F \in \mathbb{R}^p$ , un ensemble des coefficients  $w_k, k \in \{1 \dots p\}$  et l'on désigne par  $\lambda$  un nombre scalaire que l'on cherche à minimiser.

Le problème (P) devient alors:

$$P \left\{ \begin{array}{l} \min \lambda \\ f_1(x) - w_1 \lambda \leq F_1 \\ \vdots \\ f_p(x) - w_p \lambda \leq F_p \\ x \in D \end{array} \right.$$

Le principal avantage de cette méthode est qu'elle permet de travailler avec un ensemble de solutions pas nécessairement convexe.

Cependant, il peut exister des formes de domaines pour lesquelles la solution dépend de choix de but initial.

Cette méthode comme les précédentes doit être itérée plusieurs fois dans le but d'obtenir un ensemble de points pareto-optimaux ,les paramètres  $\vec{w}$  et F doivent être bien choisis par l'utilisateur .Bien que ces paramètres permettent une grande flexibilité de la recherche (orientation et but), s'ils sont mal choisis ,ils peuvent dans certaines cas extrêmes donner des résultats non cohérents.

### **Conclusion:**

Dans les trente dernières années, la plupart des travaux réalisés dans le domaine de la recherche opérationnelle ont porté sur la programmation linéaire multiobjectifs. Les raisons principales de cet intérêt sont d'une part le développement de la programmation linéaire mono-objectif en recherche opérationnelle et la facilité relative de traiter de tels problèmes, et d'autre part l'abondance de cas pratiques pouvant être formulés sous cette forme.

## **CHAPITRE 4:**

# **Etude des cas réels en programmation linéaire multi-objectifs**

## Introduction:

La programmation linéaire multi-objectif occupe une place très importante en recherche opérationnelle et en mathématiques discrètes. Son importance se justifie d'une part par la grande difficulté des problèmes associés et d'autre part par le nombre important d'applications réelles pouvant être modélisées sous sa forme : tel que le problème de production, le problème de mélange alimentaire, le problème de transport,..., etc. En effet la plupart de ces problèmes appartiennent à la classe des problèmes NP-difficiles. Dans ce chapitre nous exposerons la formulation de ces problèmes avec des exemples réels.

### 4.1 Premier cas : Problème de production:

1. Une usine qui produit deux pièces  $M_1$  et  $M_2$  sur une machine, ces produits sont tellement demandés qu'ils sont tous vendus.

- Les prix de vente sont de 3DA pour  $M_1$  et 1DA pour  $M_2$ .
- Le bénéfice reçu de la vente d'une pièce  $M_1$  est 0.6DA.
- Le bénéfice reçu de la vente d'une pièce  $M_2$  est 0.3DA.
- La production de ces produits demande 2 unités de matière première pour  $M_1$  et 1 unité pour  $M_2$ , et aussi 2min de temps de fonctionnement de la machine pour  $M_1$  et 3min pour  $M_2$ .

On veut calculer le nombre de pièces à produire pour que:

- ✓ Le chiffre d'affaire soit maximal.  
Et au même temps
- ✓ Le bénéfice soit maximal.

Sachant que le stock de matière première est de 100 unités et la disponibilité de la machine est de 3heures et 20min (200min).

Soit  $x_1$  le nombre de pièces  $M_1$  produites et  $x_2$  le nombre de pièces  $M_2$  produites.

Le bénéfice est exprimé par  $Z_1 = 0.6 x_1 + 0.3 x_2$ .

Le chiffre d'affaire est exprimé par  $z_2 = 3 x_1 + x_2$ .

Le problème multicritère se traduit comme suit:

$$\text{Max}Z_1 = 0.6x_1 + 0.3x_2$$

$$\text{Max}Z_2 = 3x_1 + x_2$$

$$\begin{cases} 2x_1 + x_2 \leq 100 \\ 2x_1 + 3x_2 \leq 200 \\ x_1 \geq 0, x_2 \geq 0 \end{cases} \quad (1)$$

$$(1) \Leftrightarrow \text{Max}Z_1 = 0.6x_1 + 0.3x_2$$

$$\text{Max}Z_2 = 3x_1 + x_2$$

$$\begin{cases} 2x_1 + x_2 + x_3 = 100 \\ 2x_1 + 3x_2 + x_4 = 200 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0 \end{cases}$$

$x_1 = (0,0,100,200)^t$  est une solution de base réalisable associée à la base  $(a_3, a_4) = I_2$  et

$$Z(x) = \begin{pmatrix} Z_1(x) \\ Z_2(x) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Le tableau du simplexe initial est:

			$C^1$	0.6	0.3	0	0
			$C^2$	3	1	0	0
$C_B^1$	$C_B^2$	<i>Base</i>	<i>B</i>	$a_1$	$a_2$	$a_3$	$a_4$
0	0	$a_3$	100	2	1	1	0 →
0	0	$a_4$	200	2	3	0	1
		$Z_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$	$\Delta^1$	-0.6	-0.3	0	0
			$\Delta^2$	-3	-1	0	0

Il n'existe pas un  $i \in \{1,2\}$  tel que  $\Delta_j^i \geq 0; \forall j \in J_N = \{1,2\}$ , donc aucun objectif n'est a son maximum.

$$\Delta_1 = \begin{pmatrix} -0.6 \\ -3 \end{pmatrix} \leq 0, x_1 \text{ est dominé.}$$

$$\theta_1 = \text{Min}\{100/2, 200/2\} = 50. \text{ (atteint pour } r=3).$$

$$\theta_2 = \text{Min}\{100/1, 200/3\} = 200/3. \text{ (atteint pour } r=4).$$

$$\theta_1 \Delta_1 = 50 \begin{pmatrix} -0.6 \\ -3 \end{pmatrix} = \begin{pmatrix} -30 \\ -150 \end{pmatrix}.$$

$$\theta_2 \Delta_2 = 200/3 \begin{pmatrix} -0.3 \\ -1 \end{pmatrix} = \begin{pmatrix} -20 \\ -200/3 \end{pmatrix}.$$

$$\theta_1 \Delta_1 \leq \theta_2 \Delta_2.$$

Introduisons  $a_1$  dans la base à la place de  $a_3$ , on aura alors le tableau suivant:

			$C^1$	0.6	0.3	0	0
			$C^2$	3	1	0	0
$C_B^1$	$C_B^2$	Base	B	$a_1$	$a_2$	$a_3$	$a_4$
0.6	3	$a_1$	50	1	1/2	1/2	0
0	0	$a_4$	100	0	2	-1	1 →
		$Z_0 = \begin{pmatrix} 30 \\ 150 \end{pmatrix}$	$\Delta^1$	0	0	3/10	0
			$\Delta^2$	0	1/2	3/2	0

$\exists i = 2$  tel que  $\Delta_j^i > 0$  tel que  $\forall j \in J_N$ , ce qui implique que le deuxième critère est à son maximum.

Donc, le point extrême  $x_2 = (50, 0, 0, 100)'$  est candidate.

$$\theta_2 = \text{Min}\{50/(1/2), 100/2\} = 50 \text{ (atteint pour } r=4 \text{)}.$$

$$\theta_3 = \text{Min}\{50/(1/2)\} = 100 \text{ (atteint pour } r=1 \text{)}.$$

$$\theta_2 \Delta_2 = 50 \begin{pmatrix} 0 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 0 \\ 25 \end{pmatrix}.$$

$$\theta_3 \Delta_3 = 100 \begin{pmatrix} 3/10 \\ 3/2 \end{pmatrix} = \begin{pmatrix} 30 \\ 150 \end{pmatrix}.$$

$$\theta_2 \Delta_2 \leq \theta_3 \Delta_3.$$

Donc  $a_2$  rentre dans la base et  $a_4$  sort de la base.

Dressons alors le troisième tableau simplexe:

			$C^1$	0.6	0.3	0	0
			$C^2$	3	1	0	0
$C_B^1$	$C_B^2$	Base	B	$a_1$	$a_2$	$a_3$	$a_4$
0.6	3	$a_1$	25	1	0	3/4	-1/4 →
0.3	1	$a_2$	50	0	1	-1/2	1/2
		$Z_0 = \begin{pmatrix} 30 \\ 125 \end{pmatrix}$	$\Delta^1$	0	0	3/10	0
			$\Delta^2$	0	0	7/4	-1/4

Il n'existe pas un  $i \in \{1,2\}; \forall \Delta_j^i > 0$  tel que  $\forall j \in J_N = \{1,2\}$ , donc aucun objectif n'est à son maximum.

$$\theta_3 = \text{Min}\{25/(3/4)\} = 100/3 \text{ (atteint pour } r = 1).$$

$$\theta_4 = \text{Min}\{50/(1/2)\} = 100 \text{ (atteint pour } r = 2).$$

$$\theta_3 \Delta_3 = 100/3 \begin{pmatrix} 3/10 \\ 7/4 \end{pmatrix} = \begin{pmatrix} 10 \\ 700/12 \end{pmatrix}.$$

$$\theta_4 \Delta_4 = 100 \begin{pmatrix} 0 \\ -1/4 \end{pmatrix} = \begin{pmatrix} 0 \\ -25 \end{pmatrix}.$$

$$\theta_4 \Delta_4 \leq \theta_3 \Delta_3.$$

Donc  $a_3$  rentre dans la base et  $a_1$  sort de la base.

Dressons alors le quatrième tableau simplexe:

			$C^1$	0.6	0.3	0	0
			$C^2$	3	1	0	0
$C_B^1$	$C_B^2$	<i>Base</i>	<i>B</i>	$a_1$	$a_2$	$a_3$	$a_4$
0	3	$a_3$	100/3	4/3	0	1	-1/3
0.3	1	$a_2$	200/3	2/3	1	0	1/3
		$Z_0 = \begin{pmatrix} 30 \\ 175 \end{pmatrix}$	$\Delta^1$	-4/10	0	0	1/10
			$\Delta^2$	7/3	0	0	1/3

Il existe  $i \in \{1,2\}$ , tel que  $\Delta_j^i \geq 0$  donc la fonction objective atteint son maximum.

