

الجمهورية الجزائرية الديمقراطية الشعبية
RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
جامعة عمار ثليجي الأغواط
UNIVERSITÉ AMAR THELIDJI LAGHOUAT



كلية العلوم
FACULTÉ DES SCIENCES
قسم الرياضيات
DÉPARTEMENT DE MATHÉMATIQUES

Thèse de Doctorat troisième cycle

Filière : Mathématiques
Spécialité : Analyse Mathématique
Option: Analyse Numérique et EDP

Présentée par : Djeloud Khalil

Intitulée:

Une nouvelle direction pour l'optimisation linéaire et application au contrôle optimal

Soutenue publiquement le : Le 08/04/2021, devant le jury composé de :

<i>Nom et Prénom</i>	<i>Grade</i>	<i>Etabl.</i>	<i>Qualité</i>
<i>Ouinten Youcef</i>	<i>Professeur</i>	<i>UAT Laghouat</i>	<i>Président</i>
<i>Bentobache Mohand</i>	<i>Professeur</i>	<i>UAT Laghouat</i>	<i>Directeur de thèse</i>
<i>Bibi Mohand Ouamer</i>	<i>Professeur</i>	<i>UAM Béjaia</i>	<i>Co-directeur de thèse</i>
<i>Mokhtari Abdelkader</i>	<i>Professeur</i>	<i>UAT Laghouat</i>	<i>Examineur</i>
<i>Abbas Karim</i>	<i>Professeur</i>	<i>UAM Béjaia</i>	<i>Examineur</i>
<i>Brahmi Belkacem</i>	<i>MCA</i>	<i>UAM Béjaia</i>	<i>Examineur</i>

Année Universitaire: 2020/2021

Au nom d'Allah, le Tout Miséricordieux, le Très Miséricordieux.

Louange à ALLAH, Seigneur des mondes. Que la paix et le salut d'Allah soient sur le Prophète Muhammad, sur sa famille, sur ses compagnons ainsi que sur tous ceux qui les suivent dans la bonne voie jusqu'au jour dernier.

Tout d'abord, je remercie DIEU, le tout puissant, pour m'avoir accordé santé, courage et patience afin d'accomplir ce travail.

Je tiens à remercier vivement le Professeur Mohand Bentobache qui a dirigé cette thèse. Il m'a proposé un sujet très intéressant qui m'a donné l'opportunité de toucher à de nombreux domaines. Il a su me laisser une grande liberté tout en restant exigeant et en recentrant mon travail lorsqu'il le fallait. En ce qui concerne la mise au point de ce travail, je remercie le Professeur Mohand Ouamer Bibi mon co-directeur de thèse pour ses multiples relectures et ses conseils précieux.

J'adresse mes remerciements au Professeur Youcef Ouinten de l'université de Laghouat, pour l'honneur qu'il m'a fait en acceptant de présider le jury.

Je remercie vivement le Professeur Abdelkader Mokhtari de l'université de Laghouat, le Professeur Karim Abbas et le Docteur Belkacem Brahmi de l'université de Béjaia pour avoir accepté de juger ce travail.

Dédicaces

Je dédie ce modeste travail

♡ *À mes très chers parents,*

♡ *À mes chers frères et soeurs.*

Khalil DJELOUD.

Table des matières

Table des matières	3
Liste des tableaux	6
Liste des figures	7
Notations	8
Introduction	10
1 Préliminaires mathématiques	13
1.1 Espace vectoriel	13
1.2 Matrices	15
1.2.1 Opérations sur les matrices	16
1.2.2 Matrices et opérations élémentaires	16
1.2.3 Inverse d'une matrice	17
1.2.4 Méthode de Gauss-Jordan pour inverser les matrices	17
1.2.5 Matrices complètement régulières	17
1.2.6 Rang d'une matrice	17
1.3 Systèmes d'équations linéaires	18
1.4 Factorisation LU	19
1.5 Mise à jour de l'inverse d'une matrice	23
1.6 Espace vectoriel normé	25
1.6.1 Normes fondamentales sur \mathbb{R}^n	25
1.6.2 Définitions	25
1.7 Analyse convexe	26
1.8 Programmation mathématique	27
1.8.1 Le problème fondamental	27
2 La méthode primale du simplexe	28
2.1 Position du problème	28
2.2 Définitions	28

2.3	Recherche d'une solution optimale	30
2.4	Méthode primale du simplexe	30
2.5	Résolution des problèmes de PL sous forme générale	32
2.6	Exemples numériques	35
3	Une Nouvelle Méthode à Direction Hybride et à Pas Multiple (MDHPM) pour la PL	41
3.1	Position du problème et définitions	41
3.2	Formule d'accroissement de la fonction objectif	42
3.3	Critère d'optimalité et de suboptimalité	43
3.4	Rappel sur la Méthode Adaptée à Direction Hybride (MADH)	45
3.4.1	Exemples numériques	48
3.5	Une itération du nouvel algorithme MDHPM	52
3.6	Changement de la solution réalisable	54
3.7	Changement de support	57
3.7.1	Règle du pas simple	57
3.7.2	Règle du pas multiple	61
3.8	Le schéma de la méthode à direction hybride et à pas multiple	62
3.9	Exemple numérique	63
4	Nouvelle technique d'initialisation des algorithmes de PL	68
4.1	Position du problème	68
4.2	Construction d'une SRS initiale	70
4.2.1	Recherche d'un support initial	70
4.2.2	Recherche d'une solution réalisable initiale	71
4.3	Exemples numériques	73
5	Résultats expérimentaux et application au contrôle optimal	86
5.1	Problème de contrôle optimal linéaire	86
5.1.1	Présentation du problème	87
5.1.2	Discrétisation du problème initial	88
5.1.3	Exemple numérique	89
5.1.4	Planification de la production : un problème de contrôle optimal	91
5.2	Résultats expérimentaux	93
5.2.1	Problèmes de PL générés aléatoirement	94
5.2.2	Problèmes de PL particuliers	94
5.2.3	Résultats obtenus sur les problèmes-test	95
	Conclusion	99
	Bibliographie	100

Liste des tableaux

5.1	Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test générés aléatoirement de dimension $m \times m$	96
5.2	Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test générés aléatoirement de dimension $m \times 2m$	96
5.3	Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test de contrôle optimal	96

Table des figures

5.1	Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test générés aléatoirement de dimension $m \times m$	97
5.2	Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test générés aléatoirement de dimension $m \times 2m$	97
5.3	Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes de contrôle optimal	98
5.4	Le nombre moyen d'itérations et le temps CPU moyen des deux algorithmes MADH et MDHPM pour les problèmes-test de contrôle optimal	98

Notations

On utilise les notations usuelles suivantes :

- \mathbb{N} l'ensemble des entiers naturels.
- \mathbb{N}^* l'ensemble des entiers naturels non nuls.
- \mathbb{R} l'ensemble des réels finis.
- \mathbb{R}^+ l'ensemble des réels finis positifs.
- T la transposée.
- \exists quantificateur existentiel (il existe).
- \forall quantificateur universel (quel que soit).
- \in signe d'appartenance ; \notin signe de non appartenance.
- \mathbb{R}^n ($n \in \mathbb{N}^*$) l'ensemble des n -tuples $(x_1, x_2, \dots, x_n)^T$, avec $x_i \in \mathbb{R}$, $i = 1, \dots, n$.
- Pour $x, y \in \mathbb{R}^n$, $x \neq y$ signifie $\exists i \in \{1, \dots, n\}$ tel que $x_i \neq y_i$.
- Pour $x, y \in \mathbb{R}^n$, $x \leq y$ signifie $x_i \leq y_i$ pour $i = 1, \dots, n$. L'écriture $x \geq y$ signifie que $y \leq x$; $x < y$ signifie que $x \leq y$ et $x \neq y$.
- Si A est une partie de B , on écrit $A \subseteq B$. Si A est une partie propre (c'est-à-dire distincte de B), on écrit $A \subset B$ (\subset inclusion stricte).
- a un élément de $A \subset \mathbb{R}^n$: $\max A = a$, c'est-à-dire, $x \leq a$, $\forall x \in A$; $\min A = a$, c'est-à-dire, $a \leq x$, $\forall x \in A$.
- $\int_a^b f(t)dt$ est l'intégrale de f au sens de Riemann sur $[a, b]$.
- $\dot{x}(t) = \frac{dx(t)}{dt}$ dérivée de $x(t)$ par rapport à t .
- $x \in \mathbb{R}$, $sign(x) = \begin{cases} 1 & \text{si } x > 0; \\ -1 & \text{si } x < 0; \\ 0 & \text{sinon.} \end{cases}$
- $x \in \mathbb{R}$, $|x| = x sign(x)$ désigne la valeur absolue de x .
- \emptyset l'ensemble vide (c'est-à-dire un ensemble ne renfermant aucun élément).
- $0_{\mathbb{R}^n}$ le vecteur nul à n composantes.

- Soit E un ensemble. On note par $Card(E)$ le nombre d'éléments de E .
- \cap intersection ; \cup réunion ; \setminus différence d'ensembles.
- \Leftrightarrow signe d'équivalence ; \Rightarrow signe d'implication.
- $a := b$ remplacer a par b .
- $a \rightarrow +\infty$ c-à-d a tend vers $+\infty$.
- $n!$ le factoriel d'un entier n .

Introduction

La programmation linéaire (PL) est une technique mathématique permettant de déterminer la meilleure solution d'un problème dont les données et les inconnues satisfont à une série d'équations et d'inéquations linéaires. De nombreux phénomènes économiques et industriels peuvent se modéliser par des systèmes mathématiques d'inégalités et d'égalités linéaires conduisant à des problèmes d'optimisation linéaire. Dans ces problèmes d'optimisation linéaire, on cherche à minimiser ou maximiser une fonction linéaire sous des contraintes linéaires portant sur les variables du problème. On parle souvent de problème de programmation linéaire (ou encore de programme linéaire).

En 1939, le célèbre mathématicien russe Léonid Kantorovitch (1912-1986), prix Nobel 1975, formulait dans toute leur amplitude les problèmes de programmation linéaire [44] et fournissait une méthode de résolution, adaptée au système économique soviétique et apparentée à celle des multiplicateurs de Lagrange. Le terme de programmation faisant référence à l'idée d'organisation et de planification lié à la nature des phénomènes modélisés. Ce terme a été introduit pendant la seconde guerre mondiale et systématiquement utilisé à partir de 1947 lorsque G.B. Dantzig inventa la méthode du simplexe pour résoudre les problèmes de programmation linéaire.

La programmation linéaire connaît un développement rapide par suite de son application directe à la gestion rationnelle des entreprises. Le facteur expliquant l'essor de la PL est la construction d'ordinateurs puissants qui ont permis de traiter les problèmes concrets de taille très grande. On l'applique surtout en gestion et en économie appliquée. On peut citer les domaines d'application de la programmation linéaire qui sont : les transports, les banques, les industries lourdes et légères, l'agriculture, les chaînes commerciales, la sidérurgie, et surtout le domaine des applications militaires.

C'est donc sur un riche terreau que des spécialistes bien connus : G.B. Dantzig (1914-2005), C.E. Lemke (1920-2004), A. Charnes (1917-1992), R. Frisch (1895-1973), E.M.L. Beale (1928-1985), H.W. Kuhn (1925-2014) et A.W. Tucker (1905-1995), ont fait fleurir de 1948 à 1958, les méthodes classiques de résolution des programmes linéaires. L'apport de G.B. Dantzig, après sa mise au point, sur une indication de Janos von Neumann, de l'algorithme du simplexe en 1947 [22] et la méthode révisée du simplexe en 1953, demeure considérable. De son côté, C.E. Lemke, dans son article, exposait, dès 1954, la méthode duale du simplexe [45], étudiée aussi, avec quelques variantes par E.M.L. Beale [5]. A.

Charnes, en 1952 [20], indiquait le procédé du petit ϵ pour mettre fin aux difficultés provoquées par l'apparition de certaines solutions dégénérées au cours d'un calcul itératif et, la même année, A. Orden donnait la méthode du grand M ou de la base artificielle. En 1955, G.B. Dantzig a développé en collaboration avec A. Orden et P. Wolfe la méthode primale-duale du simplexe [23] et en 1959, G.F. Hadley et M.A. Simonnard ont développé la méthode des deux phases [38].

Dans [32], R. Gabasov et F. M. Kirillova généralisent la méthode du simplexe en développant la méthode de support qui peut être initialisée par n'importe quel point réalisable et n'importe quel support (ensemble d'indices de base). Cette méthode utilise un critère de suboptimalité qui arrête l'algorithme lorsqu'une solution ϵ -optimale est obtenue. Plus tard, ils ont développé la méthode adaptée [35, 32, 36] basée sur une direction d'amélioration qui dépend de la solution courante. La méthode adaptée est appliquée pour résoudre les problèmes de contrôle optimal et généralisée pour résoudre les problèmes quadratiques convexes [33, 34, 12, 13, 18, 40, 42, 43].

En 1984, N. Karmarkar [39] a développé le premier algorithme de points intérieurs compétitif avec l'algorithme du simplexe pour la résolution des problèmes de grande taille. Il existe aussi d'autres méthodes pour la résolution des programmes linéaires, on cite par exemple : la méthode généralisée du simplexe de Cardoso et Clímaco (1992, [19]) ; la méthode du simplexe avec des bases déficientes de Pan (1998, [50]) ; la méthode de points extérieurs de Paparrizos et al. (2003, [51]), etc.

Dans [14, 7, 9, 8], les auteurs ont développé une nouvelle méthode appelée "Méthode Adaptée à Direction Hybride (MADH)" pour résoudre les problèmes de programmation linéaire avec des variables bornées. Cette méthode utilise le concept de direction hybride afin de passer d'une solution réalisable à une autre ayant une meilleure valeur de la fonction objectif. Plus tard, cet algorithme est généralisé pour résoudre les programmes quadratiques convexes [15] et les problèmes de contrôle optimal [55]. Le critère d'arrêt de l'algorithme MADH est basé sur une quantité appelée estimation d'optimalité, donc il ne donne que des solutions optimales ; de plus, la mise à jour du support est effectuée à l'aide de la règle du pas simple.

Dans ce travail, nous proposons une nouvelle direction hybride qui donne une meilleure amélioration locale pour la fonction objectif que l'algorithme MADH développé précédemment dans [14]. Cet algorithme utilise un critère de suboptimalité afin d'obtenir une solution suboptimale et il utilise la règle du pas multiple afin de changer le support courant [28]. Par conséquent, la formule de mise à jour de l'estimation de suboptimalité [32, 36] est un cas particulier de notre formule. Afin de comparer notre algorithme à l'algorithme MADH et à l'algorithme primal du simplexe implémenté dans le solveur de programmation linéaire open source GLPK [47], nous avons développé une implémentation avec le langage de programmation C++. Ensuite, nous avons présenté quelques résultats numériques sur des problèmes-test générés aléatoirement et sur quelques instances d'un problème de contrôle

optimal linéaire modélisant un problème de planification de la production [4]. Les résultats numériques obtenus montrent bien l'efficacité de notre algorithme.

Dans le but de résoudre des problèmes de programmation linéaire écrits sous forme générale, tels que les problèmes pratiques de la librairie NETLIB [37]¹, et en se basant sur les techniques d'initialisation proposées dans [7, 6], nous proposons une nouvelle technique d'initialisation. Cette technique consiste à résoudre avec notre algorithme un problème auxiliaire ayant le même nombre de variables que celui du problème original et une solution réalisable de support initiale évidente ; la solution optimale du problème auxiliaire est alors une solution réalisable initiale du problème original. Enfin, cette approche d'initialisation est illustrée par plusieurs exemples numériques.

Cette thèse est organisée comme suit : Dans le premier chapitre, nous avons donné la plupart des concepts dont nous avons besoin pour comprendre le contenu des autres chapitres. Dans le deuxième chapitre, nous présentons la méthode primale du simplexe. La nouvelle méthode proposée dite 'Méthode à Direction Hybride et à Pas Multiple' (MDHPM) fera l'objet du troisième chapitre. Le quatrième chapitre est consacré à la présentation d'une nouvelle technique d'initialisation des différents algorithmes pour résoudre les programmes linéaires sous forme générale. Enfin, dans le dernier chapitre, nous présentons certains résultats numériques qui comparent notre algorithme à l'algorithme primal du simplexe du solveur open source GLPK [47] et à l'algorithme MADH [14].

1. NETLIB est une librairie qui contient une collection de problèmes-test modélisant une variété de problèmes pratiques qui surviennent dans plusieurs domaines, à savoir : le raffinage du pétrole, l'ordonnancement, la gestion de production, l'allocation optimale des ressources, le traitement d'images, le problème des moindres carrés, etc.

Chapitre 1

Préliminaires mathématiques

Introduction

Ce chapitre présente les concepts mathématiques essentiels qui sont nécessaires pour comprendre le matériel présenté dans les chapitres suivants. Le traitement des sujets est concis et limité à la présentation des aspects clés du sujet. L'objectif de ce chapitre est de comprendre les notions mathématiques nécessaires pour formuler et résoudre des problèmes de programmation linéaire.

1.1 Espace vectoriel

Tout ensemble E dans lequel ont été définies une addition entre les éléments de cet ensemble et une multiplication par un scalaire réel est un espace vectoriel si les conditions suivantes sont satisfaites :

1. E forme un groupe commutatif par rapport à l'addition ;
2. $\alpha x + \beta y \in E, \forall x, y \in E, \forall \alpha, \beta \in \mathbb{R}$;
3. $\alpha(x + y) = \alpha x + \alpha y = (x + y)\alpha$.

L'ensemble des nombres réels \mathbb{R} forme un espace vectoriel pour l'addition et la multiplication usuelles. L'espace vectoriel le plus fréquemment utilisé par la suite est l'ensemble \mathbb{R}^n .

L'addition et la multiplication par un scalaire α sont définies respectivement par :

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} + \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_n + y_n \end{pmatrix},$$

$$\alpha \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \alpha \cdot x_1 \\ \alpha \cdot x_2 \\ \vdots \\ \alpha \cdot x_n \end{pmatrix}.$$

Certains vecteurs jouent un rôle particulier, comme par exemple les vecteurs unitaires dont la $i^{\text{ème}}$ composante vaut 1 et dont toutes les autres sont nulles; ces vecteurs sont parfois appelés vecteurs élémentaires et sont notés e_i .

Définition 1.1. Les vecteurs $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$ à n composantes sont dits linéairement dépendants si et seulement s'il existe des scalaires c_1, c_2, \dots, c_m , non tous nuls, tels que

$$\sum_{i=1}^m c_i \mathbf{x}_i = \mathbf{0}_{\mathbb{R}^n}.$$

De manière équivalente, on définit l'indépendance linéaire de m vecteurs ainsi :

Les vecteurs $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$ sont dits linéairement indépendants si et seulement si :

$$\sum_{i=1}^m c_i \mathbf{x}_i = \mathbf{0}_{\mathbb{R}^n} \Rightarrow c_i = 0, \forall i = 1, \dots, m.$$

Définition 1.2. On dit qu'un vecteur \mathbf{y} est une combinaison linéaire des vecteurs $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$ s'il existe des scalaires c_1, c_2, \dots, c_m tels que :

$$\mathbf{y} = \sum_{i=1}^m c_i \mathbf{x}_i.$$

Mentionnons, pour conclure, les différentes propriétés de la dépendance linéaire de vecteurs.

- Si les vecteurs $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$ sont linéairement dépendants, alors au moins l'un d'entre eux s'écrit comme une combinaison linéaire des autres.
- Soient m vecteurs $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$ à n composantes. Si $m > n$, alors $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$ sont linéairement dépendants.

Définition 1.3. La base d'un espace vectoriel V est un système de vecteurs de V , noté B , tel que chaque vecteur de V peut être représenté de façon unique comme combinaison linéaire des vecteurs de B .

Il est souvent plus pratique d'utiliser une définition équivalente basée sur la notion d'indépendance linéaire. La base d'un espace vectoriel V est un système B de vecteurs linéairement indépendants de V , tel que chaque vecteur de V dépend linéairement des vecteurs de B .

Si B est formée par un nombre fini de vecteurs, alors on dit que B est une base finie. Un espace vectoriel ayant une base finie est dit de dimension finie. Dans ce cas, on obtient le résultat 1.1.

Résultat 1.1. Si V est un espace vectoriel de dimension finie, alors deux bases quelconques de V possèdent le même nombre de vecteurs. Ce nombre est appelé dimension de V .

Dans le cas particulier de l'espace vectoriel \mathbb{R}^n , sa dimension est n . Soit $B = \{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n\}$ une base de \mathbb{R}^n . Un vecteur \mathbf{x} quelconque de \mathbb{R}^n peut donc s'écrire :

$$\mathbf{x} = \sum_{i=1}^n \alpha_i \mathbf{a}_i.$$

Les scalaires α_i sont appelés coordonnées de \mathbf{x} , par rapport à cette base.

1.2 Matrices

Définition 1.4. Une matrice A à m lignes et n colonnes est constituée de $m \times n$ éléments réels (ou complexes). On la note :

$$A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n).$$

Les éléments a_{ij} sont appelés les coefficients de la matrice. Le premier indice est celui de la ligne et le second celui de la colonne. Les éléments a_{ii} sont appelés les coefficients diagonaux. On représente aussi A sous la forme suivante :

$$A = (a_1, a_2, a_3, \dots, a_n), \text{ où } a_j = \begin{pmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{pmatrix}, j = 1, \dots, n.$$

- L'ensemble des matrices à $m \times n$ coefficients dans \mathbb{R} se note $\mathcal{M}_{m,n}(\mathbb{R})$ et simplement $\mathcal{M}_m(\mathbb{R})$ lorsque $n = m$.
- Une matrice composée de m lignes et n colonnes est dite d'ordre $m \times n$ et simplement d'ordre m lorsque $n = m$.
- Une matrice qui a le même nombre de lignes et de colonnes $n = m$ est dite carrée.
- Une matrice à une ligne ($m = 1$) s'appelle une matrice-ligne.
- Une matrice à une colonne ($n = 1$) s'appelle une matrice-colonne.
- Une matrice $A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n)$ est diagonale si A est carrée et $a_{ij} = 0$, pour $i \neq j$.

- On appelle matrice identité notée I_m la matrice diagonale d'ordre m dont les termes diagonaux sont égaux à un.
- La matrice nulle de $\mathcal{M}_{m,n}(\mathbb{R})$, notée par $0_{m,n}$, est définie par :

$$0_{m,n} = (m_{ij}, 1 \leq i \leq m, 1 \leq j \leq n), \text{ où } m_{ij} = 0, \forall (i, j).$$

- Une matrice $A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n)$ est triangulaire inférieure si $a_{ij} = 0$, pour $i < j$.
- Une matrice $A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n)$ est triangulaire supérieure si $a_{ij} = 0$, pour $i > j$.
- La matrice A^T transposée de $A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n)$ est la matrice $A^T = (a_{ji} = a_{ij}, 1 \leq j \leq n, 1 \leq i \leq m)$.
- Une matrice A est symétrique si $A = A^T$.
- Une matrice A est antisymétrique si $A^T = -A$.

1.2.1 Opérations sur les matrices

- $\forall A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n), B = (b_{ij}, 1 \leq i \leq m, 1 \leq j \leq n) \in \mathcal{M}_{m,n}(\mathbb{R})$,

$$A + B = (a_{ij} + b_{ij}, 1 \leq i \leq m, 1 \leq j \leq n).$$

- $\forall A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n) \in \mathcal{M}_{m,n}(\mathbb{R}), \forall \lambda \in \mathbb{R}$,

$$\lambda A = (\lambda a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n).$$

- $\forall A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq n) \in \mathcal{M}_{m,n}(\mathbb{R})$ et $\forall B = (b_{ij}, 1 \leq i \leq n, 1 \leq j \leq s) \in \mathcal{M}_{n,s}(\mathbb{R})$, $A \times B \in \mathcal{M}_{m,s}(\mathbb{R})$, et

$$A \times B = \left(\sum_{k=1}^n a_{ik} b_{kj}, 1 \leq i \leq m, 1 \leq j \leq s \right).$$

- L'ensemble $\mathcal{M}_{m,n}(\mathbb{R})$ muni des opérations d'addition et de multiplication par un scalaire est un espace vectoriel qui admet pour base canonique la famille des matrices $(E_{ij}, 1 \leq i \leq m, 1 \leq j \leq n)$, où E_{ij} est une matrice dont tous les coefficients sont nuls à l'exception de celui d'indices (i, j) qui est égal à 1.

1.2.2 Matrices et opérations élémentaires

On peut définir des opérations élémentaires sur les matrices :

- Permuter deux lignes ou deux colonnes entre elles respectivement.

- Ajouter un multiple d'une ligne (resp. colonne) à une autre ligne (resp. colonne).
- Multiplier une ligne ou une colonne par un nombre différent de zéro.

Définition 1.5. Une matrice E est dite élémentaire lorsqu'elle est obtenue par des opérations élémentaires sur les lignes ou colonnes de la matrice identité.

Multiplier une matrice A par une matrice élémentaire :

- à gauche (c'est à dire calculer EA) revient à effectuer l'opération élémentaire correspondante sur les lignes de A ,
- à droite (AE) revient à faire une opération élémentaire sur les colonnes.

1.2.3 Inverse d'une matrice

Soit $A, B \in \mathcal{M}_m(\mathbb{R})$. On dit que B est l'inverse de A si et seulement si $AB = BA = I_m$. Dans ce cas, A est dite non-singulière et on note l'inverse d'une matrice A par A^{-1} . L'inverse de A est unique.

1.2.4 Méthode de Gauss-Jordan pour inverser les matrices

La méthode de Gauss-Jordan pour inverser une matrice A carrée consiste à faire des opérations élémentaires sur les lignes de la matrice A jusqu'à la transformer en la matrice identité I . On fait simultanément les mêmes opérations élémentaires sur la matrice I . On aboutit alors à une matrice qui est A^{-1} . Ceci peut se résumer par le schéma suivant :

$$\left(\begin{array}{c|c} A & I \end{array} \right) \xrightarrow{\text{Par opérations élémentaires}} \left(\begin{array}{c|c} I & A^{-1} \end{array} \right).$$

1.2.5 Matrices complètement régulières

Soit $A = (a_{ij}, 1 \leq i \leq m, 1 \leq j \leq m)$ une matrice de $\mathcal{M}_m(\mathbb{R})$. La matrice A est dite complètement régulière si les matrices $A_k = (a_{ij}, 1 \leq i \leq k, 1 \leq j \leq k)$ sont inversibles pour $k = 1, 2, \dots, m$.

1.2.6 Rang d'une matrice

Le rang d'une matrice $A \in \mathcal{M}_{m,n}(\mathbb{R})$ est égal à l'ordre de la plus grande sous-matrice carrée non-singulière et il est noté par $\text{rang}(A)$. On a ainsi $\text{rang}(A) \leq \min\{m, n\}$.

Propriétés

- Si A est une matrice carrée, alors les vecteurs correspondant aux colonnes (ou lignes) de la matrice A sont linéairement indépendants si et seulement si A est non-singulière.
- Si une matrice est d'ordre $(m \times n)$ avec $n > m$, alors les vecteurs correspondant aux colonnes sont linéairement dépendants.
- Si $\text{rang}(A) = s$, alors il y a exactement s colonnes et s lignes linéairement indépendantes.

1.3 Systèmes d'équations linéaires

Un système d'équations linéaires peut s'écrire sous la forme générale :

$$Ax = b, \quad (1.1)$$

où A est la $(m \times n)$ -matrice, x le n -vecteur des inconnues et b le m -vecteur des seconds membres.

- Un vecteur x vérifiant le système (1.1) est appelé solution du système (1.1).
- Le système (1.1) est dit compatible s'il possède au moins une solution. Dans le cas contraire, il est dit incompatible.
- On appelle matrice augmentée, la matrice A à laquelle on a ajouté le vecteur b et on la note par $(A | b)$.
- Un système d'équations est dit équivalent à un autre système si toute solution du premier système est aussi une solution pour le deuxième système et inversement.
- Si le vecteur b est nul, le système (1.1) est dit homogène.

Théorème 1.1. Un système de m équations à n inconnues est compatible si et seulement si :

$$\text{rang}(A) = \text{rang}(A | b).$$

Démonstration. Appliquons les opérations élémentaires sur les lignes de la matrice augmentée, on aboutit à la matrice augmentée suivante :

$$\begin{pmatrix} 1 & 0 & \cdots & 0 & \tilde{a}_{1(r+1)} & \cdots & \tilde{a}_{1n} & \tilde{b}_1 \\ 0 & 1 & \cdots & 0 & \tilde{a}_{2(r+1)} & \cdots & \tilde{a}_{2n} & \tilde{b}_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & \tilde{a}_{r(r+1)} & \cdots & \tilde{a}_{rn} & \tilde{b}_r \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \tilde{b}_{r+1} \\ \vdots & \vdots & \cdots & 0 & 0 & \cdots & 0 & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \tilde{b}_{m-1} \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \tilde{b}_m \end{pmatrix}$$

où la notation $\tilde{a}_{ij}, \tilde{b}_i$ indique que les valeurs de a_{ij} et b_i ont été modifiées par des opérations élémentaires. Le rang de la matrice A est r .

Si $\tilde{b}_{r+1} = \tilde{b}_{r+2} = \cdots = \tilde{b}_m = 0$, alors le rang de la matrice augmentée $(A | b)$ correspond également à r et le système est compatible. En revanche, si l'une de ces valeurs est non nulle (par exemple $\tilde{b}_k \neq 0$, avec $r+1 \leq k \leq m$), le rang de la matrice $(A | b)$ vaut $r+1$. Le système est donc incompatible puisque l'équation $0 = \tilde{b}_k$ est absurde. \square

1.4 Factorisation LU

Soit A une matrice inversible de $\mathcal{M}_m(\mathbb{R})$:

$$A = A^{(1)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & \cdots & a_{1m}^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & \cdots & \cdots & a_{2m}^{(1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1}^{(1)} & a_{m2}^{(1)} & \cdots & \cdots & a_{mm}^{(1)} \end{pmatrix}, \quad a_{11}^{(1)} \neq 0,$$

et que L_i désigne la i -ème ligne de A . Nous avons déjà introduit la matrice E_{ij} de $\mathcal{M}_m(\mathbb{R})$.

Algorithme 1.1. (Algorithme de Gauss)

L'algorithme de Gauss conduit à une factorisation de la matrice A comme produit d'une matrice triangulaire inférieure dont tous les termes diagonaux sont égaux à 1 et d'une matrice triangulaire supérieure dont aucun terme diagonal n'est nul.

