

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université Mouloud Mammeri de TIZI-OUZOU  
Faculté des Sciences  
Département de Mathématiques



## *Mémoire de fin d'étude*

En vue de l'obtention du diplôme de master en mathématiques appliquées  
Spécialité : Recherche Opérationnelle

### *Thème*

---

**Optimisation des activités bancaires, modèles, algorithmes et  
implémentations Cas : CNEP-Banque de TIZI-OUZOU**

---

*Réalisé par :*

AIT AMER Lydia  
BOUHASSENE Asma



*Devant le jury*

Encadreur : CHEBBAH Mohammed

Président : AICHE Farid

Examineur : GOUBI Moloud

*2023-2024*

# *Dédicaces*

*"Le succès n'est pas final, l'échec n'est pas fatal : c'est le courage de continuer qui compte." (Winston Churchill)*

*Je remercie **Dieu** de m'avoir donné la force de surmonter les obstacles, de mener à bien ce projet et de dédier ce travail :*

*À **mes chères parents** ma boussole et mon port d'attache , qui ont toujours cru en moi et m'ont donné les moyens de réussir.*

*À mon frère **Yanis** et mes sœurs **ouerdia** et **Lyna** , pour leur amitié et leur complicité.*

*À ma binôme **Asma**, pour sa collaboration précieuse et son enthousiasme.*

**Lydia AIT AMER**

# *Dédicaces*

*Tout d'abord je remercie **Dieu** le tout puissant, de m'avoir aidé dans les moments les plus difficiles, de m'avoir aidé à accomplir ce travail.*

*À **mes parents** qui m'ont toujours encouragée à atteindre mes rêves.*

*À mes frères et sœurs qui ont su me remonter le moral dans les moments difficiles.*

*À **Lydia**, merci pour ta collaboration précieuse.*

*À **Lamia** et **Lilia**, pour votre amitié sincère et les moments partagés.*

*À **Youcef** et **Koceila**, mes amis fidèles. Ce travail est un rayon de soleil que je partage avec vous.*

**Asma BOUHASSENE**

# *Remerciements*

*Nous exprimons notre gratitude à Dieu pour nous avoir guidées et données le courage nécessaire pour mener à bien notre travail. Nous remercions chaleureusement notre promoteur, Monsieur **Chebbah Mohammed**, pour son excellent encadrement, sa patience et sa disponibilité. Nous sommes également reconnaissantes envers les membres du jury pour leur écoute et leur patience, ainsi qu'envers les enseignants du département de mathématiques pour leur soutien tout au long de notre parcours. Enfin, nous remercions tous ceux qui nous ont encouragées et ont contribué, de près ou de loin à la réalisation de ce mémoire.*

# Table des matières

<b>1</b>	<b><i>Présentation de l'organisme d'étude cas CNEP</i></b>	<b>3</b>
	Introduction . . . . .	3
1.1	Présentation et historique de la CNEP Banque : [1] . . . . .	3
1.2	Organisation de la CNEP Banque TIZI OUZOU [2] . . . . .	5
1.2.1	Missions et valeurs de la banque CNEP [2] . . . . .	6
1.2.2	Présentation du Réseau de Tizi-Ouzou . . . . .	7
1.2.3	Le processus de traitement d'une demande de crédit . . . . .	7
1.2.4	Les produits de la CNEP [3] . . . . .	8
1.2.5	Différents risques liés à l'opération de crédit [2] . . . . .	9
<b>2</b>	<b><i>Présentation générale de l'optimisation</i></b>	<b>10</b>
	Introduction . . . . .	10
2.1	Programmation linéaire . . . . .	10
2.1.1	Définition de l'optimisation : . . . . .	10
2.1.2	Définition d'un programme linéaire : . . . . .	10
2.1.3	Modélisation [4] . . . . .	11
2.1.4	Forme d'un programme linéaire : . . . . .	11
2.1.5	Domaine de solutions réalisables et solution optimale [5] : . . . . .	12
2.1.6	Propriétés des programmes linéaires . . . . .	12
2.1.7	Méthodes de résolution d'un programme linéaire : . . . . .	13
2.1.8	Initialisation de l'algorithme du simplexe : . . . . .	17
2.2	Programmation non linéaire : . . . . .	28
2.2.1	Définition d'un Programme non linéaire [6] [7] . . . . .	28
2.2.2	Optimisation sans contraintes [6] [7] . . . . .	31
2.2.3	Optimisation avec contraintes [6] [7] . . . . .	32
2.2.4	Cas particulier de la programmation mathématique . . . . .	33
2.3	Programmation flou . . . . .	35
2.3.1	Définition d'un sous ensemble flou : . . . . .	35
2.3.2	Caractéristiques d'un sous-ensemble flou [8] [9] [10] : . . . . .	36
2.3.3	Opération sur les ensembles flous [11] [12] . . . . .	37
2.3.4	Nombre flou de type L-R [13] [9] : . . . . .	38
2.3.5	Intervalle flou de type L-R [14] . . . . .	39
2.3.6	Nombre flou de type triangulaire [15] [16] : . . . . .	40
2.3.7	Opérations sur les nombres flous de type triangulaire . . . . .	40
2.3.8	Opération sur les nombres flous de type trapézoïdal . . . . .	41
2.3.9	Programmation linéaire floue : . . . . .	42
2.4	Programmation stochastique : . . . . .	52
2.4.1	Notion d'espace mesurable [8] . . . . .	52
2.4.2	Variables aléatoires [15] [12] [17] . . . . .	52

2.4.3	Fonction de répartition [15] [17]	53
2.4.4	Densité de probabilité [15] [17]	53
2.4.5	Lois de probabilité [15] [17]	53
2.4.6	Les moments d'une variable aléatoire [17]	56
2.4.7	Définition d'un programme stochastique [9]	56
2.4.8	Les différentes approches	57
<b>3</b>	<b><i>Les modèles utilisés dans les techniques bancaires</i></b>	<b>61</b>
	Introduction	61
3.1	Le modèle d'arbre de décision	62
3.2	Le problème du Sac à Dos (Knapsack Problem)	62
3.2.1	Description	62
3.2.2	Formulation	63
3.2.3	Applications	63
3.2.4	Variantes	63
3.3	Modèle de Markovitz	64
3.4	Choix d'emprunts :	65
3.4.1	Problème	65
3.4.2	Modélisation	65
3.5	Campagne publicitaire	66
3.5.1	Problème	66
3.5.2	Modélisation	66
3.6	Le problème d'affectation	67
3.7	Ventes avec affectation de personnel	68
3.7.1	Problème	68
3.7.2	Modélisation	69
3.8	Problème d'ordonnancement [18] [19] [20]	70
3.8.1	Organisation de la fabrication : Méthode PERT [18]	70
<b>4</b>	<b><i>Les modèles, modèles complémentaires et résolutions pour la problématique</i></b>	<b>74</b>
	Introduction	74
4.1	Méthode de résolution des modèles d'ordonnancement	74
4.1.1	Algorithmes basés sur la théorie des graphes : Algorithme de Ford [18] [19] [20]	74
4.1.2	Modèle mathématique (Modèle de PERT)	76
4.2	Méthode de résolution du modèle d'affectation simple	79
<b>5</b>	<b><i>Implémentation</i></b>	<b>81</b>
5.1	LINGO : Logiciel pour la résolution des programmes linéaires et non linéaires	81
5.2	Visual Xpress	86
5.3	MATLAB	89
5.4	les exemples d'applications	91
5.4.1	Exemple d'affectation simple	91
5.4.2	L'exemple par kirchoff	92
5.4.3	Exemple PERT	93

# Table des figures

1.1	Organigramme de la CNEP Banque . . . . .	6
1.2	Département crédit . . . . .	8
2.1	Figure 2.1 . . . . .	13
2.2	Comparaison entre un nombre flou et un nombre réel . . . . .	37
2.3	Quelques opérations graphiquement . . . . .	39
2.4	Représentation d'un nombre flou type L-R . . . . .	39
2.5	Représentation d'un nombre flou plat de type L-R. . . . .	40
2.6	Représentation d'un nombre flou triangulaire $(a; \alpha; \beta)$ . . . . .	41
2.7	Représentation d'un nombre flou trapézoïdale . . . . .	42
2.8	Contraintes du type $A_i(x) \lesssim b_i, i = 1, \dots, m$ . . . . .	44
2.9	Contraintes du type $A_i(x) \cong b_i, i = 1, \dots, m$ . . . . .	44
2.10	Contraintes du type $A_i(x) \gtrsim b_i, i = 1, \dots, m$ . . . . .	45
3.1	Exemple d'un arbre de décision pour l'accord d'un prêt . . . . .	62
3.2	Tableau 3.1 – taux proposés par les banques pour les différents projets . . . . .	65
3.3	Tableau 3.2 – Données pour la campagne publicitaire . . . . .	67
3.4	Tableau 3.3 – Temps unitaire de fabrication . . . . .	69
3.5	Tableau 3.4 – Transferts possibles de personnel . . . . .	69
3.6	Tableau de la durée des tâches évaluées en jours . . . . .	71
3.7	Réseau PERT . . . . .	72
5.1	La barre des menus du logiciel . . . . .	83
5.2	La barre des outils de Lingo . . . . .	84
5.3	La barre des outils de Lingo . . . . .	84
5.4	La page d'accueil de Lingo . . . . .	85
5.5	L'icône Visual Xpress . . . . .	87
5.6	Interface de Visual Xpress . . . . .	87
5.7	Espace de travail de Visual Xpress . . . . .	88
5.8	La fenêtre MATLAB . . . . .	90

# *Introduction générale*

La recherche opérationnelle (RO), est une approche quantitative permettant de produire de meilleures décisions. Elle fournit des outils pour rationaliser, simuler et optimiser l'architecture et le fonctionnement des systèmes industriels et économique. Elle propose des modèles pour analyser des situations complexes et permet aux décideurs de faire des choix efficaces. Les problèmes d'optimisations du risque de crédit sont des modèles mathématiques visant à minimiser les pertes ou à maximiser les profits attendues d'une institution financière qui prête de l'argent aux emprunteurs.

Ces dernières années, on a souvent mis l'accent sur le phénomène de risque encouru dans les activités bancaires et principalement le risque de crédit, il existe plusieurs types de risques de crédit, par exemple le non-remboursement. Mais dans ces cas il existe des solutions pour parer à ces difficultés par exemple l'hypothèque.

La Caisse Nationale d'Épargne et de Prévoyance (CNEP-Banque) dont l'activité principale est le financement du secteur de l'habitat qui constitue un secteur stratégique et privilégié de développement économique et social du pays. Le crédit est une assistance financière du banquier à l'égard de son client et c'est un terme désignant des transactions en nature ou en espèces effectuées en contrepartie d'une promesse de remboursement dans un délai généralement convenu. Toutefois, chaque demande de crédit doit faire l'objet d'une étude et d'un suivi du dossier du crédit pour minimiser les risques que la banque encourt.

Dans le cadre de notre mémoire de fin d'étude, on a été affecté à la CNEP T-O pour étudier les crédits (les emprunts), d'un point de vue théorique et pratique. Et pour voir les voies et moyens afin d'optimiser les crédit (les emprunts). Les crédits (emprunts) ont des retombées économiques très importantes pour la banque surtout si la banque en question pratique d'autres activités comme le commercial ou l'investissement, d'ailleurs c'est le cas pour la CNEP T-O qui s'est engagée dans la vente d'immobilier et immobilier de toute sorte, surtout l'opération vente véhicule à ses clients dans le cadre de l'initiative du président de rapprocher les citoyens aux banques avec les ventes de véhicules à la portée des citoyens. Pour organiser par exemple : les activités citées ci-dessous, la banque CNEP doit se doter de nouvelles techniques et technologies pour réussir cette étape cruciale afin de s'imposer dans le monde économique. Dans ce sens, nous proposons des modèles économiques en recherche opérationnelle et en informatique pour mener à bien toutes ces activités.

Ce mémoire est constitué de cinq chapitres. Dans le premier chapitre, nous avons tout d'abord présenté la CNEP et son historique, à savoir : les crédits et les risques de crédit. Le deuxième chapitre, on a cité les notions de la programmation mathématique tel que

programmation linéaire, non linéaire, flou et stochastique. Le troisième chapitre décrit les modèles mathématiques utilisés dans la banque. Dans le quatrième chapitre on trouve les méthodes de résolutions des modèles cités dans le troisième chapitre. Le dernier chapitre aborde les logiciels utilisés pour résoudre les problèmes mathématiques.

# Chapitre 1

## *Présentation de l'organisme d'étude cas CNEP*

### Introduction

La CNEP Banque Tizi Ouzou est une institution financière qui joue un rôle essentiel dans l'économie locale. Fondée il y a plusieurs années, elle s'est développée pour devenir l'une des banques les plus importantes de la région. Elle offre une large gamme de services bancaires aux particuliers et aux entreprises, tels que les opérations bancaires, les crédits et les ressources humaines entre autre. Cette présentation vise à fournir un aperçu détaillé de la CNEP Banque Tizi Ouzou, y compris son organigramme, leurs activités et les étapes pour créer un dossier de crédit par exemple.

### 1.1 Présentation et historique de la CNEP Banque : [1]

La caisse nationale d'épargne et de prévoyance créée le 10 Aout 1964, sous l'égide de la loi n° 64/227 en substitution à la caisse de Solidarité des Départements Communs d'Algérie (CSDCA).

La structure du système financier algérien après l'indépendance était dominée par la spécialisation des institutions financière qui n'ont pas pu assurer d'une manière suffisante :

- Le crédit au logement et le crédit hypothécaire
- Le financement du trésor en faveur des collectivités locales
- La récolte de la petite épargne monétaire industrielle

La CNEP a vécu plusieurs changements au cours de son évolution, que nous allons présenter comme suit :

#### **1ère étape (1964-1970) : collecte de l'épargne**

Cette période était celle de la mise en place de livret d'épargne. Les deux attributions principales assignées à la caisse d'épargne de l'époque étaient :

- La collecte de l'épargne
- L'octroi du crédit pour l'achat de logement (prêts sociaux)

**2em étape (1971-1979) : encouragement du financement de l'habitat** Durant cette période, l'effort était surtout consacré à l'encouragement du financement de l'habitat et au développement de la présence de la caisse sur le marché d'épargne.

Au mois d'avril 1971, une instruction a chargé la CNEP de financer les programmes de réalisation de logements en utilisant les fonds du trésor public. Dès lors, l'épargne des ménages va connaître un essor prodigieux. A la fin de l'année 1975 furent vendus les premiers logements au profit des titulaires de livrets d'épargne.

En 1979, le réseau de la CNEP comptait 46 Agences et bureaux de collecte.

### **La décennie 1980 : la CNEP au service de la promotion immobilière**

Il s'agit des crédits aux particuliers pour la construction de logements et le financement de l'habitat promotionnel au profit exclusif des épargnants. La CNEP entreprit une politique de diversification des crédits accordés notamment en faveur des professions libérales, des travailleurs de la santé, des coopératives de service et des transporteurs.

### **La CNEP après 1997**

A partir de 1997, la CNEP est passée du statut de caisse à celui de banque, sa nouvelle dénomination est désormais la CNEP banque, a été immatriculée au registre de commerce en date du 24/12/2000 sous le numéro 00138291300, aussi tous les dirigeants ont été agréés par le gouverneur de la banque d'Algérie.

Outre ses 209 Agences d'exploitation, la CNEP Banque a signé depuis longtemps une convention avec les PTT (actuellement Algérie Poste) pour la distribution de ses produits via le réseau postal.

**Le 31 Mai 2005 : financement des investissements dans l'immobilier** L'assemblée Générale a décidé, le 31/05/2005, de donner la possibilité à la CNEP Banque de s'impliquer davantage dans le financement des infrastructures et activités liées à la construction, notamment pour la réalisation de biens immobiliers à usage professionnel, administratif, industriel, ainsi que les Infrastructures hôtelières, de santé, éducatives et culturelles.

### **Le 17 juillet 2008 : Repositionnement stratégique de la CNEP Banque**

L'assemblée générale ordinaire du 17 juillet 2008 relative au repositionnement stratégique de la banque décide d'autoriser au titre des crédits aux particuliers :

- Les crédits hypothécaires prévus par les textes réglementaires en vigueur au sein de la banque à l'exclusion des prêts pour l'achat, la construction, l'extension et l'aménagement des locaux à usage commercial ou professionnel

- Pour le financement de la promotion immobilière, sont autorisés :

- Le financement des programmes immobiliers destinés à la vente ou à la location, y compris ceux intégrant des locaux commerciaux ou professionnels.
- Le financement de l'acquisition ou de l'aménagement des terrains destinés à la réalisation de logements.
- Le financement des entreprises

**Repositionnement stratégique de la CNEP-Banque Le 17 août 2011 :** Le repositionnement stratégique a pour objet de définir le champ d'intervention de la CNEP-Banque en matière de financement :

1. Financement de la promotion immobilière est autorisé :

- le financement de l'acquisition ou de l'aménagement de terrains destinés à la réalisation des programmes immobiliers
- le financement de la réalisation d'opérations de promotion immobilière
- le financement de l'acquisition de biens immobiliers à achever ou à rénover.

2. Financement des entreprises : sont autorisés :

- Le financement des investissements de tous les secteurs d'activité économique y compris le fonds de roulement nécessaire au démarrage de l'activité
- Les crédits par signature
- Le leasing immobilier
- Les services liés à l'habitat (bureaux d'études, entreprises d'entretien d'immeubles. . .)

## **1.2 Organisation de la CNEP Banque TIZI OUZOU [2]**

La CNEP Banque est gérée par un conseil d'Administration qui comprend outre le Président Directeur Général nommé par décret et choisi en fonction de sa compétence en matière économique et financière, cinq administrateurs qui représentent les divers ministères intéressés à sa gestion, soit :

- Le ministère de l'intérieur
- Le ministère de l'économie et des finances
- Le ministère des travaux publics
- Le ministère des affaires sociales
- Le ministère des postes et télécommunication

**Organigramme de la CNEP Banque est comme suit**

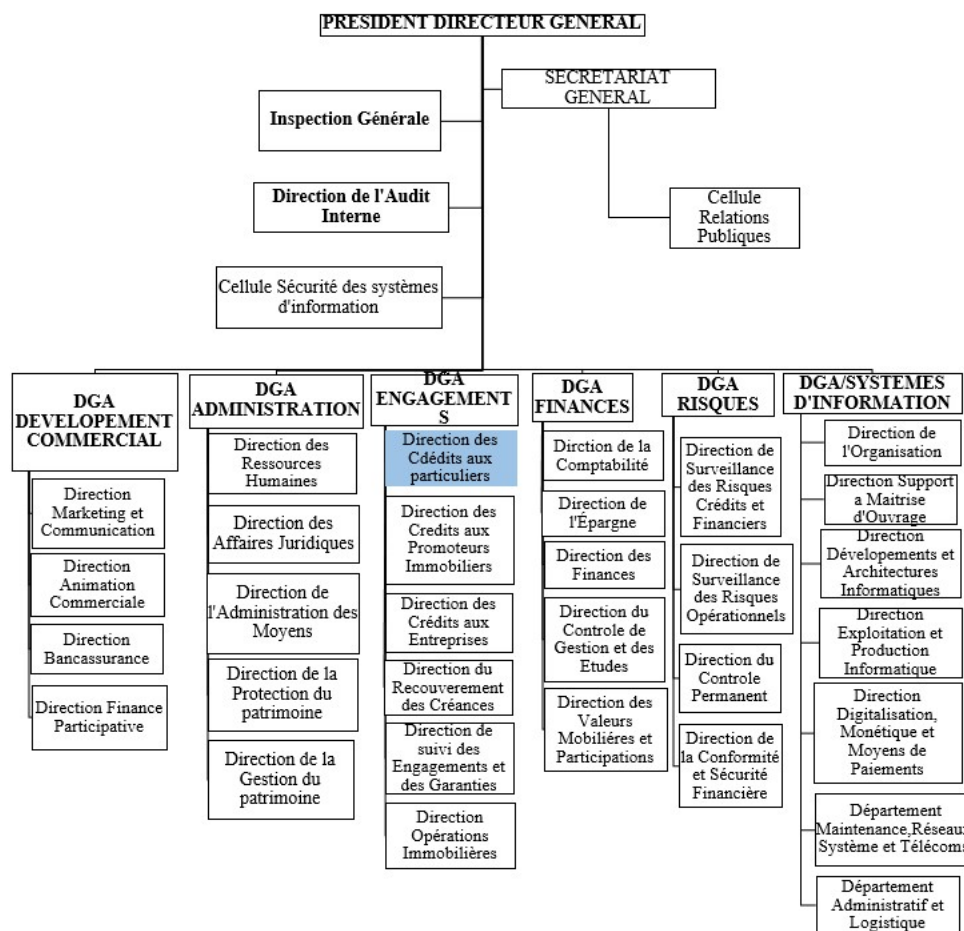


FIGURE 1.1 – Organigramme de la CNEP Banque

### 1.2.1 Missions et valeurs de la banque CNEP [2]

La banque CNEP a pour mission principale de fournir des services financiers de qualité à ses clients, tout en favorisant l'épargne nationale et en contribuant au développement économique du pays. Ses valeurs fondamentales sont l'intégrité, la transparence, l'innovation et la satisfaction client. La banque s'engage à offrir des solutions adaptées aux besoins de ses clients, à respecter les normes éthiques et à promouvoir des pratiques responsables dans toutes ses activités.

#### Définition d'un crédit :

Le crédit, est une opération qui consiste pour un prêteur (banque ou organisme financier), à mettre à disposition d'un emprunteur une certaine somme d'argent moyennant un engagement de remboursement à une date déterminée à l'avance.

#### Objectif du dossier crédit :

L'objectif principal d'un dossier crédit, est de permettre à la banque d'évaluer la solvabilité et la capacité de remboursement du demandeur. En fournissant des informations détaillées sur sa situation financière, le demandeur permet à la banque de prendre une décision éclairée quant à l'octroi ou au refus du crédit demandé. Le dossier crédit vise également à garantir la transparence et l'équité dans le processus de demande de prêt,

en offrant à la banque les éléments nécessaires pour évaluer les risques associés.

### **Importance de la création d'un dossier crédit :**

La création d'un dossier crédit, revêt une importance majeure tant pour le demandeur que pour la banque. Pour le demandeur, constituer un dossier crédit complet et bien préparé permet d'augmenter ses chances d'obtenir un crédit. En présentant toutes les informations demandées de manière précise et documentée, le demandeur montre sa capacité à gérer ses finances et à rembourser le prêt demandé. Pour la banque, il est primordial de disposer d'un dossier crédit complet afin de réaliser une évaluation rigoureuse du risque associé à l'accord d'un crédit.

Cela permet à la banque de prendre ses décisions éclairées et de protéger ses intérêts financiers.

## **1.2.2 Présentation du Réseau de Tizi-Ouzou**

L'agence réalise un volume d'activité très important par rapport aux autres agences de la ville de Tizi-Ouzou, elle est composée d'une direction qui se divise en deux parties : Front office et Back office, répartie sur plusieurs services.

### **Front office**

Front office est l'ensemble de personnels qui sont chargés de la réception de la clientèle et ont pour mission de fournir des informations sur les opérations de liquidités, les pièces nécessaires à fournir et des différentielles orientations sur les crédits hypothécaires.

Elle est composée de quatre (04) sous parties, à savoir :

- **Chargés de la clientèle (particulier et entreprise) :** Ce service s'occupe d'ouverture du compte et suivi, prospection de la clientèle, souscription de produits d'épargne et du crédit, revenue des comptes inactifs et successions.
- **Guichet payeur /caisse** Ce guichet assure les opérations transactionnelles, versement ou bien retrait d'espèce, remise cheque, remise versement déplacé, réception de la demande de la clientèle.
- **Accueil /orientation :** Qui a comme charge : Accueil (information et orientation), distribution des bordereaux, des imprimés et des listes de pièces nécessaires.
- **Direction agence :** Ayant comme tâche : animation commerciale, rôle éventuel de conseiller de clientèles entreprises.

### **Back office**

Le Back office est l'ensemble du personnel qui se charge de l'étude et des traitements des dossiers avec la décision de l'octroi du crédit

## **1.2.3 Le processus de traitement d'une demande de crédit**

Le processus de traitement d'une demande de crédit bancaire, commence par la réception et la vérification du dossier du client. Une fois validé le dossier passe par une phase du traitement qui comprend des consultations, incluant des éléments de vigilance, interdit chéquier (I.C), consultation web (C.W) et CREM, des simulations. Une fiche technique est alors élaborée. Le dossier est ensuite soumis au comité de crédit, qui peut

émettre un avis favorable, ajournement ou rejets. En cas d'avis favorable, l'Agence établit une convention et en informe le client, incluant les détails de l'assurance nécessaire. Le client signe la convention, après la direction procède à l'enregistrement et à la création du dossier dans le système.

Pour bien comprendre comment créer un dossier crédit, on a visité le département crédit qui contient deux services : service crédit aux particuliers et service promotionnel et investissement.

On s'intéresse au service crédit aux particuliers :

Le processus de demande de crédit débute par l'accueil du client au front office de l'agence, où les documents nécessaires sont recueillis. Ensuite, le dossier du client est acheminé vers le back office pour une analyse approfondie de sa solvabilité et de son historique de dettes. Une fois le dossier examiné, il est présenté au Comité Crédit Agence (CCA) si le montant du crédit sollicité est inférieur à 5.000.000 DA (5 millions de dinars). Si tel est le cas, une décision est prise et le client est enregistré dans le système d'information en vue de la création de son compte. Toutefois, si le montant du crédit dépasse 5.000.000 DA, le dossier est soumis au Comité Crédit Régional (CCR) pour une évaluation plus approfondie. Enfin, si le montant excède 20.000.000 DA, le dossier est transmis au Comité Crédit Central (CCC) pour une décision finale quant à l'octroi du crédit. Ce processus rigoureux garantit une évaluation minutieuse de chaque demande de crédit, en assurant la sécurité financière de la banque tout en répondant aux besoins des clients.

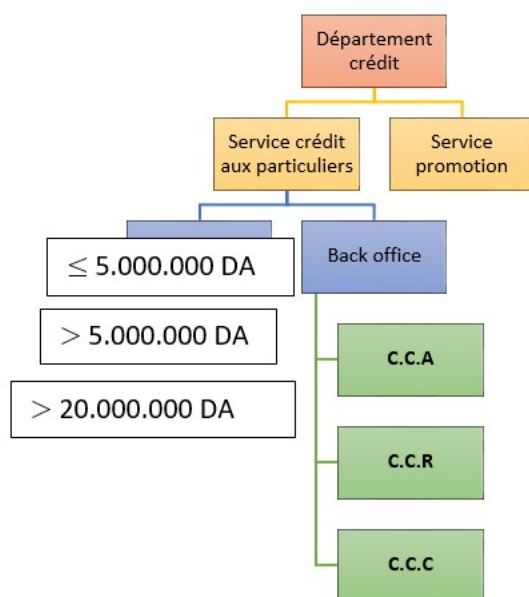


FIGURE 1.2 – Département crédit

### 1.2.4 Les produits de la CNEP [3]

Afin de réaliser les projets immobiliers, une série de produit a été mise en place par la CNEP/Banque :

- Les crédits à l'accession

- L'achat d'une habitation auprès d'un promoteur
- La vente sur plan (VSP)
- L'achat de terrain
- L'achat d'un local
- Les crédits à la construction
- Achat d'un logement auprès d'un particulier
- Logement social aidé ou participatif (LSP)

### **1.2.5 Différents risques liés à l'opération de crédit [2]**

#### **Le risque de non-remboursement**

Le risque de non-remboursement correspond au risque de perte définitive de la créance d'une banque sur son client. Celui-ci se précise quand la situation financière commence à se détériorer et/ou quand le client commence à faire preuve de mauvaise foi vis-à-vis de son bailleur de fonds. La conséquence de ces deux éléments sera l'incapacité ou le refus par le client des sommes dont il est redevable vis-à-vis de sa banque

#### **Le risque d'immobilisation**

est accentué par le non-remboursement des échéances à bonnes dates par les clients de la banque ou par une politique inadéquate de transformation des ressources à vue (dépôts) par le banquier en emplois à terme (tels les crédits).

#### **Le risque de taux**

Le risque de taux résulte de l'évolution divergente du coût des emplois de la banque avec le coût de ses ressources. Afin de minimiser ce risque, le banquier doit opter pour des taux d'intérêt variables sur le crédit, directement fonction des ressources de la banque.

#### **Le risque de chance**

Ce risque est lié à l'évolution des cours des devises par rapport à la monnaie nationale. Dans le cas d'une opération commerciale, il sera subi par l'exportateur qui voit le cours de la monnaie de facturation se déprécier par rapport à sa monnaie nationale, et par l'importateur qui voit la monnaie de transaction s'apprécier par rapport à sa monnaie nationale. Pour se protéger contre ce risque, le banquier peut faire signer à son client un engagement de prise en charge du risque de change.

# Chapitre 2

## *Présentation générale de l'optimisation*

### Introduction

Ce travail présente une vue d'ensemble des différents sujets abordés dans ce chapitre. Ce dernier est divisé en trois sections principales, la première section, intitulée programmation linéaire et non linéaire, abordera les définitions, les méthodes de résolution. La deuxième section programmation floue et stochastique, se focalisera sur les principes de la programmation floue ainsi que sur le modèle stochastique utilisée.

### 2.1 Programmation linéaire

#### 2.1.1 Définition de l'optimisation :

On général l'optimisation est une branche des mathématiques dans la pratique, on part d'un problème concret, on le modélise et on le résout mathématiquement, analytiquement et numériquement. L'optimisation joue un rôle important en recherche opérationnelle (domaine à la frontière entre l'informatique, les mathématiques et l'économie entre autres).

#### **Définition 2.1 : (fonction linéaire) [21]**

Une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite linéaire si elle s'écrit sous la forme suivante :

$$f(x) = c'x = \sum_{i=1}^n c_i x_i$$

$c$  : un vecteur de  $\mathbb{R}^n$  constant, indépendant de  $x$ . Fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  est linéaire si chacune de ses composantes  $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$   $i = 1, \dots, m$  est linéaire. Dans ce cas, elle peut s'écrire  $f(x) = Ax$  où  $A \in \mathbb{R}^{m \times n}$  est une matrice de constantes.

#### 2.1.2 Définition d'un programme linéaire :

Un problème de programmation linéaire (PL) consiste à optimiser (maximiser ou minimiser) une fonction objectif linéaire  $z$  dépendante de  $n$  variables de décision tout en vérifiant un ensemble de contraintes linéaires (égalités et /ou inégalités) . [9]

Un programme linéaire est défini par un ensemble de  $n$  variables réelles  $x_1, \dots, x_n$  et  $m$  contraintes.

$\sum_{j=1}^n a_{ij}x_j (\leq, \geq \text{ ou } =) b_i$ , pour  $i = 1, 2, \dots, m$  et une fonction d'optimisation (objectif)

$$f : \mathbb{R}^n \rightarrow \mathbb{R}$$

$$\mathbf{x} = (x_1, \dots, x_n)^T \mapsto f(\mathbf{x}) = (x_1, \dots, x_n) \cdot \sum_{j=1}^n c_j x_j$$

$$\text{où } A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix}, \quad c = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

### 2.1.3 Modélisation [4]

Un modèle est une construction mathématique utilisée pour représenter certains aspects significatifs des problèmes du monde réel. Il existe différents types de modèles mathématique, mais nous nous focaliserons sur les modèles d'optimisation (PL). Il y a trois composantes principales dans un modèle d'optimisation entre autres :

- **Variables** : elles représentent les composantes du modèle qui peuvent être modifiées pour créer des configurations différentes.
- **Contraintes** : elles représentent les limitations sur les variables.
- **Fonction objectif** : cette fonction assigne une valeur à chaque configuration différente. Le terme « objectif » vient du fait que l'objectif est d'optimiser cette fonction.