$x = (0, 200/3, 100/3, 0)$  est la solution optimale  $Z_3 = (Z_3^1 = 30, Z_3^2 = 175)$

Finalement, en considérant le problème initial (1) seuls les points extrêmes  $x^2 = (50, 0)$

et  $x^4 = (0, 200/3)$  sont non dominés avec:

$$Z(x^2) = \begin{pmatrix} 30 \\ 150 \end{pmatrix} \text{ et } Z(x^4) = \begin{pmatrix} 30 \\ 175 \end{pmatrix}.$$

On a recensé deux solutions efficaces.

2. Une entreprise de textile fabrique deux variétés de tissus  $T_1$ ,  $T_2$  à partir de deux laines  $L_1$ ,  $L_2$ . Le tableau suivant récéncé les poids (en kg) des laines intervenant dans la composition d'un mètre des tissus.

	$T_1$	$T_2$
$L_1$	1	2
$L_2$	3	1

On dispose d'un stock de 200 kg de laine  $L_1$  et 300 kg de laine  $L_2$ .

On fabrique 50% de tissus  $T_1$  et 10% de tissus  $T_2$ .

A la vente d'un mètre de tissus  $T_1$  en reçoit 2euros de bénéfice.

On veut calculer le nombre de mètre de tissus à produire pour que la production soit maximale et aussi pour que le bénéfice reçu de la vente de tissus  $T_1$  soit maximal.

$$MaxZ_1 = 0.5x_1 + 0.1x_2$$

$$MaxZ_2 = 2x_1$$

$$\begin{cases} x_1 + 2x_2 \leq 200 \\ 3x_1 + x_2 \leq 300 \\ x_1 \geq 0, x_2 \geq 0 \end{cases} \quad (1).$$

$$(1) \Leftrightarrow MaxZ_1 = 0.5x_1 + 0.1x_2$$

$$MaxZ_2 = 2x_1$$

$$\begin{cases} x_1 + 2x_2 + x_3 = 200 \\ 3x_1 + x_2 + x_4 = 300 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0 \end{cases}$$

$x_1 = (0,0,200,300)'$  est une solution de base réalisable associée à la base  $(a_3, a_4) = I_2$  et

$$Z(x) = \begin{pmatrix} Z_1(x) \\ Z_2(x) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Le tableau du simplexe initial est:

			$C^1$	0.5	0.1	0	0
			$C^2$	2	0	0	0
$C_B^1$	$C_B^2$	<i>Base</i>	<i>B</i>	$a_1$	$a_2$	$a_3$	$a_4$
0	0	$a_3$	200	1	2	1	0
0	0	$a_4$	300	3	1	0	1
$Z_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$			$\Delta^1$	-0.5	-0.1	0	0
			$\Delta^2$	-2	0	0	0

Il n'existe pas un  $i \in \{1,2\}$  tel que  $\Delta_j^i \geq 0; \forall j \in J_N = \{1,2\}$ , donc aucun objectif n'est à son maximum.

$$\Delta_1 = \begin{pmatrix} -0.5 \\ -2 \end{pmatrix} \leq 0, x_1 = (0,0,200,300)' \text{ est dominé.}$$

$$\theta_1 = \text{Min}\{200/1, 300/3\} = 100. \text{ (atteint pour } r=4 \text{).}$$

$$\theta_2 = \text{Min}\{200/2, 300/1\} = 100. \text{ (atteint pour } r=3 \text{).}$$

$$\theta_1 \Delta_1 = 100 \begin{pmatrix} -0.5 \\ -2 \end{pmatrix} = \begin{pmatrix} -50 \\ -200 \end{pmatrix}.$$

$$\theta_2 \Delta_2 = 100 \begin{pmatrix} -0.1 \\ 0 \end{pmatrix} = \begin{pmatrix} -10 \\ 0 \end{pmatrix}.$$

$$\theta_1 \Delta_1 \leq \theta_2 \Delta_2.$$

Introduisons  $a_1$  dans la base à la place de  $a_4$ , on aura alors le tableau suivant:

			$C^1$	0.5	0.1	0	0
			$C^2$	2	0	0	0
$C_B^1$	$C_B^2$	<i>Base</i>	<i>B</i>	$a_1$	$a_2$	$a_3$	$a_4$
0	0	$a_3$	100	0	5/3	1	-1/3 →
0.5	2	$a_1$	100	1	1/3	0	1/3
		$Z_0 = \begin{pmatrix} 50 \\ 200 \end{pmatrix}$	$\Delta^1$	0	1/15	0	1/6
			$\Delta^2$	0	2/3	0	2/3

$\exists i = 2$  tel que  $\Delta_j^i > 0$  tel que  $\forall j \in J_N$ , ce qui implique que le deuxième critère est à son maximum.

Donc, le point extrême  $x_2 = (100, 0, 100, 0)$  est non dominé.

$$\theta_2 = \text{Min}\{300/5, 300/1\} = 60 \text{ (atteint pour } r=3).$$

$$\theta_4 = \text{Min}\{300/1\} = 300 \text{ (atteint pour } r=1).$$



$$\theta_2 \Delta_2 = 60 \begin{pmatrix} 1/15 \\ 2/3 \end{pmatrix} = \begin{pmatrix} 4 \\ 40 \end{pmatrix}.$$

$$\theta_4 \Delta_4 = 300 \begin{pmatrix} 1/6 \\ 2/3 \end{pmatrix} = \begin{pmatrix} 50 \\ 100 \end{pmatrix}.$$

$$\theta_2 \Delta_2 \leq \theta_4 \Delta_4.$$

Donc  $a_2$  rentre dans la base et  $a_3$  sort de la base.

Dressons alors le troisième tableau simplexe:

			$C^1$	0.5	0.1	0	0
			$C^2$	2	0	0	0
$C_B^1$	$C_B^2$	Base	B	$a_1$	$a_2$	$a_3$	$a_4$
0.1	0	$a_2$	60	0	1	3/5	-1/5
0.5	1	$a_1$	80	1	0	-1/5	6/15 
		$Z_0 = \begin{pmatrix} 46 \\ 160 \end{pmatrix}$	$\Delta^1$	0	0	1/50	78/75
			$\Delta^2$	0	0	-2/5	4/5 

$\Delta^1 \geq 0$ , alors le premier objectifs est à son maximum et puisqu'il n'existe pas de colonne  $j$  telle que  $\Delta_j = 0$ .

avec  $j \in J_N$ , alors  $x_3 = (80, 60, 0, 0)$  est non dominé.

$$\theta_3 = \text{Min} \left\{ \frac{60}{3/5} \right\} = 100 \text{ (atteint pour } r=2 \text{)}.$$

$$\theta_4 = \text{Min} \left\{ \frac{80}{6/15} \right\} = 200 \text{ (atteint pour } r=1 \text{)}.$$

$$\theta_3 \Delta_3 = 100 \begin{pmatrix} 1/50 \\ -2/5 \end{pmatrix} = \begin{pmatrix} 2 \\ -40 \end{pmatrix}.$$

$$\theta_4 \Delta_4 = 200 \begin{pmatrix} 78/75 \\ 4/5 \end{pmatrix} = \begin{pmatrix} 208 \\ 250 \end{pmatrix}.$$

$$\theta_3 \Delta_3 \leq \theta_4 \Delta_4.$$

Puisque l'introduction de  $a_3$  dans la base conduit à une base exploré, donc on introduit  $a_4$ .

			$C^1$	0.5	0.1	0	0
			$C^2$	2	0	0	0
$C_B^1$	$C_B^2$	<i>Base</i>	<i>B</i>	$a_1$	$a_2$	$a_3$	$a_4$
0.1	0	$a_2$	100	1/2	1	1/2	0
0	0	$a_4$	200	15/6	0	-1/2	1
			$\Delta^1$	-39/15	0	81/150	0
			$\Delta^2$	-2	0	0	0

Il n'existe pas un  $i \in \{1,2\}$ , tel que  $\Delta_j^i \geq 0 \quad \forall j \in \{1,3\}$  donc aucun objectif n'est à son maximum.

$$\Delta_1 = \begin{pmatrix} -39/15 \\ -2 \end{pmatrix} < 0 \Rightarrow x_4 = (0,100,0,200)' \text{ est dominé.}$$

En considérant le problème initial (1) seuls les points extrêmes  $x^2 = (100,0)'$

Et  $x_*^3 = (80,60)'$  sont non dominés avec:

$$Z(x_*^2) = \begin{pmatrix} 50 \\ 200 \end{pmatrix} \text{ et } Z(x_*^3) = \begin{pmatrix} 46 \\ 160 \end{pmatrix}.$$

On a recensé deux solutions efficaces.

## 4.2 Deuxième cas : Problème de mélange alimentaire [6]:

La formulation d'une diète à moindre coût est aujourd'hui un exemple classique de problème de modélisation mathématique. Il apparait dans différents livres de recherche opérationnelle. Sous sa forme classique, le modèle qui permet de trouver un mélange alimentaire à moindre coût qui satisfait des besoins nutritionnels peut être un problème de programmation linéaire et peut être, aujourd'hui, solutionné facilement.

### 4.2.1 Formulation du problème:

Supposons que l'on ait  $n$  ingrédients. La variable de décision  $x_j$  représente la proportion de l'ingrédient  $j$  dans le mélange et  $x = (x_j)$  tel que  $j = \overline{1,n}$  est le vecteur de décision du modèle. Le coût unitaire de l'ingrédient  $j$  est représenté par  $c_j$  et  $c = (c_j)$  tel

que  $j = \overline{1, n}$  est le vecteur coût. Ainsi, le prix unitaire de mélange est donné par  $Z = c'x$  qui doit être minimisé. Le modèle classique de formulation à moindre coût apparait sous la forme suivante:

$$\left\{ \begin{array}{l} \min Z = c'x \\ \text{Sujet à} \\ x \in D = \{ x = (x_j) \text{ tel que } j = \overline{1, n} \in \mathbb{R}^k \mid Ax \{>, =, <\} b \text{ et } x \geq 0 \} \end{array} \right.$$

ou  $A$  est la matrice des apports  $a_{ij}$  des ingrédients et  $b$  est le vecteur des besoins. Notons que dans ce modèle, les coûts de fabrication, transport et entreposage ne sont pas considérés.

