



Département de Mathématiques
Faculté des Sciences Exactes
Université des Frères Mentouri,
Constantine 1

Recherche Opérationnelle

Polycopié destiné aux étudiants de la
première année master mathématiques.
Option Statistique Appliquée.

Ahlam Labdaoui

Table des matières

Avant propos	4
1 Généralités sur la programmation linéaires	5
1.1 Notions et définitions de la programmation linéaire	5
1.2 Définitions de la programmation linéaire	5
1.3 Synthèse de définition de la programmation linéaire	7
1.4 Définition d'un programme linéaire	7
1.5 Les formes d'un programme linéaire	8
1.5.1 La forme canonique	9
1.5.2 La Forme standard	9
1.5.3 La Forme matricielle	10
1.6 Notions relatives à la programmation linéaire	11
1.6.1 La fonction Économique	11
1.6.2 Les variables d'activité $[X_j : X_1, X_2, ..., X_j, ..., X_n]$	12
1.6.3 Les coefficient de la fonction objeectif $[C_j : C_1, C_2, ..., C_j, ..., C_n]$. .	12
1.6.4 Les contraintes d'activité $[a_{ij} X_j] \leq = \geq [b_i]$	12
1.6.5 Les coefficients techniques $[a_{ij}]$	13
1.6.6 Les ressources disponibles $[b_i : b_1, b_2, ..., b_i, ... b_m]$	13
1.6.7 Variables d'écart	13
1.6.8 Coefficients des variables d'écart dans la fonction économique . . .	14
1.6.9 Les conditions de formulation d'un programme linéaire.	14
1.7 Formulation d'un programme linéaire	14
1.8 Solutions d'un programme linéaire	15

2	Méthodes de résolution d'un PL	17
2.1	Méthode d'énumération	17
2.2	Méthode graphique	18
2.3	Méthode simplexe	22
2.3.1	Principe de l'algorithme de simplexe	23
2.3.2	La résolution des programmes de maximisation et type de contraintes ($\leq$)	23
2.3.3	La résolution des programmes de maximisation et type de contraintes ($=$)	32
2.3.4	La résolution des programmes de maximisation et type de contrainte ($\geq$) :	34
2.3.5	La résolution des programmes de minimisation à (n) variables et (m) contraintes	36
2.4	La méthode des pénalités (ou du grand M)	39
2.5	Méthode des deux phases	43
2.5.1	La première phase	43
2.5.2	La deuxième phase	43
3	Dualité et interprétation économique	49
3.1	Introduction	49
3.2	Dualité	50
3.2.1	Règles de construction du dual :	50
3.2.2	Relations primal/dual :	52
3.2.3	Conditions d'optimalité primal-dual	54
3.2.4	Algorithme dual du simplexe	56
3.3	Interprétation économiques des variables duales	59
3.4	Autre application numérique	61
4	Méthode du simplexe révisé	65
4.1	Introduction	65
4.2	Méthode du simplexe	65

4.2.1	Définition	65
4.2.2	Intérêt de la méthode du simplexe	66
4.3	Méthode du simplexe révisé	70
4.3.1	Définition :	70
4.3.2	L'algorithme du simplexe révisé :	70
4.3.3	Principe (algorithme du simplexe révisé) :	70
4.4	Conclusion	77
5	La théorie des jeux et la programmation dynamique	78
5.1	Introduction	78
5.2	Définitions	79
5.2.1	La théorie des jeux	79
5.2.2	Définitions importantes :	79
5.2.3	Classification des jeux	80
5.2.4	Les formes des jeux	81
5.2.5	Quelques exemples sur la théorie des jeux	88
5.3	La programmation dynamique	89
5.3.1	Définition	89
5.3.2	Problèmes d'optimisation et complexité	89
5.3.3	Équation de Bellman	90
5.3.4	Principe d'optimalité de Bellman	90
5.3.5	Equation de la programmation dynamique	91
5.4	Pratique en logicielle Python	91
5.4.1	Exemple	91
5.5	Conclusion	92
	Bibliographie	93

Avant-propos

Ce polycopié est un support de cours de "Recherche opérationnelle" destiné aux étudiants de la première année master mathématiques, option Statistique Appliquée. Il couvre notamment une généralisation sur la programmation linéaire et des méthodes pour la résolution d'un problème linéaire, la dualité et l'interprétation économique, la méthode du simplexe révisé, la théorie des jeux et la programmation dynamique. Pour élaborer ce polycopié, je me suis appuyé sur différentes références, des ouvrages reconnus dans la discipline, mais aussi des ressources en ligne qui sont de plus en plus présentes aujourd'hui dans la diffusion de la connaissance. Un résumé de contenu du polycopié est le suivant : Le premier chapitre nous rappelle quelques notions de : la programmation linéaire, définitions de la programmation linéaire, les formes d'un programme linéaire, la fonction objectif, les contraintes.... Le second chapitre présente quelques méthodes de résolution d'un PL (méthode graphique, méthode simplexe, méthode des deux phases...).

Dans le troisième chapitre, nous avons défini la dualité et ses règles de construction, et la relation entre le primal et dual, plus quelques conditions d'optimalité, et des applications numériques. Et dans le quatrième chapitre, nous avons mis en évidence la méthode du simplexe normal et révisé. Le dernier chapitre est consacré à une présentation de la théorie des jeux et la programmation dynamique.

Chapitre 1

Généralités sur la programmation linéaires

1.1 Notions et définitions de la programmation linéaire

L'importance de l'optimisation est la nécessité d'un outil simple pour modéliser des problèmes de décision dans l'entreprise, ces méthodes furent connues sous le nom de programmation linéaire, développée principalement par George B DANTZIG(né le 8 novembre 1940), mathématicien américain et créateur de la méthode du simplexe, et L.V.KANTOROVICH(1912- 1986) ; On fait de cette technique un des champs de recherches les plus actifs.

Les problèmes de programmation linéaire sont généralement liés à des problèmes d'allocations des ressources limitées, de la meilleure façon d'utiliser, afin d'optimiser,l'objectif pour suivre par l'entreprise.

1.2 Définitions de la programmation linéaire

Nous allons citer quelques définitions attribuées par des différents auteurs à la notion de la programmation linéaire, ensuite nous allons donner une synthèse de définition.

La définition donnée par Michel NEDZELA à la programmation linéaire est la suivante :<< la programmation linéaire s'applique à la répartition des ressources limitées

entre des activités en concurrence les unes avec les autres, de façon à atteindre au mieux un certain objectif>> . Donc, c'est une méthode de résolution du problème économique, soit dans le cadre d'une économie globale, soit dans celui d'une entreprise particulière ; où l'objectif est de sélectionner parmi différentes actions possibles celle qui atteindra le plus probablement l'objectif visé.

Selon Gérard BAILLARGEON :<< la programmation linéaire peut se définir comme un outil mathématique qui permet d'analyser divers types de situations dans les quelles nous retrouvons une fonction linéaire d'un certain nombre de variables, appelée fonction objectif (on utilise également dans la littérature les termes fonction économique) que l'on désire optimiser, c'est-à-dire maximiser ou minimiser>> . Ces variables appelées variables d'activité, puis après la résolution, on les qualifie de variables de décision (dont on veut en déterminer les valeurs optimales) qui sont soumises à des contraintes imposées par des ressources limitées de la situation que l'on veut analyser ; les restrictions qui sont imposées prennent forme d'équations ou d'inéquations linéaires dans la formulation mathématique d'un modèle de programmation linéaire.

Selon Jean-Philippe JAVET :<<la programmation linéaire peut se définir comme une technique mathématique permettant de résoudre des problèmes de gestion et particulièrement ceux où le gestionnaire doit déterminer, face à différentes possibilités, l'utilisation optimale des ressources de l'entreprise pour atteindre un objectif spécifique comme la maximisation des bénéfices ou la minimisation des coûts>> . Dans la plupart des cas, les problèmes de l'entreprise pouvant être traités par la programmation linéaire comportent un certain nombre de ressources. On peut mentionner, par exemple, la main-d'oeuvre, les matières premières, les capitaux, etc. Qui sont disponibles en quantités limitées et qu'on veut les affecter d'une façon optimale entre un certain nombre de processus de fabrication.

1.3 Synthèse de définition de la programmation linéaire

Dans le domaine des sciences de la gestion, on pourrait dire que la programmation linéaire est un outil scientifique qui permet d'obtenir une meilleure affectation des ressources de l'entreprise (main-d'oeuvre, matières premières, capitaux, espace, etc.), pour atteindre un objectif spécifique comme la maximisation des bénéfices ou la minimisation des coûts, << l'objectif de la programmation linéaire est de minimiser (ou maximiser) une fonction objectif linéaire sous contraintes, qui sont représentées par des égalités et/ ou inégalités linéaires >>.

On constate dans les définitions précédentes que la programmation linéaire est la branche des mathématiques qui étudient les variables d'activités sous la forme d'un programme linéaire et la résolution de certains problèmes d'optimisation sous contraintes linéaires ; elle est utilisée, en particulier, dans la production et dans la planification et l'allocation des ressources limitées en vue d'atteindre des objectifs fixés par les décideurs.

On appelle programme linéaire, le problème mathématique qui a pour objectif d'optimiser (maximiser ou minimiser) une fonction linéaire de (n) variables réelles non négatives qui sont reliées par des (m) relations linéaires appelées contraintes sous d'égalités.

1.4 Définition d'un programme linéaire

Un Programme Linéaire (PL) est un problème d'optimisation consistant à maximiser (ou minimiser) une fonction objectif linéaire de variables soumises à un ensemble de contraintes exprimées sous forme d'équations ou d'inéquations linéaires.

La forme générale d'un programme linéaire est la suivante :

$$(P_0) \left\{ \begin{array}{ll} \text{Max}(\text{Min})z = \sum_{j=1}^n c_j x_j & \text{(fonction objectif)} \\ \text{s.c} \quad \sum_{j=1}^n a_{ij} x_j \leq b_i \quad i \in I^- \subseteq \{1, \dots, m\} & \text{(contraintes inégalité)} \\ \sum_{j=1}^n a_{ij} x_j \geq b_i \quad i \in I^+ \subseteq \{1, \dots, m\} & \text{(contraintes inégalité)} \\ \sum_{j=1}^n a_{ij} x_j = b_i \quad i \in I^0 \subseteq \{1, \dots, m\} & \text{(contraintes égalité)} \\ x_j \geq 0 \quad j = 1, \dots, n & \text{(contraintes de non-négativité)} \end{array} \right.$$

où les ensembles I^- , I^+ et I^0 sont disjoints, $I = I^- \cup I^+ \cup I^0 = \{1, \dots, m\}$ et où c_j , a_{ij} et b_i ($i = 1, \dots, m$, $j = 1, \dots, n$) sont des constantes supposées connues. P_0 est un programme linéaire à n variables et m contraintes. Notons qu'il n'est pas restrictif de poser des contraintes de non-négativité pour les variables x_j ($j = 1, \dots, n$). En effet, toute variable x_j qui peut prendre une valeur réelle quelconque, positive, négative ou nulle, peut être remplacée par la différence $(x_j^+ - x_j^-)$ de deux variables x_j^+ et x_j^- astreintes, elles, à être non-négatives (voir par exemple [Minoux 83a]).

1.5 Les formes d'un programme linéaire

On trouve souvent deux formes de programme linéaires. La forme canonique et la forme standard. La première représente la forme initiale, elle se caractérise par des contraintes linéaires sous forme des inéquations ($\leq$; $\geq$) ou d'équations ($=$) comportant (j) variables et (i) contraintes comme suit :

Maximiser ou minimiser selon l'objectif une fonction économique (z) : x_j variables de décision.

$$\text{Max ou Min}(z) = c_1 x_1 + c_2 x_2 + \dots + c_i x_j + \dots + c_n x_n$$

Sous contraintes:

$$\left(\begin{array}{c} a_{11}x_1 + a_{12}x_2 + \dots + a_{1j}x_j + \dots + a_{1n}x_n (\leq, =, \geq) b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2j}x_j + \dots + a_{2n}x_n (\leq, =, \geq) b_2 \\ \dots \\ a_{i1}x_1 + a_{i2}x_2 + \dots + a_{ij}x_j + \dots + a_{in}x_n (\leq, =, \geq) b_i \\ \dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mj}x_j + \dots + a_{mn}x_n (\leq, =, \geq) b_m \\ x_1, x_2, \dots, x_j, \dots, x_n \geq 0 \end{array} \right)$$

x_j : variables de décision ;

b_i : seconds membres(les disponibilités) ;

c_j : coefficients de la fonction objectif ;

a_{ij} : coefficients des contraintes.

1.5.1 La forme canonique

Un programme linéaire est dit sous forme canonique si :

1. Les contraintes sont sous forme des inégalités d'infériorités et la fonction objectif est exprimée sous forme de maximisation.
2. Les contraintes sont sous forme des inégalités de supériorité et la fonction objectif est exprimée sous forme de minimisation. x_j : variables de décision.
3. Les contraintes sont sous forme des inégalités d'infériorités, des inégalités de supériorités et des égalités, et la fonction objectif est exprimée sous forme de maximisation ou de minimisation.

On peut obtenir la forme canonique pour n'importe quel programme linéaire à travers des transformations des contraintes.

1.5.2 La Forme standard

On dit qu'un programme linéaire est mis sous forme standard si sa fonction objectif est une fonction de maximisation et si toutes ses contraintes – en dehors des contraintes

de non négativité – sont des égalités. Un programme linéaire sous forme standard se présente donc comme suit:

$$(P_1) \begin{cases} Max z = c.x \\ \text{s.c.} \quad A.x = b \\ x \geq 0 \end{cases}$$

avec :

- n : nombre de variables ($x = (x_1, \dots, x_n)^T$),
- m : nombre de contraintes ($m \leq n$),
- $A = (a_{ij})_{\substack{i=1, \dots, m \\ j=1, \dots, n}}$: Matrice des contraintes ($m \times n, rang(A) = m$),
- $c = (c_1, \dots, c_n)$: vecteur ligne des profits (ou gains),
- $b = (b_1, \dots, b_m)^T$: vecteur colonne des seconds membres.

Notons que l'on peut toujours considérer que l'on a $rang(A) = m$. En effet, $rang(A) < m$ signifie qu'une ou plusieurs lignes de la matrice A sont une combinaison linéaire des autres lignes. Ceci revient à dire que les contraintes correspondantes sont soit redondantes, soit incompatibles avec les autres (suivant la valeur des seconds membres b_i) ; dans le premier cas, on peut éliminer ces contraintes, dans le second cas, le système $A.x = b$ n'a pas de solution.

Tout programme linéaire peut s'exprimer sous forme standard (P_1), en ajoutant éventuellement certaines variables appelées variables d'écart (par exemple, une inéquation $a.x \leq b$ devient l'équation $a.x + s = b, s \geq 0$).

Dans la suite, nous nous intéressons uniquement aux programmes linéaires exprimés sous forme standard.

1.5.3 La Forme matricielle

On appelle forme matricielle, tout système composé de (m) équations à (n) inconnues devant être vérifiées simultanément et dont l'écriture de la forme suivante :

$$MaxouMin(z) = c_1x_1 + c_2x_2 + \dots + c_ix_j + \dots + c_nx_n.$$

Sous contraintes :

$$\begin{pmatrix} a_{11}, a_{12}, \dots, a_{1j}, \dots, a_{1n} \\ a_{21}, a_{22}, \dots, a_{2j}, \dots, a_{2n} \\ \dots \\ a_{m1}, a_{m2}, \dots, a_{mj}, \dots, a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} \leq, =, \geq \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_3 \end{pmatrix}$$

1.6 Notions relatives à la programmation linéaire

Le traitement du sujet de la programmation linéaire, nécessite de mettre quelques notions relatives au sujet en exergue. << Les composants essentiels d'un problème d'optimisation sont un ensemble de variables de décision, une fonction objectif des variables à être agrandie ou réduite, et un ensemble de contraintes qui caractérisent les valeurs acceptables des variables >> .

1.6.1 La fonction Économique

Une fonction objectif, fonction Économique ou fonction critère, elle décrit la relation linéaire représentant l'objectif de l'entreprise. Par exemple : maximisation du profit global, minimisation du coût de transport, maximisation du chiffre d'affaires, minimisation des achats ... etc. << On appelle fonction objectif d'un problème d'optimisation la fonction qui doit être optimisée et le critère de choix entre les diverses solutions possibles >> .

La fonction objectif est une forme linéaire en fonction des variables de décision est de type :

$$MaxouMin(z) = c_1x_1 + c_2x_2 + \dots + c_ix_j + \dots + c_nx_n$$

c_j : C'est le coefficient de contribution de la variable (x_j) dans la fonction objectif, ce coefficient a une valeur connue avec certitude et peut prendre n'importe quelle valeur (positive ou négative). Par exemple, ce coefficient de contribution peut représenter : un profit unitaire ou un coût de transport unitaire, le prix de vente unitaire ou le prix d'achat, etc.

Cette fonction nous permet de traduire l'objectif poursuivi par l'entreprise en équation linéaire. Elle constitue des variables d'activité et des coefficients économiques.

Chaque variable de décision identifiée dans le modèle, correspond à un coefficient indiquant la contribution unitaire de la variable correspondante à l'objectif.

1.6.2 Les variables d'activité $[X_j : X_1, X_2, \dots, X_j, \dots, X_n]$

Ces variables sont appelés les variables de contrôle ou variables instrumentales ou variables de commande ou plus brièvement les commandes du programme. << On représente par (x_j) les variables d'activité du programme linéaire ; les variables (x_j) ont une signification concrète (quantités produites, vendues ou transportées, durées, valeurs monétaires...etc). Ces variables sont toujours positives ou nulles (condition de non-négativité, $x_j \geq 0$) >> . L'agent décideur a la capacité d'interpréter ces variables et son choix final sera la valeur (x_j) qui optimise la fonction objectif.

1.6.3 Les coefficient de la fonction objectif $[C_j : C_1, C_2, \dots, C_j, \dots, C_n]$

On représente par (C_j) le coefficient de la variable $[X_j : X_1, X_2, \dots, X_j, \dots, X_n]$ de la fonction économique. Dans beaucoup de cas, les coefficients des variables d'activité de la fonction économique représentent les profits ou les coûts associés à une unité des différentes activités. En d'autres termes, c'est la contribution unitaire de la variable correspondante à l'objectif poursuivi par l'entreprise.

1.6.4 Les contraintes d'activité $[a_{ij}X_j] \leq = \geq [b_i]$

Selon Daniel DEWOLF : << On appelle contraintes du problème toutes les relations limitant le choix des valeurs possibles des variables >> . Dans la problématique de la situation, il faut être en mesure d'identifier toutes les contraintes imposées par la disponibilité des facteurs de production, elles présentent les éléments qui limitent le calcul économique de l'entreprise comme (capacité de production limitée, les ventes potentielles, main-d'oeuvre, espace, budget...etc). Dans tout problème formulé sous forme d'un programme linéaire il faut prendre en considération un vecteur des ressources

disponibles b_i [$b_1, b_2, \dots, b_i, \dots, b_n$].

La forme générale d'une contrainte : $[a_{ij}X_j] \leq = \geq [b_i]$ avec :

(i) : c'est le nombre des contraintes

(a_{ij}) : c'est le coefficient technique de la variable (X_j) dans la i^{eme} contrainte ;

(b_i) : c'est le second membre (ressources disponibles).

1.6.5 Les coefficients techniques $[a_{ij}]$

On représente par (a_{ij}) le coefficient technique associé à la ressource (i) et l'activité numéro (j) , il représente une matrice de (i) lignes et de (j) colonnes. <<Les coefficients techniques (a_{ij}) représentent la quantité du facteur (i) par produit (j) >> . C'est les quantités unitaires nécessaires de chaque ressource pour pouvoir conduire une des activités considérées au niveau unitaire, c'est-à-dire lorsque la variable de décision associée à l'activité en question est égale à un (1).

1.6.6 Les ressources disponibles $[b_i : b_1, b_2, \dots, b_i, \dots, b_m]$

On représente par (b_i) les quantités des ressources disponibles et qui limitent l'optimisation de l'objectif poursuivi par l'entreprise.

1.6.7 Variables d'écart

La méthode de résolution que nous venions d'étudier nécessite que les contraintes du modèle soient exprimées sous forme d'équation linéaire au lieu d'inéquation.

On peut facilement transformer une inéquation linéaire ayant un signe $(\leq)$, en une équation linéaire en additionnant une variable non négative dite variable d'écart, << la variable d'écart est la quantité qui, ajoutée aux membres de gauche d'une contrainte, permet de transformer la contrainte en égalité >> . Elle représente l'écart entre la quantité disponible de la ressource (i) et la quantité effectivement utilisée par l'ensemble des (X_j) . Nous ajoutons autant de variable d'écart différents qu'il existe de contrainte du type $(\geq)$.

1.6.8 Coefficients des variables d'écart dans la fonction économique

Comme l'on peut considérer que les variables d'écart permettant de mesurer le niveau d'activité fictive alors, pour qu'elles n'influencent pas l'optimisation, on suppose nuls les bénéfices ou les coûts liés à ces activités.

