REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université MOULOUD MAMMERI de Tizi-Ouzou

FACULTE DES SCIENCES

DEPARTEMMENT DES MATHEMATIQUES

Mémoire

En vue de l'obtention du Diplôme de Master

Spécialité : Recherche Opérationnelle

Thème

Optimisation d'un problème de programmation linéaire en nombres entiers par la méthode Branch and Bound

Présenté par :

- ZIDANE THANINA
- HAMDAD KAHINA

Encadrées par :

- Mme Abderhmani Amel

Promotion

2016-2017

Résumé :

Quelques problèmes de la vie courante peuvent être modélisés comme des problèmes combinatoires. Dans notre travail, nous nous intéressons à la résolution des problèmes de programmation linéaire en nombres entiers par la méthode : " Séparation et Évaluation", "Branch and Bound".

Nous présentons les méthodes de résolutions de problèmes d'optimisation proposées dans la littérature. Notre travail est devisé en trois chapitres :
Le premier chapitre présente quelques définitions dont : la programmation mathématique, la programmation linéaire et quelques généralités sur la classification des problèmes et les méthodes de résolutions de problèmes d'optimisation.

Dans le deuxième chapitre, on se focalise sur les méthodes de résolutions exactes. Nous avons illustré avec un exemple chacune de ces méthodes : la méthode graphique, la méthode du simplexe et la méthode dual du simplexe. On présente aussi les deux algorithmes : L'algorithme du simplexe et l'algorithme dual du simplexe.

Le troisième chapitre est consacré à la méthode de Séparation et Évaluation "Branch and Bound". Nous avons présenté la méthode, son principe, ces trois procédures essentielles et les trois stratégies de parcours, ainsi que l'algorithme général de la méthode et un exemple d'application avec une implémentation informatique en utilisant le logiciel MATLAB version 2014 et le LpSolve IDE.

Mots clés :
Programmation linéaire, programmation mathématique, simplexe, dual du simplexe, la méthode Branch and Bound, programmation en nombres entiers .

Table des matières

Remerciements

Nous remercions tout d'abord $\ll DIEU \gg$ tout puissant de nous avoir donné suffisamment de force et surtout de courage et de persévérance pour mener notre travail à terme.
Nous tenons a adresser toutes les reconnaissances et notre gratitudes a notre promotrice : "Abderehmani Amel" , ses précieux conseils et son suivi.
Que les membres du jury trouvent ici nos remerciements les plus vifs pour avoir aimablement accepté de juger notre travail.
Sans oublier d'exprimer notre reconnaissance et gratitude à tous nos enseignants durant notre cursus au département Mathématiques.
Enfin, nous tenons a remercier tous ceux qui ont contribué de près ou de loin a la réalisation de ce travail.

Kahina et Thanina

Dédicaces

Je tiens très respectueusement à dédier ce modeste travail à :
Mes très chers parents qui ont toujours été là pour moi, et qui m'ont donné un honorable modèle de labeur et de persévérance,qu'ils trouvent ici l'expression de toute ma profonde reconnaissance et ma parfaite considération.
Mes frères et sœurs .
Mes beaux frères et belles sœurs .
Mes chers nièces et neveux.
toute ma famille, du plus vieux jusqu'au plus jeune.
Ma binôme et à toute sa famille .
Je désire le dédier à tous mes amis et camarades .

Kahina

Je dédie ce travail à :
La mémoire de mon très cher KHALI BEZZAOU SADDEK, que dieu l'accueille dans son éternel paradis.
Ma très chère mère, que dieu lui accord une longue vie.
Mon petit cher frère YOUVA à qui je souhaite la réussite au baccalauréat.Que dieu le protège .
Ma petite sœur THINHINANE, que j'adore énormément.
Tous mes amis et camarades, que j'ai connu tout au long de mes études (lycée BENI-DOUALA ; la fac et la résidence universitaire M.M.T.O).
Ma chère binôme et toute sa famille .

Thanina

Introduction générale

La Recherche Opérationnelle est la discipline utilisée pour élaborer de meilleures décisions. Elle permet d'optimiser le fonctionnement des systèmes de production ou d'organisation ou autres.
Une des parties essentielles de la recherche opérationnelle est la programmation linéaire, qui est introduite par les russes Kontorovitch en (1939) et résolue par l'Américain G.B .Dantzig en (1947).[3]
La programmation linéaire est l'un des principaux outils de modélisation en recherche opérationnelle. C'est aussi la source des méthodes avancées de la programmation mathématique, qui permettent de résoudre les programmes quadratiques, entiers, stochastiques . . . etc. On distingue dans la programmation linéaire, la programmation linéaire continue (PLC) et la programmation linéaire en nombres entiers (PLNE), comme il est possible d'avoir les deux en même temps, ce qu'on appelle la programmation linéaire en variables mixtes (PLM).

Dans notre travail, nous nous intéressons à la résolution des problèmes de programmation linéaire en nombres entiers par la méthode :
" Séparation et Évaluation " , "Branch and Bound" .
Nous présentons les méthodes de résolutions de problèmes d'optimisation proposées dans la littérature. Notre travail est devisé en trois chapitres :

Le premier chapitre présente quelques définitions dont : la programmation mathématique,la programmation linéaire et quelques généralités sur la classification des problèmes et les méthodes de résolutions de problèmes d'optimisation .

Dans le deuxième chapitre, on se focalise sur les méthodes de résolutions exactes.
Nous avons illustré avec un exemple chacune de ces méthodes : la méthode graphique, la méthode du simplexe et la méthode dual du simplexe.On présente aussi les deux algorithmes : L'algorithme du simplexe et l'algorithme dual du simplexe.

Le troisième chapitre est consacré à la la méthode de Séparation et Évaluation "Branch and Bound" .
Nous avons présenté la méthode ,son principe ,ces trois procédures essentielles et les trois stratégies de parcours ,ainsi que l'algorithme général de la méthode et un exemple d'application avec une implémentation informatique en utilisant le logiciel Matlab version 2014 et le LpSolve IDE.

Chapitre 1

Définitions et Généralités

1.1 Quelques définitions

1.1.1 La programmation mathématique

La programmation mathématique est un ensemble de méthodes ou processus mathématiques dont le but est de trouver un optimum pour un problème décisionnel donné. C'est à dire il s'agit de rechercher l'optimum d'une fonction de n variables et de m contraintes .

La programmation mathématique englobe des outils comme :

– La programmation linéaire :
  Un problème linéaire (PL) est un problème d'optimisation maximisant ou minimisant une fonction objectif linéaire à n variables de décision supposées non négatives soumises à un ensemble de contraintes exprimées sous forme d'équations ou inéquations linéaires.

– La programmation linéaire en nombres entiers :
  C'est une extension de la méthode précédente qui permet de travailler aussi sur des petites quantités entières (nombre de bus, nombre d' avions) et décisions binaires (telle action est effectuée ou non).

– La programmation non linéaire :
  Elle est utilisée lorsque les relations entre les décisions ne peuvent pas être exprimées de façon linéaire, même avec des hypothèses simplificatrices. Elle est plus générale que la programmation linéaire mais a le défaut de ne pas garantir l'obtention de meilleures décisions.

– La programmation dynamique :
  Adaptée aux problèmes décisionnels possédant une propriété de sous-optimalité, c'est-à-dire que des décisions bonnes pour le problème global sont aussi bonnes pour des sous-problèmes.

1.1.2 La programmation linéaire

Le problème le plus simple de la programmation mathématique est celui de programmation linéaire.
La programmation linéaire peut être définie comme un outil mathématique qui permet d'analyser divers types de situations dans lesquelles nous retrouvons une fonction linéaire d'un certain nombre de variables, appelée Fonction Objectif que l'on désire optimiser (maximiser ou minimiser).
Ces variables appelées variables de décision (dont on veut déterminer les valeurs optimales) sont soumises à des contraintes qui sont linéaires.[3]

1.1.3 Forme générale d'un programme linéaire

Lors de la modélisation d'un problème de programmation linéaire on est amené à déterminer trois éléments importants :

1. **Les variables de décision :**
   C'est la première étape dans le processus de modélisation qui consiste à identifier correctement toutes les variables de décision (inconnues) de la situation à modéliser.

2. **Les contraintes linéaires :**
   Dans la problématique de la situation, il faut être en mesure d'identifier tout genre de restriction qui peut limiter les valeurs que peuvent prendre les variables de décision.

3. **La fonction objectif :**
   On associe à chaque variable de décision du modèle correspond ,un coefficient économique indiquant la contribution unitaire de la variable correspondante à l'objectif poursuivi. Par la suite, on pourra en déduire la fonction objectif que l'on veut optimiser (soit à maximiser soit à minimiser).

La forme la plus générale d'un problème de (P.L) est :

$$
(PL)\begin{cases}
\textit{opt } \mathrm{Z}= \quad max(minZ) = \sum_{j=1}^{n} c_j x_j & (1,1)\\
\sum_{j=1}^{n} a_{ij}x_j < b_i \quad i \in I \subset \{1,...,m\} & (1,2)\\
\sum_{j=1}^{n} a_{lj}x_j = b_l \quad l \in L \subset \{1,...,m\} & (1,2)\\
\sum_{j=1}^{n} a_{kj}x_j > b_k \quad k \in K \subset \{1,...,m\} & (1,2)\\
x_j \in R^+, \quad j \in \{1,..,n\} & (1,3)\\
I \cap L \cap K = \emptyset & \\
I \cup L \cup K = \{1,...,m\} &
\end{cases}
$$

avec c_j,a_{ij},b_i des constantes et $x_j$variables.
(1,1)fonction économique ou fonction objectif.
(1,2)contrainte réelle .

(1,3)contrainte de non-négativité.

Remarque :
• Un programme linéaire est dit sous forme canonique si toutes les inégalités sont dans le même sens, les variables de décision sont positives ou nulles et les contraintes d'égalité en sont absentes.

$$(PL)\begin{cases} opt \text{ z=} & max(minZ) = \sum\limits_{j=1}^{n} c_j x_j \\ & \sum\limits_{j=1}^{n} a_{ij}x_j \geq b_i \qquad i\text{= }1,\ldots,m \\ & x_j \geq 0 \qquad j\text{= }1,\ldots,n \end{cases}$$

ou sous la forme :

$$(PL)\begin{cases} opt \text{ z=} & \sum\limits_{j=1}^{n} c_j x_j \\ & \sum\limits_{j=1}^{n} a_{ij}x_j \leq b_i \qquad i\text{= }1,\ldots,m \\ & x_j \geq 0 \qquad j\text{= }1,\ldots,n \end{cases}$$

• Un programme linéaire est dit sous forme standard s'il est sous forme :

$$(PL)\begin{cases} opt\text{z=} & \sum\limits_{j=1}^{n} c_j x_j \\ & \sum\limits_{j=1}^{n} a_{ij}x_j = b_i \qquad i\text{= }1,\ldots,m \\ & x_j \geq 0 \qquad j\text{= }1,\ldots,n \end{cases}$$

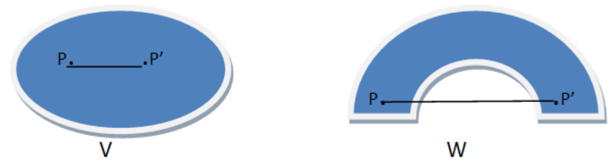
• Toute inégalité $\geq$ ($resp$ $\leq$) peut être transformée en égalité ,on rajoutant des variables d'écart.
• Toute inégalité $\geq$ est équivalente à une inégalité $\leq$ en multipliant ses termes par (-1).