Étape 1 : L'élimination des coefficients de la première colonne dont l'indice de ligne est strictement supérieur à 1 se fait en procédant aux opérations élémentaires suivantes :

$$\forall i > 1, L_i \leftarrow L_i - x_{i1} L_1 \text{ où } x_{i1} = \frac{a_{i1}^{(1)}}{a_{11}^{(1)}},$$

et conduit à la matrice

$$A^{(2)} = \left(\prod_{i=0}^{m-2} (I_m - x_{(m-i)1} E_{(m-i)1}) \right) A = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & \cdots & a_{1m}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & \cdots & a_{2m}^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{m2}^{(2)} & \cdots & \cdots & a_{mm}^{(2)} \end{pmatrix}.$$

Notons par Q_1 la matrice triangulaire inférieure dont tous les termes diagonaux sont égaux à 1. Plus précisément :

$$Q_1 = \prod_{i=0}^{m-2} (I_m - x_{(m-i)1} E_{(m-i)1}) = \prod_{i=2}^m (I_m - x_{i1} E_{i1}) = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ -x_{21} & 1 & 0 & \cdots & 0 \\ \vdots & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 0 \\ -x_{m1} & 0 & \cdots & \cdots & 1 \end{pmatrix}$$

Étape k : Soit k un entier tel que $1 < k \leq m - 1$, supposons avoir obtenu après $k - 1$ étapes une matrice

$$A^{(k)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1(k-1)}^{(1)} & a_{1k}^{(1)} & \cdots & \cdots & a_{1m}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2(k-1)}^{(2)} & a_{2k}^{(2)} & \cdots & \cdots & a_{2m}^{(2)} \\ \vdots & \ddots & \ddots & \vdots & \vdots & \cdots & \cdots & \vdots \\ 0 & \cdots & 0 & a_{(k-1)(k-1)}^{(k-1)} & a_{(k-1)k}^{(k-1)} & \cdots & \cdots & a_{(k-1)m}^{(k-1)} \\ 0 & \cdots & 0 & 0 & a_{kk}^{(k)} & \cdots & \cdots & a_{km}^{(k)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & a_{mk}^{(k)} & \cdots & \cdots & a_{mm}^{(k)} \end{pmatrix},$$

dans laquelle aucun des termes diagonaux $a_{11}^{(1)}, a_{22}^{(2)}, \dots, a_{(k-1)(k-1)}^{(k-1)}$ n'est égal à 0. La forme de la deuxième expression rappelle que lors de la j ème étape, les lignes d'indice inférieur ou égal à j ne sont plus modifiées : la première ligne de $A^{(k)}$ est égale à la première ligne de $A^{(1)}$; la deuxième ligne de $A^{(k)}$ est égale à la deuxième ligne de $A^{(2)}$, etc. Supposons alors que le coefficient $a_{kk}^{(k)}$ soit différent de 0. Il est possible de le prendre comme pivot. L'élimination des coefficients de la k ème colonne de $A^{(k)}$ dont l'indice de ligne est strictement supérieur à k , s'effectue grâce aux

opérations élémentaires suivantes :

$$\forall i > k, L_i \leftarrow L_i - x_{ik}L_k, \text{ où } x_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}.$$

L'élimination revient à multiplier $A^{(k)}$ à gauche par la matrice Q_k suivante :

$$Q_k = \prod_{i=0}^{m-k-1} (I_m - x_{(m-i)k}E_{(m-i)k}) = \prod_{i=k+1}^m (I_m - x_{ik}E_{ik})$$

$$= \begin{pmatrix} 1 & 0 & \cdots & 0 & \cdots & \cdots & 0 & 0 \\ 0 & \ddots & \ddots & \vdots & & & & \vdots \\ \vdots & \ddots & 1 & 0 & & & & \vdots \\ \vdots & & 0 & 1 & 0 & & & \vdots \\ \vdots & & 0 & -x_{(k+1)k} & 1 & \ddots & & \vdots \\ \vdots & & \vdots & \vdots & 0 & \ddots & 0 & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \cdots & 0 & -x_{mk} & 0 & \cdots & 0 & 1 \end{pmatrix}.$$

On obtient ainsi, une matrice $A^{(k+1)} = Q_k A^{(k)}$ analogue à $A^{(k)}$. Ainsi, sous réserve qu'il soit possible de choisir comme pivot à l'étape k le scalaire $a_{kk}^{(k)}$, on obtient de proche en proche, une famille de $m - 1$ matrices (Q_1, \dots, Q_{m-1}) triangulaires inférieures dont tous les termes diagonaux sont égaux à 1 et une matrice $A^{(m)}$ triangulaire supérieure dont aucun de ses termes diagonaux n'est égal à 0, telles que

$$A^{(m)} = \left(\prod_{i=1}^{m-1} Q_{m-i} \right) A = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1(k-1)}^{(1)} & a_{1k}^{(1)} & \cdots & \cdots & a_{1m}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2(k-1)}^{(2)} & a_{2k}^{(2)} & & & a_{2m}^{(2)} \\ \vdots & \ddots & \ddots & \vdots & \vdots & & & \vdots \\ 0 & \cdots & 0 & a_{(k-1)(k-1)}^{(k-1)} & a_{(k-1)k}^{(k-1)} & \cdots & \cdots & a_{(k-1)m}^{(k-1)} \\ 0 & \cdots & 0 & 0 & a_{kk}^{(k)} & \cdots & \cdots & a_{km}^{(k)} \\ \vdots & & \vdots & \vdots & 0 & \ddots & & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & a_{mm}^{(m)} \end{pmatrix}.$$

Il est clair que la matrice $Q_{m-1} \cdots Q_1$ est encore triangulaire inférieure et que tous ses termes diagonaux sont égaux à 1. Elle est donc inversible. En effet

$$L = (Q_{m-1} \cdots Q_1)^{-1} = \prod_{k=1}^{m-1} Q_k^{-1},$$

et

$$Q_k^{-1} = \begin{pmatrix} 1 & 0 & \cdots & 0 & \cdots & \cdots & 0 & 0 \\ 0 & \ddots & \ddots & \vdots & & & & \vdots \\ \vdots & \ddots & 1 & 0 & & & & \vdots \\ \vdots & & 0 & 1 & 0 & & & \vdots \\ \vdots & & 0 & x_{(k+1)k} & 1 & \ddots & & \vdots \\ \vdots & & \vdots & \vdots & 0 & \ddots & 0 & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ 0 & \cdots & 0 & x_{mk} & 0 & \cdots & 0 & 1 \end{pmatrix},$$

d'où

$$L = \begin{pmatrix} 1 & 0 & \cdots & 0 & \cdots & \cdots & 0 & 0 \\ x_{21} & \ddots & \ddots & \vdots & & & & \vdots \\ x_{31} & x_{32} & 1 & 0 & & & & \vdots \\ \vdots & & \ddots & 1 & 0 & & & \vdots \\ \vdots & & & x_{(k+1)k} & 1 & \ddots & & \vdots \\ \vdots & & \vdots & \vdots & \ddots & \ddots & 0 & \vdots \\ \vdots & & \vdots & \vdots & \vdots & \ddots & 1 & 0 \\ x_{m1} & \cdots & \cdots & x_{mk} & \cdots & \cdots & x_{m(m-1)} & 1 \end{pmatrix}.$$

Posons $U = A^{(m)}$. On aura alors

$$A = LU.$$

Théorème 1.2. [52]

Soit A une matrice de $\mathcal{M}_m(\mathbb{R})$, A est complètement régulière si et seulement si A admet une décomposition LU .

Soit $A \in \mathcal{M}_m(\mathbb{R})$ non-singulière. Supposons que nous puissions écrire la matrice A sous forme d'un produit de deux matrices :

$$A = LU, \tag{1.2}$$

où $L = (\alpha_{ij}, 1 \leq i \leq m, 1 \leq j \leq m)$ est une matrice triangulaire inférieure et $U = (\beta_{ij}, 1 \leq i \leq m, 1 \leq j \leq m)$ est une matrice triangulaire supérieure. On peut utiliser cette décomposition pour résoudre le système :

$$Ax = (LU)x = L(Ux) = b. \tag{1.3}$$

En résolvant d'abord par rapport au vecteur y le système :

$$Ly = b, \quad (1.4)$$

puis en résolvant le système

$$Ux = y. \quad (1.5)$$

Quel est l'avantage de la technique de décomposition LU pour la résolution d'un système linéaire ?

L'avantage est que la solution d'un système d'équations triangulaires est assez triviale. Ainsi, l'équation (1.4) peut être résolue par substitution directe comme suit :

$$y_1 = \frac{b_1}{\alpha_{11}}, \quad (1.6)$$

$$y_i = \frac{1}{\alpha_{ii}} \left[b_i - \sum_{j=1}^{i-1} \alpha_{ij} y_j \right], \quad i = 2, 3, \dots, m. \quad (1.7)$$

Par substitution de y par (1.6)-(1.7), le système (1.5) peut alors être résolu comme suit :

$$x_m = \frac{y_m}{\beta_{mm}}, \quad (1.8)$$

$$x_i = \frac{1}{\beta_{ii}} \left[y_i - \sum_{j=i+1}^m \beta_{ij} x_j \right], \quad i = m-1, m-2, m-3, \dots, 1. \quad (1.9)$$

Remarque 1.1. Si A est une matrice non-singulière d'ordre m , alors $A^{-1} = (y_i, 1 \leq i \leq m)$, où $Ay_i = e_i$.

1.5 Mise à jour de l'inverse d'une matrice

Soit une matrice :

$$A = (a_1, a_2, \dots, a_p, \dots, a_m)$$

non-singulière dont l'inverse A^{-1} est connue. Supposons que nous devons trouver l'inverse de la matrice

$$\bar{A} = (a_1, a_2, \dots, a_q, \dots, a_m)$$

qui ne diffère de A que par la substitution du vecteur a_q non-nul au vecteur a_p . Nous savons, par définition, que tous les vecteurs a_j de A sont linéairement indépendants, puisque A est non-singulière. Le vecteur a_q peut donc s'exprimer par une combinaison linéaire des

vecteurs a_j : il existe alors un m -vecteur non nul $y = (y_j, j = 1, 2, \dots, m)$, tel que

$$a_q = Ay = \sum_{j=1}^m a_j y_j. \quad (1.10)$$

Pour que \bar{A}^{-1} existe, il faut que les vecteurs $a_1, \dots, a_q, \dots, a_m$ soient linéairement indépendants, c'est-à-dire que y_p dans l'équation (1.10) soit différent de zéro, puisque nous savons déjà que les autres sont indépendants. Nous pouvons alors écrire :

$$\begin{aligned} a_q &= \sum_{j \neq p}^m y_j a_j + y_p a_p \\ \Rightarrow a_p &= -\frac{1}{y_p} \sum_{j \neq p}^m y_j a_j + \frac{1}{y_p} a_q \\ \Rightarrow a_p &= \bar{A}v, \end{aligned}$$

où

$$v^T = \left(-\frac{y_1}{y_p}, -\frac{y_2}{y_p}, \dots, -\frac{y_{p-1}}{y_p}, \frac{1}{y_p}, -\frac{y_{p+1}}{y_p}, \dots, -\frac{y_m}{y_p} \right).$$

La matrice \bar{A} peut être écrite sous la forme suivante :

$$\bar{A} = AE, \quad (1.11)$$

avec

$$E = I_m + (y - e_p)e_p^T = (e_1, e_2, \dots, e_{p-1}, y, e_{p+1}, \dots, e_m),$$

où e_j désigne le j -ème vecteur unitaire. En effet,

$$\bar{A} = A + (a_q - a_p)e_p^T = A[I_m + A^{-1}(a_q - a_p)a_p^T].$$

Comme $A^{-1}a_q = y$ et $A^{-1}a_p = e_p$, on obtient $\bar{A} = AE$.

Puisque $\det(E) = y_p \neq 0$, alors E est inversible et

$$E^{-1} = I_m - \frac{1}{y_p}(y - e_p)e_p^T = (e_1, e_2, \dots, e_{p-1}, v, e_{p+1}, \dots, e_m).$$

Il en résulte alors que

$$\bar{A}^{-1} = E^{-1}A^{-1}.$$

Cette technique, dite forme produit de l'inverse, est particulièrement utile dans la méthode du simplexe ainsi que la méthode proposée dans le **chapitre 3** de cette thèse, puisque à chaque itération, la base ne diffère que par la substitution du vecteur a_{j_0} entrant au vecteur a_{j_1} sortant.

1.6 Espace vectoriel normé

Définition 1.6. Soit V un espace vectoriel sur \mathbb{R} , on appelle norme une application de V dans \mathbb{R}^+ , notée $\| \cdot \|$, qui vérifie les trois propriétés suivantes :

- $\forall x \in V, \forall \lambda \in \mathbb{R}, \|\lambda x\| = |\lambda| \|x\|.$
- $\forall x \in V, \forall y \in V, \|x + y\| \leq \|x\| + \|y\|.$
- $\|x\| = 0 \Leftrightarrow x = 0.$

Un espace vectoriel muni d'une norme est dit espace vectoriel normé.

Remarque 1.2. Un espace vectoriel de dimension finie muni d'une norme est dit espace vectoriel normé de dimension finie.

1.6.1 Normes fondamentales sur \mathbb{R}^n

Pour tout entier $p \in]1, +\infty[$, on définit l'application $\| \cdot \|_p$ de \mathbb{R}^n dans \mathbb{R}^+ par

$$\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}}.$$

Cette application est une norme, appelée norme de Hölder.

Les normes

$$\|x\|_1 = \sum_{i=1}^n |x_i|, \quad \|x\|_2 = \left(\sum_{i=1}^n |x_i|^2 \right)^{\frac{1}{2}}, \quad \|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$$

sont appelées normes fondamentales sur \mathbb{R}^n .

1.6.2 Définitions

Soit V un espace vectoriel normé sur \mathbb{R} .

Définition 1.7. On dit qu'une partie non vide S de V est une partie bornée de V si

$$\exists M > 0, \forall x \in S, \|x\| \leq M.$$

Définition 1.8. On dit qu'une suite d'éléments $(x_k)_k$ de V converge vers x si et seulement si :

$$\forall \epsilon > 0, \exists \hat{k} \in \mathbb{N}, \forall k \geq \hat{k}, \|x_k - x\| \leq \epsilon.$$

Définition 1.9. Dire que $S \subseteq V$ est fermé revient à dire que si une suite $(x_k)_k$ de S converge vers x de V alors nécessairement $x \in S$.

Définition 1.10. Soient f une application de V dans \mathbb{R} et $x_0 \in V$. On dit que f est continue en x_0 si

$$\forall \epsilon > 0, \exists \delta(\epsilon) > 0, \|x - x_0\| \leq \delta \Rightarrow |f(x) - f(x_0)| \leq \epsilon.$$

L'application f est continue sur $S \subseteq V$ si f est continue en tout point de S .

Définition 1.11. On appelle partie compacte d'un espace vectoriel normé de dimension finie toute partie bornée et fermée.

1.7 Analyse convexe

Définition 1.12. Soit x_1 et x_2 deux vecteurs de \mathbb{R}^n . Le segment $[x_1, x_2]$ est l'ensemble des points, défini par :

$$[x_1, x_2] = \{x \in \mathbb{R}^n : x = \lambda x_1 + (1 - \lambda)x_2, 0 \leq \lambda \leq 1\}.$$

Définition 1.13. Un ensemble S non vide de \mathbb{R}^n est dit ensemble convexe si et seulement si :

$$\forall x_1, x_2 \in S, [x_1, x_2] \subseteq S.$$

Définition 1.14. Soit S un ensemble convexe. Un point $x \in S$ est dit point extrême de S , s'il n'existe aucun segment $[u, v] \subset S$ ($u \neq v$) qui contient x en tant que point intérieur, c'est-à-dire, si la relation

$$x = \lambda u + (1 - \lambda)v,$$

avec un certain $\lambda \in]0, 1[$ et $u, v \in S$ est possible si et seulement si $u = v = x$. Autrement dit x est un point extrême de S si on ne peut pas l'exprimer comme une combinaison convexe de deux autres points différents de S .

Définition 1.15. Le vecteur y est appelé combinaison convexe des p vecteurs x_1, x_2, \dots, x_p s'il existe des constantes non-négatives $\lambda_1, \lambda_2, \dots, \lambda_p$ avec $\sum_{i=1}^p \lambda_i = 1$, telles que $y = \sum_{i=1}^p \lambda_i x_i$.

Définition 1.16. Soit $c \in \mathbb{R}^n$ et $b \in \mathbb{R}$. L'ensemble $S = \{x \in \mathbb{R}^n : c^T x \leq b\}$ est appelé demi-espace.

Définition 1.17. Soit $c \in \mathbb{R}^n$ et $b \in \mathbb{R}$. L'ensemble $S = \{x \in \mathbb{R}^n : c^T x = b\}$ est appelé hyperplan.

Définition 1.18. Soit $A \in \mathcal{M}_{m,n}(\mathbb{R})$ et $b \in \mathbb{R}^m$. L'ensemble $S = \{x \in \mathbb{R}^n : Ax \leq b\}$ est appelé polyèdre. Un polyèdre borné est appelé polytope.

Remarque 1.3. L'hyperplan est un ensemble convexe.

1.8 Programmation mathématique

Soit V un espace vectoriel normé sur \mathbb{R} .

1.8.1 Le problème fondamental

Soient $S \subseteq V$ et une fonction f réelle définie sur S . Le problème fondamental de la programmation mathématique est le suivant :

trouver un élément y dans S tel que $f(y) \geq f(x)$ pour tout $x \in S$.

Le problème est un problème de maximisation. On dit que le point y est un maximum de f . On dit aussi que la fonction f prend sa valeur maximale au point y . Le problème de minimisation de f sur S , i.e., trouver un y dans S tel que $f(y) \leq f(x)$ pour tout x , est équivalent au problème de maximisation de la fonction $-f$ sur S . Pour voir cela, observez que si $-f(y) \geq -f(x)$ pour tout $x \in S$, alors $f(y) \leq f(x)$ pour tout $x \in S$, et inversement. Donc

$$\max_{x \in S} f(x) \Leftrightarrow \min_{x \in S} -f(x).$$

Remarque 1.4. Si $S \subseteq \mathbb{R}^n$, nous obtenons alors le problème de programmation mathématique : maximiser ou minimiser une fonction réelle f sur S , où $S = \{x \in \mathbb{R}^n, h_i(x) \leq b_i, i = 1, \dots, m\}$. Ici f est appelée fonction objectif (fonction économique), S est l'ensemble réalisable et $h_i(x) \leq b_i, i = 1, \dots, m$ sont les contraintes ; h_i est une fonction à valeurs réelles définie sur S , $b_i \in \mathbb{R}$.

Remarque 1.5. Lorsque les fonctions f et $h_i, i = 1, \dots, m$ sont linéaires, il s'agit d'un problème de programmation linéaire. Si de plus, on impose que les variables ne peuvent prendre que des valeurs entières, on parle de programmation linéaire en nombres entiers. Les problèmes dans lesquels la fonction f ou h_i sont non-linéaires font partie de la programmation non-linéaire. Un cas particulier est la programmation quadratique relative aux problèmes pour lesquels la fonction f est quadratique et les fonctions h_i sont linéaires.

Nous exposons maintenant le théorème d'existence fondamental pour la maximisation (minimisation) des fonctions avec des domaines dans \mathbb{R}^n [53, 2, 21].

Théorème 1.3. Toute fonction réelle définie et continue sur un compact S atteint son maximum et son minimum sur S .

Remarque 1.6. Nous soulignons que le théorème 1.3 donne une condition suffisante pour l'existence d'un maximum et d'un minimum. Ce n'est pas une condition nécessaire.

Par la suite, nous étudierons essentiellement les problèmes de programmation linéaire.

Chapitre 2

La méthode primale du simplexe

Introduction

Dans ce chapitre, nous allons d'abord présenter les définitions de base et les théorèmes d'existence des solutions optimales des problèmes de programmation linéaire, puis en se basant sur [17], nous aborderons la méthode primale du simplexe.

2.1 Position du problème

La programmation linéaire est un cas particulier de la programmation mathématique, où la fonction objectif et les contraintes sont linéaires. Un problème de Programmation Linéaire (PL) s'écrit donc sous la forme standard suivante :

$$\max z(x) = c^T x, \quad (2.1)$$

$$Ax = b, \quad (2.2)$$

$$l \leq x \leq u, \quad (2.3)$$

où $A \in \mathcal{M}_{m \times n}(\mathbb{R})$, $b \in \mathbb{R}^m$, $x \in \mathbb{R}^n$ est le vecteur des variables et $c, l, u \in \mathbb{R}^n$. On suppose que $\|l\| < \infty$, $\|u\| < \infty$ et $\text{rang}(A) = m$. Lorsque cette dernière condition n'est pas vérifiée, on ajoutera dans ce cas des vecteurs unitaires appropriées aux colonnes de la matrice A (voir la remarque 4.1).

Plusieurs concepts jouent un rôle important dans l'étude des problèmes de programmation linéaire, notamment la notion de solution réalisable et celle de base.

2.2 Définitions

- La matrice A est appelée la matrice des contraintes, b est dit vecteur des seconds membres, l est le vecteur des bornes inférieures et u le vecteur des bornes supé-

rieures.

- Un vecteur x est dit pseudo-solution du problème (2.1)-(2.3) s'il vérifie les contraintes (2.2).
- Un vecteur x est dit solution réalisable (SR) du problème (2.1)-(2.3) s'il vérifie les contraintes (2.2)-(2.3). Notons par S l'ensemble des solutions réalisables.
- Une solution réalisable x^* est dite solution optimale si elle maximise la fonction objectif :

$$z(x^*) = \max_{x \in S} z(x) = \max_{x \in S} c^T x.$$

- Notons par $J = \{1, 2, \dots, n\}$ l'ensemble des indices des colonnes de la matrice A . Si $J_B \subset J$ et $J_N = J \setminus J_B$, alors $A_B = (a_j, j \in J_B)$ désigne la sous-matrice constituée des colonnes correspondant aux indices de l'ensemble J_B et $A_N = (a_j, j \in J_N)$ celle constituée des colonnes correspondant aux indices de J_N . Si v est un vecteur de \mathbb{R}^n , alors on écrit $v_B = (v_j, j \in J_B)$ et $v_N = (v_j, j \in J_N)$.
- Une base est un sous-ensemble d'indices $J_B \subset J$, vérifiant les deux propriétés suivantes :
 - $\text{Card}(J_B) = m$;
 - La sous-matrice A_B est non-singulière.

Les indices de l'ensemble J_B sont appelés indices de base. Notons par $J_N = J \setminus J_B$; les indices de l'ensemble J_N sont alors appelés indices hors-base. De même, la sous-matrice A_B est appelée matrice de base, les variables $x_j, j \in J_B$ sont appelées variables de base et les variables $x_j, j \in J_N$ sont appelées variables hors-base.

- Soit J_B une base. Alors le vecteur x vérifiant les relations :

$$x_j = l_j \text{ ou } u_j, j \in J_N \quad (2.4)$$

et

$$x_B = A_B^{-1}(b - A_N x_N) \quad (2.5)$$

est appelée solution de base associée à la base J_B ou solution de base.

- Lorsque la solution de base x vérifie $l_B \leq x_B \leq u_B$, alors on dit que x est une solution de base réalisable (SBR).

Notons par $J_{Nl} = \{j \in J_N : x_j = l_j\}$ et $J_{Nu} = \{j \in J_N : x_j = u_j\}$. Donc pour une SBR x , on a $x_{Nl} = l_{Nl}$ et $x_{Nu} = u_{Nu}$.

- Une SBR est dite non-dégénérée si

$$l_B < x_B < u_B.$$

Remarque 2.1. Supposons que $S \neq \emptyset$. Si $\|l\| < \infty$ et $\|u\| < \infty$, alors S est borné mais la réciproque n'est pas toujours vraie. Dans ce cas, le problème de PL (2.1)-(2.3) est appelé

problème de PL à variables bornées. Notons également que si S est borné, alors d'après le théorème 1.3, le problème (2.1)-(2.3) possède une solution optimale finie, mais la réciproque n'est pas toujours vraie.

Remarque 2.2. Le nombre de solutions de base réalisables du problème (2.1)-(2.3) est inférieur ou égal à $C_n^m = \frac{n!}{(n-m)!m!}$.

2.3 Recherche d'une solution optimale

Comme l'ensemble S des solutions réalisables du problème (2.1)-(2.3) est borné, alors on a les théorèmes importants suivants :

Théorème 2.1. Toute solution réalisable du problème (2.1)-(2.3) peut s'écrire comme une combinaison convexe des solutions de base réalisables du problème.

Théorème 2.2. L'ensemble S possède au moins un point extrême.

Théorème 2.3. Tout point extrême est une solution réalisable de base, et inversement.

Théorème 2.4. Si le problème de PL (2.1)-(2.3) possède une solution optimale, alors il existe une solution de base réalisable optimale.

Les preuves détaillées de ces théorèmes classiques de la PL dans le cas d'un problème de PL à variables positives ou nulles, écrit sous forme standard, i.e., $S = \{x \in \mathbb{R}^n, Ax = b, x \geq 0\}$, peuvent être trouvées dans la référence [16].

Ainsi pour trouver la solution optimale, il suffit d'examiner les points extrêmes de S ou de manière équivalente les solutions réalisables de base. Dans la section suivante, nous allons présenter la méthode primale du simplexe dont le principe consiste à passer d'un point extrême à un autre point extrême adjacent de S ayant une meilleure valeur de la fonction objectif. Le processus itératif s'arrête lorsque aucune amélioration n'est possible.

2.4 Méthode primale du simplexe

Pour décrire les étapes de la méthode primale du simplexe, il faut d'abord se poser les trois questions suivantes :

1. A l'étape initiale, comment sélectionner la solution de base réalisable initiale ?
2. A l'étape itérative :
 - Quels critères utiliser pour sélectionner la variable hors-base à faire entrer en base ?
 - Comment identifier la variable à faire sortir de la base pour devenir hors-base ?

- Comment identifier la nouvelle solution sans avoir à résoudre le nouveau système d'équations linéaires ainsi obtenu ?

3. A l'étape d'arrêt, comment effectuer le test d'optimalité ?

Pour répondre à toutes ces questions, nous présentons ici l'algorithme de la méthode primale du simplexe :

Algorithme 2.1. (MPS)

Étape initiale : Nous calculons une SBR initiale du problème (2.1)-(2.3) avec la technique utilisant plusieurs variables artificielles ou celle utilisant une seule variable artificielle ou la technique qui évite carrément l'utilisation des variables artificielles que nous proposons dans la section 4.2. Soit donc x une SBR initiale du problème et J_B la base correspondante.

Étape itérative :

Étape 1 : Calculer les ensembles J_{NI} et J_{Nu} ;

Étape 2 : Calculer le vecteur des multiplicateurs $\pi^T = c_B^T A_B^{-1}$ et le vecteur des coûts réduits $\Delta = A^T \pi - c$;

Étape 2.1 : Si les conditions suivantes sont vérifiées :

$$\Delta_j \geq 0 \text{ pour } j \in J_{NI}, \Delta_j \leq 0 \text{ pour } j \in J_{Nu}, \quad (2.6)$$

alors l'algorithme s'arrête avec x une SBR optimale ;

Étape 2.2 : Sinon, déterminer l'ensemble des indices non optimaux J_{NNO} qui contient les indices hors-base qui ne vérifient pas les conditions (2.6) et sélectionner une variable d'entrée x_{j_0} , $j_0 \in J_{NNO}$, vérifiant $|\Delta_{j_0}| = \max_{j \in J_{NNO}} |\Delta_j|$;

Étape 3 : Résoudre le système d'équations linéaires $A_B y = a_{j_0}$ par rapport à y ;

Étape 4 : Calculer $\theta = \min\{\min_{1 \leq i \leq m} \theta_i, u_{j_0} - l_{j_0}\}$, où

$$\theta_i = \begin{cases} \text{sign}(\Delta_{j_0}) \frac{l_{B_i} - x_{B_i}}{y_i}, & \text{si } \Delta_{j_0} y_i < 0; \\ \text{sign}(\Delta_{j_0}) \frac{u_{B_i} - x_{B_i}}{y_i}, & \text{si } \Delta_{j_0} y_i > 0; \\ +\infty, & \text{si } y_i = 0. \end{cases}$$

Étape 5 : Poser $x_B := x_B + \theta \text{sign}(\Delta_{j_0}) y$; $z := z + \theta |\Delta_{j_0}|$;

Étape 5.1 Pour $\theta = u_{j_0} - l_{j_0}$:

Si $j_0 \in J_{Nu}$, alors poser $J_{NI} := J_{NI} \cup \{j_0\}$ et $J_{Nu} := J_{Nu} \setminus \{j_0\}$;

Sinon $J_{Nu} := J_{Nu} \cup \{j_0\}$ et $J_{NI} := J_{NI} \setminus \{j_0\}$;

Étape 5.2 Pour $\theta < u_{j_0} - l_{j_0}$: poser $x_{j_0} := x_{j_0} - \theta \text{sign}(\Delta_{j_0})$;

Soient $j_1 \in J_B$, tel que $\theta_{j_1} = \theta$ et i_1 la position de j_1 dans J_B .

Si $\Delta_{j_0} y_{i_1} < 0$, alors poser $J_{Nl} := J_{Nl} \cup \{j_1\}$;

Si $\Delta_{j_0} y_{i_1} > 0$, alors poser $J_{Nu} := J_{Nu} \cup \{j_1\}$;

Supprimer j_0 de $J_{Nl} \cup J_{Nu}$. Poser $J_B := J_B \setminus \{j_1\} \cup \{j_0\}$ et aller à l'étape 1 ;

Théorème 2.5. Si l'algorithme 2.2 s'arrête à l'étape 2.1, alors x est une solution de base réalisable optimale du problème (2.1)-(2.3).

Démonstration. Par hypothèse, x est une solution réalisable de base. Supposons que l'algorithme se termine à l'étape 2.1. Soit $\Delta = A^T \pi - c$, où π est calculé dans l'étape 2. Maintenant, pour toute solution réalisable \bar{x} , on a

$$\begin{aligned} c^T \bar{x} - c^T x &= c^T (\bar{x} - x) \\ &= -\Delta_N^T (\bar{x}_N - x_N) \\ &= -\Delta_{Nl}^T (\bar{x}_{Nl} - l_{Nl}) - \Delta_{Nu}^T (\bar{x}_{Nu} - u_{Nu}) \leq 0. \end{aligned}$$

La dernière inégalité découle du fait que $\Delta_{Nl} \geq 0$, $\bar{x}_{Nl} \geq l_{Nl}$, $\Delta_{Nu} \leq 0$ et $\bar{x}_{Nu} \leq u_{Nu}$. Par conséquent, $c^T \bar{x} \leq c^T x$. Donc x est optimale.

(Mise à jour). Il faut montrer que si la solution courante n'est pas optimale, alors la nouvelle solution x est une SBR associée à la nouvelle base J_B .

Si l'étape 5.1 se produit, le nouveau J_B est identique à l'ancien J_B , et donc évidemment une base. Pour le cas restant, il faut montrer que la nouvelle matrice A_B est non-singulière. Désignons la nouvelle matrice par \bar{A}_B , il s'en suit que

$$\bar{A}_B = A_B E,$$

où

$$E = I_m + (y - e_{i_1}) e_{i_1}^T.$$

Puisque $y_{i_1} \neq 0$ par la définition de θ , alors E est inversible. Comme A_B est inversible, on déduit qu \bar{A}_B est inversible. \square

2.5 Résolution des problèmes de PL sous forme générale

Considérons le problème de PL (2.1)-(2.3) pour lequel les valeurs $-\infty$ pour les bornes inférieures et $+\infty$ pour les bornes supérieures sont autorisées.

- Lorsque $l_j = 0$ et $u_j = +\infty$, $j = 1, 2, \dots, n$, on dit que les variables x_j sont non-négatives.
- Si $l_j = -\infty$ et $u_j = +\infty$, la variable x_j est dite libre ou sans restriction de signe.