1.6.9 Les conditions de formulation d'un programme linéaire.

La programmation linéaire comme étant un modèle admet des hypothèses(des conditions) que le décideur doit valider avant de pouvoir les utiliser pour modéliser son problème. Ces hypothèses sont :

1. Les variables de décision du problème sont positives ; on ne peut pas attribuer des valeurs négatives pour la quantité, la surface,... etc.
2. Le critère de sélection de la solution optimale est décrit par une fonction linéaire de ces variables, c'est-à dire que les variables ne sont pas élevées au carré, ne servent pas d'exposant, ne sont pas multipliées entre elles.
3. La fonction qui représente le critère de sélection est dite fonction objectif(ou fonction économique).
4. Les restrictions relatives à l'activité de l'entreprise(exemple : limitations des ressources) peuvent être exprimées par un ensemble d'équations linéaires. Ces équations forment l'ensemble des contraintes.
5. Les paramètres du problème en dehors des variables de décision ont une valeur connue avec certitude.

1.7 Formulation d'un programme linéaire

Un modèle est un moyen pour mieux comprendre la réalité d'un phénomène quelconque. Un modèle linéaire est un système d'équations ou d'inéquations appelées contraintes, qui sont linéaires. Et à partir de ces contraintes, on doit optimiser une fonction également linéaire appelée objectif.

La formulation du modèle mathématique est l'étape la plus délicate de la résolution d'un problème. Elle nécessite un effort de conception qui doit aboutir à la

détermination des trois éléments suivants :

- Les variables de décision pour lesquelles on doit décider du niveau à atteindre, tel que le niveau d'activité dans l'entreprise. On suppose que ces variables peuvent prendre n'importe quelle valeur positive.
- La fonction d'objectif qui décrit la relation linéaire représentant l'objectif de l'entreprise, à l'aide des variables de décision.
- Les contraintes du modèle qui décrivent les relations linéaires entre les variables de décision représentant les restrictions aux quelles est sujette l'entreprise.

Donc, la formulation d'un modèle linéaire consiste à identifier les variables d'activité du problème posé, c'est-à-dire, désigner l'activité de l'entreprise par des variables X_j [$X_1, X_2, \dots, X_j, \dots, X_n$], ces variables appelées aussi variables de décision après la résolution de programme, lorsque la valeur des inconnus est déterminée dans le modèle. Pour pouvoir traduire l'objectif pour suivi par l'entreprise sous forme d'une fonction économique, il s'agit de calculer les coefficients économiques qui indiquent la contribution à cet objectif. Ces variables d'activités sont associés à des contraintes linéaires.

1.8 Solutions d'un programme linéaire

Une solution d'un programme linéaire est une affectation de valeurs aux variables du problème. Une solution est réalisable si elle satisfait toutes les contraintes du problème : $x = (x_1, \dots, x_n)^T$ est une solution réalisable de (P_1) si et seulement si $A.x = b$ et $x \geq 0$. Une solution optimale est une solution réalisable, notée x^* pour laquelle la fonction objectif z prend sa valeur maximale z^* .

Considérons $A.x = b$, le système d'équations linéaires de (P_1) . On appelle base (de l'espace des colonnes de A) toute sous-matrice carrée $m \times m$, inversible de A . Soit B une telle sous matrice de A et N la sous-matrice de A telle que $A = (B|N)$ après permutation des colonnes de A ; on peut écrire :

$$A.x = b \iff (B|N). \begin{pmatrix} x_B \\ x_N \end{pmatrix} = b$$

où x_B sont les variables de base (associées à la base B) de $(P1)$ et x_N sont les variables hors-base de $(P1)$.

En multipliant ce système par B^{-1} , on obtient :

$$B^{-1} \cdot (B|N) \cdot \begin{pmatrix} x_B \\ x_N \end{pmatrix} = B^{-1} \cdot b \iff x_B + B^{-1} \cdot N \cdot x_N = B^{-1} \cdot b$$

On peut alors déterminer l'ensemble des solutions du système $A \cdot x = b$ en fixant arbitrairement les valeurs de x_N et en calculant les valeurs résultantes de x_B :

$$\begin{cases} x_N = v \in R^{n-m} \\ x_B = B^{-1} \cdot b - B^{-1} \cdot N \cdot v \end{cases}$$

La solution particulière obtenue en fixant à 0 les variables hors-base x_N est appelée solution de base (ou *solution basique*) associée à la base B ; elle est donc égale à :

$$x_B = B^{-1} \cdot b \text{ et } x_N = 0$$

Une base est dite dégénérée si le vecteur $x_B = B^{-1} \cdot b$ a une de ses composantes nulle.

Une solution de base (associée à une base B) est une solution de base réalisable si elle vérifie de plus les contraintes de non-négativité $x_B \geq 0$. Une base est réalisable si et seulement si sa solution basique l'est.

On montre (voir par exemple [Minoux83a]) que, si $(P1)$ admet une solution optimale à valeur finie, alors il existe une base réalisable B^* telle que la solution de base associée x^* est optimale.

Chapitre 2

Méthodes de résolution d'un PL

La résolution des problèmes de programmation linéaire vise à déterminer l'allocation optimale et la meilleure allocation possible des ressources limitées pour atteindre certains objectifs, ces allocations doivent minimiser ou maximiser une fonction dite objectif. En économie, ces fonctions sont le profit ou le coût. Pour ce faire, on fait appel aux méthodes de résolution d'un programme linéaire on cite : la méthode d'énumération, la méthode graphique, et la méthode de simplexe, deux phases, et grande M.

2.1 Méthode d'énumération

C'est la méthode la plus ancienne, elle est basée sur le fait que la Solution optimale est un point extrême du polyèdre défini par les contraintes.

Elle se résume selon les étapes suivantes :

- 1 - Déterminer tous les points extrêmes
- 2 - Évaluer la fonction objectif au niveau de chacun de ces points
- 3 - Comparer les valeurs obtenues.

Cependant cette méthode est très fastidieuse si nous avons à faire à des problèmes où les points extrêmes sont nombreux ou bien les composantes des vecteurs sont très nombreuses.

2.2 Méthode graphique

La méthode graphique permet la résolution de problèmes linéaires simples de manière intuitive et visuelle. Cette méthode de résolution n'est applicable que dans le cas où il n'y a que deux variables de décision, << appelée aussi résolution géométrique, elle est possible pour un programme linéaire sous forme canonique avec (n) égale à deux ou trois variables et impossible s'il y a plus de trois (03) variables d'activité >>. Et selon George b. DANTZIG, Mukund n. THAPA : << lorsque des problèmes linéaires ont exactement deux variables soumises à des contraintes d'inégalités, il est possible de les résoudre graphiquement >>. Son avantage est de pouvoir comprendre ce que fait la méthode générale du simplexe, sans entrer dans la technique purement mathématique.

Les étapes du processus de la résolution la méthode des graphique sont les suivantes :

1. Créer un système de coordonnées cartésiennes, dans lequel chaque variable de décision est représentée par un axe.
2. Etablir une échelle de mesure pour chacun des axes appropriés à sa variable associée.
3. Dessiner dans le système de coordonnées les contraintes du problème, y compris celles de non - négativité(qui seront les propres axes). Remarquer qu'une inéquation précise une région qui sera le demi-plan limité par la ligne droite qu'on obtient de considérer la contrainte comme égalité, alors que si une équation détermine une région c'est la ligne droite, elle-même.
4. L'intersection de toutes les régions détermine la région ou l'espace faisable(qui est un ensemble convexe). Si cette région est non vide, passez à l'étape suivante. Sinon, il n'y a pas de point qui satisfait toutes les contraintes simultanément, de sorte que le problème ne sera pas résolu, dit infaisable.
5. Déterminer les points extrêmes ou les sommets du polygone ou polyèdre qui forme la région faisable. Ces points seront les candidats à la solution optimale.
6. Évaluer la fonction objective à chaque sommet et celui (ou ceux) qui maximisent (ou minimisent) la valeur résultante définiront la solution optimale.

Exemple 2.2.1 *Trouver la solution optimale de (PL) suivant avec la méthode graphique.*

$$MaxZ = 3X_1 + 2X_2$$

Sous contraintes :

$$2X_1 + X_2 \leq 18$$

$$2X_1 + 3X_2 \leq 42$$

$$3X_1 + X_2 \leq 24$$

$$X_1, X_2 \geq 0$$

Solution 2.2.1 *Sous contrainte :*

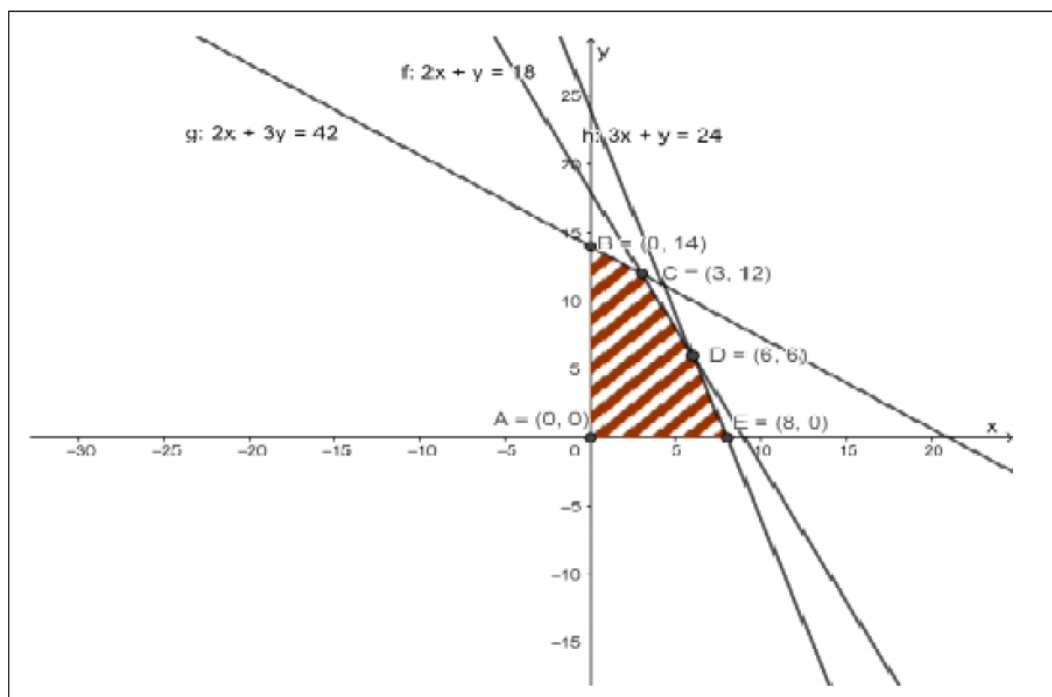
$$2X_1 + X_2 = 18$$

$$2X_1 + 3X_2 = 42$$

$$3X_1 + X_2 = 24$$

$$X_1, X_2 \geq 0$$

<i>sommet</i>	<i>coordonnées(X1, X2)</i>	<i>valeur objectif(Z)</i>
<i>A</i>	(0, 0)	0
<i>B</i>	(0, 14)	28
<i>C</i>	(3, 14)	33
<i>D</i>	(6, 6)	30
<i>E</i>	(8, 0)	24



Remarque 2.2.1

Après l'évaluation de la fonction objectif ($3X_1 + 2X_2$) dans chacun des points (résultat qu'on recueilli dans le tableau suivant). Comme le point G fournit la plus grande valeur à la fonction Z et l'objectif c'est de maximiser, ce point représente la solution optimale : $Z = 33$ avec $X_1 = 3$ et $X_2 = 12$.

Exemple 2.2.2 Trouver la solution optimale de (PL) suivant avec la méthode graphique.

$$\text{Max } Z = 10X_1 + 20X_2$$

Sous contrainte :

$$3X_1 + 5X_2 \geq 75$$

$$X_1 \leq 12$$

$$X_1, X_2 \geq 0$$

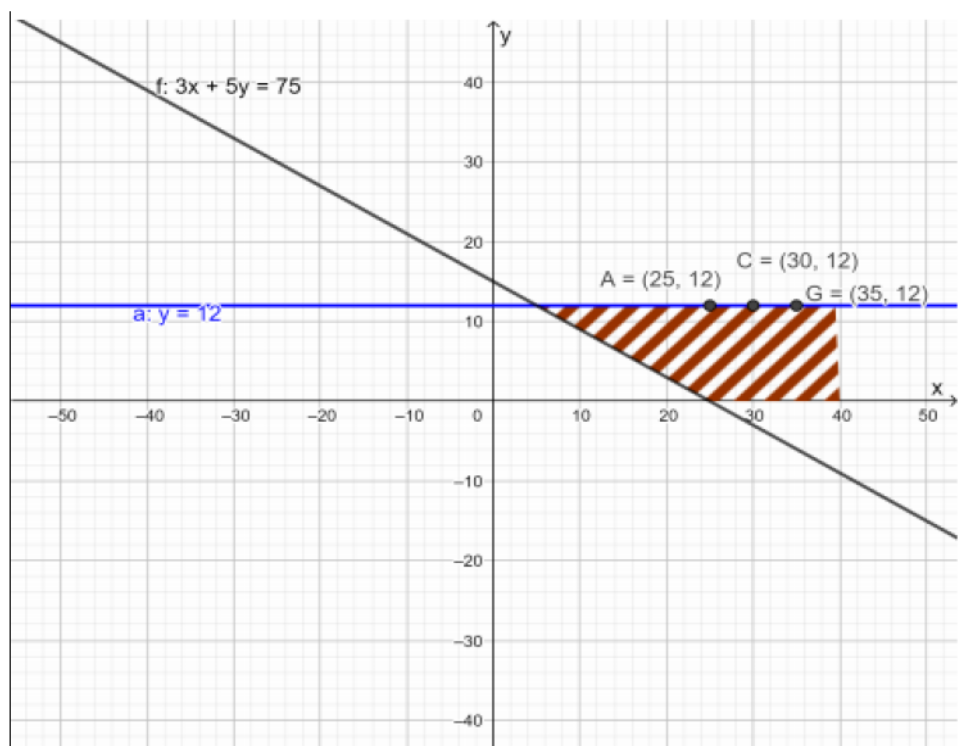
Solution 2.2.2 Sous contrainte :

$$3X_1 + 5X_2 = 75$$

$$X_2 = 12$$

$$X_1, X_2 \geq 0$$

sommet	coordonnées(X_1, X_2)	valeur objectif(Z)
A	(25, 12)	490
C	(30, 12)	540
G	(35, 12)	590
.	.	.
.	.	.



En remarque qu'il n'y a pas de solution limité.

Exemple 2.2.3 Trouver la solution optimale de (PL) suivant avec la méthode graphique.

$$\text{Max } Z = 20X_1 + 15X_2$$

Sous contraintes :

$$5X_1 + 10X_2 \leq 25$$

$$5X_1 + 10X_2 \geq 50$$

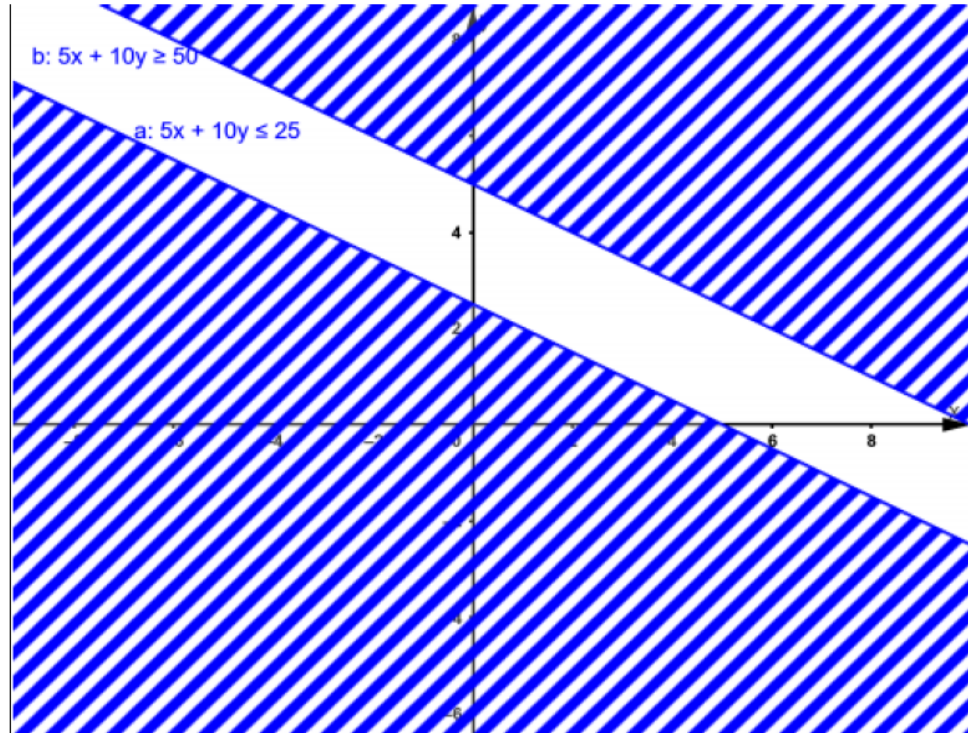
$$X_1, X_2 \geq 0$$

Solution 2.2.3 *Sous contrainte :*

$$5X_1 + 10X_2 = 25$$

$$5X_1 + 10X_2 = 50$$

$$X_1, X_2 \geq 0$$



Remarque 2.2.2 *Graphiquement en déduire qu'il n'y a pas de solution.*

2.3 Méthode simplexe

Résoudre un programme linéaire consiste à déterminer les valeurs des variables non négatives (X_j) qui permettent d'optimiser la fonction économique. Il existe plusieurs méthodes de résolution, la méthode algébrique, la méthode graphique, qui porte chacune des limites. Il faut donc trouver une autre méthode : celle du simplexe, qui est la plus utilisée dans la résolution des programmes linéaires, puisqu'elle répond aux développements des techniques de résolution par ordinateur.

L'algorithme du simplexe constitue une procédure répétitive permettant, de progresser rapidement vers la solution optimale. Elle examine comme première solution un des sommets (en général l'origine), qui constitue la solution de base de

l'algorithme. Son principe consiste à « se déplacer de sommet en sommet adjacent de façon à améliorer la fonction objectif ; après un nombre fini d'itération, il arrive à un sommet à partir duquel tout déplacement vers un autre sommet n'améliore plus cette valeur, on est alors au sommet optimal, l'algorithme simplexe consiste à passer d'une solution de base à une autre jusqu'à ce qu'une solution réalisable de base optimal soit trouvée ». En d'autres termes, la solution optimale est celle qui ne contient aucune variable positive dans la fonction objectif dans le tableau final de simplexe pour un problème de maximisation et aucune variable négative dans la fonction objectif dans le tableau final de simplexe pour un problème de minimisation. Pour calculer la solution optimale, la règle de sélection est appelée règle de pivotage.

2.3.1 Principe de l'algorithme de simplexe

La recherche systématique d'une solution optimale à l'aide de l'algorithme du simplexe peut se résumer comme suit :

1. Déterminer une première solution de base réalisable ; cette solution initiale sert de départ au cheminement vers la solution optimale (si elle existe).
2. Si la solution obtenue en (1) n'est pas optimale, déterminer une autre solution de base réalisable qui permettrait d'améliorer la fonction objectif (augmentation pour une maximisation ou d'minution pour une minimisation).
3. On répète cette procédure itérative jusqu'à ce qu'il ne soit plus possible d'améliorer la fonction objectif. La dernière solution de base réalisable obtenue constitue la solution optimale au programme linéaire.

Pour mieux expliquer l'algorithme de simplexe, nous allons prendre comme exemple le programme linéaire lorsque les contraintes sont de type $(\leq, =, \geq)$.

2.3.2 La résolution des programmes de maximisation et type de contraintes $(\leq)$

L'application de la méthode nécessite de transformer les inégalités des contraintes économiques en égalités par l'introduction des variables supplémentaires positives ou

nulles appelées variables d'écart. Ces variables d'écart sont utilisées pour justifier le stock résiduel des ressources à la solution optimale. À la solution de base, les variables d'écart égaux aux disponibilités.

On peut déterminer les variables d'écart de programme linéaire précédent comme suit :

$$\text{La variable d'écart } E_1 = X_{n+1} = b_1 - (a_{11}X_1 + a_{12}X_2 + \dots + a_{1j}X_j + \dots + a_{1n}X_n)$$

$$\text{La variable d'écart } E_2 = X_{n+2} = b_2 - (a_{21}X_1 + a_{22}X_2 + \dots + a_{2j}X_j + \dots + a_{2n}X_n)$$

$$\cdot \quad \cdot \quad \cdot \quad \cdot$$

$$\text{La variable d'écart } E_m = X_{n+m} = b_m - (a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mj}X_j + \dots + a_{mn}X_n)$$

La forme standard

Le nouveau programme linéaire après l'introduction des variables d'écart est le suivant :

$$\text{Max}(Z) = C_1X_1 + C_2X_2 + \dots + C_nX_n + 0X_{n+1} + 0X_{n+2} + \dots + 0X_{n+m}.$$

Sous contraintes :

$$\left(\begin{array}{l} a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n + X_{n+1} = b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n + X_{n+2} = b_2 \\ \dots \\ a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n + X_{n+m} = b_m \\ X_1, X_2, \dots, X_n, X_{n+1}, X_{n+2}, \dots, X_{n+m} \geq 0 \end{array} \right)$$

Définition d'un tableau de simplexe :

Selon Daniel DEWOLF : «un tableau de simplexe est constitué des coefficients des équations algébriques sans le nom des variables» . On aura donc :

1. Les coefficients de la fonction objectif ;
2. Les coefficients des variables dans le membre de gauche des contraintes ;
3. Les coefficients du membre de droite.

