

Table des matières

1	préliminaires	4
1.1	Éléments d'analyse convexe	4
1.2	Programme Mathématique	5
1.3	Fonction Lagrangienne	5
1.4	Condition d'optimalité	5
1.5	Méthode de Newton pour un problème avec contraintes d'égalité	6
1.6	Type de complexité ordonné	8
2	Rappel sur les méthodes (simplexe et karmarkar) en programmation linéaire	9
2.1	Notions du programmation linéaire	9
2.1.1	Forme de Programmation linéaire	10
2.1.2	Transformations sous forme standard	10
2.1.3	Dualité en programmation linéaire	11
2.1.4	Définition du dual dans le cas général	11
2.2	Méthode du simplexe	12
2.2.1	Tableau du simplexe	13
2.2.2	Algorithme du Simplexe	15
2.3	Méthode de Karmarkar	16
2.3.1	Algorithme de Karmarkar	16
2.3.2	Transformation projective	17
3	Méthodes de points intérieurs	21
3.1	les méthodes du potentiel	21
3.1.1	Principe général de la méthode	21
3.1.2	Méthode du potentiel primale	25

3.1.3	Algorithme potentiel primal	25
3.1.4	Complexité de l'algorithme	27
3.2	Les méthodes Affines	27
3.2.1	Méthode affine primale	27
3.2.2	Algorithme affine primal	29
3.3	Méthodes duale de chemin centrale (barrière logarithmique)	30
3.3.1	Barrière logarithmique	30
3.3.2	Chemin central et centre analytique	33
3.3.3	Algorithme de barrière logarithmique	33
3.3.4	Cas linéaire	34
3.3.5	Deux autres fonctions barrières (Gonzaga, 1992)	38
3.3.6	Développement par l'approche primale	39
3.4	conclusion	39

Introduction Générale

On considère que la programmation linéaire est parmi les modèles qui connaissent un développement important depuis sa naissance (1947) de la part du mathématicien Russe Kontorovitch. Après cette période, le mathématicien Dantzig George a découvert une nouvelle méthode de résolution de ces types de modèles qui a été nommée " la méthode de simplexe". Ces modèles ont été utilisés dans plusieurs domaines dans la gestion comme des outils importants dans la prise de décision. Par exemple on peut utiliser ces modèles dans la désignation de la distribution idéale des sources limitées.

La méthode Simplexe est restée sans concurrence jusqu'à ce que Karmarkar introduise un algorithme différent de son algorithme en 1984. Dans la méthode simplexe, nous évoluons dans le domaine réalisable, tandis que dans la méthode Karmarkar, nous restons dans l'intérieur du domaine, où la recherche intensive des méthodes de points internes a été lancée. Il y a aussi beaucoup d'algorithmes.

Ce mémoire est composé de trois chapitres.

Dans le premier, décrivez brièvement quelques concepts de base éléments d'analyse convexe, fonction lagrangienne et méthode de Newton...

Le deuxième chapitre, nous avons défini la programmation linéaire en faisant également référence aux premières méthodes qui ont émergé, à savoir la méthode simplexe, puis nous avons étudié la méthode de Karmarkar, qui est la passerelle vers l'émergence de les méthodes des points intérieurs.

Dans la dernière chapitre, la méthode Karmarkar a été améliorée, ce qui a conduit à l'apparition de nombreuses manières des méthodes de points intérieurs. Nous avons choisi trois méthodes, à savoir :

1. les méthodes du potentiel
2. Les méthodes Affines
3. Méthodes duale de chemin centrale (barrière logarithmique)

Chapitre 1

préliminaires

1.1 Éléments d'analyse convexe

Définition 1.1 (Ensemble convexe) Soit S un sous ensemble de \mathbb{R}^n . S est dit convexe si et seulement si :

$$\forall x_1, x_2 \in S, \forall \lambda \in [0, 1], \lambda x_1 + (1 - \lambda) x_2 \in S.$$

Définition 1.2 (Enveloppe convexe) Soit $S \subset \mathbb{R}^n$. on appelle enveloppe convexe de S et on le note $H(S)$, l'ensemble de toutes les combinaisons convexe de S , en d'autres termes : $x \in H(S) \iff \exists x_1, x_2, \dots, x_k \in S$ et $\lambda_1, \lambda_2, \dots, \lambda_k \in \mathbb{R}^+$ tel que $\sum_{i=1}^k \lambda_i = 1$ et $x = \sum_{i=1}^k \lambda_i x_i$.