### 2.1.4 Forme d'un programme linéaire :

Générale [9]

$$(PL) = \begin{cases} z(x) = \sum_{j=1}^n c_j x_j \rightarrow (max/min) \\ \sum_{j=1}^n a_{ij} x_j (\leq, \geq \text{ ou } =) b_i, \quad i = 1, \dots, m \\ x_j \geq 0, \quad \forall j = 1, \dots, n \end{cases} \quad (2.1)$$

où

$$(PL) = \begin{cases} z(x) = \sum_{j=1}^n c_j x_j \rightarrow (max/min) \\ \sum_{j=1}^n a_{ij} x_j (\leq, \geq \text{ ou } =) b_i, \quad i = 1, \dots, m \end{cases} \quad (2.2)$$

Les  $c_j$ ,  $a_{ij}$ ,  $b_i$  sont des nombres réels.

**canonique (matricielle)**

$$(PL) = \begin{cases} \max / \min & z(x) = c'x \\ & Ax \leq / \geq b \\ & x \in \mathbb{R}^n, \quad x \geq 0 \end{cases} \quad (2.3)$$

**Standard**

$$(PL) = \begin{cases} \max / \min & z(x) = c'x \\ & x \in \mathbb{R}^n \\ & Ax = b \\ & x \geq 0 \\ & b \in \mathbb{R}^m, \quad b \geq 0 \end{cases} \quad (2.4)$$

**Remarque 2.1 :** Les problèmes de minimisation et de maximisation sont équivalents :  $\min z = -\max(-z)$  et  $\max(z) = -\min(-z)$ .

**2.1.5 Domaine de solutions réalisables et solution optimale [5] :**

- **Solution réalisable :** tout vecteur  $x$  qui satisfait les contraintes fonctionnelles et les Contraintes de non négativité est appelé solution réalisable aux modèles de programmation Linéaire (2.3) et (2.4) .
- **Domaine des solutions réalisable :** c'est l'ensemble des solutions réalisables du programme linéaire c'est-à-dire l'ensemble des solutions qui satisfont simultanément les contraintes fonctionnelles et les contraintes de non négativité pour (2.3) et (2.4).
- **Solution optimale :** toute solution réalisable qui optimise la fonction objective pour (2.3) et (2.4).

**2.1.6 Propriétés des programmes linéaires****Définition 2.3 : (Hyperplan) [22]**

soit  $a = (a_1, a_2, \dots, a_n)' \in \mathbb{R}^n$  et  $\beta \in \mathbb{R}$ , avec le vecteur  $a \neq 0$  alors  $H = \{x \in \mathbb{R}^n / a'x = \beta\}$  est un hyperplan de  $\mathbb{R}^n$

**Propriété 2.1 :(Ensemble convexe) [22]**

Un ensemble  $C \subseteq \mathbb{R}^n$  est convexe, si seulement si  $\forall A, B \in C :$   
 $\alpha A + (1 - \alpha)B \in C, \forall \alpha \in [0, 1]$

Ou bien :  $C$  est convexe si et seulement si le segment reliant tout couple  $A$  et  $B$  de  $C$  est dans  $C$  .

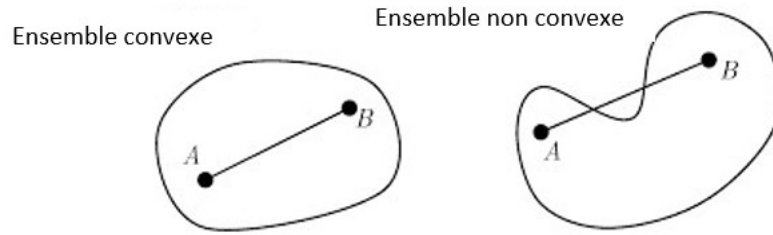


FIGURE 2.1 – Figure 2.1

**Propriétés 2.2 [22] :**

- Soit  $S_1, S_2$  deux ensembles convexes de  $\mathbb{R}^n$  alors :  
 $S_1 + S_2 = \{x \in \mathbb{R}^n / x = c + d, c \in S_1, d \in S_2\}$  est convexe.
- Soit  $S \subset \mathbb{R}^n$  convexe alors :  
 $\alpha S = \{\alpha x / x \in S\}$  est convexe.
- Soit  $S_1, S_2, \dots, S_p$ , des ensembles convexes inclus dans  $\mathbb{R}^n$  :  $S = \bigcap_{i=1}^p S_i$   
 $S$  : est un ensemble convexe

**2.1.7 Méthodes de résolution d'un programme linéaire :****Méthode simplexe [23] :**

La méthode du simplexe a été développée en 1947 par George Dantzig, elle reste d'actualité pour résoudre des problèmes de grande taille. Cette méthode évolue sur la frontière du domaine réalisable de sommet en sommet adjacent, en optimisant la valeur de l'objectif jusqu'à l'optimum. Un critère d'optimalité simple permet de reconnaître le sommet optimal éventuel.

**Base et solution de base**

Compte tenu du problème (2.4)

$I = \{1, 2, \dots, m\}$  Ensemble des indices de lignes.

$J = \{1, 2, \dots, n\}$  Ensemble des indices de colonnes.

**Base :**

C'est une sous matrice  $A_B$  d'ordre  $m$  extraite de  $A$  de déterminant non nul ( $\det(A_B) \neq 0$ ), les vecteurs colonnes de  $A_B$  sont linéairement indépendants (LI), et les variables associées aux colonnes  $A_B$  sont appelées variables de base et les  $n - m$  variables restantes sont appelées variables hors base. Dans ce cas,  $A$  et  $x$  sont décomposées de la manière suivante :

$$A = [A_B, A_H], x = \begin{pmatrix} x_B \\ x_H \end{pmatrix}$$

$$Ax = b \iff [A_B, A_H] \begin{pmatrix} x_B \\ x_H \end{pmatrix} = b \implies A_B x_B + A_H x_H = b$$

**Solution de base :**

Une solution réalisable  $x$  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 .

$J_H = J \setminus J_B$  Ensemble des indices hors base.

Autrement dit , Une solution réalisable  $x = x(J)$  est une solution de base ssi :

$$x_H = x(J_H) = 0$$

$$\det A_B \neq 0 \text{ où } A_B = A(I, J_B) \text{ et } x(J_B) \geq 0.$$

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 de hors base.

Une solution réalisable de base  $x$  est dite non dégénérée si :

$$x_j > 0, j \in J_B$$

L'accroissement de la fonction objectif  $z$  est égale à :

$$\Delta z = z(\bar{x}) - z(x) = c'\bar{x} - c'x = c'\Delta x, (\bar{x} = x + \Delta x)$$

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

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

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

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

### **Théorème 2.1 [9]**

Soit  $\{x, A_B\}$  une solution réalisable de base 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  $\{x, A_B\}$ .

#### **Algorithme du simplexe [24] (cas maximisation) :**

Soit  $\{x, A_B\}$  une solution réalisable de base de départ et supposons que le critère d'optimalité n'est pas vérifié, c'est-à-dire l'inégalité  $\Delta_j \geq 0, j \in J_H$ . Pour vérifier on va Choisirez l'indice :

- Le but de l'itération est de faire rentrer l'indice  $j_0$  dans la base (autrement dit la colonne  $a_{j_0}$  va rentrer dans la base ). Donc il faut trouver un indice  $j_1 \in J_B$ , qui sortira de la base ( à cet indice correspond la colonne  $a_{j_1} \in A_B$ , Et ceci constitue l'itération qui procède au passage de la solution de base (point extrême)  $x, A_B$  à la solution  $\{\bar{x}, \bar{A}_B\}$  sommet voisin ) et tel que  $z(\bar{x}) \geq z(x)$

La nouvelle solution de base  $\bar{x}$  sera trouvée de la manière suivante :

$$\bar{x} = x + \theta l$$

$l$  est la direction de changement de  $x$  et  $\theta$  le pas de cette direction.

- Sur  $J_H$  posons :  $l_j = \begin{cases} 0, & \text{si } j \in J_H/j_0 \\ 1, & \text{si } j = j_0 \end{cases}$
- Sur  $J_B$  :  $\bar{x}$  doit être réalisable , donc elle doit vérifier  $A\bar{x} = b$  et comme  $Ax = b$  donc  $\theta Al = 0$  de cette dernière relation on obtient :

$$l_B = l(J_B) = -A_B^{-1} A_H l_H$$

De là  $\bar{x}_H = x_H + \theta l_H = \theta l_H \geq 0$  et  $\bar{x}_B = x_B + \theta l_B = x_j - \theta A_B^{-1} a_{j_0} \geq 0$

- Si les composantes du vecteur  $A_B^{-1} a_{j_0} \leq 0$  alors  $\bar{x}_j \geq 0, \forall \theta \geq 0$ , donc on peut prendre  $\theta = \infty$  et on aura une solution infinie.

— Pour avoir  $\bar{x}_B \geq 0$ , il faut prendre un pas maximal  $\theta^0$  :

$$\theta^0 = \min_{j \in J_B} \theta_j = \min \left\{ \frac{x_j}{x_{j0j}} / x_{j0j} > 0, j \in J_B \right\} = \theta_{j_1}, \quad j_1 \in J_B.$$

La nouvelle base sera :

$$\bar{J}_B = (J_B \setminus \{j_0\}) \cup \{j_1\} \text{ and } \bar{A}_B = (A_B \setminus \{a_{j_0}\}) \cup \{a_{j_1}\}$$

**Tableau du simplexe [24] [25]**

Les différents calculs qu'on aura à effectuer dans les différentes étapes de résolution seront disposés dans le tableau suivant :

$c$			$c_1$	$c_2$	$c_3$	$\dots$	$c_m$	$c_{m+1}$	$\dots$	$c_j$	$\dots$	$c_n$	
$c_B$	Base	$b$	$a_1$	$a_2$	$a_3$	$\dots$	$a_m$	$a_{m+1}$	$\dots$	$a_j$	$\dots$	$a_n$	$\theta_j$
$c_1$	$a_1$	$b_1 = x_1$	1	0	0	$\dots$	0	$x_{1,(m+1)}$	$\dots$	$x_{1j}$	$\dots$	$x_{1n}$	$\theta_1$
$c_2$	$a_2$	$b_2 = x_2$	0	1	0	$\dots$	0	$x_{2,(m+1)}$	$\dots$	$x_{2j}$	$\dots$	$x_{2n}$	$\theta_2$
$c_3$	$a_3$	$b_3 = x_3$	0	0	1	$\dots$	0	$x_{3,(m+1)}$	$\dots$	$x_{3j}$	$\dots$	$x_{3n}$	$\theta_3$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	
$c_m$	$a_m$	$b_m = x_m$	0	0	0	$\dots$	1	$x_{m,(m+1)}$	$\dots$	$x_{mj}$	$\dots$	$x_{mn}$	$\theta_m$
$Z =$	$\mathbf{c}_B' x_B$	$\Delta_j$	$\Delta_1 = 0$	$\Delta_2 = 0$	$\Delta_3 = 0$	$\dots$	$\Delta_m = 0$	$\Delta_{m+1}$	$\dots$	$\Delta_j$	$\dots$	$\Delta_n$	

**Exemple 2.1 :** Soit le programme linéaire suivant :

$$\begin{cases} \max z = 10x_1 + 15x_2 \\ 5x_1 + 2x_2 \leq 80 \\ x_1 + x_2 \leq 20 \\ x_1 + 2x_2 \leq 30 \\ x_1 \geq 0, \quad x_2 \geq 0 \end{cases} \quad (2.6)$$

Transformons le programme (2.6) sous forme standard en ajoutant les variables d'écart  $x_3, x_4, x_5$

$$\begin{cases} \max z = 10x_1 + 15x_2 \\ 5x_1 + 2x_2 + x_3 = 80 \\ x_1 + x_2 + x_4 = 20 \\ x_1 + 2x_2 + x_5 = 30 \\ x_i \geq 0, \quad i = 1, \dots, 5 \end{cases} \quad (2.7)$$

La solution de base de départ est  $x = (0, 0, 80, 20, 30)'$  avec  $J_B = \{3, 4, 5\}$  et  $J_H = \{1, 2\}$

Le vecteur des potentiels est :

$$y' = \mathbf{c}_B A_B^{-1} = (0, 0, 0) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = (0, 0, 0)$$

Déterminons les vecteur des estimations  $\Delta$  :

$$\Delta_j = y'a_j - c_j$$

$$\Delta_1 = y'a_1 - c_1 = (0, 0, 0) \begin{pmatrix} 5 \\ 1 \\ 1 \end{pmatrix} - 10 = -10$$

$$\Delta_2 = y'a_2 - c_2 = (0, 0, 0) \begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix} - 15 = -15$$

$$\Delta_3 = \Delta_4 = \Delta_5 = 0, \text{ car } \Delta_j = 0 \text{ pour } j \in J_B.$$

Le critère d'optimalité n'est pas vérifié donc la solution de départ n'est pas optimale.

**Initialisation du tableau du simplexe :**

$C$			10	15	0	0	0	
$C_B$	Base	$b$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\theta_j$
0	$x_3$	80	5	2	1	0	0	$80 \div 2 = 40$
0	$x_4$	20	1	1	0	1	0	20
0	$x_5$	30	1	2	0	0	1	15
		$\Delta_j$	-10	-15	0	0	0	

La solution de départ n'est pas optimale car :

$$\min\{\Delta_j < 0\} \quad (\Delta_1 = -10 < 0, \text{ et } \Delta_2 = -15 < 0)$$

La variable  $x_2$  va entrer en base.

**Première itération du tableau :**

Pour déterminer la colonne qui va sortir de la base, on calculons les  $\theta_j, j \in J_B$

$$\theta_j = \frac{x_j}{x_{j0j}}, \quad x_{j0j} > 0, \quad j \in J_B$$

$$\theta_3 = \frac{80}{2} = 40$$

$$\theta_4 = \frac{20}{1} = 20$$

$$\theta_5 = \frac{30}{2} = 15$$

$$\min_{j \in J_B} \theta_j = \theta_5 = 15, \text{ donc } x_5 \text{ sort.}$$

$$\text{D'où } J_B = \{3, 4, 2\} \text{ et } J_H = \{1, 5\}$$

$C$			10	15	0	0	0	
$C_B$	Base	$b$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\theta_j$
0	$x_3$	50	4	0	1	0	-1	12.5
0	$x_4$	5	$\frac{1}{2}$	0	0	1	$-\frac{1}{2}$	10
15	$x_2$	15	$\frac{1}{2}$	1	0	0	$\frac{1}{2}$	30
		$\Delta_j$	-2.5	0	0	0	7.5	

**Deuxième itération du tableau :**

La variable  $x_1$  va remplacer  $x_4$  dans la base.

$C$			10	15	0	0	0	
$C_B$	Base	$b$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\theta_j$
0	$x_3$	10	0	0	1	-8	3	/
10	$x_4$	10	1	0	0	2	-1	/
15	$x_5$	10	0	1	0	-1	1	/
		$\Delta_j$	0	0	0	5	5	

Le critère d'optimalité est vérifié, donc la solution  $\bar{x} = (10, 10, 10, 0, 0)$  est optimale pour programme standard, donc la solution optimale de (2.7) est  $x^0 = (10, 10, 10)$  avec  $z^0 = 250$

### 2.1.8 Initialisation de l'algorithme du simplexe :

Il est impératif d'avoir une solution de base réalisable de départ pour pouvoir utiliser la méthode du simplexe. Dans plusieurs situations, celle-ci n'est pas toujours facile à obtenir. La méthode des deux phases et la M-méthode que nous allons exposer dans la suite de ce chapitre sont des variantes de la méthode du simplexe.

#### La M-Méthode [24]

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

$$\text{Soit le problème : } (P) \begin{cases} \max z = c'x \\ Ax = b \\ x_j \geq 0, \quad j = 1, \dots, n \end{cases}$$

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

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

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

Le vecteur  $X = (0, b)' = (x_i = 0, x_{n+i} = b_i, i = 1..m)'$  est une solution de base réalisable de (PM). On résout le problème (PM) par la méthode du simplexe avec une solution réalisable de base de départ  $\{(0, b), A_B\}$  et on obtient une solution optimale  $\{X^0 = (x^0, x_{n+i}^0, i = 1 \dots m), A_B^0\}$ .

#### 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  $\Delta_{j0} = 0, \quad \forall j_0 \in J_H$  alors on aura une infinité de solutions optimales.
2. Au moins une variable artificielle de la base strictement positive cela implique contraintes contradictoires.
3. À 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 2.2 :** Soit à résoudre le problème suivant :

$$(P) \begin{cases} z = -2x_1 + 2x_2 \rightarrow \max \\ x_1 + x_2 \geq 2 \\ x_1 \geq 0 \\ 2x_1 + x_2 \geq 3 \\ x_2 \geq 0 \end{cases}$$

Le M-problème correspondant à (P) est :

$$\begin{cases} z_M = -2x_1 + 2x_2 - Mx_5 - Mx_6 \rightarrow \max \\ x_1 + x_2 - x_3 + x_5 = 2 \\ 2x_1 + x_2 - x_4 + x_6 = 3 \\ x_j \geq 0, \quad j = 1, \dots, 6 \end{cases}$$

Let  $J = \{1, 2, \dots, 6\}$ ,  $J_B = \{5, 6\}$ , and  $J_H = \{1, 2, 3, 4\}$ .

$$\Delta'_B = [0] (\text{Vecteur})$$

$$\Delta'_H = C'_B A_B^{-1} A_H - C'_H$$

$$\Delta'_H = (-M, -M) \begin{pmatrix} 1 & 1 & -1 & 0 \\ 2 & 1 & 0 & -1 \end{pmatrix} - (-2, 2, 0, 0)$$

$$= (-3M, -2M, M, M) - (-2, 2, 0, 0) = (-3M + 2, -2M - 2, M, M)$$

$$x = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 2 \\ 3 \end{pmatrix} \Rightarrow z_M(x) = -5M$$

**Initialisation :**

base	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$b$
$x_5$	1	1	-1	0	1	0	2
$x_6$	2	1	0	-1	0	1	3
$\Delta'$	$-3M + 2$	$-2M - 2$	$M$	$M$	0	0	$z_M = -5M$

**1er itération :**

base	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$b$
$x_5$	0	1/2	-1	1/2	1	-1/2	1/2
$x_1$	1	1/2	0	-1/2	0	1/2	3/2
$\Delta'$	0	$-\frac{M}{2} - 3$	$M$	$-\frac{M}{2} + 1$	0	$\frac{3}{2}M - 1$	$z_M = -\frac{M}{2} - 3$

**2er itération :**

base	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$b$
$x_2$	0	1	-2	1	2	-1	1
$x_1$	1	0	1	-1	-1	1	1
$\Delta'$	0	0	-6	4	$M + 6$	$M - 4$	$z_M = 0$

**3em itération :**

base	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$b$
$x_2$	2	1	0	-1	0	1	3
$x_3$	1	0	1	-1	-1	1	1
$\Delta'$	6	0	0	-2	$M$	$M + 2$	$z_M$

Impossible de continuer  $\Rightarrow$  selon l'algorithme, la valeur de la solution optimale tend vers  $+\infty$ .

**La méthode de deux phases :**

Considérons le problème de la programmation linéaire suivant :

$$(P) \begin{cases} z = c'x \rightarrow \max \\ Ax \leq b, \quad x_j \geq 0, \quad j = 1, \dots, n \end{cases}$$

**Première phase**

La première phase de résolution du problème  $P$  consiste à déterminer une solution de base réalisable de  $P$ . Pour cela, on construit le problème suivant :

$$(P') \begin{cases} (-\sum_{i=1}^m x_{n+i} \rightarrow \max) \\ [Ax]_i + x_{n+i} = b_i, \quad i = 1, \dots, m \\ x \geq 0, \quad x_{n+i} \geq 0, \quad i = 1, \dots, m \end{cases}$$

Où  $x_{n+i}$  sont appelés des variables artificielles .

Le problème  $(P')$  possède  $n+m$  variables et  $m$  equations . Le vecteur  $x = (0, \dots, 0, b_1, \dots, b_m)$  réalisable pour  $(P')$  donc l'ensemble des solutions admissibles de  $(P')$  est non vide .

D'un autre côté, la fonction objectif  $-\sum_{i=1}^m x_{n+i} \leq 0$ , donc le problème  $(P')$  admet une solution optimale  $X^0 = (x^0, x_a^0)$ .

**Deuxième phase** Soit  $\{x^0, x_a^0\}$  une solution optimale de  $(P')$  avec des variables artificielles nulles. Alors, on utilise la solution  $x^0$  avec sa matrice  $A_B^0$  comme solution de base de départ du problème  $(P)$ , et ceci constitue la deuxième phase. Si  $x_{n+i}^0 = 0$  pour tout  $i$  et il existe un indice  $i_0$  tel que  $a_{i_0} \in A_B^0$ , alors pour revenir à la deuxième phase, pour exclure cette colonne de  $A_B^0$ .

**Exemple 2.3 :** Résoudre avec deux phase le problème suivant :

$$(P) \begin{cases} z = x_1 + 2x_2 + x_3 \\ -x_1 + x_2 - x_3 = 10 \\ 2x_1 - x_2 - x_3 = 5 \\ x_j \geq 0 \quad \text{pour } j = 1, 2, 3 \end{cases}$$

Tout d'abord construisons, le problème Auxiliaire (pb aux) du problème initial

$$(pb \text{ aux}) \begin{cases} \bar{z} = -x_4 - x_5 \rightarrow \max \\ -x_1 + x_2 - x_3 + x_4 = 10 \\ 2x_1 - x_2 - x_3 + x_5 = 5 \\ x_j \geq 0 \quad \text{pour } j = 1, \dots, 5 \end{cases}$$

$x_4$  et  $x_5$  sont des variables artificielles. On a  $J = \{1, 2, 3, 4, 5\}$  et  $J_B = \{4, 5\}$ .

**1er phase**

o Initialisation du simplexe

Base	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$b$
$x_4$	-1	1	-1	1	0	10
$x_5$	2	-1	1	0	1	5
$\Delta'$	-1	0	0	0	0	$\bar{z} = -15$

$$\Delta'_H = \mathbf{c}'_B A_B^{-1} A_H - \mathbf{c}'_H$$

$$\mathbf{c}'_B = (-1, -1), \quad A_B^{-1} = \begin{pmatrix} -1 & 1 & -1 \\ 2 & -1 & 1 \end{pmatrix}, \quad A_H = \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix}, \quad \mathbf{c}'_H = (0, 0, 0)$$

$$\mathbf{c}'_B A_B^{-1} A_H = (-1, -1) \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix} = ((-1)(-1) + (-1)(1) + (-1)(-1)) = (-1)$$

$$\Delta'_H = (-1, 0, 0) - (0, 0, 0) = (-1, 0, 0)$$

o 1er itération :

Base	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$b$
$x_4$	0	1/2	-1/2	1	1/2	25/2
$x_1$	1	-1/2	1/2	0	1/2	5/2
$\Delta'$	0	-1/2	1/2	0		$\bar{z} = -125/2$

o 2em itération :

Base	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$b$
$x_2$	0	1	-1	2	1	25
$x_1$	1	0	0	1	1	15
$\Delta'$	0	0	0	1	1	$\bar{z} = 0$

Tableau optimal, les variables artificielles sont hors base et  $z = 0$ . On a obtenu des solutions de départ pour le problème initial.

### 2em phase

On aborde la 2eme phase sur les variables artificielles :

$$z = x_1 + x_2 + x_3 \rightarrow \max$$

Base	$x_1$	$x_2$	$x_3$	$b$
$x_2$	0	1	-1	25
$x_1$	1	0	0	15
$\Delta'$	0	0	-3	$z = 65$

$$\Delta'_H = (2 \quad 1) I_2 \begin{pmatrix} -1 \\ 0 \\ -3 \end{pmatrix} - 1 = -3$$

Impossible de pivoter par manque de pivot et donc la solution optimale tend vers l'infini

La conclusion est que  $\max z = +\infty$ .

### Méthode des coupes (GOMORY) :

La méthode des coupes de Gomory est une technique utilisée en Recherche opérationnelle, pour résoudre des problèmes linéaires en nombres entiers. Cette méthode développée

par Ralph Gomory en 1958, consiste à générer des coupes pour améliorer la relaxation linéaire d'un problème.

**Algorithme de Gomory [5] :**

- Étant donné un (PLNE), résoudre le PL correspondant à l'aide d'un algorithme du simplexe. Si la solution optimale du PL est entière, elle est également une solution optimale au PLNE. La résolution est terminée.

- Si une ou plusieurs variables de base dans la solution optimale du PL ne sont pas entières, on doit alors générer à partir d'une des lignes du tableau (celle dont la partie fractionnaire pour la variable de base correspondante est plus élevée) une contrainte supplémentaire dite coupe de GOMORY. Cette contrainte est ajoutée au tableau optimal du PL et on détermine le nouveau tableau optimal à l'aide de la méthode duale du simplexe.

- Si les variables de bases dans le nouveau tableau optimal sont entières, nous avons obtenu la solution du PLNE. La résolution est terminée.

- Sinon, on doit générer à partir du dernier tableau optimal une nouvelle coupe de GOMORY, l'ajouter au dernier tableau optimal et trouver la solution optimale à valeurs entières.

**Règle pour générer la coupe de GOMORY**

On choisira, dans le tableau optimal, la variable de base ayant la plus grande partie fractionnaire et on déduit l'équation de la coupe en exprimant toutes les variables du tableau en fonction de la valeur  $x_{Bi}$  de la variable de base. Formellement la coupe de Gomory est générée comme suit :

$$x_{Bi} + \sum_{j \in J} a_{ij} x_j = b_i \tag{2.8}$$

Où  $J = j : j$  indice de variable hors base  $j = 1, \dots, n$

Notons par  $\lfloor a \rfloor$  la partie entière inférieure à  $a$ . Dans ce cas,

$$x_{Bi} + \sum_{j \in J} \lfloor a_{ij} \rfloor x_j + \sum_{j \in J} f(a_{ij}) x_j = \lfloor b_i \rfloor + f(b_i) \tag{2.9}$$

Où  $f(a) = a - \lfloor a \rfloor$ .

L'équation (2.9) est égale à :

$$x_{Bi} + \sum_{j \in J} \lfloor a_{ij} \rfloor x_j = \lfloor b_i \rfloor + f(b_i) - \sum_{j \in J} f(a_{ij}) x_j \tag{2.10}$$

Nous pouvons écrire (2.10) sous forme d'une inéquation :

$$x_{Bi} + \sum_{j \in J} \lfloor a_{ij} \rfloor x_j \leq \lfloor b_i \rfloor + f(b_i) \tag{2.11}$$

Cette inégalité satisfait aussi :

$$x_{Bi} + \sum_{j \in J} \lfloor a_{ij} \rfloor x_j \leq \lfloor b_i \rfloor \tag{2.12}$$

De (2.12) nous avons :

$$x_{Bi} = b_i - \sum_{j \in J} a_{ij}x_j \quad (2.13)$$

Nous remplaçons (2.13) dans (2.12) nous aurons l'expression la coupe de Gomory

$$\sum_{j \in J} f(a_{ij})x_j \geq f(b_i) \quad (2.14)$$

**Exemple 2.4**

Soit le problème linéaire en nombre entier suivant :

$$(PLNE) \begin{cases} \max & z(x) = x_1 + 2x_2 \\ & 2x_1 + x_2 \leq 3 \\ & x_2 \leq 2 \\ & x_1 \in \mathbb{N}, \quad x_2 \in \mathbb{N} \end{cases}$$

La forme standard et relaxé de(PLNE)

$$(P_r) \begin{cases} \max & z(x) = x_1 + 2x_2 \\ & 2x_1 + x_2 + x_3 = 3 \\ & x_2 + x_4 = 2 \\ & x_j \geq 0, \quad j = 1, \dots, 4 \end{cases}$$

On a

$$J = \{1, 2, 3, 4\}, \quad J_B = \{3, 4\}, \quad J_H = \{1, 2\}$$

Donc la solution de base réalisable est  $x = (0, 0, 3, 2)$

Dressons le premier tableau du simplexe du  $(P_r)$  :

$C_B$	Base	$B$	$x_1$	$x_2$	$x_3$	$x_4$	$\theta_j$
0	$x_3$	3	2	1	1	0	3
0	$x_4$	2	0	1	0	1	2
$z = 0$	/	$\Delta$	-1	-2	0	0	/

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

$$\min(\Delta_j/\Delta_j < 0, j \in J_H) = \Delta_2 = -2$$

Donc  $j_0 = 2$  , de là la variable  $x_2$  va rentrer dans la nouvelle base, et calculons :

$\theta^0 = \min_{j \in J_B}(\theta_j)$ , d'où  $\theta^0 = 2$  , de là la variable  $x_4$  va sortir de la base , et la nouvelle base est  $J_B = \{2, 3\}$

Dressons le deuxième tableau du simplexe pour déterminer la nouvelle solution :

$C_B$	Base	B	$x_1$	$x_2$	$x_3$	$x_4$	$\theta_j$
0	$x_3$	1	2	0	1	-1	$\frac{1}{2}$
2	$x_2$	2	0	1	0	1	/
$z = 4$	/	$\Delta$	-1	0	0	2	/

La nouvelle solution de base est  $x = (0, 2, 1, 0)$ ; qui n'est pas optimale, car  $\Delta_1 = -1 < 0$

Dressons le troisième tableau du simplexe :

$C_B$	Base	B	$x_1$	$x_2$	$x_3$	$x_4$	$\theta_j$
1	$x_1$	$\frac{1}{2}$	1	0	$\frac{1}{2}$	$-\frac{1}{2}$	/
2	$x_2$	2	0	1	0	0	/
$z = 9/2$	/	$\Delta$	0	0	0	1/2	/

La solution de la troisième itération est optimale pour le  $(P_r)$  mais pas optimale pour (PLNE).

Pour cela on exhibe la coupe de Gomory tel que :  $f_i \leq \sum_{j \in J_H} f_{ij}x_j$

$$\frac{1}{2} \leq \frac{1}{2}x_3 - \frac{1}{2}x_4$$

En utilisons le dual du simplexe on résout le dernier  $(P'_r)$  :

1ère itération :

$C_B$	Base	B	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$
1	$x_1$	1/2	1	0	1/2	-1/2	0
2	$x_2$	2	0	1	0	1	0
0	$x_5$	-1/2	0	0	-1/2	1/2	1
$z = 9/2$	/	$\Delta$	0	0	1/2	3/2	0
/	/	$\sigma$	/	/	1	3	/

Le critère d'optimalité n'est pas vérifié car  $\exists x_j < 0, j \in J_B$  on doit faire sortir  $x_5$  de la base car  $\min x_j = x_5 < 0$ . Et donc :  $\sigma_j = -\frac{\Delta_j}{x_{5j}}$  et  $\min \sigma_j = \sigma_3$ .

donc on doit faire rentrer  $x_3$  dans la base. 2ème itération :

$C_B$	Base	B	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$
1	$x_1$	0	1	0	0	-1	1
2	$x_2$	2	0	1	0	1	0
0	$x_3$	1	0	0	1	-1	-2
$z = 4$	/	$\Delta$	0	0	0	1	1

Le critère d'optimalité est vérifié, donc  $x = (0, 2, 1, 0, 0)$  est la solution optimale du problème  $(P'_r)$  avec  $Z^0 = 4$  et aussi pour le (PLNE).

### Méthode de Branch and Bound [26]

- Les méthodes de branch-and-bound sont des méthodes basées sur une énumération "intelligente" des solutions admissibles d'un problème d'optimisation combinatoire.
- Idée : prouver l'optimalité d'une solution en partitionnant l'espace des solutions.
- Exemple d'application à la programmation linéaire et non linéaire en nombres entiers utilise toute la puissance de la programmation linéaire ou non linéaire pour déterminer des bonnes bornes.