- Si $l_j = u_j \neq \infty$, alors x_j est dite une variable fixe. Notons qu'il est particulièrement plus aisé de remplacer une variable x_j sans restriction de signe par la différence de deux variables non-négatives.
- Notons l'ensemble des variables libres et celui des variables fixes, respectivement, par :

$$J_{Nlr} = \{j \in J_N : x_j \text{ est libre}\}, J_{Nfx} = \{j \in J_N : x_j \text{ est fixe}\}.$$

- Une base est un triplet (J_B, J_{Nl}, J_{Nu}) vérifiant les deux propriétés suivantes [17] :
 - La sous-matrice A_B est non-singulière ;
 - $J_{Nl} \cap J_{Nu} = \emptyset$, $J_{Nl} \cup J_{Nu} = \{j \notin J_B : x_j \text{ n'est ni fixe ni libre}\}$, $l_j > -\infty$ pour $j \in J_{Nl}$ et $u_j < +\infty$ pour $j \in J_{Nu}$.
- La solution de base x associée à la base (J_B, J_{Nl}, J_{Nu}) est donnée par :

$$\begin{aligned} x_{Nl} &= l_{Nl}, \\ x_{Nu} &= u_{Nu}, \\ x_{Nfx} &= l_{Nfx} = u_{Nfx}, \\ x_{Nlr} &= 0, \end{aligned} \tag{2.7}$$

et

$$x_B = A_B^{-1}(b - A_N x_N). \tag{2.8}$$

- Lorsque la solution de base x vérifie $l_B \leq x_B \leq u_B$, alors on dit que x est une solution de base réalisable (SBR).
- Si la valeur optimale de la fonction objectif est infinie, alors le problème de PL est dit non borné.

Pour résoudre le problème de PL sous forme générale, nous présentons l'algorithme décrit dans [17] :

Algorithme 2.2. (MPS)

Étape initiale : Nous calculons une SBR initiale du problème (2.1)-(2.3) avec la technique utilisant plusieurs variables artificielles ou celle utilisant une seule variable artificielle ou la technique qui évite carrément l'utilisation des variables artificielles que nous proposons dans la section 4.2. Soit donc x une SBR initiale du problème et J_B la base correspondante.

Étape itérative :

Étape 1 : Calculer les ensembles J_{Nl} , J_{Nu} , J_{Nlr} et J_{Nfx} ;

Étape 2 : Calculer le vecteur des multiplicateurs $\pi^T = c_B^T A_B^{-1}$ et le vecteur des coûts réduits $\Delta = A^T \pi - c$;

Étape 2.1 : Si les conditions suivantes sont vérifiées :

$$\Delta_j \geq 0 \text{ pour } j \in J_{NI}, \Delta_j \leq 0 \text{ pour } j \in J_{Nu}, \text{ et } \Delta_j = 0 \text{ pour } j \in J_{Nlr}, \quad (2.9)$$

alors l'algorithme s'arrête avec x une SBR optimale ;

Étape 2.2 : Sinon, déterminer l'ensemble des indices non optimaux J_{NNO} qui contient les indices non basiques qui ne vérifient pas les conditions (2.9) et sélectionner une variable d'entrée x_{j_0} , $j_0 \in J_{NNO}$, vérifiant $|\Delta_{j_0}| = \max_{j \in J_{NNO}} |\Delta_j|$;

Étape 3 : Résoudre le système d'équations linéaire $A_B y = a_{j_0}$ par rapport à y ;

Étape 4 : Calculer $\theta = \min\{\min_{1 \leq i \leq m} \theta_i, u_{j_0} - l_{j_0}\}$, où

$$\theta_i = \begin{cases} \text{sign}(\Delta_{j_0}) \frac{l_{B_i} - x_{B_i}}{y_i}, & \text{si } \Delta_{j_0} y_i < 0; \\ \text{sign}(\Delta_{j_0}) \frac{u_{B_i} - x_{B_i}}{y_i}, & \text{si } \Delta_{j_0} y_i > 0; \\ +\infty, & \text{si } y_i = 0. \end{cases}$$

Si $\theta = +\infty$, alors l'algorithme s'arrête. Le problème est non borné ;

Étape 5 : Poser $x_B := x_B + \text{sign}(\Delta_{j_0}) \theta y$, $z := z + \theta |\Delta_{j_0}|$;

Étape 5.1 Pour $\theta = u_{j_0} - l_{j_0}$:

Si $j_0 \in J_{Nu}$, alors poser $J_{NI} := J_{NI} \cup \{j_0\}$ et $J_{Nu} := J_{Nu} \setminus \{j_0\}$;

Sinon $J_{Nu} := J_{Nu} \cup \{j_0\}$ et $J_{NI} := J_{NI} \setminus \{j_0\}$;

Étape 5.2 Pour $\theta < u_{j_0} - l_{j_0}$: soit $j_1 \in J_B$, tel que $\theta_{j_1} = \theta$. Poser

$$x_{j_0} := \begin{cases} \theta, & \text{si } j_0 \in J_{Nlr} \text{ et } \Delta_{j_0} < 0; \\ -\theta, & \text{si } j_0 \in J_{Nlr} \text{ et } \Delta_{j_0} > 0; \\ l_{j_0} + \theta, & \text{si } j_0 \in J_{NI}; \\ u_{j_0} - \theta, & \text{si } j_0 \in J_{Nu}. \end{cases}$$

Si la variable sortante x_{j_1} est fixe, alors poser $J_{Nfx} := J_{Nfx} \cup \{j_1\}$; sinon, soit i_1 la position de j_1 dans J_B .

Si $\Delta_{j_0} y_{i_1} < 0$, alors poser $J_{NI} := J_{NI} \cup \{j_1\}$;

Si $\Delta_{j_0} y_{i_1} > 0$, alors poser $J_{Nu} := J_{Nu} \cup \{j_1\}$;

Supprimer j_0 de $J_{Nlr} \cup J_{NI} \cup J_{Nu}$. Poser $J_B := J_B \setminus \{j_1\} \cup \{j_0\}$ et aller à l'étape 1 ;

Théorème 2.6. [17] Si l'algorithme 2.2 s'arrête à l'étape 2.1, alors x est une solution de base réalisable optimale du problème (2.1)-(2.3). Si l'algorithme se termine à l'étape 4, alors la valeur optimale de la fonction objectif du problème (2.1)-(2.3) est infinie (le problème est non borné).

Démonstration. L'optimalité de la solution x et la non-singularité de la nouvelle base se démontrent d'une façon similaire à celle utilisée pour la preuve du cas borné (théorème 2.5). (**Infinité**). Supposons que $\Delta_{j_0} < 0$ et pour $\alpha \geq 0$, définissons

$$\begin{aligned}x_{j_0}^\alpha &= x_{j_0} + \alpha, \\x_B^\alpha &= x_B - \alpha y, \text{ et} \\x_j^\alpha &= x_j \text{ pour } j \in J_N \setminus \{j_0\},\end{aligned}$$

où $A_B y = a_{j_0}$ et x est donné par (2.7)-(2.8). Il est simple de voir que $Ax^\alpha = b$. Supposons que $\alpha \leq \theta$. Alors clairement $l_{j_0} \leq x_{j_0}^\alpha \leq u_{j_0}$, et donc $l_N \leq x_N^\alpha \leq u_N$, en outre $0 \leq \alpha \leq \min_i \theta_i$ implique $l_B \leq x_B^\alpha \leq u_B$. Par conséquent, x^α est une solution réalisable, et

$$\begin{aligned}z^\alpha &= c^T x^\alpha = c^T x + \alpha c_{j_0} - \alpha c_B^T y, \\&= c^T x + \alpha c_{j_0} - \alpha \pi^T a_{j_0}, \\&= c^T x - \alpha \Delta_{j_0}.\end{aligned}$$

Il s'en suit que si $\theta = +\infty$, alors z^α est infinie. D'une façon similaire on traite le cas $\Delta_{j_0} > 0$, en prenant $x_{j_0}^\alpha = x_{j_0} - \alpha$ et $x_B^\alpha = x_B + \alpha y$.

□

2.6 Exemples numériques

Les exemples suivants illustrent respectivement les trois cas suivants :

1. Le maximum existe avec un ensemble de solutions réalisables borné.
2. Le maximum existe avec un ensemble de solutions réalisables non borné.
3. Le problème est non borné.

Exemple 2.1.

$$\begin{aligned}\max z &= c^T x, \\Ax &= b, \quad l \leq x \leq u,\end{aligned}\tag{2.10}$$

où

$$A = \begin{pmatrix} 1 & -1 & 3 & 2 \\ -7 & 1 & 2 & 3 \end{pmatrix}, \quad b = (1, 1)^T, \quad c = (4, -6, -2, 2)^T, \quad u = (1, 2, 3, 4)^T, \quad l = -u.$$

Itération 1

- La solution de base initiale et la valeur de la fonction objectif z :

$$J_B = \{2, 3\}, J_{Nu} = \{1, 4\},$$

$$x_N = (1, 4)^T, x_B = A_B^{-1}[b - A_N x_N] = \left(\frac{4}{5}, \frac{-12}{5}\right)^T.$$

La valeur de la fonction objectif est $z = 12$.

- Les vecteurs π et Δ :

$$\pi^T = c_B^T A_B^{-1} = (2, -4), \Delta = A^T \pi - c = (26, 0, 0, -10)^T.$$

- Le vecteur et l'indice entrants :

$$j_0 = 1, a_{j_0} = (1, -7)^T.$$

- Les coordonnées de a_{j_0} par rapport à J_B :

$$y = A_B^{-1} a_{j_0} = \left(\frac{-23}{5}, \frac{-6}{5}\right)^T.$$

- Calcul de θ : On a $\Delta_{j_0} = 26 > 0$. Donc

$$\theta = \min\left\{\frac{14}{23}, \frac{1}{2}, 2\right\} = \frac{1}{2} \text{ et } j_1 = 3.$$

- La nouvelle solution réalisable de base :

$$x = \left(\frac{1}{2}, \frac{-3}{2}, -3, 4\right)^T, z = 25.$$

Itération 2

- La solution de base et la valeur de la fonction objectif z :

$$J_B = \{1, 2\}, J_{Nu} = \{3\}, J_{Nu} = \{4\},$$

$$x_N = (-3, 4)^T, x_B = \left(\frac{1}{2}, \frac{-3}{2}\right)^T.$$

La valeur de la fonction objectif est $z = 25$.

- Les vecteurs π et Δ :

$$\pi^T = c_B^T A_B^{-1} = \left(\frac{19}{3}, \frac{1}{3}\right), \Delta = A^T \pi - c = \left(0, 0, \frac{65}{3}, \frac{35}{3}\right)^T.$$

- Le vecteur et l'indice entrants :

$$j_0 = 4, a_{j_0} = (2, 3)^T.$$

- Les coordonnées de a_{j_0} par rapport à J_B :

$$y = A_B^{-1} a_{j_0} = \left(\frac{-5}{6}, \frac{-17}{6} \right)^T.$$

- Calcul de θ : On a $\Delta_{j_0} = \frac{35}{3} > 0$. Donc

$$\theta = \min\left\{ \frac{9}{5}, \frac{3}{17}, 4 \right\} = \frac{3}{17} \text{ et } j_1 = 2.$$

- La nouvelle solution réalisable de base :

$$x = \left(\frac{6}{17}, -2, -3, \frac{65}{17} \right)^T, \quad z = \frac{460}{17} \simeq 27.06.$$

Itération 3

- La solution de base et la valeur de la fonction objectif z :

$$J_B = \{1, 4\}, \quad J_{Nl} = \{2, 3\},$$

$$x_N = (-2, -3)^T, \quad x_B = \left(\frac{6}{17}, \frac{65}{17} \right)^T.$$

La valeur de la fonction objectif est $z = \frac{460}{17}$.

- Les vecteurs π et Δ :

$$\pi^T = c_B^T A_B^{-1} = \left(\frac{26}{17}, \frac{-6}{17} \right), \quad \Delta = A^T \pi - c = \left(0, \frac{70}{17}, \frac{100}{17}, 0 \right)^T.$$

Donc le vecteur $x = \left(\frac{6}{17}, -2, -3, \frac{65}{17} \right)^T$ est une solution réalisable de base optimale du problème (2.10) et la valeur optimale de la fonction objectif est $z = \frac{460}{17}$.

Exemple 2.2.

$$\begin{aligned} \max z &= 4x_1 + 3x_2, \\ 2x_1 + x_2 + x_3 &= 10, \\ x_1 + x_2 + x_4 &= 8, \\ 0 \leq x_1 \leq 10, \quad x_2 &\leq 4, \quad x_3 \in \mathbb{R}, \quad x_4 \geq 5. \end{aligned} \tag{2.11}$$

Itération 1

$$A = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}, \quad b = (10, 8)^T, \quad c = (4, 3, 0, 0)^T.$$

- La solution de base initiale et la valeur de la fonction objectif z :

$$J_B = \{2, 3\}, \quad J_{Nl} = \{1, 4\},$$

$$x_N = (0, 5)^T, \quad x_B = A_B^{-1}[b - A_N x_N] = (3, 7)^T.$$

La valeur de la fonction objectif est $z = 9$.

- Les vecteurs π et Δ :

$$\pi^T = c_B^T A_B^{-1} = (0, 3), \quad \Delta = A^T \pi - c = (-1, 0, 0, 3)^T.$$

- Le vecteur et l'indice entrants :

$$j_0 = 1, \quad a_{j_0} = (2, 1)^T.$$

- Les coordonnées de a_{j_0} par rapport à J_B :

$$y = A_B^{-1} a_{j_0} = (1, 1)^T.$$

- Calcul de θ : On a $\Delta_{j_0} = -1 < 0$. Donc $\theta = \min\{+\infty, +\infty, 10\} = 10$ et $j_1 = j_0 = 1$.

- La nouvelle solution réalisable de base :

$$x = (10, -7, -3, 5)^T, \quad z = 19.$$

Itération 2

- La solution de base et la valeur de la fonction objectif z :

$$J_B = \{2, 3\}, \quad J_{Nl} = \{4\}, \quad J_{Nu} = \{1\},$$

$$x_N = (10, 5)^T, \quad x_B = (-7, -3)^T.$$

La valeur de la fonction objectif est $z = 19$.

- Les vecteurs π et Δ :

$$\pi^T = c_B^T A_B^{-1} = (0, 3), \quad \Delta = A^T \pi - c = (-1, 0, 0, 3)^T.$$

Donc le vecteur $x = (10, -7, -3, 5)^T$ est une solution réalisable de base optimale du problème (2.11) et la valeur optimale de la fonction objectif est $z = 19$.

Exemple 2.3.

$$\begin{aligned} \max z &= -2x_1 + x_2 + 5x_3, \\ 5x_1 - 2x_2 + 4x_4 &= 4, \\ 5x_1 - 3x_2 + 3x_3 &= 9, \\ x_1 \leq -5, \quad x_2, &\in \mathbb{R}, \quad 0 \leq x_3 \leq 13, \quad x_4 \geq 2. \end{aligned} \tag{2.12}$$

Itération 1

- La solution de base initiale et la valeur de la fonction objectif z :

$$J_B = \{3, 4\}, J_{Nu} = \{1\}, J_{Nlr} = \{2\},$$

$$x_N = (-5, 0)^T, x_B = A_B^{-1}[b - A_N x_N] = \left(\frac{34}{3}, \frac{29}{4}\right)^T.$$

La valeur de la fonction objectif est $z = \frac{200}{3} \simeq 66.67$.

- Les vecteurs π et Δ :

$$\pi^T = c_B^T A_B^{-1} = \left(0, \frac{5}{3}\right), \Delta = A^T \pi - c = \left(\frac{31}{3}, -6, 0, 0\right)^T.$$

- Le vecteur et l'indice entrants :

$$j_0 = 1, a_{j_0} = (5, 5)^T.$$

- Les coordonnées de a_{j_0} par rapport à J_B :

$$y = A_B^{-1} a_{j_0} = \left(\frac{5}{3}, \frac{5}{4}\right)^T.$$

- Calcul de θ : On a $\Delta_{j_0} = \frac{31}{3} > 0$. Donc

$$\theta = \min\{1, +\infty, +\infty\} = 1 \text{ et } j_1 = 3.$$

- La nouvelle solution réalisable de base :

$$x = \left(-6, 0, 13, \frac{17}{2}\right)^T, z = \frac{231}{3} = 77.$$

Itération 2

- La solution de base et la valeur de la fonction objectif z :

$$J_B = \{1, 4\}, J_{Nlr} = \{2\}, J_{Nu} = \{3\},$$

$$x_N = (0, 13)^T, x_B = \left(-6, \frac{17}{2}\right)^T.$$

La valeur de la fonction objectif est $z = 77$.

- Les vecteurs π et Δ :

$$\pi^T = c_B^T A_B^{-1} = \left(0, \frac{-2}{5}\right), \Delta = A^T \pi - c = \left(0, \frac{1}{5}, \frac{-31}{5}, 0\right)^T.$$

- Le vecteur et l'indice entrants :

$$j_0 = 2, a_{j_0} = (-2, -3)^T.$$

- Les coordonnées de a_{j_0} par rapport à J_B :

$$y = A_B^{-1} a_{j_0} = \left(\frac{-3}{5}, \frac{1}{4}\right)^T.$$

- Calcul de θ : On a $\Delta_{j_0} = \frac{1}{5} > 0$. Donc

$$\theta = \min\{+\infty, +\infty, +\infty\} = +\infty.$$

Par conséquent, le problème (2.12) est non borné.

Chapitre 3

Une Nouvelle Méthode à Direction Hybride et à Pas Multiple (MDHPM) pour la PL

Introduction

Dans ce chapitre, après avoir rappelé la « Méthode Adaptée à Direction Hybride (MADH) » développée dans [14, 7], nous proposons une nouvelle méthode appelée « Méthode à Direction Hybride et à Pas Multiple (MDHPM) » pour résoudre les programmes linéaires à variables bornées [28]. L'algorithme proposé commence par une solution réalisable de support initiale, puis il passe d'une solution réalisable de support à une autre solution meilleure en suivant une nouvelle direction hybride. La direction construite donne une meilleure amélioration locale de la fonction objectif que la direction de l'algorithme MADH. Afin d'arrêter l'algorithme, nous utilisons un critère de suboptimalité et nous adaptons la règle du pas multiple pour changer le support courant.

3.1 Position du problème et définitions

Le problème de programmation linéaire à variables bornées se présente sous la forme standard suivante :

$$\max z = c^T x, \quad (3.1)$$

$$Ax = b, \quad (3.2)$$

$$l \leq x \leq u, \quad (3.3)$$

où c , x , l et u sont des n -vecteurs ; $\|l\| < \infty$, $\|u\| < \infty$; b un m -vecteur ; A est une matrice d'ordre $m \times n$, avec $\text{rang}(A) = m < n$.

- Définissons les ensembles d'indices suivants :

$$I = \{1, 2, \dots, m\}, \quad J = \{1, 2, \dots, n\}, \quad J = J_B \cup J_N, \quad J_B \cap J_N = \emptyset, \quad \text{Card}(J_B) = m.$$

- Une solution réalisable x^ϵ est appelée ϵ -optimale ou *suboptimale* si

$$z(x^0) - z(x^\epsilon) = c^T x^0 - c^T x^\epsilon \leq \epsilon,$$

où x^0 est une solution optimale du problème (3.1)-(3.3) et ϵ un nombre positif ou nul choisi à l'avance.

- Soit un sous-ensemble d'indices $J_B \subset J$ tel que $\text{Card}(J_B) = \text{Card}(I) = m$. L'ensemble J_B est alors appelé *support* si

$$\det(A_B) = \det(A(I, J_B)) \neq 0.$$

- Le couple $\{x, J_B\}$ formé de la solution réalisable x et du support J_B est appelé *solution réalisable de support* (SRS).
- Une SRS est dite non-dégénérée si

$$l_j < x_j < u_j, \quad j \in J_B.$$

3.2 Formule d'accroissement de la fonction objectif

Soit $\{x, J_B\}$ une SRS du problème (3.1)-(3.3). Considérons une autre solution réalisable quelconque $\bar{x} = x + \Delta x$, c'est-à-dire :

$$A\Delta x = 0, \tag{3.4}$$

et

$$l - x \leq \Delta x \leq u - x. \tag{3.5}$$

De la relation (3.4), nous obtenons :

$$\Delta x_B = -A_B^{-1} A_N \Delta x_N.$$

L'accroissement de la fonction objectif s'écrit

$$\begin{aligned} \Delta z &= z(\bar{x}) - z(x) = c^T \Delta x \\ &= c_B^T \Delta x_B + c_N^T \Delta x_N \\ &= -c_B^T A_B^{-1} A_N \Delta x_N + c_N^T \Delta x_N \\ &= -(c_B^T A_B^{-1} A_N - c_N^T) \Delta x_N. \end{aligned}$$

D'où

$$\Delta z = - \sum_{j \in J_N} \Delta_j \Delta x_j. \quad (3.6)$$

3.3 Critère d'optimalité et de suboptimalité

Théorème 3.1. (Critère d'optimalité [36]). Soit $\{x, J_B\}$ une SRS du problème (3.1)-(3.3). Alors les relations

$$\begin{aligned} \Delta_j &\geq 0, \text{ si } x_j = l_j; \\ \Delta_j &\leq 0, \text{ si } x_j = u_j; \\ \Delta_j &= 0, \text{ si } l_j < x_j < u_j, j \in J_N \end{aligned} \quad (3.7)$$

sont suffisantes pour l'optimalité de la solution réalisable x . Ces mêmes relations sont aussi nécessaires dans le cas où la SRS $\{x, J_B\}$ est non-dégénérée.

Démonstration. Suffisance. Soit $\{x, J_B\}$ une SRS du problème (3.1)-(3.3) vérifiant les relations (3.7). Pour toute solution réalisable $\bar{x} = x + \Delta x$ du problème (3.1)-(3.3), la formule d'accroissement (3.6) nous donne

$$z(\bar{x}) - z(x) = - \sum_{\Delta_j > 0, j \in J_N} \Delta_j (\bar{x}_j - l_j) - \sum_{\Delta_j < 0, j \in J_N} \Delta_j (\bar{x}_j - u_j).$$

Comme on a

$$l_j \leq \bar{x}_j \leq u_j \Rightarrow \bar{x}_j - l_j \geq 0 \text{ et } \bar{x}_j - u_j \leq 0,$$

on en déduit que

$$z(\bar{x}) - z(x) \leq 0 \Rightarrow z(\bar{x}) \leq z(x).$$

Par conséquent, le vecteur x est une solution optimale du problème (3.1)-(3.3).

Nécessité. Soit $\{x, J_B\}$ une SRS optimale non-dégénérée du problème (3.1)-(3.3) et supposons que les relations (3.7) ne sont pas vérifiées, c'est-à-dire, $\exists j_0 \in J_N$, tel que

$$\Delta_{j_0} > 0 \text{ et } x_{j_0} > l_{j_0} \text{ ou bien } \Delta_{j_0} < 0 \text{ et } x_{j_0} < u_{j_0}.$$

Construisons alors une autre solution réalisable $\bar{x} = x + \theta d$, où θ est un nombre réel positif et d un vecteur vérifiant

$$\begin{aligned} d_{j_0} &= -\text{sign}(\Delta_{j_0}); \\ d_j &= 0, j \neq j_0, j \in J_N; \\ d_B &= -A_B^{-1} A_N d_N. \end{aligned}$$

On a bien

$$Ad = A_B d_B + A_N d_N = 0$$

et

$$A\bar{x} = Ax + \theta Ad = b.$$

Pour que \bar{x} soit une solution réalisable du problème (3.1)-(3.3), il doit en plus vérifier l'inégalité

$$l \leq \bar{x} \leq u \Leftrightarrow l - x \leq \theta d \leq u - x,$$

soit en écrivant composante par composante :

$$\begin{aligned} l_j - x_j &\leq \theta d_j \leq u_j - x_j, \quad j \in J_B; \\ l_{j_0} - x_{j_0} &\leq -\theta \text{sign}(\Delta_{j_0}) \leq u_{j_0} - x_{j_0}. \end{aligned} \quad (3.8)$$

Puisque la SRS $\{x, J_B\}$ est non-dégénérée, on a alors

$$l_j - x_j < 0 < u_j - x_j, \quad j \in J_B.$$

De plus, on a

$$\begin{aligned} l_{j_0} - x_{j_0} &< 0, \text{ si } \Delta_{j_0} > 0; \\ u_{j_0} - x_{j_0} &> 0, \text{ si } \Delta_{j_0} < 0. \end{aligned} \quad (3.9)$$

Pour un nombre $\theta > 0$ assez petit, les relations (3.8) seront alors vérifiées et le vecteur \bar{x} sera une solution réalisable du problème (3.1)-(3.3). La formule d'accroissement (3.6) nous donne donc

$$z(\bar{x}) - z(x) = \theta |\Delta_{j_0}| > 0.$$

On en déduit que $z(\bar{x}) > z(x)$. Mais ceci est en contradiction avec le fait que x est une solution optimale du problème (3.1)-(3.3). Par conséquent, si $\{x, J_B\}$ est une SRS optimale non-dégénérée, alors les relations (3.7) sont forcément vérifiées. \square

Pour une SRS $\{x, J_B\}$ du problème (3.1)-(3.3), nous faisons la partition suivante :

$$J_N = J_N^+ \cup J_N^- \cup J_N^0,$$

où

$$J_N^+ = \{j \in J_N : \Delta_j > 0\}, \quad J_N^- = \{j \in J_N : \Delta_j < 0\} \text{ et } J_N^0 = \{j \in J_N : \Delta_j = 0\}. \quad (3.10)$$

La quantité non-négative $\beta(x, J_B)$ définie par

$$\beta = \beta(x, J_B) = \sum_{j \in J_N^+} \Delta_j (x_j - l_j) + \sum_{j \in J_N^-} \Delta_j (x_j - u_j), \quad (3.11)$$

est appelée l'estimation de suboptimalité [32]. Ainsi, nous avons les résultats suivants [32] :

Théorème 3.2. (Condition suffisante pour la suboptimalité)

Soit $\{x, J_B\}$ une SRS du problème (3.1)-(3.3) et ϵ un nombre positif arbitraire. Si $\beta(x, J_B) \leq \epsilon$, alors la solution réalisable x est ϵ -optimale.

Corollaire 3.1. Soit $\{x, J_B\}$ une SRS du problème (3.1)-(3.3). La condition $\beta(x, J_B) = 0$ est suffisante pour l'optimalité de la solution réalisable x .

3.4 Rappel sur la Méthode Adaptée à Direction Hybride (MADH)

La méthode adaptée à direction hybride [14] est une méthode itérative pour trouver la solution optimale d'un problème de programmation linéaire écrit sous la forme (3.1)-(3.3).

Dans ce chapitre, nous supposons que l'ensemble des solutions réalisables du problème (3.1)-(3.3) est non vide. Le schéma de la méthode adaptée à direction hybride est alors décrit dans les étapes suivantes [14, 7] :

Algorithme 3.1. (MADH)

Étape initiale : Soit $\eta \in [0, 1]$ et $\{x, J_B\}$ une solution réalisable de support initiale (la technique permettant de rechercher cette solution est décrite dans le chapitre suivant, section 4.2);

Étape itérative :

Étape 1 : Calculer $\pi^T = c_B^T A_B^{-1}$, $\Delta^T = \pi^T A - c^T$;

Étape 2 : Calculer les ensembles

$$\begin{aligned} J_{N+} &= \{j \in J_N : \Delta_j > \eta(x_j - l_j)\}, \quad J_{N-} = \{j \in J_N : \Delta_j < \eta(x_j - u_j)\}, \\ J_N^{P+} &= \{j \in J_N : 0 < \Delta_j \leq \eta(x_j - l_j)\}, \quad J_N^{P-} = \{j \in J_N : \eta(x_j - u_j) \leq \Delta_j < 0\}, \\ J_N^{P0} &= \{j \in J_N : \Delta_j = 0\}, \quad J_N^P = J_N^{P-} \cup J_N^{P+} \cup J_N^{P0}; \end{aligned}$$

Étape 3 : Calculer l'estimation d'optimalité $\gamma(\eta, x, J_B)$:

$$\gamma = \gamma(\eta, x, J_B) = \begin{cases} \sum_{j \in J_{N+}} \Delta_j(x_j - l_j) + \sum_{j \in J_{N-}} \Delta_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_N^{P+} \cup J_N^{P-}} \Delta_j^2, & \text{si } \eta > 0; \\ \beta(x, J_B), & \text{si } \eta = 0; \end{cases}$$

Étape 4 : Si $\gamma = 0$, alors l'algorithme s'arrête avec $\{x, J_B\}$, une SRS optimale. Stop;

Étape 5 : Calculer la direction d'amélioration d :

$$\begin{aligned} d_j &= l_j - x_j, \text{ si } j \in J_{N^+}; \\ d_j &= u_j - x_j, \text{ si } j \in J_{N^-}; \\ d_j &= \frac{-\Delta_j}{\eta}, \text{ si } j \in J_N^P, \eta \neq 0; \\ d_j &= 0, \text{ si } j \in J_N^P, \eta = 0; \\ d_B &= -A_B^{-1} A_N d_N; \end{aligned}$$

Étape 6 : Calculer $\theta_{j_1} = \min_{j \in J_B} \theta_j$, où

$$\theta_j = \begin{cases} \frac{u_j - x_j}{d_j}, & \text{si } d_j > 0; \\ \frac{l_j - x_j}{d_j}, & \text{si } d_j < 0; \\ +\infty, & \text{si } d_j = 0; \end{cases}$$

Étape 7 : Calculer $\theta^0 = \min\{1, \theta_{j_1}\}$, $\bar{x} = x + \theta^0 d$ et $\bar{z} = z + \theta^0 \gamma$;

Étape 8 : Si $\theta^0 = 1$, alors

Étape 8.1 : Si $J_N^{P^+} \cup J_N^{P^-} = \emptyset$, alors \bar{x} est optimale, Stop ;

Étape 8.2 : Sinon, poser $\eta = 0$, $x = \bar{x}$, $z = \bar{z}$ et aller à l'étape 2 ;

Étape 9 : Calculer la direction duale t :

$$\begin{aligned} \kappa &= x + d, \alpha_0 = \kappa_{j_1} - \bar{x}_{j_1}; \\ t_{j_1} &= -\text{sign}(\alpha_0), t_j = 0, j \neq j_1, j \in J_B; \\ t_N^T &= t_B^T A_B^{-1} A_N; \end{aligned}$$

Étape 10 : Calculer le pas dual $\sigma^0 = \sigma_{j_0} = \min_{j \in J_N} \sigma_j$, où

$$\sigma_j = \begin{cases} -\frac{\Delta_j}{t_j}, & \text{si } \Delta_j t_j < 0; \\ 0, & \text{si } \Delta_j = 0, t_j < 0, \kappa_j \neq u_j; \\ 0, & \text{si } \Delta_j = 0, t_j > 0, \kappa_j \neq l_j; \\ +\infty, & \text{dans les autres cas;} \end{cases} \quad j \in J_N$$

Étape 11 : Poser $\bar{\Delta} = \Delta + \sigma^0 t$, $\bar{J}_B = (J_B \setminus \{j_1\}) \cup \{j_0\}$;

Étape 12 : Poser $x = \bar{x}$, $J_B = \bar{J}_B$, $\Delta = \bar{\Delta}$, $z = \bar{z}$ et aller à l'étape 2 ;

Théorème 3.3. Si l'algorithme 3.1 s'arrête à l'étape 4, alors x est une solution réalisable optimale du problème (3.1)-(3.3). Si l'algorithme 3.1 se termine à l'étape 8.1, alors \bar{x} est une solution réalisable optimale du problème (3.1)-(3.3).