Où l'on sépare les coefficients de l'objectif des contraintes d'une barre horizontale, et les coefficients du membre de gauche des contraintes des coefficients du membre de droite par une barre verticale.

Tableau 01 : Le tableau de simplexe (solution de base $Z = 0$) :

H.Base/Base	$X_1, X_2, \dots, X_n, X_{n+1}, X_{n+2}, \dots, X_{n+m}$	b_i
X_{n+1}	$a_{11}, a_{12}, \dots, a_{1n}, 1, 0, \dots, 0$	b_1
X_{n+2}	$a_{21}, a_{22}, \dots, a_{2n}, 0, 1, \dots, 0$	b_2
.	$\dots \dots \dots \dots \dots \dots \dots \dots$	.
X_{n+m}	$a_{m1}, a_{m2}, \dots, a_{mn}, 0, 0, \dots, 1$	b_m
C_j	$C_1, C_2, \dots, C_m, 0, 0; \dots, 0$	$Z=0$

Le tableau ci-dessus représente la forme standard et la solution de base ($Z = 0$) d'un programme linéaire de (n) variables et (m) contraintes.

Le passage de la solution de base à la solution optimale afin d'améliorer la fonction économique se fait par plusieurs itérations.

Pour pouvoir passer d'une itération à l'itération suivante, il s'agit de sélectionner une variable entrante à la base et une variable sortante de la base.

Dans un programme de maximisation (minimisation), selon les critères de <<**Dantzig**>> la variable entrante est celle qui correspond à la valeur maximale (minimale) des coefficients (C_j). La variable sortante est celle qui correspond au plus petit rapport positif des disponibilités données comme suit :

$$\text{Min}(b_1/a_1^e, b_2/a_2^e, \dots, b_m/a_m^e) > 0$$

$a_1^e, \dots, a_2^e, \dots, a_m^e$: représentent le vecteur des coefficients techniques de la variable entrante. On arrête les différentes itérations dès que la ligne (C_j) ne contient que des coefficients : négatifs ou nuls pour un programme de maximisation, positifs ou nuls pour un programme de minimisation. Pour effectuer le passage de la solution de base au tableau suivant (itération suivante), nous allons considérer (C_2) comme le plus grand (petit) coefficient des coefficients (C_j), et ($b_1 = a_{12}$) est le plus petit rapport positif des rapports (b_i/a_i^e) > 0 . Donc, X_2 : variable entrante à la base et, X_{n+1} : variable sortante de la base.

Dans ce nouveau tableau nous allons diviser la ligne X_{n+1} , de tableau précédent sur le coefficient (a_{12}) pour calculer le pivot à l'intersection de la variable entrante et de la variable sortante pour obtenir le tableau, on applique les règles suivantes :

- Le pivot est égal à (1) ;
- Les coefficients de la ligne de pivot sont divisés par le pivot ;
- Les coefficients de la colonne de pivot sont nuls ;
- Les autres coefficients sont obtenus par la règle du rectangle suivante :

$$\begin{array}{c}
 \xrightarrow{A} \\
 A \times b \\
 \uparrow \div \\
 C - d
 \end{array}
 \quad D' = d - (c/a) * b$$

Tableau N02 : le tableau suivant de simplexe (itération suivant).

$H.Base/ Base$	X_1	X_2	...	X_n	X_{n+1}
X_{n+1}	(a_{11}/a_{12})	1	...	(a_{1n}/a_{12})	$(1/a_{12})$
X_{n+2}	$a_{21} - a_{22} (a_{11}/a_{12})$	...	...	$a_{2n} - a_{22} (a_{11}/a_{12})$	$0 - a_{22} (a_{11}/a_{12})$
...	...	...	...	...	...
X_{n+m}	$a_{m1} - a_{m2}(a_{11}/a_{12})$	...	...	$a_{mn} - a_{m2}(a_{11}/a_{12})$	$0 - a_{m2}(a_{11}/a_{12})$
C_j	$C_1 - C_2(a_{11}/a_{12})$	...	...	$C_m - C_2(a_{11}/a_{12})$	$0 - C_2(a_{11}/a_{12})$

...	X_{n+m}	b_i
...	0	b_1/a_{12}
...	$0 - a_{22} (a_{11}/a_{12})$	$b_2 - a_{22}(a_{11}/a_{12})$
...	...	...
...	$1 - a_{m2}(a_{11}/a_{12})$	$b_m - a_{m2}(a_{11}/a_{12})$
...	0	$Z - c_2(a_{11}/a_{12})$

La colonne des disponibilités représente les valeurs des variables non négatives (variable de décision).

Exemple 2.3.1 Trouver la solution optimale pour le modèle (LP) en utilisant la méthode simplexe ?

$$Max Z = 3X_1 + 5X_2$$

Sous contrainte :

$$2X_1 + 3X_2 \leq 30$$

$$5X_1 + 4X_2 \geq 60$$

$$X_1, X_2 \geq 0$$

Solution 2.3.1 1. La forme standard :

$$\text{Max. } Z - 3X_1 - 5X_2 + 0X_3 + 0X_4 = 0$$

Sous contrainte :

$$2X_1 + 3X_2 + X_3 = 30$$

$$5X_1 + 4X_2 + X_4 = 60$$

$$X_1, X_2, X_3, X_4 \geq 0$$

2. Tableau 01(initial) :

X3 2, 3, 1, 0, 30

X4 5, 4, 0, 1, 60

<i>H.Base/Base</i>	X_1	X_2	X_3	X_4	b_i
X_3	2	3	1	0	30
X_4	5	4	0	1	60
Z	-3	-5	0	0	0

3. La variable entrante est : (X_2)

4. La variable sortante est : (X_3)

5. Élément de pivot est : (3)

6. Équation de pivot est

$$= \left[\frac{2}{3}, \frac{3}{3}, \frac{1}{3}, \frac{0}{3}, \frac{30}{3} \right]$$

$$= \left[\frac{2}{3}, 1, \frac{1}{3}, 0, 10 \right]$$

7. Nouvelle élément de Z et X_4

$$\begin{aligned} Nouvelle(Z) &= [-3, -5, 0, 0, 0] - (-5) \times \left[\frac{2}{3}, 1, \frac{1}{3}, 0, 10\right] \\ &= \left[\frac{1}{3}, 0, \frac{5}{3}, 0, 50\right] \end{aligned}$$

$$\begin{aligned} Nouvelle(X_4) &= [5, 4, 0, 1, 60] - (-4) \times \left[\frac{2}{3}, 1, \frac{1}{3}, 0, 10\right] \\ &= \left[\frac{7}{3}, 0, \frac{-4}{3}, 1, 20\right] \end{aligned}$$

8. Tableau 2 :

$H.Base/Base$	X_1	X_2	X_3	X_4	b_i
X_2	$\frac{2}{3}$	1	$\frac{1}{3}$	0	10
X_4	$\frac{7}{3}$	0	$-\frac{4}{3}$	1	20
Z	$\frac{1}{3}$	0	$\frac{5}{3}$	0	50

Tous les Coefficients de la fonction objectif est supérieur ou égal à 0 ($C_i \geq 0$) alors la solution optimale est : $X_1 = 0, X_2 = 10, Z = 50$.

On déduit que la gestion de la production a décidé produire (10) unités de produit (X_2) et ne produire aucune du premier produit (X_1) pour le maximum de profits qui égal 50DA

La solution de l'exercice avec la méthode simplexe.

% PROGRAMME simplexe.m

% exemple :

% max $3x_1 + 5x_2$

% s.c $2x_1 + 3x_2 \leq 30$

% $5x_1 + 4x_2 \leq 60$

% $x_1, x_2 \geq 0$

% on a deux contraintes , donc on ajoute deux variables d'écrit

% x_3, x_4

% i.e on ajoute x_3 à la 1ère contrainte , x_4 à la 2ème contrainte

```
% Donc N=[ 3 4]
% qui est l'ensemble d'indice des variable d'écart
% la matrice A est donc A=[ 2 3 1 0 30 ;5 4 0 1 60 ;3 5 0 0 0]
% dont : la 1 ère ligne cotientles coefficients de la 1 ère contrainte
% la 2 ème ligne cotientles coefficients de la 2 ème contrainte
% la dernière ligne contient les coefficients de la fonction objective
% Et puis exécuter ? [ sol,val,kk] = ssimplex1(A,N)
% Note : la variable de base correspondant à la matrice de base doit etre la matrice
% unité. (cette limitation sera derrière la mise à niveau est d'améliorer)
% Résoudre le programme linéaire standard : max c*x ; st A*x =b ; pour x<=0
% Cette fonction est simplement la table initiale ,y compis : la derniÈre ligne est
% Le nombre de l'indice de base des variable initiales
% Sortie : variable sol est la solution optimale
% variable val est la valeur de la fonction objective
% kk est le nombre d'itÈrations
function [sol,val,kk] = simplex1(A,N)
A=[ 2 3 1 0 30 ;5 4 0 1 60 ;3 5 0 0 0]
N=[3 4]
[mA,nA] = size(A) ;
kk=0 ;
stop =0 ;
while ~stop
kk=kk+1 ;
if A(mA, :)<=0
stop=1 ;
sol= zeros(1,nA-1) ;
for i=1 :mA-1
sol(N(i))=A(i,nA) ;
end
disp(' ') ;
```

```
disp('LES VARIABLES DE BASE :');
disp('=====') ;
x1= sol(1)
x2= sol(2)
disp(' ')
disp('LA FONCTION ECONOMIQUE : ') ;
disp('=====') ;
val = -A(mA,nA)
disp(' ') ;
else
for i=1 :nA-1
if A(mA,i) > 0 & A(1 :mA-1,i) <= 0
disp('have infinite solution !') ;
stop=1 ;
break ;
end
end
if ~stop
temp=0 ;
for i=1 :nA-1
if A(mA,i) > temp
temp = A(mA,i) ;
inb=i ;
end
end
temp = inf ;
for i=1 :mA-1
if A(i,inb) > 0
if A(i,nA)/A(i,inb)> 0 & A(i,nA)/A(i,inb) < temp
temp=A(i,nA)/A(i,inb) ;
```

```
    outb=i ;
end
end
end
for i=1 :mA-1
    if i==outb
        N(i)=inb ;
    end
end
for i=1 :mA
    if i~= outb
        A(i, :)=A(i, :)-A(outb, :)*A(i,inb)/A(outb,inb) ;
    end
end
A(outb, :)=A(outb, :)/A(outb,inb) ;
end
end
end
end
% FIN DU PROGRAMME

A =
2 3 1 0 30
5 4 0 1 60
3 5 0 0 0

N =
3 4

LES VARIABLES DE BASE :
=====
x1 =
0
```

x2 =

10

LA FONCTION ECONOMIQUE :

=====

val =

50

2.3.3 La résolution des programmes de maximisation et type de contraintes (=)

Lorsque les contraintes de programme linéaire à résoudre sont sous forme d'équation, on ne peut pas introduire des variables d'écarts, puisqu'il n'existe pas d'écart entre la disponibilité et la contrainte.

La forme standard pour ce type de programme nécessite l'introduction des variables artificielles à la place des variables d'écart.

Soit à résoudre le programme linéaire suivant sous sa forme canonique :

$$Max(Z) = C_1X_1 + C_2X_2 + \dots + C_nX_n$$

Sous contraintes :

$$\left(\begin{array}{l} a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n = b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n = b_2 \\ \dots \\ a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n = b_m \\ X_1, X_2, \dots, X_n \geq 0 \end{array} \right)$$

La forme standard : Nous allons introduire $\langle\langle T_i \rangle\rangle$ comme variable artificielle :

$$Max(Z) = C_1X_1 + C_2X_2 + \dots + C_nX_n - MT - MT_2 - \dots - MT_m$$

Sous contraintes :

$$\left(\begin{array}{l} a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n + T_1 = b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n + T_2 = b_2 \\ \dots \\ a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n + T_m = b_m \\ X_1, X_2, \dots, X_n, T_1, T_2, \dots, T_m \geq 0 \end{array} \right)$$

Les variables artificielles $(T_1; T_2, \dots, T_m)$ sont associées aux valeurs très élevées (coefficient $M = +\infty$).

Le tableau de simplexe (solution de base $Z = 0$) :

H.Base/Base	X_1	...	X_n	T_1	...	T_m	b_i
T_1	a_{11}	...	a_{1n}	1		0	b_1
T_2	a_{21}	...	a_{2n}	0	1	0	b_2
.	...	...	...	...	...	...	.
T_m	a_{m1}	...	a_{mn}	0	...	1	b_m
C_j	C_1	...	C_m	M	...	M	Z=0

Pour pouvoir appliquer la méthode de simplexe et les critères de Dantzig sur le tableau ci-dessus, il s'agit de respecter les conditions de la solution par cette méthode.

Il faut procéder à éliminer les coefficients $(-M)$ dans la ligne (C_j) . La ligne (C_j) donne les coefficients de la fonction économique, mais pas les valeurs marginales des variables hors base ; de plus, les variables artificielles sont dans la base et devraient donc avoir des valeurs marginales nulles.

Pour calculer les valeurs marginales (m_j) qui sont nulles pour les variables de base : $m_j = C_j + (a_{11} + a_{12}, \dots + a_{mj})M$. (voir l'itération suivante) :

Tableau indiquant l'itération suivante :

H.Base/Base	X_1	...	X_n	T_1	...	T_m	b_i
T_1	a_{11}	...	a_{1n}	1	...	0	b_1
T_2	a_{21}	...	a_{2n}	0	...	0	b_2
.	...	...	...	...	...	...	.
T_m	a_{m1}	...	a_{mn}	0	0	1	b_m
C_j	C_1	...	C_m	-M	...	-M	$Z=0$
Δ ou m_i	$C_1 + M \begin{pmatrix} a_{11} \\ a_{21} \\ a_{m1} \end{pmatrix}$		$C_m + M \begin{pmatrix} a_{11} \\ a_{21} \\ a_{m1} \end{pmatrix}$	0	...	0	$Z + M \begin{pmatrix} b_1 \\ b_2 \\ b_m \end{pmatrix}$

À partir de ce tableau on peut appliquer les critères de la variable entrante et de la variable sortante. Les modifications à l'intérieur de tableau se font par la méthode de pivotage.

2.3.4 La résolution des programmes de maximisation et type de contrainte ($\geq$) :

Dans le cas d'une contrainte de signe ($\geq$) on doit, pour la transformer en équation, soustraire une variable d'écart, également appelée dans ce cas, variable d'excédent.

Soit le programme linéaire suivant :

$$Max(Z) = C_1X_1 + C_2X_2 + \dots + C_nX_n$$

Sous contraintes :

$$\begin{pmatrix} a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n \geq b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n \geq b_2 \\ \dots \\ a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n \geq b_m \\ X_1, X_2, \dots, X_n \geq 0 \end{pmatrix}$$

Le passage de la forme canonique à la forme standard nécessite l'introduction des variables d'écart négatives dans chaque contrainte de type ($\geq$).

$$Max(Z) = C_1X_1 + C_2X_2 + \dots + C_nX_n - 0X_{n+1} - 0X_{n+2} - \dots - 0X_{n+m}$$

Sous contraintes :

$$\left(\begin{array}{l} a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n - X_{n+1} \geq b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n - X_{n+2} \geq b_2 \\ \dots \\ a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n - X_{n+m} \geq b_m \\ X_1, X_2, \dots, X_n, X_{n+1}, \dots, X_{n+m} \geq 0 \end{array} \right)$$

Le tableau de simplexe (solution de base $Z = 0$) :

H.Base/Base	X_1	...	X_n	X_{n+1}	...	X_{n+m}	b_i
X_{n+1}	a_{11}	...	a_{1n}	-1	0	0	b_1
X_{n+2}	a_{21}	...	a_{2n}	0	1	0	b_2
.	...	...	...	...	...	...	.
X_{n+m}	a_{m1}	...	a_{mn}	0	...	-1	b_m
Z	C_1	...	C_m	0	...	0	b_j

On constate dans ce tableau que le diagonal de la matrice des variables d'écart est négatif :

$$\left(\begin{array}{ccccc} -1 & 0 & \dots & \dots & 0 \\ 0 & -1 & 0 & \dots & 0 \\ 0 & 0 & -1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & 0 & -1 \end{array} \right)$$

De ce fait, pour appliquer la méthode de simplexe, il faut ajouter à chaque contrainte une variable artificielle (t_i) comme suit :

$$Max(Z) = C_1X_1 + C_2X_2 + \dots + C_nX_n - 0X_{n+1} - 0X_{n+2} - \dots - 0X_{n+m} - Mt_1 - Mt_2 - \dots - Mt_m$$

Sous contraintes :

$$\left(\begin{array}{l} a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n - X_{n+1} + T_1 = b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n - X_{n+2} + T_2 = b_2 \\ \dots \\ a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n - X_{n+m} + T_m = b_m \\ X_1, X_2, \dots, X_n, X_{n+1}, X_{n+2}, \dots, X_{n+m}, T_1, T_2, T_m \geq 0 \end{array} \right)$$

Le tableau précédent devient comme suit :

Le tableau de simplexe.

H.Base/Base	X ₁	...	X _n	X _{n+1}	...	X _{n+m}	T ₁	...	T _m	b _i
T ₁	a ₁₁	...	a _{1n}	-1	0	0	1	0	0	b ₁
T ₂	a ₂₁	...	a _{2n}	0	1	0	0	...	0	b ₂
.	...	...	...	...	...	...	...	...	...	.
T _m	a _{m1}	...	a _{mn}	0	...	-1	0	0	1	b _m
Z	C ₁	...	C _m	0	...	0	-M	...	-M	b _j

La résolution de ce programme demande l'utilisation de la méthode des pénalités pour éliminer les coefficients (M) des variables artificielles. Donc, la dernière ligne du tableau précédent devient comme suit :

Δ ou m_j	$C_1 + M \begin{pmatrix} a_{11} \\ a_{22} \\ a_{m1} \end{pmatrix}, \dots, C_m + M \begin{pmatrix} a_{11} \\ a_{22} \\ a_{m1} \end{pmatrix}, -M, \dots, -M, 0, \dots, 0$	$Z + M \begin{pmatrix} b_1 \\ b_2 \\ b_m \end{pmatrix}$
-------------------	---	---

2.3.5 La résolution des programmes de minimisation à (n) variables et (m) contraintes

Le processus de résolution par la méthode de simplexe est identique pour les programmes linéaires à maximiser ou à minimiser. Le traitement de la fonction économique est indépendamment des contraintes :

1. Quelle que soit la fonction économique (Max ou Min), les modifications à porter sur les contraintes pour réaliser la forme standard sont les suivantes :

- Lorsque la contrainte est de forme $(\geq)$, on ajoute une variable d'écart négative $(-E)$ et une variable artificielle positive (T_i) .

- Type de contrainte est de forme $(\leq)$, on ajoute uniquement une variable d'écart positive (E_i) .

- La contrainte sous forme d'équation $(=)$, on ajoute uniquement une variable artificielle (T_i) .

- Lorsque l'objectif est de maximiser une fonction économique, les variables d'écart sont associées aux coefficients (-0) , les variables artificielles sont associées aux coefficients $(-M)$

- Lorsque l'objectif consiste à minimiser une fonction économique, les variables d'écart sont associées aux coefficients $(+0)$, les variables artificielles sont associées aux coefficients $(+M)$

2. Règles générales pour la résolution d'un programme linéaire par la méthode de simplexe (problème de maximisation).

Selon Eric jacquet-LAGREZE : << pour passer d'un sommet à un sommet adjacent, Dantzig a introduit deux critères (critères de Dantzig) qui sont au coeur de la méthode du simplexe : un critère de la nouvelle variable précédemment hors base à faire entrer dans la base et un critère du choix d'une variable de la base à faire sortir >>

A- Choix de la colonne pivot (critère de sélection de la variable entrante dans la base) :

La colonne pivot est définie à partir des coefficients de la fonction économique. << Le critère de sélection de la variable entrante est donc le suivant : on choisit la variable avec le coefficient objectif le plus élevé >> . On cherche à se focaliser sur la variable qui, en augmentant, augmentera le plus possible la fonction objectif. Cette variable correspond au plus grand coefficient positif de la fonction objectif.

Considérons les coefficients $[C_1, C_2, \dots, C_j, \dots, C_n]$ de la fonction économique. Parmi tous les coefficients positifs, on considère le plus grand, la colonne pivot est la colonne qui le contient. S'il existe plusieurs coefficients correspondant à cette valeur positive maximale, on peut choisir celui que l'on veut (arbitrairement). La variable correspon-

dante sera la variable entrante car elle ne va plus s'annuler. On sélectionne la variable hors base ayant le plus grand coefficient positif dans la ligne.

Pour sélectionner la variable sortante de la base, il est nécessaire de rajouter une colonne (T) (ratios) au tableau, obtenue en faisant le rapport membre à membre de la colonne (b_i) (disponibilités) et de la colonne de la variable entrante dans la base.