Définition 1.3 L'enveloppe convexe d'un nombre fini de points x_1, x_2, \dots, x_{k+1} dans \mathbb{R}^n s'appelle polytope. Si $x_1, x_2, \dots, x_k, x_{k+1}$ sont tels que : $x_2 - x_1, x_3 - x_1, \dots, x_k - x_1, x_{k+1} - x_1$ sont linéairement indépendants alors $H(x_1, x_2, \dots, x_k, x_{k+1})$ s'appelle simplexe avec les points $x_1, x_2, \dots, x_k, x_{k+1}$.

Définition 1.4 S est dit affine si :

$$\forall x_1, x_2 \in S, \lambda x_1 + (1 - \lambda) x_2 \in S, \forall \lambda \in \mathbb{R}.$$

Définition 1.5 (Fonction convexe) Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est dite convexe sur S si :

$$f[\lambda x_1 + (1 - \lambda) x_2] \leq \lambda f(x_1) + (1 - \lambda) f(x_2), \forall \lambda \in [0, 1].$$

Définition 1.6 La fonction f est dite affine si :

$$f[\lambda x_1 + (1 - \lambda) x_2] = \lambda f(x_1) + (1 - \lambda) f(x_2), \forall \lambda \in \mathbb{R}.$$

1.2 Programme Mathématique

Un programme mathématique est un problème de la forme :

$$(PM) \begin{cases} \min f(x) \\ g_i(x) \leq 0 \quad , \quad i = 1, \dots, k \\ h_j(x) = 0 \quad , \quad j = 1, \dots, m, \end{cases}$$

où $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est appelée fonction objectif de (PM) .

$D = \{x \in \mathbb{R}^n ; h_j(x) \leq 0, g_i(x) = 0, j = 1 : m, i = 1 : k\} \subset \mathbb{R}^n$ représente l'ensemble des solutions réalisables.

- Un point x de D est appelé solution réalisable (ou admissible) de (PM) .
- On appelle solution optimale de (PM) , toute solution réalisable x réalisant le minimum de f sur D , $f(x)$ sera appelée valeur optimale de (PM)

1.3 Fonction Lagrangienne

Définition 1.7 Le lagrangien du programme mathématique (PM) est défini par :

$$L(x; \lambda, \mu) = f(x) + \sum_{j=1}^m \lambda_j h_j(x) + \sum_{i=1}^k \mu_i g_i(x).$$

Les conditions d'optimalité requièrent des hypothèses sur la régularité de la solution. Essentiellement, un point x^* est régulier dans les cas où la géométrie de l'ensemble réalisable au voisinage de x^* est adéquatement représentée par la linéarisation des contraintes. Ils existent quelques conditions de régularité dont l'indépendance linéaire des gradients des contraintes actives, les conditions de Mangasarian-Fromovitz et les conditions de Slater (dans le cas convexe). Les démonstrations des conditions d'optimalité qui suivent, ainsi que diverses notions de régularité, se trouvent dans le livre de Bazaara, Sherali et Shetty (1993).