- On appelle relaxation linéaire d'un programme linéaire en nombres entiers le programme linéaire obtenu en supprimant les contraintes d'intégrité sur les variables.

**Programme en nombre entiers :**

$$(P) = \begin{cases} \max c'x \\ Ax \leq b \\ x \geq 0, x \in \mathbb{N}^n \end{cases}$$

**Relaxation linéaire :**

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

**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.

**Branchement**

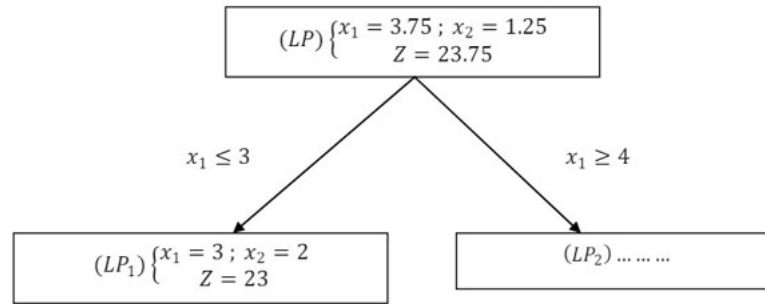
- Si la solution de (LP) n'est pas entière, soit  $x_i$  une variable prenant une valeur fractionnaire  $x_i^*$  dans la solution optimale de (LP)
- Le problème peut être divisé en deux sous-problèmes en imposant  $x_i \leq \lfloor x_i^* \rfloor$  ou  $x_i \geq \lfloor x_i^* \rfloor + 1$  Où  $\lfloor x_i^* \rfloor$  est le plus grand entier inférieur à  $x_i^*$
- La solution optimal de LP est la meilleure des solutions optimales des deux sous

$$\text{problèmes}(P_1; P_2) : (P_1) = \begin{cases} \max c'x \\ Ax \leq b \\ x_i \leq \lfloor x_i^* \rfloor \\ x \geq 0. \end{cases} \quad (P_2) = \begin{cases} \max c'x \\ Ax \leq b \\ x_i \geq \lfloor x_i^* \rfloor + 1 \\ x \geq 0. \end{cases}$$

**Exemple 2.5 [26]** Soit à résoudre le programme suivant

$$(P) = \begin{cases} \max Z = 5x_1 + 4x_2 \\ x_1 + x_2 \leq 5, \\ 10x_1 + 6x_2 \leq 45, \\ x_1, x_2 \geq 0, \\ \text{et } x_1, x_2 \text{ sont des entiers.} \end{cases}$$

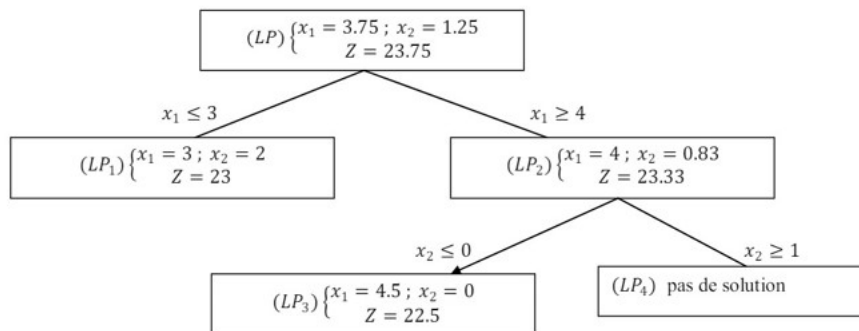
$$(LP) = \begin{cases} x_1 = 3.75, x_2 = 1.25, \\ z = 23.75, \end{cases}$$



La solution de (LP<sub>1</sub>) est une solution admissible de  $p$  et donc  $Z = 23$  est une borne inférieure sur la valeur de la solution optimale de  $P$ . Le noeud correspondant peut être éliminé vu qu'une solution entière optimale satisfaisant  $x_1 \leq 3$  a été trouvée (solution de (LP<sub>1</sub>)). La valeur de la solution de LP,  $Z = 23.75$  est une borne supérieure sur la valeur de la solution optimale de  $P$ . Vu que tous les coefficients sont entiers, on peut en déduire que la valeur de la solution optimale de  $P$  est inférieure ou égale à 23. La solution de (LP<sub>1</sub>) est donc optimale pour  $P$ .

### Règles de branchement

- Il n'y a pas de règle générale pour le choix de la variable de branchement et de la branche à examiner en premier.
- Ce choix peut avoir un impact important sur le nombre de noeuds à examiner dans l'arbre de branch-and-bound.



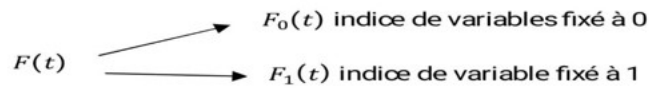
### Méthode de BALAS [27]

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 dimension (coté logiciel). Cette méthode peut être classée dans la classe des problèmes NP-difficile.

### Nœuds de séparation (branching)

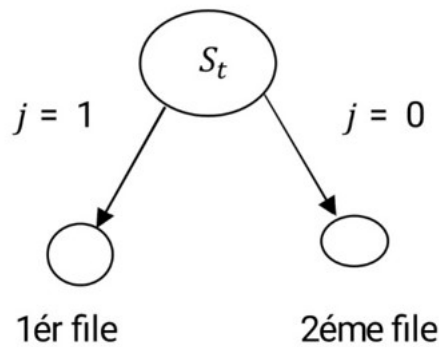
Chaque noeud de (sommet)  $S_{(t)}$  correspond à une solution partielle. Ce noeud correspond à deux sous-ensembles d'indices pour les variables.

Soient :  $F(t)$  Les sous-ensembles d'indices fixés à 0 ou à 1 pour les variables au niveau du noeud  $S_{(t)}$



$L(t)$  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 2.2 :**

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)  $S_t$

$$Z_t = \sum_{j \in L(t)} c_j x_j + \sum_{j \in F_1(t)} c_j$$

$$\sum_{j \in L(t)} a_{ij} x_j \leq b_i - \sum_{j \in F_1(t)} a_{ij} = S_i \quad i \in \{1, 2, \dots, m\}$$

Nous appelons l'évaluation  $S_t = \sum_{j \in F_1(t)} c_j$  valeurs meilleur pour ce nœud.  $S_t$  est un noeud 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 ( $S_t$ ) est supérieur ou égale à  $Z_{\text{trouvé}}$  valeur de la solution provisoire trouvée.
2.  $x$  solution obtenu en posant  $x_j = 0$  et  $j \in L(t)$  est solution réalisable pour  $pl_{(0,1)}$  comme avec  $\forall i \in \{1, \dots, m\}, S_i \geq 0$  alors le nœud est terminal en 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 à choisir)  
Balas a donné une technique pour minimiser le nombre de branching à effectuer et cela comme suivant :

Soit deux sous ensemble  $Q(t)$  et  $R(t)$  définis comme suit :

$$Q(t) = \{i \mid S_i < 0\}, \quad i = 1, \dots, m$$

$$R(t) = \{j \in L(t) \mid \exists i \in Q(t), a_{ij} < 0\}$$

### Mesure de proximité

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

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

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

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

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  $j^* \in R(t)$

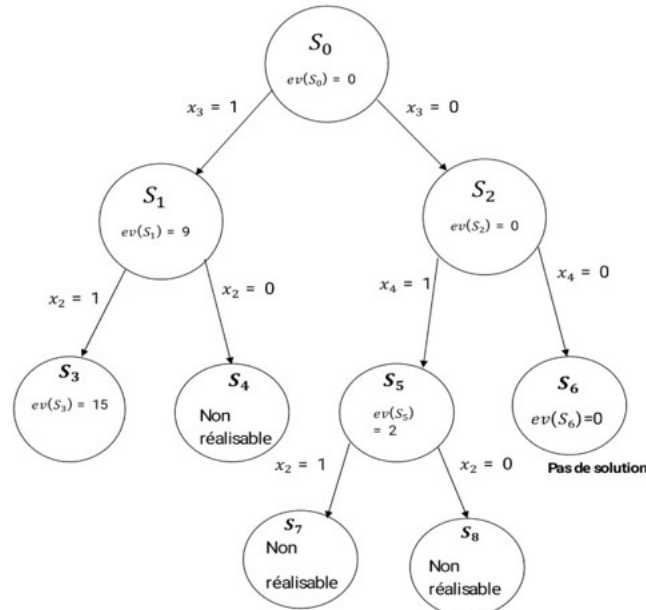
$$P(t, j^*) = \max_{j \in R(t)} P(t, j)$$

### Conclusion :

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

### Exemple 2.6

$$\left\{ \begin{array}{l} 4x_1 + 6x_2 + 9x_3 + 2x_4 + x_5 \rightarrow \min \\ -x_1 + 2x_2 - 4x_3 - x_4 + 3x_5 \leq -2 \\ x_1 - 2x_2 + x_4 + x_5 \leq -1 \\ 2x_1 - 5x_2 + 2x_3 + x_4 - 2x_5 \leq 1 \\ x_j \in \{0, 1\}, \quad j = 1, \dots, 5 \end{array} \right.$$



**Conclusion :** avec la méthode de Balas on a trouvé  
 $x_1^* = 0, x_2^* = 1, x_3^* = 1, x_4^* = 0, x_5^* = 0$   
 Et la solution optimal  $z(x^*) = 15$ .

## 2.2 Programmation non linéaire :

### 2.2.1 Définition d'un Programme non linéaire [6] [7]

La programmation non linéaire est une technique d'optimisation mathématique utilisée pour trouver la solution (les solutions) optimale globale (locale) pour un problème donné, dont la fonction objective et /ou les contraintes sont linéaire ou pas et à plusieurs variables.

**Par exemple :** a programmation non linéaire est la recherche d'un optimum d'une fonction objective non linéaire sur un sous ensemble convexe ou non convexe.

Le problème non linéaire s'écrit sous la forme, entre autres :

$$\begin{cases} \min \max & f(x) \\ h(x) & = 0 \\ g(x) & \leq 0 \\ x & \in \mathbb{R}^n \end{cases} \quad (2.15)$$

$f : \mathbb{R}^n \rightarrow \mathbb{R}, h : \mathbb{R}^n \rightarrow \mathbb{R}^p, g : \mathbb{R}^n \rightarrow \mathbb{R}^m, \text{ et } x \in \mathbb{R}^n.$

**Définition 2.4 : (fonction non linéaire) [21]**

Toute fonction qui n'est pas affine est dite non linéaire.

**Définition 2.5 : (fonction affine) [21]**

Une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite affine si elle s'écrit  $f(x) = c'x + d$ , où  $c \in \mathbb{R}^n$  est un vecteur de constantes et  $d \in \mathbb{R}$ . Une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  est affine si chacune de ses composantes  $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$   $i = 1, \dots, m$  est affine. Dans ce cas, elle peut s'écrire  $f(x) = Ax + b$ , où  $A \in \mathbb{R}^{m \times n}$  est une matrice et  $b \in \mathbb{R}^m$

**Définition 2.6 : (dérivée partielle) [26]**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continue. La fonction notée  $\nabla_i f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ , également notée  $\frac{\partial f(x)}{\partial x_i}$  est appelée  $i$  ème dérivée partielle de  $f$  est définie par :

$$\lim_{\alpha \rightarrow 0} \frac{f(x_1, \dots, x_i + \alpha, \dots, x_n) - f(x_1, \dots, x_i, \dots, x_n)}{\alpha}.$$

Cette limite peut ne pas exister.

**Définition 2.7 : (Gradient) [21]**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction différentiable. La fonction notée :  $\nabla f : \mathbb{R}^n \rightarrow \mathbb{R}^n$  est appelée le gradient de  $f$  et est définie par :

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{pmatrix}.$$

**Exemple :** Soit  $f(x_1, x_2, x_3) = e^{x_1} + x_1^2 x_3 - x_1 x_2 x_3$ . Le gradient de  $f$  est donné par :

$$\nabla f(x_1, x_2, x_3) = \begin{pmatrix} e^{x_1} + 2x_1 x_3 - x_2 x_3 \\ -x_1 x_3 \\ x_1^2 - x_1 x_2 \end{pmatrix}.$$

**Définition 2.8 : (Dérivée directionnelle) [21]**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continue. Soient  $x \in \mathbb{R}^n$  et  $d \in \mathbb{R}^n$ . La dérivée directionnelle de  $f$  en  $x$  dans la direction  $d$  est donnée par :  $\lim_{\alpha \rightarrow 0} \frac{f(x + \alpha d) - f(x)}{\alpha}$ , si la limite existe. De plus, lorsque le gradient existe, la dérivée directionnelle est le produit scalaire entre le gradient de  $f$  et la direction  $d$ , i.e.  $\nabla f(x)'d$ .

**Exemple :** Soit  $f(x_1, x_2, x_3) = e^{x_1} + x_1^2 x_3 - x_1 x_2 x_3$ , et soit  $d = (d_1, d_2, d_3)$ .

La dérivée directionnelle de  $f$  dans la direction  $d$  est :

$$(d_1, d_2, d_3) \cdot \nabla f(x_1, x_2, x_3) = d_1(e^{x_1} + 2x_1 x_3 - x_2 x_3) - d_2 x_1 x_3 + d_3(x_1^2 - x_1 x_2).$$

**Définition 2.9 : (fonction différentiable) [21]**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  une fonction continue. Si, pour tout  $d \in \mathbb{R}^n$ , la dérivée directionnelle de  $f$  dans la direction  $d$  existe, alors la fonction  $f$  est dite différentiable.

**Remarque 2.1 :** La dérivée partielle = dérivée directionnelle dans la direction des axes de coordonnées :  $\frac{\partial f(x)}{\partial x_i} = (\nabla f(x))'e_i$

Avec :  $e_i = (0, \dots, 1, \dots, 0)'$  : le  $i$ ème vecteur de la base canonique de  $\mathbb{R}^n$ .

**Définition 2.10 : (matrice Jacobienne) [21]**

soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ , la fonction  $J(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$  définie par :

$$J(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x) & \cdots & \frac{\partial f_1}{\partial x_n}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1}(x) & \cdots & \frac{\partial f_m}{\partial x_n}(x) \end{bmatrix} \text{ est appelée matrice Jacobienne}$$

**Définition 2.11 : (matrice Hessienne)**

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  deux fois différentiable ; la fonction notée  $\nabla^2 f(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$

définie par :  $H(x) = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \dots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}$  est appelée matrice hessienne

**Exemple 2.2 :** Soit  $f(x_1, x_2, x_3) = e^{x_1} + x_1^2 x_3 - x_1 x_2 x_3$

Le gradient de  $f$  est donné par :  $\nabla f(x_1, x_2, x_3) = \begin{pmatrix} e^{x_1} + 2x_1 x_3 - x_2 x_3 \\ -2x_1 x_3 \\ x_1^2 - x_1 x_2 \end{pmatrix}$

La Hessienne de  $f$  est donnée par :

$H(x) = \nabla^2 f(x_1, x_2, x_3) = \begin{pmatrix} e^{x_1} + 2x_3 & -x_3 & 2x_1 \\ -2x_3 & 0 & -2x_1 \\ 2x_1 - x_2 & -x_1 & 0 \end{pmatrix}$

**Remarque 2.2 :**

La matrice Hessienne est toujours symétrique.

**Définition 2.12 :**

Soit  $A$  une matrice carrée ( $n \times n$ ) symétrique,  $A$  est dite :

- Semi définie positive (respectivement semi définie négative) si  $x'Ax \geq 0$  ( $x'Ax \leq 0$ ) avec  $x \in \mathbb{R}^n$
- définie positive (respectivement définie négative)  $x'Ax > 0$  ( $x'Ax < 0$ ) avec  $x \in \mathbb{R}^n$   $x \neq 0$

**Définition 2.13 : (fonction convexe) [21]**

Une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite convexe si pour tout  $x, y \in \mathbb{R}^n$  et pour tout  $\lambda \in [0, 1]$

On a :  $f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$

**Définition 2.14 : (fonction concave) [21]** Une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite concave si -  $f$  est une fonction convexe, c'est-à-dire si pour tout  $x, y \in \mathbb{R}^n$  et pour tout  $\lambda \in [0, 1]$  On a :  $f(\lambda x + (1 - \lambda)y) \geq \lambda f(x) + (1 - \lambda)f(y)$

**Remarque 2.3 :**

- Si Hessienne  $f$  est définie positive alors  $f$  est strictement convexe.
- Si Hessienne  $f$  est définie négative alors  $f$  est strictement concave.
- Si en utilisant les valeurs propres : Soit  $\lambda_i$  les valeurs propres de  $A$  sont telles que :
  - $\lambda_i > 0$  alors  $A$  est définie positive.
  - $\lambda_i \geq 0$  alors  $A$  est semi définie positive
  - $\lambda_i < 0$  alors  $A$  est définie négative.
  - $\lambda_i \leq 0$  alors  $A$  est semi définie négative

**Définition 2.15 :**

Un programme mathématique est un problème d'optimisation dans  $\mathbb{R}^n$  s'écrit sous la forme suivante :

$$\begin{cases} \min / \max f(x) \\ x \in D \end{cases} \quad (2.16)$$

$$D = \{x \in \mathbb{R}^n : g_i(x) \leq 0, \quad i = 1, \dots, m, \quad h_j(x) = 0 \quad \text{Où, } j = 1, \dots, p\}$$

avec

$$g_i, h_j : \mathbb{R}^n \rightarrow \mathbb{R} \text{ et } f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}$$

On appelle  $f$  la fonction objectif et  $D$  l'ensemble des solution réalisables (admissibles)  
Pour le problème (2.16) ci-dessus  $x^* \in D$  est un :

— Minimum local si  $f(x^*) \leq f(x) \quad \forall x \in V(x)$

— Maximum local si  $f(x^*) \geq f(x) \quad \forall x \in V(x)$

Tel que  $V(x)$  le voisinage de  $x$ .

— Minimum global si  $f(x^*) \leq f(x) \quad \forall x \in D$

— Maximum local si  $f(x^*) \geq f(x) \quad \forall x \in D$

## 2.2.2 Optimisation sans contraintes [6] [7]

Un problème d'optimisation sans contraintes s'écrit sous la forme suivante :

$$\begin{cases} \min f(x) \\ x \in \mathbb{R}^n \end{cases} \quad (2.17)$$

Où  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est une fonction continue différentiable.

### Les conditions nécessaires du 1er ordre (cas minimum)

soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  différentiable,  $x^*$  est un minimum local ou global alors  $\nabla f(x^*) = 0$

**Les conditions suffisantes du 1er ordre (cas minimum)(cas convexe)** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  différentiable et convexe, alors  $x^*$  est un minimum local (ou global)  $\nabla f(x^*) = 0$

### Les conditions nécessaires du 2ème ordre (cas minimum)

**Théorème** soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  de classe  $C^2$ , est un minimum local (ou global) alors :

—  $\nabla f(x^*) = 0$

—  $Hf(x^*) = 0$  est un semi définie positif (SDP)

### Les conditions suffisantes du 2ème ordre (cas minimum)

#### Théorème

soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  de classe  $C^2$  si

—  $\nabla f(x^*) = 0$

—  $Hf(x^*)$  est définie positif (DP)

Alors  $x^*$  est un minimum local (ou global) strict.

### 2.2.3 Optimisation avec contraintes [6] [7]

#### Définition 2.16

On définit le problème d'optimisation avec contraintes sous la forme suivante :

$$\begin{cases} \min f(x) \\ g_i(x) \leq 0, i = 1, \dots, m \\ h_j(x) = 0, j = 1, \dots, p \\ x \in \mathbb{R}^n \end{cases} \quad (2.18)$$

où  $g_i, h_j : \mathbb{R}^n \rightarrow \mathbb{R}$  et  $f : \mathbb{R}^n \rightarrow \mathbb{R}$

#### Définition 2.17

Les contraintes sont dites actives au point  $x^0$  si  $\nabla g_i(x^0) = 0$

#### Qualification des contraintes (QC) [7]

Pour que la qualification des contraintes soit vérifiée en tout point  $x \in D$  il suffit que l'une des conditions (1) ou (2) soit réalisée :

1. Toutes les fonctions  $g_i, i = 1, \dots, m$  et  $h_j, j = 1, \dots, p$  sont linéaires ou affines (Karlin).
2. Toutes les fonctions  $g_i, i = 1, \dots, m$  sont convexe et les  $h_j, j = 1, \dots, p$  sont linéaires et il existe  $\bar{x} \in \mathbb{R}^n$  vérifiant  $g_i(\bar{x}) < 0, i = 1, \dots, m$   $h_j(\bar{x}) = 0 (j = 1, \dots, p)$  (Slater)  
Pour que (QC) soit vérifiée en un point  $x^0$ , il suffit que l'on ait :
3. Les gradients des contraintes actives  $\nabla g_i(x^0), i \in \{1, \dots, m\}$  et  $\nabla h_j(x^0), j \in \{1, \dots, p\}$  sont linéairement indépendant (Fiacco-McCormick) .

#### Définition 2.18 :

Le Lagrangien du programme mathématique (2.18) est défini par :

$$L(x, \lambda, \mu) = f(x) + \sum_{i=1}^m \lambda_i g_i(x) + \sum_{j=1}^p \mu_j h_j(x) \text{ avec } \lambda_i \geq 0 \text{ et } \mu_j \in \mathbb{R}$$

### Les conditions nécessaires de Karush-Kuhn- Tucker (CN du 1er ordre) [6] [7]

#### Théorème

On suppose que les fonctions  $f, g_i, i = 1, \dots, m$  et  $h_j, j = 1, \dots, p$  sont continument différentiables, et que l'hypothèse de (QC) est vérifiée en  $x^0 \in D$  avec :

$$D = \{x \in \mathbb{R}^n : g_i(x) \leq 0, i = 1, \dots, m, h_j(x) = 0, j = 1, \dots, p\}$$

Alors la condition nécessaire pour que  $x^*$  soit un optimum local (global) de (2.18) et qu'il existe des multiplicateurs  $\lambda^* \in \mathbb{R}_+^m$  et  $\mu^* \in \mathbb{R}^p$  tel que :

$$\begin{cases} \nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) + \sum_{j=1}^p \mu_j^* \nabla h_j(x^*) = 0 \\ \text{avec} \\ \nabla_x L(x^*, \lambda^*, \mu^*) = 0 \quad (\text{condition d'optimalité}) \\ \lambda_i^* g_i(x^*) = 0, \quad i = 1, \dots, m \quad (\text{condition de complémentarité}) \\ \lambda_i^* \geq 0, \quad i = 1, \dots, m \\ g_i(x^*) \leq 0, \quad i = 1, \dots, m \\ g_i(x^*) \leq 0, \quad i = 1, \dots, p \end{cases}$$

**Les conditions nécessaires du 2ème ordre****Théorème**

Soient les fonctions  $f, g_i, i = 1, \dots, m$  et  $h_j, j = 1, \dots, p$  de classe  $C^2$ ,  $x^*$  un minimum local qualifié alors il existe des multiplicateurs  $\lambda^* \in \mathbb{R}_+^m$  et  $\mu^* \in \mathbb{R}^p$  tel que :

- $x^*$  Vérifie les CN de KTT
- $H_x L(x^*, \lambda^*, \mu^*)$  est définie positive sur l'espace tangent  $T(x^*)$  au point  $x^*$  du problème (2.18) qui est définie par :

$$T(x^*) = \{y \in \mathbb{R}^n : \nabla g_i(x^*) \cdot y \leq 0, i = 1, \dots, m \text{ et } \nabla h_j(x^*) \cdot y = 0, j = 1, \dots, p\}$$

Alors  $x^*$  est un minimum local strict.

**2.2.4 Cas particulier de la programmation mathématique****Résolution d'un problème quadratique convexe**

La programmation quadratique est une branche de l'optimisation non linéaire, où la fonction objectif à minimiser est une fonction quadratique et les contraintes définissant le domaine des solutions réalisables sont linéaires (et,ou) quadratiques.

Une version du problème quadratique peut s'écrire comme suit :

$$\begin{cases} \min f(x) = \frac{1}{2}x'Dx + c'x + g \\ Ax = b \\ Qx \leq h \\ x \in \mathbb{R}^n \end{cases} \quad (2.19)$$

Où  $D$  est une matrice symétrique d'ordre  $n$ ,  $c$  et  $x$  sont des  $n$ -vecteurs,  $A$  est une  $(p, n)$  matrice avec  $\text{rang}(A) = p$ ,  $Q$  est une  $(m, n)$  matrice,  $\text{rang}(A) = m$ ,  $b$  est un  $p$ -vecteur et  $h$  est un  $m$ -vecteur, et  $g \in \mathbb{R}$ .

**Méthode de Wolfe (1959) [28]**

De multiples approches existent pour résoudre les problèmes de programmation quadratique. Parmi celles-ci, on s'intéresse à la méthode la plus classique de Wolfe qui n'est que la méthode du simplexe légèrement modifiée. Le principe de cette méthode est la résolution du système de Kuhn-Tucker et consiste à trouver une solution réalisable pour un système linéaire avec une condition supplémentaire de type  $x_j \delta_j$  Où  $x$  et  $\delta$  sont des vecteurs de même dimension.

**Algorithme de Wolf [28]**

Début

1. Introduire les données  $D, A, b, c$   
 Appliquer les conditions de KKT au problème  
 Déterminer les équations de KKT  
 Détermination des paramètres du programme linéaire.
2. Déterminations des paramètres du programme linéaire  
 Introduire les variables artificielles  
 Construire la matrice des contraintes  
 Construire le vecteur des coûts.

3. Initialiser le vecteur de solution  $(x, \lambda, \delta, v)$   
 Déterminer l'ensemble des indices  $J_B$  et  $J_H$   
 Extraire les éléments de base  $x_B, c_B, A_B$
4. Calculer le vecteur des potentiels  $y' = c'_B A_B^{-1}$   
 Calculer le vecteur des estimations  $\Delta'_H = y' A_H - c'_H$   
 Si  $\delta_H \geq 0$  alors la solution actuelle est optimale  
 Fin si  
 Sinon aller à 5
5. Déterminer la variable qui entre en base tout en vérifiant la condition  $\lambda_j x_j = 0, j = (1, n)$   
 Déterminer la variable qui sort de la base  
 Mettre à jour  $A_B, x_B, c_B, J_B, J_H$  et aller en 4  
 Fin.

**Exemple 2.3**

Soit à résoudre le problème non linéaire suivant :

$$(P) = \begin{cases} \min f(x) = 2x_1^2 + x_1x_2 + 12x_2^2 + x_1 + 4x_2 \\ x_1 + x_2 \leq 3 \\ 2x_1 + x_2 \leq 2 \\ x_1 \geq 0, x_2 \geq 0 \end{cases} \quad (2.20)$$

Le problème sous la forme standard

$$(P) = \begin{cases} \min f(x) = 2x_1^2 + x_1x_2 + 12x_2^2 + x_1 + 4x_2 \\ x_1 + x_2 + x_3 = 3 \\ 2x_1 + x_2 + x_4 = 2 \\ x_j \geq 0, j = 1, 2, 3, 4 \end{cases} \quad (2.21)$$

Calculons  $\nabla f(x) : \nabla f(x) = \begin{pmatrix} 4x_1 + x_2 + 1 \\ x_1 + 24x_2 + 4 \\ 0 \\ 0 \end{pmatrix}$

Calculons  $H : H(x) = \nabla^2 f(x) = \begin{pmatrix} 4 & 1 & 0 & 0 \\ 1 & 24 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

La matrice  $H(x) = \nabla^2 f(x)$  est semi définie positive.

Calculons le Lagrangien  $L$  pour le problème (2.21),  $\lambda_1, \lambda_2 \in \mathbb{R}, \delta_j \geq 0, j = 1, \dots, 4$

$$L(x_1, x_2, x_3, x_4, \lambda_1, \lambda_2, \delta_1, \delta_2, \delta_3, \delta_4) = 2x_1^2 + x_1x_2 + 12x_2^2 + x_1 + 4x_2 + \lambda_1(x_1 - x_2 + x_3 - 3) + \lambda_2(2x_1 + x_2 + x_4 - 2) - \delta_1x_1 - \delta_2x_2 - \delta_3x_3 - \delta_4x_4$$

On applique le théorème de KKT

$$\nabla L_x = 0, \delta_j x_j = 0, \delta_j \geq 0, x_j \geq 0, j = 1, \dots, 4$$

On obtient

$$\begin{cases} 4x_1 + x_2 + 1 + \lambda_1 + 2\lambda_2 - \delta_1 = 0 \\ x_1 + 24x_2 + 4 + \lambda_1 + \lambda_2 - \delta_2 = 0 \\ \lambda_1 - \delta_3 = 0 \\ \lambda_2 - \delta_4 = 0 \end{cases}$$

En remplaçant on obtient :

$$\begin{cases} 4x_1 + x_2 + 1 + \delta_3 + 2\delta_4 - \delta_1 = 0 \\ x_1 + 24x_2 + 4 + \delta_3 + \delta_4 - \delta_2 = 0 \end{cases}$$

On les multiplie par -1 on obtient

$$\begin{cases} -4x_1 - x_2 + \delta_1 - \delta_3 - 2\delta_4 = 1 \\ -x_1 - 24x_2 + \delta_2 - \delta_3 - \delta_4 = 4 \end{cases}$$

La solution réalisable basique du système d'optimalité du problème quadratique est le vecteur  $(x, \delta) : (x_1, x_2) = (0, 0), (\delta_1, \delta_2) = (1, 4)$ .

## 2.3 Programmation flou

### 2.3.1 Définition d'un sous ensemble flou :

En théorie des ensembles, un ensemble flou, introduit par Lotfi Zadeh en 1965, est une extension des ensembles classiques qui permet de modéliser des concepts imprécis en attribuant des degrés d'appartenance à leurs éléments. Contrairement aux ensembles traditionnels où un élément est soit membre, soit non-membre de l'ensemble, les ensembles flous permettent de définir des éléments avec un degré d'appartenance compris entre 0 et 1. Ainsi, un ensemble flou peut contenir des éléments de manière partielle, reflétant ainsi l'imprécision et l'incertitude inhérentes à de nombreux domaines.

#### Remarque 2.4 :

On utilise souvent le terme d'ensemble flou au lieu de sous-ensemble flou ou vice versa, par abus de langage [10] .

#### Définition 2.19

Soit  $X$  un référentiel, on définit un ensemble flou  $\tilde{A}$  dans  $X$  par la donnée d'une fonction :

$\mu_{\tilde{A}} : X \rightarrow [0, 1]$  qui associe à chaque élément  $x \in X$ , une valeur  $\mu_{\tilde{A}(x)}$  désignant le degré avec lequel  $x$  appartient à  $A$ .

Cette fonction est appelée "fonction d'appartenance" de l'ensemble flou  $\tilde{A}$ .

$\tilde{A}$  sera noté par  $\tilde{A} = (X, \mu_{\tilde{A}})$  ou  $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in X\}$  avec :

$$\mu_{\tilde{A}}(X) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \\ ]0, 1[ & \text{pour les cas intermédiaires} \end{cases}$$

### 2.3.2 Caractéristiques d'un sous-ensemble flou [8] [9] [10] :

#### Sous-ensembles de niveau $\alpha$ [8]

Un sous ensemble de niveau  $\alpha$  noté  $\tilde{A}^\alpha$  est l'ensemble  $\tilde{A}^\alpha = \{x \in X / \mu_{\tilde{A}}(x) \geq \alpha\}$  Il contient tous les éléments de  $X$  qui sont compatibles avec  $\tilde{A}$  à un niveau au moins égale à  $\alpha$ .