1.1.4 Polyèdre et polytope

1. **Définitions :**

   •H=$\{x \in R^n/ax = b, a \in R^n\}$ définie un hyperplan H de dimension *(n-1)* et divise R^n en deux demi-espaces convexes :
   H_+=$\{x \in R^n/ax \geq b\}$= $\{x \in R^n/a_{i1}x_1 + ... + a_{ij}x_j + ... + a_{in}x_n \geq b_i\}$.
   i= $1,\ldots,m$ et j= $1,\ldots,n$.
   H_-=$\{x \in R^n/ax \leq b\}$= $\{x \in R^n$ / $a_{i1}x_1 + ... + a_{ij}x_j + ... + a_{in}x_n \leq b_i\}$.
   i= $1,\ldots,m$ et j= $1,\ldots,n$. [7]
   • Un ensemble E est dit convexe si :
   $\forall x, y \in E, [\lambda x + (1 - \lambda)y] \in E$ pour tout λ,$0 \leq \lambda \leq 1$.
   Géométriquement :
   V est dit convexe si tout segment de droite dont les extrémités appartiennent à V est inclus dans V .
   Considérons les deux ensembles illustrés ci-dessous :

W est non convexe,car les points P et P' sont les extrémités d'un segment dont au moins un point n'appartient pas à W.



2. **Polyèdre :**

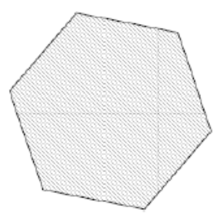
   Un polyèdre $P \subset R^n$ est l'ensemble des solutions d'un système fini d'inégalités linéaires .

   $$P=\{x \in R^n / Ax \leq b\}.$$

   C'est l'intersection d'un nombre fini de demi-espaces
   $H_{-}=\{x \in R^n / ax \leq b\}= \{x \in R^n \ / \ a_{i1}x_1 + ... + a_{ij}x_j + ... + a_{in}x_n \leq b_i\}.a \in R^n, b \in R^n$

3. **polytope :**

   Un polytope convexe est un polyèdre convexe et borné .



polytope

1.1.5 Optimum global

Une solution x^* est dite optimale (ou optimum global) si les deux contraintes suivantes sont vérifiées :
1. Elle est réalisable : tirée de l'ensemble de solutions possibles et respecte toutes les contraintes du problème posé.
2. on a :

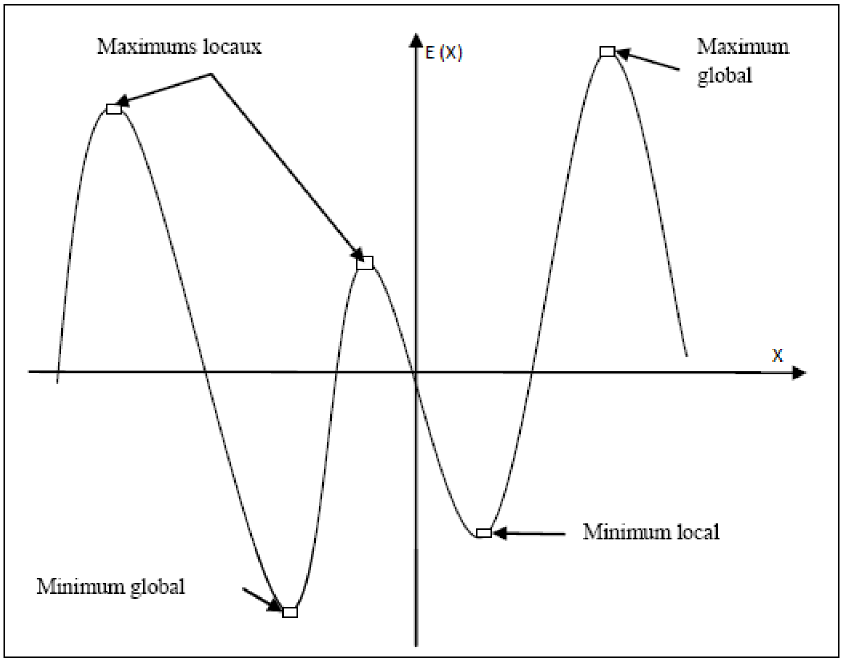
$$\begin{cases} f(x^*) = \max\limits_{x \in X}\{f(x)\} \text{ en cas de problème de } \textit{maximisation}. \\ f(x^*) = \min\limits_{x \in X}\{f(x)\} \text{ en cas de problème de } \textit{minimisation}. \end{cases}$$

Où X est l'ensemble des solutions réalisables .
Autrement dit, l'optimum global est le meilleur optimum local. Ainsi, une solution est dite optimum global si :

$$\begin{cases} f(x^*) \geq f(x) \forall x \in E(X) \text{ en cas de problème de } \textit{maximisation}.. \\ f(x^*) \leq f(x) \forall x \in E(X) \text{ en cas de problème de } \textit{minimisation}. \end{cases}$$

Où E(X) est l'ensemble d'optimums locaux.
Il est à noter que l'optimum global est aussi nommé maximum global au cas de problème de maximisation et minimum global au cas de problème de minimisation.

La figure suivante représente une courbe représentant les optimums locaux et les optimums globaux d'une fonction d'évaluation. [2]



Courbe représentant les optimums locaux et les optimums globaux

1.1.6 Relaxation

La relaxation d'un problème $P^{(i)}$ consiste à l'élargir en un autre problème tel que :

$$P^{(i)} \begin{cases} F_i = & minF(x) \\ & x \in Z_+^n \end{cases} \Rightarrow (PR^{(i)}) \begin{cases} F_i^R = & minF(x) \\ & x \in R_+^n \end{cases}$$

De sorte que $F_i^R \leq F_i$ pour un problème de minimisation et respectivement $F_i^R \geq F_i$ pour un problème de maximisation.
En général dans notre problème ,la relaxation linéaire (PR)de (P) est obtenue par relâchement des contraintes d'intégrité .[4]

1.2 Classification des problèmes

1.2.1 Les contraints et l'objectif

Parmi les problèmes de la programmation linéaire on trouve :

- **Les problèmes linéaires en variables mixtes :**
  Un problème linaire en variables mixtes est un problème d'optimisation pour lequel certaines variables sont entières et d'autres sont réelles .

$$
(PL)\begin{cases}
opt \text{ Z}= & max(minZ)=\sum\limits_{j=1}^{n} c_j x_j \\
 & \sum\limits_{j=1}^{n} a_{ij}x_j \lesseqqgtr b_i, \quad i\in\{1,...,m\} \\
 & \forall j\in J_1, x_j \in R^+, \quad j\in\{1,...,n\} \\
 & \forall j\in J_2, x_j \in \mathbb{N}, \quad j\in\{1,...,n\} \\
 & J=J_1\cup J_2, card(J)=n.
\end{cases}
$$

- **Les problèmes linéaires en nombres entiers(PLNE) :**
  Un problème linéaires en nombres entiers est un problème d'optimisation dont toutes les variables sont entières .

$$
(PL)\begin{cases}
opt \text{ Z}= & max(minZ)=\sum\limits_{j=1}^{n} c_j x_j \\
 & \sum\limits_{j=1}^{n} a_{ij}x_j \gtreqqless b_i, \quad i\in\{1,...,m\} \\
 & x_j\in\mathbb{N} \quad j\in\{1,...,n\}
\end{cases}
$$

- **Les problèmes linéaires en variables binaires :**
  Un problèmes linéaire en variables binaires ou **(0 et 1)** est un problème d'optimisation en nombres entiers ,telle que les variables entières prennent leurs valeurs dans l'ensemble $\{0,1\}$.

$$
(PL)\begin{cases}
opt \text{ Z}= & max(minZ)=\sum\limits_{j=1}^{n} c_j x_j \\
 & \sum\limits_{j=1}^{n} a_{ij}x_j \gtreqqless b_i, \quad i\in\{1,...,m\} \\
 & x_j\in\{0,1\} \quad j\in\{1,...,n\}
\end{cases}
$$

1.2.2 La complexité

Afin de mesurer la difficulté d'un problème donné et la comparer avec celles des autres problèmes , nous pouvons calculer la complexité algorithmique de chacun d'entre eux.
La complexité d'un problème donné est discutée sur deux cotés :
Le coté temporel "complexité temporelle".
Le coté spatial "complexité spatiale".
Bien que la théorie de la complexité se concentre sur des problèmes de décision, elle peut être étendue aux problèmes d'optimisation ,elle classe les problèmes selon leurs complexités en deux classes principales :

- **La classe P (Polynomial time) :**
  Un problème est dit polynomial s'il existe un algorithme polynomial pour le résoudre,la classe P est une classe d'algorithme efficace et facile à résoudre .
  Exemple :le problème du plus court chemin .
- **La classe NP (Non déterministe Polynomial time) :**
  Un problème P est dit NP s'il peut être résolu par une machine de Turing non déterministe .
  Exemple :la recherche d'un cycle hamiltonien .

En outre, la classe NP partage les problèmes de la classe NP en deux sous classes :

- **La classe NP-complet** :
  Un problème est dit NP-complet lorsqu'il est dans la classe NP et si on peut le ramener par une transformation polynomial à un problème de la classe NP.
  Exemple :problème de voyageur de commerce .
- **La classe NP-difficile :**
  Un problème P est dit NP-difficile s'il existe une réduction polynomial de problème de satis fiabilité à P .
  Exemple :les PLNE (problèmes linéaire en nombres entiers).[2]

1.3 Méthodes d'optimisation

1.3.1 Les méthodes exactes

La méthode de résolution exacte doit permettre l'obtention d'au moins une solution optimale . Ce genre de méthode demande en général des temps d'exécution très élevés ainsi que des ressources mémoire importantes sur les instances de grande taille
Nous citons quelques méthodes de la classe des algorithmes exacts, ces méthodes donnent une garantie de trouver la solution optimale pour une instance de taille finie dans un temps limité et de prouver son optimalité .

- La méthode de séparation et d'evaluation Branche and Bound.
- La méthode des coupes planes (cutting-plane).
- La méthode (Branch and Cut).

1.3.2 Les méthodes approchées(heuristiques)

En optimisation combinatoire, une heuristique est un algorithme approché qui permet d'identifier en un temps polynomial au moins une solution réalisable rapide, pas obligatoirement optimale. L'usage d'une heuristique est efficace pour calculer une solution approchée d'un problème et ainsi accélérer le processus de résolution . Généralement une heuristique est conçue pour un problème particulier, en s'appuyant sur sa structure propre sans offrir aucune garantit quant à la qualité de la solution calculée. Les heuristiques peuvent être classées en deux catégories :
- Méthodes constructives qui génèrent des solutions à partir d'une solution initiale en essayant d'en ajouter petit à petit des éléments jusqu'à ce qu'une solution complète soit obtenue.
- Méthodes de fouilles locales qui démarrent avec une solution initialement complète (probablement moins intéressante) et de manière répétitive essaie d'améliorer cette solution en explorant son voisinage.

1.3.3 Comparaison entre les méthodes exactes et les méthodes approchées

1. Les méthodes exactes :
   - Elles trouvent la solution optimale.
   - Elles peuvent prendre un nombre exponentiel d'itérations.
2. Les heuristiques :
   - Elles produisent une solution sous-optimale.
   - Elles ne produisent pas de mesure de qualité de la solution.
   - En général, elles ne prennent pas un nombre exponentiel d'itérations.

Chapitre 2

Méthodes de résolution

2.1 Introduction

La résolution de différentes sortes de problèmes rencontrés dans notre vie quotidienne a poussé les chercheurs à proposer des méthodes de résolution et à réaliser de grands efforts pour améliorer leurs performances en termes de temps de calcul nécessaire et de la qualité de la solution proposée. Au fil des années, de nombreuses méthodes de résolution de problèmes de différentes complexités ont été proposées.
Les méthodes de résolution exactes sont nombreuses et se caractérisent par le fait qu'elles permettent d'obtenir une ou plusieurs solutions dont l'optimalité est garantie.
Parmi ces méthodes, on peut citer la méthode graphique ,l'algorithme du simplexe et l'algorithme dual du simplexe.