1.4 Condition d'optimalité

Théorème 1.1 (Karush-Kuhn-Tucher (KKT))

Si x^* est une solution optimale locale (PM) satisfaisant l'une des conditions de qualification précédentes, alors il existe des multiplicateurs $\lambda \in \mathbb{R}_+^m$ et $\mu \in \mathbb{R}^k$ tels que :

$$\begin{cases} \nabla f(x^*) + \sum_{j=1}^n \lambda_j \nabla h_j(x^*) + \sum_{i=1}^k \mu_i \nabla g_i(x^*) = 0 & \text{(condition d'optimalité)} \\ \lambda_j h_j(x^*) = 0, \quad j = 1, \dots, n & \text{(condition de complémentarité)} \\ g_i(x^*) = 0. \end{cases}$$

Si en plus (PM) est convexe, les conditions précédentes sont à la fois nécessaires et suffisantes pour que x^* soit optimum global pour (PM) alors $h_i(x) < 0 \Rightarrow \lambda_i = 0$

Les multiplicateurs de Kuhn-Tucher généralisent ceux de lagrange .Autrement dit, si toutes les contraintes sont des égalités on retrouve la condition classique de lagrange : $\nabla f(x^*) + \sum_{i=1}^n \lambda_i h_i(x^*) = 0, \lambda_i \in \mathbb{R}$
Si les contraintes ne sont pas qualifiées en x^* , les conditions de KKT ne s'appliquent pas (x^* peut être optimal sans vérifier ces condition)

1.5 Méthode de Newton pour un problème avec contraint es d'égalité

On considère le problème :

$$\begin{cases} \min f(x) \\ h(x) = 0 \\ x \in \mathbb{R}^n. \end{cases}$$

Les conditions d'optimalité de Karush-Khun-Tucker sont données par le système :

$$(S_{kt}) \begin{cases} \nabla f(x) + \lambda^T \nabla h(x) = 0 \\ h(x) = 0, \end{cases}$$

où λ est le multiplicateur de Lagrange associé à la contrainte $h(x) = 0$. Pour résoudre ce système, on peut utiliser la méthode de Newton qui consiste à partir d'un point (x^k, λ^k) , à linéariser (S_{kt}) a u voisinage de x^k et à définir (x^{k+1}, λ^{k+1}) comme la solution du système linéarisé

$$(S_{ktl}) \begin{cases} \nabla f(x^k) + \sum_{j=1}^n \lambda_j^k \nabla h_j(x^k) + \left[\nabla^2 f(x^k) + \sum_{j=1}^n \lambda_j^k \nabla^2 h_j(x^k) \right] (x^{k+1} - x^k) \\ + \sum_{j=1}^n (\lambda_j^{k+1} - \lambda_j^k) \nabla h_j(x^k) = 0 \\ h(x^k) + \nabla h(x^k) (x^{k+1} - x^k) = 0. \end{cases}$$

Le système (S_{ktl}) est équivalent à :

$$(S'_{ktl}) \begin{pmatrix} H^k & (J^k)^T \\ J^k & 0 \end{pmatrix} \begin{pmatrix} x^{k+1} - x^k \\ \lambda^{k+1} - \lambda^k \end{pmatrix} = \begin{pmatrix} -\nabla f(x^k) - (J^k)^T \lambda^k \\ -h(x^k) \end{pmatrix},$$

où

$$H^k = \nabla^2 f(x^k) + \sum_{j=1}^n \lambda_j^k \nabla^2 h_j(x^k),$$

et

$$J^k = [\nabla h_1(x^k), \nabla h_2(x^k), \dots, \nabla h_n(x^k)].$$

Après simplification, (S'_{ktl}) s'écrit :

$$(S_N) \begin{pmatrix} H^k & (J^k)^T \\ J^k & 0 \end{pmatrix} \begin{pmatrix} x^{k+1} - x^k \\ \lambda^{k+1} \end{pmatrix} = \begin{pmatrix} -\nabla f(x^k) \\ -h(x^k) \end{pmatrix}.$$

Si H^k est inversible, la solution de (S_N) est donnée par :

$$\begin{pmatrix} x^{k+1} - x^k \\ \lambda^{k+1} \end{pmatrix} = \begin{pmatrix} H^{-1} - H^{-1} J^T (JH^{-1} J^T)^{-1} JH^{-1} & H^{-1} J^T (JH^{-1} J^T)^{-1} \\ (JH^{-1} J^T)^{-1} JH^{-1} & -(JH^{-1} J^T)^{-1} \end{pmatrix} \begin{pmatrix} -\nabla f(x^k) \\ -h(x^k) \end{pmatrix}.$$