Ce modèle a également été formulé pour minimiser plusieurs fonctions objectives. Nous pouvons par exemple vouloir minimiser à la fois le coût du mélange, mais également le rejet de certains nutriments. Dans ce cas on parle de modèle **multicritères**. Ce modèle a été présenté par Steuer en 1986. Il s'écrit sous sa forme générale de la manière suivante:

$$\left\{ \begin{array}{l} \min Z_1 = c_1'x_1 \\ \vdots \\ \min Z_k = c_n'x_n \\ \text{Sujet à} \\ x \in D \end{array} \right.$$

Une telle modélisation permis d'optimiser plusieurs fonctions objectives en même temps. Ici par exemple  $Z_1, \dots, Z_k$ .

**4.2.2 Exemple :**

Une diététicienne doit réaliser, pour un patient, un plat qui se compose de 2 aliments (A, B) valant respectivement 130DA/kg et 60DA/kg. De telle sorte que le coût du plat soit minimal. La quantité de lipides ne doit pas dépasser 40g, la quantité de glucides ne doit pas dépasser 30g et la quantité de protéine doit être d’au moins 400g.

Le tableau suivant donne les quantités de lipides, glucides et protéines par gramme, pour chacun des deux aliments.

	Lipides(g)	Glucides(g)	Protéines(g)
<b>Aliment A</b>	0.05	0.06	0.9
<b>Aliment B</b>	0.2	0.5	0.25

Soit  $x_1$  la quantité en grammes de l’Aliment A et  $x_2$  la quantité en grammes de l’aliment B. Formuler le programme linéaire (MO) qui permettra à la diététicienne de déterminer la quantité de chaque aliment afin de minimiser les coûts et aussi pour que la quantité de l’aliment B dans le mélange soit minimale on respectant les contraintes.

**Le programme linéaire (MO) associé :**

$$\left\{ \begin{array}{l}
 \min Z_1 = 130x_1 + 60x_2 \\
 \min Z_2 = x_2 \\
 \text{s.c} \\
 0.5x_1 + 0.2x_2 \leq 40, \\
 0.6x_1 + 0.5x_2 \leq 30, \\
 0.6x_1 + 0.25x_2 \geq 400, \\
 x_1, x_2 \geq 0
 \end{array} \right.$$

$x_1$  : La quantité de l'aliment A dans le mélange.

$x_2$  : La quantité de l'aliment B dans le mélange.

Ce problème peut être résolu par le simplexe multicritères.

### 4.3. Troisième cas: Problème de transport multicritères(MOTP) [3]:

Le problème de transport multicritère est une extension du problème d'affectation multicritère. Supposons qu'on transporte un produit à partir de  $r$  usines (dépôts) disposant de quantités  $q_1, q_2, \dots, q_r$  vers  $s$  destinations (ateliers, magasins, . . .) ou des demandes  $d_1, d_2, \dots, d_s$  doivent être satisfaites.

On associe des valeurs  $c_{ij}^k$  le coût de transport, délai de livraison, sécurité de livraison, quantité fournie au transport d'une unité de produit de l'origine  $i$  à la destination  $j$ . Il s'agit de déterminer comment approvisionner les magasins à partir des dépôts de manière à minimiser le coût total et le délai de livraison et d'autres objectifs de transport.

#### 4.3.1 Formulation du problème:

Soit  $x_{ij}$  la quantité de biens à transporter de  $i$  vers  $j$  : le problème de transport multicritères peut se formuler comme suit:

$$\begin{array}{l}
 \text{(MOTP)} \quad \left\{ \begin{array}{l}
 \min \sum_{i \in \{1, \dots, r\}} \sum_{j \in \{1, \dots, s\}} c_{ij}^k x_{ij}, \quad k = 1, \dots, p \\
 \text{s.c} \quad \sum_{j \in \{1, \dots, s\}} x_{ij} = q_i, \quad i = 1, \dots, r, \\
 \\
 \sum_{i \in \{1, \dots, r\}} x_{ij} = d_j, \quad j = 1, \dots, s, \\
 \\
 x_{ij} \geq 0, \quad \forall i, j.
 \end{array} \right.
 \end{array}$$

#### 4.3.2 Exemple:

Une firme automobile a trois usines à Los Angeles, New Orléans et Detroit, et trois centres de distribution à Denver, Miami et Orlando.

- Les capacités des 3 usines sont de 20000, 30000 et 35000 respectivement, et les demandes aux centres de distribution sont de 20000,45000 et 20000 voitures.
- Coûts de transport par avion ( $\times 1000\text{euros}$ ):

	Denver	Miami	Orlando
Los Angles	3	2	4
New Orléans	1	4	3
Detroit	4	2	5

- Coûts de transport par chemin de fer ( $\times 1000\text{euros}$ ):

	Denver	Miami	Orlando
Los Angles	0	1	1
New Orléans	0	1	3
Detroit	4	2	5

**Formulation de problème:**

$$\left\{ \begin{array}{l} \text{Min}Z_1 = 3x_{11} + 2x_{12} + 4x_{13} + x_{21} + 4x_{22} + 3x_{23} + 4x_{31} + 2x_{32} + 5x_{33} \\ \text{Min}Z_2 = 0x_{11} + x_{12} + x_{13} + 0x_{21} + x_{22} + 3x_{23} + 4x_{31} + 2x_{32} + 5x_{33} \\ \text{s.c} \\ x_{11} + x_{12} + x_{13} = 20 \\ x_{21} + x_{22} + x_{23} = 30 \\ x_{31} + x_{32} + x_{33} = 35 \\ x_{11} + x_{21} + x_{31} = 20 \\ x_{12} + x_{22} + x_{32} = 45 \\ x_{13} + x_{23} + x_{33} = 20 \\ x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23} \geq 0. \end{array} \right.$$

- Tableau de transport pour le premier critère  $Z_1$  :

	Denver	Miami	Orlando	$q_i$
Los Angles	3	2	4	20
New Orléans	1	4	3	30
Detroit	4	2	5	35
$d_j$	20	45	20	85

- Tableau de transport pour le deuxième critère  $Z_2$  :

	Denver	Miami	Orlando	$a_i$
Los Angles	0	1	1	20
New Orléans	0	1	3	30
Detroit	4	2	5	35
$b_j$	20	45	20	85

On appliquant le simplexe pour  $Z_1$  on trouve:

	Denver	Miami	Orlando	$a_i$
Los Angles	0	10	10	20
New Orléans	20	0	10	30
Detroit	0	35	0	35
$b_j$	20	45	20	85

$$x_1 = (0,10,10,20,0,10,0,35,0)'$$

Donc:

$$Z_1 = 180 \quad \text{et} \quad Z_2 = 120$$

$$Z_1^* = \begin{pmatrix} Z_1(x_3) \\ Z_2(x_3) \end{pmatrix} = \begin{pmatrix} 180 \\ 120 \end{pmatrix}.$$

$Z_1^*$  est efficace car le vecteur des estimations positif strict sur  $(J_H / J_N)$  à l'optimum.

Appliquons maintenant le simplexe pour  $Z_2$  :

	Denver	Miami	Orlando	$a_i$
Los Angles	0	0	20	20
New Orléans	20	10	10	30
Detroit	0	35	0	35
$b_j$	20	45	20	85

$$x_2 = (0,0,20,20,10,10,0,35,0)'$$

Donc:

$$Z_1 = 210 \quad \text{et} \quad Z_2 = 100$$

$$Z_2^* = \begin{pmatrix} Z_1(x_4) \\ Z_2(x_4) \end{pmatrix} = \begin{pmatrix} 210 \\ 100 \end{pmatrix}.$$

$$Z_1^* = \begin{pmatrix} 180 \\ 120 \end{pmatrix} \quad \text{et} \quad Z_2^* = \begin{pmatrix} 210 \\ 100 \end{pmatrix} \quad \text{sont incomparable donc ils sont efficaces.}$$

On applique la méthode de l'élément minimal pour  $Z_1$  on trouve:

	Denver	Miami	Orlando	$a_i$
Los Angles	0	20	0	20
New Orléans	20	0	10	30
Detroit	0	25	10	35
$b_j$	20	45	20	85

$$x_3 = (0,20,0,20,0,10,0,25,10)'$$

Donc:

$$Z_1 = 190 \quad \text{et} \quad Z_2 = 180$$

$$Z_3^* = \begin{pmatrix} Z_1(x_1) \\ Z_2(x_1) \end{pmatrix} = \begin{pmatrix} 190 \\ 180 \end{pmatrix}$$

On applique la même méthode pour  $Z_2$  on trouve:

	Denver	Miami	Orlando	$a_i$
Los Angles	20	0	0	20
New Orléans	0	30	0	30
Detroit	0	15	20	35
$b_j$	20	45	20	85

$$x_4 = (20, 0, 0, 0, 30, 0, 0, 15, 20)$$

Donc:

$$Z_2 = 160 \quad \text{et} \quad Z_1 = 310$$

$$Z_4^* = \begin{pmatrix} Z_1(x_2) \\ Z_2(x_2) \end{pmatrix} = \begin{pmatrix} 310 \\ 160 \end{pmatrix}$$

On remarque que  $Z_3^* = \begin{pmatrix} 190 \\ 180 \end{pmatrix}$  et  $Z_4^* = \begin{pmatrix} 310 \\ 160 \end{pmatrix}$  sont incomparables.