2.2 Résolution graphique

La méthode graphique permet la résolution de problèmes linéaires simples .
Cette méthode est limitée aux problèmes à deux ou trois variables de décision puisqu'il n'est pas possible d'illustrer graphiquement plus de trois dimensions. Bien qu'en général on puisse difficilement trouver des problèmes avec seulement deux ou trois variables de décision, cette méthodologie de résolution est cependant très utile. Lors de la reproduction graphique de situations possibles, tels que :
L'existence d'une solution optimale unique, des solutions optimales alternatives.
La non -existence de solution et l'absence de partie bornée.
La méthode graphique nous offre une aide visuelle pour interpréter et comprendre l'algorithme de la méthode du Simplexe.
Les étapes de résolution, de problèmes par la méthode graphique sont les suivantes :

- Créer un système de coordonnées cartésiennes, dans lequel chaque variable de décision est représentée par un axe.

- Établir une échelle de mesure pour chacun des axes appropriés à sa variable associée.

- Dessiner dans le système de coordonnées les contraintes du problème,y compris celles des variables de décision .

– Remarquer qu'une inéquation précise une région qui sera le demi-plan limité par la ligne droite, qui représente la contrainte qu'on considère comme une contrainte d'égalité alors que si une équation linéaire détermine une région c'est la ligne droite, elle-même .

– L'intersection de toutes les régions détermine la région ou l'espace faisable (qui est un ensemble convexe).
Si non , si il n'y a pas de point qui satisfait toutes les contraintes simultanément de sorte que le problème ne sera pas résolu,on dit que le problème est infaisable .

– Déterminer les points extrêmes ou les sommets du polyèdre qui forme la région faisable.

– Ces points seront les candidats à la solution optimale.

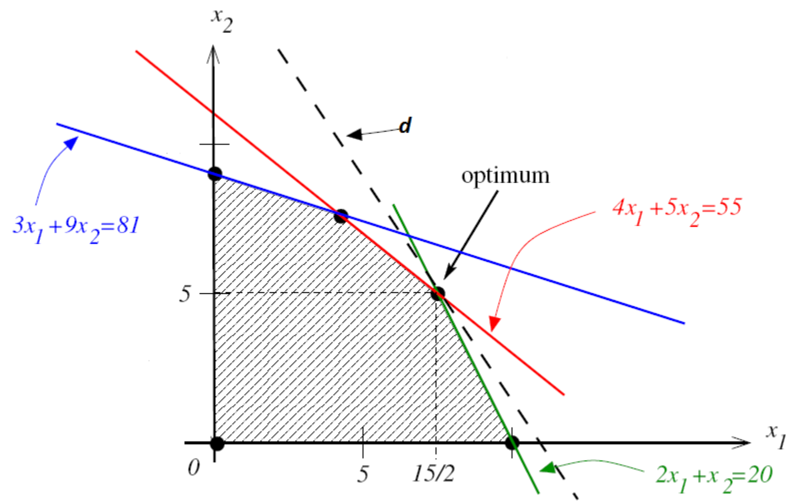
– Évaluer la fonction objectif à chaque sommet et celui qui maximise la fonction objectif définie la solution optimale (pour un problème de maximisation) .[8]

• **Exemple 01 :**
Soit le problème suivant :

$$(P)\begin{cases} \text{max Z=} & 6x_1 + 4x_2 \\ 3x_1 + 9x_2 & \leq 81 \\ 4x_1 + 5x_2 & \leq 55 \\ 2x_1 + x_2 & \leq 20 \\ x_1, x_2 \geq 0 & \end{cases}$$

– Les contraintes où apparaissent des inégalités correspondent géométriquement à des demi-plans.

– L'intersection des demi-plans est un polyèdre qui représente l'ensemble de solutions réalisables du problème .



•Notations :
J_B :Ensemble des indices de variables de base.
J_H :Ensemble des indices hors base.
A_B :Sous matrice de A d'ordre m*m.
A_H :Sous matrice de A d'ordre m*(n-m).
$\min_{x \in X} f(x) = -\max_{x \in X}(-f(x))$.

2.3 Méthode du Simplexe

Dans la plupart des problèmes réels on a plus de deux variables à déterminer .
Une procédure algébrique pour résoudre les programmes linéaires avec plus de deux variables fera l'objet de cette section.C'est la méthode du simplexe.
La méthode du simplexe est une méthode itérative,elle démarre d'un point réalisable (sommet de départ) et passe de ce point à un autre en augmentant la valeur de la fonction objectif dans le cas de maximisation .
On arrête le déroulement de l'algorithme lorsqu'il n'est plus possible d'augmenter la valeur de la fonction objectif .

2.3.1 L'algorithme du simplexe

Le principe de résolution nécessite un certain nombre d'étapes dont la démarche est la suivante :
On l'applique pour un problème de maximisation :

1. Écrire le système sous la forme standard.

2. Le point de départ de l'algorithme est l'existence d'une solution réalisable soit :
   $x = (x_B, x_H) = (x_B, 0)$ et A_B^{-1} :La matrice inversible de A_B.
3. Construire le premier tableau correspondant à la forme standard.

4. Calculer les $E_j = z_j - c_j$, tq : $Z_j = C_B^T * A_B^{-1}$.
(critère d'optimalité).

5. Si tous les $E_j \geq 0$,$\forall j \in J_H$,$1 \leq j \leq (n-m)$.La solution $(x_B, 0)$ est optimale.
Si non :
   – Si $\exists k \in J_H$ tel que :$E_k < 0$ et $A_B^{-1}a_k < 0$.
   Le (PL) n'admet pas un maximum (non borné), arrêt .
   – Sinon si $\exists k \in J_H$ tel que :$E_k < 0$ et $A_B^{-1}a_k > 0$ alors ,aller à 6) :

6. Choisir la variable x_k qui entre dans la base tq :E_k=$\min\limits_{j \in J_H} E_j,\ E_j < 0$.

7. Choisir la variable x_l qui sort de la base tel que :
θ_l=min$\{\dfrac{b_i}{a_{ik}}, a_{ik} > 0, 1 \leq i \leq m\}$,la variable sortante se trouve sur la $l^{éme}$ ligne .

8. Encadrer le pivot a_{lk} .

9. Multiplier la ligne du pivot par le rapport : $\dfrac{1}{a_{lk}}$.

10. Calculer les valeurs a'_{ij} des autres lignes :
a'_{ij}=$a_{ij} - (\dfrac{a_{ik} * a_{lj}}{a_{lk}})$
Aller à 4).

• **Exemple 02 :**
On prend un exemple et on applique le simplexe .

1. Écrire le système sous la forme standard :

   Il s'agit de convertir le programme établi sous la forme canonique (système d'inéquation) a la forme standard (système d'équation avec variables d'écarts).

   • La forme canonique est :

$$(P)\begin{cases} \text{max Z}= 30x_1 + 50x_2 & \\ 3x_1 + 2x_2 & \leq 1800 \\ x_1 & \leq 400 \\ x_2 & \leq 600 \\ x_1 \geq 0, x_2 \geq 0. & \end{cases}$$

   • La forme standard est :

   x_3,x_4,x_5 représentent les variables d'écarts.

$$(P)\begin{cases} \text{max Z}= 30x_1 + 50x_2 & \\ 3x_1 + 2x_2 + x_3 & = 1800 \\ x_1 + x_4 & = 400 \\ x_1 + x_5 & = 600 \\ x_j \geq 0, j = 1, .., 5. & \end{cases}$$

2. La solution de départ est :
   X=(0,0,1800,400,600).

3. Construire le premier tableau correspondant à la forme standard :

		c_i	30	50	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_3	1800	3	2	1	0	0
0	x_4	400	1	0	0	1	0
0	x_5	600	0	1	0	0	1

4. Calculer les $E_j = z_j - c_j$.

		c_i	30	50	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_3	1800	3	2	1	0	0
0	x_4	400	1	0	0	1	0
0	x_5	600	0	1	0	0	1
		E_j	-30	-50	0	0	0

5. test d'optimalité .
   $\exists E_j \leq 0$ d'où aller à 6).

6. Choisir la variable x_k qui entre dans la base tq :E_k=min E_j.
   min E_j=-50 d'où $x_k = x_2$ entre dans la base.

7. Choisir la variable x_l qui sort de la base tq :
   θ_l=min$\{\frac{b_i}{a_{ik}}, a_{ik} > 0\}$.

		c_i	30	50	0	0	0	
c_b	base	b	x_1	x_2	x_3	x_4	x_5	θ
0	x_3	1800	3	2	1	0	0	1800/2
0	x_4	400	1	0	0	1	0	/
0	x_5	600	0	1	0	0	1	600/1
		E_j	-30	-50	0	0	0	

   $\theta_l = \theta_3$=600/1 d'où $x_l = x_5$ sort de la base(car elle se trouve à la $3^{éme}$ligne).

8. Encadrer le pivot :a_{32}

		c_i	30	50	0	0	0	
c_b	base	b	x_1	x_2	x_3	x_4	x_5	θ
0	x_3	1800	3	2	1	0	0	1800/2
0	x_4	400	1	0	0	1	0	/
0	x_5	600	0	$\boxed{1}$	0	0	1	600/1
		E_j	-30	-50	0	0	0	

9. Multiplier la ligne du pivot par le rapport : $\dfrac{1}{a_{lk}}$:

		c_i	30	50	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_3						
0	x_4						
50	x_2	600	0	1	0	0	1

10. Calculer les valeurs a'_{ij} des autres lignes :
    Cette opération consiste à transformer a_{ij} des autres lignes en $a_{ij'}$, nous effectuons un calcul matriciel.
    $a'_{ij}=a_{ij}-(\dfrac{a_{ik}*a_{lj}}{a_{lk}})$.

		c_i	30	50	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_3	600	3	0	1	0	-2
0	x_4	400	1	0	0	1	0
50	x_2	600	0	1	0	0	1

Aller à l'étape 4).

•Nouveau passage :
On refait les même étapes et on aura les résultats suivants :

		c_i	30	50	0	0	0	
c_b	base	b	x_1	x_2	x_3	x_4	x_5	θ
0	x_3	600	**3**	0	1	0	-2	200
0	x_4	400	1	0	0	1	0	400
50	x_2	600	0	1	0	0	1	/
		E_j	-30	0	0	0	50	

x_1 entre dans la base.
x_3 sort de la base.

		c_i	30	50	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
30	x_1	200	1	0	1/3	0	-2/3
0	x_4	200	0	0	-1/3	1	2/3
50	x_2	600	0	1	0	0	1
		E_j	0	0	10	0	30

• Test d'optimalité :
Tous les E_j sont positifs . La solution de ce problème est :

$$\begin{bmatrix} \text{max Z=36000} \\ x_1\text{=200} \\ x_2\text{=600} \end{bmatrix}$$

• Remarque :
L'optimisation d'un programme linéaire à l'aide de l'algorithme du simplexe nous oblige parfois à introduire des variables artificielles pour obtenir une solution de départ lorsque les contraintes sont de type :
$\sum\limits_{j=1}^{n} a_{ij}x_j \geq b_i$, $i \in \{1, ..., m\}$.
On applique la M-méthode ou la méthode de deux phases.

2.4 Méthode dual du simplexe

Pour éviter l'utilisation des variables artificielles ,quand on a un problème avec des contraintes sous forme :
$\sum\limits_{j=1}^{n} a_{ij}x_j \geq b_i, i \in \{1, ..., m\}$.
On utilise l'algorithme dual du simplexe afin de réduire la taille du (PL) et le nombre d'itération .[6]

2.4.1 Principe de la dualité

Son principe est de garder le critère d'optimalité satisfait à chaque itération et de rendre positifs les b_i qui sont négatifs.