Notons qu'en posant $d^k = x^{k+1} - x^k$, (S_N) s'écrit :

$$\begin{cases} H^k d^k + (J^k)^T \lambda^{k+1} = -\nabla f(x^k) \\ J^k d^k + h(x^k) = 0, \end{cases}$$

où d^k est solution du problème quadratique :

$$(Q) \begin{cases} \min \frac{1}{2} d^T H^k d + \nabla f^T(x^k) d \\ J^k d^k + h(x^k) = 0, \end{cases}$$

et λ^{k+1} est le vecteur dual optimal de (Q) .

Application au cas linéaire

On considère le problème :

$$\begin{cases} \min c^T x - \mu \sum_{i=1}^n \log x_i \\ Ax = b \\ x > 0, \end{cases}$$

où μ est un réel strictement positif. Les conditions de *K.K.T* de ce problème s'écrivent :

$$\begin{cases} c - \mu X^{-1}e + A^T \lambda = 0 \\ Ax = b. \end{cases}$$

D'après ce qui précède la direction de Newton est déterminée par :

$$\begin{pmatrix} \mu X^{-2} & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} d_N \\ -\lambda \end{pmatrix} = \begin{pmatrix} -c + \mu X^{-1}e \\ 0 \end{pmatrix}.$$

Théorème 1.2 (Dualité faible) (PL) et (D) deux programmations linéaires alors : $c^T x \geq b^T y$, $\forall x$ réalisable de (PL), $\forall y$ réalisable de (D).

Théorème 1.3 (Dualité forte) Si $x \in \mathcal{F}_{(PL)}$ et $(y, z) \in \mathcal{F}_{(D)}$, tel que $c^T x - b^T y = 0$, alors x et (y, z) sont des solution optimales de (PL) et (D).

Définition 1.8 Une solution est dite réalisable si elle vérifie l'ensemble de contraintes.

1.6 Type de complexité ordonné

considérons ce nombre d'opération $Op(n)$ d'algorithme et $Op(n)$ est en $Op(f(n))$

1 : Constante (indépendante de la taille de la donnée)

$\log n$: Logarithmique

n : Linéaire

$n \log$: Quasi-linéaire ($n \log(n)$)

n^2 : Quadratique

n^3 : Cubique

n^p : Polynomiale, $p > 3$

$n^{\log(n)}$: Quasi-polynomiale

2^n : Exponentielle

$n!$: Factorielle

Chapitre 2

Rappel sur les méthodes (simplexe et karmarkar) en programmation linéaire

La programmation linéaire est l'un des programmes de recherche opérationnelle les plus réussis, d'une part grâce à la puissance de modélisation qu'elle offre malgré les limitations inhérentes imposées par la linéarité des fonctions impliquées, et d'autre part une profusion de théories permettant le développement des algorithmes très efficaces à résoudre. Il reste le modèle d'optimisation le plus largement utilisé, depuis la formulation et le développement de la méthode simplexe pour la résoudre vers la fin des années 1940, en raison de la robustesse des algorithmes disponibles.

La théorie se base de la programmation linéaire à évolué, de même que les énormes éléments novateurs qui ont abouti aux méthodes de programmation linéaire, à la recherche linéaire et aux points internes.

En 1984, Karmarkar propose les méthodes de points intérieurs : polynomiales et efficaces en pratique sur des problèmes creux de grande taille.

2.1 Notions de programmation linéaire

La programmation linéaire constitue la pierre angulaire de toute la recherche opérationnelle. Il faut bien sûr éviter de forcer tout modèle à être linéaire. Par contre, un très grand nombre de modèles constituent des extensions de programmes linéaires. Elle peut se définir comme une technique mathématique permettant de résoudre des problèmes de gestion et particulièrement ceux où le gestionnaire doit déterminer, face à différentes possibilités, l'utilisation optimale des ressources de l'entreprise pour atteindre un objectif spécifique comme la maximisation des bénéfices ou la minimisation des coûts.