Démonstration. (L'optimalité de x). Soit $\{x, J_B\}$ une SRS du problème (3.1)-(3.3).

Si $\gamma(\eta, x, J_B) = 0$, alors

$$x_j = l_j, \text{ pour tout } j \in J_{N^+}, x_j = u_j, \text{ pour tout } j \in J_{N^-} \text{ et } J_N^{P^-} \cup J_N^{P^+} = \emptyset.$$

Pour toute solution réalisable y du problème (3.1)-(3.3), on a

$$\begin{aligned} c^T(y - x) &= \Delta^T(x - y) \\ &= \sum_{j \in J_N} \Delta_j(x_j - y_j) \\ &= \sum_{j \in J_N, \Delta_j > 0} \Delta_j(x_j - y_j) + \sum_{j \in J_N, \Delta_j < 0} \Delta_j(x_j - y_j) \\ &= \sum_{j \in J_N^{P^+} \cup J_{N^+}} \Delta_j(x_j - y_j) + \sum_{j \in J_N^{P^-} \cup J_{N^-}} \Delta_j(x_j - y_j) \\ &\leq \sum_{j \in J_N^{P^+} \cup J_{N^+}} \Delta_j(x_j - l_j) + \sum_{j \in J_N^{P^-} \cup J_{N^-}} \Delta_j(x_j - u_j) = 0. \end{aligned}$$

On en déduit alors que $c^T x \geq c^T y$. Par conséquent, x est optimale.

(L'optimalité de \bar{x}). Si $\theta^0 = 1$ et $J_N^{P^-} \cup J_N^{P^+} = \emptyset$, c'est-à-dire, $\bar{x} = x + d$ et $J_N^{P^-} \cup J_N^{P^+} = \emptyset$.

Alors pour toute solution réalisable y du problème (3.1)-(3.3), on a

$$\begin{aligned} c^T(\bar{x} - y) &= \sum_{j \in J_N^{P^+} \cup J_{N^+}} \Delta_j(y_j - \bar{x}_j) + \sum_{j \in J_N^{P^-} \cup J_{N^-}} \Delta_j(y_j - \bar{x}_j) \\ &= \sum_{j \in J_{N^+}} \Delta_j(y_j - x_j - d_j) + \sum_{j \in J_{N^-}} \Delta_j(y_j - x_j - d_j) \\ &= \sum_{j \in J_{N^+}} \Delta_j(y_j - l_j) + \sum_{j \in J_{N^-}} \Delta_j(y_j - u_j) \geq 0. \\ &\Rightarrow c^T \bar{x} \geq c^T y. \end{aligned}$$

Par conséquent, le vecteur \bar{x} est une solution optimale du problème (3.1)-(3.3). □

Remarque 3.1. Si $\eta = 0$, nous obtenons la méthode adaptée [36].

3.4.1 Exemples numériques

Exemple 3.1.

$$\begin{aligned} \max z &= c^T x, \\ Ax &= b, l \leq x \leq u, \end{aligned} \quad (3.12)$$

où

$$A = \begin{pmatrix} 1 & -12 & 3 & 0 \\ -2 & 2 & 0 & 1 \end{pmatrix}, b = \begin{pmatrix} 63 \\ 40 \end{pmatrix}, c = \begin{pmatrix} -6 \\ 3 \\ 0 \\ 0 \end{pmatrix}, l = \begin{pmatrix} -24 \\ -10 \\ -1 \\ -8 \end{pmatrix}, u = \begin{pmatrix} -18 \\ -2 \\ 10 \\ 28 \end{pmatrix}.$$

Posons $\eta = 1$ et commençons par la SRS initiale suivante :

$$J_B = \{3, 4\}, J_N = \{1, 2\}, x = (-21, -6, 4, 10)^T.$$

Itération 1

- La valeur de la fonction objectif en x est $z = 108$.
- Calcul du vecteur des multiplicateurs et du vecteur des coûts réduits :

$$\pi^T = c_B^T A_B^{-1} = (0, 0), \Delta^T = \pi^T A - c^T = (6, -3, 0, 0).$$

- Calcul des ensembles J_{N+} , J_{N-} , J_N^{P+} et J_N^{P-} :

$$\eta(x - u) = (-3, -4, -6, -18)^T, \eta(x - l) = (3, 4, 5, 18)^T,$$

$$J_{N+} = \{1\}, J_{N-} = \emptyset, J_N^{P+} = \emptyset, J_N^{P-} = \{2\}.$$

- Calcul de l'estimation d'optimalité : $\gamma(\eta, x, J_B) = 27$.
- Calcul de la direction d'amélioration d :

$$d = (-3, 3, 13, -12)^T.$$

- Calcul du pas le long de la direction d :

$$\theta^0 = \min\{1, \theta_3, \theta_4\} = \min\left\{1, \frac{6}{13}, \frac{3}{2}\right\} = \frac{6}{13} \Rightarrow j_1 = 3.$$

- Calcul de la nouvelle solution réalisable \bar{x} :

$$\bar{x} = x + \theta^0 d = \left(\frac{-291}{13}, \frac{-60}{13}, 10, \frac{58}{13}\right)^T, \bar{z} = z + \theta^0 \gamma(\eta, x, J_B) = \frac{1566}{13} \simeq 120.46.$$

- Calcul de la direction duale t :

$$\kappa = (-24, -3, 17, -2)^T, \alpha_0 = 7;$$

$$t = \left(\frac{-1}{3}, 4, -1, 0\right)^T.$$

- Calcul du pas dual et de l'indice entrant j_0 :

$$\sigma^0 = \min\{\sigma_1, \sigma_2\} = \min\{18, \frac{3}{4}\} \Rightarrow j_0 = 2, \sigma^0 = \frac{3}{4}.$$

$$\bar{J}_B = \{2, 4\}, \bar{J}_N = \{1, 3\}, \bar{\Delta} = \Delta + \sigma^0 t = \left(\frac{23}{4}, 0, \frac{-3}{4}, 0\right)^T.$$

Itération 2

- La SRS courante est

$$J_B = \{2, 4\}, J_N = \{1, 3\}, x = \left(\frac{-291}{13}, \frac{-60}{13}, 10, \frac{58}{13}\right)^T.$$

- La valeur de la fonction objectif en x est $z = \frac{1566}{13}$.
- Le vecteur des coûts réduits est

$$\Delta = \left(\frac{23}{4}, 0, \frac{-3}{4}, 0\right)^T.$$

- Les ensembles J_{N^+} , J_{N^-} , $J_N^{P^+}$ et $J_N^{P^-}$ sont

$$\eta(x - u) = \left(\frac{-57}{13}, \frac{-34}{13}, 0, \frac{-306}{13}\right)^T, \eta(x - l) = \left(\frac{21}{13}, \frac{70}{13}, 11, \frac{162}{13}\right)^T,$$

$$J_{N^+} = \{1\}, J_{N^-} = \{3\}, J_N^{P^+} = \emptyset, J_N^{P^-} = \emptyset.$$

- L'estimation d'optimalité est $\gamma(\eta, x, J_B) = \frac{483}{52}$.
- La direction d'amélioration est

$$d = \left(\frac{-21}{13}, \frac{-7}{52}, 0, \frac{-77}{26}\right)^T.$$

- Le pas le long de la direction d est :

$$\theta^0 = \min\{1, \theta_2, \theta_4\} = \min\{1, 40, \frac{324}{77}\} = 1.$$

Puisque $\theta^0 = 1$ et $J_N^{P^+} \cup J_N^{P^-} = \emptyset$, alors la solution optimale et l'optimum sont

$$\bar{x} = x + \theta^0 d = \left(-24, \frac{-19}{4}, 10, \frac{3}{2}\right)^T, \bar{z} = z + \theta^0 \gamma(\eta, x, J_B) = \frac{519}{4} \simeq 129.75.$$

Exemple 3.2.

$$\begin{aligned} \max z &= c^T x, \\ Ax &= b, \quad l \leq x \leq u, \end{aligned} \quad (3.13)$$

où

$$A = \begin{pmatrix} 0 & 0 & -1 & -2 & 1 & 0 & 0 \\ -2 & 0 & 0 & 3 & 0 & 1 & 0 \\ -1 & 0 & 2 & 0 & 0 & 0 & -1 \\ 0 & 6 & 0 & 4 & 0 & 0 & 0 \\ 1 & 0 & 2 & 3 & 0 & 0 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{3} \\ 21 \\ 24 \end{pmatrix}, \quad c = \begin{pmatrix} \frac{1}{3} \\ -\frac{1}{2} \\ -\frac{1}{6} \\ \frac{1}{6} \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$l = \begin{pmatrix} -2 \\ -\frac{1}{2} \\ 0 \\ 6 \\ 0 \\ -33 \\ 0 \end{pmatrix}, \quad u = \begin{pmatrix} 9 \\ 1 \\ 4 \\ 10 \\ 19 \\ 10 \\ 35 \end{pmatrix}.$$

Posons pour ce problème $\eta = 1$ et commençons pas la SRS initiale suivante :

$$J_B = \{5, 6, 7, 2, 1\}, \quad J_N = \{3, 4\}, \quad x = \left(\frac{8}{3}, \frac{-1}{2}, \frac{5}{3}, 6, \frac{41}{3}, \frac{-38}{3}, \frac{1}{3}\right)^T.$$

Itération 1

- La valeur de la fonction objectif en x est $z = \frac{67}{36} \simeq 1.86$.
- Calcul du vecteur des multiplicateurs et du vecteur des coûts réduits :

$$\pi^T = c_B^T A_B^{-1} = (0, 0, 0, \frac{-1}{12}, \frac{1}{3}), \quad \Delta^T = \pi^T A - c^T = (0, 0, \frac{5}{6}, \frac{1}{2}, 0, 0, 0).$$

- Calcul des ensembles J_{N^+} , J_{N^-} , $J_N^{P^+}$ et $J_N^{P^-}$:

$$\eta(x - u) = \left(\frac{-19}{3}, \frac{-3}{2}, \frac{-7}{3}, -4, \frac{16}{3}, \frac{-68}{3}, \frac{-69}{2}\right)^T, \quad \eta(x - l) = \left(\frac{14}{3}, 0, \frac{5}{3}, 0, \frac{41}{3}, \frac{61}{3}, \frac{1}{3}\right)^T;$$

$$J_{N^+} = \{4\}, \quad J_{N^-} = \emptyset, \quad J_N^{P^-} = \emptyset, \quad J_N^{P^+} = \{3\}.$$

- Calcul de l'estimation d'optimalité : $\gamma = \frac{25}{36}$.

- Calcul de la direction d'amélioration d :

$$d = \left(\frac{5}{3}, 0, \frac{-5}{6}, 0, \frac{-5}{6}, \frac{10}{3}, \frac{-10}{3}\right)^T.$$

- Calcul du pas le long de la direction d :

$$\theta^0 = \frac{1}{10} \text{ et } j_1 = 7.$$

- Calcul de la nouvelle solution réalisable \bar{x} :

$$\bar{x} = x + \theta^0 d = \left(\frac{17}{6}, \frac{-1}{2}, \frac{19}{12}, 6, \frac{163}{12}, \frac{-37}{3}, 0\right)^T, \quad \bar{z} = z + \theta^0 \gamma = \frac{139}{72} \simeq 1.93.$$

- Calcul de la direction duale t :

$$\kappa = \left(\frac{13}{3}, \frac{-1}{2}, \frac{5}{6}, 6, \frac{77}{6}, \frac{-28}{3}, -3\right)^T, \quad \alpha_0 = -3;$$

$$t = (0, 0, -4, -3, 0, 0, 1)^T.$$

- Calcul du pas dual et de l'indice entrant j_0 :

$$\sigma^0 = \frac{1}{6} \text{ et } j_0 = 4;$$

$$\bar{J}_B = \{5, 6, 4, 2, 1\}, \quad \bar{J}_N = \{3, 7\}, \quad \bar{\Delta} = \Delta + \sigma^0 t = (0, 0, \frac{1}{6}, 0, 0, 0, \frac{1}{6})^T.$$

Itération 2

- Calcul des ensembles J_{N^+} , J_{N^-} , $J_N^{P^+}$ et $J_N^{P^-}$:

$$\eta(x-u) = \left(\frac{-37}{3}, -3, \frac{-29}{6}, -8, \frac{-65}{6}, \frac{-134}{3}, -35\right)^T, \quad \eta(x-l) = \left(\frac{29}{6}, 0, \frac{19}{12}, 0, \frac{163}{12}, \frac{62}{3}, 0\right)^T;$$

$$J_{N^+} = \{7\}, \quad J_{N^-} = \emptyset, \quad J_N^{P^-} = \emptyset, \quad J_N^{P^+} = \{3\}.$$

- Calcul de l'estimation d'optimalité : $\gamma = \frac{1}{36}$.

- Calcul de la direction d'amélioration d :

$$d = \left(\frac{-1}{3}, \frac{-4}{27}, \frac{-1}{6}, \frac{2}{9}, \frac{5}{18}, \frac{-4}{3}, 0\right)^T.$$

- Calcul du pas le long de la direction d et de l'indice sortant j_1 :

$$\theta^0 = 0 \text{ et } j_1 = 2. \text{ Donc } \bar{x} = x, \quad \bar{z} = z.$$

- Calcul de la direction duale t :

$$\kappa = \left(\frac{5}{3}, \frac{-35}{54}, \frac{17}{12}, \frac{56}{9}, \frac{499}{18}, \frac{-41}{3}, 0 \right)^T, \quad \alpha_0 = \frac{-4}{27};$$

$$t = \left(0, 1, \frac{-8}{9}, 0, 0, 0, \frac{2}{9} \right)^T.$$

- Calcul du pas dual et de l'indice entrant j_0 :

$$\sigma^0 = \frac{3}{16} \text{ et } j_0 = 3;$$

$$\bar{J}_B = \{5, 6, 4, 3, 1\}, \quad \bar{J}_N = \{2, 7\}, \quad \bar{\Delta} = \Delta + \sigma^0 t = \left(0, \frac{3}{16}, 0, 0, 0, 0, \frac{5}{24} \right)^T.$$

Itération 3

- Calcul des ensembles J_{N+} , J_{N-} , J_N^{P+} et J_N^{P-} :

$$\eta(x-u) = \left(\frac{-37}{3}, -3, \frac{-29}{6}, -8, \frac{-65}{6}, \frac{-134}{3}, -35 \right)^T, \quad \eta(x-l) = \left(\frac{29}{6}, 0, \frac{19}{12}, 0, \frac{163}{12}, \frac{62}{3}, 0 \right)^T;$$

$$J_{N+} = \{2, 7\}, \quad J_{N-} = \emptyset, \quad J_N^{P+} = \emptyset, \quad J_N^{P-} = \emptyset.$$

- Calcul de l'estimation d'optimalité : $\gamma = 0$.

Donc la solution optimale et l'optimum du problème (3.13) sont

$$x = \left(\frac{17}{6}, \frac{-1}{2}, \frac{19}{12}, 6, \frac{163}{12}, \frac{-37}{3}, 0 \right)^T \text{ et } z = \frac{59}{24} \simeq 2.46.$$

3.5 Une itération du nouvel algorithme MDHPM

Soit $\{x, J_B\}$ une solution réalisable du problème (3.1)-(3.3) et $\eta > 0$. Nous introduisons les ensembles d'indices suivants :

$$\begin{aligned} J_{NE}^+ &= \{j \in J_N : \Delta_j > \eta(x_j - l_j) \text{ et } x_j > l_j\}, \\ J_{NE}^- &= \{j \in J_N : \Delta_j < \eta(x_j - u_j) \text{ et } x_j < u_j\}, \\ J_{NI}^+ &= \{j \in J_N : 0 < \Delta_j \leq \eta(x_j - l_j)\}, \\ J_{NI}^- &= \{j \in J_N : \eta(x_j - u_j) \leq \Delta_j < 0\}, \\ J_{NR}^+ &= \{j \in J_N : \Delta_j > 0 \text{ et } x_j = l_j\}, \\ J_{NR}^- &= \{j \in J_N : \Delta_j < 0 \text{ et } x_j = u_j\}, \\ J_{N0} &= \{j \in J_N : \Delta_j = 0\}, \\ J_{NE} &= J_{NE}^+ \cup J_{NE}^-, \quad J_{NI} = J_{NI}^+ \cup J_{NI}^-, \quad J_{NR} = J_{N0} \cup J_{NR}^+ \cup J_{NR}^-. \end{aligned} \tag{3.14}$$

Alors

$$J_N = J_{NE} \cup J_{NI} \cup J_{NR}.$$

Lemme 3.1. Si $\{x, J_B\}$ n'est pas optimale, alors $J_{NE} \cup J_{NI} \neq \emptyset$.

Démonstration. Nous supposons qu'il existe un indice $j_* \in J_N$ qui ne vérifie pas les conditions d'optimalité (3.7), alors deux cas peuvent se présenter :

cas1 : Si $\Delta_{j_*} < 0$ et $x_{j_*} < u_{j_*}$, alors $j_* \in J_{NE}^- \cup J_{NI}^-$. Par conséquent, $J_{NE} \cup J_{NI} \neq \emptyset$.

cas2 : Si $\Delta_{j_*} > 0$ et $x_{j_*} > l_{j_*}$, alors $j_* \in J_{NE}^+ \cup J_{NI}^+$. Par conséquent, $J_{NE} \cup J_{NI} \neq \emptyset$.

□

Pour $\eta > 0$, définissons les quantités γ et μ comme suit :

$$\gamma = \sum_{j \in J_{NI}^+} \Delta_j(x_j - l_j) + \sum_{j \in J_{NI}^-} \Delta_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \Delta_j^2, \quad (3.15)$$

$$\mu = - \sum_{j \in J_{NE}^+} \Delta_j(x_j - l_j) - \sum_{j \in J_{NE}^-} \Delta_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \Delta_j^2. \quad (3.16)$$

Lemme 3.2. Pour $\eta > 0$, les inégalités suivantes sont vraies :

$$\beta = \gamma - \mu \leq \gamma, \quad \gamma \geq 0, \quad \mu \geq 0. \quad (3.17)$$

Démonstration. Tout d'abord, nous remarquons que

$$\begin{aligned} \beta &= \sum_{j \in J_N^+} \Delta_j(x_j - l_j) + \sum_{j \in J_N^-} \Delta_j(x_j - u_j) \\ &= \sum_{j \in J_{NE}^+ \cup J_{NI}^+} \Delta_j(x_j - l_j) + \sum_{j \in J_{NE}^- \cup J_{NI}^-} \Delta_j(x_j - u_j). \end{aligned}$$

Alors

$$\beta + \mu = \sum_{j \in J_{NI}^+} \Delta_j(x_j - l_j) + \sum_{j \in J_{NI}^-} \Delta_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \Delta_j^2 = \gamma.$$

De plus, pour $\eta > 0$, nous avons

$$\begin{aligned} \beta &= \sum_{j \in J_{NE}^+} \Delta_j(x_j - l_j) + \sum_{j \in J_{NE}^-} \Delta_j(x_j - u_j) \\ &+ \sum_{j \in J_{NI}^+} \Delta_j(x_j - l_j) + \sum_{j \in J_{NI}^-} \Delta_j(x_j - u_j) \\ &\leq \sum_{j \in J_{NI}^+} \Delta_j(x_j - l_j) + \sum_{j \in J_{NI}^-} \Delta_j(x_j - u_j) \\ &+ \sum_{j \in J_{NE}^+} \frac{\Delta_j^2}{\eta} + \sum_{j \in J_{NE}^-} \frac{\Delta_j^2}{\eta}. \end{aligned}$$

En effet,

$$\text{Pour } j \in J_{NE}^+ : \eta(x_j - l_j) < \Delta_j \Rightarrow x_j - l_j < \frac{\Delta_j}{\eta} \Rightarrow \Delta_j(x_j - l_j) < \frac{\Delta_j^2}{\eta}.$$

$$\text{Pour } j \in J_{NE}^- : \eta(x_j - u_j) > \Delta_j \Rightarrow x_j - u_j > \frac{\Delta_j}{\eta} \Rightarrow \Delta_j(x_j - u_j) < \frac{\Delta_j^2}{\eta}.$$

Par conséquent,

$$0 \leq \beta \leq \sum_{j \in J_{NI}^+} \Delta_j(x_j - l_j) + \sum_{j \in J_{NI}^-} \Delta_j(x_j - u_j) + \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \frac{\Delta_j^2}{\eta} = \gamma.$$

Puisque $\gamma \geq \beta$ et $\beta + \mu = \gamma$, nous déduisons que $\mu = \gamma - \beta \geq 0$. \square

Remarque 3.2. Puisque $\beta \leq \gamma$, la condition $\gamma \leq \epsilon$, où ϵ est un nombre non négatif, est suffisante pour la suboptimalité de la solution réalisable x .

3.6 Changement de la solution réalisable

Soit $\{x, J_B\}$ une SRS du problème (3.1)-(3.3) et $\eta > 0$. On construit alors une nouvelle solution réalisable \bar{x} :

$$\bar{x} = x + \theta^0 d,$$

où d est une direction définie comme suit :

$$\begin{aligned} d_j &= l_j - x_j, \text{ si } j \in J_{NI}^+; d_j = u_j - x_j, \text{ si } j \in J_{NI}^-; \\ d_j &= \frac{-\Delta_j}{\eta}, \text{ si } j \in J_{NE}^+ \cup J_{NE}^-; d_j = 0, \text{ si } j \in J_{NR}; \\ d_B &= -A_B^{-1} A_N d_N. \end{aligned} \quad (3.18)$$

Le nombre θ^0 est le pas le long de cette direction. Il se calcule de façon à ce que les contraintes de bornes sur le vecteur \bar{x} soient vérifiées, c'est-à-dire,

$$l \leq x + \theta^0 d \leq u \Leftrightarrow \begin{cases} l_j - x_j \leq \theta^0 d_j \leq u_j - x_j, & \text{pour } j \in J_B; \\ l_j - x_j \leq \theta^0 d_j \leq u_j - x_j, & \text{pour } j \in J_N. \end{cases}$$

Donc θ^0 est calculé comme suit :

$$\theta^0 = \min\{1, \theta_{j_1}, \theta_{j_2}\}, \theta_{j_1} = \min_{j \in J_B} \theta_j, \theta_{j_2} = \begin{cases} \min_{j \in J_{NE}} \theta_j, & \text{si } J_{NE} \neq \emptyset; \\ \infty, & \text{sinon,} \end{cases} \quad (3.19)$$

où

$$\theta_j = \begin{cases} \frac{u_j - x_j}{d_j}, & \text{si } d_j > 0; \\ \frac{l_j - x_j}{d_j}, & \text{si } d_j < 0; \\ \infty, & \text{sinon,} \end{cases} \quad (3.20)$$

et le pas 1 correspond aux indices de l'ensemble J_{NI} .

Puisque $Ad = 0$, alors le vecteur d est une direction admissible. De plus, on a $z(\bar{x}) \geq z(x)$, c'est-à-dire que d est une direction d'amélioration. En effet, l'accroissement de la fonction objectif est donné par

$$\begin{aligned} \bar{z} - z &= c^T \bar{x} - c^T x = c^T (\bar{x} - x) \\ &= \theta^0 c^T d \\ &= \theta^0 (c_N^T d_N + c_B^T d_B) \\ &= \theta^0 (c_N^T d_N - c_B^T A_B^{-1} A_N d_N) \\ &= -\theta^0 \Delta_N^T d_N \\ &= \theta^0 \left[- \sum_{j \in J_{NI}^+} \Delta_j d_j - \sum_{j \in J_{NI}^-} \Delta_j d_j - \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \Delta_j d_j \right] \\ &= \theta^0 \left[\sum_{j \in J_{NI}^+} \Delta_j (x_j - l_j) + \sum_{j \in J_{NI}^-} \Delta_j (x_j - u_j) + \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \frac{\Delta_j^2}{\eta} \right]. \end{aligned}$$

Par conséquent,

$$\bar{z} - z = \theta^0 \gamma = \theta^0 (\beta + \mu) \geq 0. \quad (3.21)$$

Lemme 3.3. Pour $\eta > 0$, la quantité $\bar{\beta} = \beta(\bar{x}, J_B)$ peut être calculée comme suit :

$$\bar{\beta} = (1 - \theta^0) \beta - \theta^0 \mu \leq \beta. \quad (3.22)$$

Démonstration. Pour $\eta > 0$, nous avons

$$\begin{aligned}
\bar{\beta} &= \sum_{j \in J_N^+} \Delta_j(\bar{x}_j - l_j) + \sum_{j \in J_N^-} \Delta_j(\bar{x}_j - u_j) \\
&= \sum_{j \in J_N^+} \Delta_j(x_j - l_j) + \sum_{j \in J_N^-} \Delta_j(x_j - u_j) + \theta^0 \sum_{j \in J_N^+} \Delta_j d_j + \theta^0 \sum_{j \in J_N^-} \Delta_j d_j \\
&= \beta + \theta^0 \left(\sum_{j \in J_{NE}^+ \cup J_{NI}^+} \Delta_j d_j + \sum_{j \in J_{NE}^- \cup J_{NI}^-} \Delta_j d_j \right) \\
&= \beta + \theta^0 \left(\sum_{j \in J_{NI}^+} \Delta_j d_j + \sum_{j \in J_{NI}^-} \Delta_j d_j + \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \Delta_j d_j \right) \\
&= \beta - \theta^0 \left(\sum_{j \in J_{NI}^+} \Delta_j(x_j - l_j) + \sum_{j \in J_{NI}^-} \Delta_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \Delta_j^2 \right) \\
&= \beta - \theta^0 \gamma \leq \beta.
\end{aligned}$$

Puisque nous avons $\gamma = \beta + \mu$, nous trouvons

$$\bar{\beta} = (1 - \theta^0)\beta - \theta^0\mu. \quad (3.23)$$

□

Proposition 3.1. (Condition suffisante pour l'optimalité de \bar{x})

Si $\theta^0 = 1$, alors \bar{x} est une solution optimale pour le problème (3.1)-(3.3).

Démonstration. Tout d'abord, nous prouvons que si $\theta^0 = 1$, alors $J_{NE}^+ \cup J_{NE}^- = \emptyset$. Supposons que $\theta^0 = 1$ et $J_{NE}^+ \cup J_{NE}^- \neq \emptyset$, c'est-à-dire $\exists j_* \in J_{NE}^+ \cup J_{NE}^-$.

Si $j_* \in J_{NE}^+$, alors $d_{j_*} = \frac{-\Delta_{j_*}}{\eta} < 0$ et $\eta(x_{j_*} - l_{j_*}) < \Delta_{j_*}$. Donc $\theta_{j_*} = \frac{\eta(x_{j_*} - l_{j_*})}{\Delta_{j_*}} < 1$.

Si $j_* \in J_{NE}^-$, alors $d_{j_*} = \frac{-\Delta_{j_*}}{\eta} > 0$ et $\eta(x_{j_*} - u_{j_*}) > \Delta_{j_*}$. Donc $\theta_{j_*} = \frac{\eta(x_{j_*} - u_{j_*})}{\Delta_{j_*}} < 1$.

Ainsi, nous déduisons que $\theta_{j_*} < 1 \Rightarrow \theta^0 < 1$, ce qui conduit à une contradiction. Par conséquent,

$$\theta^0 = 1 \Rightarrow J_{NE}^+ \cup J_{NE}^- = \emptyset \Rightarrow \mu = 0 \Rightarrow \bar{\beta} = 0 \Rightarrow \bar{x} \text{ est optimale.}$$

□

Si $\bar{\beta} \leq \epsilon$, alors \bar{x} est une solution ϵ -optimale. Si $\theta^0 = \theta_{j_2}$, alors on pose $x := \bar{x}$ et nous commençons une nouvelle itération avec le SRS $\{x, J_B\}$. Sinon ($\theta^0 = \theta_{j_1} < 1$), on change le support J_B .

Remarque 3.3. Lorsque $\mu = 0$, nous trouvons la formule de mise à jour classique de

$\beta(x, J_B)$ [32, 36] :

$$\beta(\bar{x}, J_B) = (1 - \theta^0)\beta(x, J_B).$$

3.7 Changement de support

3.7.1 Règle du pas simple

Nous définissons le n -vecteur κ et le nombre réel α_0 comme suit :

$$\kappa = x + d \text{ et } \alpha_0 = \kappa_{j_1} - \bar{x}_{j_1},$$

où j_1 est l'indice vérifiant la relation (3.19). Donc, la direction duale est donnée par

$$t_{j_1} = -\text{sign}(\alpha_0), t_j = 0, j \neq j_1, j \in J_B; t_N^T = t_B^T A_B^{-1} A_N. \quad (3.24)$$

Remarque 3.4. Nous avons $\alpha_0 = \kappa_{j_1} - \bar{x}_{j_1} = x_{j_1} + d_{j_1} - x_{j_1} - \theta^0 d_{j_1} = (1 - \theta^0)d_{j_1}$. Puisque $0 \leq \theta^0 < 1$, alors $t_{j_1} = -\text{sign}(\alpha_0) = -\text{sign}(d_{j_1})$.