#### Sous-ensembles de niveau $\alpha$ strict [8]

Un sous ensemble de niveau  $\alpha$  strict noté  $\tilde{A}^{\bar{\alpha}}$  est l'ensemble :

$$\tilde{A}^{\bar{\alpha}} = \{x \in X / \mu_{\tilde{A}}(x) > \alpha\}$$

#### Support de $\tilde{A}$ [8]

Le support de  $\tilde{A}$  noté  $\text{sup } \tilde{A}$  est la partie de  $X$  où la fonction d'appartenance de  $\tilde{A}$  n'est pas nulle c'est-à-dire :

$$\text{sup } \tilde{A} = \{x \in X / \mu_{\tilde{A}}(x) > 0\}$$

#### Hauteur de $\tilde{A}$ :

La hauteur de  $\tilde{A}$  notée  $\text{Haut } \tilde{A}$  est définie comme suit :

$$\text{Haut } \tilde{A} = \sup \mu_{\tilde{A}}(x); \quad x \in X$$

#### Sous-ensemble flou normalisé :

Un sous-ensemble flou  $\tilde{A}$  de  $X$  est dite normalisé si sa hauteur est telle que :

$$\text{Haut } \tilde{A} = 1$$

#### Noyau d'un ensemble flou $\tilde{A}$ [24] :

Le noyau de  $\tilde{A}$  noté  $\text{Noy}(\tilde{A})$  est un ensemble d'élément de  $X$  où la fonction d'appartenance de  $\tilde{A}$  vaut 1 c'est-à-dire :

$$\text{Noy}(\tilde{A}) = \{x \in X / \mu_{\tilde{A}}(x) = 1\}$$

#### Sous-ensemble flou convexe :

$\tilde{A}$  est dit convexe si quelque soient  $x_1$  et  $x_2$  appartenant à  $X$  et  $\lambda \in [0, 1]$  on a :

$$\mu_{\tilde{A}}(\lambda x_1 + (1 - \lambda)x_2) \geq \min[\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)]$$

#### La cardinalité du sous-ensemble $\tilde{A}$ :

La cardinalité du sous-ensemble  $\tilde{A}$  est définie par :

$$|\tilde{A}| = \sum_{x \in X} \mu_{\tilde{A}}(x)$$

#### Exemple 2.4 :

$$X = \{x_1, x_2, x_3, x_4, x_5\}$$

$$\tilde{A} = \{(x_1, 0.9), (x_2, 1), (x_3, 0.5), (x_4, 0.1), (x_5, 0.3)\}$$

$$\text{sup } \tilde{A} = \{x_1, x_2, x_3, x_4, x_5\}$$

$$\text{Haut } \tilde{A} = \text{Supp } \mu_{\tilde{A}}(x) = 1$$

$\tilde{A}$  est normalisé car  $\text{Haut } \tilde{A} = 1$

$$\tilde{A}^{0.5} = \{x_1, x_2, x_3\}$$

$$\tilde{A}^{\bar{0}.5} = \{x_1, x_2\}$$

$$|\tilde{A}| = \sum_{x \in X} \mu_{\tilde{A}}(x) = 2,8$$

**Définition Nombre flou [8] [9] :**

Un nombre flou est un ensemble flou  $\tilde{A}$  convexe et normalisé, de l'ensemble des nombres réels  $X = \mathbb{R}$ .

**Remarque 2.5 :**

Si le  $\text{Noy}(\tilde{A})$  est un intervalle de  $\mathbb{R}$ , on parle alors d'intervalle flou.

**Exemple 2.5 :**

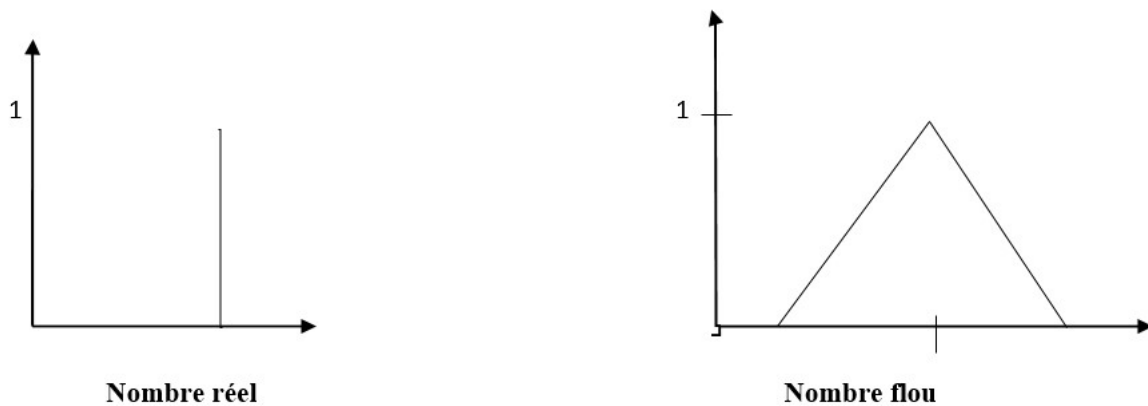


FIGURE 2.2 – Comparaison entre un nombre flou et un nombre réel

### 2.3.3 Opération sur les ensembles flous [11] [12]

Les opérations sur les ensembles flous sont généralement des extensions des opérations connues sur les ensembles classiques (inclusion, égalité, réunion, intersection et complément). Elles s'appliquent d'ailleurs aux ensembles classiques lorsque les fonctions d'appartenance se réduisent à des valeurs de 0 ou 1.

**Inclusion** Un ensemble flou  $\tilde{A}$  est inclu dans un autre ensemble flou  $\tilde{B}$ , cela se traduit par :

$$\tilde{A} \subseteq \tilde{B} \iff \mu_{\tilde{A}}(x) \leq \mu_{\tilde{B}}(x), \forall x \in X$$

### Égalité

Deux sous-ensembles flous  $\tilde{A}$  et  $\tilde{B}$  de  $X$  sont égaux si :

$$\mu_{\tilde{A}}(x) = \mu_{\tilde{B}}(x), \forall x \in X$$

### Complément

Le complément d'un ensemble flou est  $\tilde{A}$  un l'ensemble flou  $\overline{\tilde{A}}$  dont la fonction d'appartenance est :  $\mu_{\overline{\tilde{A}}}(x) = 1 - \mu_{\tilde{A}}(x)$

### Union

L'union de deux sous-ensembles flous  $\tilde{A}$  et  $\tilde{B}$  dans  $X$  est définie par sa fonction d'appartenance suivante :

$$\mu_{(\tilde{A} \cup \tilde{B})}(x) = \max(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x))$$

### Intersection

L'intersection de deux ensembles flous  $\tilde{A}$  et  $\tilde{B}$  dans  $X$  est définie par sa fonction d'appartenance suivante :

$$\mu_{(\tilde{A} \cap \tilde{B})}(x) = \min\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\}$$

### Propriétés de l'union et de l'intersection :

- Commutativité :  $\tilde{A} \cup \tilde{B} = \tilde{B} \cup \tilde{A}$  ;  $\tilde{A} \cap \tilde{B} = \tilde{B} \cap \tilde{A}$ .
- Associativité :  $\tilde{A} \cup (\tilde{B} \cup \tilde{C}) = (\tilde{A} \cup \tilde{B}) \cup \tilde{C}$  ;  $\tilde{A} \cap (\tilde{B} \cap \tilde{C}) = (\tilde{A} \cap \tilde{B}) \cap \tilde{C}$ .
- Distributivité :  $\tilde{A} \cap (\tilde{B} \cup \tilde{C}) = (\tilde{A} \cap \tilde{B}) \cup (\tilde{A} \cap \tilde{C})$ .
- Les relations de Morgan :  $\overline{\tilde{A} \cup \tilde{B}} = \overline{\tilde{A}} \cap \overline{\tilde{B}}$ ;  $\overline{\tilde{A} \cap \tilde{B}} = \overline{\tilde{A}} \cup \overline{\tilde{B}}$
- Les lois d'absorption :  $\tilde{A} \cup (\tilde{A} \cap \tilde{B}) = \tilde{A}$  et  $\tilde{A} \cap (\tilde{A} \cup \tilde{B}) = \tilde{A}$ .

### Exemple 2.6 :

$$X = \{x_1, x_2, x_3, x_4, x_5\}$$

$$\tilde{A} = \{(x_1, 1), (x_2, 0.3), (x_3, 0.7), (x_4, 0.4), (x_5, 0)\}$$

$$\tilde{B} = \{(x_1, 0.9), (x_2, 0.2), (x_3, 0.1), (x_4, 0.6), (x_5, 0.5)\}$$

$$\tilde{A} \subsetneq \tilde{B}$$

$$\tilde{A} \neq \tilde{B}$$

$$\tilde{A} = \{(x_1, 0), (x_2, 0.7), (x_3, 0.3), (x_4, 0.6), (x_5, 1)\}$$

$$\mu_{\tilde{A} \cup \tilde{B}} = \{(x_1, 1), (x_2, 0.3), (x_3, 0.7), (x_4, 0.6), (x_5, 0.5)\}$$

$$\mu_{\tilde{A} \cap \tilde{B}} = \{(x_1, 0.9), (x_2, 0.2), (x_3, 0.1), (x_4, 0.4), (x_5, 0)\}$$

## 2.3.4 Nombre flou de type L-R [13] [9] :

Un nombre flou  $\tilde{A}$  est de représentation L-R [8] [13] , s'il existe deux fonctions  $L$  et  $R$  telle que sa fonction d'appartenance est définie par :

$$\mu_{\tilde{A}}(x) = \begin{cases} L\left(\frac{m-x}{\alpha}\right) & \text{pour } x \leq m, \text{ avec } \alpha > 0 \\ R\left(\frac{x-m}{\beta}\right) & \text{pour } x \geq m, \text{ avec } \beta > 0 \end{cases}$$

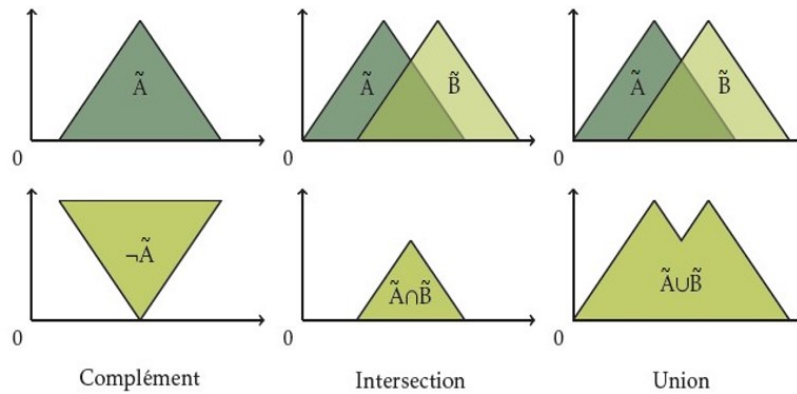


FIGURE 2.3 – Quelques opérations graphiquement

Où :  $L$  et  $R$  sont des fonctions dites de référence du nombre flou  $\tilde{A}$  vérifiant les propriétés suivantes :

- $L$  et  $R$  fonctions non croissantes sur  $[0, +\infty]$
- $L$  et  $R$  fonctions symétriques :  $L(x) = L(-x); R(x) = R(-x), \forall x$
- $L(0) = R(0) = 1$

On note  $\tilde{A} = (m; \alpha; \beta)_{(L-R)}$

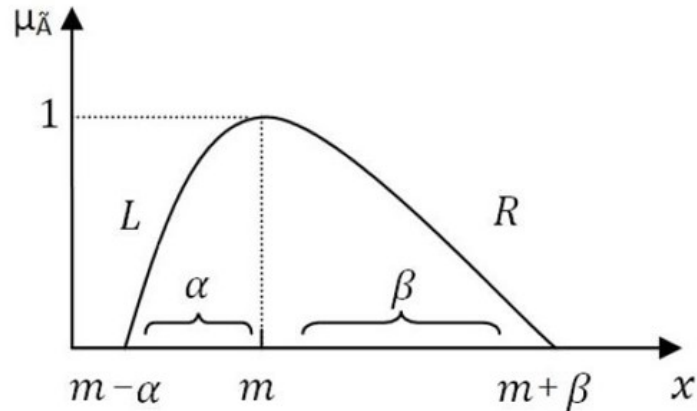


FIGURE 2.4 – Représentation d'un nombre flou type L-R

### 2.3.5 Intervalle flou de type L-R [14]

Un nombre flou plat de type L-R ou intervalle flou, est tel qu'il existe  $m, n \in R$ , avec  $m < n$  de sorte que :  $\mu_{\tilde{A}}(x) = 1 \quad \forall x \in [m, n]$ .

Sa fonction d'appartenance est définie par :

$$\mu_{\tilde{A}}(x) = \begin{cases} L\left(\frac{m-x}{\alpha}\right) & \text{si } x \leq m \\ 1 & \text{si } m < x \leq n \\ R\left(\frac{x-n}{\alpha}\right) & \text{si } x \geq n \end{cases}$$

Nous désignerons un tel nombre flou  $\tilde{A}$  de type L-R par :

$$\tilde{A} = (m, n, \alpha, \beta)_{(L-R)}$$

Où :

$\alpha$  et  $\beta$  sont les écarts à gauche et à droite de  $\tilde{A}$  respectivement,  $m$  et  $n$  sont les valeurs modales inférieures et supérieures de  $\tilde{A}$  respectivement ou bien la moyenne à gauche et à droite de  $\tilde{A}$  respectivement,  $[m, n]$  est le noyau de  $\tilde{A}$ .

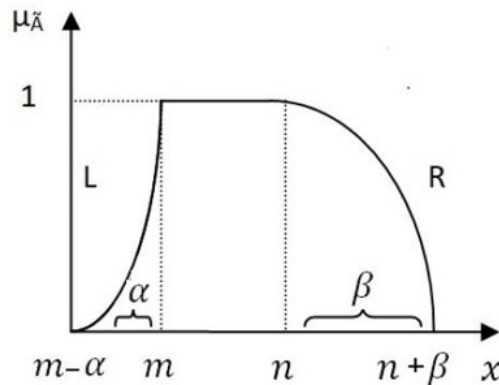


FIGURE 2.5 – Représentation d'un nombre flou plat de type L-R.

**Remarque 2.6 :**

Il existe plusieurs types de nombres flous de type L-R. Lorsque les fonctions de références  $L$  et  $R$  sont linéaires, on parle alors de nombre flou de type triangulaire ou de type trapézoïdal.

**2.3.6 Nombre flou de type triangulaire [15] [16] :**

Un nombre flou est dit de type triangulaire noté  $(a, \alpha, \beta)$  si sa fonction d'appartenance est définie par :  $\alpha > 0, \beta > 0$

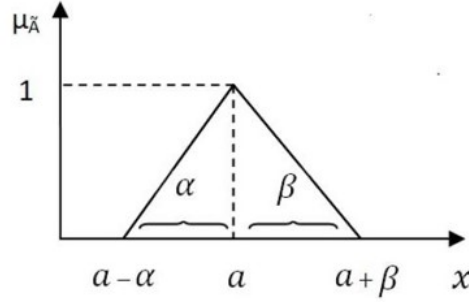
$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x-a+\alpha}{\alpha} & \text{si } a - \alpha \leq x \leq a \\ 1 & \text{si } x = a \\ \frac{a+\beta-x}{\alpha} & \text{si } a \leq x \leq a + \beta \end{cases}$$

**2.3.7 Opérations sur les nombres flous de type triangulaire**

Soient  $\tilde{A}$  et  $\tilde{B}$  deux nombres flous de type triangulaire

**Multiplications scalaires [15]**

Soit le nombre flou triangulaire  $\tilde{A} = (a, \alpha_1, \beta_1)$  on obtient :


 FIGURE 2.6 – Représentation d'un nombre flou triangulaire  $(a; \alpha; \beta)$ 

$$\begin{cases} \text{si } \lambda > 0, \lambda \in R : \lambda \otimes \tilde{A} = (\lambda a, \lambda \alpha_1, \lambda \beta_1) \\ \text{si } \lambda < 0, \lambda \in R : \lambda \otimes \tilde{A} = (\lambda a, \lambda \beta_1, \lambda \alpha_1) \end{cases}$$

#### Addition [15]

Soit deux nombres flous triangulaires  $\tilde{A} = (a, \alpha_1, \beta_1)$  et  $\tilde{B} = (b, \alpha_2, \beta_2)$  On obtient :  
 $\tilde{A} \oplus \tilde{B} = (a + b, \alpha_1 + \alpha_2, \beta_1 + \beta_2)$

#### Soustraction [15]

Étant donnée  $\tilde{A} = (a, \alpha_1, \beta_1)$  et  $\tilde{B} = (b, \alpha_2, \beta_2)$  deux nombres flous triangulaire On obtient :

$$-\tilde{B} = (-b, \alpha_2, \beta_2)$$

Et

$$\tilde{A} \ominus \tilde{B} = (a - b, \alpha_1 + \beta_2, \beta_1 + \alpha_2)$$

#### Comparaison de deux nombres flous triangulaire

Soient deux nombres flous triangulaire  $\tilde{A} = (a, \alpha_1, \beta_1)$  et  $\tilde{B} = (b, \alpha_2, \beta_2)$

$$\begin{aligned} \tilde{A} = \tilde{B} &\iff a = b, \quad \alpha_1 = \alpha_2, \quad \beta_1 = \beta_2 \\ \tilde{A} \leq \tilde{B} &a \leq b, \quad a - \alpha_1 \leq b - \alpha_2, \quad a + \beta_1 \leq b + \beta_2 \end{aligned}$$

#### Nombre flou de type trapézoïdal [16]

Un nombre flou est dit de type trapézoïdal noté  $(a^L, a^U, \alpha, \beta)$  si sa fonction d'appartenance est donnée par :

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x - a^L + \alpha}{\alpha} & \text{if } a^L - \alpha \leq x \leq a^L \\ 1 & \text{if } a^L < x \leq a^U \\ \frac{a^U + \beta - x}{\beta} & \text{if } a^U \leq x \leq a^U + \beta \end{cases}$$

### 2.3.8 Opération sur les nombres flous de type trapézoïdal

Soient  $\tilde{A}$  et  $\tilde{B}$  deux nombres flous trapézoïdaux

#### Multiplication scalaire [22]

Soit le nombre flou  $\tilde{A} = (a^L, a^U, \alpha_1, \beta_1)$  de type trapézoïdal

On obtient :

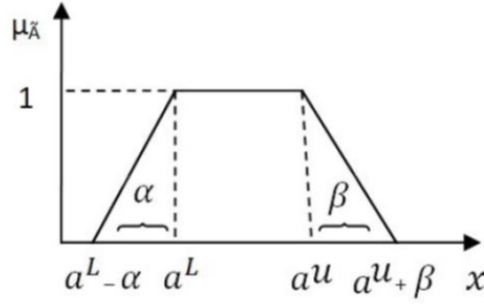


FIGURE 2.7 – Représentation d'un nombre flou trapézoïdale

$$\begin{aligned} \text{Si } \lambda > 0, \lambda \in \mathbb{R} : \lambda \otimes \tilde{A} &= (\lambda a^L, \lambda a^U, \lambda \alpha_1, \lambda \beta_1) \\ \text{Si } \lambda < 0, \lambda \in \mathbb{R} : \lambda \otimes \tilde{A} &= (\lambda a^U, \lambda a^L, -\lambda \beta_1, -\lambda \alpha_1) \end{aligned}$$

**Addition [29]**

Soient deux nombres flous  $\tilde{A} = (a^L, a^U, \alpha_1, \beta_1)$  et  $\tilde{B} = (b^L, b^U, \alpha_2, \beta_2)$  de type trapézoïdaux, on obtient :  $\tilde{A} \oplus \tilde{B} = (a^L + b^L, a^U + b^U, \alpha_1 + \alpha_2, \beta_1 + \beta_2)$

**Soustraction [29]**

Étant donnée deux nombres flous trapézoïdaux  $\tilde{A} = (a^L, a^U, \alpha_1, \beta_1)$  et  $\tilde{B} = (b^L, b^U, \alpha_2, \beta_2)$   
On obtient :

$$\begin{aligned} -\tilde{B} &= -(b^L, b^U, \alpha_2, \beta_2) = (-b^U, -b^L, \beta_2, \alpha_2) \\ \tilde{A} \ominus \tilde{B} &= (a^L - b^U, a^U - b^L, \alpha_1 + \beta_2, \beta_1 + \alpha_2) \end{aligned}$$

**Comparaison de deux nombres flous trapézoïdaux**

Soient deux nombres flous trapézoïdaux  $\tilde{A} = (a^L, a^U, \alpha_1, \beta_1)$  et  $\tilde{B} = (b^L, b^U, \alpha_2, \beta_2)$

$$\begin{aligned} \tilde{A} = \tilde{B} &\Leftrightarrow a^L = b^L, a^U = b^U, \alpha_1 = \alpha_2, \beta_1 = \beta_2 \\ \tilde{A} \leq \tilde{B} &\Leftrightarrow a^L \leq b^L, a^U \leq b^U, a^L - \alpha_1 \leq b^L - \alpha_2, a^U + \beta_1 \leq b^U + \beta_2 \end{aligned}$$

**Remarque 2.7**

On note par  $F(\mathbb{R})$  l'ensemble des nombres flous de type trapézoïdal.

### 2.3.9 Programmation linéaire floue :

La programmation mathématique floue (PLF), également connue sous le nom de logique floue, est une branche des mathématiques et de l'informatique qui traite de la modélisation de l'incertitude. Son origine est l'article de Zadeh et Bellmann publié en 1970, qu'introduisent les concepts des contraintes floues, objectif flou, décision floue. La programmation floue trouve des applications dans de nombreux domaines tels que le contrôle des systèmes, la prise de décision, la reconnaissance de formes, l'optimisation, la robotique, etc. . .

On distingue deux cas :

1. Le cas où les inégalités (ou les égalités) sont relaxées on parlera de programmation flexible.
2. Le cas où les données imprécises sont représentées par des sous-ensembles flous, on parlera de programmation robuste [26] .

**Définition (2.18) [5] :**

Un modèle général d'un (PLF) est présenté sous la forme suivante :

$$\begin{cases} \min \tilde{z} = \tilde{c}'x \\ \tilde{A}x \quad (\tilde{\leq} \tilde{\geq} \tilde{=} ) \quad \tilde{b} \\ x \geq 0 \\ \text{avec } \tilde{A} = (\tilde{a}_{ij})_{\substack{i=1,\dots,m \\ j=1,\dots,n}}, \quad x \in \mathbb{R}^n \end{cases} \quad (\text{PLF})$$

Où  $\tilde{c}_j, \tilde{a}_{ij}, \tilde{b}_i$  sont des nombres flous

Les symboles  $\tilde{\min}$  et  $\tilde{\leq}$  (inégalité floue appelée aussi inégalité flexible) respectivement représentent les versions flous de minimiser et  $\leq$ .

**1) Programmation flexible [30] :**

Un problème dont l'objectif et les contraintes sont vaguement définies.

**Décision dans un environnement flou**

On considère un problème dont l'objectif et les  $m$  contraintes sont vaguement définis, représentés par des ensembles flous d'un même référentiel  $X$  dont les fonctions d'appartenance respectives sont  $\mu_0$  et  $\mu_i$  pour  $i = 1, \dots, m$ . La décision qui doit satisfaire l'objectif et les  $m$  contraintes est donc représentée par un ensemble flou, intersection de ces derniers et dont la fonction d'appartenance est  $\mu_D$  telle que [21] :

$$\mu_D(x) = \min(\mu_I(x)/i = 0, 1, 2, \dots, m)$$

La meilleure décision est déterminée par la résolution du problème suivant :

$$\max(\mu_D(x)/x \in X) = \max\{\min(\mu_i(x)/i = 0, 1, 2, \dots, m) / x \in X\}$$

**Résolution d'un programme linéaire flexible [8] [13] [30]**

Considérons le programme linéaire flexible suivant [21] [22] [26] :

$$\begin{cases} \tilde{\max} z = c'x \\ A_i x \quad \tilde{\theta} \quad b_i, \quad i = 1, \dots, m \\ x \geq 0 \end{cases} \quad (2.22)$$

où  $\tilde{\theta} \in \{\tilde{\leq}, \tilde{\geq}, \tilde{=}\}$  et  $\tilde{\leq}, \tilde{\geq}, \tilde{=}$  sont les versions flexibles de  $\leq, \geq, =$  respectivement .

La notation-désigne le fait que l'objectif et les contraintes ne sont pas des impératifs stricts.

Selon Zimmermann, le programme (2.22) peut s'interpréter comme suit :  $x \in \mathbb{R}_n^+$  telle que :

$$\begin{cases} c'x \quad \tilde{\leq} \quad z_0 \\ A_i x \quad \tilde{\theta} \quad b_i, \quad i = 1, \dots, m \\ x \geq 0 \end{cases} \quad (2.23)$$

$z_0$  : est une valeur fixée

L'objectif flou et les contraintes floues peuvent être représentés par les ensembles flous respectifs  $U_0$  et  $U_i, i = 1, \dots, m$  dont les fonctions d'appartenance sont respectivement  $\mu_0$  et  $\mu_i, i = 1, \dots, m$  est définie selon que  $\tilde{\theta}$  est  $\lesssim, \gtrsim$  ou  $\cong$  comme suit :

$\tilde{\theta}$  est  $\lesssim$

$$\mu_i(x) = \begin{cases} 1 & \text{si } A_i x \leq b_i \\ 1 - \frac{A_i x - b_i}{d_i} & \text{si } b_i \leq A_i x \leq b_i + d_i \\ 0 & \text{si } A_i x > b_i + d_i \\ i = 1, \dots, m \end{cases} \quad (2.24)$$

Représentation graphique :

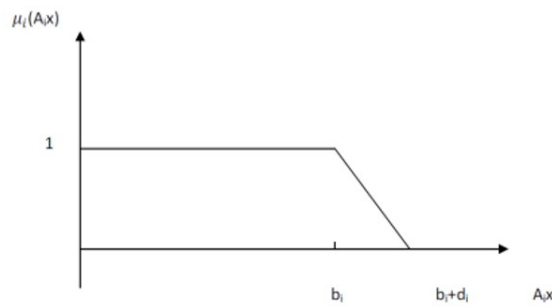


FIGURE 2.8 – Contraintes du type  $A_i(x) \lesssim b_i, i = 1, \dots, m$

$\tilde{\theta}$  est  $\cong$

$$\mu_i(x) = \begin{cases} 1 & \text{si } A_i x = b_i \\ 1 - \frac{A_i x - b_i}{d_i} & \text{si } b_i < A_i x \leq b_i + d_i \\ 1 - \frac{b_i - A_i x}{d_i} & \text{si } b_i - d_i \leq A_i x < b_i \\ 0 & \text{si } A_i x > b_i + d_i, A_i x < b_i - d_i \\ i = 1, \dots, m \end{cases} \quad (2.25)$$

Représentation graphique :

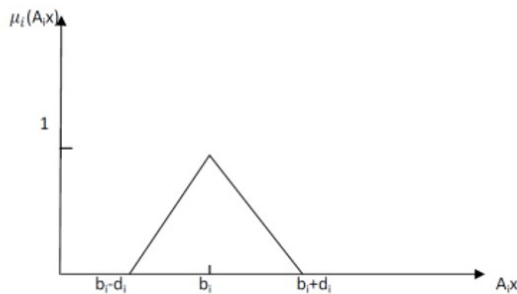


FIGURE 2.9 – Contraintes du type  $A_i(x) \cong b_i, i = 1, \dots, m$

$\tilde{\theta}$  est  $\gtrsim$

$$\mu_i(x) = \begin{cases} 1 & \text{si } A_i x \geq b_i \\ 1 - \frac{b_i - A_i x}{d_i} & \text{si } b_i - d_i \leq A_i x \leq b_i \\ 0 & \text{si } A_i x < b_i - d_i \end{cases} \quad (2.26)$$

$$i = 1, \dots, m$$

Représentation graphique :

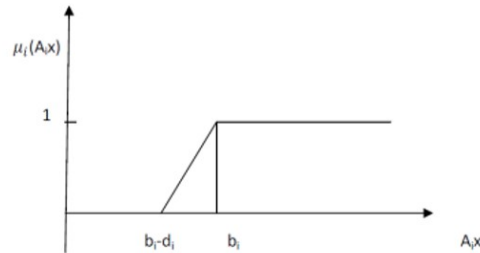


FIGURE 2.10 – Contraintes du type  $A_i(x) \gtrsim b_i, i = 1, \dots, m$

**Remarque 2.8 :**

La méthode flexible joue un rôle identique à la fonction objectif et aux contraintes, on détermine la fonction d'appartenance de la fonction objectif de la même manière que les contraintes. Pour se faire, il faut fixer un objectif à atteindre et le degré de satisfaction.