- On appliquant la méthode coin nord-ouest on trouve une solution identique à  $Z_4^*$

qu'il est donc dominé.

On aura alors:

$$Z_1^* = \begin{pmatrix} 180 \\ 120 \end{pmatrix} \quad \text{et} \quad Z_2^* = \begin{pmatrix} 210 \\ 100 \end{pmatrix} \quad \text{sont incomparables donc ils sont efficaces.}$$

$$Z_1^* = \begin{pmatrix} 180 \\ 120 \end{pmatrix} \quad \text{et} \quad Z_3^* = \begin{pmatrix} 190 \\ 180 \end{pmatrix} \quad \text{sont comparable donc } Z_3^* \text{ est dominé.}$$

$$Z_1^* = \begin{pmatrix} 180 \\ 120 \end{pmatrix} \quad \text{et} \quad Z_4^* = \begin{pmatrix} 310 \\ 160 \end{pmatrix} \quad \text{sont comparable, } Z_4^* \text{ est dominé.}$$

**Remarque:**

Pour générer toutes les solutions efficaces du problème il suffit de générer toutes les solutions réalisables de bases du problème.

**Conclusion:**

Dans ce chapitre, nous avons présenté quelques cas réels d'optimisations multi-objectifs existants et leurs résolution avec différentes méthodes.

# **CHAPITRE 5:**

## **Partie informatique (logiciel LINGO)**

## 5.1 LINGO: Logiciel pour la résolution des programmes linéaires et non linéaires.

### 5.1.1 Introduction:

Lingo est un logiciel utilisé pour résoudre les modèles d'optimisation linéaire, entier et quadratique, il est aussi utilisé pour résoudre les modèles d'optimisation globale non linéaire. Une des caractéristiques de lingo c'est qu'il offre des outils qui peuvent aider à l'analyse des modèles en utilisant la méthode du simplexe.



### 5.1.2 Installation de logiciel:

Pour utiliser cette version de lingo il est conseillé d'avoir au moins un processeur 486 et 8 Mo de mémoire RAM. Il faut aussi prévoir un espace disque dur de 2 Mo pour pouvoir l'installer.

Les étapes de l'installation sont:

1. Démarrer Windows.
2. Insérer CD-ROM.
3. Cliquer sur l'icône setup (Install) dans votre explorateur de Windows.
4. Suivre les instructions de l'écran.

Pour plus d'information sur ce logiciel visiter l'adresse web [www.lingo.com](http://www.lingo.com).

### 5.1.3 Présentation de logiciel « LINGO »:

Nous permettons de formuler rapidement et facilement nos problèmes d'optimisation linéaire, non linéaire ou en nombres entiers. Grâce à ses outils de modélisation, les modèles sont exprimés de manière transparente à l'aide de sommes et de variables indicées.

La méthode ne diffère guère de la méthode traditionnelle avec crayon et papier par exemple, mais les modèles seront plus faciles à réutiliser et mettre à jour. Il possède quatre solveurs afin de résoudre les différents types de modèles:

- Solveur direct.
- Solveur indirect.
- Solveur linéaire.
- Solveur non linéaire.
- Méthode de type séparation et évaluation.

De plus, LINGO est:

- Un moyen pour confirmer que l'optimum trouvé est l'optimum global.
- Possibilités pour résoudre les problèmes plus rapidement.
- Un moyen amélioré pour résoudre beaucoup de types de problèmes.
- Un moyen de décomposition si un modèle contient des sous-modèles.

LINGO est livré avec un jeu de solveurs pour l'optimisation linéaire, non-linéaire convexe ou non convexe), simplifiable, sous contraintes, et en nombre entier.

Nous n'avons même pas à nous préoccuper du choix du solveur : en effet, LINGO réinterprète lui-même nos formulations et sélectionne automatiquement le solveur adapté à chaque problème. Un modèle d'optimisation se compose de trois parties:

1. La fonction objective : c'est la formule simple qui décrit exactement ce que le modèle devrait optimiser.
2. Les variables : ce sont les quantités qui peuvent être changées pour déterminer la valeur optimale de la fonction objective.
3. Les contraintes : ce sont les formules qui définissent les limites sur les valeurs des variables. Les fonctions utilisées dans un modèle de LINGO sont:

@FOR - utilisée pour produire des contraintes.

@ SUM- calcul de la somme.

@MAX - recherche de maximum.

@MIN - recherche de minimum.

Type de variables dans LINGO : Toutes les variables dans un modèle de LINGO sont considérées non négatives et continus, les fonctions variables d'un modèle de LINGO sont :

GIN- toute valeur positive de nombre entiers.

BIN - une valeur binaire (0 ou 1).

FREE - toute valeur positive ou négative réelle.

BND - toute valeur bornée par des limites indiquées.

La forme générale pour la déclaration d'une variable  $x$ , en utilisant les fonctions variables GIN, BIN, FREE est : fonction ( $x$ ) ; La fonction BND inclut les limites inférieures et supérieures des variables, sa forme générale est:

BND (limite inférieure,  $x$ , limite supérieure).

### 5.1.4 Interface de logiciel:

Juste après la barre des menus LINGO, une fenêtre pour saisir le modèle de problème à résoudre apparaît:

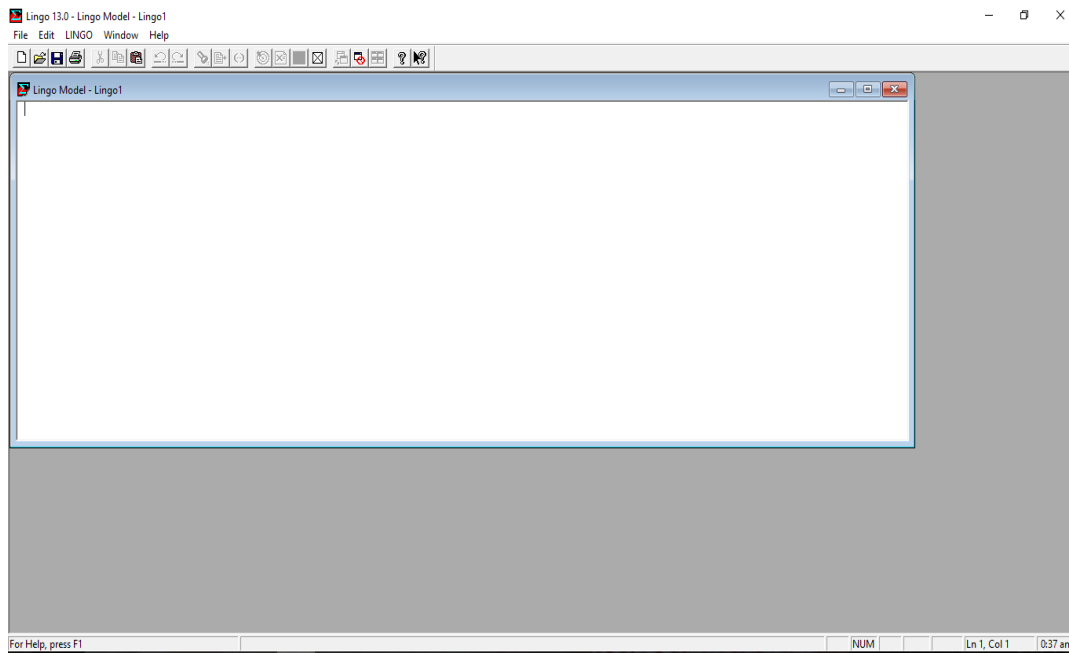


Figure 14: la barre des menus de logiciel

La barre des outils de LINGO est la suivante:

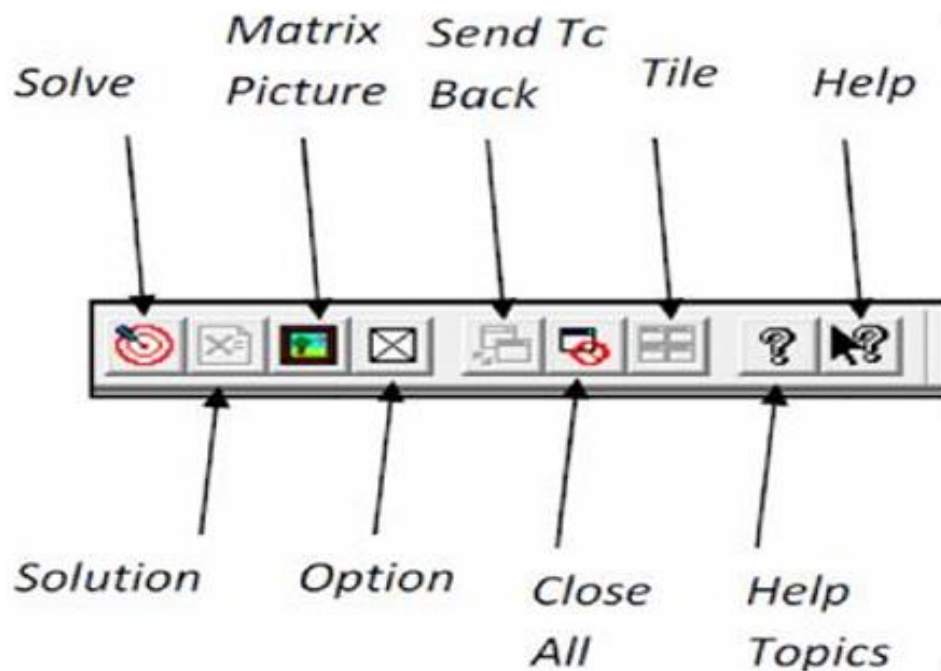
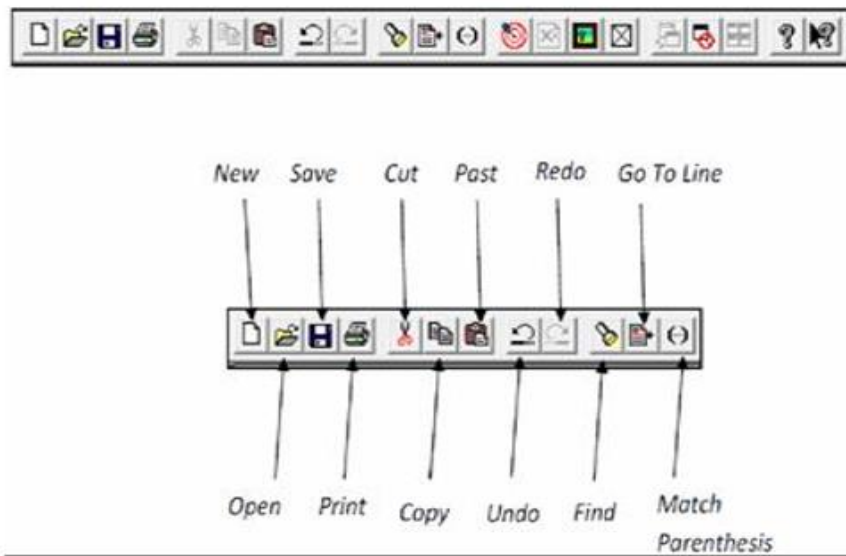