Remarque 3.5. La direction duale t et la direction d'amélioration d sont orthogonales. En effet,

$$t^T d = t_B^T d_B + t_N^T d_N = t_B^T (-A_B^{-1} A_N d_N) + (t_B^T A_B^{-1} A_N) d_N = 0.$$

Définissons les ensembles suivants :

$$J_{N_0}^+ = \{j \in J_{N_0} : t_j > 0\} \text{ et } J_{N_0}^- = \{j \in J_{N_0} : t_j < 0\} \quad (3.25)$$

et la quantité :

$$\alpha = -|\alpha_0| + \sum_{j \in J_{N_0}^+ \cup J_{NE}^+} t_j (\kappa_j - l_j) + \sum_{j \in J_{N_0}^- \cup J_{NE}^-} t_j (\kappa_j - u_j) \quad (3.26)$$

Le nouveau vecteur des coûts réduits et le nouveau support sont calculés comme suit :

$$\bar{\Delta} = \Delta + \sigma^0 t \text{ et } \bar{J}_B = (J_B \setminus \{j_1\}) \cup \{j_0\},$$

où

$$\sigma^0 = \sigma_{j_0} = \min_{j \in J_N} \sigma_j, \sigma_j = \begin{cases} \frac{-\Delta_j}{t_j}, & \text{si } \Delta_j t_j < 0; \\ 0, & \text{si } j \in J_{N_0}^+ \text{ et } \kappa_j \neq l_j \\ 0, & \text{si } j \in J_{N_0}^- \text{ et } \kappa_j \neq u_j \\ \infty, & \text{sinon.} \end{cases} \quad (3.27)$$

L'estimation de suboptimalité correspondant à la nouvelle solution réalisable et au nouveau

support est donnée par

$$\bar{\beta} = \sum_{j \in \bar{J}_N, \bar{\Delta}_j > 0} \bar{\Delta}_j (\bar{x}_j - l_j) + \sum_{j \in \bar{J}_N, \bar{\Delta}_j < 0} \bar{\Delta}_j (\bar{x}_j - u_j).$$

Proposition 3.2. L'estimation de suboptimalité $\bar{\beta}$ peut s'écrire comme suit

$$\bar{\beta} = \bar{\beta} + \sigma^0 \alpha.$$

Démonstration. Nous avons

$$\bar{J}_N = (J_N \setminus \{j_0\}) \cup \{j_1\}.$$

Si $\sigma^0 = 0$, alors

$$j_0 \in J_{N0}^+ \text{ ou } j_0 \in J_{N0}^- \Rightarrow \Delta_{j_0} = 0 \Rightarrow \bar{\Delta}_{j_0} = 0.$$

Si $\sigma^0 > 0$, alors $\bar{\Delta}_{j_0} = \Delta_{j_0} + \sigma^0 t_{j_0} = \Delta_{j_0} - \frac{\Delta_{j_0}}{t_{j_0}} t_{j_0} = 0$. Donc

$$\bar{\beta} = \begin{cases} \sum_{j \in J_N, \bar{\Delta}_j > 0} \bar{\Delta}_j (\bar{x}_j - l_j) + \sum_{j \in J_N, \bar{\Delta}_j < 0} \bar{\Delta}_j (\bar{x}_j - u_j) + \bar{\Delta}_{j_1} (\bar{x}_{j_1} - l_{j_1}), & \text{si } \bar{\Delta}_{j_1} > 0; \\ \sum_{j \in J_N, \bar{\Delta}_j > 0} \bar{\Delta}_j (\bar{x}_j - l_j) + \sum_{j \in J_N, \bar{\Delta}_j < 0} \bar{\Delta}_j (\bar{x}_j - u_j) + \bar{\Delta}_{j_1} (\bar{x}_{j_1} - u_{j_1}), & \text{si } \bar{\Delta}_{j_1} < 0. \end{cases}$$

Puisque $\Delta_{j_1} = 0$, ($j_1 \in J_B$), deux cas peuvent se produire :

- Si $\bar{\Delta}_{j_1} > 0$, alors $t_{j_1} > 0 \Rightarrow d_{j_1} < 0$. Par conséquent,

$$\bar{x}_{j_1} = x_{j_1} + \theta^0 d_{j_1} = x_{j_1} + \frac{l_{j_1} - x_{j_1}}{d_{j_1}} d_{j_1} = l_{j_1} \Rightarrow \bar{\Delta}_{j_1} (\bar{x}_{j_1} - l_{j_1}) = 0.$$

- Si $\bar{\Delta}_{j_1} < 0$, alors $t_{j_1} < 0 \Rightarrow d_{j_1} > 0$. Par conséquent,

$$\bar{x}_{j_1} = x_{j_1} + \theta^0 d_{j_1} = x_{j_1} + \frac{u_{j_1} - x_{j_1}}{d_{j_1}} d_{j_1} = u_{j_1} \Rightarrow \bar{\Delta}_{j_1} (\bar{x}_{j_1} - u_{j_1}) = 0.$$

Puisque σ^0 est choisi de telle manière que Δ_j et $\bar{\Delta}_j$ gardent le même signe, on en déduit que

$$\bar{\Delta}_j > 0 \Leftrightarrow [(\Delta_j > 0) \text{ ou } (\Delta_j = 0 \text{ et } t_j > 0)],$$

et

$$\bar{\Delta}_j < 0 \Leftrightarrow [(\Delta_j < 0) \text{ ou } (\Delta_j = 0 \text{ et } t_j < 0)].$$

Par conséquent,

$$\begin{aligned}
\bar{\beta} &= \sum_{j \in J_N, \bar{\Delta}_j > 0} \bar{\Delta}_j(\bar{x}_j - l_j) + \sum_{j \in J_N, \bar{\Delta}_j < 0} \bar{\Delta}_j(\bar{x}_j - u_j) \\
&= \sum_{j \in J_N, \Delta_j > 0} (\Delta_j + \sigma^0 t_j)(\bar{x}_j - l_j) + \sum_{j \in J_N, \Delta_j < 0} (\Delta_j + \sigma^0 t_j)(\bar{x}_j - u_j) \\
&+ \sum_{j \in J_N, \Delta_j = 0, t_j > 0} \sigma^0 t_j(\bar{x}_j - l_j) + \sum_{j \in J_N, \Delta_j = 0, t_j < 0} \sigma^0 t_j(\bar{x}_j - u_j) \\
&= \left(\sum_{j \in J_N, \Delta_j > 0} \Delta_j(\bar{x}_j - l_j) + \sum_{j \in J_N, \Delta_j < 0} \Delta_j(\bar{x}_j - u_j) \right) \\
&+ \sigma^0 \left(\sum_{j \in J_N, \Delta_j = 0, t_j > 0} t_j(\bar{x}_j - l_j) + \sum_{j \in J_N, \Delta_j = 0, t_j < 0} t_j(\bar{x}_j - u_j) \right) \\
&+ \sigma^0 \left(\sum_{j \in J_N, \Delta_j > 0} t_j(\bar{x}_j - l_j) + \sum_{j \in J_N, \Delta_j < 0} t_j(\bar{x}_j - u_j) \right) \\
&= \bar{\beta} + \sigma^0(\alpha_1 + \alpha_2),
\end{aligned}$$

où

$$\begin{aligned}
\alpha_1 &= \sum_{j \in J_{N0}^+} t_j(\bar{x}_j - l_j) + \sum_{j \in J_{N0}^-} t_j(\bar{x}_j - u_j), \\
\alpha_2 &= \sum_{j \in J_{NE}^+ \cup J_{NI}^+ \cup J_{NR}^+} t_j(\bar{x}_j - l_j) + \sum_{j \in J_{NE}^- \cup J_{NI}^- \cup J_{NR}^-} t_j(\bar{x}_j - u_j).
\end{aligned}$$

Pour $j \in J_{N0}^+ \cup J_{N0}^- \subseteq J_{NR}$, $d_j = 0 \Rightarrow \bar{x}_j = x_j = \kappa_j$. Alors

$$\alpha_1 = \sum_{j \in J_{N0}^+} t_j(\kappa_j - l_j) + \sum_{j \in J_{N0}^-} t_j(\kappa_j - u_j).$$

Maintenant, calculons α_2 :

Pour $j \in J_{NR}^+$, $d_j = 0$ et $x_j = l_j \Rightarrow \bar{x}_j = l_j \Rightarrow \sum_{j \in J_{NR}^+} t_j(\bar{x}_j - l_j) = 0$.

Pour $j \in J_{NR}^-$, $d_j = 0$ et $x_j = u_j \Rightarrow \bar{x}_j = u_j \Rightarrow \sum_{j \in J_{NR}^-} t_j(\bar{x}_j - u_j) = 0$. Par conséquent,

$$\begin{aligned} \alpha_2 &= \sum_{j \in J_{NE}^+ \cup J_{NI}^+} t_j(\bar{x}_j - l_j) + \sum_{j \in J_{NE}^- \cup J_{NI}^-} t_j(\bar{x}_j - u_j) \\ &= \sum_{j \in J_{NE}^+} t_j(\bar{x}_j - l_j) + \sum_{j \in J_{NE}^-} t_j(\bar{x}_j - u_j) + \sum_{j \in J_{NI}^+} t_j(x_j - l_j) + \sum_{j \in J_{NI}^-} t_j(x_j - u_j) \\ &\quad + \theta^0 \left(\sum_{j \in J_{NI}^+} t_j d_j + \sum_{j \in J_{NI}^-} t_j d_j \right). \end{aligned}$$

Puisque $x_j - l_j = -d_j$ pour $j \in J_{NI}^+$ et $x_j - u_j = -d_j$ pour $j \in J_{NI}^-$, on aura

$$\alpha_2 = \sum_{j \in J_{NE}^+} t_j(\bar{x}_j - l_j) + \sum_{j \in J_{NE}^-} t_j(\bar{x}_j - u_j) - (1 - \theta^0) \left(\sum_{j \in J_{NI}^+} t_j d_j + \sum_{j \in J_{NI}^-} t_j d_j \right).$$

D'autre part, nous avons

$$t^T d = \sum_{j \in J_B} t_j d_j + \sum_{j \in J_N} t_j d_j = \sum_{j \in J_B} t_j d_j + \sum_{j \in J_{NE}^+} t_j d_j + \sum_{j \in J_{NE}^-} t_j d_j + \sum_{j \in J_{NR}} t_j d_j + \sum_{j \in J_{NI}^+} t_j d_j + \sum_{j \in J_{NI}^-} t_j d_j.$$

Puisque $\sum_{j \in J_B} t_j d_j = t_{j_1} d_{j_1} = -\text{sign}(d_{j_1}) d_{j_1} = -|d_{j_1}|$, $t^T d = 0$, $d_j = 0$, $j \in J_{NR}$, alors

$$\sum_{j \in J_{NI}^+} t_j d_j + \sum_{j \in J_{NI}^-} t_j d_j = - \sum_{j \in J_{NE}^+ \cup J_{NE}^-} t_j d_j + |d_{j_1}|.$$

Ainsi, la quantité α_2 devient

$$\begin{aligned} \alpha_2 &= \sum_{j \in J_{NE}^+} t_j(\bar{x}_j - l_j) + \sum_{j \in J_{NE}^-} t_j(\bar{x}_j - u_j) - (1 - \theta^0)|d_{j_1}| + (1 - \theta^0) \sum_{j \in J_{NE}^+ \cup J_{NE}^-} t_j d_j \\ &= -(1 - \theta^0)|d_{j_1}| + \sum_{j \in J_{NE}^+} t_j[\bar{x}_j - l_j + (1 - \theta^0)d_j] + \sum_{j \in J_{NE}^-} t_j[\bar{x}_j - u_j + (1 - \theta^0)d_j]. \end{aligned}$$

Puisque $(1 - \theta^0)|d_{j_1}| = |\alpha_0|$, $\bar{x}_j = x_j + \theta^0 d_j$, $\kappa_j = x_j + d_j$, alors on obtient

$$\alpha_2 = -|\alpha_0| + \sum_{j \in J_{NE}^+} t_j(\kappa_j - l_j) + \sum_{j \in J_{NE}^-} t_j(\kappa_j - u_j).$$

Donc,

$$\alpha_1 + \alpha_2 = -|\alpha_0| + \sum_{j \in J_{N0}^+ \cup J_{NE}^+} t_j(\kappa_j - l_j) + \sum_{j \in J_{N0}^- \cup J_{NE}^-} t_j(\kappa_j - u_j) = \alpha,$$

et

$$\bar{\beta} = \bar{\beta} + \sigma^0 \alpha.$$

□

Remarque 3.6. Si $J_{NE} = \emptyset$, alors

$$\alpha = -|\alpha_0| + \sum_{j \in J_{N_0}^+} t_j(\kappa_j - l_j) + \sum_{j \in J_{N_0}^-} t_j(\kappa_j - u_j).$$

Nous trouvons donc la formule de mise à jour de $\beta(\bar{x}, \bar{J}_B)$ [32, 36].

Si $\alpha \leq 0$, alors nous modifions le support J_B avec la règle du pas multiple décrite ci-dessous dans la Procédure 3.2, sinon deux cas peuvent se présenter :

- (a) Si $J_{N_0}^+ \cup J_{N_0}^- \neq \emptyset$, alors choisir un indice $j_0 \in J_{N_0}^+ \cup J_{N_0}^-$, poser $\sigma^0 = \sigma_{j_0} = 0$, $\bar{J}_B = (J_B \setminus \{j_1\}) \cup \{j_0\}$, et commencer une nouvelle itération avec le SRS $\{\bar{x}, \bar{J}_B\}$.
- (b) Si $J_{N_0}^+ \cup J_{N_0}^- = \emptyset$, alors calculer la valeur $\bar{\eta}$ avec la procédure suivante et commencer une nouvelle itération avec $\eta := \bar{\eta}$ et la SRS $\{\bar{x}, J_B\}$.

Procédure 3.1. (Procédure de mise à jour de la valeur de η)

- (1) Calculer les ensembles :

$$\bar{J}_{NE}^+ = \{j \in J_N : \Delta_j > \eta(\bar{x}_j - l_j) \text{ et } \bar{x}_j > l_j\},$$

$$\bar{J}_{NE}^- = \{j \in J_N : \Delta_j < \eta(\bar{x}_j - u_j) \text{ et } \bar{x}_j < u_j\};$$

- (2) Si $\bar{J}_{NE}^+ \cup \bar{J}_{NE}^- = \emptyset$, alors poser $\bar{\eta} = \eta$, sinon, passer à l'étape (3) ;
- (3) Si $\bar{J}_{NE}^+ \neq \emptyset$, alors poser $\eta_0 = \max_{j \in \bar{J}_{NE}^+} \frac{\Delta_j}{\bar{x}_j - l_j}$, sinon poser $\eta_0 = 0$;
- (4) Si $\bar{J}_{NE}^- \neq \emptyset$, alors poser $\eta_1 = \max_{j \in \bar{J}_{NE}^-} \frac{\Delta_j}{\bar{x}_j - u_j}$, sinon poser $\eta_1 = 0$;
- (5) Poser $\bar{\eta} = \max\{\eta_0, \eta_1\}$.

3.7.2 Règle du pas multiple

Nous pouvons modifier la procédure présentée dans [36], appelée règle du pas multiple, afin de calculer le pas le long de la direction duale dans notre méthode. Cette procédure est décrite dans les étapes suivantes :

Procédure 3.2. (Règle du pas multiple)

- (1) Calculer $\sigma^0 = \min_{j \in J_N} \sigma_j$, où les σ_j sont calculés avec les relations (3.27) ;
- (2) Ordonner les indices $\{j \in J_N : \sigma_j \neq \infty\}$ par ordre croissant par rapport aux σ_j :

$$\sigma_{i_1} \leq \sigma_{i_2} \leq \dots \leq \sigma_{i_p}; \quad i_k \in J_N, \quad \sigma_{i_k} \neq \infty, \quad k = 1, \dots, p;$$

- (3) Si $p = 1$, alors $i_q = i_1$; poser $V_0 = \alpha$ et aller à l'étape (8);
 (4) Pour tout i_k , $k = 1, \dots, p$, calculer $\Delta V_{i_k} = |t_{i_k}|(u_{i_k} - l_{i_k})$;
 (5) Calculer V_{i_k} , $k = 0, \dots, p$, où

$$\begin{cases} V_{i_0} = V_0 = \alpha; \\ V_{i_k} = V_0 + \sum_{s=1}^k \Delta V_{i_s} = V_{i_{k-1}} + \Delta V_{i_k}, \quad k = 1, \dots, p; \end{cases}$$

- (6) Choisir l'indice i_q tel que $V_{i_{q-1}} < 0$ et $V_{i_q} \geq 0$;
 (7) Soit j_1 l'indice calculé par (3.19). Poser $\bar{J}_B = (J_B \setminus \{j_1\}) \cup \{i_q\}$, $\sigma^0 = \sigma_{i_q}$ et $\bar{\Delta} = \Delta + \sigma^0 t$;
 (8) Calculer

$$\bar{\beta} = \begin{cases} \bar{\beta} + \sigma_{i_1} V_0, & \text{si } q = 1; \\ \bar{\beta} + \sigma_{i_1} V_0 + \sum_{k=2}^q (\sigma_{i_k} - \sigma_{i_{k-1}}) V_{i_{k-1}}, & \text{si } q \geq 2. \end{cases}$$

3.8 Le schéma de la méthode à direction hybride et à pas multiple

Soient $\{x, J_B\}$ une SRS initiale du problème (3.1)-(3.3), ϵ un nombre arbitraire positif ou nul, et $\eta > 0$. Le schéma de l'algorithme de la méthode à direction hybride et à pas multiple (MDHPM) est décrit dans les étapes de l'algorithme 3.2.

Algorithme 3.2. (MDHPM)

- Étape 1 :** Calculer $\pi^T = c_B^T A_B^{-1}$, $\Delta^T = \pi^T A - c^T$;
Étape 2 : Calculer l'estimation de suboptimalité β avec (3.11);
Étape 3 : Si $\beta = 0$, alors l'algorithme s'arrête avec $\{x, J_B\}$, une SRS optimale.
Étape 4 : Si $\beta \leq \epsilon$, alors l'algorithme s'arrête avec $\{x, J_B\}$, une SRS ϵ -optimale.
Étape 5 : Calculer les vecteurs $\eta(x - u)$ et $\eta(x - l)$;
Étape 6 : Calculer les ensembles J_{NI}^+ , J_{NI}^- , J_{NE}^+ et J_{NE}^- avec (3.14);
Étape 7 : Calculer μ avec (3.16);
Étape 8 : Calculer la direction d'amélioration d avec (3.18);
Étape 9 : Calculer θ_{j_1} , θ_{j_2} et θ^0 avec (3.19);
Étape 10 : Calculer $\bar{x} = x + \theta^0 d$ et $\bar{z} = z + \theta^0(\beta + \mu)$;
Étape 11 : Si $\theta^0 = 1$, alors l'algorithme s'arrête avec \bar{x} une solution optimale.
Étape 12 : Calculer $\bar{\beta}$ avec (3.23); si $\bar{\beta} = 0$, alors l'algorithme s'arrête avec $\{\bar{x}, J_B\}$, une SRS optimale. Si $\bar{\beta} \leq \epsilon$, alors l'algorithme s'arrête avec $\{\bar{x}, J_B\}$, une SRS ϵ -optimale.

Étape 13 : Si $\theta^0 = \theta_{j_2}$, poser $x = \bar{x}$, $z = \bar{z}$, $\beta = \bar{\beta}$ et aller à l'étape 5.

Étape 14 : Calculer $\kappa = x + d$ et $\alpha_0 = \kappa_{j_1} - \bar{x}_{j_1}$;

Étape 15 : Calculer la direction duale t avec (3.24) ;

Étape 16 : Calculer J_{N_0} avec (3.10), $J_{N_0}^+$ et $J_{N_0}^-$ avec (3.25) ;

Étape 17 : Calculer α avec (3.26) ; si $\alpha \leq 0$, alors passer à l'étape 18, sinon

(a) Si $J_{N_0}^+ \cup J_{N_0}^- \neq \emptyset$, alors choisir un index $j_0 \in J_{N_0}^+ \cup J_{N_0}^-$, poser $\bar{\Delta} = \Delta$, $\bar{J}_B = (J_B \setminus \{j_1\}) \cup \{j_0\}$, $\bar{\beta} = \bar{\beta}$, et aller à l'étape 19 ;

(b) Si $J_{N_0}^+ \cup J_{N_0}^- = \emptyset$, alors calculer la valeur $\bar{\eta}$ avec la procédure 3.1 ; poser $\eta := \bar{\eta}$, $x := \bar{x}$, $z := \bar{z}$, $\beta := \bar{\beta}$ et aller à l'étape 5.

Étape 18 : Calculer le nouveau support \bar{J}_B , le nouveau vecteur des coût réduits $\bar{\Delta}$ et la nouvelle estimation de suboptimalité $\bar{\beta}$ avec le règle du pas multiple (Procédure 3.2) ;

Étape 19 : Poser $x := \bar{x}$, $z := \bar{z}$, $J_B := \bar{J}_B$, $\Delta := \bar{\Delta}$, $\beta := \bar{\beta}$ et aller à l'étape 3.

Remarque 3.7. Comme l'estimation de suboptimalité décroît à chaque itération de l'algorithme MDHPM et la condition $\beta(x, J_B) = 0$ est une condition nécessaire et suffisante pour l'optimalité d'une SFS non-dégénérée $\{x, J_B\}$.

Remarque 3.8. Si le nombre de solutions dégénérées est fini, alors notre algorithme converge en un nombre fini d'itérations vers une solution optimale du problème (3.1)-(3.3).

Remarque 3.9. Si $\eta \rightarrow +\infty$, alors $J_{NE}^+ = J_{NE}^- = \emptyset$, $\mu = 0$ et $\beta = \gamma$, donc nous obtenons la méthode adaptée [36].

3.9 Exemple numérique

Résolvons le problème de PL suivant en utilisant MDHPM :

$$\begin{aligned} \max z &= c^T x, \\ Ax &= b, \quad l \leq x \leq u, \end{aligned} \tag{3.28}$$

où

$$A = \begin{pmatrix} 3 & -1 & 1 & 1 & 0 \\ -1 & -4 & 1 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \quad c = (1, -3, 1, 0, 0)^T, \\ l = 0_{\mathbb{R}^5}, \quad u = (1, 4, 5, 5, 19)^T.$$

Posons $\eta = 1$ et commençons par la SRS initiale :

$$J_B = \{4, 5\}, \quad J_N = \{1, 2, 3\}, \quad x = (0, 0, 0, 1, 2)^T.$$

Itération 1

- La valeur de la fonction objectif en x est $z = 0$.
- Le vecteur des multiplicateurs et le vecteur des coûts réduits sont

$$\pi^T = c_B^T A_B^{-1} = (0, 0)^T, \quad \Delta^T = \pi^T A - c^T = (-1, 3, -1, 0, 0)^T.$$

- Les ensembles J_N^+, J_N^- et l'estimation de suboptimalité sont

$$J_N^+ = \{2\}, \quad J_N^- = \{1, 3\},$$

$$\beta = \sum_{j \in J_N^+} \Delta_j(x_j - l_j) + \sum_{j \in J_N^-} \Delta_j(x_j - u_j) = 6 > 0.$$

- Les vecteurs $\eta(x - u)$ et $\eta(x - l)$ sont

$$\eta(x - u) = (-1, -4, -5, -4, -17)^T, \quad \eta(x - l) = (0, 0, 0, 1, 2)^T.$$

- Les ensembles $J_{NI}^+, J_{NI}^-, J_{NE}^+$ et J_{NE}^- sont

$$J_{NI}^+ = \emptyset, \quad J_{NI}^- = \{1, 3\}, \quad J_{NE}^+ = J_{NE}^- = \emptyset.$$

- La valeur de l'estimation μ est

$$\mu = - \sum_{j \in J_{NE}^+} \Delta_j(x_j - l_j) - \sum_{j \in J_{NE}^-} \Delta_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_{NE}} \Delta_j^2 = 0.$$

- La direction d'amélioration d est

$$d_N = (1, 0, 5)^T, \quad d_B = -A_B^{-1} A_N d_N = (-8, -4)^T,$$

$$d = (1, 0, 5, -8, -4)^T.$$

- Le pas le long de la direction d est

$$\theta_{j_1} = \min\{\theta_4, \theta_5\} = \min\left\{\frac{1}{8}, \frac{1}{2}\right\} = \frac{1}{8} \Rightarrow j_1 = 4.$$

$$\theta_{j_2} = \infty \quad (J_{NE} = \emptyset), \quad \theta^0 = \min\{\theta_{j_1}, \theta_{j_2}, 1\} = \theta_{j_1} = \frac{1}{8}.$$

- La nouvelle solution est

$$\bar{x} = x + \theta^0 d = \left(\frac{1}{8}, 0, \frac{5}{8}, 0, \frac{3}{2}\right)^T.$$

- La nouvelle valeur de la fonction objectif est $\bar{z} = z + \theta^0(\beta + \mu) = \frac{3}{4} \simeq 0.75$.

- L'estimation de suboptimalité $\bar{\beta}$

$$\bar{\beta} = (1 - \theta^0)\beta - \theta^0\mu = \frac{21}{4} > 0.$$

Puisque $\theta^0 = \theta_{j_1} < 1$, on change le support J_B .

- Le vecteur κ et la valeur de α_0 sont

$$\kappa = x + d = (1, 0, 5, -7, -2)^T, \quad \alpha_0 = \kappa_{j_1} - \bar{x}_{j_1} = -7.$$

- La direction duale est calculée comme suit :

$$t_B^T = (t_4, t_5) = (1, 0), \quad t_N^T = (t_1, t_2, t_3) = t_B^T A_B^{-1} A_N = (3, -1, 1),$$

$$t = (3, -1, 1, 1, 0)^T.$$

Puisque $J_{N0}^+ = J_{N0}^- = J_{NE}^+ = J_{NE}^- = \emptyset$, la quantité α est égale à $\alpha = -|\alpha_0| = -7$.

- Le pas le long de la direction duale t et l'indice entrant i_q sont

$$\sigma_1 = \frac{1}{3}, \quad \sigma_2 = 3, \quad \sigma_3 = 1.$$

Nous avons $\sigma_1 < \sigma_3 < \sigma_2 \Rightarrow i_1 = 1, i_2 = 3, i_3 = 2$, et $p = 3$. Donc

$$\begin{cases} \Delta V_{i_1} = \Delta V_1 = |t_1|(u_1 - l_1) = 3, \\ \Delta V_{i_2} = \Delta V_3 = |t_3|(u_3 - l_3) = 5, \\ \Delta V_{i_3} = \Delta V_2 = |t_2|(u_2 - l_2) = 4. \end{cases}$$

Les valeurs de V_{i_k} sont

$$\begin{cases} V_{i_0} = V_0 = \alpha = -7 < 0, \\ V_{i_1} = V_1 = V_{i_0} + \Delta V_{i_1} = -4 < 0, \\ V_{i_2} = V_3 = V_{i_1} + \Delta V_{i_2} = 1 > 0. \end{cases}$$

Nous avons $V_{i_1} = -4 < 0$ et $V_{i_2} = 1 > 0$, donc l'indice entrant est $i_q = i_2 = 3$. Par conséquent, $q = 2$ et le pas dual est

$$\sigma^0 = \sigma_{i_q} = \sigma_3 = 1.$$

- La nouvelle estimation de suboptimalité est

$$\bar{\beta} = \bar{\beta} + \sigma_{i_1} V_0 + (\sigma_{i_2} - \sigma_{i_1}) V_1 = \bar{\beta} + \sigma_1 V_0 + (\sigma_3 - \sigma_1) V_1 = \frac{1}{4} > 0.$$

- Le nouveau vecteur des coûts réduits et le nouveau support sont

$$\bar{J}_B = \{3, 5\}, \bar{J}_N = \{1, 2, 4\}, \bar{\Delta} = \Delta + \sigma^0 t = (2, 2, 0, 1, 0)^T.$$

Remarque 3.10. Si nous utilisons la règle du pas simple, nous obtenons

$$\sigma^0 = \min\{\sigma_1, \sigma_2, \sigma_3\} = \min\{\frac{1}{3}, 3, 1\} = \frac{1}{3} \Rightarrow j_0 = i_q = 1.$$

Ainsi, le nouveau vecteur des coûts réduits, le nouveau support et la nouvelle estimation de suboptimalité seront égaux à

$$\bar{J}_B = \{1, 5\}, \bar{\Delta} = (0, \frac{8}{3}, \frac{-2}{3}, \frac{1}{3}, 0)^T, \bar{\beta} = \bar{\beta} + \sigma^0 \alpha = \frac{35}{12} > \frac{1}{4}.$$

Il est donc important de voir ici que la règle du pas multiple donne une meilleure diminution pour l'estimation de suboptimalité que la règle du pas simple.

Itération 2

- Le support courant est $J_B = \{3, 5\}$, $J_N = \{1, 2, 4\}$.
- La solution réalisable courante et la valeur objectif en cette solution sont

$$x = (\frac{1}{8}, 0, \frac{5}{8}, 0, \frac{3}{2})^T \text{ et } z = \frac{3}{4}.$$

- Le vecteur des coûts réduits est

$$\Delta = (2, 2, 0, 1, 0)^T.$$

- Les ensembles J_N^+ , J_N^- et l'estimation de suboptimalité sont

$$J_N^+ = \{1, 2, 4\}, J_N^- = \emptyset, \beta = \frac{1}{4} > 0.$$

- Les vecteurs $\eta(x - u)$ et $\eta(x - l)$ sont

$$\eta(x - u) = (\frac{-7}{8}, -4, \frac{-35}{8}, -5, \frac{-35}{2})^T, \eta(x - l) = (\frac{1}{8}, 0, \frac{5}{8}, 0, \frac{3}{2})^T.$$

- Les ensembles J_{NE}^+ , J_{NE}^- , J_{NI}^+ , J_{NI}^- et la valeur de μ sont

$$J_{NE}^+ = \{1\}, J_{NE}^- = \emptyset, J_{NI}^+ = \emptyset, J_{NI}^- = \emptyset \text{ et } \mu = \frac{15}{4}.$$

- La direction d'amélioration d est

$$d_N = (-2, 0, 0)^T, d_B = (6, -8)^T, d = (-2, 0, 6, 0, -8)^T.$$

- Le pas le long de la direction d est

$$\theta_{j_1} = \min\{\theta_3, \theta_5\} = \min\{\frac{35}{48}, \frac{3}{16}\} = \frac{3}{16} \Rightarrow j_1 = 5,$$

$$\theta_{j_2} = \min\{\theta_1\} = \frac{1}{16} \Rightarrow j_2 = 1, \theta^0 = \min\{\theta_{j_1}, \theta_{j_2}, 1\} = \theta_{j_2} = \frac{1}{16}.$$

- La nouvelle solution et la nouvelle valeur de la fonction objectif sont

$$\bar{x} = x + \theta^0 d = (0, 0, 1, 0, 1)^T \text{ et } \bar{z} = 1.$$

- La nouvelle valeur de l'estimation de suboptimalité est

$$\bar{\beta} = (1 - \theta^0)\beta - \theta^0 \mu = 0.$$

Par conséquent, \bar{x} est optimale.

Chapitre 4

Nouvelle technique d'initialisation des algorithmes de PL

Introduction

L'une des étapes les plus importantes dans le processus de résolution des problèmes de programmation linéaire est l'étape d'initialisation qui consiste à trouver la solution réalisable initiale ainsi que la base (support) initiale. L'efficacité de l'algorithme utilisé pour trouver la solution optimale dépend du choix de la solution initiale. Il existe plusieurs techniques de recherche d'une SRS initiale, à savoir : la technique utilisant plusieurs variables artificielles [36], la technique utilisant une seule variable artificielle [7, 6].

Dans ce chapitre, nous présentons la procédure proposée par Bixby [17] qui permet de trouver une base initiale et qui constitue l'une des procédures utilisées pour l'initialisation de la méthode du simplexe implémentée dans GLPK. De plus, nous proposons une nouvelle procédure qui nous permet de trouver une solution réalisable initiale.

4.1 Position du problème

Considérons le problème de programmation linéaire écrit sous la forme générale suivante :

$$\begin{aligned} \max z &= \tilde{c}^T y, \\ b^- &\leq Hy \leq b^+ \\ \tilde{l} &\leq y \leq \tilde{u}, \end{aligned} \tag{4.1}$$

où $H \in \mathcal{M}_{m,s}(\mathbb{R})$; $b^-, b^+ \in \mathbb{R}^m$; $\tilde{l}, \tilde{u}, \tilde{c}, y \in \mathbb{R}^s$. Les ensembles

$$\mathcal{C} = \{1, 2, \dots, m\} \text{ et } \mathcal{V} = \{1, 2, \dots, s\}$$

sont respectivement l'ensemble des indices des contraintes et l'ensemble des indices des variables originales du problème (4.1).