**Méthode d'agrégation des degrés de satisfaction :**

Cette méthode est basée sur le principe d'agrégation de Bellman et Zadeh pour déduire une fonction objectif finale à maximiser. Cette fonction exprime le degré total de satisfaction.

$$\mu_D(x) = \min(\mu_0(x), \mu_1(x), \dots, \mu_m(x))$$

Chercher la solution qui réalise le meilleur degré de satisfaction. La solution optimale est alors déterminée par résolution du problème déterministe.

$$\max_{x \geq 0} \mu_D(x)$$

Qui est équivalent au problème suivant [14] :

$$\begin{cases} \max z = \lambda \\ \lambda \leq \mu_i(x), \quad \forall i = 0, 1, 2, \dots, m \\ 0 \leq \lambda \leq 1 \\ x \geq 0 \end{cases} \quad (2.27)$$

**Exemple 2.7 :**

On considère le programme flexible suivant :

$$\begin{cases} \widetilde{\max} z & 5x_1 - x_2 \\ & -x_1 + 2x_2 \leq 5 \\ & 3x_1 + x_2 \leq 11 \\ & x_1 \geq 0, \quad x_2 \geq 0 \end{cases} \quad (2.28)$$

1. On détermine les fonctions d'appartenance de l'objectif et des contraintes sachant que l'objectif n'excède la valeur  $z_0 = 6$  et que les écarts de tolérance de l'objectif et des contraintes sont respectivement  $d_0 = 2$ ,  $d_1 = 5$ ,  $d_2 = 1$  : Selon Zimmermann, le programme (2.28) peut s'interpréter comme suit :

$$\begin{cases} 5x_1 - x_2 \leq 6 \\ -x_1 + 2x_2 \leq 5 \\ 3x_1 + x_2 \leq 11 \\ x_1 \geq 0, \quad x_2 \geq 0 \end{cases} \quad (2.29)$$

### Transformation de l'objectif

$$\mu_0(x) = \begin{cases} 1 & \text{si } 5x_1 - x_2 \leq 6 \\ 1 - \frac{5x_1 - x_2 - 6}{2} & \text{si } 6 \leq 5x_1 - x_2 \leq 8 \\ 0 & \text{si } 5x_1 - x_2 > 8 \end{cases} \quad (2.30)$$

$$= \begin{cases} 1 & \text{si } 5x_1 - x_2 \leq 6 \\ \frac{-5x_1 + x_2 + 8}{2} & \text{si } 6 \leq 5x_1 - x_2 \leq 8 \\ 0 & \text{si } 5x_1 - x_2 > 8 \end{cases} \quad (2.31)$$

### Transformation de la 1ere contrainte :

$$\mu_1(x) = \begin{cases} 1 & \text{si } -x_1 + 2x_2 \leq 5 \\ 1 - \frac{-x_1 + 2x_2 - 5}{5} & \text{si } 5 \leq -x_1 + 2x_2 \leq 10 \\ 0 & \text{si } -x_1 + 2x_2 > 10 \end{cases} \quad (2.32)$$

$$= \begin{cases} 1 & \text{si } -x_1 + 2x_2 \leq 5 \\ \frac{x_1 - 2x_2 + 10}{5} & \text{si } 5 \leq -x_1 + 2x_2 \leq 10 \\ 0 & \text{si } -x_1 + 2x_2 > 10 \end{cases} \quad (2.33)$$

### Transformation de la 2eme contrainte :

$$\mu_2(x) = \begin{cases} 1 & \text{si } 3x_1 + x_2 \leq 11 \\ 1 - (3x_1 + x_2 - 11) & \text{si } 11 \leq 3x_1 + x_2 \leq 12 \\ 0 & \text{si } 3x_1 + x_2 > 12 \end{cases} \quad (2.34)$$

$$= \begin{cases} 1 & \text{si } 3x_1 + x_2 \leq 11 \\ -3x_1 - x_2 + 12 & \text{si } 11 \leq 3x_1 + x_2 \leq 12 \\ 0 & \text{si } 3x_1 + x_2 > 12 \end{cases} \quad (2.35)$$

2. 2) Le problème déterministe associé à (2.28) est :

$$\begin{cases} \max z = \lambda \\ \lambda \leq \frac{-5x_1 + x_2 + 8}{2} \\ \lambda \leq \frac{x_1 - 2x_2 + 10}{5} \\ \lambda \leq -3x_1 - x_2 + 2 \\ x_1 \geq 0, \quad x_2 \geq 0, \quad 0 \leq \lambda \leq 1 \end{cases} \quad (2.36)$$

$$\begin{cases} \max z = \lambda \\ 2\lambda \leq -5x_1 + x_2 + 8 \\ 5\lambda \leq x_1 - 2x_2 + 10 \\ \lambda \leq -3x_1 - x_2 + 12 \\ x_1 \geq 0, \quad x_2 \geq 0, \quad 0 \leq \lambda \leq 1 \end{cases} \quad (2.37)$$

$$\begin{cases} \max z = \lambda \\ 2\lambda + 5x_1 - x_2 \leq 8 \\ 5\lambda - x_1 + 2x_2 \leq 10 \\ \lambda + 3x_1 + x_2 \leq 12 \\ x_1 \geq 0, \quad x_2 \geq 0, \quad 0 \leq \lambda \leq 1 \end{cases} \quad (2.38)$$

**La forme canonique (standard)**

$$\begin{cases} \max z = \lambda \\ 2\lambda + 5x_1 - x_2 + x_3 = 8 \\ 5\lambda - x_1 + 2x_2 + x_4 = 10 \\ \lambda + 3x_1 + x_2 + x_5 = 12 \\ \lambda + x_6 = 1 \\ x_j \geq 0, \quad j = 1, 2, 3, 4, 5, 6 \end{cases} \quad (2.39)$$

**Résolution du problème (2.28) par la méthode du simplexe :**

$C$			1	0	0	0	0	0	1	$\theta_j$
	Base	$b$	$\lambda$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	
0	$x_3$	8	2	5	-1	1	0	0	0	4
0	$x_4$	10	5	-1	2	0	1	0	0	2
0	$x_5$	12	1	3	1	0	0	1	0	12
0	$x_6$	4	1	0	0	0	4	0	4	1
		$\Delta_j$	-1	0	0	0	0	0	0	
0	$x_3$	6	0	5	-1	1	0	0	-2	
0	$x_4$	5	0	-1	2	0	1	0	-5	
0	$x_5$	11	0	3	1	0	0	1	-1	
1	$\lambda$	1	1	0	0	0	0	0	1	
		$\Delta_j$	0	0	0	0	0	0	1	

## 2) Programmation robuste [24] [13] [31] :

Lorsque les coefficients des contraintes sont des données imprécises, l'inégalités remplacées par  $\subseteq$ , on parle alors de la programmation robuste (généralisation d'un problème linéaire inexacte).

### Programmation linéaire inexacte (Solster) :

Un programme linéaire inexact est un programme de la forme [32] :

$$\begin{cases} \max z = c'x \\ x_1K_1 + x_2K_2 + \dots + x_nK_n \subseteq K \\ x_j \geq 0, j = 1, \dots, n \end{cases} \quad (2.40)$$

Où :

$(K_j)$ ;  $j = 1, \dots, n$  et  $K$  sont des ensembles convexes de  $\mathbb{R}^n$ ,  $\subseteq$  est l'inclusion entre ensembles, et  $+$  représente l'addition ensembliste, elle est définie comme suit : soient  $A$  et  $B$  deux ensembles, alors  $A+B = \{a+b, a \in A \text{ et } b \in B\}$ .

### Résolution d'un Programme robuste :

Un programme robuste est un programme de la forme : [32]

$$\begin{cases} \max z = c'x \\ x_1 \odot A_1 \oplus x_2 \odot A_2 \oplus \dots \oplus x_n \odot \tilde{A}_n \subseteq \tilde{B} \\ x_j \geq 0, j = 1, \dots, n \end{cases} \quad (2.41)$$

Où :

$(\tilde{A}_j)$ ,  $j = 1, \dots, n$  sont des sous-ensembles flous de  $\mathbb{R}$ .

$\oplus$  : Addition des ensembles flous.

$\subseteq$  : Inclusion entre ensembles flous.

On représente l'ensemble des contraintes de (2.40) par :

$$E = \left\{ \{x \in \mathbb{R}^n / \quad x_1 \odot \tilde{A}_1 \oplus x_2 \odot \tilde{A}_2 \oplus \dots \oplus x_n \odot \tilde{A}_n \subseteq \tilde{b}, \quad x \geq 0\} \right\}$$

**Théorème [6]**

$x^0 \in E$  est optimale pour (2.40) si et seulement si  $x^0 \in E$  est optimal pour le programme suivant :

$$\begin{cases} \max z = c'x \\ x_1 A_1^\alpha + x_2 A_2^\alpha + \dots + x_n A_n^\alpha \subseteq \tilde{b}^\alpha, \quad \forall \alpha \in [0, 1] \\ x_j \geq 0, j = 1, \dots, n \end{cases} \quad (2.42)$$

**Preuve : [24]**

$\Rightarrow$  (CN)  $x_0 \in E$ , alors  $x_1^0 A_1^\sim + x_2^0 A_2^\sim + \dots + x_n^0 A_n^\sim \subseteq \tilde{b}$ .

$\alpha \in [0, 1]$ ,  $(x_1^0 A_1^\sim + x_2^0 A_2^\sim + \dots + x_n^0 A_n^\sim)^\alpha \subseteq \tilde{b}^\alpha$ .

$\alpha \in [0, 1]$ ,  $(x_1^0 A_1^\sim)^\alpha + (x_2^0 A_2^\sim)^\alpha + \dots + (x_n^0 A_n^\sim)^\alpha \subseteq \tilde{b}^\alpha$ .

$\alpha \in [0, 1]$ ,  $x_1^0 A_1^{\sim\alpha} + x_2^0 A_2^{\sim\alpha} + \dots + x_n^0 A_n^{\sim\alpha} \subseteq \tilde{b}^\alpha$ .

$\Leftarrow$  (CS) Soit  $x_1^0 A_1^{\sim\alpha} + x_2^0 A_2^{\sim\alpha} + \dots + x_n^0 A_n^{\sim\alpha} \subseteq \tilde{b}^\alpha$  (\*).

(2.42) est un programme semi-infini, c'est à dire un programme avec une infinité de contraintes. Si on suppose que les images des fonctions d'appartenance des sous-ensembles flous sont discrètes et finies, alors on obtient un programme linéaire avec un nombre fini de contraintes comme suit :

**Proposition [32]**

Si  $Im(\mu \tilde{A}_j = \{\alpha_1, \alpha_2, \dots, \alpha_p\})$  avec  $0 = \alpha_1 < \alpha_2 < \dots < \alpha_p < 1$

Alors

$$E = \{x \in \mathbb{R}^n / x_1 \odot \tilde{A}_1^\oplus \oplus x_2 \odot \tilde{A}_2^\oplus \dots \oplus x_n \odot \tilde{A}_n^\oplus \subseteq ]\tilde{b}, x \geq 0\}.$$

Si seulement si

$$E' = \{x \in \mathbb{R}^n / x_1 \tilde{A}_1^{\sim\alpha_k} + x_2 \tilde{A}_2^{\sim\alpha_k} + \dots + x_n \tilde{A}_n^{\sim\alpha_k} \subseteq \tilde{b}^{\alpha_k}, k = 1, \\ x_j \geq 0; j = 1, 2, \dots, n\}$$

**Cas des nombres flous [8]**

$\tilde{a}_{ij}$  de  $\tilde{A}$  et  $\tilde{b}_i$  de  $\tilde{B}$  sont des nombres flous dont les ensembles de niveau respectifs sont des intervalles compacts. Alors, le programme (2.41) s'écrit sous la forme suivante :

$$\begin{cases} \max z = c'x \\ [\tilde{a}_{11}^{\alpha_1} \bar{\tilde{a}}_{11}^{\alpha_1}]x_1 + [\tilde{a}_{12}^{\alpha_1} \bar{\tilde{a}}_{12}^{\alpha_1}]x_2 + \dots + [\tilde{a}_{1n}^{\alpha_1} \bar{\tilde{a}}_{1n}^{\alpha_1}]x_n \subseteq [\tilde{b}_1^{\alpha_1} \bar{\tilde{b}}_1^{\alpha_1}] \\ [\tilde{a}_{21}^{\alpha_2} \bar{\tilde{a}}_{21}^{\alpha_2}]x_1 + [\tilde{a}_{22}^{\alpha_2} \bar{\tilde{a}}_{22}^{\alpha_2}]x_2 + \dots + [\tilde{a}_{2n}^{\alpha_2} \bar{\tilde{a}}_{2n}^{\alpha_2}]x_n \subseteq [\tilde{b}_2^{\alpha_2} \bar{\tilde{b}}_2^{\alpha_2}] \\ \vdots \\ [\tilde{a}_{m1}^{\alpha_p} \bar{\tilde{a}}_{m1}^{\alpha_p}]x_1 + [\tilde{a}_{m2}^{\alpha_p} \bar{\tilde{a}}_{m2}^{\alpha_p}]x_2 + \dots + [\tilde{a}_{mn}^{\alpha_p} \bar{\tilde{a}}_{mn}^{\alpha_p}]x_n \subseteq [\tilde{b}_m^{\alpha_p} \bar{\tilde{b}}_m^{\alpha_p}] \\ x_j \geq 0 \quad k = 1, \dots, p \end{cases} \quad (2.43)$$

Où :

$$\begin{aligned}\tilde{a}_{ij}^{\alpha_k} &= \{x \in \mathbb{R} / \mu_{\tilde{a}_{ij}}(x) \geq \alpha_k\} \\ \underline{\tilde{a}}_{ij}^{\alpha_k} &= \inf\{x \in \mathbb{R} / \mu_{\tilde{a}_{ij}}(x) \geq \alpha_k\} \\ \overline{\tilde{a}}_{ij}^{\alpha_k} &= \sup\{x \in \mathbb{R} / \mu_{\tilde{a}_{ij}}(x) \geq \alpha_k\} \\ \tilde{b}_i^{\alpha_k} &= \{x \in \mathbb{R} / \mu_{\tilde{b}_i}(x) \geq \alpha_k\} \\ \underline{\tilde{b}}_i^{(\alpha_k)} &= \inf\{x \in \mathbb{R} / \mu_{\tilde{b}_i}(x) \geq \alpha_k\} \\ \overline{\tilde{b}}_i^{(\alpha_k)} &= \sup\{x \in \mathbb{R} / \mu_{\tilde{b}_i}(x) \geq \alpha_k\}\end{aligned}$$

**Remarque [11] :**

- $\tilde{a}_{ij}$  et  $\tilde{b}_i$  sont des nombres flous, donc ce sont des ensembles normalisés de  $\mathbb{R}$ .
- Les ensembles  $\tilde{a}_{ij}^{\alpha_k}$  et  $\tilde{b}_i^{\alpha_k}$  sont des convexe dans  $\mathbb{R}$ , donc des intervalles de  $\mathbb{R}$ .
- Et par hypothèse  $\tilde{a}_{ij}^{\alpha_k}$  et  $\tilde{b}_i^{\alpha_k}$  sont compact.
- Par conséquent :

**Cas des nombres flous de types L-R :**

Soit les composante  $\tilde{a}_{ij}$  de  $\tilde{A}$  et  $\tilde{b}_i$  de  $\tilde{b}$  sont des nombres flous de type L-R [5].

On a :

$$\tilde{a}_{ij}^{\alpha_k} = [(\underline{\tilde{a}})_{ij}^{\alpha_k}, (\overline{\tilde{a}})_{ij}^{\alpha_k}], \quad \tilde{b}_i^{\alpha_k} = [(\underline{\tilde{b}})_i^{\alpha_k}, (\overline{\tilde{b}})_i^{\alpha_k}]$$

$$\text{Où } \begin{cases} \alpha_{ij} > 0 \\ \beta_{ij} > 0 \\ \alpha_i > 0, \quad \beta_i > 0 \end{cases}$$

(2.41) Deviendra :

$$\begin{cases} \max z = cx \\ \sum_{j=1}^n m_{ij}x_j = m_i \\ \sum_{j=1}^n n_{ij}x_j = n_i \\ \sum_{j=1}^n \alpha_{ij}x_j = \alpha_i \\ \sum_{j=1}^n \beta_{ij}x_j = \beta_i \\ x_j > 0 \quad \forall j = 1, 2, \dots, n \end{cases}$$

**Exemple (robuste) :** posant  $\alpha = 0.5$  et en utilisant l'algorithme du simplexe, on trouve la solution optimal du problème (2.44) :

$$\begin{cases} \max z(\mathbf{x}) = x_1 + 2x_2 \\ \tilde{a}_1 x_1 + \tilde{a}_2 x_2 \supseteq \tilde{b} \\ x_1 \geq -1 \\ x_1 > 0, \quad x_2 > 0 \end{cases} \quad (2.44)$$

Sachant que :

$$\tilde{a}_1 = \{1/0.7, 2/0.5, 3/0.6, 5/0.3\}$$

$$\tilde{a}_2 = \{5/0.2, 3/0.9, 1/0.7, 2/0.6\}$$

$$\tilde{b} = \{6/0.7, 1/0.2, -2/0.8, 5/0.4\}$$

On aura :

$$\begin{cases} \max z(x) = x_1 + 2x_2 \\ [1, 3]x_1 + [1, 3]x_2 \subseteq [-2, 6] \\ x_1 \geq -1 \\ x_1 > 0, \quad x_2 > 0 \end{cases} \quad (2.45)$$

$$\begin{cases} \max z(x) = x_1 + 2x_2 \\ x_1 + x_2 \geq -2 \\ 3x_1 + 3x_2 \leq 6 \\ x_2 \geq -1 \\ x_1 \geq 0, \quad x_2 \geq 0 \end{cases} \quad (2.46) \Rightarrow \begin{cases} \max z(x) = x_1 + 2x_2 \\ x_1 + x_2 - x_3 = -2 \\ 3x_1 + 3x_2 + x_4 = 6 \\ x_2 - x_5 = -1 \\ x_j \geq 0, \quad j = 1, \dots, 5 \end{cases} \quad (2.47) \Rightarrow \begin{cases} \max z(x) = x_1 + 2x_2 \\ -x_1 - x_2 + x_3 = 2, \\ 3x_1 + 3x_2 + x_4 = 6, \\ -x_2 + x_5 = 1, \\ x_j \geq 0, \quad j = 1, \dots, 5. \end{cases}$$

(2.48)

Résolution par simplexe :

	$C$		1	2	0	0	0	$\theta$
	Base	$b$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	
0	$x_3$	2	-1	-1	1	0	0	/
0	$x_4$	6	3	3	0	1	0	2
0	$x_5$	1	0	-1	0	0	1	/
		$\Delta_j$	-1	-2	0	0	0	
1	$x_3$	4	0	0	1	$\frac{1}{3}$	0	
0	$x_2$	2	1	1	0	$\frac{1}{3}$	0	
0	$x_5$	3	1	0	0	$\frac{1}{3}$	1	
		$\Delta_j$	1	0	0	$\frac{2}{3}$	0	

Le critère d'optimalité est vérifié, tous les  $\Delta_j > 0$ .

La solution optimale du problème (2.44) est  $x^* = (0.2)$ , et  $z^* = 4$ .

## 2.4 Programmation stochastique :

Dans divers domaines tel que la finance, les sciences humaines, la physique et l'ingénierie...etc. on s'intéresse à de nombreux phénomènes dans lesquels apparaît souvent l'effet d'un hasard, dont ces phénomènes sont caractérisés par le fait que les résultats des observations varient d'une expérience à l'autre. En théorie des probabilités on dit qu'un phénomène est stochastique s'il dépend de variable (s) aléatoire (s) et une variable est dite aléatoire si l'ensemble de ses valeurs dépend du hasard. Un processus est un enchaînement ordonné de faits ou de phénomènes, répondant à un certain schéma et aboutissant à quelque chose. En finance, les processus stochastiques sont utilisés pour modéliser les mouvements aléatoires des marchés financiers, tels que les prix des actions, des obligations et des devises. Les équations différentielles stochastiques, telles que le modèle de Black-Scholes, sont utilisées pour évaluer les options et les dérivés financiers.

### 2.4.1 Notion d'espace mesurable [8]

Soit  $\Omega$  l'espace fondamental, il peut être fini, infini, dénombrable ou non.

#### Tribus d'évènements

Une famille  $F$  de sous-ensembles de  $\Omega$  s'appelle tribus si :

$$\Omega \in F.$$

$$\forall A \in F, \Rightarrow \bar{A} \in F, \text{ où } \bar{A} \text{ désigne le complémentaire de } A.$$

$$\forall A \in F, \forall B \in F, \Rightarrow A \cup B \in F.$$

Quel que soit la suite  $(A_n)_{n \in \mathbb{N}}$  d'éléments de  $F$ , on a  $\bigcup_{n \in \mathbb{N}} A_n \in F$ .

#### Espace probabilisable

Soit  $(\Omega, F)$  un espace probabilisable, une probabilité  $P$  est une application de  $F \rightarrow [0, 1]$  Telle que :

$$P(\Omega) = 1.$$

$$P(A \cup B) = P(A) + P(B) - P(A \cap B).$$

$$P\left(\bigcup_{i \in I} A_i\right) = \sum_{i \in I} P(A_i) \text{ où les } (A_i)_{i \in I} \text{ sont deux à deux disjoints et } I \text{ dénombrable,}$$

$(\Omega, F, P)$  s'appelle espace probabilisé.

#### Applications mesurables

Soient  $(\Omega, F)$  et  $(E, T)$  deux espace probabilisable, toute application :

$$X : \Omega \longrightarrow E \text{ est dite mesurable si : } \forall A \in T \Rightarrow X^{-1}(A) \in F.$$

### 2.4.2 Variables aléatoires [15] [12] [17]

#### Définition 2.19

Soient  $(\Omega, A)$  et  $(E, F)$  deux espaces probabilisables et  $X$  une application

$$\begin{aligned} X : \Omega &\rightarrow E \\ \omega &\rightarrow X(\omega). \end{aligned}$$

$X$  est dit variable aléatoire (v.a) si :

$B \in F, X^{-1}(B) \in A, X^{-1}(B) = \{\omega \in \Omega / X(\omega) \in B\}$ .

Donc, l'image réciproque de tout évènement par  $X$  est aussi un évènement.

### Propriétés :

Soit  $X$  et  $Y$  sont deux variables aléatoires alors :

- $aX + bY$  est aussi une variable aléatoire  $\forall a, b \in R$
- $\frac{X}{Y}$  avec  $Y \neq 0$  est aussi de variables aléatoires
- $\forall f$  continue,  $f(X)$  est aussi une variable aléatoire

### 2.4.3 Fonction de répartition [15] [17]

On appelle fonction de répartition d'une variable aléatoire  $X$ , la fonction notée  $F$  définie par :

$$F : R \rightarrow [0, 1]$$

$$F(x) = P(X \leq x), P \text{ est une probabilité}$$

### 2.4.4 Densité de probabilité [15] [17]

Si la fonction de répartition  $F$  d'une variable aléatoire continue  $X$  est dérivable en tout point  $x \in R$ , de dérivée  $f(x)$ , sauf en un nombre fini de points, et si :

$$\forall x \in \mathbb{R}, P(X \leq x) = F(x) = \int_{-\infty}^x f(t), dt$$

On dit que  $X$  est une (v.a) absolument continue, appelée la densité de probabilité de  $X$  ou bien la fonction de distribution de  $X$ .

### Fonction de densité

Soit  $f$  une application de  $R$  dans  $R$ ,  $f$  est dite fonction de densité si :

- $f(x) \geq 0, \forall x \in R$ .
- $f$  est intégrable.
- $\int_{-\infty}^{\infty} f(x) dx = 1$

### 2.4.5 Lois de probabilité [15] [17]

La loi de probabilité  $P(x)$ , est une fonction qui associe à chaque valeur  $x$  de la variable aléatoire  $X$  sa probabilité  $P(X = x)$ . On écrit :  $P(x) = P(X = x)$  et on l'appelle loi de probabilité de  $X$ .

### Variable discrète

Une (v.a) est dite discrète si l'ensemble des valeurs qu'elle prend est soit fini soit infini dénombrable.

**o Loi uniforme discrète**

Soit  $\{x_1, \dots, x_N\}$  une partie de  $R$  à  $N$  éléments. Une variable aléatoire  $X : \Omega \rightarrow R$  suit une loi uniforme sur  $\{x_1, \dots, x_N\}$  si :

- $X(\Omega) = \{x_1, \dots, x_N\}$
- $P(X = x_1) = \dots = P(X = x_N) = \frac{1}{N}$ .

La variable aléatoire  $X$  admet alors une espérance et une variance données par :

$$E(X) = \frac{n+1}{2} \quad \text{et} \quad V(X) = \frac{n^2-1}{12}$$

**Loi de Bernoulli**

On appelle v.a indicatrice de l'événement  $A$ , la v.a définie par  $X = 1_A$  :

$$X(\omega) = 1_A(\omega) = \begin{cases} 1 & \text{si } \omega \in A \\ 0 & \text{si } \omega \in \bar{A} \end{cases}$$

Ainsi  $X(\Omega) = \{0, 1\}$  avec  $P_X(X = 0) = P(\bar{A}) = q$  et  $P_X(X = 1) = P(A) = p$ . On dit que  $X$  suit une loi de Bernoulli de paramètre  $p$ , ce qu'on écrit  $X \sim B(1, p)$ , de moments  $E(X) = p$  et  $V(X) = pq$ .

**Loi de Binomiale**

On dit qu'une variable aléatoire ( $X : \Omega \rightarrow R$  suit une loi binomiale de paramètres  $n \geq 1$  et  $p \in [0, 1]$ , ce que l'on note  $X \sim B(n, p)$  si :

- $X$  prend ses valeurs dans  $\{0, \dots, n\}$
- Pour tout  $k$  dans  $\{0, \dots, n\}$ , on a  $P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$

$X$  admet une espérance et une variance données par :

$$E(X) = np \quad \text{et} \quad V(X) = np(1-p)$$

**Loi de Poisson**

Soit  $\lambda > 0$ , On dit qu'une variable aléatoire  $X$  suit la loi de Poisson de paramètre  $\lambda$ , ce que l'on note  $X \sim P(\lambda)$  si :

- $X(\Omega) = \mathbb{N}$
- Pour tout  $k \geq 0$ ;  $P(X = k) = \frac{e^{-\lambda} \lambda^k}{k!}$

$X$  admet alors une espérance et une variance  $E(X) = \lambda$  et  $V(X) = \lambda$ .

**Variable continue**

Soit  $X$  une variable aléatoire continue, alors la loi de  $X$  est caractérisée par l'ensemble des probabilités :  $P(a < X < b) = \int_a^b f_X(x) dx$ .

où  $f_X(x)$  sont deux est la densité de probabilité de  $X$  et  $a, b$  nombres réels infinis (ou pas).

**Loi uniforme**

Si  $a$  et  $b$  sont deux réels, la loi uniforme sur l'intervalle  $[a, b]$  est notée  $(a, b)$ , elle a pour densité :

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{si } x \in [a, b], \\ 0 & \text{sinon.} \end{cases}$$

Et sa fonction de répartition définie par :

$$F(x) = \begin{cases} 0 & \text{si } x < a, \\ \frac{x-a}{b-a} & \text{si } a \leq x < b \\ 1 & \text{si } x \geq b. \end{cases}$$

Son espérance et sa variance mathématiques :

$$E(X) = \frac{a+b}{2} \quad \text{et} \quad V(X) = \frac{(b-a)^2}{12}$$

### Loi normale

On dit qu'une variable aléatoire  $X$  suit la loi normale de paramètres  $m \in \mathbb{R}$  et  $\sigma^2$  avec  $\sigma > 0$ , ce que l'on note  $X \sim \mathcal{N}(m, \sigma^2)$  si elle est continue et admet une densité comme suit :

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right)$$

Avec  $x \in \mathbb{R}$

Et sa fonction de répartition définie par :

$$F(x) = P(X \leq x) = \int_{-\infty}^x \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-m}{\sigma}\right)^2\right) dx$$

$X$  admet alors une espérance et une variance  $E(X) = m$  et  $V(X) = \sigma^2$ .

### Loi exponentielle

On dit que  $X$  suit une loi exponentielle de paramètre  $\lambda > 0$  notée  $\epsilon(\lambda)$ , si la loi de  $X$  a pour densité :

$$f_X(x) = \begin{cases} \lambda \exp(-\lambda x) & \text{si } x \geq 0, \\ 0 & \text{si } x < 0 \end{cases}$$

Et sa fonction de répartition définie par :

$$F(x) = P(X \leq x) = 1 - \exp(-\lambda x)$$

Son espérance et sa variance

$$E(X) = \frac{1}{\lambda} \quad \text{et} \quad V(X) = \frac{1}{\lambda^2}$$

## 2.4.6 Les moments d'une variable aléatoire [17]

### Espérance

Si  $X$  (v. a) discrète, qui prend les valeurs dans  $\{x_1, x_2, \dots, x_k\}$  l'espérance de  $X$  est définie par :  $E(X) = \sum_{i=1}^k x_i p(X = x_i)$ .

Si  $X$  est une (v. a) absolument continue de densité  $f$ , l'espérance de  $X$  est définie par :

$$E(X) = \int_{-\infty}^{+\infty} x f(x) dx$$

### Propriétés de l'espérance

— Soient  $a$  et  $b$  deux constantes et  $X$  une (v. a) :

$$E(aX + b) = a \cdot E(X) + b$$

— Soient  $X$  et  $Y$  deux (v. a) :

$$E(X + Y) = E(X) + E(Y)$$

— Soient  $X$  et  $Y$  deux (v. a) indépendantes :

$$E(X \cdot Y) = E(X) \cdot E(Y)$$

### Variance

La variance de la variable aléatoire  $X$  :

$$V(X) = E[(X - E[X])^2]$$

### L'écart type

On appelle l'écart type la racine carrée de la variance est définie par :

$$\sigma(X) = \sqrt{V(X)}$$

### La covariance

La covariance de deux (v. a)  $X, Y$  notée  $\text{cov}(X, Y)$  :

$$\text{cov}(X, Y) = E(XY) - E(X)E(Y)$$

## 2.4.7 Définition d'un programme stochastique [9]

Dans un cas général, le problème linéaire stochastique s'écrit :

$$\begin{cases} \text{"min"} Z(\omega) = c'(\omega)x \\ A(\omega)x \leq b(\omega) \\ x \in T_1 \end{cases} \quad (2.49)$$

Où  $(A, c, b)$  dont les paramètres de dimension respective  $(m \times n)$ ,  $(n \times 1)$  et  $(m \times 1)$  est un vecteur aléatoire sur un espace de probabilité  $(\omega, F, P)$  et  $T_1$  est un polyèdre convexe déterministe, par exemple :  $T_1 = \{x/x \geq 0, A_1 x \leq b\}$ .

Nous supposons que les contraintes sont fournies sous forme d'inégalités (introduction de variables d'écart) mais ceci sans aucune restriction.

Le symbole "" signifie qu'il s'agit d'une optimisation imprécise du point de vue mathématique.

## 2.4.8 Les différentes approches

On distingue deux approches de la programmation linéaire stochastique :

- L'approche passive ou "wait and see"
- L'approche active ou "here and now"

### L'approche passive ou "wait and see"

Dans cette approche le décideur peut attendre la réalisation des variables aléatoires et résoudre le programme déterministe résultant. Dans ce cas on s'intéresse généralement à la distribution de probabilité de la valeur optimale ou son espérance mathématique (et/ou) sa variance.

### L'approche active ou "here and now"

Cette approche est basée sur la décision sur  $x$  ou stratégie sur  $x$  qui est prise à l'avance avant la réalisation des variables aléatoires.

### Critère d'optimisation du problème équivalent

#### Cas des objectifs aléatoires

Plusieurs façons de définir la fonction objective du problème équivalent peuvent être considérées.

Soit :  $T = \{x \in \mathbb{R}^n \mid A(x) \leq b, x \geq 0\}$ .

#### o Le critère de l'espérance mathématique (E-modèle) ou critère de Bayes

Ce critère consiste à remplacer la variable aléatoire de l'objectif par son espérance mathématique

$$\begin{cases} \min E(Z(\omega)) \\ x \in T \end{cases} \quad (2.50)$$

#### o Le critère de la variance (V-modèle)

Ce critère consiste à minimiser la variance de l'objectif comme suit :

$$\begin{cases} \min \sigma^2(Z(\omega)) = \min x'Vx \\ x \in T \end{cases} \quad (2.51)$$

Avec  $V$  est la matrice de covariance du vecteur aléatoire  $c(\omega)$ .

#### o Le critère espérance-variance (E-V modèle)

Ce modèle consiste à minimiser la variance de l'objectif  $Z(\omega)$  tout en réalisant un niveau de rendement minimum  $Z_0$  fixé préalablement par le décideur.

$$\begin{cases} \sigma^2(Z(\omega)) \\ x \in T \\ E(Z(\omega)) \geq Z_0 \end{cases} \quad (2.52)$$

Le problème est de choisir  $Z_0$  convenable.

**o Le critère de risque minimal (P-modèle)**

Ce critère est basé sur la maximisation de la probabilité que la valeur de l'objectif est au moins égale à un certain niveau  $u$  choisi par le décideur, où ce paramètre  $u$  à fixer, correspond à un niveau de risque à respecter.

Le problème est le suivant :

$$\begin{cases} \max P(c'(\omega)) \\ x \in T \end{cases} \quad (2.53)$$

La solution de ce problème, dans le cas gaussien est donnée par le programme fractionnel suivant :

$$\begin{cases} \max \left( \frac{-\bar{c}'x + u}{\sqrt{x'Vx}} \right) \\ x \in T \end{cases} \quad (2.54)$$

$\bar{c}$  : Représente l'espérance mathématique du vecteur aléatoire  $c(\omega)$ .

$V$  : La matrice covariance du vecteur aléatoire  $c(\omega)$ .

$x'Vx$  : La variance de l'objectif  $c'(\omega)x$ .

**o Le critère de Katoka [8]**

Supposons que  $\alpha \in ]0, 1[$  donnée, Soit l'interprétation

$$\begin{cases} \min u \\ P(c'(\omega)x \leq u) = \alpha \\ x \in T \end{cases} \quad (2.55)$$

Dans le cas gaussien on a :

$$\begin{aligned} P(\omega/c'(\omega)x \leq u) &= P\left\{ \omega/c'(\omega)x - c'x/\sqrt{x'Vx} \leq u - c'x/\sqrt{x'Vx} \right\} = \\ &\phi\left(u - c'x/\sqrt{x'Vx}\right) = \alpha \\ \Rightarrow \left(u - c'x/\sqrt{x'Vx}\right) &= \phi^{-1}(\alpha) \Rightarrow u = c'x + \phi^{-1}(\alpha)\sqrt{x'Vx} \end{aligned}$$

$\phi$  : est la fonction de répartition de la variable aléatoire normale centrée réduite.