• Au programme linéaire primal :

$$(PL)\begin{cases} \text{max Z= } c^T x \\ Ax \leq b \\ x \geq 0 \end{cases}$$

On associe le programme linéaire dual :

$$(PLD)\begin{cases} \text{minZ'= } b^T y \\ A^T y \geq c \\ y \geq 0 \end{cases}$$

•Les différentes transformations sont résumées dans le tableau suivant :[1]

Primal	**Dual**
maxZ	minZ'
coefficient c de Z	second membre c
second membre b	coefficient b de Z'
i-éme contrainte =	$y_i \in R$
i-éme contraintes $\geq$	$y_i \geq 0$
i-éme contraintes $\leq$	$y_i \leq 0$
$x_j \in R$	j-éme contraintes =
$x_j \geq 0$	j-éme contraintes $\leq$
$x_j \leq 0$	j-éme contraintes$\geq$

- **Exemple 03 :**

Programme linéaire primal :

$$(PL) \begin{cases} \text{max Z}= 6x_1 + 4x_2 & \\ 3x_1 + 9x_2 & \leq 81 \\ 4x_1 + 5x_2 & \leq 55 \\ 2x_1 + x_2 & \leq 20 \\ x_1, x_2 \geq 0 & \end{cases}$$

On associe le programme linéaire dual :

$$(PLD) \begin{cases} \text{min Z'}= 81y_1 + 55y_2 + 20y_3 & \\ 3y_1 + 4y_2 + 2y_3 & \geq 6 \\ 9y_1 + 5y_2 + y_3 & \geq 4 \\ y_1, y_2, y_3 \geq 0 & \end{cases}$$

2.4.2 L'algorithme dual du simplexe

Le principe de résolution nécessite un certain nombre d'étapes contenues au travers de l'algorithme dual du simplexe dont la démarche est la suivante ici on l'applique pour un problème de maximisation :

1. Écrire le système sous la forme standard avec les $c_j \geq 0, \forall j = 1, ..., n$.

2. Construire le premier tableau correspondant à la forme standard.

3. Calculer les $E_j = Z_j - c_j$.

4. Choisir la variable x_r qui sort de la base :
   $b_r = \min\ b_i$ tq : $b_i < 0, i \in J_B$.
   Si b_r n'existe pas alors fin,l'optimum est atteint.
   Sinon aller à 5).

5. Choisir la variable x_k qui entre dans la base :
   $\dfrac{E_k}{a_{rk}} = \max\ \{\dfrac{z_j - c_j}{a_{rj}}\}$ tq :$a_{rj} < 0, j \in J_H$
   Le pivot doit être négatif, sinon fin,le (PL)est non borné.

6. Encadrer le pivot .

7. Multiplier la ligne du pivot par le rapport :$\dfrac{1}{a_{rk}}$.

8. Calculer les valeurs a'_{ij} des autres lignes :
   $a'_{ij} = a_{ij} - (\dfrac{a_{ik} * a_{rj}}{a_{rk}})$
   Aller à 3).

• **Exemple 04 :**
On prend un exemple et on applique le dual du simplexe .

1. Écrire le système :
   • Sous la forme canonique :

$$(P)\begin{cases} \max\ \text{Z}= -2x_1 - x_3 & \\ x_1 + x_2 - x_3 & \geq 5 \\ x_1 - 2x_2 + 4x_3 & \geq 8 \\ x_j \geq 0, \forall_{j=1..3} & \end{cases}$$

   •Sous la forme standard :

   x_4 ;x_5 représentent les variables d'écart.

$$(P)\begin{cases} \max\ \text{Z}= -2x_1 - x_3 & \\ x_1 + x_2 - x_3 - x_4 & = 5 \\ x_1 - 2x_2 + 4x_3 - x_5 & = 8 \\ x_j \geq 0, \forall_{j=1..5} & \end{cases}$$

   • On multiplie par (-1).

$$(P)\begin{cases} \max\ \text{Z}= -2x_1 - x_3 & \\ -x_1 - x_2 + x_3 + x_4 & = -5 \\ -x_1 + 2x_2 - 4x_3 + x_5 & = -8 \\ x_j \geq 0, \forall_{j=1..5} & \end{cases}$$

2. Construire le premier tableau correspondant à la forme standard :

		c_i	-2	0	-1	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_4	-5	-1	-1	1	1	0
0	x_5	-8	-1	2	-4	0	1

3. Calculer les $E_j = z_j - c_j$.

		c_i	-2	0	-1	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_4	-5	-1	-1	1	1	0
0	x_5	-8	-1	2	-4	0	1
		E_j	2	0	1	0	0

4. Choisir la variable x_r qui sort de la base :
   b_r = min b_i =-8.
   D'où $x_r = x_5$ sort de la base(x_5 qui se trouve à la $2^{éme}$ligne).

5. Choisir la variable x_k qui entre dans la base :
   $\frac{E_k}{a_{rk}}$= max $\{\frac{z_j - c_j}{a_{rj}}\} =\} = \max\{\frac{2}{-1}, \frac{1}{-4}\}$.
   D'où $x_k = x_3$ entre dans la base.

6. Encadrer le pivot :

		c_i	-2	0	-1	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_4	-5	-1	-1	1	1	0
0	x_5	-8	-1	2	$\boxed{-4}$	0	1
		E_j	2	0	1	0	0

7. Multiplier la ligne du pivot par le rapport : $\frac{1}{a_{rk}}$:

		c_i	-2	0	-1	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_4						
-1	x_3	2	1/4	-1/2	1	0	-1/4

8. Calculer les valeurs des autres lignes :
   a'_{ij}=$a_{ij} - (\frac{a_{ik} * a_{lj}}{a_{rk}})$
   Cette opération consiste à transformer a_{ij} des autres lignes en $a_{ij'}$, nous effectuons un calcul matriciel.

		c_i	-2	0	-1	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_4	-7	-5/4	-1/2	0	1	1/4
-1	x_3	2	1/4	-1/2	1	0	-1/4

Aller à l'étape 3).

•Nouveau passage :
On refait les même étapes et on aura les résultats suivants :

		c_i	-2	0	-1	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_4	-7	-5/4	**-1/2**	0	1	1/4
-1	x_3	2	1/4	-1/2	1	0	-1/4
		E_j	7/4	1/2	0	0	1/4

x_4 sort de la base.
x_2 entre dans la base.

		c_i	-2	0	-1	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5
0	x_2	14	5/2	1	0	-2	-1/2
-1	x_3	9	3/2	0	1	-1	-1/2
		E_j	1/4	0	0	1	0

• Test d'optimalité :
Tous les $b_i > 0$.
La solution de ce problème est :

$$\begin{bmatrix} \text{max Z=-9} \\ x_1\text{=0} \\ x_2\text{=14} \\ x_3\text{=9} \end{bmatrix}$$

2.5 Conclusion

Dans ce chapitre, nous avons constaté l'efficacité des deux algorithmes :le simplexe et le dual du simplexe à déterminer la solution optimale exacte .

Chapitre 3

La méthode de Séparation et Évaluation(BRANCH AND BOUND)

3.1 Introduction

Pour plusieurs problèmes , en particulier les problèmes combinatoires,l'espace de solutions est fini (dénombrable). Il est donc possible en principe d'énumérer toutes les solutions et ensuite de prendre la meilleure .L'inconvénient majeur de cette approche est le temps de calcul qui est en général énorme . La méthode de branch and bound "procédure par évaluation et séparation" est basée sur une méthode arborescente qui consiste à réduire par des découpages l'ensemble des solutions qui ne génèrent pas de meilleures solutions. Toutes les séparations sont permises à condition de ne perdre aucune information. La complexité algorithmique diminue alors dans la mesure où on ne calcule pas toutes les solutions du problème.[5]

3.2 Principe de la méthode

Il est aisé en général de déterminer l'ensemble de solutions réalisables du problème.Mais cet ensemble est généralement trop grand pour qu'il soit possible d'en extraire la solution optimale .En conséquence ,on procède à une subdivision de cet ensemble S en un nombre fini de sous-ensembles $S^{(i)}_{i=1...p}$ de plus en plus petits en veillant à ce que :

$$\bigcup_{i=1}^{p} S^{(i)} = S$$

jusqu'à l'obtention d'un sous-problème suffisamment restreint "ensemble sondé"pour qu'on puisse extraire la solution optimale . La méthode de branch and bound ,est essentiellement présentée par les trois procédures suivantes :

- **La procédure de séparation** :

  La séparation consiste à diviser le problème en sous-problèmes .
  Son principe doit satisfaire aux trois règles suivantes :

  a)La règle de finitude :le nombre totale de nœuds engendrés doit être fini.
  b)La règle de conservation : aucune solution d'un sous-problème ne peut être éliminée

par la séparation c'est-à-dire

$$\bigcup_{k=1}^{p} S^{(ik)} = S^{(i)}$$

et qu'il vérifie également $S^{(ik)} \cap S^{(il)} = \emptyset \quad k \neq l$,où $S^{(ik)} \quad k = 1, ..., p$ représentent les sous-ensembles (enfants)du sous-ensemble (parent)$S^{(i)}$.
c)La Règle d'arrêt :Un nœud ou sous-ensemble terminal de l'arborescence noté $S^{(t)}$ est défini comme un nœud qu'il n'est plus possible de séparer ,pour un tel sous-ensemble :
-Soit $S^{(t)} = \emptyset$
-Soit qu'il est possible de déterminer une solution optimale du problème $p^{(i)}$ qui est le sous-problème réduit de (P) et qui est défini comme suit :

$$P^{(i)} \begin{cases} \min & F(x) \\ & x \in S^{(i)} \end{cases}$$

- **La procédure d'évaluation** :

  L'évaluation d'un sous-problème $P^{(i)}$ de (P) consiste à évaluer la valeur optimale de sa fonction économique .Plus précisément à déterminer une borne supérieure si la fonction économique est à maximiser respectivement une borne inférieure si elle est à minimiser .
  L'évaluation permet de réduire l'espace de recherche en éliminant quelques sous-ensembles qui ne contiennent pas la solution optimale et elle a pour but de déterminer le sous problème qu'on doit séparer.
  L'exploration d'une branche est éliminée si :

  a) Le sommet de l'arborescence ne peut être séparé c.à.d que le sommet(nœud)de l'arborescence est associé à une solution entière .

  b) Le sous-problème est non-réalisable .

  c) La valeur de Z associée au sous-problème est supérieure à Z^{opt} si le problème est à maximiser respectivement inférieur à Z^{opt} si le problème est à minimiser .

  d) La valeur Z d'un sommet associée à une solution non-entière inférieure ou égale à une valeur Z d'une solution entière respectivement supérieure pour un problème de minimisation .

- **La procédure de cheminement** :

  Cette procédure indique le sous-ensemble à analyser et dans quel ordre.
  Bien évidemment,il est souhaitable d'examiner le moins possible de sous-problèmes selon la stratégie choisie , certains d'entre eux pourront ne pas être séparés car, par exemple,leurs analyse mettra en évidence qu'ils ne contiennent pas de solutions meilleures que celles déjà trouvées .Nous dirons qu'un tel sous-ensemble ou le nœud correspondant de l'arborescence est sondé.C'est parce que certains sous-ensembles de solutions ne devront pas être examinés explicitement, que la méthode «B and B» est parfois appelée méthode d'énumération implicite.

Lorsqu'un nœud de l'arborescence est sondé,il conviendra de remonter dans l'arborescence vers un autre nœud situé à un niveau inférieure ou égale .

3.2.1 Stratèges de parcours

Il est évident d'examiner la totalité des sommets de l'arborescence pour réaliser une énumération implicite efficace .
Pour savoir quel sommet doit-on séparer on utilise des stratégies ,on peut distinguer trois d'entre elles :
La largeur d'abord :
Cette stratégie favorise les sommets les plus proches de la racine (nœud père) en faisant moins de séparations du problème initial. Elle est moins efficace que les deux autres stratégies .
La profondeur d'abord :
Cette stratégie avantage les sommets les plus éloignés de la racine (de profondeur la plus élevée) en appliquant plus de séparations au problème initial. Cette voie mène rapidement à une solution optimale en économisant la mémoire.
Le meilleur d'abord :
Cette stratégie consiste à explorer les sous-problèmes possédant la meilleure borne. Elle permet aussi d'éviter l'exploration de tous les sous-problèmes qui possèdent une mauvaise évaluation par rapport à la valeur optimale.