Figure 15: la barre des outils de LINGO

**Signification:**

**New** : ouvrir une nouvelle fenêtre.

**Open** : ouvrir un dossier enregistré.

**Save** : enregistrer un modèle.

**Print** : imprimer la fenêtre courante.

**Cut** : copier et déplacer le texte sélectionné.

**Copy** : copier le texte sélectionné.

**Past** : coller le contenu sélectionné dans le document.

**Undo** : refaire l'action précédente.

**Redo** : revenir à l'action suivante.

**Find** : chercher un document.

**Go To Line** : déplacer le curseur à un certain nombre de ligne.

**Match Parenthesis** : trouvez la parenthèse étroite qui correspond à la parenthèse ouverte que vous avez choisie.

**Solve** : résoudre le modèle.

**Solution** : faire la fenêtre de rapport de la solution du modèle courant.

**Option** : ensemble des options de système.

**Send To Back** : placer la fenêtre courante derrière les autres fenêtres ouvertes.

**Close All** : fermer toutes les fenêtres ouvertes.

**Tile** : placer les fenêtres ouvertes dans la même place dans la fenêtre de programme LINGO.

**Help Topics** : ouvrir le manuel de LINGO.

**Help** : Ouvrir l'aide de LINGO.

5.1.5 Les étapes de programme:

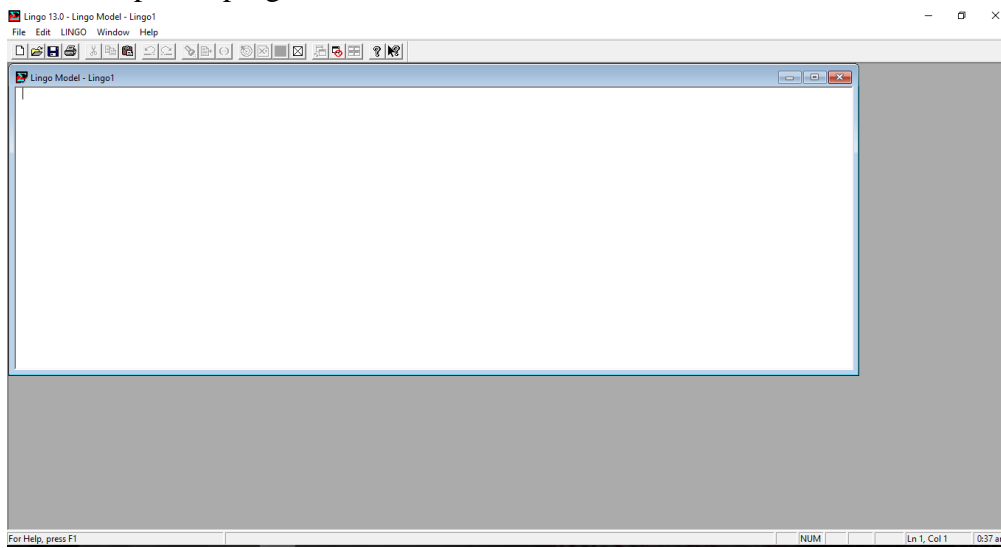


Figure 16:page d'accueil de logiciel

Nous cliquons sur l'icône **FILE** puis **NEW** (figure 17) et l'écran de l'espace de travail s'ouvre:

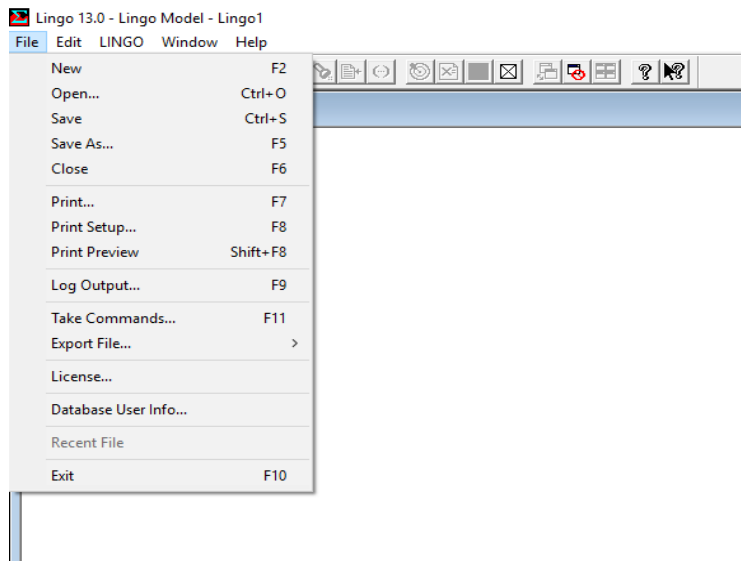


Figure 17:ouverture d'une nouveau espace de travail

Après avoir saisi notre modèle linéaire nous cliquons sur **SOLVE** (figure 5)

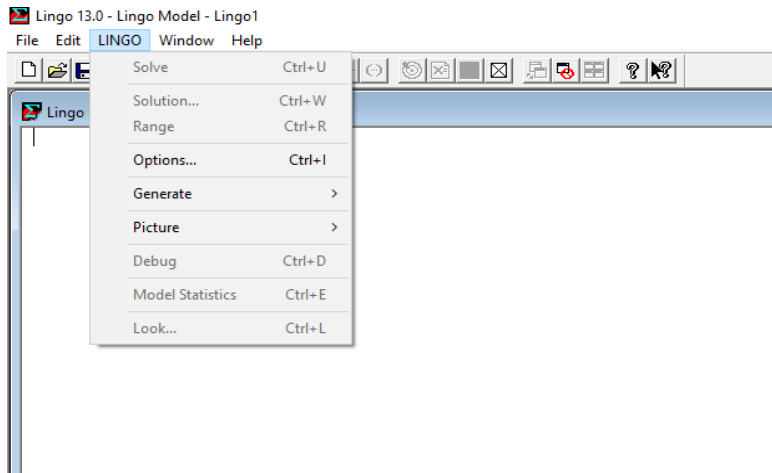


Figure 18 :l'exécution de modèle linéaire

**5.1.6 Exemples d'application:**

**Exemple1:**

Soit le problème bi-objectif de la programmation linéaire suivant:

$$(P) \left\{ \begin{array}{l} \max(Z_1) = 4x_1 + 6x_2 \\ \max(Z_2) = 3x_3 \\ 2x_1 + 3x_2 \geq 12 \\ 5x_1 + 2x_2 \leq 15 \\ x_1 + x_3 = 10 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0 \end{array} \right.$$

La transformation de problème bi-objectifs (P) en un problème mono-objectifs par la méthode d'agrégation linéaire donne le problème ( $P_\lambda$ ):

On fixe  $\lambda_1 =0.8$  et  $\lambda_2 =0.2$  (par priorité des objectifs)

On obtient:

$$P_\lambda \left\{ \begin{array}{l} \max(Z) = 0.8(4x_1 + 6x_2) + 0.2(3x_3) \\ 2x_1 + 3x_2 - x_4 + x_6 = 12 \\ 5x_1 + 2x_2 + x_5 = 15 \\ x_1 + x_3 = 10 \\ x_j \geq 0, j = \overline{1,6} \end{array} \right.$$

### 1) Saisir notre modèle linéaire ( $P_\lambda$ )

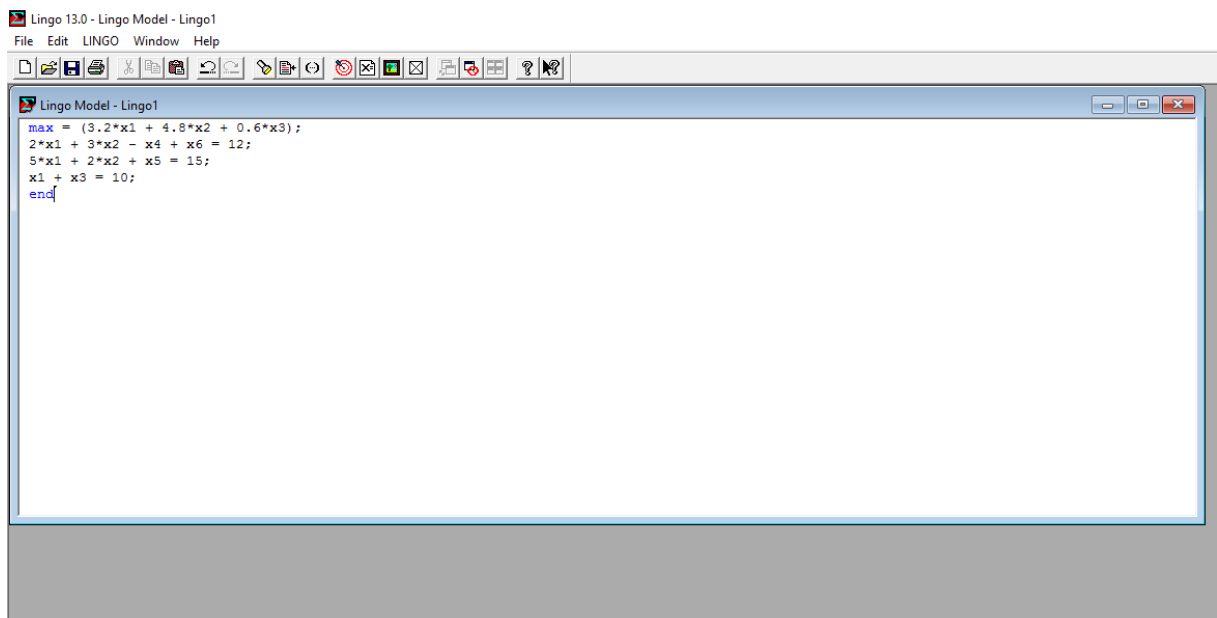


Figure 19 : la saisie de notre modèle linéaire

2) On clique sur « solve » dans la barre d'outils:

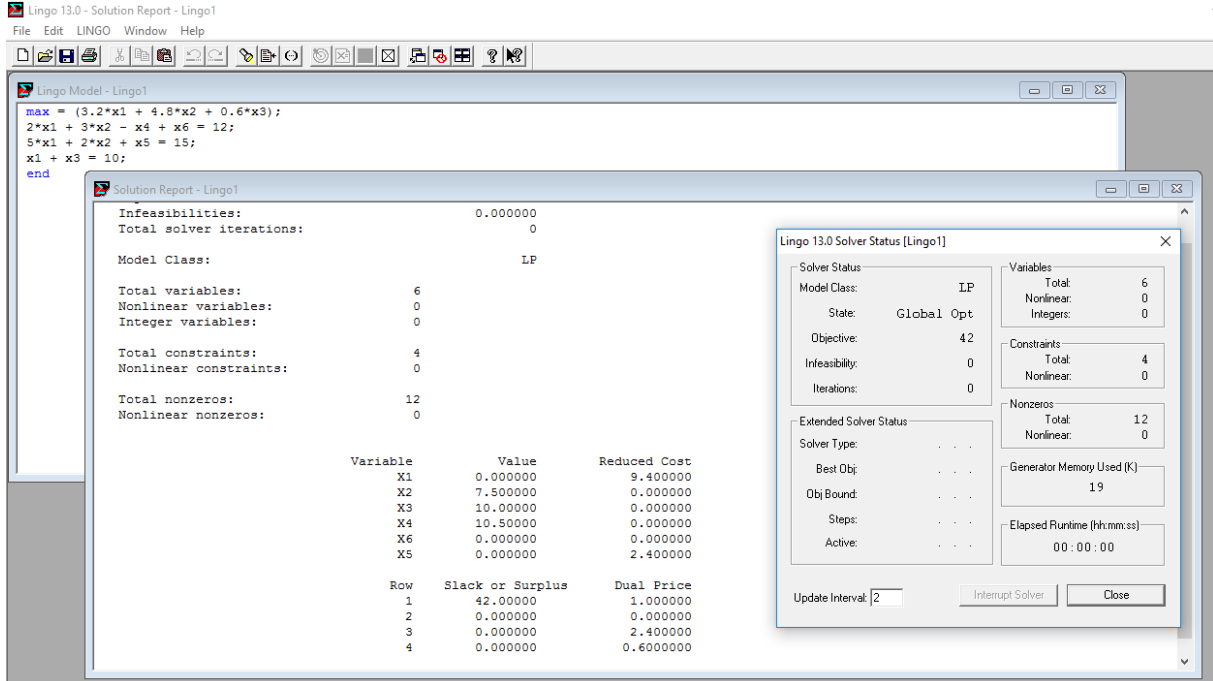


Figure 20 : l'écran d'affichage en cliquant sur « Solve »

**Exemple2:**

Prenons l'exemple traité dans le chapitre 3 par la méthode de  $\epsilon$ -contraintes:

$$\text{MOP} \left\{ \begin{array}{l} \min f(x) = (f_1(x) = -3x_1 + x_2; f_2(x) = x_1 + 2x_2; f_3(x) = 2x_1 + x_2) \\ 3x_1 - x_2 \leq 6 \\ x_2 \leq 2 \\ x = (x_1, x_2) \in R_+^2 \end{array} \right.$$

$$(P_\epsilon) \left\{ \begin{array}{l} \min f_{i=3}(x) \\ x \in D \\ f_j(x) \leq \epsilon_j, \forall j \neq 3, j = \overline{1,3} \end{array} \right.$$

$$(P_\varepsilon) \left\{ \begin{array}{l} \min 2x_1 + x_2 \\ 3x_1 - x_2 \leq 6 \\ x_2 \leq 2 \\ f_1(x) \leq \varepsilon_1 \rightarrow -3x_1 + x_2 \leq 1 \\ f_2(x) \leq \varepsilon_2 \rightarrow x_1 + 2x_2 \leq 3 \\ x = (x_1, x_2) \in R_+^2 \end{array} \right.$$

1. Saisir notre modèle après l'introductions des variables d'écarts:

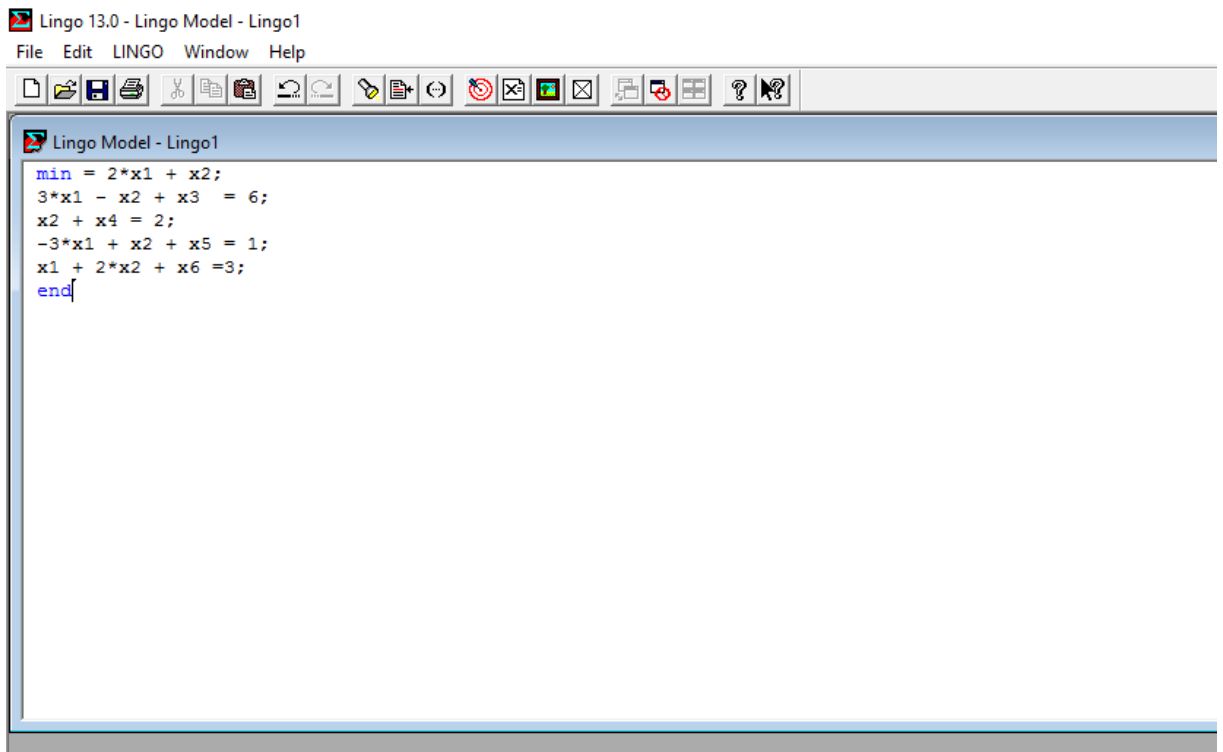


Figure 21 :la saisie du modèle avec des variables d'écarts

2. On clique sur « Solve » dans la barre d’outils :

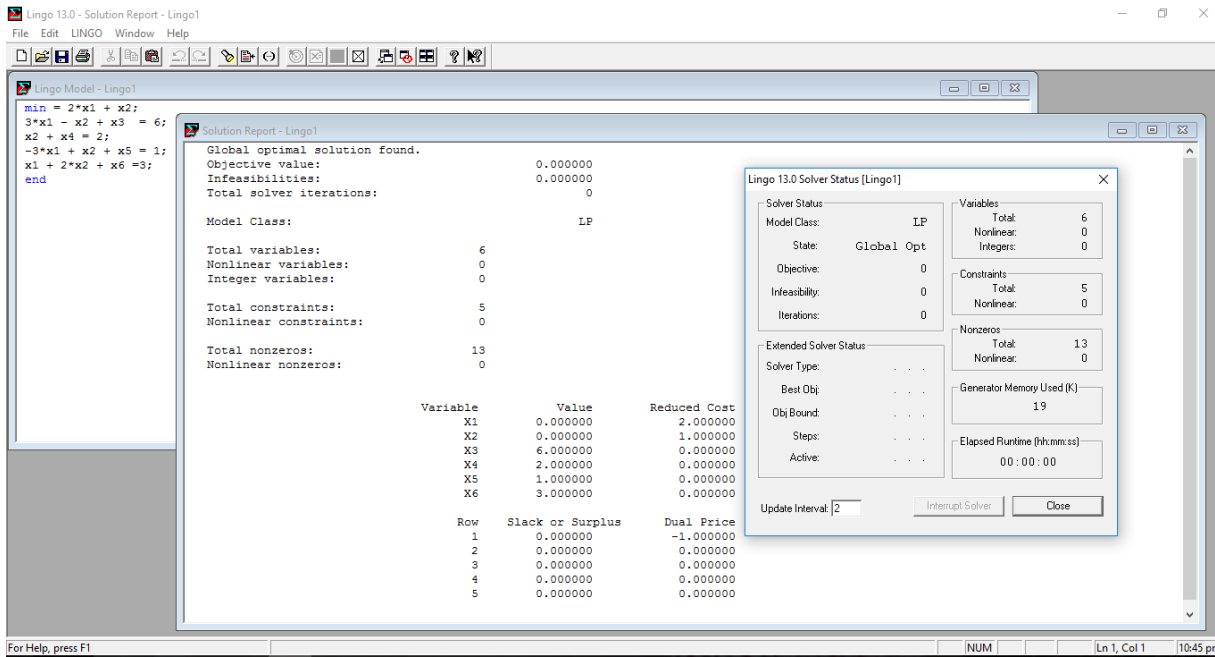


Figure 22 : l’exécution de modèle avec « Solve »