Par conséquent résoudre le problème (2.55) revient, dans ce cas gaussien, à résoudre le problème suivant :

$$\begin{cases} \min \left( c'x + \phi^{-1}(\alpha)\sqrt{x'Vx} \right) \\ x \in T \end{cases} \quad (2.56)$$

$\min \left( c^{-1}x + \phi^{-1}(\alpha)\sqrt{x'Vx} \right)$  est convexe si  $\phi^{-1}(\alpha) \geq 0 \iff \alpha \geq \frac{1}{2}$  ce qui revient à dire, si on revient au problème (2.56) si  $P\left(\frac{\omega}{c'(\omega)x} \leq u\right) = \alpha \geq \frac{1}{2}$ ; ; avoir le minimum de perte avec une probabilité supérieure ou égale à  $\frac{1}{2}$  si  $\alpha = 1$  on revient au E-modèle).

### Cas de contraintes aléatoires

Dans cette partie nous supposons que l'objectif est déterministe ou qu'il a été rendu déterministe en appliquant l'un des critères précédents. Mais les contraintes sont stochastiques  $A(\omega)x \leq b(\omega)$ .

La première méthode utilisée pour la résolution d'un tel programme linéaire stochastique consiste à remplacer chacune des variables aléatoires des contraintes par leurs espérances mathématiques respectives et résoudre le programme déterministe résultant.

#### Exemple

Soit le programme stochastique suivant :

$$\begin{cases} \min 2x_1 + x_2 \\ b_1(\omega)x_1 + x_2 \geq 6 \\ b_2(\omega)x_1 + x_2 \geq 4 \\ x_1 \geq 0, \quad x_2 \geq 0 \end{cases} \quad (2.57)$$

Où  $b_1(\omega) \in [2, 4]$  et  $b_2(\omega) \in [1, 3]$  suivant des lois uniformes indépendante sur leur intervalles. En remplaçant  $b_1(\omega)$  et  $b_2(\omega)$  par leur espérance mathématique avec :

$$E(b_1(\omega)) = \frac{2+4}{2} = 3 \text{ et } E(b_2(\omega)) = \frac{1+3}{2} = 2$$

Le problème déterministe équivalent est :

$$\begin{cases} \min 2x_1 + x_2 \\ 3x_1 + x_2 \geq 6 \\ 2x_1 + x_2 \geq 4 \\ x_1 \geq 0, \quad x_2 \geq 0 \end{cases} \quad (2.58)$$

La solution optimale est :  $x_1^* = 2$ ,  $x_2^* = 0$  et  $Z^* = 4$ .

Cherchons la probabilité pour que cette solution soit réalisable :

$$P(b_2(\omega)x_1^* + x_2^* \geq 6) \cdot P(b_2(\omega)x_1^* + x_2^* \geq 4) = P(b_1(\omega) \geq 3) \cdot P(b_2(\omega) \geq 2)$$

$$= (1 - F_{b_1}(3)) \cdot (1 - F_{b_2}(2)) = \frac{1}{4}$$

Où  $F_{b_1}$  et  $F_{b_2}$  sont les fonctions de répartitions des variables aléatoires respectives  $b_1(\omega)$  et  $b_2(\omega)$ .

La probabilité pour que cette solution soit réalisable est donc faible, ce qui montre le manque de réalisme d'une telle méthode.

Donc pour la résolution d'un programme linéaire stochastique, on utilise deux modèles essentiels : Modèles avec seuil de probabilités sur les contraintes (« chance constrained programming ») Modèle avec recours.

# Chapitre 3

## *Les modèles utilisés dans les techniques bancaires*

### Introduction

Les modèles mathématiques jouent un rôle crucial dans le secteur bancaire en offrant des outils puissants pour la prise de décisions éclairées. Ces modèles permettent aux institutions financières de mieux comprendre et analyser les risques, de prévoir les tendances du marché, d'optimiser les opérations et de gérer efficacement les ressources. Leur utilisation contribue à renforcer la performance des banques et à garantir une prise de décision stratégique. En outre, ces modèles offrent une plus grande transparence et fiabilité dans les projets d'investissement et les opérations financières.

#### **Définition 3.1 :**

Les modèles sont des outils mathématiques qui permettent aux banques d'optimiser leurs opérations et de prendre des décisions plus éclairées.

#### **Quelques exemples de modèles utilisés dans les techniques bancaires :**

##### 1. Modèles de gestion des risques :

● **La gestion des risques** est le processus qui permet d'identifier et d'évaluer les risques en vue d'élaborer un plan visant à minimiser et à maîtriser ces risques et leurs conséquences potentielles pour une entreprise. Les risques représentent une probabilité de perte ou de dommage.

**a. Modèles de Value at Risk (Var) :** Ces modèles estiment la perte maximale qu'une banque peut subir sur un horizon donné avec un certain niveau de confiance. Ils sont utilisés pour la gestion du risque de marché et la prise de décisions d'investissement.

**b. Modèles de Monte Carlo :** Ces modèles simulent des scénarios aléatoires pour évaluer le risque de marché et le rendement des portefeuilles d'investissement.

##### 2. Modèles d'optimisation :

**a. Modèles de programmation linéaire :** Ces modèles permettent d'optimiser l'allocation des ressources dans un contexte de contraintes. Ils sont utilisés pour la gestion des portefeuilles d'actifs et la planification financière.

**b. Modèles d'arbres de décision** Ces modèles permettent d'analyser les différentes options et de prendre la meilleure décision possible dans un contexte d'incertitude

### 3. Modèles d'intelligence artificielle :

**a. Modèles de robo-advisors :** (Un robo-advisor : est une plateforme en ligne de conseil en investissement financier) Ces modèles permettent aux clients de gérer leurs finances de manière automatisée.

**b. Modèles d'analyse de sentiment :** Ces modèles analysent les opinions des clients sur les réseaux sociaux et d'améliorer leurs services en conséquence.

## 3.1 Le modèle d'arbre de décision

En théorie des graphes, un arbre est un graphe acyclique et connexe. Sa forme évoque en effet la ramification des branches d'un arbre.

### Qu'est-ce qu'un arbre de décisions ?

Un arbre de décision est une représentation visuelle d'un algorithme de classification de données suivant différents critères qu'on appellera décisions (ou nœuds).

Voici un exemple :

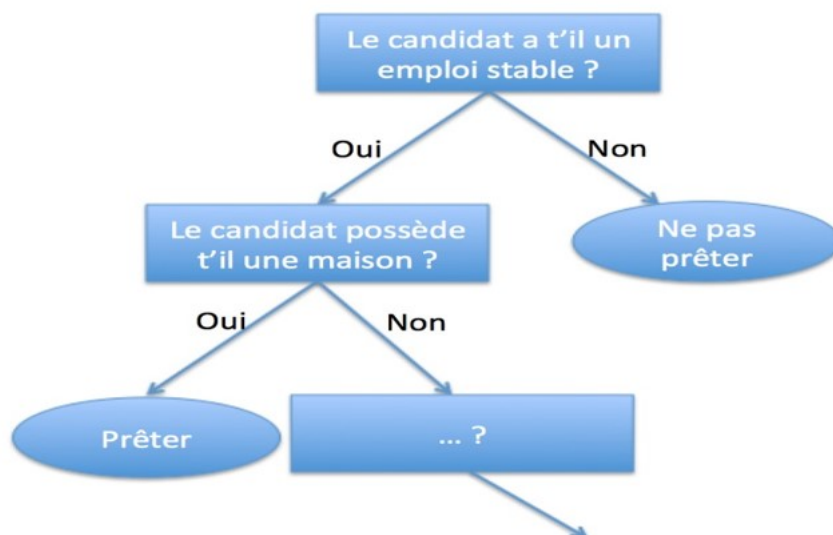


FIGURE 3.1 – Exemple d'un arbre de décision pour l'accord d'un prêt

Cet arbre de décision permet en fonction de quelques questions de déterminer si une banque doit prêter ou pas au client qui demande un prêt.

Ce dernier est très simplifié mais la plupart des banques utilisent des systèmes similaires permettant aux agents une prise de décision experte.

- Un nœud de décision (représenté par un rectangle) (on pose une question)
- Un bloc de fin qui est représenté par une feuille. (On a trouvé la salle)

## 3.2 Le problème du Sac à Dos (Knapsack Problem)

### 3.2.1 Description

Le problème du sac à dos, aussi noté KP (en anglais, Knapsack Problem) est un problème d'optimisation combinatoire. Il modélise une situation analogue au remplissage

d'un sac à dos, ne pouvant supporter plus d'un certain poids, avec tout ou partie d'un ensemble d'objets ayant chacun un poids et une valeur. Les objets mis dans le sac à dos doivent maximiser la valeur totale, sans dépasser le poids maximum.

### Exemple

Pour justifier son nom, le problème se pose lorsqu'un randonneur au moment de préparer son périple est confronté au problème de la capacité limitée de son sac à dos. Il lui faut donc trancher pour prendre les choses dont il a le plus besoin (maximiser la valeur) sans dépasser la capacité du sac à dos (15 kg).



### 3.2.2 Formulation

Etant donné un ensemble de  $n$  objets chacun ayant un certain poids  $a_j$  et une certaine valeur (coût)  $c_j$ , et soit  $b$  un réel qui représente la charge (poids, volume, capacité) maximale que l'on peut emporter dans un sac à dos. La formulation du problème conduit à un PLNE 0-1 à une seule contrainte :

$$\begin{cases} \max Z = \sum_{j=1}^n c_j x_j \\ \sum_{j=1}^n a_j x_j \leq b, x_j = \text{signifiant que l'objet } j \text{ est choisi } x_j \in \{0, 1\} \end{cases}$$

### 3.2.3 Applications

Le problème du sac à dos est utilisé pour modéliser diverses situations, quelquefois en tant que sous problème :

- La sélection d'investissement « Capital Budgeting Problem » de manière à maximiser le rendement, sans bien sûr, dépasser la somme disponible.
- Dans le chargement de bateau ou d'avion « Cargo Loading Problem » : tous les bagages à destination doivent être amenés, sans être en surcharge ;
- Dans la découpe de matériaux « Cutting Stock Problem » : pour minimiser les chutes lors de la découpe de sections de longueurs diverses dans des barres en fer.
- Dans la cryptographie ; plusieurs algorithmes de chiffrement asymétrique sont basés sur le problème du sac à dos mais ils ont tous été cassés à ce jour.

### 3.2.4 Variantes

Le problème du sac à dos possède une multitude de variantes dont voici quelques-unes : **Knapsack entier**

S'il existe plusieurs objets de chaque type ; soit  $N_j$  le nombre d'objets de type  $j$ , on obtient un PLNE avec  $x_j \in \{0, 1, \dots, N_j\}$  (ou  $x_j$  entier si  $N_j < \infty$ ) Le problème ainsi obtenu est appelé Knapsack entier

#### Knapsack multi-dimensionnel

On considère ici que le sac à dos à  $d$  dimensions, avec  $d > 0$  (d-KP). Par exemple, on peut imaginer une boîte. Chaque objet a trois dimensions, et il ne faut pas déborder sur aucune des dimensions. La contrainte est alors remplacée par  $d$  contraintes :  $\sum_{j=1}^n a_j x_j^k \leq b^k, \forall k \in \{1, \dots, d\}$

En pratique, la version multidimensionnelle peut servir à modéliser et résoudre le problème du remplissage d'un container dont le volume et la charge maximale sont limités.

#### Knapsack quadratique

Le problème de sac à dos quadratique est noté QKP. On a ici un gain  $g_{ij}$  supplémentaire lorsque deux objets ( $i$  et  $j$ ) sont pris simultanément. La fonction objective s'écrit alors :

$$\max Z = \sum_{j=1}^n c_j x_j + \sum_{i=1}^n \sum_{j=1}^n g_{ij} x_i x_j$$

### 3.3 Modèle de Markovitz

#### Constitution d'un portefeuille de titre

On suppose que l'on a sélectionné sur les  $n$  titres  $T_i, i = 1..n$ .

Chaque titre a un rendement  $r_i$ .  $r_i$  est une variable aléatoire dont la moyenne et la variance  $V(r_i)$  ont été déterminées par une étude statistique. On sait aussi que les  $r_i$  sont des v.a indépendantes les unes des autres.

On veut constituer un porte-feuille de titres en prenant une proportion de chacun de ces titres. Par exemple supposons 3 titres  $T_1, T_2, T_3$ .

Dans le porte-feuille on peut décider de mettre  $\frac{1}{4}$  de  $T_1, \frac{1}{4}$  de  $T_2, \frac{1}{2}$  de  $T_3$ . Cela signifie que si l'on investit par exemple 1000 Euros dans le porte-feuille, on achètera 250 Euros de  $T_1, 250$  Euros de  $T_2$  et 500 Euros de  $T_3$ . Ce que rapporte le portefeuille est alors  $250r_1 + 250r_2 + 500r_3 = 1000 * (\frac{1}{4}r_1 + \frac{1}{4}r_2 + \frac{1}{2}r_3)$ . En désignant par  $R$  le rendement du porte-feuille, on peut poser  $R = \frac{1}{4}r_1 + \frac{1}{4}r_2 + \frac{1}{2}r_3$ .

Le rendement du porte-feuille est lui aussi une variable aléatoire composée d'une combinaison linéaire des v.a.  $r_i$ . Par exemple le rendement du porte-feuille précédent est donné par  $R = \frac{1}{4}r_1 + \frac{1}{4}r_2 + \frac{1}{2}r_3$ .

L'enjeu est de constituer un porte-feuille optimal vis-à-vis du risque, mais garantissant un rendement moyen  $p$  fixé. Le risque est mesuré par la variance du porte-feuille. Rappelons que la variance est la moyenne des écarts (au carré) par rapport à la valeur moyenne. Donc plus la variance est petite et plus le porte-feuille a des chances de rapporter son rendement moyen.

On introduit les variables  $x_i, i = 1..n$  représentant la proportion de titre  $T_i$  choisie ( $0 < x_i < 1$ ) rendement du porte-feuille est donc  $R = \sum_{i=1}^n x_i r_i$ . La valeur moyenne du rendement du porte-feuille est  $E(R) = \sum_{i=1}^n x_i E(r_i)$  et sa variance est  $V(R) = \sum_{i=1}^n (x_i)^2 V(r_i)$ , les  $r_i$  étant des v.a. indépendantes deux à deux il n'y a pas de terme de la forme  $x_i x_j$ .

Le problème que l'on a à résoudre est donc :

$$\text{Minimiser } f(x) = \sum_{i=1}^n (x_i)^2 V(r_i) \\ \text{sous les contraintes}$$

$$\begin{aligned}
 -\sum_{i=1}^n x_i E(r_i) + p &\leq 0 & (1) \\
 -\sum_{i=1}^n x_i + 1 &= 0 & (2) \\
 -x_i &\leq 0 \text{ pour } i = 1, \dots, n & (3)
 \end{aligned}$$

On note S l'ensemble des solutions des conditions (1), (2), (3).

### 3.4 Choix d'emprunts :

#### 3.4.1 Problème

M. Chebbah, Le Directeur d'une chaine de cafétéria, désire ouvrir trois nouvelles cafétéria : une à Tizi Ouzou (C.T), une à Azazga (C.A) et une à DBK (C.D). L'ouverture de chaque nouvelle cafétéria lui coutera respectivement 2,8 millions, 1,2 million et 1,8 million d'euro pour financer ses projets, il fait appel à trois différentes banques.

	<b>C.T</b>	<b>C.A</b>	<b>C.D</b>
<b>CNEP Banque TO</b>	5%	6,5%	6,1%
<b>CNEP Banque AZAZGA</b>	5,2%	6,2%	6,2%
<b>CNEP Banque DBK</b>	5,5%	5,8%	6,5%

FIGURE 3.2 – Tableau 3.1 – taux proposés par les banques pour les différent projets

En fonction de l'emplacement de ces cafeterias et des risques évalués, chaque banque décide de financer au plus 3 millions sur 8 ans et propose des taux différents suivant les boutiques (tableau 3.1). Déterminer le montant à emprunter à chacune des banques pour financer chaque boutique de façon à minimiser les dépenses totales de M. Chebbah.

#### 3.4.2 Modélisation

Notons  $x_{ij}$  montant emprunté à la banque  $i$  pour financer la boutique  $j$ . Soit  $C_a$  et  $B_a$  respectivement le nombre de cafétéria et le nombre de banques acceptant de les financer,  $n$  le nombre d'années sur lequel s'étend le remboursement,  $M_{\max}$  le montant maximal que chaque banque est prête à financer. Le modèle mathématique est le suivant :

$$\begin{aligned}
 (1) \quad \min & \sum_{i=1}^{B_a} \sum_{j=1}^{C_a} x_{ij} \frac{T_{ij}}{1-(1+T_{ij})^{-n}} \\
 (2) \quad \forall j = 1 \dots C_a & : \sum_{i=1}^{B_a} x_{ij} = P_j \\
 (3) \quad \forall j = 1 \dots B_a & \sum_{i=1}^{C_a} x_{ij} \leq M_{\max} \\
 (4) \quad x_{ij} & \geq 0
 \end{aligned}$$

L'objectif est de minimiser les dépenses de M. Chebbah, c'est à dire de minimiser la somme des annuités qu'il aura à verser. Si la somme  $x_{ij}$  est empruntée à la banque  $i$  pour

financer la cafeteria  $j$  au taux  $T_{ij}$  pendant  $n$  années, l'annuité que M.Chebbah aura à verser pour cette boutique à la banque  $i$  est donnée par la relation (5).

$$(5) \frac{T_{ij}}{1-(1+T_{ij})^{-n}}$$

Cette annuité est à payer pour toutes les cafétéria  $j$  à toutes les banques  $i$  pendant  $n$  années. Mais puisque ces annuités sont les mêmes chaque année, pour minimiser la somme des annuités sur les  $n$  années, il suffit de minimiser la somme des annuités d'une année. On obtient donc la fonction objectif (1). Chaque doit être entièrement financée. Donc les sommes empruntées pour financer chaque boutique  $j$  doit être égale à son montant  $P_j$ . Cette contrainte est traduite par (2). La contrainte (3) indique que chaque banque ne doit pas financer plus de  $M_{\max}$  et la contrainte (4) est la contrainte de positivité des variables.

## 3.5 Campagne publicitaire

### 3.5.1 Problème

La PME Pronuevo lance un nouveau produit sur un marché régional et désire en faire la campagne publicitaire sur plusieurs médias à la fois. Elle s'adresse à une société de publicité, Régional Pub, spécialisée dans ce type de campagne régionale et lui confie cette tâche pour un budget total de 250 000 Euro. Cette dernière connaît bien le marché et l'impact de diffusion d'une publicité dans une revue locale ou à la radio, ou d'un spot publicitaire télévisé sur la chaîne régionale. Elle prose de s'attaquer au marché pendant deux mois par six médias différents. Pour chacun des médias, elle connaît le coût de diffusion et le nombre de personnes sur lequel ce media a un impact. Un indice de qualité de perception de la campagne est également connu pour media.

La société impose aussi un nombre maximal d'utilisations de chaque média (par exemple, pas plus de 8 diffusion d'un spot télévisé). Le tableau 3.2 regroupe ces informations. La société qui lance le produit souhaite que l'impact de la campagne publicitaire atteigne au moins 100 000 personnes. Quels seront les médias choisis, et dans quelles proportions pour que l'indice de qualité de perception soit maximal ?

### 3.5.2 Modélisation

On note  $n$  le nombre de médias que l'on pourra utiliser pour cette campagne. Les constantes  $c_{a_i}, c_i$  désignent respectivement le nombre de clients potentiellement atteints, le coût unitaire d'utilisation, le nombre maximal d'utilisation et l'indice de qualité de chacun des médias utilisés. Les variables de décision  $m_i$  que l'on choisit seront le nombre d'utilisations de chaque media  $i$  pour cette campagne.

Avant tout, la contrainte budgétaire devra être respectée : la contrainte (2) indique que la somme des coûts relatifs à chaque média doit être inférieur au budget alloué  $B$ . Chaque média ne devra pas être utilise plus que le nombre maximal autorisé (contrainte (3)). La cible  $C$  (nombre minimal de clients) devra être atteinte, c'est à dire que la somme des nombres de clients potentiellement atteints pour chaque média doit être supérieure aux 100 000 individus visés (contrainte (4)). La fonction objectif (1) doit maximiser la somme des indices de qualité de chaque média utilisé. Les variables de décision  $m_i$  doivent être entières, contrainte (5).

$N^\circ$	Type de média	Clients potentiellement atteints	Coût d'utilisation unitaire	Utilisation maximale	Indice de qualité de perception
1	Journal hebdomadaire à diffusion gratuite	13 000	1 600	4 semaines	4
2	Journal mensuelle	1 600	8 100	2 mois	7
3	Journal hebdomadaire non gratuit	3 000	13 000	8 semaines	8
4	Séquence audio	7 000	10 000	60 diffusions sur 2 mois	3
5	Placard publicitaire	3 000	24 000	4 panneaux sur 2 mois	6
6	Séquence publicitaire	8 000	50 000	8 diffusions sur 2 mois	9

FIGURE 3.3 – Tableau 3.2 – Données pour la campagne publicitaire

$$\begin{aligned}
 (1) & \max \sum_{i=1}^n q_i m_i \\
 (2) & \sum_{i=1}^n c_i m_i \leq B \\
 (3) & \forall i = 1 \dots n : m_i \leq n d_i \\
 (4) & \sum_{i=1}^n c_a m_i \geq C \\
 (5) & \forall i = 1 \dots n : m_i \in \mathbb{N}
 \end{aligned}$$

### 3.6 Le problème d'affectation

Le problème d'affectation est un cas particulier de la programmation linéaire en variable binaires. Il consiste à effectuer  $m$  ressources (produits, personnes...etc.) à  $n$  activités (compartiments, tâches, etc...) tout en optimisant le cout total de l'affectation.

On peut formuler ce problème comme suit : Pour  $m = n$

$$\left\{ \begin{array}{l} z(x) = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \rightarrow (\min / \max) \\ \sum_{j=1}^n x_{ij} = 1, \quad i = 1, \dots, n \\ \sum_{i=1}^n x_{ij} = 1, \quad j = 1, \dots, n \\ \forall i, \forall j, x_{ij} \in \{0, 1\} \text{ ou } x_{ij} \geq 0 \end{array} \right.$$

Une solution réalisable pour le problème d'affectation dans ce cas est une matrice carrée de  $n^2$  éléments, où :

$$x_{ij} = \begin{cases} 1 & \text{si } i \text{ est associé à } j \text{ réalisé} \\ 0 & \text{sinon} \\ \forall i = 1, \dots, n \\ \forall j = 1, \dots, n \end{cases}$$

**Types d'affectation :**

— **Affectation en série :**

Exemple pratique de la chaîne de production : L'exécution des tâches est très influencée par l'élément (personne) de faible rendement à identifier avec son rendement  $P_{\min}$  ce qui se traduit par :

$$z = \min_i \left( \sum_{j=1}^n c_{ij} x_{ij} \right) = P_{\min} \rightarrow \max$$

Sous contrainte :

$$\begin{aligned} (2) \quad & \sum_{j=1}^n x_{ij} = 1 \quad \forall i = 1 \dots n \\ (3) \quad & \sum_{i=1}^n x_{ij} = 1 \quad \forall j = 1 \dots n \\ (4) \quad & x_{ij} \in \{0, 1\} \quad 1 \leq i \leq n, \quad 1 \leq j \leq n \end{aligned}$$

Pour résoudre ce problème on le reformule en problème linéaires suivant :

$$\begin{aligned} \sum_{j=1}^n c_{ij} x_{ij} &\geq P_{\min} \quad \forall i = 1..n \\ \sum_{j=1}^n x_{ij} &= 1 \quad \forall i = 1 \dots n \\ \sum_{i=1}^n x_{ij} &= 1 \quad \forall j = 1 \dots n \\ x_{ij} &\in \{0, 1\} \quad 1 \leq i \leq n, \quad 1 \leq j \leq n \\ P_{\min} &\rightarrow \max \end{aligned}$$

— **Affectation en parallèle :**

Le modèle mathématique pour ce cas est le suivant :

Pour le rendement

$$\begin{aligned} \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} &\rightarrow \max \\ \sum_{j=1}^n x_{ij} &= 1 \quad \forall i = 1, \dots, n \\ \sum_{i=1}^n x_{ij} &= 1 \quad \forall j = 1, \dots, n \\ x_{ij} &\in \{0, 1\} \\ 1 \leq i \leq n, \quad 1 \leq j \leq n \end{aligned}$$

## 3.7 Ventes avec affectation de personnel

### 3.7.1 Problème

La société line production décide de planifier la production de quatre de ses produits (**P1**, **P2**, **P3**, **P4**) sur ses cinq lignes de production. Elle prévoit un profit de 6 Euro pour les produits **P1** et **P4**, 9 Euro pour **P2** et 10 Euro pour le produit **P3**. Sur les cinq lignes de production, les durées maximales de travail pour la période considérée sont différentes. Les capacités maximales de travail pour les lignes **L1** a **L5** sont 4 500 h, 5 000 h, 4 500 h, 1500 h et 2 500 h. le **tableau 3.3** donne le temps de travail unitaire (en heures) nécessaire à la production d'une unité de chacun des produits sur chaque ligne de production. Quelles devront être les quantités à produire pour chaque catégorie **P1** a **P4** si on cherche à maximiser le gain total ?

Si, par la suite, on autorise un transfert de personnel (et donc d'heures de travail) d'une ligne a une autre pendant la période considérée, comme le prévoit le tableau 3.4, quel sera le gain maximal ? Combien d'heures seront transférées, et dans quelles conditions ?

Lignes	L1	L2	L3	L4	L5
P1	1,2	0,8	2,0	0,2	0,8
P2	1,7	1,7	1,5	0,5	1,1
P3	1,3	1,3	1,2	1,0	1,5
P4	0,8	1,2	1,0	0,8	1,0

FIGURE 3.4 – Tableau 3.3 – Temps unitaire de fabrication

Destination Origine	L1	L2	L3	L4	L5	Nombre maximal d'heures <u>transférables</u>
L1	-	Oui	Oui	Oui	Non	400
L2	Non	-	Oui	Non	Oui	800
L3	Oui	Oui	-	Oui	Non	500
L4	Non	Non	Non	-	Oui	200
L5	Oui	Oui	Oui	Non	-	300

FIGURE 3.5 – Tableau 3.4 – Transferts possibles de personnel

### 3.7.2 Modélisation

Le nombre de produits est noté  $n$ , le nombre de lignes  $L$ . Le gain du produit  $i$  est noté  $g_i$  durée de traitement du produit  $i$  sur la ligne  $j$ ,  $d_{ij}$  et le nombre maximal d'heures de travail de la ligne  $j$ ,  $m_j$ . Comme toujours en planification de production, une variable  $q_i$  désignera la quantité à produire pour le produit  $i$ . La modélisation correspondant à la première question est très simple. L'objectif est de maximiser les gains (ligne (1)) tout en respectant la capacité maximale de travail de chaque ligne de production (contrainte (2)). On a donc le modèle simple suivant :

$$\begin{aligned}
 (1) \quad & \max \sum_{i=1}^n g_i \cdot q_i \\
 (2) \quad & \forall l = 1, \dots, L : \sum_{i=1}^n d_{ij} \cdot q_i \leq m_l \\
 (3) \quad & \forall i = 1, \dots, n : q_i \geq 0
 \end{aligned}$$

Si maintenant on peut transférer du personnel d'une ligne a une autre, des variables  $h_i$  sont nécessaires. Elles correspondent aux heures de travail effectuées sur la ligne de production  $i$ . Les variables  $t_{ij}$  représentent le nombre d'heures de travail transférées de la ligne  $i$  à la ligne  $j$ . La contrainte (2') remplace la contrainte (2) du précédent modèle et spécifie que le nombre d'heures travaillées est égale a la production fois la durée unitaire de travail. La contrainte (4) impose l'équilibre entre les heures de travail effectuées sur la ligne, les heures transférées et le nombre d'heures maximal de travail sur une ligne. On travaille  $h_i$  heures équivalant au nombre d'heures disponibles pour cette ligne, augmentée des heures qui sont transférées depuis d'autres lignes et diminue des heures transférées vers d'autres lignes ( $T_{ij}$  vaut 1 si on peut transférer des heures de la ligne  $i$  à la ligne  $j$ , 0 sinon). La dernière contrainte (contrainte (5)) est celle qui permet de ne pas transférer plus d'heures que le maximum  $tm_j$  autorisé.

$$\begin{aligned}
 (1) \quad & \max \sum_{i=1}^n g_i q_i \\
 (2') \quad & \forall l = 1, \dots, L : \sum_{i=1}^n d_{il} q_i \leq h_l
 \end{aligned}$$

$$\begin{aligned}
 (3) \quad & \forall i = 1 \dots n : q_i \geq 0 \\
 (4) \quad & \forall j = 1 \dots L : h_j - \sum_{l=1, T_{lj}=1}^L t_{lj} + \sum_{l=1, T_{jl}=1}^L t_{jl} = m_j \\
 (5) \quad & \forall j = 1 \dots L : \sum_{l=1, T_{jl}=1}^L t_{jl} \leq tm_j \\
 (6) \quad & \forall j = 1 \dots L, \forall l = 1 \dots L : t_{jl} \geq 0 \\
 (7) \quad & \forall j = 1 \dots L : h_j \geq 0
 \end{aligned}$$

### 3.8 Problème d'ordonnancement [18] [19] [20]

**Définition 3.7** Résoudre un problème d'ordonnancement consiste à organiser la réalisation de tâches dans le temps : déterminer leurs dates d'exécution et l'ordre dans lequel elles sont traitées sur l'ensemble des ressources disponibles.

**Les données d'un problème d'ordonnancement :** Les différentes données d'un problème d'ordonnancement sont les tâches, les ressources, les contraintes et les critères. Ainsi, étant donné un ensemble de tâches et un ensemble de ressources, il s'agit de programmer les tâches et affecter les ressources de façon à optimiser un ou plusieurs objectifs (un objectif correspondant à un critère de performance), en respectant un ensemble de contraintes.

**Les tâches :** Une tâche est une entité élémentaire de travail localisée dans le temps par une date de début et une date de fin d'exécution et qui consomme des ressources avec des quantités déterminées. Un coût (ou poids) est attribué à une tâche pour estimer sa priorité, son degré d'urgence, ou son coût d'immobilisation dans le système. On distingue deux types de tâches :

- les tâches morcelables (préemptives) qui peuvent être exécutées en plusieurs fois, facilitant ainsi la résolution de certains problèmes

- les tâches non morcelables (indivisibles) qui doivent être exécutées en une seule fois et ne sont interrompues qu'une fois terminées.

On note en général  $N = \{J_1, J_2, \dots, J_n\}$  l'ensemble des tâches, chaque tâche est caractérisée par :

- La durée opératoire de la tâche  $i$  sur la machine  $j$  :  $(p_{ij})$
- La date de disponibilité de la tâche  $i$  :  $(r_i)$
- La date de début d'exécution de la tâche  $i$  :  $(s_i)$
- La date de fin d'exécution de la tâche  $i$  :  $(C_i)$
- La date d'achèvement souhaitée de la tâche  $i$  :  $(d_i)$
- Le retard de la tâche  $i$  :  $(T_i = \max(C_i - d_i, 0))$

#### 3.8.1 Organisation de la fabrication : Méthode PERT [18]

##### 1. Généralités :

La réalisation d'un projet nécessite souvent une succession de tâches auxquelles s'attachent certaines contraintes

- De temps : délais à respecter pour l'exécution des tâches ;
- D'antériorité : certaines tâches doivent être exécutées avant d'autres ;

- De simultanéité : certaines tâches peuvent être réalisées en même temps ;
- De production : temps d'occupation du matériel ou des hommes qui l'utilisent.