3.3 Algorithme général

X^* : la solution du $(P) \to$ problème (PLNE)
X^{opt} :la solution du $(PR) \to$ problème (PLNR)
Z^* : la fonction économique du $(P) \to$ problème (PLNE)
Z^{opt} :la fonction économique du $(PR) \to$ problème (PLNR)
1)Résolution du problème relaxé(PR) par la méthode du simplexe :

-Si X^{opt} est entier :fin.
-Sinon, aller à 2).

2)Initialisation :

Soit Z^{opt} obtenu dans(PR) une borne supérieure pour un problème de maximisation respectivement (borne inférieure pour un problème de minimisation)pour (P_i).
Soit n_1 le sommet initial de l'arborescence et son ensemble$(S_1 = S)$.
$Z_1 = Z^{opt}$.

3)Séparation($k^{ième}$ itération) :

Choisir une variable non entière x_l,créer deux branches (deux sommets fils n_{i+1} et n_{i+2}),on obtient deux sous-problèmes sous la forme :

$$\begin{cases} P_{i+1,k} : P_i + \text{ la contrainte } x_l \leq [x_l] \\ P_{i+2,k} : P_i + \text{ la contrainte } x_l \geq [x_l] + 1 \\ \end{cases}$$

Avec $[x_l]$: la partie entiére de x_l

4)Résolution des sous-problèmes :

Résoudre chaque sous-problème en utilisant le simplexe ou le dual simplexe.

5)Évaluation :

Examiner chaque sous-ensemble :
On peux tailler un sommet si :
* La solution est non réalisable.
* $Z_1 \leq$Z pour un problème de maximisation ($Z_1 \geq$Z pour un problème de minimisation), avec Z la solution du sous-problème.
* La solution est non entière et son Z inférieur ou égale à une solution entière pour un problème de maximisation (supérieure ou égale pour un problème de minimisation) .
Il est inutile de sépare si :
* La solution est entière.

6)test :

S'il y a plus de sous-ensembles à séparer ,alors :
On compare tous les Z des solutions entières et on prend la plus grande d'entre elles soit Z^* pour un problème de maximisation (la plus petite pour un problème de minimisation). Elle sera la valeur de la fonction économique de la solution optimale X^* de notre problème (P).
Sinon retour à 3).

Remarque :

$Z^* \leq Z^{opt}$ pour un problème de maximisation ($Z^* \geq Z^{opt}$ pour un problème de minimisation).
Dans l'itération 3)pour séparer les sous-problèmes on utilise l'une des stratégies cités précédemment.

3.4 Exemple d'application

Forme générale du problème :

$$(P)\begin{cases} (PR)\begin{cases} \max \text{Z}= x_1 + 2x_2 & \\ 4x_1 - 3x_2 & \leq 2 \\ -2x_1 + x_2 & \leq 1 \\ -6x_1 + 14x_2 & \leq 35 \\ x_1, x_2 \geq 0 & \end{cases} \\ x_1, x_2 \in \mathbf{N} \end{cases}$$

1)Résolution du problème (PR) :

- Sa forme standard :

$$(PR)\begin{cases} maxZ = x_1 + 2x_2 & \\ 4x_1 - 3x_2 + x_3 & = 2 \\ -2x_1 + x_2 + x_4 & = 1 \\ -6x_1 + 14x_2 + x_5 & = 35 \\ x_j \geq 0, j = 1, ..., 5. & \end{cases}$$

• Résoudre le(PR)par la méthode du simplexe :

		c_i	1	2	0	0	0	
c_b	base	b	x_1	x_2	x_3	x_4	x_5	θ
0	x_3	2	4	-3	1	0	0	/
0	x_4	1	-2	**1**	0	1	0	1
0	x_5	35	-6	14	0	0	1	35/14
		E_j	-1	**-2**	0	0	0	

•x_4 sort de la base.
•x_2 entre dans la base.

		c_i	1	2	0	0	0	
c_b	base	b	x_1	x_2	x_3	x_4	x_5	θ
0	x_3	5	-2	0	1	3	0	/
2	x_2	1	-2	1	0	1	0	/
0	x_5	21	**22**	0	0	-14	1	21/22
		E_j	**-5**	0	0	2	0	

•x_5 sort de la base.
•x_1 entre dans la base.

		c_i	1	2	0	0	0	
c_b	base	b	x_1	x_2	x_3	x_4	x_5	θ
0	x_3	76/11	0	0	1	**19/11**	1/11	76/19
2	x_2	32/11	0	1	0	-3/11	1/19	/
1	x_1	21/22	1	0	0	-7/11	1/22	/
		E_j	0	0	0	**-13/11**	5/22	

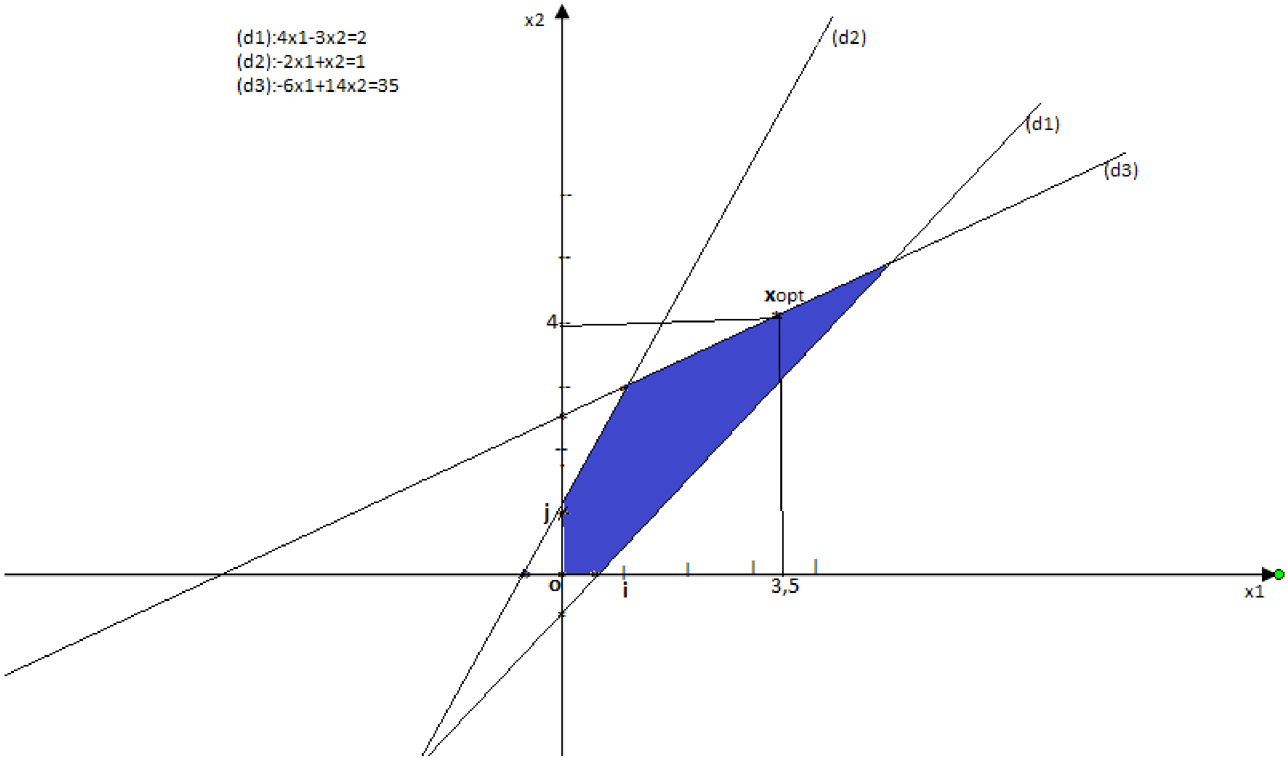
•x_3 sort de la base.
•$x_4$entre dans la base.

		c_i	1	2	0	0	0	
c_b	base	b	x_1	x_2	x_3	x_4	x_5	θ
0	x_4	4	0	0	11/19	1	1/19	/
2	x_2	4	0	1	3/19	0	2/19	/
1	x_1	7/2	1	0	7/19	0	3/38	/
		E_j	0	0	13/19	0	11/38	

• tous les $E_j \geq 0 \Rightarrow X^{opt} = (7/2, 4)$.
• Z^{opt}=(7/2)+2(4)=23/2.

$$d'ou \begin{cases} x^{opt}\text{=(3.5,4).} \\ Z^{opt}\text{=11.5.} \end{cases}$$

• **Résolution graphique de (PR) :**



2)Initialisation :

• Z^{opt}est la borne supérieure de (P).

• Soit (P_1) le problème initial (le sommet initial de l'arborescence) tel que :

$$\begin{bmatrix} Z_1 = Z^{opt} = 11.5. \\ X_1 = 3.5. \\ X_2 = 4. \end{bmatrix}$$

(P_1) a des variables qui ne vérifient pas les contraintes d'intégrités d'où on passe à l'étape 3).

3)Séparation :

x_1 n'est pas entier on aura le modèle linéaire courant, ici (P_1) est divisé en deux sous-problèmes sous la forme :

$$\begin{cases} (P_2) = (P_1) + \textit{contrainte } x_1 \leq [x_1]. \\ (P_3) = (P_1) + \textit{contrainte } x_1 \geq [x_1] + 1. \end{cases}$$

avec $[x_1]$ la partie entière de x_1.

4)Résolution des sous-problèmes :

a)On résout (P_2) tq :

$(P_2) = (P_1)+$ la contrainte $x_1 \leq [x_1]$ c-à-d :

$(P_1)+$ la contrainte $x_1 \leq 3 \Rightarrow x_1 + x_6 = 3$......★.

On rajoute au dernier tableau de (PR)la contrainte★et on applique le dual du simplexe :

		c_i	1	2	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6
0	x_4	4	0	0	11/19	1	1/19	0
2	x_2	4	0	1	3/19	0	2/19	0
1	x_1	7/2	1	0	7/19	0	3/38	0
0	x_6	3	**1**	0	0	0	0	1

•x_1 est une variable de base,on multiplie la $3^{éme}$ ligne par(-1) , on l'additionne à la $4^{éme}$ ligne . On obtient le tableau suivant :

		c_i	1	2	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6
0	x_4	4	0	0	11/19	1	1/19	0
2	x_2	4	0	1	3/19	0	2/19	0
1	x_1	7/2	1	0	7/19	0	3/38	0
0	x_6	**-1/2**	0	0	**-7/19**	0	-3/38	1
		E_j	0	0	13/19	0	11/38	0

•x_6 sort de la base.
•x_3 entre dans la base.

		c_i	1	2	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6
0	x_4	45/14	0	0	0	1	-1/14	11/7
2	x_2	53/14	0	1	0	0	1/14	3/7
1	x_1	3	1	0	0	0	0	1
0	x_3	19/14	0	0	1	0	3/14	-19/7
		E_j	0	0	0	0	1/7	13/7

• Tous les $b_j \geq 0$,la solution de (P_2) est :

$$\begin{bmatrix} Z_2{=}10.57. \\ X_1{=}3. \\ X_2{=}3.79. \end{bmatrix}$$

b)On résout (P_3) tq :
(P_3)=(P_1) +la contrainte $x_1 \geq [x_1]+1$ c-à-d :
(P_1) +la contrainte $x_1 \geq 4 \Rightarrow x_1$-$x_6$=4
$-x_1+x_6 = -4....\bigstar$.
On rajoute la contrainte $\bigstar$ au dernier tableau de (PR)et on applique le dual simplexe :

		c_i	1	2	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6
0	x_4	4	0	0	11/19	1	1/19	0
2	x_2	4	0	1	3/19	0	2/19	0
1	x_1	7/2	1	0	7/19	0	3/38	0
0	x_6	-4	**-1**	0	0	0	0	1