B- Choix de la ligne pivot (critère de sélection de la variable sortante de la base) :

La variable entrante va prendre la place d'une des variables de base, appelé variable sortante. Il faut maintenant trouver quelle valeur maximum peut prendre cette variable entrante afin de maximiser la fonction objectif. Pour un programme linéaire à plusieurs variables on fait diviser, chaque coefficient de la colonne (b_i) sur le coefficient correspondant de la colonne pivot et on calcule les rapports ($b_i = a_{ij}$). On sélectionne le plus petit rapport, la variable correspondante à cette ligne est la variable sortante. On sélectionne la variable dans la base ayant le plus petit coefficient positif dans la colonne (T). <<Le critère de sélection de la variable sortante est donc le suivant : prendre comme variable sortante la première variable de base à s'annuler. De manière générale, on calcule le minimum du rapport du coefficient du membre de droite sur le coefficient de la variable entrante dans la même ligne, lorsque celui-ci est positif. La variable sortante est celle dont on lit la valeur dans la ligne où ce minimum se produit>> . Et l'élément pivot correspond à l'intersection de la colonne et la ligne pivot <<on appelle élément pivot le coefficient situé à l'intersection de la colonne pivot et de la ligne pivot>> .

C- Itération :

S'il existe un coefficient (C_j) positif dans le nouveau tableau, on retourne à la première étape (choix du pivot) puis à la deuxième (itération suivante). On réitère ce processus jusqu'à ce que tous les coefficients de la fonction économique soient négatifs.

D- Critère d'arrêt des itérations :

Si tous les coefficients de la ligne relatifs aux variables hors base, sont négatifs ou nuls pour la maximisation, ou positifs ou nuls pour la minimisation, la solution trouvée est optimale.

La solution d'un (PL) avec la méthode simplexe cas de minimisation (min) la fonction objectif, ie : tous les contraintes ($\geq$ ou $=$), dans des cas spécial en utilisé l'une des méthode suivant :

1. La méthode des pénalités (ou du grand M).
2. Méthode des deux phases.

2.4 La méthode des pénalités (ou du grand M)

Cette méthode permet de tenir compte des variables artificielles. On les pénalise en leur affectant un coefficient de valeur très élevée dans la fonction économique ($-M$) pour un problème du maximum, ($+M$) pour un problème du minimum . Les pénalités ont pour objet de provoquer l'élimination des variables artificielles au fil des itérations.

Les étapes de la méthode :

1. Transformation de la forme standard à la forme canonique et ajoutée les variables (X_i) à la fonction objectif.
2. La formulation de la fonction objectif (Z) avec l'utilisation de les variables (X_i) et (X_j):
3. Construire qui un tableau qui contient la solution en fonction des coefficient (T_i , X_i , X_j).
4. Trouvé le variable entrante et le variable sortante.
5. En suit, répète les étapes de la méthode simplexe.
6. Finalement si tout les coefficient de fonction objectif est ($C_j \leq 0$) alors la solution optimal est trouver

Exemple 2.4.1 *Trouver la solution optimale pour le modèle (LP) en utilisant la méthode (Big.M)?*

$$\text{Min} Z = 2X_1 + X_2$$

Sous contrainte :

$$X_1 + 3X_2 \geq 30$$

$$4X_1 + 2X_2 \geq 40$$

$$X_1, X_2 \geq 0$$

Solution 2.4.1 La forme standard :

$$MinZ = 2X_1 + X_2 + 0X_3 + 0X_4$$

Sous contrainte :

$$X_1 + 3X_2 - X_3 = 30$$

$$X_1 + 3X_2 - X_4 = 40$$

$$X_1; X_2; X_3; X_4 \geq 0$$

on a X_3 et X_4 sont négative $X_3 = -30, X_4 = -40$; contradiction avec la non négative de (X_3 et X_4); alors on ajoutant des variable artificielles aux contraintes et à la fonction objectif (Z).

- On ajoutant les variable artificielles comme suit :

$$MinZ = 2X_1 + X_2 + 0X_3 + 0X_4 + MT_1 + MT_2$$

Contrainte :

$$X_1 + 3X_2 - X_3 + T_1 = 30.....(1)$$

$$4X_1 + 2X_2 - X_4 + T_2 = 40.....(2)$$

$$X_1, X_2, X_3, X_4, T_1, T_2 \geq 0$$

M :est très grand.

-Formulation de la fonction objectif (Z) en termes des variables (X_j, X_i) : De (1) et (2) on trouve (T_1) et (T_2) comme suit :

$$T_1 = 30 - X_1 - 3X_2 + X_3$$

$$T_2 = 40 - 4X_1 - 2X_2 + X_4$$

On remplace les valeurs de T_1 et T_2 dans la fonction objectif (Z);on trouve :

$$\begin{aligned}
 Z &= 2X_1 + X_2 + M(30 - X_1 - 3X_2 + X_3) + M(40 - 4X_1 - 2X_2 + X_4) \\
 &= (2 - 5M)X_1 + (1 - 5M)X_2 + MX_3 + 70M \\
 &= 2X_1 + X_2 + M30 - MX_1 - 3MX_2 + MX_3 + M40 - 4MX_1 - 2MX_2 + MX_4 \\
 &= (2 - 5M)X_1 + (1 - 5M)X_2 + MX_4 + MX_3 + 70M \\
 70M &= Z - (2 - 5M)X_1 - (1 - 5M)X_2 - MX_3 - MX_4
 \end{aligned}$$

Tableaux de solution de base :

Variable de base	variable non basic						b_i	Ration
	X_1	X_2	X_3	X_4	T_1	T_2		
Z	$-1 + 5M$	$-1 + 5M$	$-M$	$-M$	0	0	$70M$	—
T_1	1	3	-1	0	1	0	30	10
T_2	4	2	0	-1	0	1	40	20

-La variable entrante est (X_2) ($M = 100$).

-La variable sortante est (T_1).

- Élément de pivot est (3).

-Équation de pivot = $[\frac{1}{3}, 1, -\frac{1}{3}, 0, \frac{1}{3}, 0, 10]$

-On trouve la Nouvelle valeurs de (Z) et (T2) comme suit :

$$\begin{aligned}
 Nouvelle(Z) &= (-2 + 5M, -1 + 5M, -M, -M, 0, 0) - (-1 + 5M) \times [\frac{1}{3}, 1, -\frac{1}{3}, 0, \frac{1}{3}, 0, 10] \\
 &= [-\frac{5}{3} + \frac{10}{3}M, 0, -\frac{1}{3} + \frac{2}{3}M, -M, \frac{1}{3} - \frac{5}{3}M, 0, 10 + 20M]
 \end{aligned}$$

$$\begin{aligned}
 Nouvelle(T_2) &= [4, 2, 0, -1, 0, 1, 40] - 2 \times [\frac{1}{3}, 1, -\frac{1}{3}, 0, \frac{1}{3}, 0, 10] \\
 &= [\frac{10}{3}, 0, \frac{2}{3}, -1, \frac{-2}{3}, 1, 20]
 \end{aligned}$$

-On place les résultats dans le tableau suivant :

Variable de base	variable non basic						b_i	Ration
	X_1	X_2	X_3	X_4	T_1	T_2		
Z	$-\frac{5}{3} + \frac{10}{3}M$	0	$-\frac{1}{3} + \frac{2}{3}M$	$-M$	$\frac{1}{3} - \frac{5}{3}M$	0	$10 + 20M$	—
X_2	$\frac{1}{3}$	1	$-\frac{1}{3}$	0	$\frac{1}{3}$	0	10	30
T_2	$(\frac{10}{3})$	0	$\frac{2}{3}$	-1	$\frac{-2}{3}$	1	20	6

-La variable entrante est (X_1) ($M = 100$).

- La variable sortante est (T_2).

- Elément de pivot est ($\frac{10}{3}$).

-Équation de pivot

$$\begin{aligned}
 &= \left[\frac{\frac{10}{3}}{\frac{10}{3}}, \frac{0}{\frac{10}{3}}, \frac{\frac{2}{3}}{\frac{10}{3}}, \frac{-1}{\frac{10}{3}}, \frac{\frac{-2}{3}}{\frac{10}{3}}, \frac{1}{\frac{10}{3}}, \frac{20}{\frac{10}{3}} \right] \\
 &= \left[1, 0, \frac{1}{5}, \frac{-3}{10}, \frac{-1}{5}, \frac{3}{10}, 6 \right]
 \end{aligned}$$

-On trouve la Nouvelle valeurs de (Z) et (X_2) comme suit :

$$\begin{aligned}
 \text{Nouvelle}(Z) &= \left[\frac{-5}{3} + \frac{10}{3}M, 0, \frac{1}{3} + \frac{2}{3}M, -M, \frac{1}{3} - \frac{5}{3}M, 0, 10 + 20M \right] \\
 &\quad - \left[-\frac{5}{3} + \frac{10}{3}M \right] \times \left[1, 0, \frac{1}{5}, \frac{-3}{10}, \frac{-1}{5}, \frac{3}{10}, 6 \right] \\
 &= \left[0, 0, 0, \frac{-1}{2}, -M, \frac{1}{2}, -M, 20 \right]
 \end{aligned}$$

$$\begin{aligned}
 \text{Nouvelle}(X_2) &= \left[\frac{1}{3}, 1, -\frac{1}{3}, 0, \frac{1}{3}, 0, 10 \right] - \left[\frac{1}{3} \right] \times \left[1, 0, \frac{1}{5}, \frac{-3}{10}, \frac{-1}{5}, \frac{3}{10}, 6 \right] \\
 &= \left[0, 1, \frac{-2}{3}, \frac{1}{10}, \frac{2}{5}, -\frac{1}{10}, 8 \right]
 \end{aligned}$$

-On place ces résultats dans la table suivante :

Variable de base	variable non basic						b_i
	X_1	X_2	X_3	X_4	T_1	T_2	
Z	0	0	0	$-\frac{1}{2}$	$-M$	$\frac{-1}{2} - M$	20
X_2	0	1	$-\frac{2}{5}$	$\frac{1}{10}$	$\frac{2}{5}$	$-\frac{1}{10}$	8
X_1	1	0	$\frac{1}{5}$	$-\frac{3}{10}$	$\frac{-1}{5}$	$\frac{3}{10}$	6

Tous les coefficients de la fonction objectif est supérieur ou égale à 0 ($C_i \geq 0$)
alors la solution optimal est :

$$X_1 = 6, X_2 = 8, Z = 20.$$

2.5 Méthode des deux phases

La méthode des deux phases est plus simple que la méthode de grande (M) pour pour trouvée la solution optimale sur la (PL) dans le cas de minimisation , on peut trouver la solution optimale si la valeur de la nouvelle fonction (t) est égale à zéros(0), si non on ne peut pas trouver la solution optimale, cette méthode est décomposer à deux phases :

2.5.1 La première phase

1. Transformation de la forme canonique à la forme standard et ajouté les variable artificielles (T_i) à les contraintes.

2. La Formulation de la fonction objectif (t) avec l'utilisation de les variables artificielle (T_i) ie : $r = T_1 + T_2 + ...T_n \rightarrow Min$.

3. Construit un tableau qui contient la solution primal en fonction des coefficient des variables (T_i, X_j) et la nouvelle fonction objective.

4. Suivons les étapes précédents jusque'à ce que nous obtenons la valeur ($t = 0$) ce qui signifie qu'il y a une solution ie : ($C_j \leq 0$) de la fonction de toutes les coefficient objectif (t)

2.5.2 La deuxième phase

1. Utilise la solution finale de la (4) étape de la première phase après la suppression des variables artificielle (T) la fonction objectif (t)

2. Utilise la fonction objectif (Z) et améliore leur valeur pour trouver la solution optimal.

3. Si on trouve les coefficient (C_j) soient obtenus inférieure ou égale zéros (0) ie :
($C_j \leq 0$) ce la indique que la solution optimale est trouve.

Exemple 2.5.1 *Trouver la solution optimale d'un modèle (LP) en utilisant la méthode de deux phases ?*

$$\text{Min. } Z = 2X_1 + X_2$$

Sous contraintes :

$$X_1 + 3X_2 \geq 30$$

$$4X_1 + 2X_2 \geq 40$$

$$X_1, X_2 \geq 0$$

Solution 2.5.1 * *La première phase :*

Transforme le modèle (PL) de la forme canonique à la forme standard

$$\text{Min } Z = 2X_1 + X_2 - 0T_1 - 0T_2$$

Sous contrainte :

$$X_1 + 3X_2 - X_3 = 30 \dots (1)$$

$$4X_1 + 2X_2 - X_4 = 40 \dots (2)$$

$$X_1, X_2, X_3, X_4 \geq 0$$

on a X_3 et X_4 sont négative ($X_3 = -30, X_4 = -40$), contradiction avec la non négative de T_1, T_2 ($T_1, T_2 \geq 0$), alors on ajoutant des variable artificielles (T_1, T_2) aux les contraintes comme suit :

$$X_1 + 3X_2 - X_3 + T_1 = 30 \dots (1)$$

$$4X_1 + 2X_2 - X_4 + T_2 = 40 \dots (2)$$

$$X_1, X_2, X_3, X_4, T_1, T_2 \geq 0$$

Formulation de la nouvelle fonction objectif (t) en termes de les valeurs de T_1 et T_2 avec la fonction (t) est égale un valeurs constantes ,de l'équation (1) et (2) on obtient :

$$t = T_1 + T_2 \rightarrow \text{Min}$$

$$T_1 = 30 - X_1 - 3X_2 + X_3$$

$$T_2 = 40 - 4X_1 - 2X_2 + X_4$$

On remplace les valeurs de T_1 et T_2 dans la fonction objectif (Z) on trouve :

$$\begin{aligned} t &= (30 - X_1 - 3X_2 + X_3) + (40 - 4X_1 - 2X_2 + X_4) \\ &= 70 - 5X_1 - 5X_2 + X_3 + X_4 \end{aligned}$$

$$t + 5X_1 + 5X_2 - X_3 - X_4 = 70$$

Tableau (1) (la solution optimale) :

Variable de base	variable non basic						b_i	Ration
	X_1	X_2	X_3	X_4	T_1	T_2		
t	5	5	-1	-1	0	0	70	—
T_1	1	3	0	0	1	0	30	30
T_2	(4)	2	0	0	0	1	40	10

La variable entrante est (X_1) et la variable sortante est (T_2);et élément de pivot est (4)

$$\text{Équation de pivot} = [1, \frac{1}{2}, 0, -\frac{1}{4}, 0, \frac{1}{4}, 10]$$

Nouvelle valeur de T_1 et (t)

$$\begin{aligned} \text{Nouvelle}(t) &= [5, 5, 1, -1, 0, 0, 70] - 5 * [1, \frac{1}{2}, 0, \frac{1}{4}, 0, \frac{1}{4}, 0] \\ &= [0, \frac{5}{2}, -1, \frac{1}{4}, 0, \frac{-5}{4}, 20] \end{aligned}$$

$$\text{Nouvelle}(T_1) = [0, \frac{5}{2}, -1, \frac{1}{4}, 1, \frac{-1}{4}, 30]$$

On place les résultats précédente dans le tableau(2)

Variable de base	variable non basic						b_i	Ration
	X_1	X_2	X_3	X_4	T_1	T_2		
t	0	$\frac{5}{2}$	-1	$\frac{1}{4}$	0	$-\frac{5}{4}$	20	—
T_1	0	$(\frac{5}{2})$	-1	$\frac{1}{4}$	1	$-\frac{1}{4}$	30	8
T_2	1	$\frac{1}{2}$	0	$-\frac{1}{4}$	0	$\frac{1}{4}$	10	20

La variable entrante est (X_2) et la variable sortante est (T_1);et élément de pivot est $(\frac{5}{2})$

$$\text{Équation de pivot} = [0, 1, -\frac{2}{5}, \frac{1}{10}, \frac{2}{5}, -\frac{1}{10}, 8]$$

Nouvelle valeur de X_1 et (t)

$$\begin{aligned} \text{Nouvelle}(t) &= [0, \frac{5}{2}, -1, \frac{1}{4}, 0, -\frac{5}{4}, 20] - \frac{5}{2} * [0, 1, -\frac{2}{5}, \frac{1}{10}, \frac{2}{5}, -\frac{1}{10}, 8] \\ &= [0, 0, 0, 0, -1, -1, 0] \end{aligned}$$

$$\begin{aligned} \text{Nouvelle}(X_1) &= [1, \frac{1}{2}, 0, -\frac{1}{4}, 0, -\frac{1}{4}, 10] - \frac{1}{2} * [0, 1, -\frac{2}{5}, \frac{1}{10}, \frac{2}{5}, -\frac{1}{10}, 8] \\ &= [1, 0, \frac{1}{5}, -\frac{3}{10}, -\frac{1}{5}, \frac{3}{10}, 6] \end{aligned}$$

On place les résultats précédente dans le tableau (3)

Variable de base	variable non basic						b_i	Ration
	X_1	X_2	X_3	X_4	T_1	T_2		
t	0	0	0	0	-1	-1	0	—
X_2	0	1	$-\frac{2}{5}$	$\frac{1}{10}$	$\frac{2}{5}$	$-\frac{1}{10}$	8	—
X_1	1	0	$\frac{1}{5}$	$-\frac{3}{10}$	$-\frac{1}{5}$	$\frac{3}{10}$	6	—

On a la valeur de la fonction objectif ($t = 0$) ie : ($C_j \leq 0$) alors la solution de le modèle est exisete.

On va passe à la deusième phase...

Deusième phase :

Utilisée la solution finale de la 4 ème étape de première phase après la supperssion de (X_3) et la f onction objectif (t).

Utilisée la fonction objectif (Z)

$$Z = 2X_1 + X_2 - 0X_3 - 0X_4$$

et optimisée leur valeur pour trouvée la solution optimal final comme suit :

Variable de base	variable non basic				b_i
	X_1	X_2	X_3	X_4	
t	0	0	0	0	0
X_2	0	1	$-\frac{2}{5}$	$\frac{1}{10}$	8
X_1	1	0	$\frac{1}{5}$	$-\frac{3}{10}$	6

Après la suppression de (T1 et T2) et la fonction objective(t) de tableau de la solution finale et ajoutée la fonction (Z) on écrit les contraintes en fonction des résultats de la table finale

$$\begin{cases} X_2 - \frac{2}{5}X_3 + \frac{1}{10}X_4 = 8 \dots\dots\dots (1) \\ X_1 - \frac{1}{5}X_3 - \frac{3}{10}X_4 = 6 \dots\dots\dots (2) \end{cases}$$

De l'équation (1) et (2), on obtient les valeurs (X_1, X_2) :

$$\begin{cases} X_2 = 8 + \frac{2}{5}X_3 - \frac{1}{10}X_4 \dots\dots\dots (1) \\ X_1 = 6 - \frac{1}{5}X_3 + \frac{3}{10}X_4 \dots\dots\dots (2) \end{cases} \dots\dots\dots (3)$$

On remplace les valeurs de (X_1, X_2) de la relation (3) dans la fonction objectif (Z), on trouve :

$$\begin{aligned} Z &= 2X_1 + X_2 \\ &= 2 * [6 - \frac{1}{5}X_3 + \frac{3}{10}X_4] + [8 + \frac{2}{5}X_3 - \frac{1}{10}X_4] \\ &= 12 - \frac{2}{5}X_3 + \frac{6}{10}X_4 + 8 + \frac{2}{5}X_3 - \frac{1}{10}X_4 \\ &= 20 + \frac{5}{10}X_4 \\ Z - \frac{1}{2}X_4 &= 20 \end{aligned}$$

On place la résultat finale de la fonction (Z) dans le tableau de la solution optimale finale :

Variable de base	variable non basic				b_i
	X_1	X_2	X_3	X_4	
Z	0	0	0	$-\frac{1}{2}$	20
X_2	0	1	$-\frac{2}{5}$	$\frac{1}{10}$	8
X_1	1	0	$\frac{1}{5}$	$-\frac{3}{10}$	6

Toutes les coefficients de la fonction objectif (Z) est inférieure ou égale zéros (0) ($C_j \leq 0$) alors la solution finale est :

$$X_1 = 6, X_2 = 8, Z = 20$$

Remarque 2.5.1 Il est la même résultat par la méthode grande M

On déduire que la gestion de la production à décidé produire 6 unités de le produit (X_1) et 8 unités de le produit (X_2) pour le moindre cout productivété qui égale 20DA.

Chapitre 3

Dualité et interprétation économique

3.1 Introduction

La notion de dualité est un aspect très important de la programmation linéaire, elle a été introduite pour la première fois par Von Neumann en 1947, puis développée par Gale, Kuhn et Tucker en 1951. Les propriétés fondamentales des problèmes de dualité ont été définies par Goldman and Tucker en 1956.

Depuis, cette notion a suscité une attention très particulière dans plusieurs domaines tels que optimisation non convexe, programmation non linéaire, théorie des jeux ...

Dans un contexte d'entreprise, la résolution du primal (P) permet d'obtenir une utilisation optimale des ressources, quand au dual permet d'attribution de valeurs monétaires à ces ressources. Toutefois, l'on doit noter que la signification économique du dual n'est figée, elle diffère selon l'angle du vu et selon la nature du problème abordé.

D'un point de vue théorique, la notion de dualité permet :

- En résolvant le dual permet d'obtenir également la solution optimale du primal.
- Améliorer éventuellement le processus de résolution en résolvant l'un ou l'autre programme
- Elaborer d'autres techniques et algorithmes de résolution comme l'algorithme dual du simplexe qui est fondamentale dans l'étude de la programmation linéaire en nombres entiers et analyse post-optimal.

3.2 Dualité

Notons qu'à chaque programme linéaire primal (P) il existe un autre programme linéaire dual (D) et vice versa.

Les deux programmes contiennent les mêmes éléments A, b et c, mais arrangés de manière différente.