Pour présenter ces problèmes d'ordonnancement, on peut utiliser la méthode PERT (Program Evaluation Research Task) qui consiste à mettre en ordre sous la forme d'un graphe, plusieurs tâches qui grâce à leur dépendance et à leur chronologie concourent toutes à la réalisation d'un projet. Cet outil a été créé en 1957 pour l'US Navy ; et permet de calculer le meilleur temps de réalisation d'un projet et d'établir le planning correspondant.

## 2. Exemples de constructions des hangars

On devait déterminer la durée maximale des travaux nécessaires à la construction des hangars

Les tâches	La durée des tâches évaluée en jours
<b>A</b> : étude, réalisation et acceptation des plans	4
<b>B</b> : préparation du terrain	2
<b>C</b> : commande de matériaux (bois, briques, ciment, tôle, pour le toit)	1
<b>D</b> : creusage des fondations	1
<b>E</b> : commande portes, fenêtres	2
<b>F</b> : livraison des matériaux	2
<b>G</b> : coulage des fondations	2
<b>H</b> : livraison portes, fenêtres	10
<b>I</b> : construction des murs, du toit	4
<b>J</b> : mise en place portes et fenêtres	1

FIGURE 3.6 – Tableau de la durée des tâches évaluées en jours

En répondant aux questions suivantes :

- Quelle(s) tâche (s) doit être terminée immédiatement avant qu'une autre ne commence ?
- Quelle tâche doit suivre une tâche déterminée ?

On obtient le tableau suivant

Tâches antérieures obligatoirement terminées	Tâches	Tâches suivantes
-	A	C, E
-	B	D
A	C	F
A,B	D	G
A	E	H
C	F	G
D,F	G	I
F	H	J
G	I	J
H,I	J	-

### Pour tracer le réseau PERT :

Un réseau est constitué par des étapes et des tâches A, B, C, D...

Le code de présentation est le suivant :

- On symbolise une étape par un cercle.
- Un arc fléché pour signifier la tâche.

Pour représenter un réseau PERT, il existe des règles :

- Chaque tâche est représentée par un arc et un seul.
- Deux tâches ne peuvent être identifiées par deux arcs ayant la même origine et la même

- Extrémité.

Les tâches peuvent être :

- Successives : elles se déroulent les unes après les autres ; séparées par des étapes.
- Simultanés : elles se déroulent en même temps.
- Convergentes : elles aboutissent à une même étape

Voilà le tracé du réseau PERT

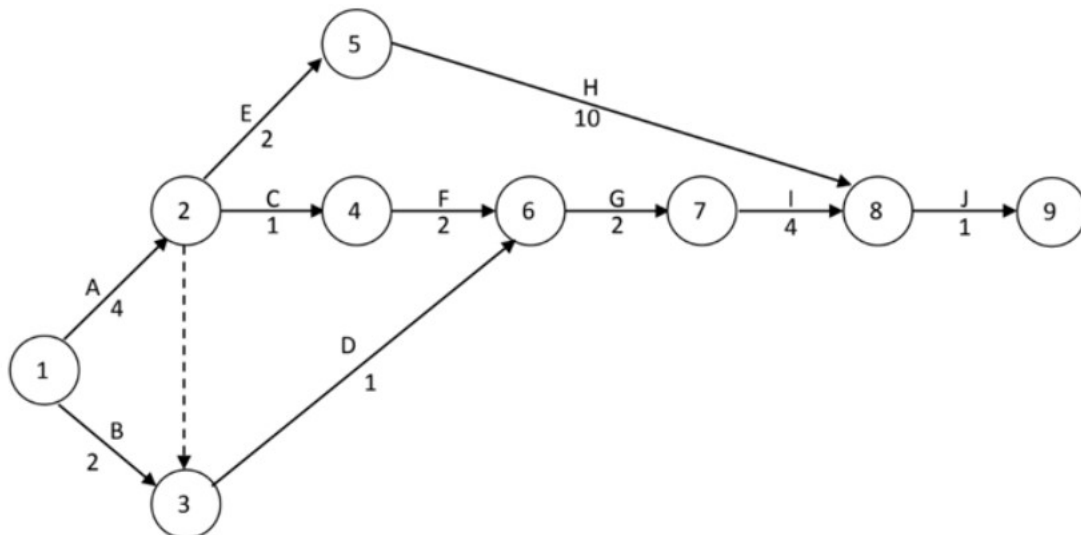


FIGURE 3.7 – Réseau PERT

**Marge total :**

La marge totale indique le retard maximal que l'on peut admettre dans sa réalisation, sans allonger la durée optimale du projet. Elle se calcule comme suit : la différence entre la date au plus tard et la date au plus tôt de la tâche en question.

$MT_i$  : La marge totale de la tâche avec  $MT_i = T_i^* - T_i$

**Marge libre :** On note :

$ML_i$  : La marge libre de la tâche  $i$

$T_j$  : La date au plus tôt de la tâche  $j$

$ML_i = T_j - T_i - d_i$

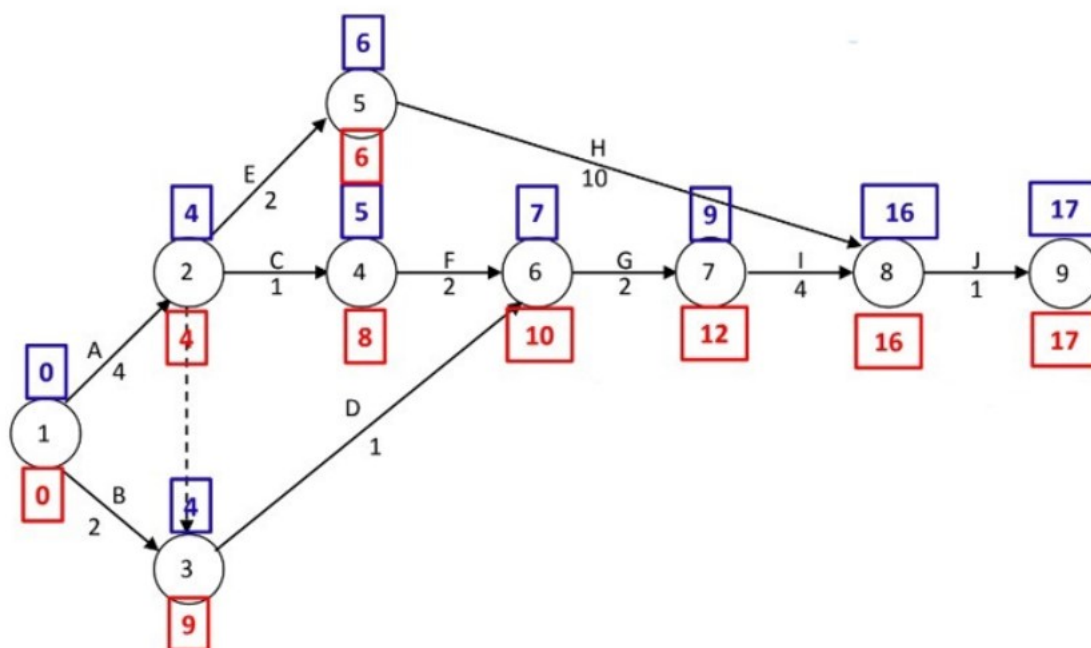
**Détermination du chemin critique :**

Pour déterminer le chemin critique on calcule les marges libres et les marges totales

Chemin critique : les marges libres = 0 et les marges totales = 0

Taches	A	B	C	D	E	F	G	H	I	J
Marges libres	4-0- 4=0	4-0- 2=2	5-4- 1=0	7-4- 1=2	6-4- 2=0	7-5- 2=0	9-7- 2=0	16-6- 10=0	16-9- 4=3	17-16- 1=0

Taches	A	B	C	D	E	F	G	H	I	J
Marges totales	4-0- 4=0	9-0- 2=7	8-4- 1=3	10-4- 1=5	6-4- 2=0	10-5- 2=3	12-7- 2=3	16-6- 10=0	16-9- 1=6	17-16- 1=0



Le chemin critique est : (A-E-H-J)

# Chapitre 4

## *Les modèles, modèles complémentaires et résolutions pour la problématique*

### Introduction

Les banques font face à une multitude de défis complexes, pour relever ces défis, elles s'appuient sur une variété de modèles d'optimisation qui permettent de prendre des décisions éclairées et d'optimiser leurs opérations. Ce chapitre se concentrera sur des modèles couramment utilisés dans le secteur bancaire

### 4.1 Méthode de résolution des modèles d'ordonnement

#### **Définition :**

Le modèle d'ordonnement consiste à déterminer l'ordre optimal dans lequel un ensemble de tâches doit être exécuté. Il s'agit d'un problème crucial dans de nombreux domaines bancaires, tel que la gestion des files d'attente, la planification des transactions et la gestion des portefeuilles de crédit.

Il existe plusieurs méthodes de résolution pour le problème d'ordonnement, telles que les algorithmes (Bellman), (Dijkstra) et (Ford) : pour appliquer l'algorithme de Bellman le graphe doit être sans circuit et pour l'algorithme de Dijkstra les distance doivent être positives, on s'intéresse au cas général qui est l'algorithme de (Ford).

#### 4.1.1 Algorithmes basés sur la théorie des graphes : Algorithme de Ford [18] [19] [20]

On applique l'algorithme général pour la recherche d'un plus court chemin sur un réseau quelconque avec  $d(u) \in \mathbb{R}$

Cet algorithme permet soit :

De mettre en évidence un circuit absorbant si celui-ci existe.

De déterminer une arborescence des plus courts chemins de racine  $e$  dans un réseau s'il ne contient pas de circuit absorbant.

#### **Présentation de l'algorithme**

Soit le graphe (réseau)  $R = (X, U, d)$  d'entrée  $e$  et de sortie  $s$ ;  $d : U \rightarrow \mathbb{R}$ .

Pour un réseau  $R$  quelconque.

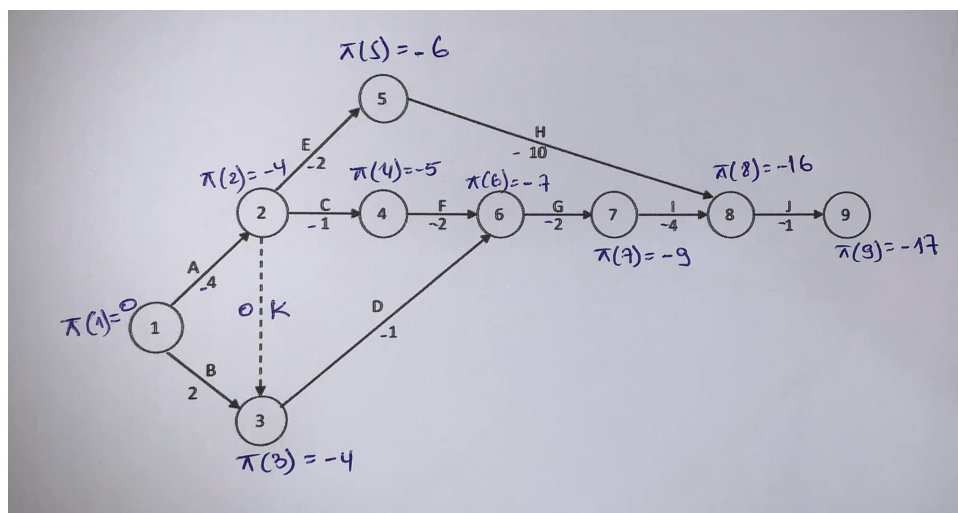
Extraire de  $R$  une arborescence de racine  $e$  réalisable

- Si  $e$  n'est pas racine, terminé.
- Si  $e$  est racine de  $R$ , soit  $A$  l'arborescence exhibée, notons  $\pi(x)$  longueur du plus court chemin de  $e$  à  $x$  dans  $(X, A)$  (l'arborescence)
  1. Chercher un arc  $u = (i, j) / u \notin A$  à l'arborescence
 
$$\delta(u) = \pi(j) - \pi(i) - d(i, j) > 0$$
    - Si un tel arc n'existe pas, terminé,  $(X, A)$  est optimale.
    - Si un tel arc existe; aller à 2.
  2. Tester si  $(X, A \cup \{u\})$  contient un circuit,
    - Si oui; examiner si ce circuit est absorbant (s'il l'est; terminée. Le problème n'admet pas de solution).
    - Sinon aller en 3.
  3. Chercher un arc  $v \in A$  tel que  $T(v) = j = T(u)$ 
    - On pose :  $A = A \cup \{u\} \setminus \{v\}$
    - Soit  $X' = \{j\} \cup \{\text{descendants de } j \text{ dans l'arborescence } A\}$
    - On pose :  $\pi(y) = \pi(y) - \delta(u) \quad \forall y \in X'$  aller en 1.

**Exemple 4.1 :**

On reprend le modèle de la construction des hangars :

On exhibe l'arborescence suivante :A-E-H-G-C-F-J-K



On calcule les plus court distance :

- $\pi(1) = 0$
- $\pi(2) = -4$
- $\pi(5) = -6$
- $\pi(8) = -16$
- $\pi(9) = -17$
- $\pi(3) = -5$
- $\pi(4) = -5$
- $\pi(6) = -7$
- $\pi(7) = -9$

L'arborescence est A-E-H-G-C-F-J-K (à partir du graphe)

$$\delta(1, 3) = \pi(3) - \pi(1) - d(1, 3) = -4 - 0 - (-2) = -2$$

$$\delta(3, 6) = \pi(6) - \pi(3) - d(3, 6) = -7 - (-4) - (-1) = -2$$

$$\delta(7, 8) = \pi(8) - \pi(7) - d(7, 8) = -16 - (-9) - (-4) = -3$$

Il n'existe aucun arc  $u$  n'appartenant pas à l'arborescence tel  $\delta(u) > 0$ , d'où l'arborescence déjà trouvée est optimale.

### 4.1.2 Modèle mathématique (Modèle de PERT)

Premièrement on doit construire le modèle linéaire qui cherche à calculer la durée minimale du projet en l'absence de toute compression. Le modèle comprend les variables de décision suivantes :

Soit une tâche  $t$  et son successeur immédiat  $t'$ .

$D(t)$  : Instant où débute la tâche  $t$  .

$D(t')$  : Instant où débute la tâche  $t'$ .

$d_t$  : Durée de la tâche  $t$ .

$d(p)$  : Durée de projet.

L'objectif consiste à minimiser la durée totale du projet :

$$\min(z) = d(p) \quad (d(p) \rightarrow \min)$$

Les contraintes du modèle se regroupent en deux catégories :

**1er cas :**

- Si  $t$  est prédécesseur immédiate de  $t'$ , alors  $t'$  ne peut démarer avant la réalisation de  $t$  :

$$D(t') \geq D(t) + d_t \quad (4.1)$$

**2ème cas :**

- Si  $t$  n'admet aucun successeur, alors :

$$d(p) \geq D(t) + d_t \quad (4.2)$$

Le modèle linéaire correspondant est :

$$\begin{cases} \min(z) = d(p) \quad (d(p) \rightarrow \min) \\ -D(t) + D(t') \geq d_t \quad \text{1er cas} \\ -D(t) + d(p) \geq d_t \quad \text{2ème cas} \\ D(t) \geq 0, \quad D(t') \geq 0, \quad d(p) \geq 0 \end{cases} \quad (4.3)$$

**Exemple 4.2 :**

$$\begin{aligned}
 D(2) - D(1) &\geq 4 \\
 D(5) - D(2) &\geq 2 \\
 D(8) - D(5) &\geq 10 \\
 D(9) - D(8) &\geq 1 \\
 D(3) - D(1) &\geq 2 \\
 D(3) - D(2) &\geq 0 \\
 D(6) - D(3) &\geq 1 \\
 D(4) - D(2) &\geq 1 \\
 D(6) - D(4) &\geq 2 \\
 D(7) - D(6) &\geq 2 \\
 D(8) - D(7) &\geq 4 \\
 D(9) &\rightarrow \min
 \end{aligned}$$

On minimise  $D(9)$  et on trouve  $Z^* = 17$  ( avec simplexe)

On va adapter le modèle linéaire précédent pour déterminer quelles tâches à accélérer et la période d'accélération de chacune, de façon à réaliser le projet dans ses délais impartis en minimisant les coûts d'accélération. Tout d'abord on modélise les variables  $Acc_t \geq 0$ .

Où  $t$  est une tâche qui peut être accélérée.

Soit  $T_1$  la durée normale du projet (durée minimale du projet sans compression d'aucune tâche), et  $T_2$  la durée minimale du projet après l'accélération de certaines Tâches, avec  $T_1 > T_2$ , on adopte la fonctionnelle  $z$  au nouvel objectif de minimisation des coûts d'accélération ensuite, si une tâche peut être accélérée on remplace dans les inéquations (4.1) et (4.2), la durée normale  $d_t$  de  $t$  par sa durée accélérée  $d_t - Acc_t$ .

- Si  $t$  est prédécesseur immédiate de  $t'$  alors :

$$D(t') \geq D(t) + (d_t - Acc_t) \quad (4.4)$$

- Si  $t$  n'admet aucun successeur, alors :

$$d(p) \geq D(t) + (d_t - Acc_t) \quad (4.5)$$

On doit tenir compte des bornes supérieures pour les variables  $d(p)$  et  $Acc_t$

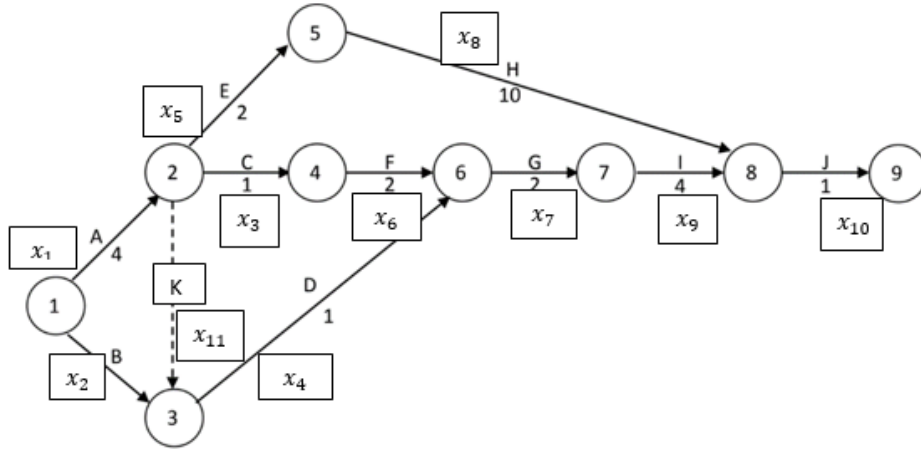
$$\begin{aligned}
 d(p) &\leq T_2 \\
 Acc_t &\leq d_t - d'_t
 \end{aligned}$$

Avec  $d'_t$  : Nouvelle durée de la tâche après accélérations.

Soient  $y_1, y_2, \dots, y_m$  les coûts d'accélération pour les tâches  $1, 2, \dots, m$  respectivement

$$d(p) \leq T_2 \quad \left\{ \begin{array}{l}
 \min z = y_1 Acc_1 + y_2 Acc_2 + \dots + y_m Acc_m \\
 -D(t) + D(t') + Acc_t \geq d_t \\
 -D(t) + d(p) + Acc_t \geq d_t \\
 Acc_t \leq d_t - d'_t \\
 D(t) \geq 0, \quad D(t') \geq 0, \quad d(p) \geq 0, \quad Acc_t \geq 0
 \end{array} \right. \quad (4.6)$$

**Exemple 4.3 :** En utilisant les lois de Kirchoff



Soit le réseau  $R(X, u, d)$  tel que :

$X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$  : l'ensemble de sommet

$U = \{A, B, C, D, E, F, G, H, I, J, K\}$  : l'ensemble des arcs

Pour chaque arc, on a associé une variable booléenne  $x_j$  définie comme suit :

$$x_j = \begin{cases} 1 & \text{si l'arc associé est dans le chemin optimal} \\ 0 & \text{sinon} \end{cases}$$

Afin d'établir notre modèle mathématique, on applique les lois de Kirchoff aux sommets de  $X$

tel que  $x_j \in \{0, 1\} \quad 1 \leq j \leq 11$

Les lois de Kirchoff aux sommets de  $X$ , donne la fonction objectif dans notre cas est  $Z$  à maximiser.

$$Z = -4x_1 - 2x_2 - x_3 - x_4 - 2x_5 - 2x_6 - 2x_7 - 10x_8 - 4x_9 - x_{10} - 0x_{11} \rightarrow \max$$

Les contraintes :

$$x_1 + x_2 = 1$$

$$x_1 = x_5 + x_3 + x_{11}$$

$$x_8 = x_5$$

$$x_8 + x_9 = x_{10}$$

$$x_3 = x_6$$

$$x_6 + x_4 = x_7$$

$$x_7 = x_9$$

$$x_{10} = 1$$

$$x_2 + x_{11} = x_4$$

On applique les règles de la RO : on trouve le chemin maximum égale 17 (simplexe)

## 4.2 Méthode de résolution du modèle d'affectation simple

On peut résoudre le modèle d'affectation simple par l'algorithme de Stepping Stone :

La méthode de Stepping Stone évolue à partir d'une solution réalisable de base de départ vers une solution améliorée, et pour trouver la solution départ on utilise la méthode de Vogel.

### Méthode de Vogel : (Méthode adaptée du problème de transport)

1. Construire la matrice des coûts en incluant les disponibilités et demandes. Ajouter une destination fictive ou une origine fictive pour que  $\sum_i a_i = \sum_j b_j$
2. Evaluer la différence entre les deux coûts les plus petits pour chaque ligne et chaque colonne. Nous obtenons ainsi m différences pour les lignes n différences pour les colonnes.
3. Choisir la ligne ou la colonne ayant le maximum des différences; faire un choix arbitraire si le maximum des différences n'est pas unique.
4. Allouer la quantité la plus grande possible (tout en respectant les contraintes) à la cellule possédant le coût le plus faible de la ligne ou la colonne obtenue en 3.
5. Rayer la ligne ou la colonne qui est saturée.
6. Retourner à 2. Mais cette fois en effectuant les calculs sur la matrice résultante. La procédure se termine lorsque toutes les lignes et colonnes sont saturées.

### Algorithme et principe de Stepping Stone

A partir de la solution réalisable de base de départ

1. Evaluer pour tous les parcours possibles (les sommes algébriques, cout de parcours à considérer que les cases hors bases )
2. Si toutes les sommes algébriques  $\geq 0$  , arrêter la solution est optimale

Si on ajuste la nouvelle solution à partir du parcours ayant la somme algébrique (cout) minimale (négative) aller à 1.

**Remarque** : l'ajustement de la nouvelle solution se fait comme par la méthode des potentiels

#### Exemple 4.4 :

soit le problème d'affectation simple suivant :

15	6(1)	9	8
3(0)	13	7	6(1)
8	9	4(1)	10(0)
3(1)	5(0)	7	11

Est-ce que la solution est optimale

On calcule les parcours : on a 9 cases hors base donc on aura 9 parcours.

Case hors base	Le parcours
(1,1)	(1,1)→(1,2)→(4,2)→(4,1)
(1,3)	(1,3)→(1,2)→(4,2)→(4,1)→(2,1)→(2,4)→(3,4)→(3,3)
(1,4)	(1,4)→(1,2)→(4,2)→(4,1)→(2,1)→(2,4)
(2,2)	(2,2)→(2,1)→(4,1)→(4,2)
(2,3)	(2,3)→(2,4)→(3,4)→(3,3)
(3,1)	(3,1)→(2,1)→(2,4)→(3,4)
(3,2)	(3,2)→(3,4)→(2,4)→(2,1)→(4,1)→(4,2)
(4,3)	(4,3)→(3,3)→(3,4)→(2,4)→(2,1)→(4,1)
(4,4)	(4,4)→(2,4)→(2,1)→(4,1)

On calcule les couts des parcours  $(\Delta_{ij})$  Hors Base :

$$\Delta_{11} = c_{11} - c_{12} + c_{42} - c_{41} = 15 - 6 + 5 - 3 = 11 > 0$$

$$\Delta_{13} = c_{13} - c_{12} + c_{42} - c_{41} + c_{21} - c_{24} + c_{34} - c_{33} = 8$$

$$\Delta_{14} = c_{14} - c_{12} + c_{42} - c_{41} + c_{21} - c_{24} = 1$$

$$\Delta_{22} = c_{22} - c_{21} + c_{41} - c_{42} = 13 - 3 + 3 - 5 = 8$$

$$\Delta_{23} = c_{23} - c_{24} + c_{34} - c_{33} = 7 - 6 + 10 - 4 = 7$$

$$\Delta_{31} = c_{31} - c_{21} + c_{24} - c_{34} = 8 - 3 + 6 - 10 = 1$$

$$\Delta_{32} = c_{32} - c_{34} + c_{24} - c_{21} + c_{41} - c_{42} = 9 - 10 + 6 - 3 + 3 - 5 = 0$$

$$\Delta_{43} = c_{43} - c_{33} + c_{34} - c_{24} + c_{21} - c_{41} = 7$$

$$\Delta_{44} = c_{44} - c_{24} + c_{21} - c_{41} = 5$$

Solution optimale  $Z^* = 19$

# Chapitre 5

## *Implémentation*

### Introduction

La programmation informatique est un ensemble d'outils et de techniques permettant de résoudre des problèmes mathématiques par ordinateurs par exemple, elle sert à trouver une solution optimale de n'importe quel type de problème. Le processus de résolution d'un problème mathématique exige un grand nombre de calculs donc il est mieux de l'exécuter par machine. Pour cela on a choisi les logiciels LINGO, VISUEL XPRESS et MATLAB qui fournissent un environnement de calcul matriciel simple, efficace, interactif permettant la mise en œuvre des algorithmes.

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

#### **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.



#### **Installation de logiciel :**

Pour utiliser cette version de lingo, il est conseillé d'avoir au moins un processeur 486 et 8Mo 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)

### Présentation du logiciel « LINGO »

Nous permettons de formuler nos problèmes d'optimisation linéaire, non linéaire ou en nombre entiers par exemples. Grace 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
- Méthodes 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 entre autres.

@ 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).

**Interface du logiciel :**

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

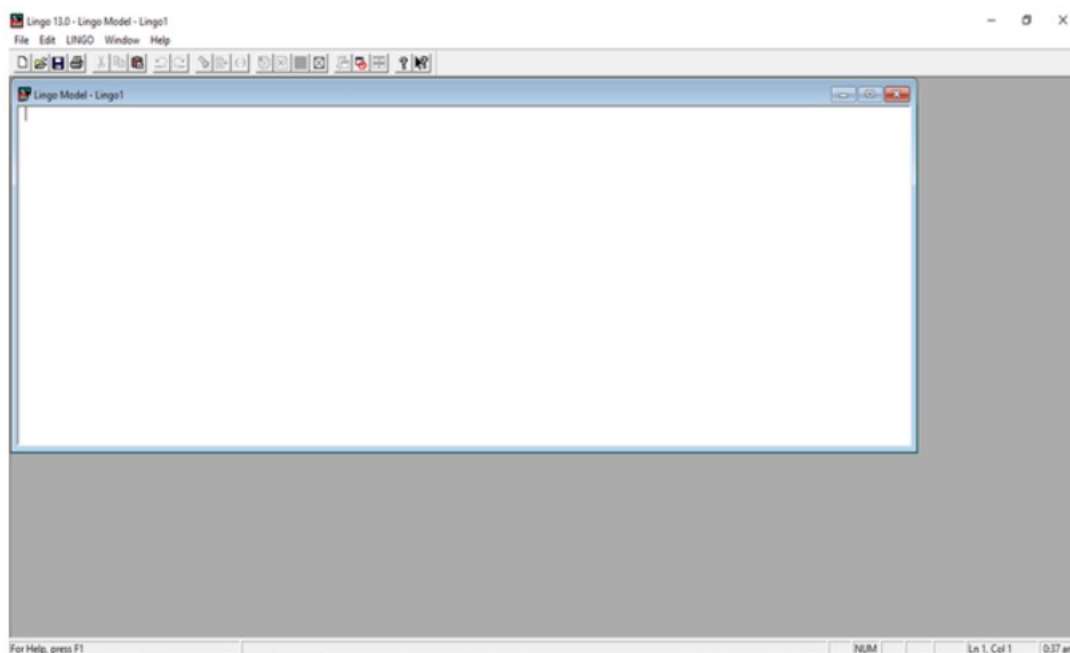


FIGURE 5.1 – La barre des menus du logiciel

La barre des outils de LINGO est la suivante :

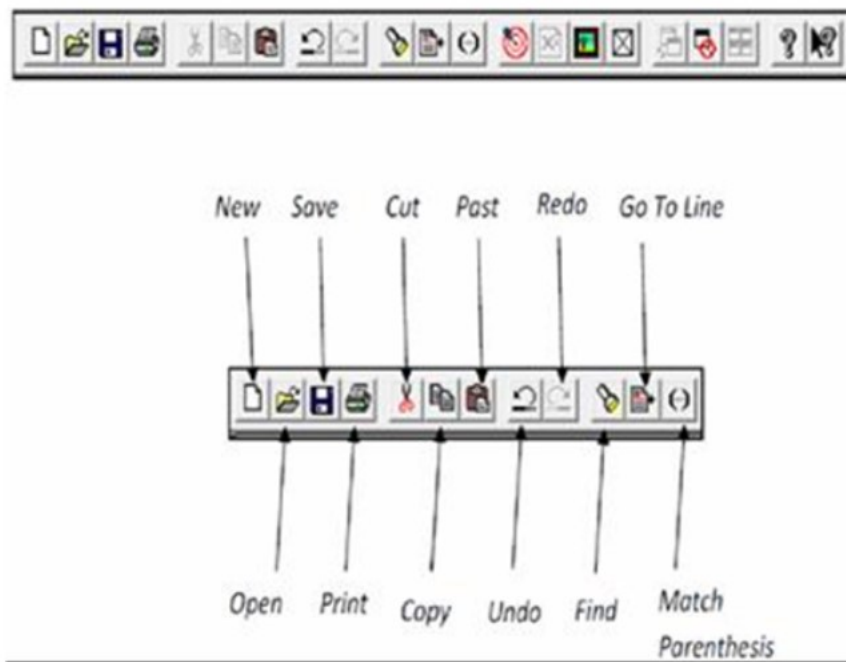


FIGURE 5.2 – La barre des outils de Lingo

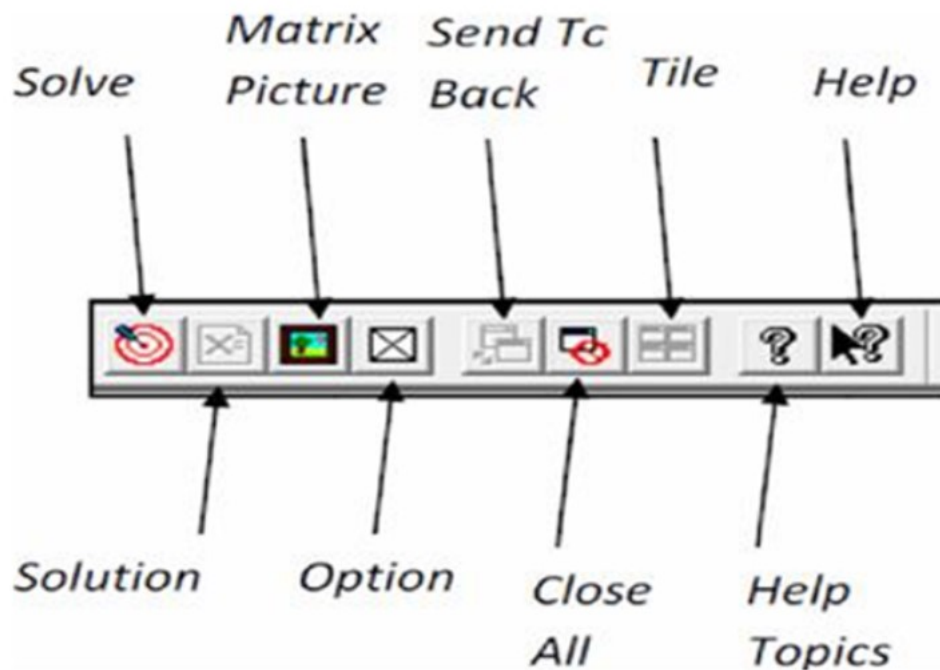


FIGURE 5.3 – La barre des outils de Lingo

**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 du 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.  
**Les étapes de programme :**