•x_1 est une variable de base,on additionne la $3^{éme}$ ligne à la $4^{éme}$ ligne.On obtient le tableau suivant :

		c_i	1	2	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6
0	x_4	4	0	0	11/19	1	1/19	0
2	x_2	4	0	1	3/19	0	2/19	0
1	x_1	7/2	1	0	7/19	0	3/38	0
0	x_6	-1/2	0	0	7/19	0	3/38	1
		E_j	0	0	13/19	0	11/38	0

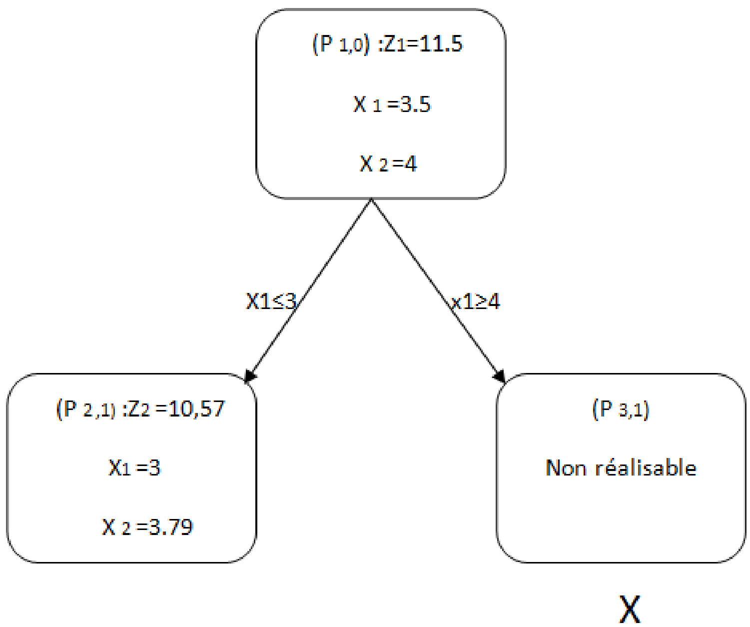
- Pas de pivot négatif ,(P_3) est non-réalisable .

5)Évaluation :

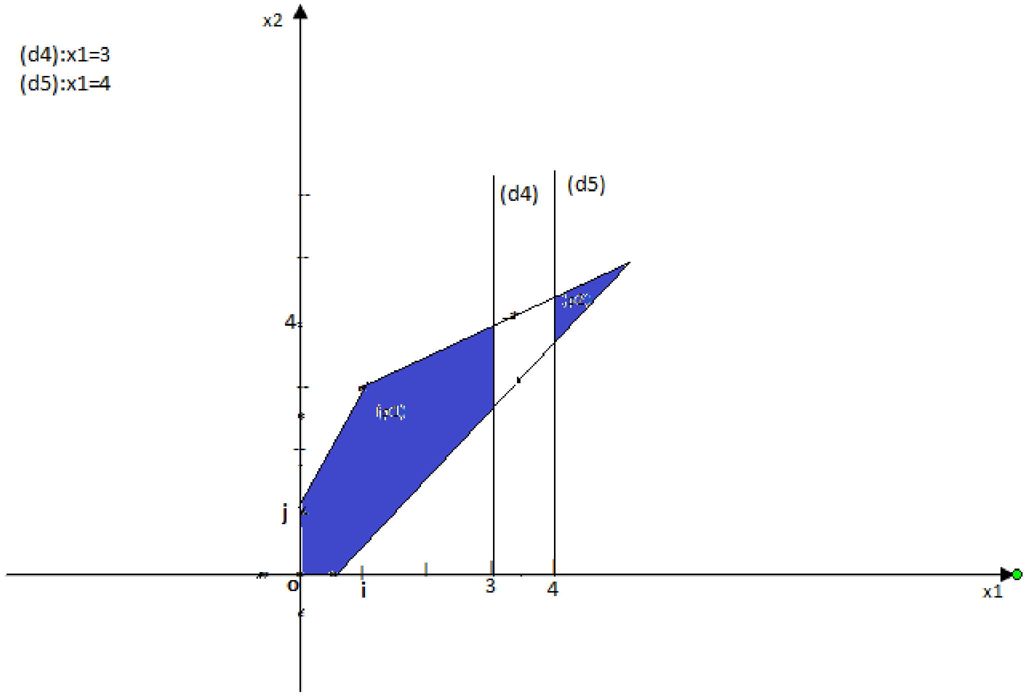
•(P_2)admet une solution non-entière et la valeur de $Z_2 = 10.75 \leq Z_1$ donc on peut séparer le sommet S_2 .
•(P_3) non réalisable donc on taille le sommet S_3 .

6) Test :

Il existe un sous-ensemble qui peut être séparé donc on retourne à 3).



- **Interprétation graphique de la séparation à partir de(P_1) :**



3)Séparation :

x_2 n'est pas entier d'où le modèle linéaire courant ici (P_2)est divisé en deux sous-problèmes sous la forme :

$$\begin{cases} (P_4) = (P_2) + contrainte\ x_2 \leq [x_2]. \\ (P_5) = (P_2) + contrainte\ x_2 \geq [x_2] + 1. \end{cases}$$

avec $[x_2]$ la partie entière de x_2.

4)Résolution des sous-problèmes :

a)On résout (P_4) tq :
$(P_4) = (P_2) + contrainte\ x_2 \leq [x_2]$ c-à-d :
$(P_2) + contrainte\ x_2 \leq 3 \Longrightarrow x_2 + x_7 = 3......\bigstar$.
On rajoute au dernier tableau de(P_2) la contrainte$\bigstar$ et on applique le dual simplexe :

		c_i	1	2	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	x_4	45/14	0	0	0	1	-1/14	11/7	0
2	x_2	53/14	0	1	0	0	1/14	3/7	0
1	x_1	3	1	0	0	0	0	1	0
0	x_3	19/14	0	0	1	0	3/14	-19/7	0
0	x_7	3	0	**1**	0	0	0	0	1

•x_2 est une variable de base on multiplie la $2^{éme}$ligne par (-1),on l'additionne à la $5^{éme}$ ligne.On obtient le tableau suivant :

		c_i	1	2	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	x_4	45/14	0	0	0	1	-1/14	11/7	0
2	x_2	53/14	0	1	0	0	1/14	3/7	0
1	x_1	3	1	0	0	0	0	1	0
0	x_3	19/14	0	0	1	0	3/14	-19/7	0
0	x_7	-11/14	0	0	0	0	**-1/14**	-3/7	1
		E_j	0	0	0	0	1/7	13/7	0

•x_7 sort de la base.
•x_5 entre dans la base .

		c_i	1	2	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	x_4	4	0	0	0	1	0	2	-1
2	x_2	3	0	1	0	0	0	0	1
1	x_1	3	1	0	0	0	0	1	0
0	x_3	-1	0	0	1	0	0	**-4**	3
0	x_5	11	0	0	0	0	1	6	-14
		E_j	0	0	0	0	0	1	2

•x_3 sort de la base.
•x_6 entre dans la base .

		c_i	1	2	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	x_4	7/2	0	0	1/2	1	0	0	1/2
2	x_2	3	0	1	0	0	0	0	1
1	x_1	11/4	1	0	1/4	0	0	0	3/4
0	x_6	1/4	0	0	-1/4	0	0	1	-3/4
0	x_5	19/2	0	0	3/2	0	1	0	-19/2
		E_j	0	0	1/4	0	0	0	11/4

• tous les $b_j \geq 0$.La solution de (P_4) est :

$$\begin{bmatrix} Z_4{=}8.75. \\ X_1{=}2.75. \\ X_2{=}3. \end{bmatrix}$$

b)On résout (P_5) tq : $(P_5) = (P_2)$+la contrainte $x_2 \geq [x_2]$ +1 c-à-d :
(P_2) +la contrainte $x_2 \geq 4 \Longrightarrow x_2 - x_7 = 4 \Longrightarrow -x_2 + x_7 = -4 ......\bigstar$.
on rajoute au dernier tableau de(P_2) la contrainte$\bigstar$ et on applique le dual simplexe :

		c_i	1	2	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	x_4	45/14	0	0	0	1	-1/14	11/7	0
2	x_2	53/14	0	1	0	0	1/14	3/7	0
1	x_1	3	1	0	0	0	0	1	0
0	x_3	19/14	0	0	1	0	3/14	-19/7	0
0	x_7	-4	0	**-1**	0	0	0	0	1

•x_2 est une variable de base on additionne la $2^{éme}$ ligne à la $5^{éme}$ ligne.On obtient le tableau suivant :

		c_i	1	2	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	x_4	45/14	0	0	0	1	-1/14	11/7	0
2	x_2	53/14	0	1	0	0	1/14	3/7	0
1	x_1	3	1	0	0	0	0	1	0
0	x_3	19/14	0	0	1	0	3/14	-19/7	0
0	x_7	-3/14	0	0	0	0	1/14	3/7	1
		E_j	0	0	0	0	1/7	13/7	0

• Pas de pivot négatif.(P_5) n'admet pas de solution .

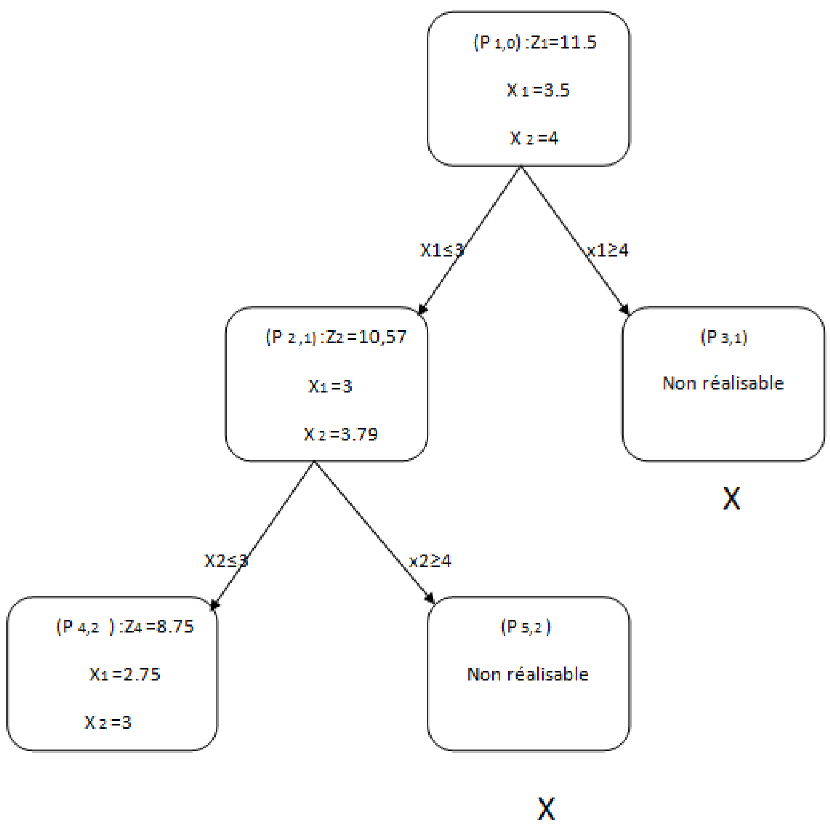
5)Évaluation :

•(P_4) a une solution non entière et la valeur de $Z_4 \leq Z_1$ donc on peut le séparer .