Problème primal :

$$\text{Max } c^T x$$

$$\text{s.c } Ax = b$$

$$x \geq 0$$

$$n \text{ variables, } m \text{ contraintes, } m < n, c, x \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{n \times m}$$

Problème dual :

$$\text{Min } b^T y$$

$$\text{s.c } A^T y \geq c$$

$$(y \text{ non restreint})$$

$$m \text{ variables, } n \text{ contraintes, } m < n, c \in \mathbb{R}^n, b, y \in \mathbb{R}^m, A \in \mathbb{R}^{n \times m}$$

3.2.1 Règles de construction du dual :

On sait qu'un problème de programmation linéaire pouvait se présenter sous la forme standard en introduisant des variables d'écart en vue de sa résolution avec la méthode de simplexe. Dans cette section nous allons parler de la forme canonique d'un programme linéaire qui est nécessaire pour l'élaboration du programme linéaire dual associé.

Forme canonique d'un programme linéaire :

On dit qu'un programme linéaire est sous forme canonique si :

- Toutes ses inégalités sont de type $\leq$ et si sa fonction objectif est de type max
- Toutes ses inégalités sont de type $\geq$ et si sa fonction objectif est de type min

Pour les programmes linéaires ayant des contraintes d'égalité =, il n'est pas en question de les changer dans sa forme canonique.

Après avoir mettre le primal sous la forme canonique , le tableau suivant récapitule les différent cas pour formuler le dual.

<i>Primal(P)</i>	<i>deviens</i>	<i>Dual(D)</i>
<i>la fonction objectif à max</i>	$\Leftrightarrow$	<i>la fonction objectif à min</i>
<i>Le nombre de variables = n</i>	$\Leftrightarrow$	<i>Le nombre de contraintes = n</i>
<i>Le nombre de contraintes = m</i>	$\Leftrightarrow$	<i>Le nombre de variables = m</i>
<i>Les coefficients économiques c_i</i>	$\Leftrightarrow$	<i>des seconds membres des contraintes b_i</i>
<i>Des seconds membres des contraintes b_i</i>	$\Leftrightarrow$	<i>des coefficients économiques c_i</i>
<i>Les coefficients technologiques a_{ij}</i>	$\Leftrightarrow$	<i>les coefficients technologiques a_{ji}</i>
<i>Une contraintes i de type =</i>	$\Leftrightarrow$	<i>la variable y_i est s.r.s</i>
<i>Une variable x_j s.r.s</i>	$\Leftrightarrow$	<i>la contraintes j de type =</i>

D'une autre manière, on peut formuler le dual à partir de son primal :

$$\begin{aligned}
 \text{Primal}(P) \left\{ \begin{array}{l} \max z = \sum_{j=1}^n c_j x_j \\ \sum_{j=1}^n a_{ij} x_j \leq b_i \quad i = 1..m \\ x_j \geq 0 \end{array} \right\} & \text{dual}(D) \left\{ \begin{array}{l} \min w = \sum_{i=1}^m b_i y_i \\ \sum_{i=1}^m a_{ji} y_i \geq c_j \quad j = 1..n \\ y_i \geq 0 \end{array} \right\} \\
 \text{Primal}(P) \left\{ \begin{array}{l} \min z = \sum_{j=1}^n c_j x_j \\ \sum_{j=1}^n a_{ij} x_j \geq b_i \quad i = 1..m \\ x_j \geq 0 \end{array} \right\} & \text{dual}(D) \left\{ \begin{array}{l} \max w = \sum_{i=1}^m b_i y_i \\ \sum_{i=1}^m a_{ji} y_i \leq c_j \quad j = 1..n \\ y_i \geq 0 \end{array} \right\}
 \end{aligned}$$

Exemple:

Formuler les programmes duals des PL primals suivant :

$$(P_1) \left\{ \begin{array}{l} \max z = x_1 + 2x_2 + 3x_3 \\ \text{s.c} \quad 2x_1 + x_2 + 3x_3 \leq 12 \\ \quad \quad x_1 - 4x_2 + x_3 \geq 8 \\ \quad \quad x_j \geq 0 \quad j = 1, 2, 3 \end{array} \right.$$

$$(P_2) \left\{ \begin{array}{l} \min \quad z = 2x_1 + x_2 - x_3 + x_4 \\ \text{s.c} \quad x_1 - x_2 + x_3 + x_4 \leq 45 \\ \quad \quad -2x_2 + 3x_3 \geq 30 \\ \quad \quad x_1 + 3x_4 = 20 \\ \quad \quad x_j \geq 0 \quad j = 1, 2, 3, 4 \end{array} \right.$$

Après avoir mis, ces deux primals sous la forme canonique, leurs programmes duals sont :

$$(D_1) \left\{ \begin{array}{l} \min w = 12y_1 - 8y_2 \\ \text{s.c} \quad 2y_1 - y_2 \geq 1 \\ \quad \quad y_1 + 4y_2 \geq 2 \\ \quad \quad 3y_1 - y_2 \geq 3 \\ \quad \quad y_j \geq 0 \quad j = 1, 2 \end{array} \right.$$

$$(D_2) \left\{ \begin{array}{l} \max w = -45y_1 + 30y_2 + 20y_3 \\ \text{s.c} \quad -y_1 + y_2 \leq 2 \\ \quad \quad -y_1 - 2y_2 \leq 1 \\ \quad \quad -y_1 + 3y_2 \leq -1 \\ \quad \quad -y_1 + 3y_3 \leq 1 \\ \quad \quad y_j \geq 0 \quad j = 1, 2, y_3 \text{ s.r.s} \end{array} \right.$$

3.2.2 Relations primal/dual :

Dualité faible :

Considérons la paire primale-duale :

$$\begin{array}{ll} \max C^T x & \min b^T y \\ \text{s.c } Ax = b & \text{s.c } A^T y \geq c \\ x \geq 0 & \end{array}$$

– Si x est une solution admissible du primal et y une solution admissible du dual, alors :

$$C^T x \leq b^T y$$

– S'il y a égalité, alors x est une solution optimale du primal et y une solution optimale du dual.

Dualité forte :

Considérons la paire primale-duale:

$$\begin{array}{ll} \text{Max } C^T x & \text{Min } b^T y \\ \text{s.c. } Ax = b & \text{s.c. } A^T y \geq c \\ x \geq 0 & \end{array}$$

– Si le primal et le dual admettent tous les deux une solution admissible, ils ont tous deux une solution optimale finie et la même valeur objectif optimale.

– Si le primal (dual) est non borné, le dual (primal) n'admet pas de solution admissible.

Complémentarité :

Considérons la paire primale-duale :

$$\begin{array}{ll} \text{Max } C^T x & \text{Min } b^T y \\ \text{s.c. } Ax = b & \text{s.c. } A^T y \geq c \\ x \geq 0 & \end{array}$$

Si x est une solution optimale du primal et y une solution optimale du dual, alors

$$x_i (a_i^T y - c_i) = 0$$

où a_i est la i -ème colonne de A .

En d'autres termes :

$$x_i > 0 \Rightarrow a_i^T y = c_i$$

$$a_i^T y > c_i \Rightarrow x_i = 0$$

-Le dual du dual est le primal .

Exemple :

(Résolution du dual par les règles de complémentarité).

★ Primal (P) :

$$\max z = 5x_1 + 12x_2 + 4x_3$$

$$s.c. x_1 + 2x_2 + x_3 \leq 10 \quad (y_1)$$

$$2x_1 - x_2 + 3x_3 = 8 \quad (y_2)$$

$$x_1, x_2, x_3 \geq 0$$

★ Dual (D) :

$$\min w = 10y_1 + 8y_2$$

$$s.c. y_1 + 2y_2 \geq 5 \quad (x_1)$$

$$2y_1 - y_2 \geq 12 \quad (x_2)$$

$$y_1 + 3y_2 \geq 4 \quad (x_3)$$

$$y_1 \geq 0$$

Solution optimale de (P) :

$$(x_1, x_2, x_3) = \left(\frac{26}{5}, \frac{12}{5}, 0\right)$$

$$Z = \frac{274}{5}$$

$$x_1 > 0 \Rightarrow y_1 + 2y_2 = 5$$

$$x_2 > 0 \Rightarrow 2y_1 - y_2 = 12$$

Solution optimale de (D) :

$$(y_1, y_2) = \left(\frac{29}{5}, \frac{-2}{5}\right)$$

$$w = \frac{274}{5}$$

3.2.3 Conditions d'optimalité primal-dual

L'ensemble primal-dual possède un certain nombre de conditions et de propriétés indissociables raccordées l'une à l'autre, précisées comme suit :

Lors de la résolution de deux programmes linéaires en dualité (un primal et un dual) un et un seul des 3 cas suivants se trouve réalisé.

Cas 1 :

L'un des deux Pl n'a pas de solution admissible , l'autre son optimum est rejeté à l'infini.

Illustration : soit le Pl suivant

$$\max Z = 2x_1 + 5x_2$$

$$-3x_1 + 2x_2 \leq 6$$

$$-x_1 + 2x_2 \geq 2$$

$$x_1 \geq 0, x_2 \geq 0$$

Le Pl admet une solution illimitée

Le dual du Pl : On multiplie la contrainte 2 par -1 pour avoir la forme $\leq$ (primal), on obtient

$$x_1 - 2x_2 \leq -2$$

Le dual :

$$\min Z' = 6y_1 - 2y_2$$

$$-3y_1 + y_2 \geq 2$$

$$2y_1 - 2y_2 \geq 5$$

$$y_1 \geq 0, y_2 \geq 0$$

En ajoutant la première contrainte à la deuxième :

$$-y_1 - y_2 \geq 7 \quad \text{impossible}$$

Cas 2 :

Ni le primal ni le dual n'ont de solutions admissibles.

Illustration : Soit le PL suivant

$$\begin{aligned}\max Z &= 2x_1 + x_2 \\ x_1 - x_2 &\leq 3 \\ -x_1 + x_2 &\leq -4 \\ x_1 &\geq 0, x_2 \geq 0\end{aligned}$$

Le dual $\rightarrow$

$$\begin{aligned}\min Z' &= 3y_1 - 4y_2 \\ y_1 - y_2 &\geq 2 \\ -y_1 + y_2 &\geq 1 \\ y_1 &\geq 0, y_2 \geq 0\end{aligned}$$

Le primal n'a pas de solution admissible, contrainte 1 + 2 entraîne $0 \leq -1$ impossible. Le dual non plus car contraintes (1) + contrainte (2) entraîne $0 \geq 3$ impossible. Contraintes contradictoires.

Cas 3 :

Le primal et le dual ont des solutions optimales finies, respectivement x^* et y^* . Alors à l'optimum, les valeurs des fonctions objectives du primal et du dual sont égales : $Z^* = Z'^*$

3.2.4 Algorithme dual du simplexe

Cette méthode est utilisée pour résoudre le problème dual à partir du primal sans avoir recours à des variables artificielles. En ajoutant des variables d'écarts, on obtient facilement la solution de base de départ.

Exemple :

Soit à résoudre le PI suivant

$$\text{Min} Z = x_1 + 2x_2$$

$$\text{S.C } x_1 - 2x_2 + x_3 \geq 4$$

$$2x_1 + x_2 - x_3 \geq 6$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0$$

étape 1 : mettre le problème sous la forme canonique (fonction max , contraintes $\leq$) puis ajouter des variables d'écarts pour avoir les contraintes en égalité.

$$\text{Max}(-Z) = -x_1 - 2x_2$$

$$\text{S.C } -x_1 + 2x_2 - x_3 + x_4 = -4$$

$$-2x_1 - x_2 + x_3 + x_5 = -6$$

$$X_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0$$

Solution de base de départ : $x_1 = x_2 = x_3 = 0, x_4 = -4, x_5 = -6$. Solution non réalisable.

Tableau initial :

C_j	X_j	-1	-2	0	0	0	b
		x1	x2	x3	x4	x5	
CB	xB						
0	x4	-1	2	-1	1	0	-4
0	x5	-2	-1	1	0	1	-6
Zj-Cj		1	2	0	0	0	Z=0

La solution de départ est non réalisable (variables de base < 0) mais optimale (Tous les $Z_j - C_j \geq 0$). On essayera dans les étapes suivantes de parvenir à une solution réalisable sans perdre l'optimalité.

étape 2 : Choix de la variable sortante de la base. Identifier la variable de base qui est la plus négatif et la choisir comme variable sortante. Critère

$$X_s = \min i(XB_i, XB_i < 0) \quad i = 1, \dots, m.$$

$$X_s = \min(XB_1, XB_2) = \min(-4, -6) = -6$$

correspond à la variable x_5 qui sort.

étape 3 : Choix de la variable qui entre en base X_e . Critère

$$\theta = \min_j \left(\frac{Z_j - C_j}{-X_{ij}}, X_{ij} < 0 \right) \quad j = 1, \dots, n$$

en ignorant toute les colonnes j avec $X_{ij} \geq 0$. X_{ij} étant le vecteur ligne relatif à la variable X_i sortante.

Dans le cas de l'exemple c'est la variable x_1 qui entre.

$$\theta = \min \left(\frac{Z_1 - C_1}{-X_{21}}, \frac{Z_2 - C_2}{-X_{22}} \right) = \min \left(\frac{1}{2}, \frac{2}{1} \right).$$

étape 4 : Faire le pivotage (changement de la base).

Même façon que la méthode du simplexe. Répéter les étapes 2 , 3 et 4 jusqu'à ce que le tableau de droite devient non négatif autrement dit jusqu'à ce que la solution devienne réalisable.

Tableau 1:

$\begin{matrix} C_j \\ X_j \end{matrix}$		-1	-2	0	0	0	B
		x1	x2	x3	x4	x5	
CB	xB						
0	x4	0	5/2	-3/2	1	-1/2	-1
-1	x1	1	1/2	-1/2	0	-1/2	3
Zj-Cj		0	3/2	1/2	0	1/2	Z=-3

$$L1 = L1 + L2$$

X_4 sort de la base $.x_4 = -1$

$$\theta = \min \left(\frac{Z_3 - C_3}{-x_{13}}, \frac{Z_5 - C_5}{-x_{15}} \right) = \min \left(\frac{1}{3}, 1 \right)$$

X_3 entre dans la base

Tableau 2:

C_j X_j		-1	-2	0	0	0	b
		x1	x2	x3	x4	x5	
CB	xB						
0	x3	0	-5/3	1	-2/3	1/3	2/3
-1	x1	1	-1/3	0	-1/3	-1/3	10/3
Zj-Cj		0	7/3	0	1/3	1/3	Z=-10/3

Critère d'arrêt satisfait tous les b_i sont positifs. La solution obtenue est optimale et réalisable avec $X_1^* = \frac{10}{3}$, $X_2^* = 0$, $X_3^* = \frac{2}{3}$, $X_4^* = 0$, $X_5^* = 0$ et $Z^* = -\frac{10}{3}$, Soit le

$$\min Z = \frac{10}{3}$$

3.3 Interprétation économiques des variables duales

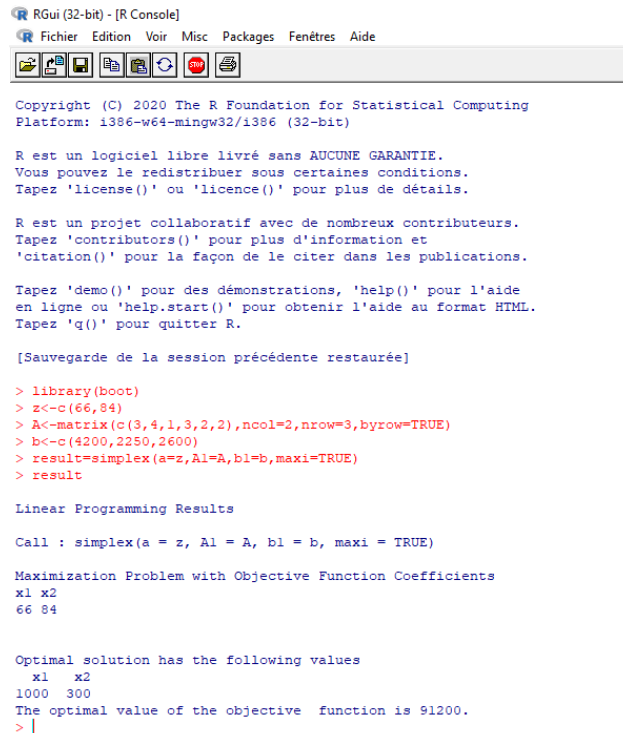
Notons que la signification des variables duales n'est pas toujours figée; elle diffère selon la nature et le contexte du problème à résoudre. Sur le plan économique, le programme primal consiste à l'utilisation optimale des ressources quand au programme dual modélise la situation d'un concurrent qui veut acquérir ces ressources. Pour mieux comprendre la notion de dualité, partons d'un problème réel, une entreprise fabrique deux modèles de téléviseurs TV Slim et TV Ultra. L'assemblage se fait en utilisant 3 types de ressources :Assembleur A, et Assembleur B et vérification . Le problème linéaire qui maximise son profit est le PL suivant :

$$\begin{aligned}
 \max z &= 66x_1 + 84x_2 \quad \text{profit total} \\
 3x_1 + 4x_2 &\leq 4200 \quad \text{Assembleur A} \\
 x_1 + 3x_2 &\leq 2250 \quad \text{Assembleur B} \\
 2x_1 + 2x_2 &\leq 2600 \quad \text{vérification} \\
 x_1, x_2 &\geq 0
 \end{aligned}$$

Avec x_1 et nombre de TV Slim et x_2 celui TV Ultra, et les contraintes représentent les capacités des ressources.La première contrainte représente les limite quotidienne sur

le fonctionnement de l'assembleur A (en heure) où un TV Slim exige 3h, et un TV Ultra exige 4h pour être assemblé. De même, la deuxième contrainte décrit les limites sur l'utilisation de l'assembleur B, quand la troisième contrainte montre les limites horaires sur la vérification et emballage en heures.

Donc pour résoudre ce problème linéaire on va utiliser le logiciel R et on obtient les résultats suivants :



```
RGui (32-bit) - [R Console]
Fichier Edition Voir Misc Packages Fenêtres Aide

Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R est un logiciel libre livré sans AUCUNE GARANTIE.
Vous pouvez le redistribuer sous certaines conditions.
Tapez 'license()' ou 'licence()' pour plus de détails.

R est un projet collaboratif avec de nombreux contributeurs.
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

[Sauvegarde de la session précédente restaurée]

> library(boot)
> z<-c(66,84)
> A<-matrix(c(3,4,1,3,2,2),ncol=2,nrow=3,byrow=TRUE)
> b<-c(4200,2250,2600)
> result=simplex(a=z,A1=A,b1=b,maxi=TRUE)
> result

Linear Programming Results

Call : simplex(a = z, A1 = A, b1 = b, maxi = TRUE)

Maximization Problem with Objective Function Coefficients
x1 x2
66 84

Optimal solution has the following values
x1 x2
1000 300
The optimal value of the objective function is 91200.
> |
```

La solution optimale de ce modèle est

$$x_1^* = 1000, x_2^* = 300 \text{ et } z^* = 91200$$

La formulation du dual nous conduit à inciter l'entreprise à ajouter des heures supplémentaires sur ces ressources. Mais Combien l'entreprise devrait-elle payer ces heures? Combien pourrait-elle gagner? Conformément aux résultats trouvés précédemment, si on augmente la capacité de l'assembleur A d'une seule heure, l'entreprise aura $y_1 = 18$ u.m de gain supplémentaire le montant du bénéfice sera majoré de 18, il passe de 91200 à $91200+18=91218$ u.m, l'entreprise n'a pas intérêt d'augmenter la capacité de l'assembleur B car cette dernière reste oisive pendant 350 heures, si on augmente

la capacité de vérification d'une seule heure, le montant du bénéfice sera majoré $y_3 = 6$ u.m, il passe de 91 200 à $91\,200 + 6 = 91\,206$ u.m.