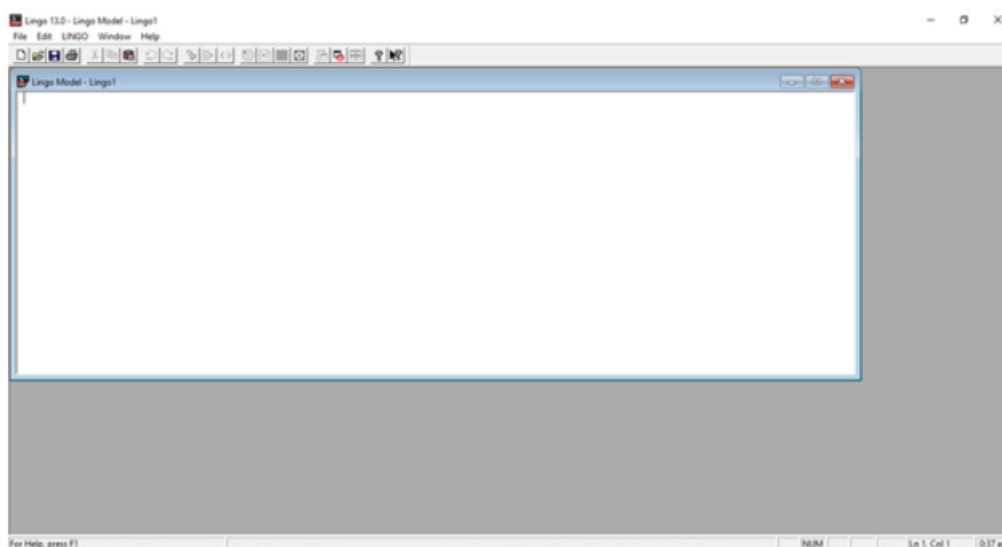
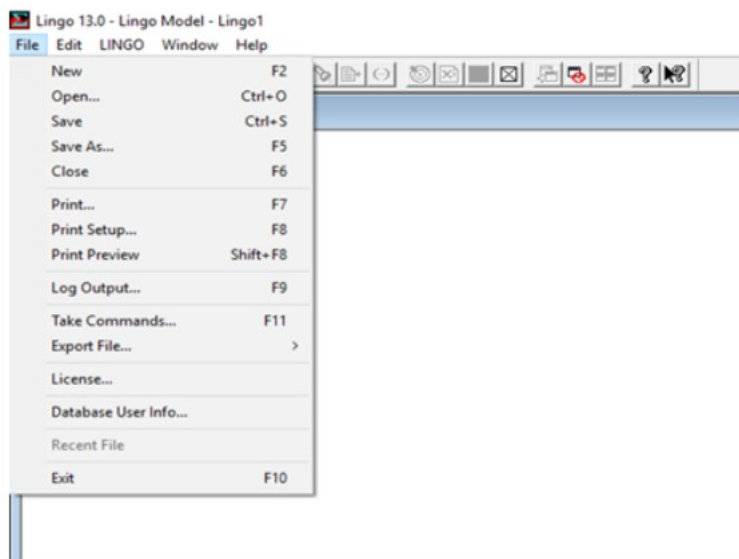
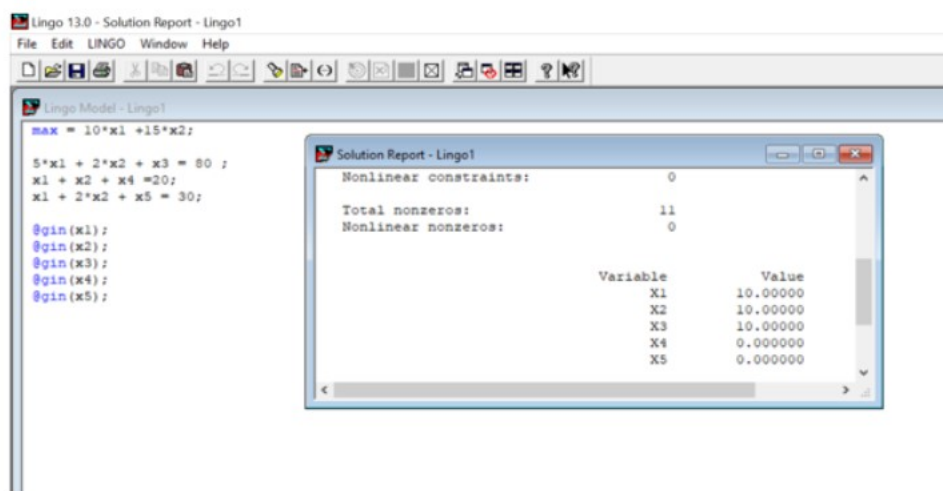


FIGURE 5.4 – La page d'accueil de Lingo

Nous cliquons sur l'icône **FILE** puis **NEW** (figure 5.4) et l'écran de l'espace de travail s'ouvre



**Exemple d'application :** voir exemple de deuxième chapitre (exemple 2.1) (2.7)



## 5.2 Visual Xpress

### Définition :

C'est un logiciel simple à utiliser qui comporte un langage de modélisation qui permet d'écrire les programmes linéaires sous une forme symbolique proche de l'écriture mathématique, permettent ainsi de modifier les données, enlever ou rajouter des contraintes, comparer deux modèles similaires, analyser la sensibilité des solutions par rapport aux données... etc.

### Présentation du logiciel :

Pour commencer, il est conseillé d'installer le logiciel Visual Xpress

Pour démarrer il faut double cliquer sur l'icône (figure 5.5), l'espace de travail standard (figure 5.6) apparaît



## Visual Xpress.Ink

FIGURE 5.5 – L’icone Visual Xpress

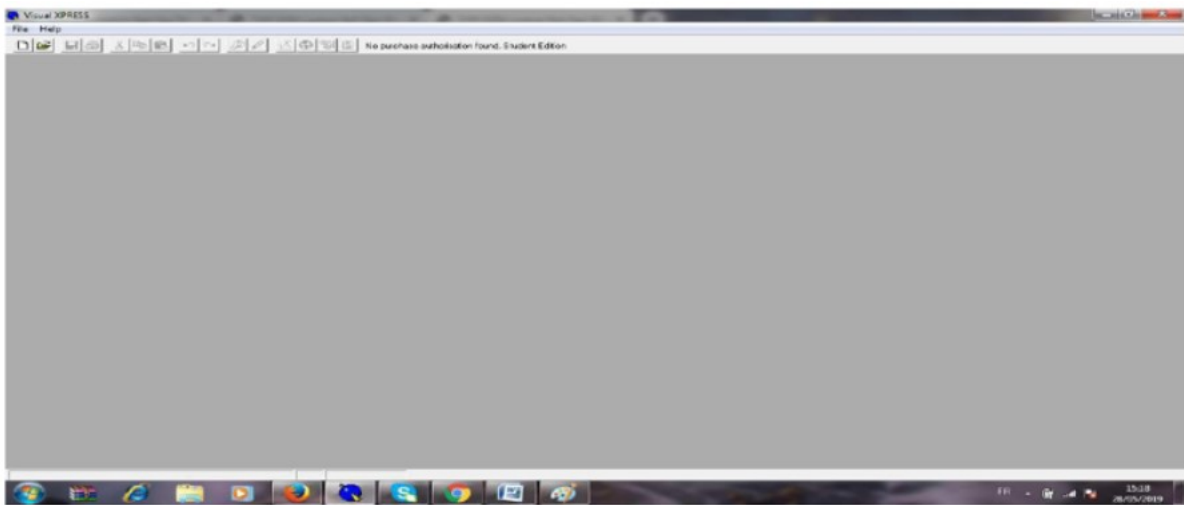
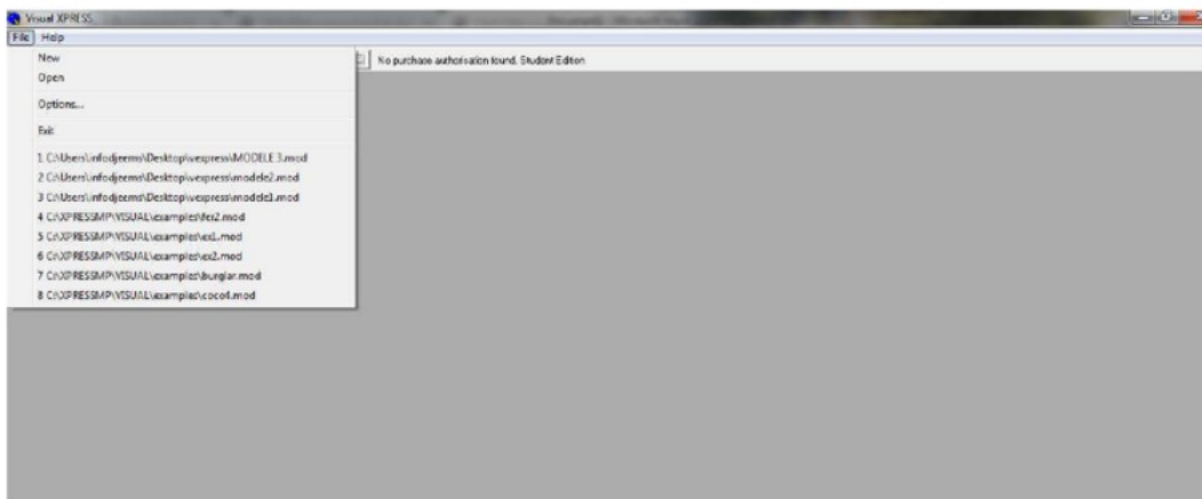


FIGURE 5.6 – Interface de Visual Xpress

Nous cliquons sur l’icône File et on aura l’interface suivante



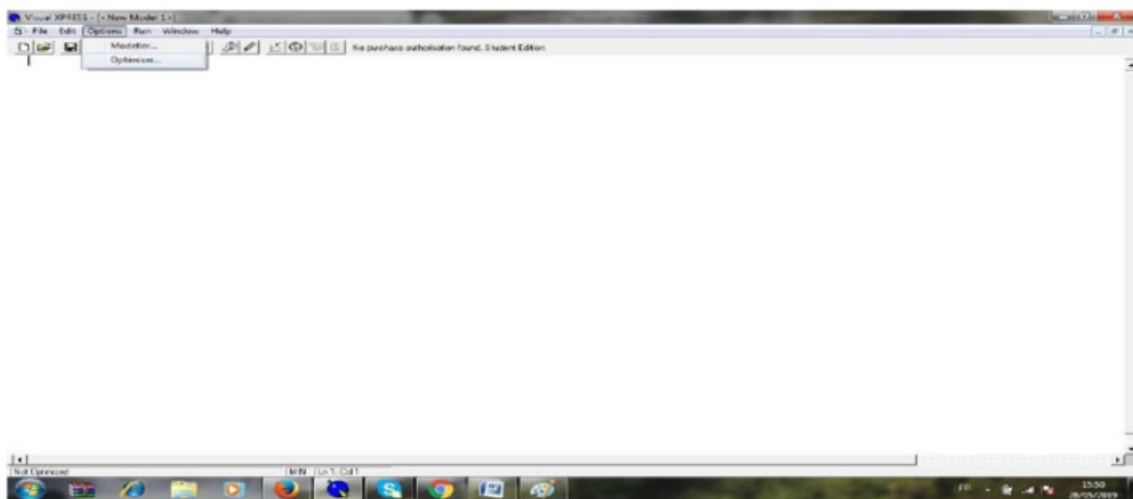
Puis nous cliquons sur new et l’écran de l’espace de travail s’ouvre et on aura l’interface suivante :



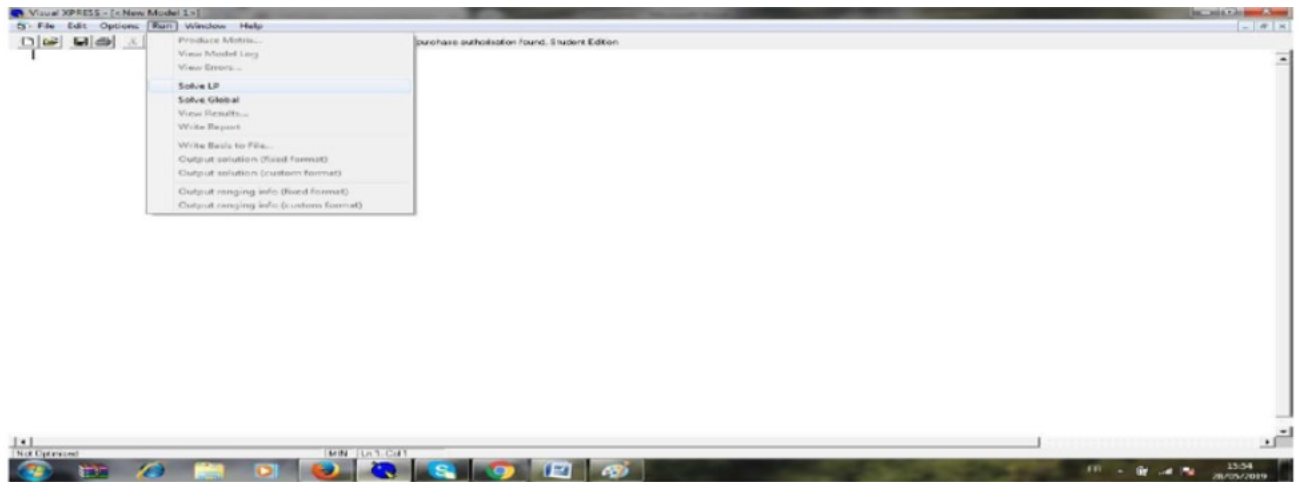
FIGURE 5.7 – Espace de travail de Visual Xpress

**Les étapes du programme :**

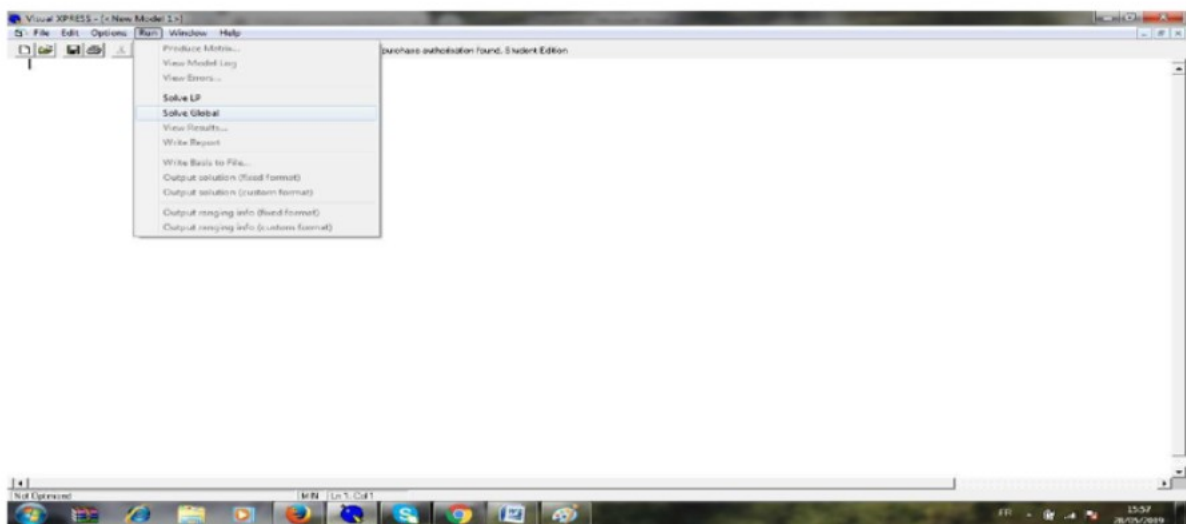
On saisit notre programme dans l'espace de travail et l'exécution se fera comme suit :  
 Avant d'exécuter un problème chargé dans l'éditeur, il faut désigner le sens de l'optimisation (MAX ou MIN) dans option/optimiser comme le montre la figure ci-dessous



On va résoudre le problème linéaire continu avec l'icône Solve LP dans le menu Run / Solve LP



Si le problème linéaire est en nombre entiers, il faut utiliser la commande Run / Solve Globale



## 5.3 MATLAB

### Définition

Matlab est un environnement de programmation et un langage de calcul numérique utilisé principalement dans les domaines scientifiques et d'ingénierie. Il permet de résoudre des problèmes complexes en manipulant des matrices, en effectuant des calculs numériques avancés, en créant des visualisations graphiques et en développant des algorithmes. Matlab est largement utilisé pour la modélisation, la simulation et l'analyse de données, offrant une grande variété d'outils et de fonctions spécialisées pour les applications techniques et scientifiques.

### Description de la fenêtre MATLAB



FIGURE 5.8 – La fenêtre MATLAB

**La barre de titre :**

La fenêtre MATLAB est surmontée par une barre de titre, contenant à sa gauche une icône à sa droite les trois boutons « mise en icône », « minimisation/maximisation » et « fermeture ».

**La barre du menu :** Contient 5 fenêtres (en général)

- File (fichier) permet d'obtenir l'éditeur de programme ;
- Edit (Edition) permet de copier/ couper / coller dans la ligne de commande et autre ;
- Debug permet l'exécution d'un programme et autres,
- Window (fenêtre) permet le passage aux différentes rubriques du logiciel ;
- Help (aide) accède au menu d'aide.

**La barre d'outils :** La barre d'outils qui est souvent des raccourcis de fonctions contenues dans les menus.

De gauche à droite (entre autre) :

- Ouvrir un nouveau fichier dans l'éditeur.
- Rappeler un ancien fichier dans l'éditeur.
- Couper
- Copier
- Coller
- Annuler
- Appeler l'aide

**La fenêtre de commande :** Elles se divisent en deux zones :

- **La zone historique :** qui ne peut être modifiée, mais dont on peut copier des parties ;
- **La zone de commande éditable :** la zone de commande permet (comme son nom l'indique) de taper une commande qui sera accepté à l'aide de la touche < return > ou <entrée>.

## 5.4 les exemples d'applications

### 5.4.1 Exemple d'affectation simple

Voir exemple 4.4 , en utilisant visuel Xpress

```

MODEL TT

VARIABLES
    X11
    X12
    X13
    X14

    X21
    X22
    X23
    X24

    X31
    X32
    X33
    X34

    X41
    X42
    X43
    X44
CONSTRAINTS

PROFIT: 15*X11 + 6*X12 + 9*X13 + 8*X14 + 3*X21 + 13*X22 + 7*X23 + 6*X24 + 8*X31 + 9*X32 + 4*X33 + 10*X34 + 3*X41 + 5*X42 + 7*X43 + 11*X44 = 5

tr1 : X11 + X12 + X13 + X14 = 1
tr11 : X21 + X22 + X23 + X24 = 1
tr111 : X31 + X32 + X33 + X34 = 1
tr121 : X41 + X42 + X43 + X44 = 1

tr21 : X11 + X21 + X31 + X41 = 1
tr22 : X12 + X22 + X32 + X42 = 1
tr33 : X13 + X23 + X33 + X43 = 1
tr44 : X14 + X24 + X34 + X44 = 1

BOUNDS
    X11.ui.1
    X12.ui.1
    X13.ui.1
    X14.ui.1

    X21.ui.1
    X22.ui.1
    X23.ui.1
    X24.ui.1

    X31.ui.1
    X32.ui.1
    X33.ui.1
    X34.ui.1

    X41.ui.1
    X42.ui.1
    X43.ui.1
    X44.ui.1
END

```

voir le resultat suivant

```

***
X44
CONSTRAINTS

```

```

PROFIT: 15*X11 + 6*X12 + 9*X13 + 8*X14 + 3*X21 + 13*X22 + 7*X23 + 6*X24 + 8*X31 + 9*

```

```

tr1 : X11 + X12 + X13 + X14 = 1
tr11 : X21 + X22 + X23 + X24 = 1
tr111 : X31 + X32 + X33 + X34 = 1
tr121 : X41 + X42 + X43 + X44 = 1

tr21 : X11 + X21 + X31 + X41 = 1
tr22 : X12 + X22 + X32 + X42 = 1
tr33 : X13 + X23 + X33 + X43 = 1
tr44 : X14 + X24 + X34

```

```

BOUNDS
X11.ui.1

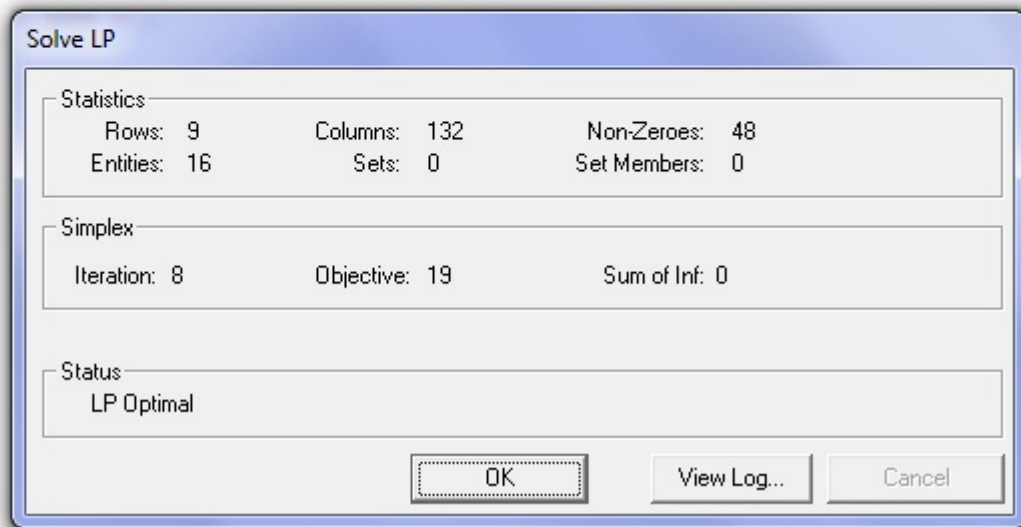
```

```

X12.ui.1
X13.ui.1
X14.ui.1

X21.ui.1
X22.ui.1
X23.ui.1
X24.ui.1

```



### 5.4.2 L'exemple par kirchoff

Voir exemple 4.3 , en utilisant visuel Xpress

```

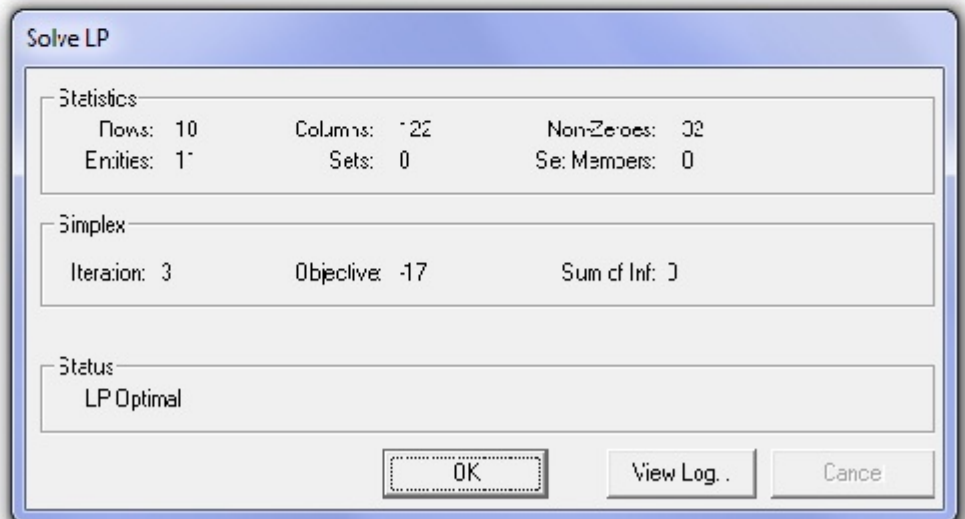
MODEL TT
VARIABLES
  X1
  X2
  X3
  X4
  X5
  X6
  X7

  X8
  X9
  X10
  X11
CONSTRAINTS
PROFIT:  -4*X1 - 2*X2 - X3 - X4 - 2*X5 - 2*X6 - 2*X7 - 10*X8 - 4*X9 - X10 - 0*X11 $

TR1:  X1 + X2 = 1
TR2:  X1 = X5 + X3 + X11
TR3:  X8 = X5
TR4:  X8 + X9 = X10
TR5:  X3 = X6
TR6:  X6 + X4 - X7
TR7:  X7 = X9
TR8:  X10 = 1
TR9:  X2 + X11 = X4
BOUNDS
  X1.ui.1
  X2.ui.1
  X3.ui.1
  X4.ui.1
  X5.ui.1
  X6.ui.1
  X7.ui.1

  X8.ui.1
  X9.ui.1
  X10.ui.1
  X11.ui.1
END

```



### 5.4.3 Exemple PERT

Voir Exemple 4.2, en utilisant visuel Xpress



## *Conclusion générale*

En conclusion, ce mémoire met en lumière l'importance cruciale de la recherche opérationnelle dans toutes les activités bancaires : les investissements, gestion du portefeuille Markovitz, les emprunts, les affectations spécialisées, la théorie des graphes , méthode d'ordonnement ... etc. Ces modèles cités avec leurs extensions s'appliquent pour toutes les banques en particulier à la Caisse Nationale d'Épargne et de Prévoyance (CNEP-Banque de Tizi-Ouzou) ou autres. À travers des approches théoriques et quantitatives, nous avons exploité des outils puissants pour résoudre les modèles cités ci-dessus de manière équivalente sur les plans théoriques , pratiques et informatique.

Les perspectives offertes par les nouvelles technologies et les outils informatiques entre autres constituent un atout majeur pour la CNEP, lui permettant d'améliorer ses stratégies aux défis économiques actuels. La mise en œuvre de ces modèles et méthodes de résolution discutés tout au long de ce mémoire à améliorer permettra à la banque de naviguer avec succès dans un environnement financier complexe et hostile , en constante évolution.

Enfin, il est essentiel pour la CNEP-Banque de continuer à innover et à investir dans la recherche opérationnelle, garantissant ainsi une gestion efficace de ses activités de crédit et une meilleure réponse aux besoins de ses clients entre autres. Ce travail ouvre la voie à de futures recherches et à l'élaboration de pratiques encore plus optimisées pour le secteur bancaire.

# *Résumé*

La recherche opérationnelle (RO) est une approche quantitative essentielle pour améliorer la prise de décision dans les systèmes industriels et économiques. Elle utilise des outils de modélisation pour analyser des situations complexes, notamment en matière de risque de crédit, afin de minimiser les pertes ou maximiser les profits des institutions financières, entre autre.

Le mémoire est structuré en cinq chapitres : le premier présente la CNEP et son historique en matière de crédits et de risques de crédit ; le deuxième traite des concepts de programmation mathématique ; le troisième décrit les modèles mathématiques appliqués dans le secteur bancaire ; le quatrième aborde les méthodes de résolution de ces modèles ; et le dernier chapitre se concentre sur les logiciels utilisés pour résoudre des problèmes mathématiques.

**Les mots clés :** La programmation linéaire, non linéaire, stochastique, flou, ordonnancement.

# Bibliographie

- [1] KADRI.R. *De la CNEP-caisse à la CNEP-Banque.* revu CNEP p.
- [2] MECIBAH.M. *La CNEP-Banque au service de devloppement économique de pays.* Janvier 2003.
- [3] *Le système de payement de masse à la CNEP-Banque.* édition spéciale n°12.
- [4] Bastin Fabian. *Modèle de recherche opérationnelle.* Université de Montériat, 2010.
- [5] Amor F. *Programmation linéaire.* publication universitaires.
- [6] Bierlaire.M. *Introduction à l'optimisation différentiable.* presses polytechniques et universitaires romandes première édition, 2006.
- [7] Michel.M. *Programmation mathématique.* Théorie et Algorithmes 2ème édition, TEC et DOC, 2009.
- [8] Aiche.F. *thèse de doctorat sur la programmation linéaire multi objectif floues Stochastique.* Université Mouloud Mammeri Tizi Ouzou, 2013.
- [9] Teghem J. *Programmation linéaire.* Edition ellipses SMA éditions de l'université de Bruxelles, 1996.
- [10] ZIANE O. *Sur les nombres flous et ses opérations.* Mémoire de master en algèbre et mathématiques discrets, Université Mohamed BOUDIAFDE, M'SILA. 43 p, 2018.
- [11] Ambapour S. *Théorie des ensembles flous : application à la mesure de la pauvreté au Congo.* Bureau d'application des méthodes statistiques et informatiques, 2009.
- [12] Sifhi.M. Meguellati.B. *Optimisation discrète et application aux problèmes stochastiques.* Mémoire de master en mathématique appliquées à la gestion, Université de Tizi Ouzou, 2018.
- [13] Kebbal S. Hacour S. *Programmation mathématique floue et applications.* Mémoire de master en mathématiques appliquées à la gestion, Université de Tizi Ouzou, 2018.
- [14] *Exercices et problèmes résolus de recherche opérationnelle.*
- [15] Piccioli N. Amzallag E. *Introduction à la statistique.* Paris, France, 1993.
- [16] AMIRI D. Bouam Y. *Extension de la dualité dans un environnement flou.* Mémoire de master en mathématiques appliquées à la gestion, Université de Tizi Ouzou, 2017.
- [17] Yadolah D. *premieres pas en statistique.* springer-verlag, France, 2003.
- [18] N. BELHARRAT (et) collectif. *la recherche opérationnelle (théorie des graphes),* les pages bleu ., ALGERIE, 2005.
- [19] OUKACHA ET AIDANE. *théorie des graphes.* Page bleu.
- [20] P. TOLLA R.FAURE, C.ROUCAIROL. *chemin et flots, ordonnancement.* BORDAS, PARIS, 1696.
- [21] Michel Bierlaire. *Introduction à l'optimisation defferentiable.*

- 
- [22] M.Aiden. B.oukacha. *les manuels de l'étudiant Recherche Opirational*.
- [23] Kebbiche.Z. *Etude st extensions d'Algorithmes de points intérieurs pour la programmation non linéaire*. Thèse doctorat en mathématiques appliquées, Université Ferhat Abbas, Sétif, 2007.
- [24] Oukacha B Aiden M. *Programmation linéaire*. Editions Pages Bleues, 2005.
- [25] Bitran G.R and Novaes G. *Linear Programmingwith a fractional objective functionOperations Research 24*. 1976.
- [26] C.Guéret C.Prins M.sevaux. *Programmation linéaire*. N° éditeur :6186, Février 2003.
- [27] 1.7 La Méthode de Balas : C. Guéret C. Prins M. Sevaux. *Titre du Livre*. N° éditeur :6186, Février 2003.
- [28] Wolfe P. *the simplex method for quadratic programming*. Econometrica, 1959.
- [29] Zielinski.P Chanas.S. *Ranking fuzzy interval numbers in the setting of random sets-further results*. . Information Sciences 117, 191-200, 1999.
- [30] Kacher F. *Concept d'équilibre pour un jeu non coopératif sous forme normale avec paramètre indéterminés flous*. Thèse de doctorat, université de Tizi ouzou, 2006.
- [31] Luhandjula M.K. *Fuzzy optimization : An appraisal, Fuzzy Sets and Systems 187-203*. 2004.
- [32] Belouaar H. *Modélisation d'une approche basée agent et logique floue pour la qualité des services Web*. Thèse Doctorat en informatique, Université de Mohammed Khider, Biskra, 2019.