2.1.1 Forme de Programmation linéaire

Forme canonique

$$(PLC) \begin{cases} \min c^T x \\ Ax \geq b \quad (\text{ou } \leq) \\ x \geq 0. \end{cases}$$

Forme standard

$$(PL) \begin{cases} \min c^T x \\ Ax = b \\ x \geq 0, \end{cases}$$

où $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m, c \in \mathbb{R}^n$

Forme générale

$$(PLG) \begin{cases} \min c^T x \\ Ax \leq b \\ Bx \geq b', B \in \mathbb{R}^{p \times n}, b' \in \mathbb{R}^p. \end{cases}$$

2.1.2 Transformations sous forme standard

$$\exists b_i < 0 \implies \text{multiplier } b_i \text{ et ligne } A_i \text{ par } -1 \tag{2.1}$$

$$\max c^T x \iff \min -c^T x \tag{2.2}$$

$$Ax \geq b \iff Ax - y = b \quad y \geq 0 \tag{2.3}$$

$$Ax \leq b \iff Ax + y = b \quad y \geq 0 \tag{2.4}$$

$$\begin{aligned} x_{\min} \leq x \leq x_{\max} &\implies z = x - x_{\min} \geq 0 \\ z + y &= x_{\max} - x_{\min} \quad y \geq 0. \end{aligned} \tag{2.5}$$

2.1.3 Dualité en programmation linéaire

Le dual de (P) est défini sous la forme :

$$(D) \begin{cases} \max b^T y \\ A^T y \leq c, \end{cases}$$

ou équivalent :

$$(D) \begin{cases} \max b^T y \\ A^T y + z = c \\ z \geq 0, \end{cases}$$

où $y \in \mathbb{R}^m, z \in \mathbb{R}^n$.

On notera par la suite :

$$\mathcal{F}_{(PL)} = \{x \in \mathbb{R}^n : Ax = b, x \geq 0\}$$

$$\mathcal{F}_{(PL)}^0 = \{x \in \mathbb{R}^n : Ax = b, x > 0\}$$

$$\mathcal{F}_{(D)} = \{(y, z) \in \mathbb{R}^{m+n} : A^T y + z = c, z \geq 0\}$$

$$\mathcal{F}_{(D)}^0 = \{(y, z) \in \mathbb{R}^{m+n} : A^T y + z = c, z > 0\}$$

les ensembles des solutions réalisables et strictement réalisables des deux problèmes de (P) et (D) , respectivement.

2.1.4 Définition du dual dans le cas général

Evidemment, on peut définir le dual d'un programme linéaire quelconque (pas nécessairement sous forme standard). Le tableau suivant résume les correspondances entre primal et dual et d'écrire directement le dual d'un programme linéaire quelconque.

Primal	Dual
Fonction économique (min)	Second membre
Second membre	Fonction économique (max)
A matrice des contraintes	A^T matrice des contraintes
Contrainte $i : \geq$	Variable $u_i \geq 0$
Contrainte $i : =$	Variable $u_i >< 0$
Variables $x_j \geq 0$	Contrainte $j : \leq$
Variables $x_j >< 0$	Contrainte $j : =$

Exemple 2.1 Primal

$$(P) \begin{cases} \min 2x_1 - 3x_2 \\ x_1 - x_2 \leq 1 \\ 2x_1 + 3x_2 \geq 4 \\ x_1 + x_2 = 3 \\ x_1 \geq 0, x_2 >< 0. \end{cases}$$

Dual

$$(D) \begin{cases} \max u_1 + 4u_2 + 3u_3 \\ u_1 + 2u_2 + u_3 \leq 2 \\ -u_1 + 3u_2 + u_3 = -3 \\ u_1 \leq 0, u_2 \geq 0, u_3 >< 0. \end{cases}$$