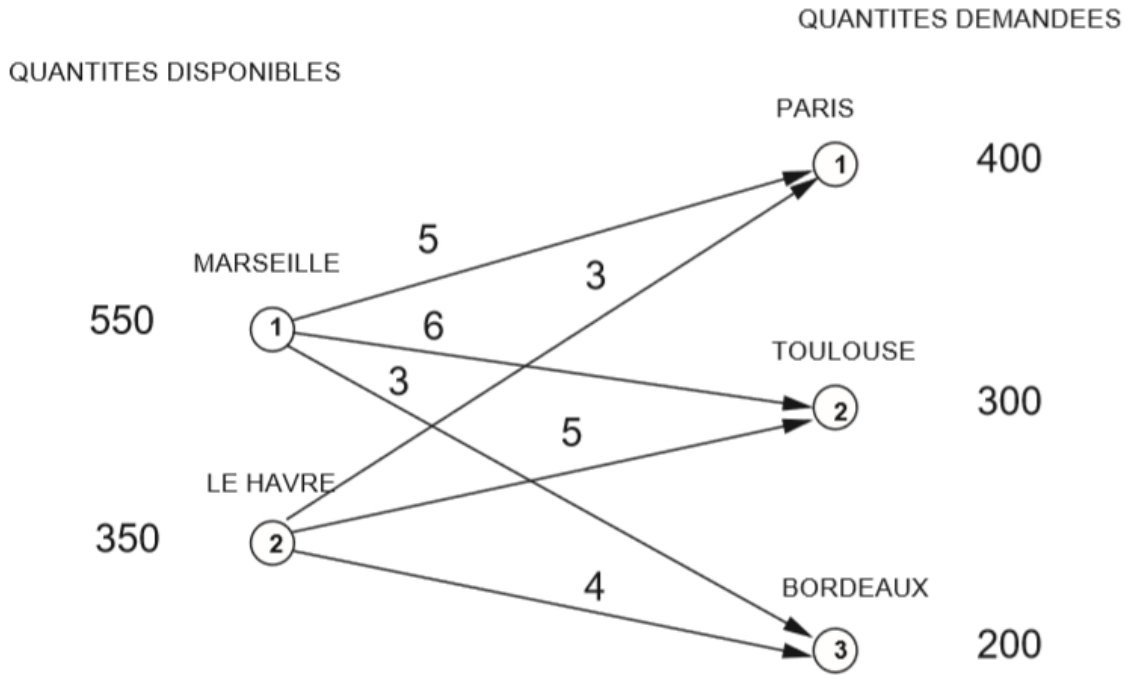
En outre, le programme dual peut décrire la situation d'un concurrent qui veut racheter cette entreprise. Le gérant de cette entreprise décide de ne pas utiliser les 3 ressources, il envisage de les vendre, mais il ne connaît pas leurs valeurs monétaires? Cependant, le gérant de cette entreprise exigera évidemment un montant qui sera supérieure au profit généré par ces ressources. Ainsi, le concurrent doit imputer des valeurs monétaires pour chaque unité des ressources considérées en formulant le dual (D). Ainsi, y_1 est le prix de soumission d'une heure de travail de l'assembleur A, et y_2 est le prix de soumission d'une heure de travail de l'assembleur B et y_3 est le prix de soumission d'une heure de vérification. Cet acheteur doit proposer un prix au moins égal à

$$w = 4200y_1 + 2250y_2 + 2600y_3$$

pour acquérir les 3 ressources et doit être minimisé. Très important : Dans certaines situations réelles où on peut les décrire à travers d'un programme linéaire en vue d'optimisation, son dual pourra de ne pas avoir aucune interprétation.

3.4 Autre application numérique

Voici des valeurs numériques relatives au problème de transport:



Le problème linéaire qui minimise son profit est le PL suivant :

$$\text{Min} z = 5x_{11} + 6x_{12} + 3x_{13} + 3x_{21} + 5x_{22} + 4x_{23}$$

sous les contraintes:

$$x_{11} + x_{12} + x_{13} \leq 550$$

$$x_{21} + x_{22} + x_{23} \leq 350$$

$$-x_{11} - x_{21} \leq -400$$

$$-x_{12} - x_{22} \leq -300$$

$$-x_{13} - x_{23} \leq -200$$

$$x_{ij} \geq 0 \text{ pour } i = 1, 2, \text{ pour } j = 1, 2, 3.$$

Le problème primal exprime le point de vue du constructeur qui cherche à minimiser ses coûts de production.

Le problème dual s'écrit ainsi :

$$\text{Max} z = -550u_1 - 350u_2 + 400v_1 + 300v_2 + 200v_3$$

Sous les contraintes

$$v_1 - u_1 \leq 5$$

$$v_2 - u_1 \leq 6$$

$$v_3 - u_1 \leq 3$$

$$v_1 - u_2 \leq 3$$

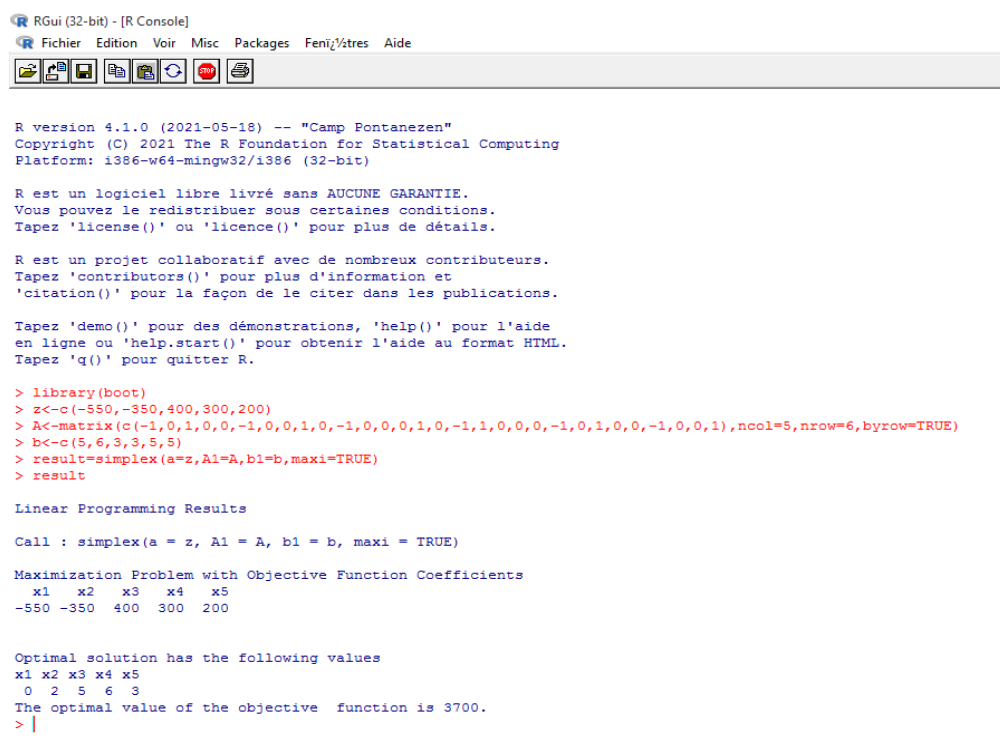
$$v_2 - u_2 \leq 5$$

$$v_3 - u_2 \leq 5$$

$$u_i \geq 0 \text{ pour } i = 1, 2, v_j \geq 0 \text{ pour } j = 1, 2, 3$$

Le problème dual exprime le point de vue du transporteur qui veut maximiser son profit. Ses variables sont les prix d'achat à Marseille (u_1) et au Havre (u_2) et ses prix de vente à Paris (v_1), Toulouse (v_2) et Bordeaux (v_3). Les quantités qu'il doit acheter et vendre sont fixées. Les contraintes du problème dual expriment que les prix sont compétitifs, c'est à dire acceptables pour le constructeur.

Donc pour solver ce problème lineaire on va utiliser le logiciel R et on obtient les résultats suivants :



```
RGui (32-bit) - [R Console]
Fichier Edition Voir Misc Packages Fenêtres Aide

R version 4.1.0 (2021-05-18) -- "Camp Pontanezen"
Copyright (C) 2021 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R est un logiciel libre livré sans AUCUNE GARANTIE.
Vous pouvez le redistribuer sous certaines conditions.
Tapez 'license()' ou 'licence()' pour plus de détails.

R est un projet collaboratif avec de nombreux contributeurs.
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

> library(boot)
> z<-c(-550,-350,400,300,200)
> A<-matrix(c(-1,0,1,0,0,-1,0,0,1,0,-1,0,0,0,1,0,-1,1,0,0,0,-1,0,1,0,0,-1,0,0,1),ncol=5,nrow=6,byrow=TRUE)
> b<-c(5,6,3,3,5,5)
> result=simplex(a=z,A1=A,b1=b,maxi=TRUE)
> result

Linear Programming Results

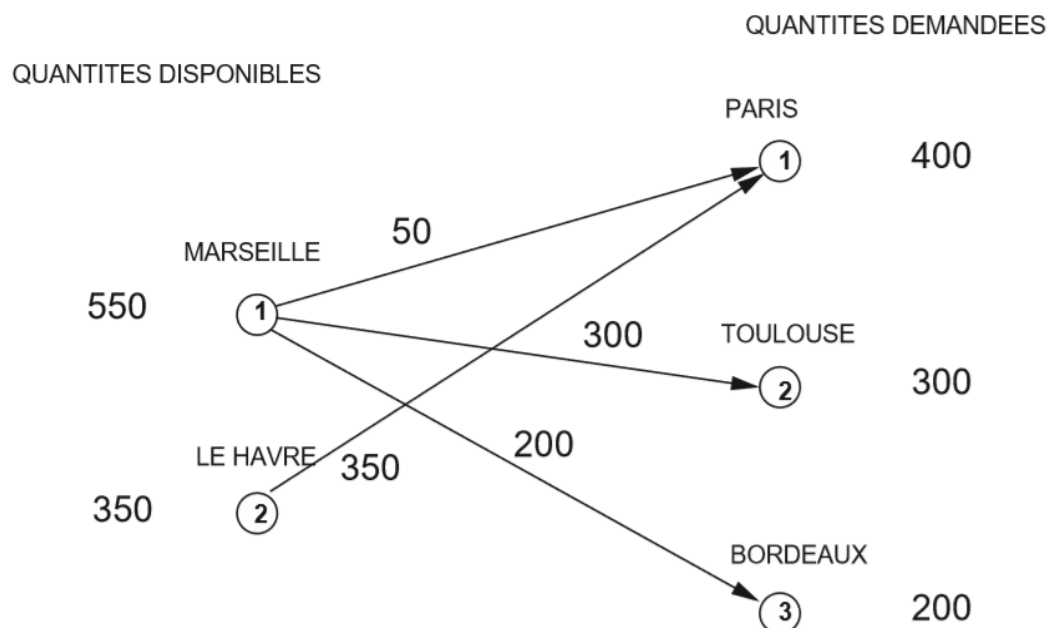
Call : simplex(a = z, A1 = A, b1 = b, maxi = TRUE)

Maximization Problem with Objective Function Coefficients
  x1  x2  x3  x4  x5
-550 -350 400 300 200

Optimal solution has the following values
 x1 x2 x3 x4 x5
  0  2  5  6  3
The optimal value of the objective function is 3700.
> |
```

La valeur optimale du critère est 3700 \$.

Elle correspond au graphe d'approvisionnement suivant :



Chapitre 4

Méthode du simplexe révisé

4.1 Introduction

Cette méthode a été mise au point par Dantzig, Orden et Wolfe à la “Rand Corporation” au début des années 1950. Les principes du simplexe révisé sont les mêmes que ceux de la méthode conventionnelle du simplexe. La nouveauté provient du fait que la procédure révisée utilise avec les coefficients originaux, alors que la procédure conventionnelle du simplexe transforme ces coefficients au fur et à mesure des itérations. Il en résulte un gain de place lors du stockage des coefficients dans la mémoire d’un ordinateur et, par conséquent, un gain de temps appréciable. D’autre part, le nombre total d’itération est généralement plus faible avec la méthode du simplexe révisé qu’avec la méthode originale du simplexe.

4.2 Méthode du simplexe

4.2.1 Définition

La méthode du simplex est une procédure itérative permettant d’effectuer une exploration dirigée de l’ensemble des solutions réalisables de base.

L’application de la méthode nécessite la connaissance d’une solution réalisable de base, au départ.

La méthode consiste à calculer à chaque itération un programme (une solution réalisable) « voisin » de celui qui vient d'être calculée « au moins aussi bon » que celui-ci.

On peut aussi s'assurer, moyennant certaines précautions, que la même base ne puisse jamais apparaître dans deux itérations distinctes, ce qui suffit à assurer la convergence du procédé.

4.2.2 Intérêt de la méthode du simplexe

Converger vers une solution de base réalisable optimale existe.

Vérifier la compatibilité de l'équation sous la redondance du système.

Savoir si le problème est possible ou non, dans l'affirmative.

Trouver une solution réalisable de base initiale.

Mettre en évidence l'absence de solution réalisable optimale finie.

Exemple :

Une entreprise veut déménager son matériel composé de 450 machines de trois types : M1, M2, M3. Elle décide de louer des camions. La société de location dispose de trois sortes de véhicules : V1, V2, V3 dont les tarifs sont respectivement 500, 800 et 1200 DA pour un voyage.

Le camion V1 peut transporter une machine M1, 4 machines M2 et 10 machines M3. Pour des raisons techniques la place d'une machine d'un type donnée ne peut être utilisée pour une machine d'un autre type. Chaque camion V2 peut transporter 2 machines M1, une machine M2 et 20 machines M3.

On veut transporter en un seul convoi 30 machines M1, 120 machines M2 et 300 machines M3.

L'entreprise veut déterminer le nombre de véhicules à louer pour minimiser le coût total de transport, donner le modèle linéaire de ce problème.

Solution :

Pour exprimer ce problème en tant que PL on considère l'énoncé suivant :

Pour ce problème il y'a 3 variables de décision :

X_1 Pour le véhicule 1

X_2 Pour le véhicule 2

X_3 Pour le véhicule 3

► Pour les contraintes on a :

◆ Pour la machine M1 :

$$X_1 + 2X_2 + 4X_3 \geq 30$$

◆ Pour la machine M2 :

$$4X_1 + 6X_2 + 20X_3 \geq 120$$

◆ Pour la machine M3 :

$$10X_1 + 20X_2 + 24X_3 \geq 300$$

Avec X_1, X_2, X_3 sont supérieures ou égales à zéro.

► La fonction objective à optimiser se rapporte au profit P et elle est à minimiser :

$$\min F = 500X_1 + 800X_2 + 1200X_3$$

La fonction est linéaire par rapport aux variables de décision.

La solution par logiciel python :

C'est un langage de script de haut niveau, interprété, interactif et orienté objet. Python est conçu pour être très lisible. Il utilise fréquemment des mots-clés anglais alors que d'autres langues utilisent la ponctuation, et il a moins de constructions syntaxiques que d'autres langues.

```
import numpy as np
import scipy as sp
F = [500,800,1200]
A = [[1,2,4], [4,6,20], [10,20,24]]
b = [30,120,300]
x0_bounds = (0, None)
x1_bounds = (0, None)
x2_bounds = (0, None)
from scipy.optimize import linprog
# Solve the problem by Simplex method in Op- timization
res = linprog(F, A_ub=A, b_ub=b, bounds=(x0_bounds,x1_bounds,x2_bounds),method='simplex',options={"disp": True})
print(res)
```

```
Optimization terminated successfully.
      Current function value: 0.000000
      Iterations: 7
      con: array([], dtype=float64)
      fun: 0.0
message: 'Optimization terminated successfully.'
      nit: 7
      slack: array([ 30., 120., 300.])
      status: 0
      success: True
           x: array([0., 0., 0.])
```

► Pour le min F:

$$\min F = 500x_1 + 800x_2 + 1200x_3$$

S/C:

$$x_1 + 2x_2 + 4x_3$$

$$4x_1 + 6x_2 + 20x_3$$

$$10x_1 + 20x_2 + 24x_3$$

$$x_1, x_2, x_3 \geq 0$$

La forme standard:

$$x_1 + 2x_2 + 4x_3 - e_1 + A_1 = 30$$

$$4x_1 + 6x_2 + 20x_3 - e_2 + A_2 = 120$$

$$10x_1 + 20x_2 + 24x_3 - e_3 + A_3 = 300$$

$$x_1, x_2, x_3 \geq 0$$

avec

$$\min F = 500x_1 + 800x_2 + 1200x_3 + 0e_1 + 0e_2 + 0e_3 + MA_1 + MA_2 + MA_3$$

			500	800	1200	0	0	0	M	M	M
	Var base	Quantité	x_1	x_2	x_3	e_1	e_2	e_3	A_1	A_2	A_3
M	A_1	30	1	2	4	-1	0	0	1	0	0
M	A_2	120	4	6	20	0	-1	0	0	1	0
M	A_3	300	10	20	24	0	0	-1	0	0	1
			15M	28M	48M	-M	-M	-M	M	M	M
			15M - 500	28M - 800	48M - 1200	-M	-M	-M	M	M	M

Quand on prend $(x_1, x_2, x_3) = (0, 0, 0) \rightarrow -e_1 = 30, -e_2 = 120, -e_3 = 300$

Ce qui ne confirme avec la non négativité de variable car on a les variables sont positive toujours donc on ajoute des variables artificielles (M) $M \rightarrow +\infty$

La base réalisable :

$$x_1 = 0, x_2 = 0, x_3 = 0$$

$$e_1 = 0, e_2 = 0, e_3 = 0$$

$$A_1 = 30, A_2 = 120, A_3 = 300$$

On fait les calculs sur le tableau suivant :

			500	800	1200	0	0	0	M	M	M
	<i>Var base</i>	<i>Qantité</i>	x_1	x_2	x_3	e_1	e_2	e_3	A_1	A_2	A_3
M	A_1	30	1	2	4	-1	0	0	1	0	0
M	A_2	120	4	6	20	0	-1	0	0	1	0
M	A_3	300	10	20	24	0	0	-1	0	0	1
			$15M$	$28M$	$48M$	$-M$	$-M$	$-M$	M	M	M
			$15M - 500$	$28M - 800$	$48M - 1200$	$-M$	$-M$	$-M$	M	M	M

La solution optimale de notre PL=12564.

La solution de base est $Z = (0, 12, 18, 2, 35)$.

4.3 Méthode du simplexe révisé

4.3.1 Définition :

Le simplexe révisé se présente sous deux formes différentes, selon que l'on ajoute des variables d'écart ou des variables artificielles. Pour la première formulation, nous supposons qu'une matrice identité est présente après l'ajout éventuel de variables d'écart. Si des variables artificielles sont ajoutées, on utilise la seconde formulation qui correspond à la méthode des deux phases.

4.3.2 L'algorithme du simplexe révisé :

Est un algorithme de résolution des problèmes d'optimisation linéaire.

Il a été introduit par George Dantzig à partir de 1947 (l'algorithme permet de minimiser ou maximiser une fonction objective, qui est elle aussi linéaire).

4.3.3 Principe (algorithme du simplexe révisé) :

Soit le modèle de programme linéaire suivant (forme standard) qui comporte n variables (variables de décision et variables d'écart) et m contraintes fonctionnelles :

$$\text{Max ou Min}(z) = c_1x_1 + c_2x_2 + \dots + c_ix_j + \dots + c_nx_n.$$

Sous contraintes :

$$\left(\begin{array}{l} a_{11}X_1 + a_{12}X_2 + \dots + a_{1n}X_n = b_1 \\ a_{21}X_1 + a_{22}X_2 + \dots + a_{2n}X_n = b_2 \\ \dots \\ a_{m1}X_1 + a_{m2}X_2 + \dots + a_{mn}X_n = b_m \\ X_j \geq 0 \ j = 1..n, \ b_i \geq 0, \ i = 1..m \end{array} \right)$$

Ce modèle s'écrit, sous forme matricielle :

$$\max(\min)Z = CX$$

$$S.C \quad : \quad AX = b$$

$$X \geq 0$$

avec:

$$A = \left[\begin{array}{c} a_{11}, a_{12}, \dots, a_{1j}, \dots, a_{1n} \\ a_{21}, a_{22}, \dots, a_{2j}, \dots, a_{2n} \\ \dots \\ a_{m1}, a_{m2}, \dots, a_{mj}, \dots, a_{mn} \end{array} \right]$$

$$C = (C_1, \dots, C_n)$$

$$X = \left(\begin{array}{c} x_1 \\ \cdot \\ \cdot \\ x_j \\ \cdot \\ \cdot \\ x_n \end{array} \right)$$

Simplexe révisé

```

{ Données : une partition  $\llbracket B \mid N \rrbracket$  de  $\mathbb{R}^n$ , une représentation }
{ de la matrice  $B^{-1}$  ainsi qu'une solution réalisable  $[x_B, 0]$ . }
optimal  $\leftarrow$  faux
non_borné  $\leftarrow$  faux
répéter
  { Production d'un vecteur de coûts réduits }
   $\pi \leftarrow \text{SOLUTION}(\pi B = c_B)$ 
   $z_N \leftarrow \pi N$ 
   $d \leftarrow c_N - z_N$ 

  { Choix d'une variable entrante }
  si ( $d > 0$ ) alors optimal  $\leftarrow$  vrai
  sinon
     $\bar{j} \leftarrow \text{Calcule\_}\bar{j}$ 
     $\delta x \leftarrow \text{SOLUTION}(B \delta x = N_j)$ 
    { Choix d'une variable sortante }
    si ( $\delta x < 0$ ) alors non_borné  $\leftarrow$  vrai
    sinon
       $\bar{i} \leftarrow \arg \min_{i: \delta x_i > 0} (\frac{\delta x_i}{\delta x_i})$ 
      { Mises-à-jour diverses }
       $\llbracket B \mid N \rrbracket \leftarrow \llbracket B \cup \{j\} \setminus \{i\} \mid N \cup \{i\} \setminus \{j\} \rrbracket$ 
       $B^{-1} \leftarrow \text{Mise\_à\_jour}(B^{-1}, \llbracket B \mid N \rrbracket, i, j)$ 
    fin
  fin
jusqu'à ( optimal  $\vee$  non_borné )

```

En pratique, les deux résolutions de systèmes linéaires sont efficaces car la matrice B est représentées par une décomposition appropriée à la solution. Cette décomposition est donc mise à jour en même temps que la partition des composantes ; dans l'algorithme, nous avons nommé cette représentation B^{-1} , mais nous verrons qu'il s'agit plutôt d'une décomposition de la matrice $B = LU$

Exemple (par la main) :

Reprenons le problème sous la forme canonique :

$$\max z = 5x_1 + 4x_2$$

S/C:

$$x_1 + x_2 = 6$$

$$\frac{1}{4}x_1 + x_2 + x_4 = 6$$

$$3x_1 + 2x_2 + x_5 = 22$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0$$

Le sommet $x = (4 ; 5 ; 2 ; 0 ; 0)$ correspond aux variables de base $\{x_1, x_2, x_3\}$. De même, le sommet

$Y = (6 ; 2 ; 0 ; 2.5 ; 0)$ est associé aux variables de base $\{x_1, x_2, x_4\}$. Les deux sommets sont adjacents ce qui est conforme au graphique de l'ensemble K projeté dans $\mathbb{R}^2$.