Partitionnons l'ensemble \mathcal{C} de la manière suivante : $\mathcal{C} = \mathcal{C}_1 \cup \mathcal{C}_2$, où \mathcal{C}_1 représente l'ensemble des indices des contraintes d'inégalités et \mathcal{C}_2 celui des indices des contraintes d'égalités, c'est-à-dire, les contraintes d'indice i pour lesquels $b_i^- = b_i^+$. Supposons sans perte de généralité que

$$\mathcal{C}_1 = \{1, 2, \dots, k\}, \mathcal{C}_2 = \{k+1, k+2, \dots, m\}, k \leq m.$$

On peut alors écrire et fractionner les vecteurs b^-, b^+ et la matrice H de la manière suivante :

$$b^- = (b_i^-, i \in \mathcal{C}) = \begin{pmatrix} b_I^- \\ b_E^- \end{pmatrix}, b_I^- = (b_i^-, i \in \mathcal{C}_1), b_E^- = (b_i^-, i \in \mathcal{C}_2);$$

$$b^+ = (b_i^+, i \in \mathcal{C}) = \begin{pmatrix} b_I^+ \\ b_E^+ \end{pmatrix}, b_I^+ = (b_i^+, i \in \mathcal{C}_1), b_E^+ = (b_i^+, i \in \mathcal{C}_2);$$

$$H = (h_{ij}, i \in \mathcal{C}, j \in \mathcal{V}) = \begin{pmatrix} H^1 \\ H^2 \end{pmatrix}, \text{ où}$$

$$H^1 = (h_{ij}, i \in \mathcal{C}_1, j \in \mathcal{V}), H^2 = (h_{ij}, i \in \mathcal{C}_2, j \in \mathcal{V}).$$

Effectuons les changements de variable suivants :

$$x^1 = y, x^2 = H^1 x^1. \quad (4.2)$$

Les vecteur x^1 est celui des variables originales du problème (4.1) et x^2 est celui des variables d'écart. Le problème (4.1) devient

$$\begin{aligned} \max z &= \tilde{c}^T x^1, \\ -H^1 x^1 + I_k x^2 &= 0_{\mathbb{R}^k}, \\ H^2 x^1 &= b_E^+, \\ \tilde{l} \leq x^1 \leq \tilde{u}, \quad b_I^- \leq x^2 \leq b_I^+. \end{aligned} \quad (4.3)$$

On pose alors

$$A = \begin{pmatrix} -H^1 & I_k \\ H^2 & 0_{m-k, k} \end{pmatrix},$$

$$x = \begin{pmatrix} x^1 \\ x^2 \end{pmatrix}, c = \begin{pmatrix} \tilde{c} \\ 0_{\mathbb{R}^k} \end{pmatrix}, b = \begin{pmatrix} 0_{\mathbb{R}^k} \\ b_E^+ \end{pmatrix}, l = \begin{pmatrix} \tilde{l} \\ b_I^- \end{pmatrix}, u = \begin{pmatrix} \tilde{u} \\ b_I^+ \end{pmatrix}.$$

Finalement, on obtient le problème écrit sous la forme standard suivante :

$$\begin{aligned} \max z &= c^T x, \\ Ax &= b, \quad l \leq x \leq u, \end{aligned} \tag{4.4}$$

où $A = (a_{ij}) \in \mathcal{M}_{m,s+k}(\mathbb{R})$; $c, l, u, x \in \mathbb{R}^{s+k}$. Notons par $J = \{1, 2, \dots, n = s + k\}$ et supposons que $\text{rang}(A) = m$. Lorsque cette condition n'est pas vérifiée, on ajoute des vecteurs unitaires aux colonnes de A d'une manière appropriée.

Remarque 4.1. Lorsque $\text{rang}(A) = r < m$, les opérations élémentaires de Gauss-Jordan nous donnent la matrice $\bar{A} = (e_{j_1}, e_{j_2}, \dots, e_{j_r}, a'_{j_{r+1}}, \dots, a'_{j_n})$, où la notation a'_{j_p} , $p = r + 1, \dots, n$ indique que les vecteurs colonnes a_{j_p} , $p = r + 1, \dots, n$ ont été modifiés par des opérations élémentaires. Dans ce cas, il suffit d'ajouter des vecteurs unitaires e_{j_p} , $p = r + 1, \dots, m$ pour obtenir la matrice identité, et poser les variables associées égales à 0.

Jusqu'à présent, nous avons toujours étudié des cas où une solution réalisable de support initiale était connue. Cependant, cette situation (idéale) ne se rencontre pas dans tous les problèmes de programmation linéaire. Nous verrons donc dans la section suivante comment trouver une solution réalisable de support de départ pour initialiser les algorithmes MADH et MDHPM.

4.2 Construction d'une SRS initiale

Les méthodes MADH et MDHPM nécessitent en entrée une SRS initiale. Si une telle SRS n'est pas disponible, il est nécessaire de commencer par construire une sorte de problème auxiliaire. Ce problème auxiliaire est ensuite résolu (Phase I) et la SRS résultante est utilisée comme une SRS de départ pour résoudre le problème original (Phase II).

Dans cette section, nous allons nous intéresser aux techniques permettant de trouver une SRS pour initialiser MADH et MDHPM. On commence d'abord par la recherche d'un support initial; puis on procède à la recherche d'une solution réalisable initiale pour le problème original.

4.2.1 Recherche d'un support initial

Afin de rechercher un support initial, on pourra utiliser la technique de diagonalisation de Gauss-Jordan, la technique d'initialisation présentée dans [7] ou bien la procédure proposée dans [17] qui est décrite ci-dessous :

Procédure 4.1.

1. Pour $i \in \mathcal{C}_1$, on pose $I_i = 1$, $r_i = 1$, $v_i = +\infty$;
 Pour $i \in \mathcal{C}_2$, on pose $I_i = 0$, $r_i = 0$, $v_i = +\infty$. Poser $J_B = \{s + 1, s + 2, \dots, s + k\}$;

2. Pour $j \in \mathcal{V}$,

- (a) Soit $\alpha = |a_{i_1 j}| = \max_{r_i=0} |a_{ij}|$.
Si $\alpha \geq 0.99$, alors poser

$$\begin{aligned} J_B &:= J_B \cup \{j\}; \\ v_{i_1} &= \alpha; \\ I_{i_1} &= 1; \\ r_i &:= r_i + 1, \text{ pour tout } i, \text{ tel que } |a_{ij}| \neq 0; \end{aligned}$$

Poser $j := j + 1$; aller à l'étape (a);

- (b) Pour $i \in \mathcal{C}_2$: si $|a_{ij}| > 0.01v_i$, alors on pose $j := j + 1$. Sinon soit $\alpha = |a_{i_1 j}| = \max_{I_i=0} |a_{ij}|$;

- i. Si $\alpha = 0$, alors on pose $j := j + 1$, aller à l'étape (b);
ii. Sinon, poser

$$\begin{aligned} J_B &:= J_B \cup \{j\}; \\ v_{i_1} &= \alpha; \\ I_{i_1} &= 1; \\ r_i &:= r_i + 1, \text{ pour tout } i, \text{ tel que } |a_{ij}| \neq 0; \\ j &:= j + 1; \text{ aller à l'étape (b)}; \end{aligned}$$

3. S'il existe $I_i = 0$, Pour $i \in \mathcal{C}_2$, alors $\text{rang}(A) = m' < m$, donc on ajoute $(m - m')$ vecteurs unitaires e_{s+i} à la matrice A , puis on pose $J_B := J_B \cup \{s+k+i\}$, $l_{s+k+i} = 0$ et $u_{s+k+i} = 0$ (voir la remarque 4.1).

4.2.2 Recherche d'une solution réalisable initiale

Dans cette section, nous proposons une technique qui permet de trouver une solution réalisable initiale, et ce, sans ajouter de variables artificielles. Le problème de programmation linéaire auxiliaire obtenu a la même dimension que celle du problème original.

Supposons que J_B est un support du problème de PL (4.4) et $J_N = J \setminus J_B$. Soit x^0 une pseudo-solution réalisable du problème (4.4), telle que

$$l_N \leq x_N^0 \leq u_N \text{ et } x_B^0 = A_B^{-1}(b - A_N x_N^0).$$

Si $l_B \leq x_B^0 \leq u_B$, alors $\{x^0, J_B\}$ est une SRS du problème (4.4). Sinon, formons le problème auxiliaire suivant :

$$\max \psi(x) = \hat{c}^T x, \text{ s.c. } Ax = b, \hat{l} \leq x \leq \hat{u}, \quad (4.5)$$

où

$$\hat{l}_N = l_N, \hat{u}_N = u_N \text{ et } \hat{c}_N = 0,$$

et pour $j \in J_B$, on pose :

$$\begin{aligned} \hat{l}_j &= -M, \hat{u}_j = l_j, \hat{c}_j = 1, \text{ si } x_j^0 < l_j; \\ \hat{l}_j &= u_j, \hat{u}_j = M, \hat{c}_j = -1, \text{ si } x_j^0 > u_j; \\ \hat{l}_j &= l_j, \hat{u}_j = u_j, \hat{c}_j = 0, \text{ si } l_j \leq x_j^0 \leq u_j, \end{aligned}$$

M étant un nombre positif très grand. La fonction objectif du problème auxiliaire peut alors s'écrire :

$$\psi(x) = \sum_{x_j^0 < l_j, j \in J_B} x_j - \sum_{x_j^0 > u_j, j \in J_B} x_j.$$

La paire $\{x^0, J_B\}$ est évidemment une SRS pour le problème auxiliaire (4.5).

Soit $\{x^*, J_B^*\}$ la SRS optimale du problème auxiliaire (4.5), obtenue avec l'algorithme MADH ou MDHPM et considérons la valeur suivante :

$$w^* = \sum_{x_j^0 < l_j, j \in J_B} l_j - \sum_{x_j^0 > u_j, j \in J_B} u_j.$$

Théorème 4.1. *On a toujours $\psi(x^*) \leq w^*$. Si $\psi(x^*) = w^*$, alors $\{x^*, J_B^*\}$ est une SRS pour le problème original (4.4).*

Démonstration. On a :

$$w^* - \psi(x^*) = \sum_{x_j^0 < l_j, j \in J_B} (l_j - x_j^*) + \sum_{x_j^0 > u_j, j \in J_B} (x_j^* - u_j).$$

Puisque x^* est une solution réalisable du problème auxiliaire, alors on a

$$l_j - x_j^* \geq 0, \text{ pour } x_j^0 < l_j, j \in J_B \text{ et } x_j^* - u_j \geq 0, \text{ pour } x_j^0 > u_j, j \in J_B.$$

Donc $w^* - \psi(x^*) \geq 0 \Leftrightarrow \psi(x^*) \leq w^*$.

Si $\psi(x^*) = w^*$, alors on a

$$\sum_{x_j^0 < l_j, j \in J_B} (l_j - x_j^*) = \sum_{x_j^0 > u_j, j \in J_B} (x_j^* - u_j) = 0.$$

Donc x^* vérifie forcément les relations

$$\begin{cases} x_j^* = l_j, \text{ si } x_j^0 < l_j; \\ x_j^* = u_j, \text{ si } x_j^0 > u_j; \\ l_j \leq x_j^* \leq u_j, \text{ si } l_j \leq x_j^0 \leq u_j, j \in J_B. \end{cases}$$

D'où $l_B \leq x_B^* \leq u_B$.

Comme x^* est réalisable pour le problème auxiliaire, donc elle vérifie aussi les relations $Ax^* = b$ et $l_N \leq x_N^* \leq u_N$. Par conséquent, x^* est une solution réalisable pour le problème original.

□

4.3 Exemples numériques

Exemple 4.1. Considérons le problème suivant :

$$\begin{aligned} \max \phi &= p^T y + q, \\ b^0 &\leq A^1 y \leq b^1, \\ A^2 y &\geq b^2, \\ A^3 y &= b^3, \\ \tilde{l} &\leq y \leq \tilde{u}, \end{aligned} \tag{4.6}$$

où

$$A^1 = \begin{pmatrix} 0 & 0 & 1 & 2 \\ 2 & 0 & 0 & -3 \end{pmatrix}, \quad A^2 = \begin{pmatrix} -1 & 0 & 2 & 0 \end{pmatrix}, \quad A^3 = \begin{pmatrix} 0 & 6 & 0 & 4 \\ 1 & 0 & 2 & 3 \end{pmatrix},$$

$$b^0 = (0, -33)^T, \quad b^1 = (19, 10), \quad b^2 = \frac{1}{3}, \quad b^3 = (21, 24), \quad q = 10,$$

$$p = \begin{pmatrix} \frac{1}{3} \\ -\frac{1}{2} \\ -\frac{1}{6} \\ \frac{1}{6} \end{pmatrix}, \quad \tilde{l} = \begin{pmatrix} -2 \\ -\frac{1}{2} \\ 0 \\ 6 \end{pmatrix}, \quad \tilde{u} = \begin{pmatrix} 9 \\ 1 \\ 4 \\ 10 \end{pmatrix}.$$

Si l'on applique le changement de variables (4.2) et on pose $z = \psi - q$, alors on obtient le problème sous la forme standard suivante :

$$\begin{aligned} \max z &= c^T x, \\ Ax &= b, \quad l \leq x \leq u. \end{aligned} \tag{4.7}$$

où

$$A = \begin{pmatrix} 0 & 0 & -1 & -2 & 1 & 0 & 0 \\ -2 & 0 & 0 & 3 & 0 & 1 & 0 \\ 1 & 0 & -2 & 0 & 0 & 0 & 1 \\ 0 & 6 & 0 & 4 & 0 & 0 & 0 \\ 1 & 0 & 2 & 3 & 0 & 0 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 21 \\ 24 \end{pmatrix}, \quad c = \begin{pmatrix} \frac{1}{3} \\ -\frac{1}{2} \\ -\frac{1}{6} \\ \frac{1}{6} \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$l = \begin{pmatrix} -2 \\ -\frac{1}{2} \\ 0 \\ 6 \\ 0 \\ -33 \\ \frac{1}{3} \end{pmatrix}, \quad u = \begin{pmatrix} 9 \\ 1 \\ 4 \\ 10 \\ 19 \\ 10 \\ M \end{pmatrix},$$

où M est un nombre positif très grand.

Réolvons le problème (4.6) en utilisant MDHPM, avec $\eta = 1$.

Soient le support initial $J_B = \{1, 2, 5, 6, 7\}$, et $x_N = (0, 6)^T$, on obtient alors $x_B = (6, \frac{-1}{2}, 12, -6, -6)^T$.

Phase 1 : Pour trouver une solution réalisable initiale pour le problème (4.7), on forme le problème auxiliaire suivant :

$$\begin{aligned} \max \psi &= x_7, \\ Ax &= b, \quad \hat{l} \leq x \leq \hat{u}, \end{aligned} \tag{4.8}$$

où

$$\hat{l} = \begin{pmatrix} -2 \\ -\frac{1}{2} \\ 0 \\ 6 \\ 0 \\ -33 \\ -M \end{pmatrix}, \quad \hat{u} = \begin{pmatrix} 9 \\ 1 \\ 4 \\ 10 \\ 19 \\ 10 \\ \frac{1}{3} \end{pmatrix}.$$

Itération 1

- Le support initial est $J_B = \{1, 2, 5, 6, 7\}$ et $J_N = \{3, 4\}$.
- La solution réalisable initiale x et la valeur $\psi(x)$ sont

$$x = (6, \frac{-1}{2}, 0, 6, 12, -6, -6)^T, \quad \psi(x) = -6.$$

- Le vecteur des multiplicateurs et le vecteur des coûts réduits sont

$$\pi^T = c_B^T A_B^{-1} = (0, 0, 1, 0, -1), \quad \Delta = \pi^T A - c^T = (0, 0, -4, -3, 0, 0, 0)^T.$$

- Les ensembles J_N^+ , J_N^- et l'estimation de suboptimalité sont

$$J_N^+ = \emptyset, \quad J_N^- = \{3, 4\},$$

$$\beta = \sum_{j \in J_N^+} \Delta_j (x_j - \hat{l}_j) + \sum_{j \in J_N^-} \Delta_j (x_j - \hat{u}_j) = 28 > 0.$$

- Les vecteurs $\eta(x - \hat{u})$ et $\eta(x - \hat{l})$ sont

$$\eta(x - \hat{u}) = \left(-3, \frac{-3}{2}, -4, -4, -7, -16, \frac{-19}{3}\right)^T,$$

$$\eta(x - \hat{l}) = (8, 0, 0, 0, 12, 27, M - 6)^T.$$

- Les ensembles J_{NI}^+ , J_{NI}^- , J_{NE}^+ et J_{NE}^- sont

$$J_{NI}^+ = \emptyset, \quad J_{NI}^- = \{3, 4\}, \quad J_{NE}^+ = \emptyset, \quad J_{NE}^- = \emptyset.$$

- La valeur de l'estimation μ est

$$\mu = - \sum_{j \in J_{NE}^+} \Delta_j (x_j - \hat{l}_j) - \sum_{j \in J_{NE}^-} \Delta_j (x_j - \hat{u}_j) + \frac{1}{\eta} \sum_{j \in J_{NE}} \Delta_j^2 = 0.$$

- La direction d'amélioration d est

$$d_N = (4, 4)^T, \quad d_B = -A_B^{-1} A_N d_N = \left(-20, \frac{-8}{3}, 12, -52, 28\right)^T,$$

$$d = \left(-20, \frac{-8}{3}, 4, 4, 12, -52, 28\right)^T.$$

- Le pas le long de la direction d est

$$\theta_{j_1} = \min\{\theta_1, \theta_2, \theta_5, \theta_6, \theta_7\} = \min\left\{\frac{2}{5}, 0, \frac{7}{12}, \frac{27}{52}, \frac{3}{14}\right\} = 0 \Rightarrow j_1 = 2.$$

$$\theta_{j_2} = \infty \quad (J_{NE} = \emptyset), \quad \theta^0 = \min\{\theta_{j_1}, \theta_{j_2}, 1\} = \theta_{j_1} = 0.$$

- La nouvelle solution \bar{x} est

$$\bar{x} = x + \theta^0 d = \left(6, \frac{-1}{2}, 0, 6, 12, -6, -6\right)^T.$$

- La nouvelle valeur de la fonction objectif est $\psi(\bar{x}) = \psi(x) + \theta^0(\beta + \mu) = -6$.

- La nouvelle estimation de suboptimalité $\bar{\beta}$ est

$$\bar{\beta} = (1 - \theta^0)\beta - \theta^0\mu = 28 > 0.$$

Puisque $\theta^0 = \theta_{j_1} < 1$, on change le support J_B .

- Le vecteur κ et la valeur de α_0 sont

$$\kappa = x + d = (-14, \frac{-19}{6}, 4, 10, 24, -58, 22)^T, \alpha_0 = \kappa_{j_1} - \bar{x}_{j_1} = \frac{-8}{3}.$$

- La direction duale est calculée comme suit :

$$t_B^T = (0, 1, 0, 0, 0), t_N^T = t_B^T A_B^{-1} A_N = (0, \frac{2}{3}), t = (0, 1, 0, \frac{2}{3}, 0, 0, 0)^T;$$

$$J_{N_0}^+ = \emptyset, J_{N_0}^- = \emptyset.$$

- La quantité α est égale à

$$\alpha = -\frac{8}{3}.$$

- Le pas dual et l'indice entrant j_0 sont

$$\sigma_3 = +\infty, \sigma_4 = \frac{9}{2} \Rightarrow \sigma^0 = \frac{9}{2}.$$

- La nouvelle estimation de suboptimalité $\bar{\bar{\beta}}$ est

$$\bar{\bar{\beta}} = \bar{\beta} + \sigma^0\alpha = 16 > 0.$$

- Le nouveau vecteur des coûts réduits et le nouveau support sont

$$\bar{J}_B = \{1, 4, 5, 6, 7\}, \bar{J}_N = \{2, 3\}, \bar{\Delta} = \Delta + \sigma^0 t = (\frac{9}{2}, -4)^T.$$

Itération 2

- Le support courant est $J_B = \{1, 4, 5, 6, 7\}$, $J_N = \{2, 3\}$.
- La solution réalisable courante et la valeur courante de la fonction objectif sont

$$x = (6, \frac{-1}{2}, 0, 6, 12, -6, -6)^T \text{ et } \psi(x) = -6.$$

- Le vecteur des coûts réduits est

$$\Delta^T = (0, \frac{9}{2}, -4, 0, 0, 0, 0).$$

- L'estimation de suboptimalité est $\beta = 16 > 0$.

- Les vecteurs $\eta(x - \hat{u})$ et $\eta(x - \hat{l})$ sont

$$\eta(x - \hat{u}) = \left(-3, \frac{-3}{2}, -4, -4, -7, -16, \frac{-19}{3}\right)^T, \quad \eta(x - \hat{l}) = (8, 0, 0, 0, 12, 27, M - 6)^T.$$

- Les ensembles $J_{NE}^+, J_{NE}^-, J_{NI}^+, J_{NI}^-$ et la valeur de μ sont

$$J_{NE}^- = \emptyset, J_{NE}^+ = \emptyset, J_{NI}^+ = \emptyset, J_{NI}^- = \{3\} \text{ et } \mu = 0.$$

- La direction d'amélioration d est

$$d_N = (0, 4)^T, d_B = (-8, 0, 4, -16, 16)^T, d = (-8, 0, 4, 0, 4, -16, 16)^T.$$

- Le pas le long de la direction d est

$$\theta_{j_1} = \min\{\theta_1, \theta_4, \theta_5, \theta_6, \theta_7\} = \min\left\{1, +\infty, \frac{7}{4}, \frac{27}{16}, \frac{19}{48}\right\} = \frac{19}{48} \Rightarrow j_1 = 7.$$

$$\theta_{j_2} = \infty, \theta^0 = \min\{\theta_{j_1}, \theta_{j_2}, 1\} = \theta_{j_1} = \frac{19}{48}.$$

- La nouvelle solution et la nouvelle valeur de la fonction objectif sont

$$\bar{x} = x + \theta^0 d = \left(\frac{17}{6}, \frac{-1}{2}, \frac{19}{12}, 6, \frac{163}{12}, \frac{-37}{3}, \frac{1}{3}\right)^T \text{ et } \psi(\bar{x}) = \frac{1}{3}.$$

- La nouvelle valeur de l'estimation de suboptimalité est

$$\bar{\beta} = (1 - \theta^0)\beta - \theta^0\mu = \frac{29}{3}.$$

- Le vecteur κ et la valeur de α_0 sont

$$\kappa = x + d = \left(-2, \frac{-1}{2}, 4, 6, 16, -22, 10\right)^T, \quad \alpha_0 = \kappa_{j_1} - \bar{x}_{j_1} = \frac{29}{3}.$$

- La direction duale est calculée comme suit :

$$t_B^T = (0, 0, 0, 0, -1), \quad t_N^T = t_B^T A_B^{-1} A_N = \left(\frac{-9}{2}, 4\right), \quad t = \left(0, \frac{-9}{2}, 4, 0, 0, 0, -1\right)^T;$$

$$J_{N_0}^+ = \emptyset, J_{N_0}^- = \emptyset.$$

- La quantité α est $\alpha = \frac{-29}{3}$.

- Le pas dual et l'indice entrant j_0 sont

$$\sigma_2 = 1, \sigma_3 = 1 \Rightarrow \sigma^0 = \sigma_2 = 1.$$

- La nouvelle estimation de suboptimalité est

$$\bar{\beta} = \bar{\beta} + \sigma^0 \alpha = 0.$$

- Le nouveau support sont

$$\bar{J}_B = \{1, 2, 4, 5, 6\}, \bar{J}_N = \{3, 7\}.$$

Phase 2 :

Itération 1

- Le support initial est $J_B = \{1, 2, 4, 5, 6\}$ et $J_N = \{3, 7\}$
- La solution réalisable initiale et la valeur $z(x)$ sont

$$x = \left(\frac{17}{6}, \frac{-1}{2}, \frac{19}{12}, 6, \frac{163}{12}, \frac{-37}{3}, \frac{1}{3} \right)^T, z(x) = \frac{139}{72} \simeq 1.93.$$

- Le vecteur des multiplicateurs et le vecteur des coûts réduits sont

$$\pi^T = c_B^T A_B^{-1} = \left(0, 0, \frac{5}{24}, \frac{-5}{96}, \frac{1}{8} \right), \Delta = \pi^T A - c^T = \left(0, \frac{3}{16}, 0, 0, 0, \frac{5}{24} \right)^T.$$

- Les ensembles J_N^+, J_N^- et l'estimation de suboptimalité sont

$$J_N^+ = \{2, 7\}, J_N^- = \emptyset,$$

$$\beta = \sum_{j \in J_N^+} \Delta_j (x_j - l_j) + \sum_{j \in J_N^-} \Delta_j (x_j - u_j) = 0.$$

Donc x est une SRS optimale du problème (4.7). Par conséquent,

$$y = \left(\frac{17}{6}, \frac{-1}{2}, \frac{19}{12}, 6 \right)^T$$

est une SRS optimale pour le problème (4.6), avec $\phi(y) = \frac{299}{24} \simeq 12.46$.

Exemple 4.2. Considérons le problème suivant :

$$\begin{aligned} \max z &= p^T y, \\ Ay &= b, \tilde{l} \leq y \leq \tilde{u}, \end{aligned} \quad (4.9)$$

où

$$A = \begin{pmatrix} 1 & -1 & 3 & 2 \\ -7 & 1 & 2 & 3 \end{pmatrix}, b = (1, 1)^T,$$

$$p = \begin{pmatrix} 4 \\ -6 \\ -2 \\ 2 \end{pmatrix}, \tilde{l} = \begin{pmatrix} -1 \\ -2 \\ -3 \\ -4 \end{pmatrix}, \tilde{u} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}.$$

Phase 1 : Soit $J_B = \{1, 2\}$ un support du problème (4.9). Si $x_N = \tilde{l}_N = (-3, -4)^T$, alors $x_B = (-\frac{37}{6}, -\frac{145}{6})^T$. Remarquons que x n'est pas une solution réalisable du problème (4.9). Pour trouver une solution réalisable initiale pour le problème (4.9), on forme le problème auxiliaire suivant :

$$\begin{aligned} \max \psi &= c^T x, \\ Ax &= b, l \leq x \leq u, \end{aligned} \quad (4.10)$$

où

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}, c = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}, l = \begin{pmatrix} -M \\ -M \\ -3 \\ -4 \end{pmatrix}, u = \begin{pmatrix} -1 \\ -2 \\ 3 \\ 4 \end{pmatrix},$$

où M est un nombre positif très grand.

Appliquons l'algorithme MDHPM, avec $\eta = 1$.

Itération 1

- Le support initial est $J_B = \{1, 2\}$ et $J_N = \{3, 4\}$.
- La solution réalisable initiale x et sa valeur $\psi(x)$ sont

$$x = \left(-\frac{37}{6}, -\frac{145}{6}, -3, -4\right)^T, \psi = -\frac{91}{3}.$$

- Le vecteur des multiplicateurs et le vecteur des coûts réduits sont

$$\pi^T = c_B^T A_B^{-1} = \left(-\frac{4}{3}, -\frac{1}{3}\right), \Delta_N = \pi^T A_N - c_N^T = \left(-\frac{14}{3}, -\frac{11}{3}\right)^T.$$

- Les ensembles J_N^+ , J_N^- et l'estimation de suboptimalité sont

$$J_N^+ = \emptyset, J_N^- = \{3, 4\},$$

$$\beta = \sum_{j \in J_N^+} \Delta_j(x_j - l_j) + \sum_{j \in J_N^-} \Delta_j(x_j - u_j) = \frac{172}{3} > 0.$$

- Les vecteurs $\eta(x - u)$ et $\eta(x - l)$ sont

$$\eta(x - u) = \left(\frac{-31}{6}, \frac{-133}{6}, -6, -8\right)^T, \eta(x - l) = \left(\frac{-37}{6} - M, \frac{-145}{6} - M, 0, 0\right)^T.$$

- Les ensembles J_{NI}^+ , J_{NI}^- , J_{NE}^+ et J_{NE}^- sont

$$J_{NI}^+ = \emptyset, J_{NI}^- = \{3, 4\}, J_{NE}^+ = \emptyset, J_{NE}^- = \emptyset.$$

- La valeur de l'estimation μ est

$$\mu = - \sum_{j \in J_{NE}^+} \Delta_j(x_j - l_j) - \sum_{j \in J_{NE}^-} \Delta_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_{NE}} \Delta_j^2 = 0.$$

- La direction d'amélioration d est

$$d_N = (6, 8)^T, d_B = -A_B^{-1} A_N d_N = \left(\frac{35}{3}, \frac{137}{3}\right)^T,$$

$$d = \left(\frac{35}{3}, \frac{137}{3}, 6, 8\right)^T.$$

- Le pas le long de la direction d est

$$\theta_{j_1} = \min\{\theta_1, \theta_2\} = \min\left\{\frac{31}{70}, \frac{133}{274}\right\} = \frac{31}{70} \Rightarrow j_1 = 1.$$

$$\theta_{j_2} = +\infty, \theta^0 = \min\{\theta_{j_1}, \theta_{j_2}, 1\} = \theta_{j_1} = \frac{31}{70}.$$

- La nouvelle solution \bar{x} est

$$\bar{x} = x + \theta^0 d = \left(-1, \frac{-138}{35}, \frac{-12}{35}, \frac{-16}{35}\right)^T.$$

- La nouvelle valeur de la fonction objectif est

$$\psi(\bar{x}) = \psi(x) + \theta^0(\beta + \mu) = \frac{-173}{35}.$$

- L'estimation de suboptimalité $\bar{\beta}$

$$\bar{\beta} = (1 - \theta^0)\beta - \theta^0\mu = \frac{1118}{35} > 0.$$

Puisque $\theta^0 = \theta_{j_1} < 1$, on change le support J_B .

- Le vecteur κ et la valeur de α_0 sont

$$\kappa = x + d = \left(\frac{11}{2}, \frac{43}{2}, 3, 4\right)^T, \quad \alpha_0 = \kappa_{j_1} - \bar{x}_{j_1} = \frac{13}{2}.$$

- La direction duale est

$$t_B^T = (-1, 0), \quad t_N^T = t_B^T A_B^{-1} A_N = \left(\frac{5}{6}, \frac{5}{6}\right), \quad t = (-1, 0, \frac{5}{6}, \frac{5}{6})^T;$$

$$J_{N_0}^+ = \emptyset, \quad J_{N_0}^- = \emptyset.$$

- La quantité α est $\alpha = \frac{-13}{2}$.
- Le pas dual et l'indice entrant j_0 sont

$$\sigma_3 = \frac{28}{5}, \quad \sigma_4 = \frac{22}{5}.$$

Nous avons $\sigma_4 < \sigma_3 \Rightarrow i_1 = 4, i_2 = 3$, et $p = 2$. Donc

$$\begin{cases} \Delta V_{i_1} = |t_4|(u_4 - l_4) = \frac{20}{3}, \\ \Delta V_{i_2} = |t_3|(u_3 - l_3) = 5. \end{cases}$$

Les valeurs de V_{i_k} sont

$$\begin{cases} V_{i_0} = \alpha = \frac{-13}{2} < 0, \\ V_{i_1} = V_{i_0} + \Delta V_{i_1} = \frac{7}{3} > 0. \end{cases}$$

Nous avons $V_{i_0} < 0$ et $V_{i_1} > 0$, donc l'indice entrant est $i_q = i_1 = 4$. Par conséquent, $q = 1$ et le pas dual est

$$\sigma^0 = \sigma_{i_q} = \sigma_4 = \frac{22}{5}.$$

- La nouvelle estimation de suboptimalité est

$$\bar{\beta} = \bar{\beta} + \sigma_{i_1} V_{i_0} = \frac{117}{35} > 0.$$

- Le nouveau vecteur des coûts réduits et le nouveau support sont

$$\bar{J}_B = \{2, 4\}, \quad \bar{J}_N = \{1, 3\}, \quad \bar{\Delta} = \Delta + \sigma^0 t = \left(-\frac{22}{5}, 0, -1, 0\right)^T.$$

Itération 2

- Le support courant est $J_B = \{2, 4\}$, $J_N = \{1, 3\}$.