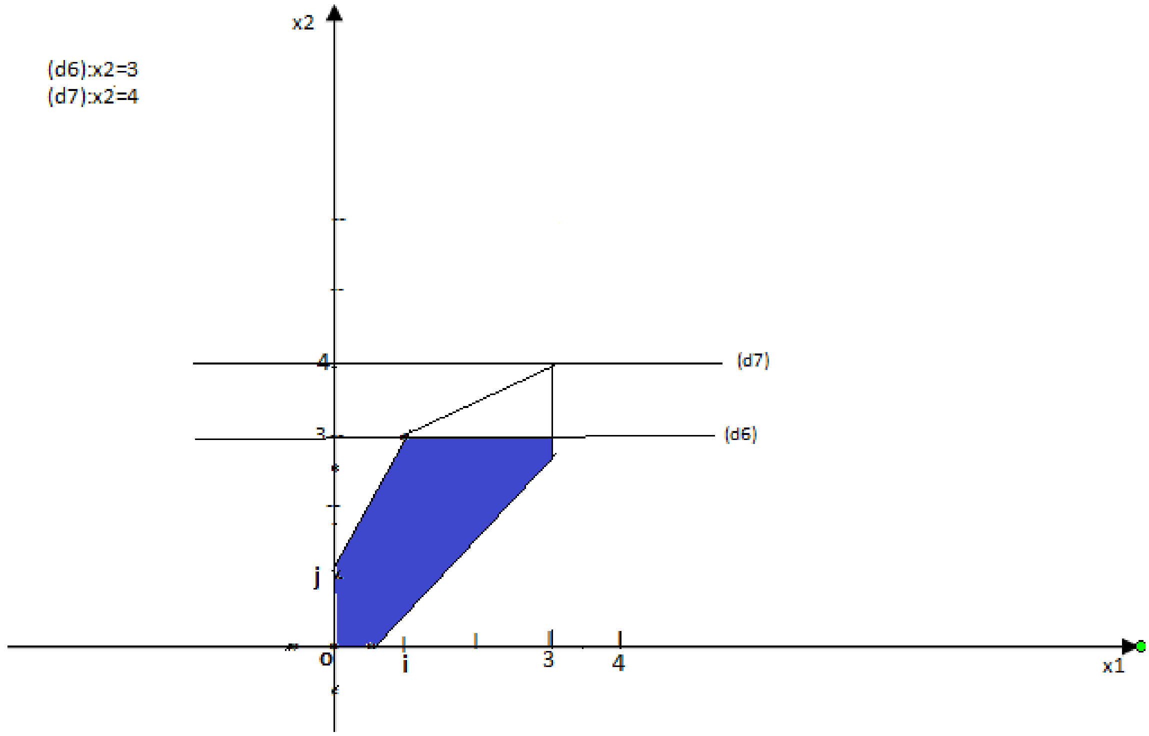
•(P_5) non réalisable donc on peut le tailler .

6)Test :

Il existe un sous-ensemble qui peut être séparé donc retour à 3).



• **Interprétation graphique de la séparation à partir de (P_2) :**



3)Séparation :
x_1 n'est pas entier d'où le modèle linéaire courant ici (P_4) est divisé en deux sous-problèmes sous la forme :

$$\begin{cases} (P_6) = (P_4) + contrainte\ x_1 \leq [x_1]. \\ (P_7) = (P_4) + contrainte\ x_1 \geq [x_1] + 1. \end{cases}$$

avec $[x_1]$ la partie entière de x_1.

4)Résolution des sous-problèmes :

a)On résout (P_6) tq :
$(P_6) = (P_4) + contrainte\ x_1 \leq [x_1]$ c-à-d :
(P_4) +la contrainte $x_1 \leq 2 \Longrightarrow x_1 + x_8 = 2......\bigstar$.
On rajoute au dernier tableau de(P_4) la contrainte$\bigstar$ et on applique le dual simplexe :

		c_i	1	2	0	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
0	x_4	7/2	0	0	1/2	1	0	0	1/2	0
2	x_2	3	0	1	0	0	0	0	1	0
1	x_1	11/4	1	0	1/4	0	0	0	3/4	0
0	x_6	1/4	0	0	-1/4	0	0	1	-3/4	0
0	x_5	19/2	0	0	3/2	0	1	0	-19/2	0
0	x_8	2	**1**	0	0	0	0	0	0	1

•x_1 est une variable de base on multiplie la $3^{éme}$ ligne par (-1) et on l'additionne à la $6^{éme}$ ligne.On obtient le tableau suivant :

		c_i	1	2	0	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
0	x_4	7/2	0	0	1/2	1	0	0	1/2	0
2	x_2	3	0	1	0	0	0	0	1	0
1	x_1	11/4	1	0	1/4	0	0	0	3/4	0
0	x_6	1/4	0	0	-1/4	0	0	1	-3/4	0
0	x_5	19/2	0	0	3/2	0	1	0	-19/2	0
0	x_8	-3/4	0	0	**-1/4**	0	0	0	-3/4	1
		E_j	0	0	1/4	0	0	0	11/4	0

•x_8 sort de la base.
•x_3 entre dans la base.

		c_i	1	2	0	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
0	x_4	2	0	0	0	1	0	0	-1	2
2	x_2	3	0	1	0	0	0	0	1	0
1	x_1	2	1	0	0	0	0	0	0	1
0	x_6	1	0	0	0	0	0	1	0	-1
0	x_5	5	0	0	0	0	1	0	14	6
0	x_3	3	0	0	1	0	0	0	3	-4
		E_j	0	0	0	0	0	0	2	1

• Tous les $b_j \geq 0$.La solution de (P_6) est :

$$\begin{bmatrix} Z_6{=}8. \\ X_1{=}2. \\ X_2{=}3. \end{bmatrix}$$

b)On résout (P_7) telle que :
$(P_7) = (P_4)$+ la contrainte $x_1 \geq [x_1]$ +1 c-à-d :
(P_4)+ la contrainte $x_1 \geq 3 \Longrightarrow x_1 - x_8 = 3 \Longrightarrow -x_1 + x_8 = -3$......★.
On rajoute au dernier tableau de(P_4) la contrainte ★ et on applique le dual simplexe :

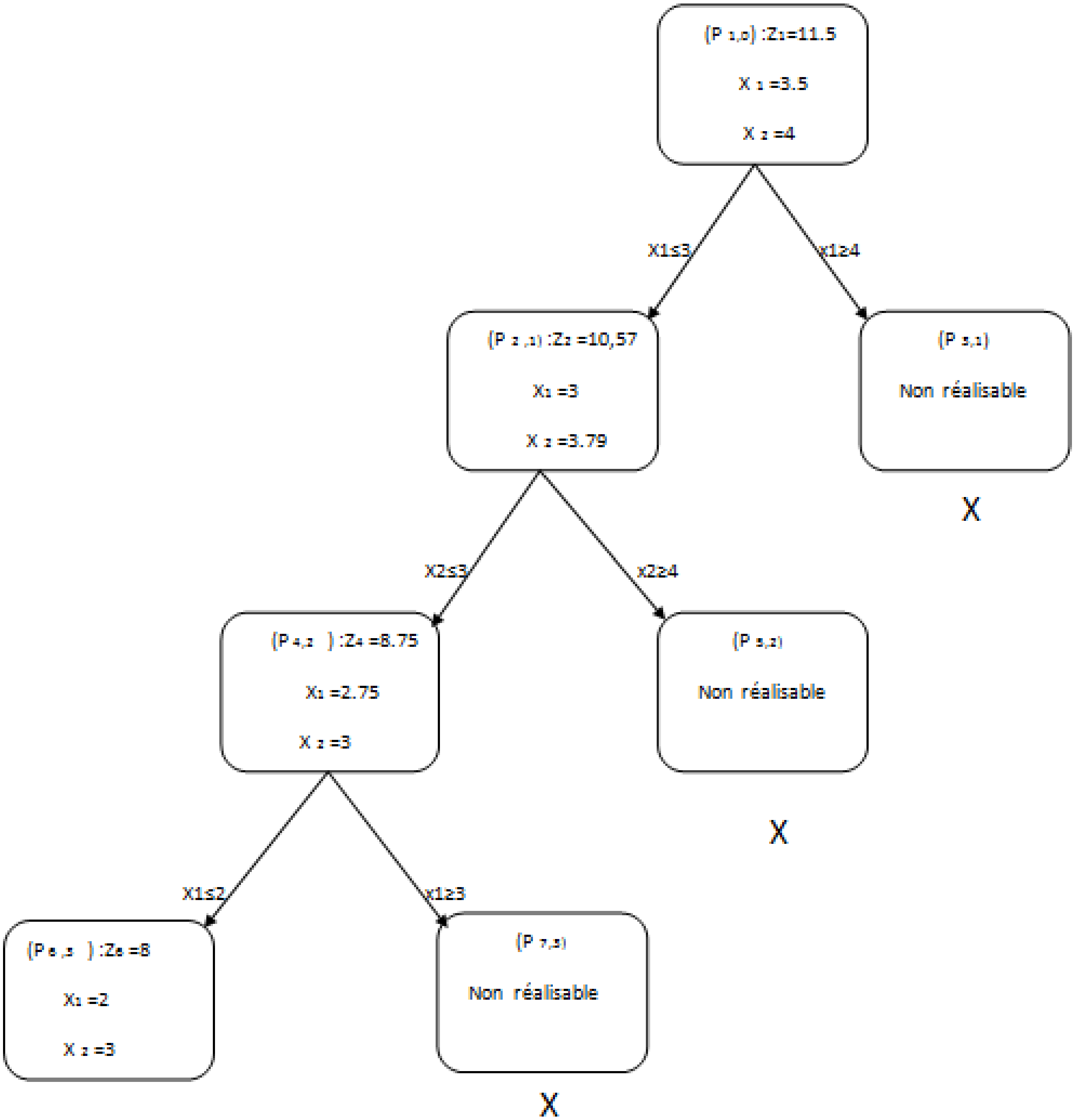
		c_i	1	2	0	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
0	x_4	7/2	0	0	1/2	1	0	0	1/2	0
2	x_2	3	0	1	0	0	0	0	1	0
1	x_1	11/4	1	0	1/4	0	0	0	3/4	0
0	x_6	1/4	0	0	-1/4	0	0	1	-3/4	0
0	x_5	19/2	0	0	3/2	0	1	0	-19/2	0
0	x_8	-3	**-1**	0	0	0	0	0	0	1

•x_1 est une variable de base d'où on additionne la $3^{éme}$ligne à la $6^{éme}$ ligne.On obtient le tableau suivant :