Le système s'écrit :

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ \frac{1}{4} & 1 & 0 & 1 & 0 \\ 3 & 2 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \\ 22 \end{bmatrix}$$

Pour calculer la solution de base (4 ; 5 ; 2 ; 0 ; 0), il suffit d'extraire les 3 colonnes de la matrice A et de résoudre le système carré par la méthode d'élimination de Gauss. Toutefois, lorsque quel'on voudra calculer la nouvelle solution de base (6 ; 2 ; 0 ; 2.5 ; 0), il faudra recommencer l'élimination de Gauss avec les nouvelles colonnes de base. Il est plus avantageux de poursuivre l'élimination de Gauss à partir du premier calcul.

Voici un exemple de calcul.

a) En premier, on forme la matrice augmentée :

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 6 \\ \frac{1}{4} & 1 & 0 & 1 & 0 & 6 \\ 3 & 2 & 0 & 0 & 1 & 22 \end{bmatrix}$$

b) On applique l'élimination de Gauss-Jordan pour les variables de base $\{x_1, x_2, x_3\}$:

$$\begin{bmatrix} 1 & 0 & 0 & \frac{-4}{5} & \frac{2}{5} & 4 \\ 0 & 1 & 0 & \frac{6}{5} & \frac{-1}{10} & 5 \\ 0 & 0 & 1 & \frac{4}{5} & \frac{-2}{5} & 2 \end{bmatrix}$$

Donc:

$$\begin{aligned} x_1 &= 4 + \frac{4}{5}x_4 - \frac{2}{5}x_5 \\ x_2 &= 5 - \frac{6}{5}x_4 + \frac{1}{10}x_5 \\ x_3 &= 2 - \frac{4}{5}x_4 + \frac{2}{5}x_5 \end{aligned}$$

En posant les variables hors-bases $x_4=0, x_5=0$, on obtient bien la solution de base $x = (4 ; 5 ; 2 ; 0 ; 0)$.

c) Maintenant, on désire calculer la solution de base adjacente liées aux variables de base $\{x_1, x_2, x_4\}$

Pour cela, on poursuit l'élimination de Gauss-Jordan à partir du pivot $a_{3,4}$.

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 6 \\ 0 & 1 & -\frac{3}{2} & 0 & \frac{1}{2} & 2 \\ 0 & 0 & \frac{5}{4} & 1 & -\frac{1}{2} & \frac{5}{2} \end{bmatrix}$$

Donc:

$$\begin{aligned} x_1 &= 6 - x_3 \\ x_2 &= 2 + \frac{3}{2}x_3 - \frac{1}{2}x_5 \\ x_3 &= \frac{5}{2} - \frac{5}{4}x_3 + \frac{1}{2}x_5 \end{aligned}$$

En posant les variables hors-bases $x_3=0, x_5=0$, on obtient bien la solution de base $Y = (6; 2; 0; 2.5; 0)$.

d) Pour suivons à un autre sommet adjacent

$Z = (6; 0; 0; 4.5; 4)$ dont les variables de basesont $\{x_1, x_4, x_5\}$. Ce sommet est adjacent à y mais pas à x. Poursuivons l'éliminationde Gauss-Jordan à partir du pivot $a_{2,5}$.

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 6 \\ 0 & 2 & -3 & 0 & 1 & 4 \\ 0 & 1 & -\frac{1}{4} & 1 & 0 & \frac{9}{2} \end{bmatrix}$$

On obtient les relations:

$$\begin{aligned} x_1 &= 6 - x_3 \\ x_2 &= 4 - 2x_2 + 3x_3 \\ x_3 &= \frac{9}{2} - x_2 + \frac{1}{4}x_3 \end{aligned}$$

En posant les variables hors-bases $x_2 = x_3 = 0$, on obtient bien la solution de base $= (6; 0; 0; 4.5; 4)$.

L'opération décrite ci-dessus est aussi connue sous le nom de pivotement. Cette stratégie sera à la base de la méthode du simplexe révisé.

Exemple (par python) :

Reprenons le problème sous la forme canonique :

$$\max F = 4x_1 + 12x_2 + 3x_3$$

S/C:

$$x_1 + x_4 = 1000$$

$$x_2 + x_5 = 500$$

$$x_3 + x_6 = 1500$$

$$3x_1 + 6x_2 + 2x_3 + x_7 = 6750$$

$$x_1, x_2, x_3, x_4, x_5, x_6, x_7 \geq 0$$

La solution de base : (4,12 ,3 , 0, 0, 0 ,0).

```
File Edit Format Run Options Window Help
# coding=utf8
import numpy as np # nécessaire pour avoir le calcul numérique

# méthode du simplexe pour résoudre
# max c.x sous contraintes
# Ax=b
# x >= 0
# permut(0..m-1) : variables dans la base initiale
# permut(m..n-1) : variables hors base initiale
# ne fonctionne que si la base initiale est admissible
# et les composantes de b >=0
# renvoie la solution et le gain maximal
def simplexe(A,b,c,permut):
    # m: nombre de variables dans la base
    # n: nombres d'inconnues du pb d'optimisation linéaire
    m,n = A.shape
    if m>n or c.shape[0] != n or b.shape[0] != m:
        return 'dimensions incompatibles'

    if min(b) < 0:
        return 'le vecteur b doit être >=0'

    while True : # Accumulation du do-while en python...
        # matrice des colonnes permutes
        Ap=np.column_stack((A[:,permut[i]] for i in range(n)))
        # coefficients du gain permutes
        cp=np.array([c[permut[i]] for i in range(n)])

        # vérification que le problème n'est pas dégénéré
        # en python ":" désigne le range "0,1,...,m-1",
        # "m:" le range "m,m+1,...,n-1" et ":" le range "0,1,...,n-1"
        if np.linalg.det(Ap[:m,:m]) == 0:
            return 'matrice non inversible'

        # expression des variables de base en
        # fonctions des variables hors base
        # x_b = Chb x_hb + bbase
        invAp=np.linalg.inv(Ap[:m,:m])
        Chb=np.dot(invAp,Ap[:m,m:])
        bbase=np.dot(invAp,b)
```

```

# coefficients du gain dans les variables hors base
cbase=-np.dot(cp[:m],Chb)+cp[m:]
cmax=max(cbase)
# si tout les coeffs sont <0 on ne peut plus améliorer
if cmax<=0:
    break # sortie du do-while
# sinon choix de la variable optimale pour le gain
# cette variable rentrera dans la base
ihb=np.argmax(cbase)+m
# choix de la variable qui s'annule en premier
# quand la variable optimale augmente
# cette variable sortira de la base
xrmax=np.array([bbase[i]/Chb[i][ihb-m] for i in range(m)])
vmax=max(xrmax)
# on met les valeurs négatives à une valeur grande
for i in range(m):
    if xrmax[i]<= 0:
        xrmax[i]=vmax+1

# recherche de l'indice de "première sortie"
ib=np.argmin(xrmax)
print ('out=',permut[ib], ' in=',permut[ihb])

# actualisation de la permutation
permut[ib],permut[ihb] = permut[ihb],permut[ib] # swap en python
# fin du do while

# fin de l'algorithme
# on complète le vecteur des variables
# dans la base par des zéros
xp=np.hstack((bbase,np.zeros(n-m)))
# prise en compte de la permutation
x=np.empty(n)
for i in range(n):
    x[permut[i]]=xp[i]
# renvoie la solution et le gain
return x,np.dot(c,x)
# fin de la fonction simplexe

# matrice des contraintes Ax=b
A=np.array([[1,0,0,1,0,0,0],[0,1,0,0,1,0,0],[0,0,1,0,0,1,0],[3,6,2,0,0,0,1]])
# second membre des contraintes Ax=b
b=np.array([1000,500,1500,6750])
# coefficients de la fonction à maximiser
c=np.array([4,12,3,0,0,0,0])
# base initiale
# permutation contenant les m variables dans la base (x>=0) puis
# les variables hors base (n-m variables nulles)
permut=np.array([6,5,4,3,2,1,0])
print (simplexe(A,b,c,permut))
|

```

La solution:

```

File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/HP/Desktop/RO et Logiciels/ro.py =====

out= 4    in= 1
out= 3    in= 0
out= 6    in= 2
out= 5    in= 3
(array([ 250.,  500., 1500.,  750.,    0.,    0.,    0.]), 11500.0)
>>> |

```

On obtient bien la solution de base $Z = (250, 500, 1500, 750, 0, 0, 0)$, et la solution optimale est :

PL=11500.

4.4 Conclusion

L'algorithme du simplexe est un algorithme de résolution des problèmes d'optimisation linéaire.. C'est probablement le premier algorithme permettant de minimiser une fonction sur un ensemble défini par des inégalités. De ce fait, il a beaucoup contribué au démarrage de l'optimisation numérique. L'algorithme du simplexe a longtemps été la méthode la plus utilisée pour résoudre les problèmes d'optimisation linéaire. il est concurrencé par les méthodes de points intérieurs, mais garde une place de choix dans certaines circonstances (en particulier si l'on a une idée des contraintes d'inégalité active en la solution). Le nom de l'algorithme est dérivé de la notion de simplexe et a été suggéré par Motzkin. En réalité, l'algorithme n'utilise pas de simplexes, mais certaines interprétations de l'ensemble admissible du problème renvoient au concept de simplexe.

Chapitre 5

La théorie des jeux et la programmation dynamique

5.1 Introduction

La recherche opérationnelle (RO) est la discipline des mathématiques appliquées qui traite des questionnes d'utilisation optimale des ressources dans l'industrie et dans le secteur public.

Depuis une dizaine d'années le champ d'application de la RO s'est élargi à des domaines comme l'économie, la finance, le marketing et la planification d'entreprise. Plus récemment, la RO a été utilisé la gestion des systèmes de la santé et d'éducation, pour la résolution de problème environnementaux et dans d'autres domaines d'intérêt public.

La RO vise à résoudre certains problèmes par des méthodes et des techniques scientifiques parmi eux : la théorie de jeux et la programmation dynamique que nous allons savoir dans notre recherche.

5.2 Définitions

5.2.1 La théorie des jeux

La théorie des jeux consiste à étudier les situations de conflits qui peuvent exister entre des individus en interaction. Ces conflits sont souvent présentés dans notre vie réelle, sociale, économique ou tout autre domaine où les individus interagissent entre eux.

Le but principal pour chaque individu consiste à savoir comment réagir et quelle sera la décision à prendre pour satisfaire son intérêt personnel selon le principe de la rationalité qui vise à maximiser son utilité (gain, profit) ou minimiser son coût. Pour répondre à ces besoins, plusieurs études ont été faites pour pouvoir analyser et, dans certains cas, résoudre ces conflits. Cette étude de conflits d'intérêts est appelée Théorie des jeux.

5.2.2 Définitions importantes :

Un jeu : est une situation où les individus (les joueurs) sont conduits à faire des choix parmi un certain nombre d'actions possibles, et dans un cadre défini à l'avance (les règles de jeu), les résultats de ces choix constituant une issue du jeu, à laquelle est associé un gain, positif ou négatif, pour chacun des participants.

Un joueur : est un acteur ou une entité pouvant être une personne, une entreprise, un gouvernement, une cellule, un virus ...etc., agissant dans leur propre intérêt selon le principe de la rationalité.

On notera N l'ensemble des joueurs participant à un même jeu : $N = \{1, \dots, n\}$ où N désigne leur nombre ($N \geq 2$).

Conflit : Un conflit est une situation où les questions suivantes ont un sens

1. Qui participe à cette situation ?
2. Quel sont les résultats possibles de cette situation ?
3. Qui est intéressé par ces résultats ? Comment ?

Stratégie : Une stratégie d'un joueur est un choix parmi la liste de décision qu'il envisage de prendre. Le résultat des choix de tous les joueurs constitue une (issue) ou profil d'action du jeu.

Interaction : Toute action choisie par un joueur aura une influence sur celle des autres joueurs.

Gains(profit) : Le gain d'un joueur est le bénéfice négatif (perte) ou positif qui résulte des choix de tous les joueurs.

Coalition : Une coalition est une partie de l'ensemble des participants (joueurs), qui s'organise d'une certaine façon.

Théorie des jeux : est permet une analyse formelle des problèmes posé par l'interaction stratégique d'un groupe d'agents rationnels pour suivant des buts qui leur sont propre :

- Groupe
- Interaction
- Stratégique
- Rationnel

5.2.3 Classification des jeux

Les jeux peuvent être classés selon le comportement des joueurs, selon l'information que possèdent les joueurs et selon le gain :

► **Jeux Coopératifs :**

On dit que le jeu est coopératif, si les joueurs peuvent se grouper dans des coalitions,

Où le choix de leurs stratégies est d'décide en commun, afin d'améliorer les gains de tous les joueurs coalisés. Notons que les coalitions sont formées ou d'définies au début du jeu, ainsi donc on ne parlera pas de coalitions se formant durant le jeu, ou devenant interdites. Les jeux coopératifs se divisent en deux catégories : Les jeux sans paiements latéraux et les jeux avec paiements latéraux.

- **Exemple :** Le premier joueur murmure à l'oreille de son voisin un message, qu'il doit à son tour dire à son voisin, et ainsi de suite .la dernière personne doit dire à haute voix ce qu'elle a compris. Cela sera souvent très différent du message initial.

► **Jeux Non Coopératifs :**

On appelle jeu non coopératif, un jeu où les joueurs ne peuvent pas former de coalitions,

Ils ne communiquent pas ou ne peuvent pas communiquer entre eux agissent selon le principe de la rationalité économique : chacun cherche à apprendre les meilleures décisions pour lui-même (c'est-à-dire cherche à maximiser égoïstement ses gains individuels). Ce dernier type de jeu fait intervenir les probabilités.

► **Jeu à information complète / incomplète :**

Un jeu est dit à information complète si chacun des joueurs connaît la structure du jeu, c'est à dire : l'ensemble des joueurs, les préférences des joueurs, les règles du jeu et le type d'information qu'à chaque moment du jeu chaque joueur possède sur les actions entreprises par les autres joueurs au cours des phases précédentes. Donc chaque joueur peut se mettre à la place de tous les autres joueurs et du modélisateur. Si au moins un des joueurs ne connaît pas entièrement la structure du jeu, le jeu est dit à information incomplète.

► **Jeu à information parfaite / imparfaite :**

Un jeu est dit à information parfaite si chacun des joueurs, au moment de choisir son action, a une connaissance parfaite de l'ensemble des décisions prises antérieurement par les autres joueurs. Un jeu est à information imparfaite si un des joueurs ne connaît pas, à un moment du déroulement du jeu, ceux qu'a joué un autre joueur. Ceci peut arriver dans le cas où on cache l'information aux joueurs ou parce que les joueurs jouent simultanément.

► **Jeu à somme nulle /non nulle :**

On dit qu'un jeu est à somme nulle si le montant total des gains à la fin de la partie est à somme nulle, en d'autres termes si le montant total gagné par un joueur est égal au montant perdu par l'autre, par exemple les échecs ou le Poker sont des jeux à somme nulle.

5.2.4 Les formes des jeux

Tous ces jeux peuvent être définis de deux manières différentes qui sont :

1. Forme stratégique (normale).
2. Forme extensive (arborescente).

Jeu sous forme stratégique :

La forme normale, également connue sous le nom de forme stratégique, est la représentation la plus familière des interactions stratégiques dans la théorie des jeux. Un jeu écrit de cette façon s'élève à une représentation de l'utilité de chaque joueur, l'ensemble de ses stratégies et les issues possibles du jeu.

Définition:

Un jeu sous forme normale peut être représenté sous la forme :

$$(N, \{S_i\}_{i \in N}, \{f_i\}_{i \in N})$$

Où :

* $N = \{1, \dots, n\}$ est l'ensemble des joueurs, un joueur quelconque est appelé i et donc $i \in \{1..n\}$.

* $S_i \subseteq R^{n_i}$ Désigne l'ensemble des stratégies du joueur $i \in N$.

* $S = \prod_{i=1}^n S_i$ est l'ensemble des issues du jeu.

* f_i est la fonction d'utilité du joueur.

Remarque : Un jeu sous forme normale (stratégique) est un jeu ayant les caractéristiques suivantes :

- Il s'agit d'un jeu à information complète.
- Il s'agit d'un jeu simultané (statique) où chaque joueur choisit une stratégie indépendamment du choix de l'autre joueur et le jeu ne se répète pas.
- Les joueurs sont rationnels et leur objectif est la maximisation de leurs paiements.

Exemple:

Considérons le jeu suivant appelé pile ou face. Dans ce jeu, deux enfants placent secrètement une pièce d'un dinar dans la crue de leurs mains. Puis simultanément chaque enfant ouvre sa main et montre à l'autre sa pièce. Si les deux pièces sont toutes les deux de côté pile ou de côté face, l'enfant 1 donne un dinar à l'enfant 2. Dans le cas contraire c'est l'enfant 2 qui donne un dinar à l'enfant 1. C'est un jeu à information imparfaite car les deux enfants ouvrent les mains simultanément sans savoir comment l'un et l'autre avaient placé la pièce dans leur main.

La forme normale de ce jeu apparaît dans la figure suivante :

Enfant 2 \ Enfant 1	Pile	Face
Pile	(-1,1)	(1,-1)
Face	(1,-1)	(-1,1)

Dans les quatre cases du tableau 1 :

- Le premier chiffre est le paiement que recoite l'enfant 1.
- Le second chiffre est le paiement que recoite l'enfant 2.

Jeux finis à N joueurs

Définition :

Le jeu (1.1) est dit fini, si chacun des joueurs a un ensemble fini de stratégies, càd :

$$(N, \{S_i\}_{i \in N}, \{f_i\}_{i \in N})$$

Cardinal de $S_i = |S_i| < +\infty; \forall i \in N$.

Concepts de solution :

Maintenant que nous avons défini un jeu sous forme normale et quelles sont les stratégies disponibles aux joueurs, la question est comment raisonner au sujet de tels jeux. Les théoriciens de jeu traitent ce problème en identifiant certains sous-ensembles de résultats, appelés les concepts de solution. Dans cette section, nous décrivons quelques concepts de solution et le concept le plus fondamental à savoir : *équilibre de Nash*.

Définition 1 :

Considérons le jeu sous forme normale (1.1). On dit que la stratégie $s_i \in S_i$ est une stratégie dominante, si :

$$\forall s'_i \in S_i, \forall s_{-i} \in S_{-i}, f_i(s_i, s_{-i}) \geq f_i(s'_i, s_{-i})$$

La stratégie $s_i \in S_i$ est dominante pour le joueur i, si s_i domine S'_i .

Définition 2 :

Une stratégie $s_i \in S_i$ est une stratégie strictement dominante pour le joueur i dans le jeu (1.1), si :

$$\forall s'_i \in S_i, s'_i \neq s_i, \forall s_{-i} \in S_{-i}, f_i(s_i, s_{-i}) > f_i(s'_i, s_{-i})$$

Définition 3 :

Une stratégie s_i est faiblement dominante, si :

$$\begin{aligned} \forall s'_i \in S_i, s'_i \neq s_i, \forall s_{-i} \in S_{-i}, f_i(s_i, s_{-i}) &\geq f_i(s'_i, s_{-i}) \\ \exists s_{-i} \in S_{-i}, f_i(s_i, s_{-i}) &\succ f_i(s'_i, s_{-i}) \end{aligned}$$

Equilibre en Stratégies dominantes : Une situation $s = (s_1^*, s_2^*, \dots, s_N^*) \in S$ est appelée équilibre en stratégies dominantes dans le jeu (1.1), si chaque composante $s_i^* \in S_i$ est une stratégie dominante pour le joueur $i, \forall i \in N$.

Equilibre de Nash :

Le concept clé de la théorie des jeux est le concept d'équilibre de Nash introduit par le mathématicien John Nash en 1950. Un équilibre de Nash correspond donc à une situation où aucun joueur n'a intérêt à dévier unilatéralement de la situation d'équilibre.

Définition :

Un profil de stratégies $s^* = (s_1^*, s_2^*, \dots, s_N^*)$ est un équilibre de Nash pour le jeu (1.1) si aucun des joueurs n'a intérêt à dévier unilatéralement de sa stratégie s_i^* (quand les autres joueurs continuent à jouer le profil s_i^* Autrement dit :

$$\forall i \in N, \forall s_i \in S_i, f_i(s_1^*, s_2^*, \dots, s_n^*) \geq f_i(s_i, s_{-i}^*)$$

Exemple : « Le dilemme du prisonnier » :

Deux cambrioleurs, notés 1 et 2, sont arrêtés par la police et placés en garde à vue dans des cellules différentes. La police n'a pas assez de preuves, donc elle leur propose un marché :

Si les deux ne coopèrent en ne se dénonçant pas mutuellement, chacun va écoper d'un an de prison. Si l'un trahit en accusant l'autre, alors celui qui a trahi va être libéré, et celui qui s'est tu va écoper 10 ans de prison. Si les deux se dénoncent mutuellement, alors chacun va écoper de 5 ans de prisons.