- La solution réalisable courante et sa valeur $\psi(x)$ sont

$$x = \left(-1, \frac{-138}{35}, \frac{-12}{35}, \frac{-16}{35}\right)^T \text{ et } \psi(x) = -\frac{173}{35} \simeq -4.94.$$

- Le vecteur des coûts réduits est

$$\Delta^T = \left(\frac{-22}{5}, 0, -1, 0\right).$$

- L'estimation de suboptimalité est $\beta = \frac{117}{35} > 0$.

- Les vecteurs $\eta(x - u)$ et $\eta(x - l)$ sont

$$\eta(x - u) = \left(0, \frac{-68}{35}, \frac{-117}{35}, \frac{-156}{35}\right)^T,$$

$$\eta(x - l) = \left(-1 + M, \frac{-138}{35} + M, \frac{93}{35}, \frac{124}{35}\right)^T.$$

- Les ensembles J_{NE}^+ , J_{NE}^- , J_{NI}^+ , J_{NI}^- et la valeur de μ sont

$$J_{NE}^- = \emptyset, J_{NE}^+ = \emptyset, J_{NI}^+ = \emptyset, J_{NI}^- = \{3\} \text{ et } \mu = 0.$$

- La direction d'amélioration d est

$$d_N = \left(0, \frac{117}{35}\right)^T, d_B = \left(\frac{117}{35}, \frac{-117}{35}\right)^T, d = \left(0, \frac{117}{35}, \frac{117}{35}, \frac{-117}{35}\right)^T.$$

- Le pas le long de la direction d est

$$\theta_{j_1} = \min\{\theta_2, \theta_4\} = \min\left\{\frac{68}{117}, \frac{124}{117}\right\} = \frac{68}{117} \Rightarrow j_1 = 2.$$

$$\theta_{j_2} = +\infty, \theta^0 = \min\{\theta_{j_1}, \theta_{j_2}, 1\} = \theta_{j_1} = \frac{68}{117}.$$

- La nouvelle solution et la nouvelle valeur de la fonction objectif sont

$$\bar{x} = x + \theta^0 d = \left(-1, -2, \frac{56}{35}, \frac{-84}{35}\right)^T \text{ et } \psi(\bar{x}) = -3.$$

- L'estimation de la suboptimalité $\bar{\beta}$

$$\bar{\beta} = (1 - \theta^0)\beta - \theta^0\mu = \frac{7}{5} > 0.$$

Puisque $\theta^0 = \theta_{j_1} < 1$, on change le support J_B .

- Le vecteur κ et la valeur de α_0 sont

$$\kappa = x + d = \left(-1, \frac{-3}{5}, 3, \frac{-19}{5}\right)^T, \quad \alpha_0 = \kappa_{j_1} - \bar{x}_{j_1} = \frac{7}{5}.$$

- La direction duale est

$$t = \left(\frac{17}{5}, -1, 1, 0\right)^T, \quad J_{N_0}^+ = \emptyset, \quad J_{N_0}^- = \emptyset.$$

- La quantité α est $\alpha = \frac{-7}{5}$.

- Le pas dual et l'indice entrant j_0 sont

$$\sigma_1 = \frac{22}{17}, \quad \sigma_3 = 1.$$

Nous avons $\sigma_3 < \sigma_1 \Rightarrow i_1 = 3, i_2 = 1$, et $p = 2$. Donc

$$\begin{cases} \Delta V_{i_1} = |t_4|(u_4 - l_4) = 6, \\ \Delta V_{i_2} = |t_3|(u_3 - l_3) = \frac{17}{5}(M - 1). \end{cases}$$

Les valeurs de V_{i_k} sont

$$\begin{cases} V_{i_0} = \alpha = \frac{-7}{5} < 0, \\ V_{i_1} = V_{i_0} + \Delta V_{i_1} = \frac{23}{5}. \end{cases}$$

Nous avons $V_{i_0} < 0$ et $V_{i_1} > 0$, donc l'indice entrant est $i_q = i_1 = 3$. Par conséquent, $q = 1$ et le pas dual est

$$\sigma^0 = \sigma_{i_q} = \sigma_3 = 1.$$

- La nouvelle estimation de suboptimalité est

$$\bar{\beta} = \bar{\beta} + \sigma_{i_1} V_{i_0} = 0.$$

- Le nouveau vecteur des coûts réduits et le nouveau support sont

$$\bar{J}_B = \{3, 4\}, \quad \bar{J}_N = \{1, 2\}, \quad \bar{\Delta} = \Delta + \sigma^0 t = (-1, -1, 0, 0)^T.$$

Phase 2 :

Itération 1

- Le support initial est $J_B = \{3, 4\}$ et $J_N = \{1, 2\}$.
- La solution réalisable initiale et sa valeur $z(x)$ sont

$$x = \left(-1, -2, \frac{56}{35}, \frac{-84}{35}\right)^T, \quad z(x) = 0.$$

- Le vecteur des multiplicateurs et le vecteur des coûts réduits sont

$$\Delta = \pi^T A - c^T = (-20, 10, 0, 0)^T.$$

- Les ensembles J_N^+ , J_N^- et l'estimation de suboptimalité sont

$$J_N^+ = \{4\}, J_N^- = \{3\},$$

$$\beta = \sum_{j \in J_N^+} \Delta_j(x_j - l_j) + \sum_{j \in J_N^-} \Delta_j(x_j - u_j) = 40.$$

- Les vecteurs $\eta(x - u)$ et $\eta(x - l)$ sont

$$\eta(x - u) = (-2, -4, \frac{-7}{5}, \frac{-32}{5})^T, \eta(x - l) = (0, 0, \frac{23}{5}, \frac{8}{5})^T.$$

- Les ensembles J_{NI}^+ , J_{NI}^- , J_{NE}^+ et J_{NE}^- sont

$$J_{NI}^- = \emptyset, J_{NI}^+ = \emptyset, J_{NE}^+ = \emptyset, J_{NE}^- = \{1\}.$$

- La valeur de l'estimation μ est

$$\mu = - \sum_{j \in J_{NE}^+} \Delta_j(x_j - l_j) - \sum_{j \in J_{NE}^-} \Delta_j(x_j - u_j) + \frac{1}{\eta} \sum_{j \in J_{NE}} \Delta_j^2 = 360.$$

- La direction d'amélioration d est

$$d_N = (20, 0)^T, d_B = -A_B^{-1} A_N d_N = (-68, 92)^T,$$

$$d = (2, 0, -68, 92)^T.$$

- Le pas le long de la direction d est

$$\theta_{j_1} = \min\{\theta_3, \theta_4\} = \min\{\frac{23}{340}, \frac{32}{460}\} = \frac{23}{340} \Rightarrow j_1 = 3.$$

$$\theta_{j_2} = +\infty, \theta^0 = \min\{\theta_{j_1}, \theta_{j_2}, 1\} = \theta_{j_1} = \frac{23}{340}.$$

- La nouvelle solution \bar{x} est

$$\bar{x} = x + \theta^0 d = (\frac{6}{17}, -2, -3, \frac{65}{17})^T.$$

- La nouvelle valeur de la fonction objectif est

$$z(\bar{x}) = z(x) + \theta^0(\beta + \mu) = \frac{460}{17} \simeq 27.06.$$

- L'estimation de suboptimalité $\bar{\beta}$ est

$$\bar{\beta} = (1 - \theta^0)\beta - \theta^0\mu = \frac{220}{17} > 0.$$

Puisque $\theta^0 = \theta_{j_1} < 1$, on change le support J_B .

- Le vecteur κ et la valeur de α_0 sont

$$\kappa = x + d = (19, -2, \frac{-332}{5}, \frac{448}{5})^T, \alpha_0 = \kappa_{j_1} - \bar{x}_{j_1} = \frac{-317}{5}.$$

- La direction duale est

$$t_B^T = (1, 0), t_N^T = t_B^T A_B^{-1} A_N = (\frac{17}{5}, -1), t = (\frac{17}{5}, -1, 1, 0)^T;$$

$$J_{N_0}^+ = \emptyset, J_{N_0}^- = \emptyset.$$

- La quantité α est $\alpha = \frac{-11}{5}$.

- Le pas dual et l'indice entrant j_0 sont

$$\sigma_1 = \frac{100}{17}, \sigma_2 = 10.$$

Nous avons $\sigma_1 < \sigma_2 \Rightarrow i_1 = 1, i_2 = 2$, et $p = 2$. Donc

$$\begin{cases} \Delta V_{i_1} = |t_1|(u_1 - l_1) = \frac{34}{5}, \\ \Delta V_{i_2} = |t_2|(u_2 - l_2) = 4. \end{cases}$$

Les valeurs de V_{i_k} sont

$$\begin{cases} V_{i_0} = \alpha = \frac{-11}{5} < 0, \\ V_{i_1} = V_{i_0} + \Delta V_{i_1} = \frac{23}{5} > 0. \end{cases}$$

Nous avons $V_{i_0} < 0$ et $V_{i_1} > 0$, donc l'indice entrant est $i_q = i_1 = 1$. Par conséquent, $q = 1$ et le pas dual est

$$\sigma^0 = \sigma_{i_q} = \sigma_1 = \frac{100}{17}.$$

- La nouvelle estimation de suboptimalité est $\bar{\bar{\beta}} = \bar{\beta} + \sigma_{i_1} V_{i_0} = 0$.

- Le nouveau vecteur des coûts réduits et le nouveau support sont

$$\bar{J}_B = \{1, 4\}, \bar{J}_N = \{2, 3\}, \bar{\Delta} = \Delta + \sigma^0 t = (0, \frac{70}{17}, \frac{100}{17}, 0)^T.$$

Donc \bar{x} est une SRS optimale du problème (4.9).

Chapitre 5

Résultats expérimentaux et application au contrôle optimal

Introduction

Dans la première partie de ce chapitre, nous donnons un bref aperçu du problème de contrôle optimal linéaire ; puis nous montrons comment trouver une solution approchée pour ce problème en utilisant l'algorithme que nous avons proposé au [chapitre 3](#). Nous présentons également un problème de planification de la production [4]. Ce modèle de contrôle optimal linéaire est ensuite utilisé pour générer des problèmes-test pour la comparaison de notre méthode avec la méthode du simplexe.

Dans la deuxième partie de ce chapitre, l'algorithme proposé est implémenté avec le langage de programmation C ++, puis une étude numérique est menée sur des problèmes-test générés aléatoirement et sur certaines instances générées à partir du modèle de contrôle optimal présenté dans la première partie de ce chapitre. Les résultats numériques obtenus montrent que notre algorithme est compétitif avec MADH et l'algorithme primal du simplexe de GLPK.

5.1 Problème de contrôle optimal linéaire

La théorie du contrôle optimal consiste à trouver une commande qui optimise une fonctionnelle sur un domaine défini par un système d'équations différentielles et des contraintes de borne sur la commande. Cette théorie est appliquée dans divers domaines des sciences de l'ingénieur : aéronautique, physique, finances, etc. Vu l'importance de cette théorie, plusieurs chercheurs se sont intéressés à la mise au point de méthodes numériques efficaces pour la résolution de ce type de problèmes. Dans cette partie, nous allons utiliser notre algorithme pour la résolution approchée des problèmes de contrôle optimal linéaire.

5.1.1 Présentation du problème

On considère le problème terminal de contrôle optimal suivant :

$$\max J(u) = c^T x(t^*), \quad (5.1)$$

$$\dot{x}(t) = Ax(t) + bu(t), \quad x(t_*) = x_0, \quad (5.2)$$

$$Hx(t^*) = g, \quad (5.3)$$

$$f_* \leq u(t) \leq f^*, \quad t \in \mathbb{T} = [t_*, t^*], \quad (5.4)$$

où $J(u)$ est le critère de qualité, $A \in \mathcal{M}_{n \times n}(\mathbb{R})$ est la matrice dynamique du système, $x(t) \in \mathbb{R}^n$ est le vecteur d'état du système, $b \in \mathbb{R}^n$, $H \in \mathcal{M}_{m \times n}(\mathbb{R})$ est une matrice de rang $m \leq n$, $g \in \mathbb{R}^m$, $u(t) \in \mathbb{R}$ est une commande constante par morceaux, bornée par f_* , $f^* \in \mathbb{R}$, et $c \in \mathbb{R}^n$. L'objectif du problème est la détermination d'une commande admissible u^0 qui avec la trajectoire optimale x^0 , maximise le critère de qualité $J(u)$:

$$J(u^0) = \max_u J(u).$$

La solution du système (5.2) est donnée par

$$x(t) = F(t) \left(x_0 + \int_{t_*}^t F^{-1}(\tau) bu(\tau) d\tau \right), \quad t \in \mathbb{T}, \quad (5.5)$$

où $F(t)$, $t \in \mathbb{T}$, est la solution du système

$$\begin{cases} \dot{F}(t) = AF(t), \\ F(t_*) = I_n, \quad t \in \mathbb{T}. \end{cases}$$

En remplaçant l'expression (5.5) dans le problème (5.1)-(5.4), on trouve

$$\begin{aligned} J(u) = c^T x(t^*) &= c^T F(t^*)x_0 + \int_{t_*}^{t^*} c^T F(t^*)F^{-1}(t)bu(t)dt, \\ HF(t^*)x_0 + \int_{t_*}^{t^*} HF(t^*)F^{-1}(t)bu(t)dt &= g. \end{aligned}$$

Si l'on pose

$$\begin{aligned} C(t) &= c^T F(t^*)F^{-1}(t)b, \\ \phi(t) &= HF(t^*)F^{-1}(t)b, \\ g_0 &= g - HF(t^*)x_0, \end{aligned}$$

on aura alors

$$\max J(u) = c^T F(t^*)x_0 + \int_{t_*}^{t^*} C(t)u(t)dt, \quad (5.6)$$

$$\int_{t_*}^{t^*} \phi(t)u(t)dt = g_0, \quad (5.7)$$

$$f_* \leq u(t) \leq f^*, \quad t \in \mathbb{T} = [t_*, t^*]. \quad (5.8)$$

5.1.2 Discrétisation du problème initial

On choisit un sous-ensemble $\mathbb{T}_h \subset \mathbb{T}$, tel que

$$\mathbb{T}_h = \{t_*, t_* + h, \dots, t^* - h\}, \quad h = \frac{t^* - t_*}{N}, \quad \text{et } N \in \mathbb{N}^*.$$

Faisons une discrétisation de la fonction $u(t)$, $t \in \mathbb{T}$:

$$u(t) \equiv u(\tau), \quad t \in [\tau, \tau + h], \quad \tau \in \mathbb{T}_h.$$

En utilisant cette discrétisation, le problème (5.6)-(5.8) devient

$$\max J(u) = c^T F(t^*)x_0 + \sum_{\tau \in \mathbb{T}_h} q(\tau)u(\tau), \quad (5.9)$$

$$\sum_{\tau \in \mathbb{T}_h} d(\tau)u(\tau) = g_0, \quad (5.10)$$

$$f_* \leq u(\tau) \leq f^*, \quad \tau \in \mathbb{T}_h, \quad (5.11)$$

où

$$q(\tau) = \int_{\tau}^{\tau+h} C(s)ds, \quad d(\tau) = \int_{\tau}^{\tau+h} \phi(s)ds. \quad (5.12)$$

Pour un pas de discrétisation assez petit, la résolution du programme linéaire (5.9)-(5.11) avec MDHPM nous donne une solution optimale qui est une solution approchée pour le problème original (5.1)-(5.4).

5.1.3 Exemple numérique

On considère le problème suivant

$$\begin{cases} \max J(u) = c^T x(2), \\ \dot{x}(t) = Ax(t) + bu(t), x(0) = 0, \\ Hx(2) = g, \\ -1 \leq u(t) \leq 1, t \in \mathbb{T} = [0, 2], \end{cases} \quad (5.13)$$

où

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, b^T = (0, 1), H = (1, -2), g = \frac{1}{2}, c^T = (0, 1).$$

On a

$$\begin{aligned} F(t) &= \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}, F^{-1}(t) = \begin{pmatrix} 1 & -t \\ 0 & 1 \end{pmatrix}, \\ C(t) &= c^T F(2)F^{-1}(t)b = 1, \\ \phi(t) &= HF(2)F^{-1}(t)b = -t. \end{aligned}$$

Considérons la commande admissible suivante :

$$u(t) = \begin{cases} \frac{1}{2}, & \text{si } t \in [0, 1]; \\ -\frac{1}{2}, & \text{si } t \in [1, 2]. \end{cases}$$

La trajectoire correspondante est

$$x(t) = \begin{cases} \begin{pmatrix} \frac{t^2}{4} \\ \frac{t}{2} \end{pmatrix}, & \text{si } t \in [0, 1]; \\ \begin{pmatrix} -\frac{t^2}{4} + t - \frac{1}{2} \\ -\frac{t}{2} + 1 \end{pmatrix}, & \text{si } t \in [1, 2]. \end{cases}$$

On choisit $h = 0.5$, donc $\mathbb{T}_h = \{0, \frac{1}{2}, 1, \frac{3}{2}\}$.

On aura alors

$$d(\tau) = \int_{\tau}^{\tau+h} -s ds = -\frac{1}{2}\tau - \frac{1}{8}, q(\tau) = \int_{\tau}^{\tau+h} ds = \frac{1}{2}.$$

En posant $u_i = u(\tau_i)$, $\tau_i \in T_h$, on obtient le programme linéaire suivant :

$$\begin{aligned} \max z(u) &= \frac{1}{2}u_1 + \frac{1}{2}u_2 + \frac{1}{2}u_3 + \frac{1}{2}u_4, \\ &-\frac{1}{8}u_1 - \frac{3}{8}u_2 - \frac{5}{8}u_3 - \frac{7}{8}u_4 = \frac{1}{2}, \\ &-1 \leq u_i \leq 1, \quad i = 1, 2, 3, 4. \end{aligned}$$

Ce programme linéaire s'écrit sous la forme matricielle suivante :

$$\max z(u) = q^T u, \quad Au = b, \quad L \leq u \leq U, \quad (5.14)$$

avec

$$\begin{aligned} q &= \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right)^T, \quad u = (u_1, u_2, u_3, u_4)^T, \quad b = \frac{1}{2}, \quad A = \left(-\frac{1}{8}, -\frac{3}{8}, -\frac{5}{8}, -\frac{7}{8}\right), \\ L &= (-1, -1, -1, -1)^T, \quad U = (1, 1, 1, 1)^T. \end{aligned}$$

Résolvons le programme linéaire (5.14) en utilisant la méthode MDHPM avec $\eta = 1$. Prenons la solution réalisable de support initiale $\{u, J_B\}$ suivante :

$$u = \left(\frac{1}{2}, \frac{1}{2}, \frac{-1}{2}, \frac{-1}{2}\right)^T, \quad z(u) = 0, \quad J_B = \{3\}, \quad J_N = \{1, 2, 4\}.$$

Itération 1

- Calcul du vecteur des multiplicateurs et du vecteur des coûts réduits :

On a

$$A_B^{-1} = -\frac{8}{5}, \quad A_N = \left(\frac{-1}{8}, \frac{-3}{8}, \frac{-7}{8}\right).$$

Donc

$$\pi^T = q_B^T A_B^{-1} = \frac{-4}{5}, \quad \Delta = A^T \pi - c = \left(\frac{-2}{5}, \frac{-1}{5}, 0, \frac{1}{5}\right)^T.$$

- Calcul de l'estimation de suboptimalité :

$$\beta = \sum_{\Delta_j > 0, j \in J_N} \Delta_j (u_j - L_j) + \sum_{\Delta_j < 0, j \in J_N} \Delta_j (u_j - U_j) = \frac{2}{5}.$$

- Calcul des vecteur $\eta(u - U)$, $\eta(u - L)$ et les différents ensembles :

$$\eta(u - U) = \left(\frac{-1}{2}, \frac{-1}{2}, \frac{-3}{2}, \frac{-3}{2}\right)^T, \quad \eta(u - L) = \left(\frac{3}{2}, \frac{3}{2}, \frac{1}{2}, \frac{1}{2}\right)^T.$$

$$J_{NE}^+ = \emptyset, \quad J_{NE}^- = \emptyset, \quad J_{NI}^+ = \{4\}, \quad J_{NI}^- = \{1, 2\}, \quad J_{NR}^+ = \emptyset, \quad J_{NR}^- = \emptyset, \quad J_{N0} = \emptyset.$$

- La valeur de l'estimation μ :

$$\mu = - \sum_{j \in J_{NE}^+} \Delta_j (u_j - L_j) - \sum_{j \in J_{NE}^-} \Delta_j (u_j - U_j) + \frac{1}{\eta} \sum_{j \in J_{NE}^+ \cup J_{NE}^-} \Delta_j^2 = 0.$$

- Calcul de la direction d'amélioration d :

On a

$$d = \left(\frac{1}{2}, \frac{1}{2}, \frac{3}{10}, \frac{-1}{2}\right)^T.$$

- Calcul du pas le long de la direction d et de la nouvelle solution \bar{u} :

On a $\theta^0 = \min\{5, 1\} = 1$. Alors la nouvelle solution est

$$\bar{u} = \left(1, 1, -\frac{1}{5}, -1\right)^T, \quad z(\bar{u}) = z(u) + \theta^0(\beta + \mu) = \frac{2}{5} = 0.4.$$

Cette solution est optimale pour le programme linéaire (5.14). Par conséquent, la commande

$$u(t) = \begin{cases} 1 & \text{si } t \in [0, 1[, \\ -\frac{1}{5} & \text{si } t \in [1, \frac{3}{2}[, \\ -1 & \text{si } t \in [\frac{3}{2}, 2[. \end{cases}$$

est une solution optimale approchée du problème continu (5.13).

Afin de trouver une bonne solution approchée, nous avons résolu le programme linéaire (5.14), pour $N = 1000$ et $h = 0.002$. La solution optimale obtenue est $z(u) = 0.449489$.

5.1.4 Planification de la production : un problème de contrôle optimal

Une entreprise souhaite déterminer le taux de production sur l'horizon de planification des T prochaines semaines de telle sorte que la demande connue soit satisfaite et que le coût total de production et de stock soit minimisé. Supposons que le taux de la demande connu au temps t soit $g(t)$ et, de la même manière, désignons le taux de production et le niveau des stocks au temps t par $z(t)$ et $y(t)$, respectivement. De plus, supposons que le niveau du stock initial au temps 0 est y_0 et que le niveau du stock souhaité à la fin de l'horizon de planification soit égal à y_T . Supposons que le coût de stock est proportionnel aux nombres d'unités en stock, de sorte que le coût de stock soit égal à $k_1 \int_0^T y(t) dt$, où $k_1 > 0$ est connu. Supposons également que le coût de production soit proportionnel au taux de production, et soit donc donné par $k_2 \int_0^T z(t) dt$. Le coût total est alors

$$\int_0^T [k_1 y(t) + k_2 z(t)] dt.$$

Notons également que le stock à tout moment t est donné par la relation

$$y(t) = y_0 + \int_0^t [z(\tau) - g(\tau)] d\tau, \quad t \in [0, T].$$

Supposons que toute demande doit être satisfaite. De plus, supposons que la capacité de fabrication actuelle limite le taux de production de sorte qu'il ne dépasse à aucun moment la valeur v . Supposons également que la capacité de stockage disponible limite le niveau du stock à une valeur inférieure ou égale à une valeur maximale w . Par conséquent, le problème d'ordonnement de la production peut être énoncé comme suit :

$$\begin{aligned}
\min \psi(T) &= \int_0^T [k_1 y(t) + k_2 z(t)] dt, \\
y(t) &= y_0 + \int_0^t [z(\tau) - g(\tau)] d\tau, \quad t \in [0, T], \\
y(T) &= y_T, \\
0 \leq y(t) &\leq w, \quad t \in [0, T], \\
0 \leq z(t) &\leq v, \quad t \in [0, T].
\end{aligned} \tag{5.15}$$

Discrétisation du problème

Le modèle précédent est un problème de contrôle optimal linéaire continu, où la variable de contrôle est le taux de production $z(t)$ et la variable d'état est le niveau de stock $y(t)$. Le problème peut être approché par un programme linéaire en discrétisant les variables continues z et y . Pour ce faire, l'horizon de planification $[0, T]$ est divisé en n périodes plus petites :

$$[0, h], [h, 2h], [2h, 3h], \dots, [(n-1)h, nh], \text{ où } nh = T.$$

Notons le taux de production, le niveau des stocks et le taux de la demande pendant la période j par z_j , y_j et g_j , respectivement et supposons qu'ils sont constants sur chaque période. Le problème d'ordonnement de la production peut alors être approché par le programme linéaire suivant :

$$\begin{aligned}
\min f(y, z) &= \sum_{j=1}^n (k_1 h) y_j + \sum_{j=1}^n (k_2 h) z_j, \\
y_j &= y_{j-1} + (z_j - g_j) h, \quad j = 1, \dots, n, \\
y_n &= y_T, \\
0 \leq y_j &\leq w, \quad j = 1, \dots, n-1, \\
0 \leq z_j &\leq v, \quad j = 1, \dots, n.
\end{aligned} \tag{5.16}$$

Les relations

$$y_j = y_{j-1} + (z_j - g_j) h, \quad j = 1, \dots, n; \quad y_n = y_T$$

s'écrivent aussi :

$$\begin{aligned}
 y_1 &= y_0 + (z_1 - g_1)h, \\
 y_2 &= y_0 + (z_1 - g_1)h + (z_2 - g_2)h, \\
 y_3 &= y_0 + (z_1 - g_1)h + (z_2 - g_2)h + (z_3 - g_3)h, \\
 &\vdots \\
 y_T &= y_0 + (z_1 - g_1)h + (z_2 - g_2)h + (z_3 - g_3)h + \cdots + (z_n - g_n)h.
 \end{aligned} \tag{5.17}$$

Le système (5.17), peut être écrit sous la forme matricielle suivante :

$$Ax = b,$$

où

$$A = (e_1, e_2, \dots, e_{n-1}, -hE) \in \mathcal{M}_{n \times 2n-1}(\mathbb{R}), \quad b = y_0 e - hEg - y_T e_n \in \mathbb{R}^n,$$

avec

$$E = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix} \in \mathcal{M}_{n \times n}(\mathbb{R}),$$

$$e = (1, 1, \dots, 1)^T \in \mathbb{R}^n, \quad x = (x_i, i = 1, 2, \dots, 2n-1),$$

$$x_i = y_i, \quad 1 \leq i \leq n-1; \quad x_{n+i-1} = z_i, \quad 1 \leq i \leq n,$$

et e_j représente le j -ième vecteur-colonne de la matrice identité I_n .

Si l'on pose

$$c_i = -k_1 h, \quad l_i = 0, \quad u_i = w, \quad \text{pour } 1 \leq i \leq n-1,$$

$$c_i = -k_2 h, \quad l_i = 0, \quad u_i = v, \quad \text{pour } n \leq i \leq 2n-1,$$

on obtiendra alors le programme linéaire :

$$\begin{aligned}
 \max f(x) &= c^T x + f_0, \\
 Ax &= b, \quad l \leq x \leq u,
 \end{aligned} \tag{5.18}$$

$$\text{où } f_0 = -k_1 h y_T.$$

5.2 Résultats expérimentaux

Le "GNU Linear Programming Kit" (GLPK)¹ est un outil performant pour résoudre des problèmes d'optimisation linéaire à variables continues ou mixtes (entières et continues). Ce

1. GLPK a été développé par Andrew Makhorin de l'Institut d'aviation de Moscou. La première version publique a été publiée en octobre 2000.

kit est composé d'un langage de modélisation GNU MathProg et d'une librairie de fonctions C utilisant le solveur GlpSOL. L'avantage de ce kit est d'être en libre accès et relativement facile à installer et à utiliser.

GLPK est une librairie de fonctions C qui utilisent les principales méthodes de résolution de problème d'optimisation (méthode du simplexe, branch and bound, méthodes de points intérieurs, etc.) ainsi que tous les outils nécessaires aux créations d'objets, à l'appel des solveurs et à l'affichage des résultats.

Afin de comparer l'algorithme proposé (MDHPM) avec la Méthode Primale du Simplexe (MPS) implémentée dans GLPK et la Méthode Adaptée à Direction Hybride (MADH), une implémentation utilisant le langage de programmation C++ a été développée. Dans notre implémentation, le vecteur des multiplicateurs π , les composantes de base de la direction réalisable d_B et les composantes non basiques de la direction duale t_N sont calculées en résolvant trois systèmes d'équations linéaires.

Nous avons testé les trois algorithmes sur deux ensembles de problèmes-test : un ensemble de programmes linéaires générés aléatoirement et un ensemble de problèmes de contrôle optimal modélisant le problème présenté à la [sous-section 5.1.4](#).

5.2.1 Problèmes de PL générés aléatoirement

Les problèmes de PL générés aléatoirement sont écrits sous la forme suivante :

$$\max z = c^T x, \quad Ax \leq b, \quad l \leq x \leq u, \quad (5.19)$$

où $A = (a_{ij}) \in \mathcal{M}_{m,n}(\mathbb{R})$ dont 5% de ses éléments sont non nuls, $x, c, l, u \in \mathbb{R}^n$, $b \in \mathbb{R}^m$. Les nombres c_j, a_{ij}, b_j, l_j et u_j sont générés aléatoirement avec la distribution uniforme. Nous avons considéré des problèmes avec une matrice de contraintes A de dimension $m \times m$ et $m \times 2m$, où $m \in \{200, 400, 600, 800, 1000\}$. Pour chaque taille, nous avons généré dix problèmes-test. Au total, nous avons généré 100 problèmes de PL.

5.2.2 Problèmes de PL particuliers

Le problème de contrôle optimal précédent (5.15) est transformé en problème de PL suivant :

$$\max c^T x, \quad Ax = b, \quad l \leq x \leq u, \quad (5.20)$$

où A est une matrice de dimension $n \times (2n - 1)$, n est le nombre de sous-intervalles de discrétisation de l'intervalle $[0, T]$; $b \in \mathbb{R}^n$, $x, l, u \in \mathbb{R}^{2n-1}$. Nous fixons $T = 20$ et pour chaque dimension $n = 50, 200, 400, 600, 800, 1000$, nous générons 10 problèmes-test. Au total, nous avons généré 60 problèmes-test pour le problème de contrôle optimal.

5.2.3 Résultats obtenus sur les problèmes-test

Nous présentons les résultats numériques de la comparaison de notre méthode (MDHPM) : l'algorithme 3.2 avec la MPS (l'algorithme 2.2) et la MADH (l'algorithme 3.1). De plus, nous avons posé $\eta = 1$ dans MADH et MDHPM. Nous avons résolu l'ensemble des 160 problèmes-test avec les trois algorithmes considérés, sur un ordinateur avec 4Go de RAM et un processeur Pentium (R) Dual-Core CPU E5700 @ 3.00GHz 2.99GHz, fonctionnant sous le système d'exploitation 64 bits Windows 8.1. Les résultats numériques sont présentés dans les tableaux 5.1, 5.2 et 5.3, où "cput" et "nit" représentent respectivement le temps CPU moyen en secondes et le nombre moyen d'itérations des dix problèmes générés pour chaque taille.

Nous traçons le temps CPU et le nombre d'itérations pour les trois algorithmes. Les graphiques sont présentés dans les figures 5.1, 5.2, 5.3 et 5.4. À partir des différents tableaux et graphiques, nous pouvons voir que l'algorithme MDHPM est compétitif avec la méthode primale du simplexe de GLPK en termes de temps CPU et en terme de nombre d'itérations sur les problèmes-test considérés. De plus, nous pouvons constater que l'algorithme proposé dans cette thèse est plus efficace que MADH. Cependant, des expériences numériques sur d'autres ensembles de problèmes-test tels que ceux de la librairie Netlib [37] sont nécessaires pour obtenir des conclusions plus raffinées.