Exemple 3 : (exemple traité dans le chapitre 4) :

$$\begin{cases}
 \text{Min}Z_1 = 3x_{11} + 2x_{12} + 4x_{13} + x_{21} + 4x_{22} + 3x_{23} + 4x_{31} + 2x_{32} + 5x_{33} \\
 \text{Min}Z_2 = 0x_{11} + x_{12} + x_{13} + 0x_{21} + x_{22} + 3x_{23} + 4x_{31} + 2x_{32} + 5x_{33} \\
 \\
 \text{s.c} \\
 \\
 x_{11} + x_{12} + x_{13} = 20 \\
 x_{21} + x_{22} + x_{23} = 30 \\
 x_{31} + x_{32} + x_{33} = 35 \\
 x_{11} + x_{21} + x_{31} = 20 \\
 x_{12} + x_{22} + x_{32} = 45 \\
 x_{13} + x_{23} + x_{33} = 20 \\
 \\
 x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23} \geq 0.
 \end{cases}$$

1. Saisir notre modèle:

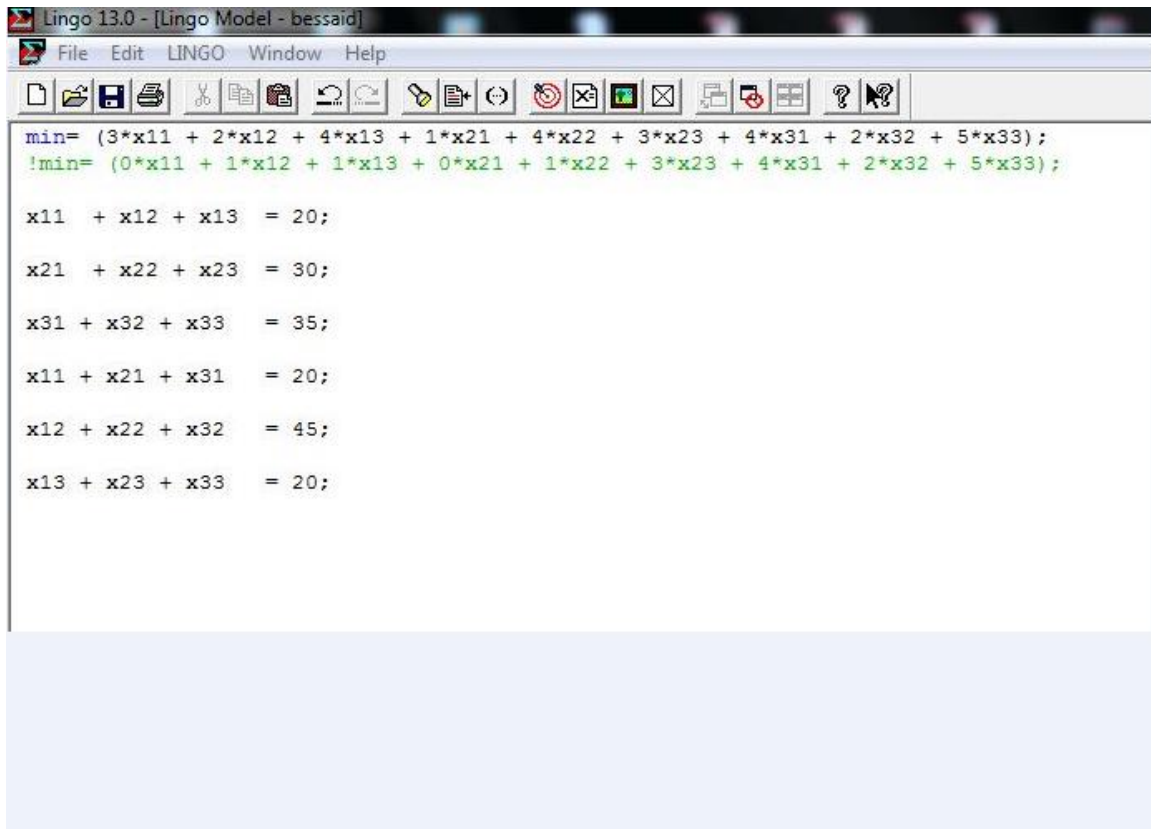


Figure 23: la saisie de modèle

2. On clique sur solve dans la barre d'outils:

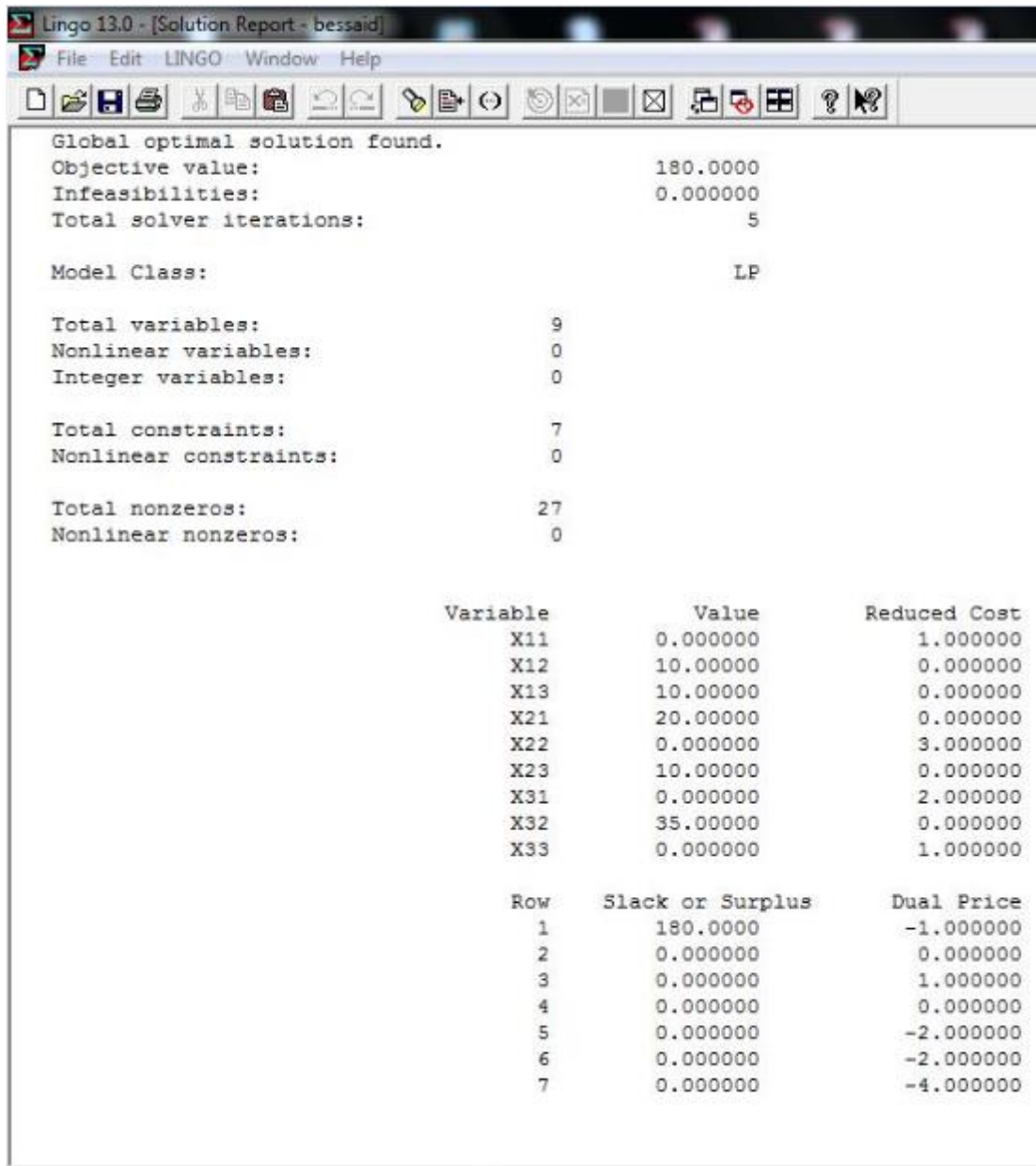


Figure 24: l'exécution de premier critère de modèle avec solve

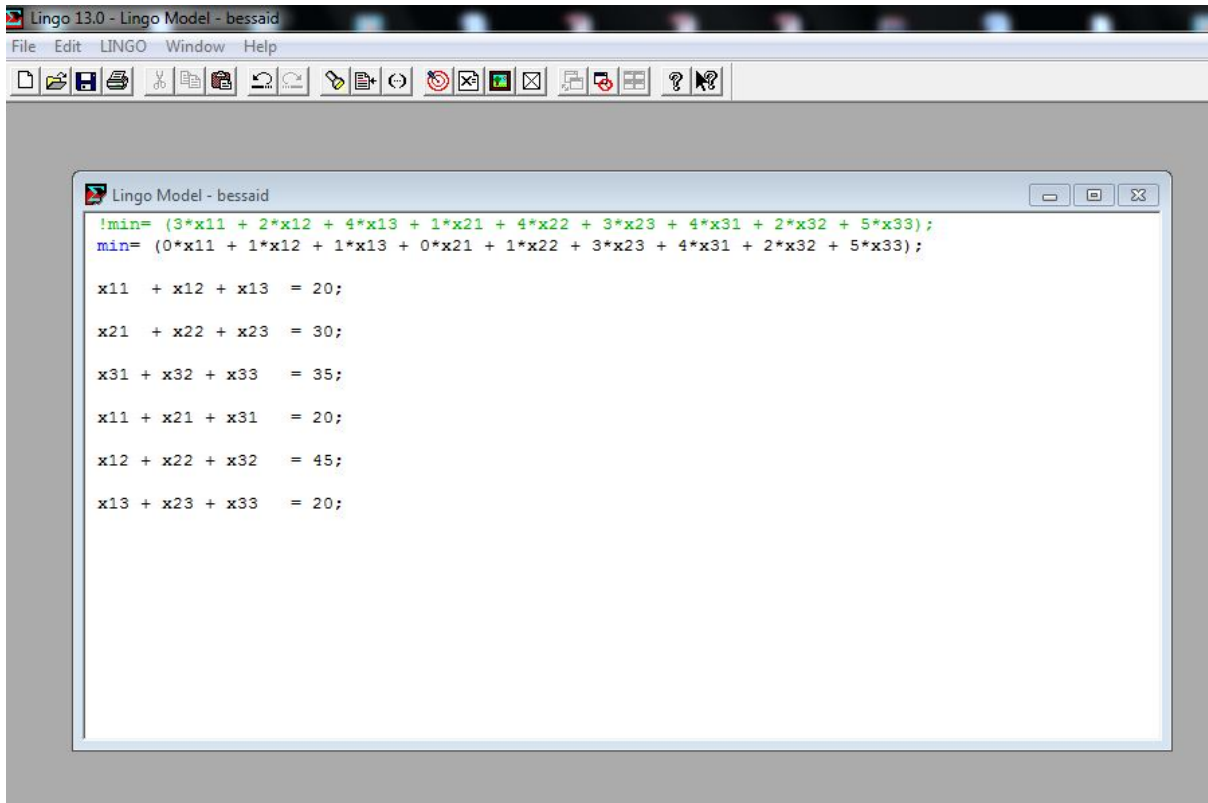


Figure 25: la saisie de modèle

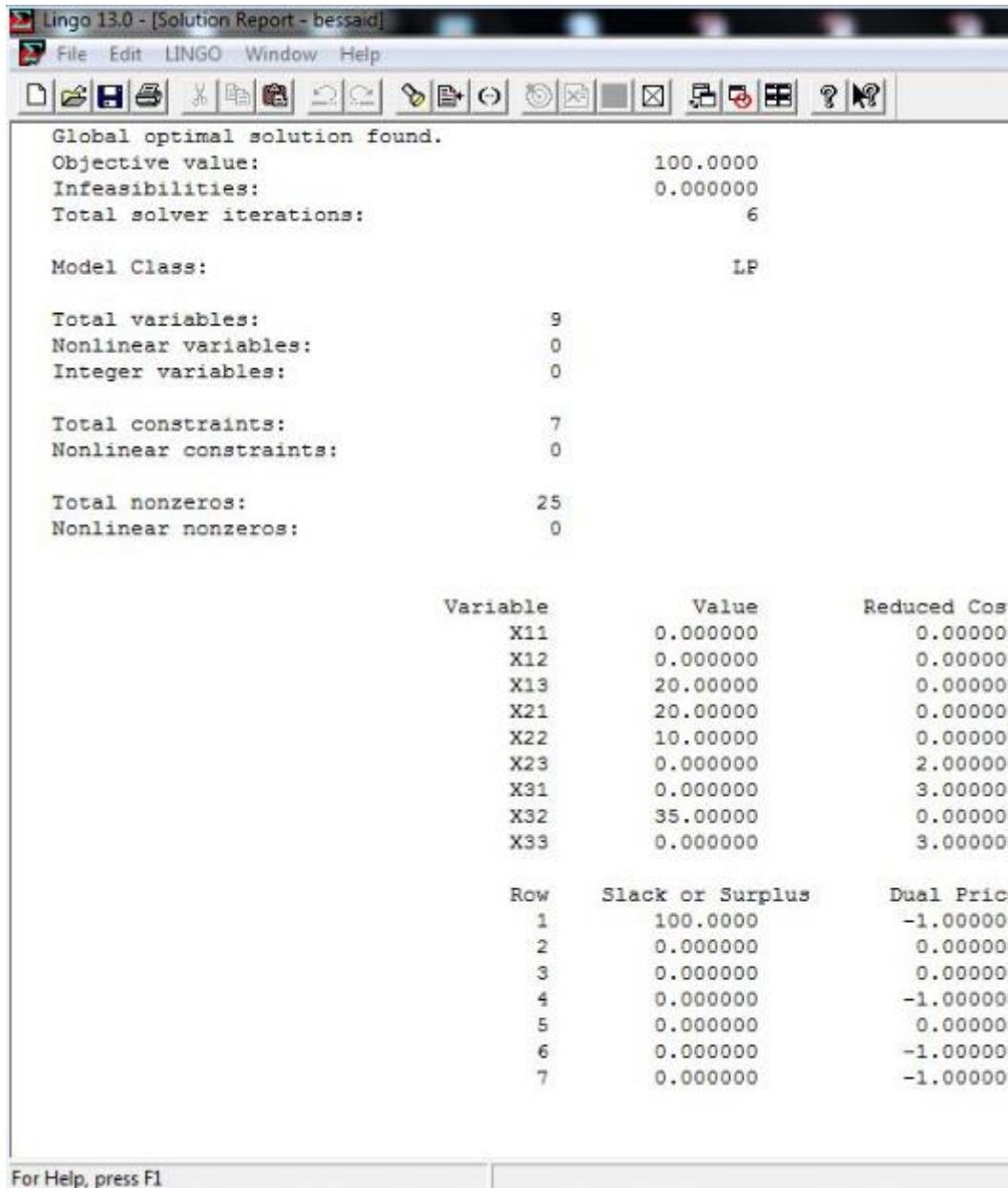


Figure 26: l'exécution de deuxième critère avec solve

# Conclusion générale

## Conclusion générale :

Les problèmes d'optimisation multi-objectifs sont de plus en plus étudiés car ils sont d'un grand intérêt pour l'industrie, où plusieurs objectifs concurrents doivent être satisfaits. Ils sont dans la plupart du temps NP-difficiles même si leur version mono-objectif ne l'est pas en générale. Contrairement aux problèmes d'optimisation mono-objectif, il existe des solutions Pareto optimales ou efficaces en optimisation multi-objectifs, c'est-à-dire des solutions qu'on ne peut améliorer sur un objectif sans les détériorer sur un autre. Comme il n'existe aucune solution meilleure en tout point qu'une autre, un compromis différent doit être choisi. Le choix est donc subjectif, et il est indispensable de proposer l'ensemble des choix possibles afin de ne pas exclure une possibilité. L'optimisation multi-objectifs est donc avant tout un outil d'aide à la décision, et c'est une personne entre autres qui prendra la décision finale.

Nous avons présenté dans ce mémoire les travaux de recherche sur les méthodes d'optimisation des problèmes multi-objectifs linéaires.