Suspect1 \ Suspect2	Se taire (ST)	Dé noncer(D)
	Se taire (ST)	Dé noncer(D)
Se taire (ST)	(-1,-1)	(-10,0)
Dé noncer(D)	(0,-10)	(-5,-5)

Pour le suspect 1 :

$$f_1 (ST, ST) = 1 < f_1 (D, ST) = 0$$

Pour le suspect 2 :

$$f_2 (ST, ST) = -1 < f_2 (ST, D) = 0$$

$\Rightarrow (ST, ST)$ n'est pas un équilibre de Nash. Par contre le profil (D, D) est un équilibre de Nash.

En effet :

$$f_1 (D, D) = -5 > f_1 (ST, D) = -10$$

$$f_2 (D, D) = -5 > f_2 (D, ST) = -10$$

Équilibre de Nash et Meilleures réponses:

Définition: Pour chaque joueur i et profil d'actions de ses adversaires s_{-i} , on dit que s_i est meilleure réponse contre s_{-i} si :

$$\forall s'_i \in S_i, f_i(s_i, s_{-i}) \geq f_i(s'_i, s_{-i})$$

On appelle correspondance de meilleures réponses du joueur i l'application multivoque

MR_i définie de dans S_{-i} l'ensemble des parties 2^{S_i} de S_i qui à chaque s_{-i} associe l'ensemble des meilleures réponses du joueur i :

$$\begin{aligned} MR_i : s_{-i} &\longrightarrow 2^{S_i} \\ s_{-i} &\longrightarrow MR_i(s_{-i}) = \{s_i \in S_i, f_i(s_i, s_{-i}) \geq f_i(s'_i, s_{-i}), \forall s'_i \in S_i\} \\ &= \{s_i \in S_i, f_i(s_i, s_{-i}) = \max_{s'_i} f_i(s'_i, s_{-i})\} \\ &= \arg \max_{s'_i} f_i(s'_i, s_{-i}) \end{aligned}$$

Définition :

On appelle correspondance de meilleures réponses du jeu (1.1) l'application Multivoque :

$$\begin{aligned} MR_i s &\longrightarrow 2^s \\ s &\longrightarrow MR(s) = \prod_{i=1}^n MR_i(s_{-i}) \end{aligned}$$

Proposition:

$s^* \in S$ est un équilibre de Nash pour le jeu (1,1) si et seulement si s^* est un point fixe de la correspondance de meilleure réponses MR de ce jeu, i.e: $s^* \in MR(s^*)$

Théorème:

Un jeu sous forme normale (1.1) admet un équilibre de Nash, si les conditions suivantes sont vérifiées

* Les ensembles S_i , des stratégies sont convexes et compact $\forall i \in 1..N$

* Les fonctions $f_i: \prod_{i=1}^n s_{-i} \rightarrow R$, sont continués $\forall i \in 1..N$

* Les fonctions $f_i (s_i, s_{-i}) s_i \rightarrow R$, sont concaves, $\forall i \in 1..N \forall s_i \in N : \prod_{j \in N, j \neq i}^n s_j$

Jeu sous forme extensive

Comme on l'a vu, la formule normale d'un jeu n'est très pratique pour représenter des jeux longs, car elle donne lieu à une colonne /ligne par stratégie possible de chaque joueur. Pour éviter des représentations peu pratiques, on représente en général les jeux impliquant plus d'un mouvement de l'un ou autre des agents sous une autre forme, la forme extensive. cette manière de représenter un jeu consiste à créer un arbre, qui montre les possibles actions de chaque joueur.

Définition :

Un jeu en forme extensive est donné par :

Un arbre de jeu contenant un nœud initial, des nœuds de décisions, des nœuds terminaux et des branches reliant chaque nœud à ceux qui lui succèdent.

Un ensemble de $N \geq 1$ joueurs, indicés par $i=1 \dots N$.

Pour chaque nœud de décision, le nom du joueur qui a le droit de choisir une stratégie à ce nœud.

Pour chaque joueur i la spécification de l'ensemble des stratégies permises à chaque nœud où il a le droit de prendre une décision.

La spécification des gains de chaque jeu à chaque nœud terminal.

Exemple : (le pilote et le terroriste)

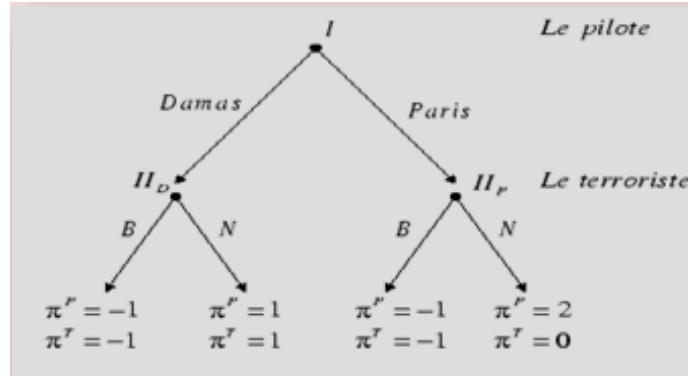
Un terroriste monte sur l'avion Strasbourg-Paris.

Après 10mn de vol, quand le terroriste s'approche du pilot et le menace de faire exploser l'avion s'il ne n'atterrit pas à Damas.

Le pilote a le choix entre continuer le vol vers Paris ou d'adopter la direction de Damas : $A^P \{ P, D \}$.

Après avoir observé le choix du pilote, le terroriste a le choix entre faire exploser la bombe ou abandonner : $a^T \in \{ B, N \}$.

L'arbre suivant représente ce jeu :



Le premier nœud de cet arbre représente la décision du premier joueur.

A la suite de chacun de ses choix, la possibilité est donnée au joueur suivant d'effectuer son choix.

Une fois que tous les joueurs ont pris leurs décisions, on arrive à un résultat du jeu auquel les gains correspondants sont associés.

Ainsi l'arbre permet-il de faire apparaître les gains associés aux différents résultats possibles.

▲ On peut représenter cet exemple en forme normale :

La définition des stratégies que nous venons d'introduire nous permet de représenter aisément ce jeu sous une forme normale

		T			
		(B, B)	(B, N)	(N, B)	(N, N)
P	D	$(-1, -1)$	$(-1, -1)$	$(1, 1)$	$(1, 1)$
	P	$(-1, -1)$	$(2, 0)$	$(-1, -1)$	$(2, 0)$

Nous pouvons alors examiner les équilibres de ce jeu :

- Le pilote n'a pas de stratégie dominante.
- Le terroriste a une stratégie faiblement dominante : (N, N)
- La meilleure réponse du pilote face à cette stratégie est P.

- Une issue possible du jeu est donc $(P, (N, N))$.
- C'est aussi un équilibre de Nash.
- Malheureusement ce n'est pas le seul.
- $(P, (B, N))$ et $(D, (N, B))$ sont aussi des équilibres de Nash.
- Nous ne pouvons donc pas prédire précisément l'issue du jeu. Nous devons trouver un moyen de discriminer parmi ces équilibres de Nash et d'affiner nos prédictions, en nous concentrant sur l'issue la plus probable.

5.2.5 Quelle qu'exemples sur la théorie des jeux

Exemple 1 :

Quels sont les équilibres de Nash du jeu du rendez-vous à Champs-Sur-Marne ?

Il y a deux équilibres de Nash, $(E; E)$ et $(C; C)$.

Le jeu du dilemme du prisonnier est résoluble par élimination itérée des stratégies Strictement dominées, et l'équilibre de Nash est compatible avec cette prédiction.

	E	C
E	(1,1)	(0,0)
C	(0,0)	(1,1)

Exemple 2 :

Soit le jeu : Quatre équilibres de Nash : $(H ; G)$, $(M ; G)$, $(M ; D)$, $(B ; D)$. Notons que ce jeu admet des équilibres de Nash, alors qu'il n'est pas résoluble par élimination itérée des stratégies strictement dominées.

	G	D
H	(1;1)	(0;0)
M	(1;1)	(2;1)
B	(0;0)	(2;1)

Exemple 3:

Le jeu suivant est-il résoluble par élimination itérée des stratégies strictement dominées ? Quels sont ses équilibres de Nash ?

Ce jeu n'est pas résoluble par élimination itérée des stratégies strictement dominées. L'unique équilibre de Nash est (m; M).

	G	M	D
h	(5,3)	(0,4)	(3,5)
m	(4,0)	(5,5)	(4,0)
d	(3,5)	(0,4)	(5,3)

5.3 La programmation dynamique

La programmation dynamique (PD) est un d'outil de base de la recherche opérationnelle il s'agit d'une technique permettant de résoudre un vaste ensemble de problèmes.

Le terme (programmation) n'est de plus pas vraiment approprié. Il a été proposé par Richard Bellman dans des années 50 dans la foulée du succès de la programmation linéaire. Bellman est le premier à avoir décrit en détail et popularisé la P. D. elle est très utilisée, notamment en économie, dans l'industrie et dans la sante par exemple pour l'optimisation de procédés chimique ou la gestion des stocks.

5.3.1 Définition

La programmation dynamique est une méthode algorithmique pour résoudre des problèmes d'optimisation, elle consiste à résoudre un problème en le décomposant en sous-problème puis à résoudre les sous problèmes plus petits aux plus grands en stockant les résultats intermédiaires. Elle s'appuie sur le principe d'optimalité de Bellman

5.3.2 Problèmes d'optimisation et complexité

*Programmation dynamique utilisée pour les Problèmes d'optimisation dans lesquels le principe d'optimalité de Bellman est vérifié :

« Une solution optimale pour le problème contient les solutions optimales pour tous les sous problèmes »

*Sous structure optimale :

Une solution optimale peut être obtenue à partir des solutions optimales de certains sous-problèmes.

- Il existe un ordre sur les sous problèmes compatible avec la sous structure optimale.

5.3.3 Équation de Bellman

Elle régit un « système dynamique » sur $t = 0, 1, \dots, T$

-Temps discret, horizon fini.

♦ x_t : état du système à l'instant t

♦ $a_t \in \tau(x_t)$: action parmi celles possibles lorsque système en x_t

Il passe alors en $x_{t+1} = \phi(x_t, a_t)$

♦ $F(x_t, a_t)$: « cout » à choisir l'action a_t si système en x_t

♦ $P_t(x_{t+1}) = \min_{a_t} \{ \sum_{t=0}^T F(x_t, a_t) \}$

Cout global après T étapes lorsque l'on arrive en x_{T+1} .

Alors :

$$P_T(x_{T+1}) = \min_{\{a_t \in \tau(x_t) \mid x_{T+1} = \phi(x_t, a_t)\}} \left\{ \sum_{t=0}^T F(x_t, a_t) + P_{T-1}(x_T) \right\}$$

5.3.4 Principe d'optimalité de Bellman

La programmation dynamique est fondée sur le principe d'optimalité de Bellman :

Soit f une fonction réelle de x et $y = (y_1, y_2, \dots, y_n)$.

Si f est décomposable, sous la forme :

$$f(x; y) = f_1(x; f_2(y))$$

Alors, on a :

$$Opt_{x,y} f(x, y) = Opt_x \{ f_1(x, Opt_y \{ f_2(y) \}) \}$$

Où Opt représente Min ou Max .

5.3.5 Equation de la programmation dynamique

Pour une séquence de décisions $u_0, u_1, \dots, u_T$

Une séquence d'états $x_0, x_1, \dots, x_T$,

Une fonction de transition

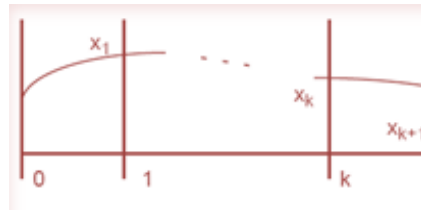
$$x_{k+1} = \theta_k(x_k, u_k)$$

Et une fonction objectif séparable.

On définit pour chaque étape une fonction de cout optimal à venir (cas additif) :

$$F_k(x_k) = Opt_{u_k} \{f_k(x_k, u_k) + F_{k+1}(\theta_k(x_k, u_k))\}$$

On peut alors procéder récursivement à partir de la période terminale T (Procédure en arrière) :



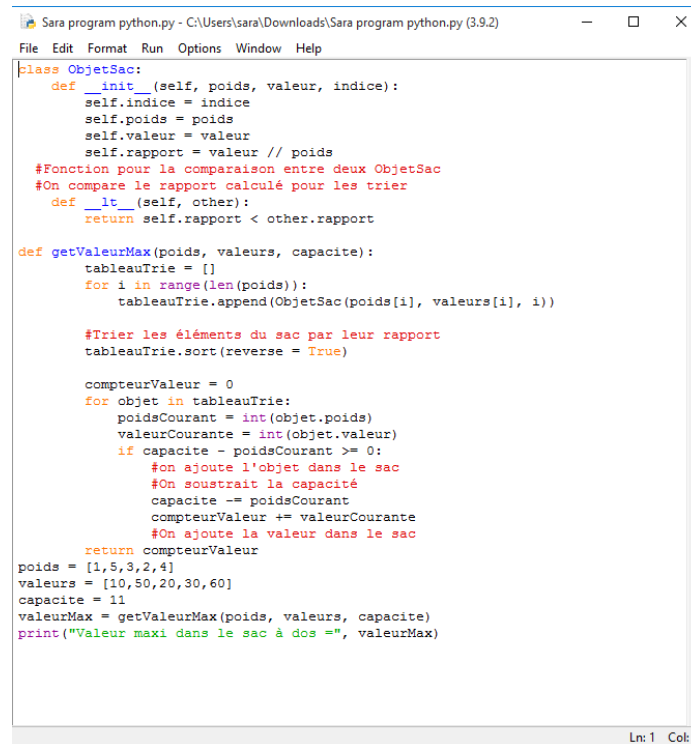
5.4 Pratique en logicielle Python

5.4.1 Exemple

L'énoncé de ce problème est simple : « Étant donné plusieurs objets possédants chacun un poids et une valeur et étant donné un poids maximum pour le sac, quels objets faut-il mettre dans le sac de manière à maximiser la valeur totale sans dépasser le poids maximal autorisé pour le sac ? ».

Afin d'illustrer ce type de problèmes que nous rencontrons souvent dans la vie courante, nous vous proposons de jouer au jeu ci-dessous. Le but est de remplir un sac à dos avec quatre types d'objets. Le nombre d'objets de chaque type mis à disposition dépend de la difficulté choisie. En mode deux joueurs, le gagnant sera celui qui aura rempli le sac sans dépasser la limite de poids maximum avec la plus grande valeur totale et le plus rapidement possible.

Le programme dans python comme suit:



```
class ObjetSac:
    def __init__(self, poids, valeur, indice):
        self.indice = indice
        self.poids = poids
        self.valeur = valeur
        self.rapport = valeur // poids
    #Fonction pour la comparaison entre deux ObjetSac
    #On compare le rapport calculé pour les trier
    def __lt__(self, other):
        return self.rapport < other.rapport

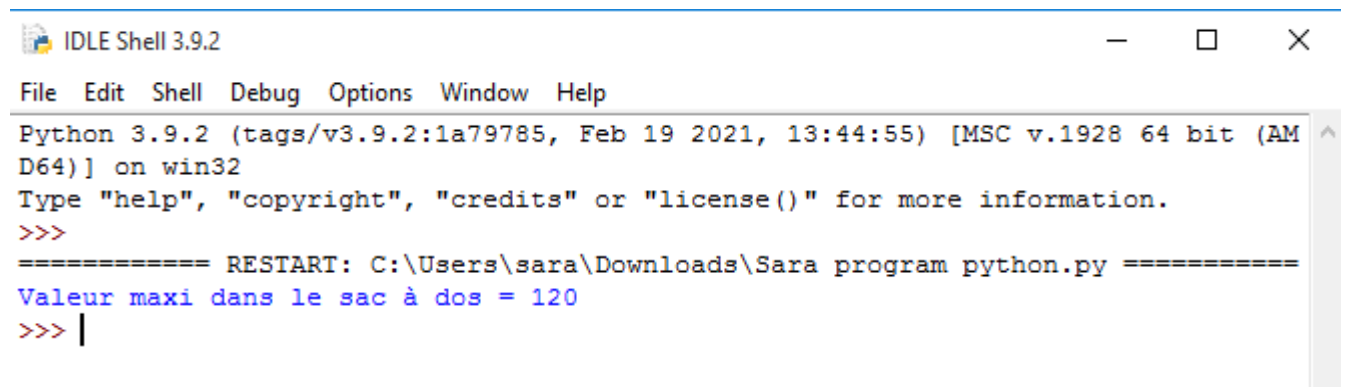
def getValeurMax(poids, valeurs, capacite):
    tableauTrie = []
    for i in range(len(poids)):
        tableauTrie.append(ObjetSac(poids[i], valeurs[i], i))

    #Trier les éléments du sac par leur rapport
    tableauTrie.sort(reverse = True)

    compteurValeur = 0
    for objet in tableauTrie:
        poidsCourant = int(objet.poids)
        valeurCourante = int(objet.valeur)
        if capacite - poidsCourant >= 0:
            #on ajoute l'objet dans le sac
            #On soustrait la capacité
            capacite -= poidsCourant
            compteurValeur += valeurCourante
            #On ajoute la valeur dans le sac
        return compteurValeur

poids = [1,5,3,2,4]
valeurs = [10,50,20,30,60]
capacite = 11
valeurMax = getValeurMax(poids, valeurs, capacite)
print("Valeur maxi dans le sac à dos =", valeurMax)
```

Résultat



```
IDLE Shell 3.9.2
File Edit Shell Debug Options Window Help
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\sara\Downloads\Sara program python.py =====
Valeur maxi dans le sac à dos = 120
>>> |
```

5.5 Conclusion

La recherche opérationnelle vise à résoudre certains problèmes par des méthodes scientifiques à l'aide des ordinateurs. L'optimisation est une activité essentielle dans toutes les disciplines importantes.

La théorie de jeux a pour objet de formaliser des situations conflictuelles inhérentes à une communauté d'individus en interaction de discuter puis de proposer des solutions à ces problèmes

La programmation dynamique est séduisante car sont formalismes et assez génériques, ce qui laisse libre court à deux multiples variantes et à la résolution de problèmes divers, qu'ils soient déterministes ou non.

Bibliographie

- [1] A. Gourdin, M.Boumahrat, Méthodes Numériques Appliquées
- [2] ANTOINE Joseph, CORNIL jean-paul, Lexique thématique de la comptabilité, dictionnaire spécialisé explicatif édition revue, augmentée et mise à jour avec la collaboration de Stéphane mercier, Belgique 2002
- [3] A. Alotaibi, P. L. Combettes, and N. Shahzad. Solving coupled composite monotone inclusions by successive Fejér approximations of their KuhnTucker set. SIAM J. Optim., 24(4) :2076–2095, Dec. 2014.
- [4] BAILLARGEON Gérard, programmation linéaire appliqué, outils d’optimisation et d’aide à la décision, copyright © 1996, Édition SMG.
- [5] BAIR Jacques, algèbre linéaire, pour l’économie et les sciences sociales, deuxième édition, édition De Boeck, Bruxelles 1994.
- [6] BARRE.R, LORY.R, RICHEZ.M, comptabilité analytique d’exploitation, édition IS-TRA, Paris 1980.
- [7] BASTIN Fabian, modèle de recherche opérationnelle, édition viaduc de Millau,France 2006.
- [8] BENGHEZAL Amour Farouk, programmation linéaire, revues et augmentée,édition de l’office des publications universitaire, 2eme édition, Alger 2006 .
- [9] BOUQUIN Henri, comptabilité de gestion, édition economica, Paris, 2000.

- [10] Boussard Jean-marc, programmation mathématique et théorie de la production agricole, éditions cujus, paris 1970.
- [11] COLLETTE.Yann, SIARRY.Patrick, multiobjective optimization, principles and case studies series editor, publication due august, France 2003.
- [12] Dantzig G.B , linear programming and extension press,princeton , Jersey,1963.
- [13] Gass S.I Linear programming ,MacGr-Hill , New York,1958.
- [14] In G. Nash, éditeur, A History of Scientific Computing, ACM Press Hist. Ser.Origins of the simplex method. pages 141–151. ACM Press, Reading, MA, États-Unis.
- [15] John Wiley & Sons Inc.Linear programming. New York(1983)
- [16] M.V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. 20(3) :681–695, Mar. 2011.
- [17] M. Minoux. Programmation mathématique, théorie et algorithmes,tome 1. Dunod, Paris, 1983.
- [18] Mokhtar S.Bazaraa, J.arvis, Linear programming and network Flows.
- [19] Springer-Verlag Linear Optimisation and Extensions, deuxième édition, (1999).
- [20] S.ACHMANOV, Programmation linéaire.
- [21] Zoubir ramdani Université Mohamed Elbachir Elibrahimi de B.B.A Faculté des mathématiques et d’informatique Département de Recherche Opérationnell, cours.
- [22] <https://dl.ummo.dz/bitstream/handle/ummo/2537/Acharaiou%2C%20Tinhinane.pdf?sequence=1>
- [23] <https://laviedesidees.fr/A-quoi-sert-la-theorie-des-jeux.html#:~:text=La%20th%C3%A9orie%20des%20jeux>
- [24] <http://www.cril.univartois.fr/~konieczny/enseignement/TheorieDesJeux.pdf>
- [25] http://www.lsis.org/master/ancien_site/document.

- [26] <https://www.lri.fr/~jcohen/documents/enseignement/Cours-prog-dynamique.pdf>.
- [27] <https://www.techno-science.net/definition/5373.html>