$m \times m$	<i>MPS</i>		<i>MADH</i>		<i>MDHPM</i>	
	nit	cput	nit	cput	nit	cput
200 × 200	123,6	0,063	85,8	0,050	63,2	0,036
400 × 400	354,0	0,251	97,0	0,072	73,5	0,052
600 × 600	549,9	0,409	142,1	0,113	98,5	0,079
800 × 800	772,1	0,742	236,8	0,231	171,4	0,161
1000 × 1000	1013,7	1,019	273,4	0,394	215,2	0,333

TABLE 5.1 – Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test générés aléatoirement de dimension $m \times m$

$m \times 2m$	<i>MPS</i>		<i>MADH</i>		<i>MDHPM</i>	
	nit	cput	nit	cput	nit	cput
200 × 400	204,8	0,157	88,5	0,078	66,2	0,055
400 × 800	596,5	0,437	147,8	0,120	106,7	0,085
600 × 1200	1576,7	1,080	190,7	0,145	145,3	0,107
800 × 1600	2067,9	2,121	299,0	0,335	208,3	0,229
1000 × 2000	3085,6	4,926	358,1	0,607	282,6	0,440

TABLE 5.2 – Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test générés aléatoirement de dimension $m \times 2m$

$m \times 2m$	<i>MPS</i>		<i>MADH</i>		<i>MDHPM</i>	
	nit	cput	nit	cput	nit	cput
50 × 100	36,6	0,021	5,5	0,005	4,0	0,004
200 × 400	172,9	0,176	12,0	0,036	9,6	0,028
400 × 800	354,6	0,642	19,8	0,109	16,8	0,085
600 × 1200	486,2	1,236	27,0	0,231	22,2	0,202
800 × 1600	709,3	3,322	36,6	0,455	31,2	0,404
1000 × 2000	860,2	5,752	45,2	0,838	37,9	0,727

TABLE 5.3 – Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test de contrôle optimal

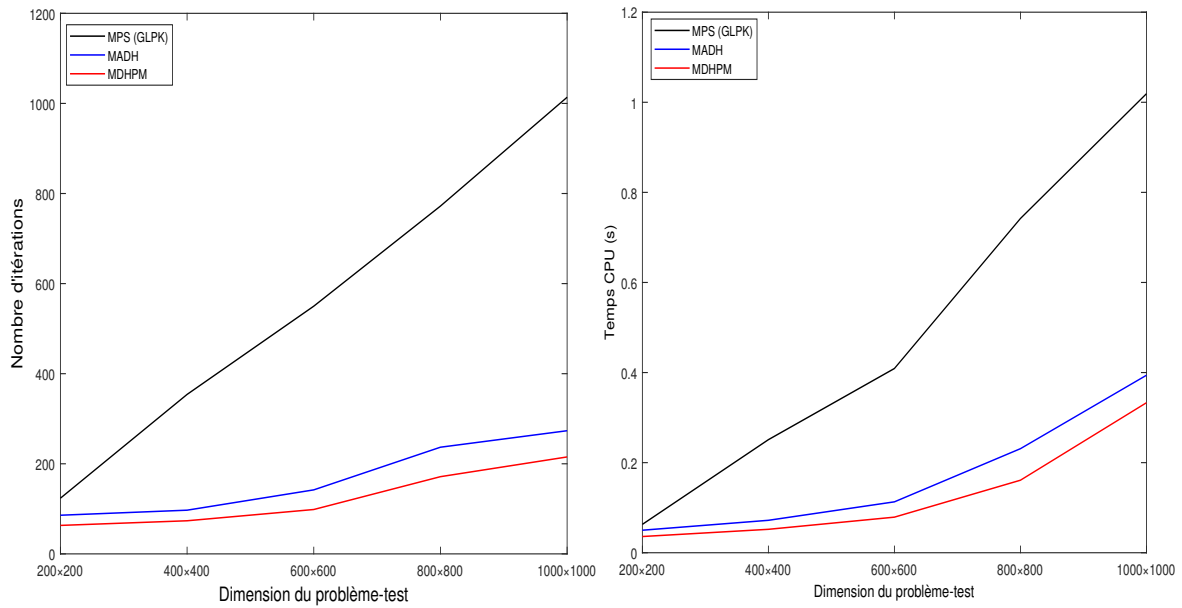


FIGURE 5.1 – Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test générés aléatoirement de dimension $m \times m$

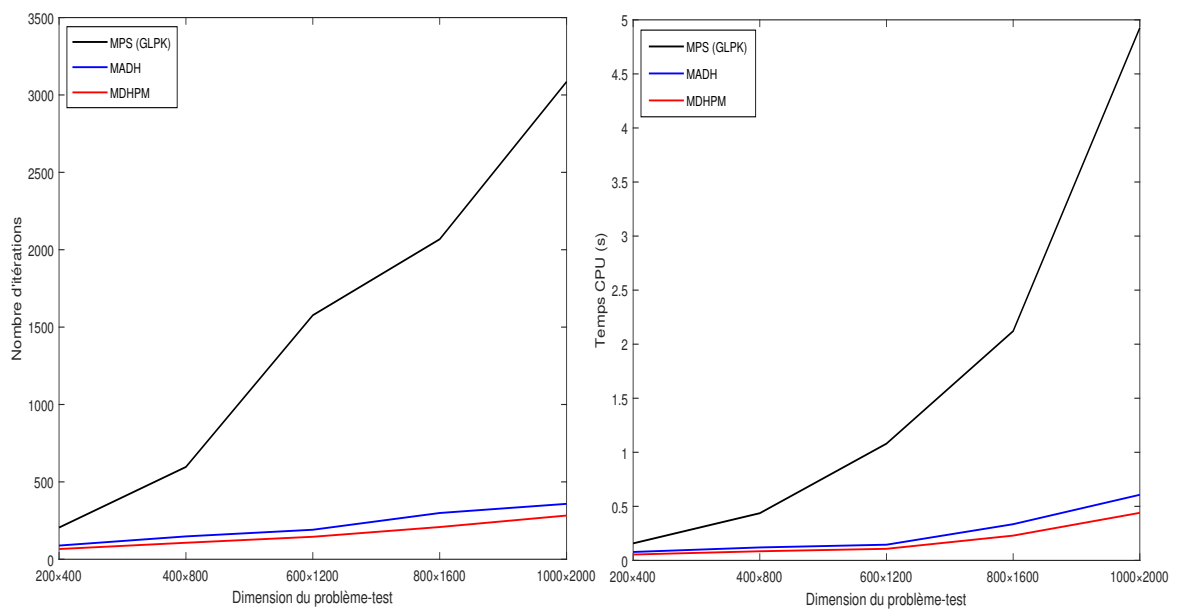


FIGURE 5.2 – Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes-test générés aléatoirement de dimension $m \times 2m$

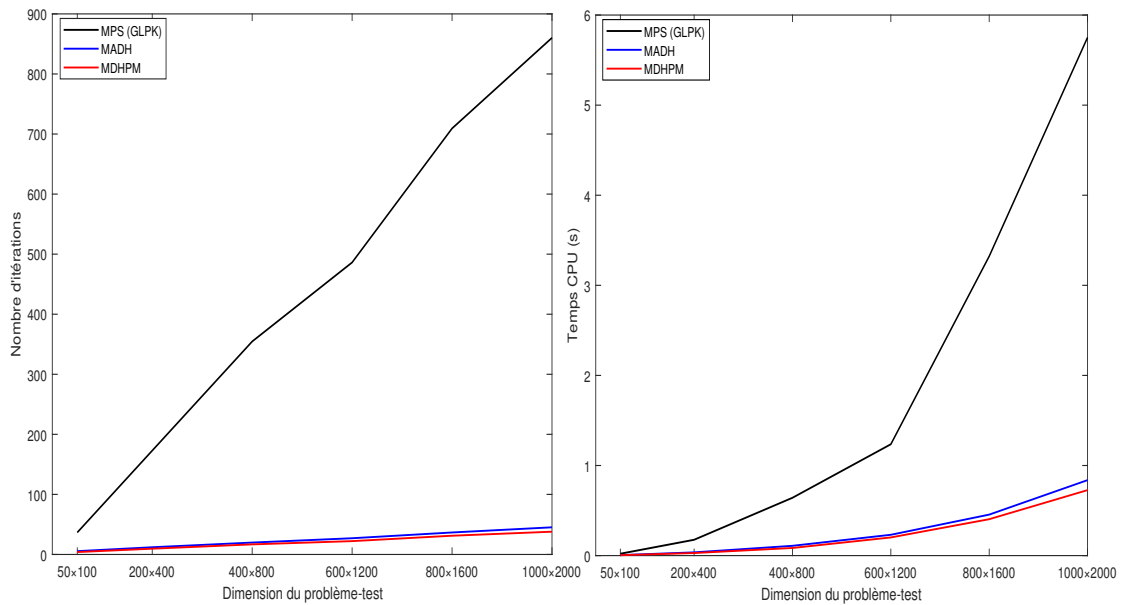


FIGURE 5.3 – Le nombre moyen d'itérations et le temps CPU moyen des trois algorithmes pour les problèmes de contrôle optimal

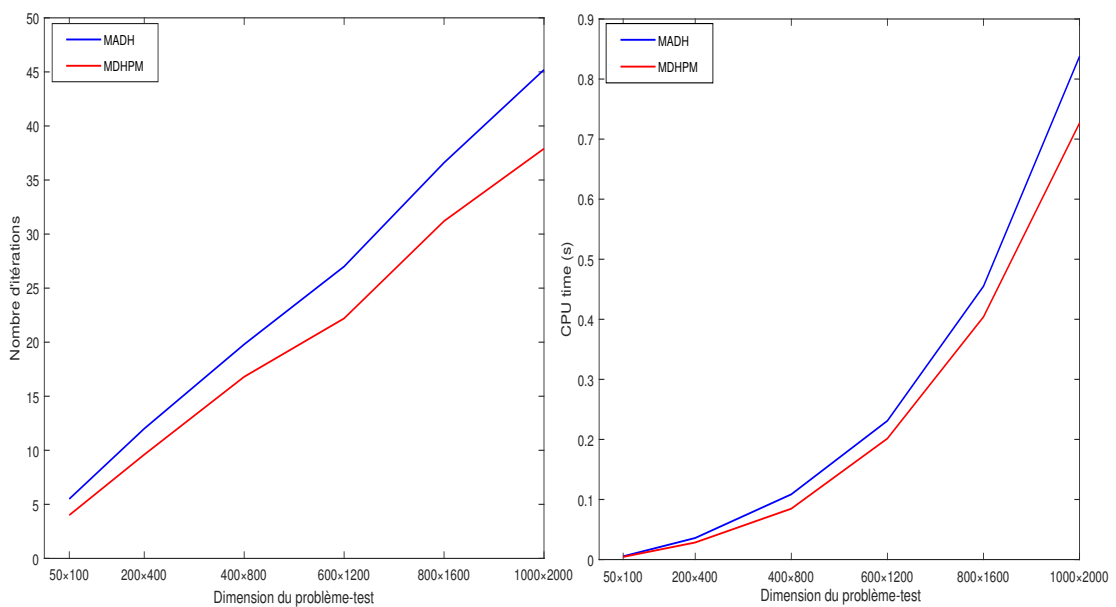


FIGURE 5.4 – Le nombre moyen d'itérations et le temps CPU moyen des deux algorithmes MADH et MDHPM pour les problèmes-test de contrôle optimal

Conclusion

Dans ce travail, nous avons proposé une nouvelle méthode à direction hybride pour résoudre des problèmes de programmation linéaire à variables bornées. Les conditions utilisées pour caractériser l'optimalité de la solution courante sont basées sur l'estimation de suboptimalité, qui est mise à jour à chaque itération en utilisant une formule plus générale que celle proposée en [36]. Afin d'améliorer la solution réalisable courante, nous avons construit une nouvelle direction réalisable hybride qui donne une meilleure amélioration locale de la fonction objectif que la direction de l'algorithme MADH proposé dans [14] (Algorithme 3.1). De plus, la règle du pas multiple est décrite pour changer le support courant dans notre méthode et une procédure qui effectue la mise à jour du paramètre η est utilisée.

Dans le but de résoudre les problèmes de programmation linéaire écrits sous forme générale sans connaissance au préalable d'une solution réalisable de support ou de base initiale, nous avons proposé une nouvelle technique d'initialisation des différents algorithmes, qui évite l'utilisation des variables artificielles, et ce, en résolvant un problème auxiliaire ayant le même nombre de variables que le problème original. Cette technique permet aussi de conclure si le problème original est non réalisable. En outre, nous avons montré comment appliquer notre méthode pour trouver une solution approchée pour certains problèmes de contrôle optimal linéaire.

Afin de comparer notre algorithme à l'algorithme primal du simplexe implémenté dans le solveur de programmation linéaire open source GLPK et l'algorithme MADH, nous avons développé une implémentation avec le langage de programmation C++. Les résultats numériques obtenus sur des problèmes-test générés aléatoirement et certaines instances d'un problème de contrôle optimal linéaire modélisant un problème pratique de planification de la production montrent que notre algorithme est compétitif avec MPS et MADH.

Dans un travail futur, nous appliquerons la procédure de calcul de support initial (Procédure 4.1) et la technique de recherche d'une solution réalisable initiale proposée dans la sous-section 4.2.2 afin d'initialiser notre algorithme avec une bonne SRS initiale, nous comparerons ensuite ses performances avec l'algorithme primal du simplexe sur les problèmes-test de la librairie Netlib [37].

Bibliographie

- [1] M. L. Balinski, P. Wolfe. Nondifferentiable optimization. *Mathematical programming study, vol. 3, North-Holland Publishing Company, Amsterdam, 1975.*
- [2] R. G. Bartle. The elements of real analysis. *John Wiley and Sons, Inc, New York, second ed., 1976.*
- [3] M. S. Bazaraa, H. D. Sherali, C. M. Shetty. Nonlinear programming : Theory and Algorithms. *John Wiley and Sons, Inc., New York, 1979.*
- [4] M. S. Bazaraa, J. J. Jarvis, H. D. Sherali. Linear programming and network flows. *John Wiley and Sons, Inc, Fourth Edition, Canada, 2010.*
- [5] E. M. L. Beale. An alternative method for linear programming. *Mathematical Proceedings of the Cambridge Philosophical Society, vol. 50, no. 4, pp. 513-523, October 1954.*
- [6] M. Bentobache, M. O. Bibi. A Two-phase Support Method for Solving Linear Programs : Numerical Experiments. *Mathematical Problems in Engineering, Article ID 482193, 28 pages doi :10.1155/2012/482193, 2012.*
- [7] M. Bentobache. Sur les méthodes mathématiques de la programmation linéaire et quadratique. *Thèse de doctorat, Université de Béjaïa, Algérie, 2013.*
- [8] M. Bentobache, M. O. Bibi. A hybrid direction algorithm with long step rule for linear programming : Numerical experiments. *Advances in Intelligent Systems and Computing, vol. 359, pp. 333-344, 2015.*
- [9] M. Bentobache, M. O. Bibi. Méthodes numériques de la programmation linéaire et quadratique : théorie et algorithmes. *Presses Académiques Francophones, Allemagne, 2016.*
- [10] L. D. Berkovitz. Convexity and optimization in \mathbb{R}^n . *John Wiley and Sons, Inc, New York, 2002.*
- [11] J. F. Maurras. Programmation Linéaire, Complexité : Séparation et Optimisation. *Springer, New York, 2002.*

- [12] M. O. Bibi. Optimization of a linear dynamic system with double terminal constraints on the trajectories. *Optimization*, vol. 30, no. 4, pp. 359–366, 1994.
- [13] M. O. Bibi. Support method for solving a linear-quadratic problem with polyhedral constraints on control. *Optimization*, vol. 37, no. 2, pp. 139–147, 1996.
- [14] M. O. Bibi, M. Bentobache. A hybrid direction algorithm for solving linear programs. *International Journal of Computer Mathematics*, vol. 92, no. 1, pp. 201–216, 2015.
- [15] M. O. Bibi, N. Ikheneche, M. Bentobache. A hybrid direction algorithm for solving a convex quadratic problem. *International Journal of Mathematics in Operational Research*, vol. 16, no. 2, pp. 159–178, 2020.
- [16] M. Bierlaire. Introduction à l'optimisation différentiable. *Presses polytechniques et universitaires romandes*, Italie, 2006.
- [17] R. E. Bixby. Implementing the simplex method : The initial basis. *ORSA Journal on Computing*, vol. 4, no. 3, pp. 1-18, 1992.
- [18] B. Brahmi, M. O. Bibi. Dual support method for solving convex quadratic programs. *Optimization*, vol. 59, no. 6, pp. 851–872, 2010.
- [19] D. M. Cardoso, J. N. Clímaco. The generalized simplex method. *Operations Research Letters*, vol. 12, pp. 337-448, 1992.
- [20] A. Charnes. Optimality and Degeneracy in Linear Programming. *Econometrica*, vol. 20, no. 2, pp. 160-170, 1952.
- [21] E. K. P. Chong and S. H. Žak. An Introduction to Optimization. *John Wiley and Sons, Inc*, New York, second ed, 2001.
- [22] G. B. Dantzig. Maximization of a linear function of variables subject to linear inequalities. In *T. C. Koopmans (ed.), Activity Analysis of Production and Allocation*, John Wiley & Sons, Inc., New York, pp. 339-347, 1951.
- [23] G. B. Dantzig, A. Orden and P. Wolfe. The generalized simplex method for minimizing a linear form under linear inequality restraints. *Pacific Journal of Mathematics*, vol. 5, no. 2, 183-195, 1955.
- [24] G. B. Dantzig and P. Wolfe. The decomposition algorithm for linear programming. *Econometrica*, vol. 29, pp. 767-778, 1961.
- [25] G. B. Dantzig. Linear programming and extensions. *Princeton University Press*, Princeton, N.J, 1963.

- [26] K. Djeloud, M. Bentobache, M. O. Bibi. A hybrid direction method for solving linear programs with bounded variables. *COSI'2017, University of Bouira, Algeria*, 14-16 May, 2013.
- [27] K. Djeloud, Algorithmes à direction hybride pour la résolution des programmes linéaires à variables bornées. *Doctoriales de Mathématiques, ENS Constantine, Algérie*, 28-31 Octobre 2017.
- [28] K. Djeloud, M. Bentobache and M. O. Bibi. A new method with hybrid direction for linear programming. *Concurrency and Computation : Practice and Experience*, <https://doi.org/10.1002/cpe.5836>, 2020.
- [29] Y. Dodge, S. G. Weber et J. P. Renfer. Optimisation appliquée. *Springer-Verlag, France*, 2005.
- [30] V. Dua, E. N. Pistikopoulos. An Algorithm for the Solution of Multiparametric Mixed Integer Linear Programming Problems. *Annals of Operations Research*, vol. 99, pp. 123–139, 2000.
- [31] Y. R. Fan and G. H. Huang. A Robust Two-Step Method for Solving Interval Linear Programming Problems within an Environmental Management Context. *Journal of Environmental Informatics*, vol. 19, no. 1, pp. 1-9, 2012.
- [32] R. Gabasov and F. M. Kirillova. Methods of linear programming. *Vol. 1, 2 and 3, Edition of the Minsk University*, 1977, 1978 and 1980 (in Russian).
- [33] R. Gabasov, F. M. Kirillova and V. M. Raketsky, On methods for solving the general problem of convex quadratic programming. *Soviet. Math. Dokl.*, Vol. 23, pp.653–657, 1981.
- [34] R. Gabasov, F. M. Kirillova and O. I. Kostyukova. Solution of linear quadratic extremal problems. *Soviet. Math. Dokl.*, Vol. 31, pp.99–103, 1985.
- [35] R. Gabasov. Adaptive method of linear programming. *Preprints of the university of karlsruhe*, Institute of statistics and Mathematics, Germany, 1993.
- [36] R. Gabasov, F. M. Kirillova, S. V. Prischepova. Optimal Feedback Control. *Springer-Verlag, London*, 1995,
- [37] D. M. Gay. Electronic Mail Distribution of Linear Programming Test Problems. *Mathematical Programming Society COAL Newsletter*, vol. 13, pp. 10-12 , 1985. Data available at <http://www.numerical.rl.ac.uk/cute/netlib.html>.
- [38] G. F. Hadley and M. A. Simonnard. A simplified two-phase technique for the simplex method. *John Wiley and Sons, Inc*, vol. 6, pp. 221-226, 1959.

- [39] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, vol. 4, pp. 373-395, 1984.
- [40] N. Khimoum and M. O. Bibi. Primal-dual method for solving a linear-quadratic multi-input optimal control problem. *Optimization Letters*, vol 14, pp. 653–669, 2020.
- [41] T. C. Koopmans. Activity Analysis of Production and Allocation. *John Wiley and Sons*, New York, 1957.
- [42] E. A. Kostina and O. I. Kostyukova. An algorithm for solving quadratic programming problems with linear equality and inequality constraints. *Computational Mathematics and Mathematical Physics*, vol. 41, no. 7, pp. 960–973, 2001.
- [43] E. A. Kostina. The long step rule in the bounded-variable dual simplex method : Numerical experiments. *Mathematical Methods of Operations Research*, vol 55, pp. 413–429, 2002.
- [44] L. V. Kantorovich. Mathematical Methods in the Organization and Planning of Production. *Publication House of the Leningrad State University, 1939, Translated in management science*, vol. 6, pp. 366-422, 1960.
- [45] C. E. Lemke. The dual method of solving the linear programming problem. *John Wiley and Sons, Inc.*, vol 1, pp. 36-47, 1954.
- [46] S. Lim and S. Park. LPAKO : A simplex-based linear programming program. *Optimization Methods and Software*, vol. 17, pp. 717-745, 2002.
- [47] A. Makhorin. GNU Linear Programming Kit. *Reference Manual Version 4.9*, Draft Edition, 2006.
- [48] I. Maros, G. Mitra. Strategies for creating advanced bases for large-scale linear programming problems. *INFORMS Journal on Computing-Spring*, vol. 10, no. 2, pp. 248-260, 1998.
- [49] S. Mehrotra. On the implementation of a primal-dual interior point method. *Siam J. Optimization*, vol. 2, no. 4, pp. 575-601, 1992.
- [50] P. Q. Pan. A basis-deficiency-allowing variation of the simplex method. *Computers and Mathematics with applications*, vol. 36, no. 3, pp. 33-53, 1998.
- [51] K. Paparrizos, N. Samaras and G. Stephanides. An efficient simplex type algorithm for sparse and dense linear programs. *European Journal of Operational Research*, vol. 148, pp. 323-334, 2003.
- [52] H. Roudier. Algèbre linéaire. *Vuibert*, Paris, 2008.

-
- [53] W. Rudin. Principles of Mathematical Analysis. *McGraw-Hill Book Co*, third ed, New York, 1976.
- [54] R. J. Vanderbei. Linear programming : Foundations and Extensions. *Kluwer academic publishers*, Princeton University, 2001.
- [55] M. A. Zaitri, M. O. Bibi and M. Bentobache. A hybrid direction algorithm for solving optimal control problems. *Cogent Mathematics and Statistics*, vol. 6, no. 1, doi : 10.1080/25742558.2019.1612614, 2019.

Index

- addition, 13
- algorithme, 32, 34
- application, 25
- approchée, 89

- base, 14
- base canonique, 16
- base finie, 15
- branch and bound, 97

- changement de support, 57
- classique, 56
- coefficients de la matrice, 15
- coefficients diagonaux, 15
- colonnes, 15
- combinaison linéaire, 14
- commande, 89
- commande admissible, 90
- commande constante par morceaux, 90
- compatible, 18
- constantes, 14
- continue, 25
- contrôle optimal linéaire, 89
- contradiction, 56
- contraintes, 27, 69
- contraintes d'égalités, 69
- contraintes d'inégalités, 69
- converge, 25
- coordonnées, 15
- critère de qualité, 90
- croissant, 61

- demi-espace, 26
- dimension finie, 15

- diminution, 66
- direction admissible, 55
- direction d'amélioration, 48, 55
- direction duale, 49, 57
- discrétisation, 91, 95

- ensemble convexe, 26
- ensembles d'indices, 52
- entreprise, 94
- espace vectoriel, 13
- espace vectoriel de dimension finie, 15
- espace vectoriel normé, 25
- espace vectoriel normé de dimension finie, 25
- estimation d'optimalité, 45
- estimation de suboptimalité, 44, 57
- exemple numérique, 63

- factorisation LU, 19
- fermé, 25
- figures, 98
- fonction objectif, 27
- forme standard, 28, 70
- formule, 75
- formule d'accroissement, 43
- fractionner, 69

- générale, 68
- GLPK, 89
- GNU Linear Programming Kit, 96
- GNU MathProg, 97

- homogène, 18
- hyperplan, 26

- inégalité, 53
- incompatible, 18
- indice entrant, 36
- indices de base, 29
- indices hors-base, 29
- infinie, 34, 35
- inversement, 27
- itération, 56
- itération de l'algorithme, 52

- l'accroissement de la fonction objectif, 55
- le pas dual, 65
- le pas le long, 54
- le schéma de la méthode à direction hybride, 62
- librairie, 97
- lignes, 15
- linéairement dépendants, 14
- linéairement indépendants, 14

- méthode adaptée, 47, 63
- méthode adaptée à direction hybride, 45
- méthode primale du simplexe, 31
- méthodes de points intérieurs, 97
- méthodes numériques, 89
- matrice, 15
- matrice dynamique, 90
- matrice antisymétrique, 16
- matrice augmentée, 18
- matrice carrée, 15
- matrice de base, 29
- matrice des contraintes, 28
- matrice diagonale, 15
- matrice identité, 16
- matrice nulle, 16
- matrice symétrique, 16
- matrice transposée, 16
- matrice triangulaire inférieure, 16
- matrice triangulaire supérieure, 16
- matrice-colonne, 15

- matrice-ligne, 15
- meilleure, 66
- mise à jour, 32, 56
- mise à jour classique, 61
- multiplication, 13

- non-dégénérée, 29, 43
- non-linéaires, 27
- non-négatif, 44
- non-singulière, 17, 29, 32, 33
- norme, 25
- norme de Hölder, 25
- normes fondamentales, 25

- optimalité, 43
- optimum, 49
- orthogonale, 57

- partie bornée, 25
- partie compacte, 26
- phase, 79
- PL générés aléatoirement, 97
- PL particuliers, 97
- planification de la production, 94
- plusieurs variables artificielles, 68
- point extrême, 26
- polytope, 26
- problème de maximisation, 27
- problème de minimisation, 27
- problème de programmation linéaire, 27, 28
- procédure, 61
- procédure de mise à jour, 61
- processeur, 98
- programmation linéaire en nombres entiers, 27
- programmation mathématique, 27
- programmation non-linéaire, 27
- programmation quadratique, 27
- pseudo-solution, 29

- quadratique, 27

- quantité, 53
- résultats expérimentaux, 96
- règle du pas multiple, 61
- règle du pas simple, 57
- rang d'une matrice, 17
- sans perte de généralité, 69
- scalaires, 13
- seconds membres, 28
- segment, 26
- seule variable artificielle, 68
- signe, 58
- solution, 18
- solution de base initiale, 37
- solution de base réalisable, 29, 30, 33
- solution de base réalisable optimale, 32, 34
- solution optimale, 29
- solution optimale approchée, 94
- solution réalisable, 29, 47
- solution réalisable optimale, 46
- solveur Glsol, 97
- sommet, 26
- sous-matrice, 29
- SRS initiale, 62
- suboptimalité, 43
- suite d'éléments, 25
- support, 56
- système d'équations linéaires, 18
- système d'exploitation, 98

- tableaux, 98
- technique, 68
- trajectoire optimale, 90

- variable d'entrée, 31, 34
- variable fixe, 33
- variable libre, 32
- variable sortante, 34
- variables d'écart, 69
- variables de base, 29
- variables hors-base, 29
- variables non négatives, 32
- variables originales, 69
- vecteur d'état, 90
- vecteur des bornes inférieures, 28
- vecteur des bornes supérieures, 29
- vecteur des coûts réduits, 49, 57
- vecteur des multiplicateurs, 64
- vecteurs, 13

ملخص

في هذا العمل، اقترحنا خوارزمية جديدة لحل مشاكل البرمجة الخطية ذات المتغيرات المحدودة. تنتقل هذه الخوارزمية من الحل الحالي إلى حل أفضل باستعمال اتجاه هجين جديد. يعطي هذا الاتجاه تحسينًا محليًا لدالة الهدف أفضل من الإتجاه المستعمل في الخوارزمية MADH المقترحة مسبقًا من طرف م.و. ببيبي و م. بن طوباش في 2015. لإيقاف الخوارزمية المقترحة، يتم استخدام معيار شبه الأمثلة ويتم تطبيق قاعدة الخطوات المتعددة لتغيير السند الحالي. تم برمجة الخوارزمية المقترحة باستخدام لغة C++، ثم تم إجراء مقارنة عددية على عدة برامج خطية. نتائج المقارنة العددية المتحصل عليها تبين أن خوارزمتنا أسرع من MADH وخوارزمية السمبلكس لـ GLPK. من أجل إيجاد حل أولي لبدأ الخوارزمية المقترحة، تم تطوير تقنية جديدة.

الكلمات المفتاحية: البرمجة الخطية، الإتجاه الهجين، تقدير شبه الأمثلة، قاعدة الخطوات المتعددة، المقارنة العددية، GLPK.

Résumé

Dans ce travail, nous avons proposé un nouvel algorithme pour résoudre les problèmes de programmation linéaire à variables bornées. Cet algorithme passe d'une solution réalisable de support à une autre meilleure en suivant une nouvelle direction hybride. La direction construite donne une meilleure amélioration locale de la fonction objectif que celle de l'algorithme MADH proposé dans [Bibi M O, Bentobache M. A hybrid direction algorithm for solving linear programs. International Journal of Computer Mathematics 2015; 92(2):200–216]. Afin d'arrêter l'algorithme, un critère de suboptimalité est utilisé et la règle du pas multiple est appliquée pour changer le support courant. L'algorithme proposé est implémenté avec C++, puis une étude numérique est menée sur des problèmes-test générés aléatoirement et sur certaines instances d'un problème de contrôle optimal. Les résultats numériques obtenus montrent que notre algorithme est compétitif avec MADH et l'algorithme primal du simplexe de GLPK. Afin d'initialiser l'algorithme proposé, une nouvelle technique d'initialisation a été développée.

Mots clés : Programmation linéaire, direction hybride, estimation de suboptimalité, règle du pas multiple, expériences numériques, GLPK.

Abstract

In this work, we propose a new algorithm for solving linear programs with bounded variables. This algorithm moves from a support feasible solution to a better one, following a new hybrid direction. The constructed direction gives a better local improvement of the objective function than the direction of the AMHD algorithm proposed in [Bibi M O, Bentobache M. A hybrid direction algorithm for solving linear programs. International Journal of Computer Mathematics 2015; 92(2):200–216]. In order to stop the algorithm, a suboptimality criterion is used and the long step rule is applied for changing the current support. The proposed algorithm is implemented with C++, then a numerical study is conducted on randomly generated test problems and some instances of an optimal control problem. The obtained numerical results show that our algorithm is competitive with AMHD and the primal simplex algorithm of GLPK. In order to initialize our algorithm, a new initialization technique was developed.

Keywords : Linear programming, hybrid direction, suboptimality estimate, long step rule, numerical experiments, GLPK.