Dans notre étude, nous nous sommes particulièrement intéressées aux problèmes de programmation multi-objectifs linéaires et nous avons traité quelques cas réels tel que le problème de transport, le problème de production, le problème de mélanges alimentaires où nous avons utilisé les méthodes de résolutions des programmes linéaires multi-objectifs.

Notre travail est certes achevé mais reste ouvert à toute amélioration future tant dans le domaine théorique qu'au domaine pratique.

## Bibliographie:

- [1] : Aiden M. Oukacha B. « les manuels de l'étudiant Recherche Opérationnelle Programmation linéaire ». Copyright Eurl Pages Bleues internationales, Maison d'édition pour l'enseignement et la formation,2005.
- [2] : C. Guéret C. Prins M. Sevaux, La Méthode de Balas, Programmation linéaire, N° éditeur : 6186,Février 2003.
- [3] : Daoud. M, (2016) Exploitation de l'information pour résoudre les problèmes d'optimisation combinatoire multicritères. USTHB, Alger, Algérie.
- [4] : Gomory, R. E. Outline of an algorithm for integer solutions to linear programs and an algorithm for the mixed integer problem. In 50 Years of Integer Programming 1958-2008. Springer, 2010, pp. 77–103.
- [5] : Garfinkel, R. S., and Nemhauser, G. L. Integer programming, vol. 4. Wiley New York, 1972.
- [6] : JOANNOPOULOS. E, (2012) Nouvelles approches de modélisation et d'optimisation de diète animale. Sherbrooke, Québec : Canada.
- [7] : Klein D. and Hannan E., An algorithm for multiple objective integer linear programming problem, European journal of operational research 9, pp. 152- 159, (1982).
- [8] : Madani Bezoui , Cours « Optimisation multiobjectifs,Lesson 1 : Concepts de bases ».
- [9] : Pareto A., Cours d'économie politique, vol. 1 et 2, F. Rouge, Lausanne, (1896).
- [10] : Rabia . F (Cours Programmation lineaire,2019) ,UMMTO.
- [11] : Sakarovitch, M. Optimisation combinatoire : Programmation discrète, vol. 2. Editions Hermann, 1984.
- [12] : Taous Makhtoub et Lynda Toubal. « Optimisation d'une fonction sur l'ensemble Pareto d'un problème multi-objectifs discret ».UMMTO.2019.
- [13] : Yann C., Siarry P., Optimisation multiobjectif, groupe Eyrolles, (2002).
- [14] : Pareto A., Cours d'économie politique, vol. 1 et 2, F. Rouge, Lausanne, (1896).