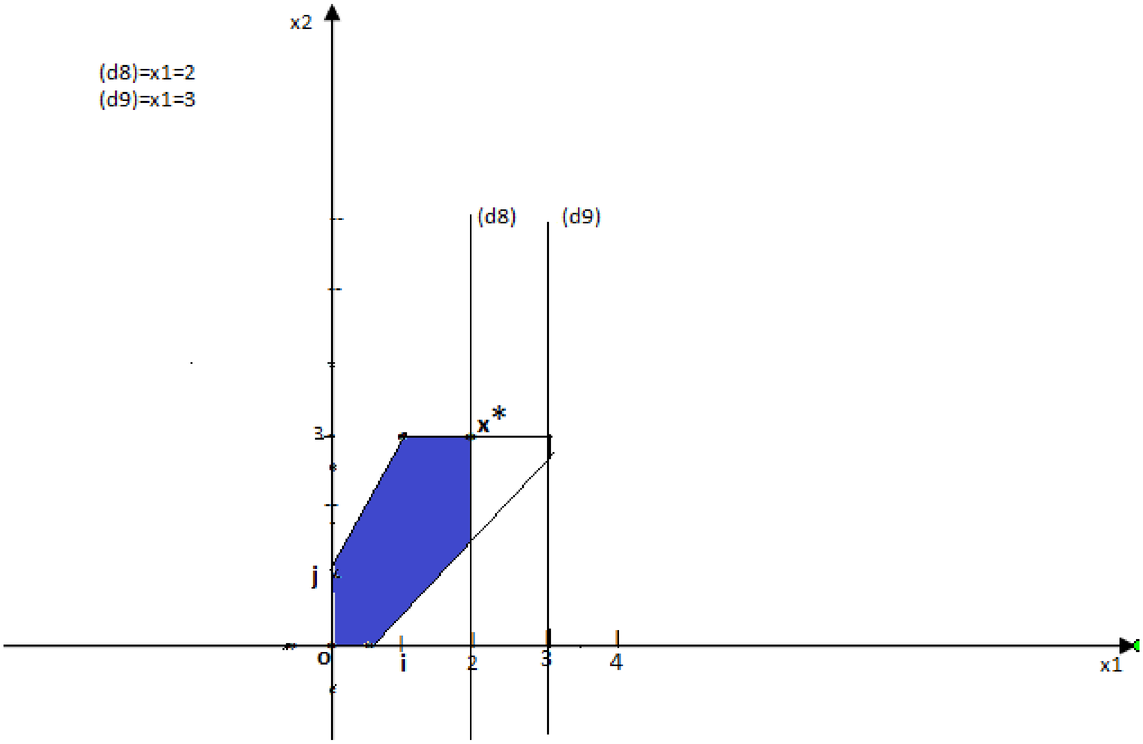
		c_i	1	2	0	0	0	0	0	0
c_b	base	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
0	x_4	7/2	0	0	1/2	1	0	0	1/2	0
2	x_2	3	0	1	0	0	0	0	1	0
1	x_1	11/4	1	0	1/4	0	0	0	3/4	0
0	x_6	1/4	0	0	-1/4	0	0	1	-3/4	0
0	x_5	19/2	0	0	3/2	0	1	0	-19/2	0
0	x_8	-1/4	0	0	1/4	0	0	0	3/4	1
		E_j	0	0	1/4	0	0	0	11/4	0

• Pas de pivot négatif .(P_7) non-réalisable.

5)évaluation :
• On a (P_6) a une solution entière et son $Z_6 \leq Z_1$ donc inutile de le séparer.
• On a (P_7) non réalisable donc on peut le tailler.
6)Test :
Il n'y a plus de sous-ensembles à séparer alors on compare tous les Z des solutions entières et on prend la plus grande car on a un problème de maximisation .Ici on a $Z_6 = 8$ est la seule qui a une solution entière .
D'où la solution de notre problème est :
X^*=(2,3)et son $Z^* = 8$.



• **Interprétation graphique de la séparation à partir de** (P_4) **:**



Remarque :
On a utilisé la stratégie :
Le meilleur d'abord.

- **En utilisant Matlab 2014 :**

Le problème correspondant est le suivant :
f=[-1 ;-2 ;0 ;0 ;0] ;
Aeq=[4 -3 1 0 0 ; -2 1 0 1 0 ; -6 14 0 0 1] ;
beq=[2 ;1 ;35] ;
ub=[Inf ;Inf ;Inf ;Inf ;Inf] ;
lb=[0 ;0 ;0 ;0 ;0] ;
intcon=[1,2] ;
x=intlinprog(f,intcon,[ ],[ ],Aeq,beq,ub,lb) ;
Le résultat obtenu par ce programme est :

```
>> f= [-1;-2;0;0;0];
Aeq=[ 4 -3 1 0 0; -2 1 0 1 0  ; -6 14 0 0 1];
beq=[2;1;35];
ub=[Inf;Inf;Inf;Inf;Inf];
lb=[0;0;0;0;0];
intcon=[1,2];
x=intlinprog(f,intcon,[],[],Aeq,beq,lb,ub)
LP:                 Optimal objective value is -11.500000.

Cut Generation:     Applied 1 gomory cut.
                    Lower bound is -10.571429.


Optimal solution found.

Intlinprog stopped at the root node because the objective value is within a gap tolerance of the optimal value;
options.TolGapAbs = 0 (the default value). The intcon variables are integer within tolerance, options.TolInteger = 1e-05
(the default value).


x =

    2.0000
    3.0000
    3.0000
    2.0000
    5.0000
```

- **En utilisant(LP Solve IDE) :**

Le problème correspondant est le suivant :

/* objective function */

max :x_1+2x_2 ;

/* Variable bounds */

$4x_1 - 3_2 + x_3 = 2$;

-2$x_1 + x_2 + x_4$=1 ;

-6$x_1 + 14x_2 + x_5 = 35$;

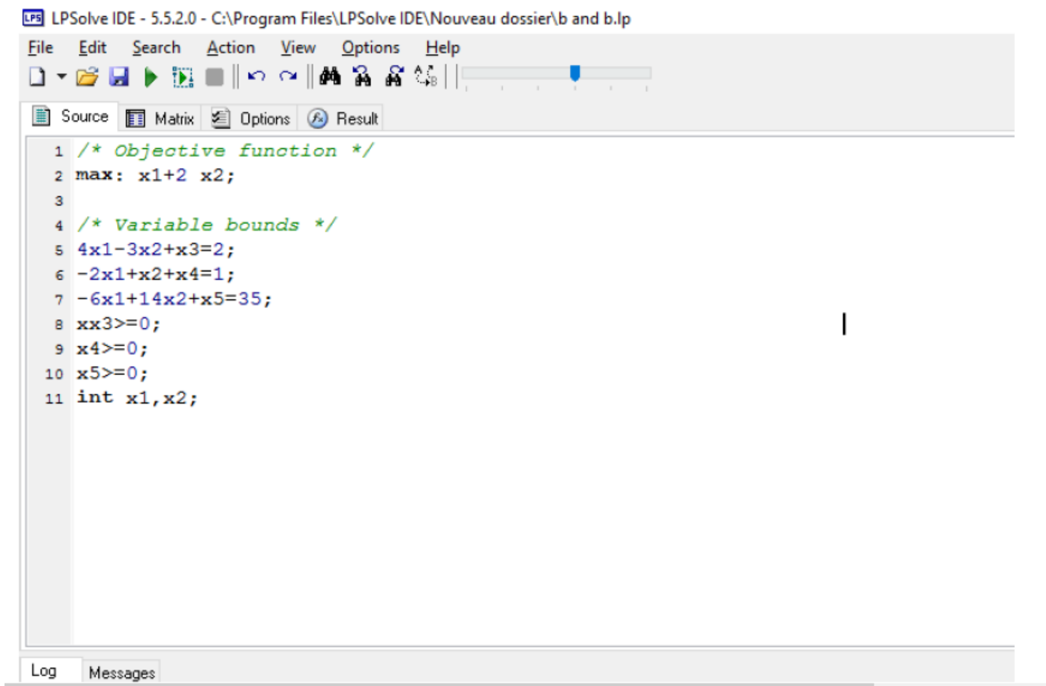
$x_3 >= 0$;

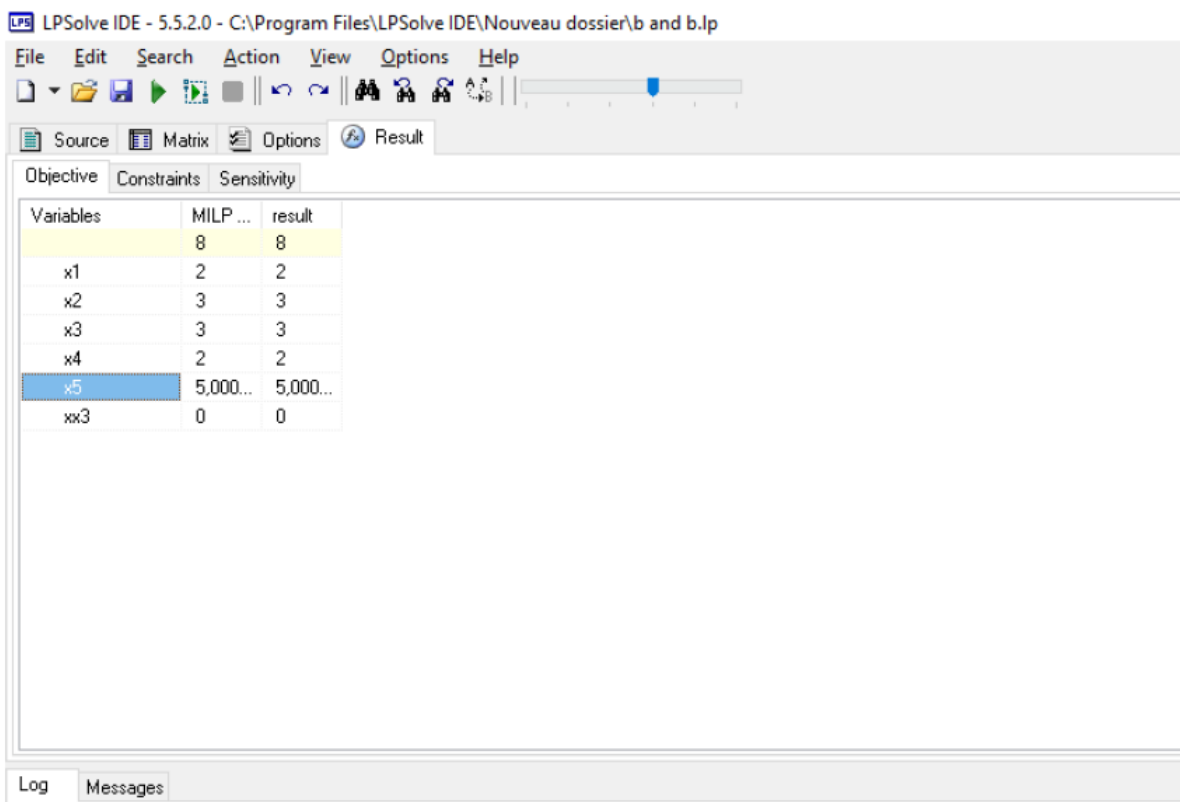
$x_4 >= 0$;

$x_5 >= 0$;

int x_1,x_2 ;

Le résultat obtenu par ce programme est :





3.5 Conclusion

Dans ce chapitre nous nous sommes intéressés à la résolution des programmes linéaires en nombres entiers avec la méthode de Séparation et d'Évaluation "Branch and Bound "qui se repose sur le principe d'énumération.
Son efficacité est liée directement au principe d'élagage qui permet de diminuer l'espace de

recherche des solutions.

$P_6 : Z_6=8$
$X_1=2$ $X_2=3$

Conclusion générale

Quelques problèmes de la vie courante peuvent être modélisés comme des problèmes combinatoires .
Dans notre travail,on a présenté un état de l'art des méthodes de résolutions des problèmes de programmation linéaire en nombres entiers .
On a défini les concepts de base de la programmation linéaire ensuite on a présenté la méthode Branch and Bound en appliquant un exemple numérique.
Nous espérons que ce travail sera utile pour tous ceux qui sont concernés par le développement des systèmes de programmation linéaire, de modélisation et d'optimisation.

Bibliographie

[1] Aidene M. Oukacha B., « les manuels de l'étudiant Recherche Opérationnelle, Programmation Linéaire ». Copyright Eurl Pages Bleues internationales, Maison d'Edition pour l'enseignement et la formation, 2005.

[2] Amira Gherboudj ,thèse pour l'obtention du diplôme de Doctorat 3éme cycle LMD Spécialité informatique,2013.

[3] Dantzig ,Linear programming and extensions, Princeton University Press, Princeton, 1963.

[4] Jaques Teghem,RECHERCHE OPÉRATIONNELLE TOME 1,Septembre 2012.

[5] Land A.H., Doig A.G., An automatic method of solving discrete programming problems , Econometrica 28 (3), 497-520, 1960.

[6] Lemke C.E., « The dual method for solving the linear programming problem », Naval Research Logistic Quarterly 1, 36-47, 1954.

[7] P.Fouilhoux,Programmation mathématique discrète et modèle linéaire ,Université Pierre et Marie Cunie ,2012.

[8] Yves Nobert , Roch Ouellet,Régis parent,La recherche opérationnelle3^{e} édition,2001.