2.2 Méthode du simplexe

Au cours du siècle dernier, G.Danzig a développé le mode simple, évoluant de haut en bas sur les limites des champs vérifiables, réduisant ainsi la valeur de l'objectif à l'optimum. Un critère simple permet de reconnaître le sommet optimal. Le nombre de pics qui est limite, c'est-à-dire que l'algorithme spécifique utilise le nombre de fréquences ne dépasse pas le nombre C_n^m , en supposant que tous les pics visités ne se décomposent pas, l'algorithme peut tourner dans le cas d'une dégradation, nous trouvons les techniques appropriées pour éviter le phénomène. En général, la méthode simple a un comportement numériques très satisfaisant grâce à ses multiples applications permettant de résoudre un grand nombre de problèmes pratiques. la méthode ne sont pas considérées comme efficaces et sont évaluées comme inefficaces en raison de la complexité de calcul exponentielle de $O(2^n)$.

2.2.1 Tableau du simplexe

-Idée fondamentale du simplexe : déplacement de sommet en sommet adjacent de manière à améliorer la fonction objectif.

-Transformation des inégalités en égalités : forme standard du programme linéaire - système de m équations à n inconnues ($m < n$).

-Puisque la solution optimale est un sommet (point extrême), identification algébrique des sommets : correspondance avec les bases d'un système d'équations.

-Cette méthode passe à partir d'une solution de base admissible à une solution de base voisine qui améliore la valeur de l'objectif, elle présente 3 étapes :

1. Détermination de la variable entrante (colonne pivot) .
2. Détermination de la variable sortante (ligne pivot) .
3. Pivotage.

On commence par mettre le programme linéaire standard

$$(PL) \begin{cases} \min c^T x \\ Ax = b \\ x \geq 0, \end{cases}$$

sous la forme équivalente

$$(PL) \begin{cases} \min z \\ c^T x - z = 0 \\ Ax = b \\ x \geq 0. \end{cases}$$

Si B est une base réalisable (initiale), on peut réécrire la dernière formulation (sans objectif) comme suit

$$\begin{cases} c_B^T x_B + c_N^T x_N - z = 0 \\ Bx_B + Nx_N = b, x \geq 0. \end{cases}$$

D'ou la représentation

$$\begin{array}{cccc} x_B^T & x_N^T & z & SM \\ c_B^T & c_N^T & -1 & 0 \\ B & N & 0 & b \end{array}$$

Puis , en exprimant x_B en fonction de x_N , et en supprimant la colonne z , le tableau devient :

$$\begin{array}{ccc}
 & x_B^T & x_N^T & SM \\
 -z & 0 & c_N^T - c_B^T B^{-1}N & -c_B^T \bar{b} \\
 x_B & I & B^{-1}N & \bar{b} = B^{-1}b
 \end{array}$$

Exemple 2.2

$$\begin{cases} \min x_1 - 3x_2 \\ -x_1 + 2x_2 \leq 6 \\ x_1 + x_2 \leq 5 \\ x_1, x_2 \geq 0 \end{cases} \implies \begin{cases} \min x_1 - 3x_2 + 0x_3 + 0x_4 \\ -x_1 + 2x_2 + x_3 = 6 \\ x_1 + x_2 + x_4 = 5 \\ x_i \geq 0, i = 1 : 4 \end{cases}$$

Nous avons : $n = 4$ et $m = 2 = \text{rang}(A)$

$$A = \begin{pmatrix} -1 & 2 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}, \quad b = \begin{pmatrix} 6 \\ 5 \end{pmatrix}, \quad c = (1, -3, 0, 0)^T$$

Prenons : $B = [A_3, A_4] = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I_2$

$\bar{b} = B^{-1}b = b > 0$, d'où la solution réalisable initiale : $x^0 = (0, 0, 6, 5)^T$ et le tableau correspondant .

Itération 01 :

$$\left| \begin{array}{cccccc}
 \text{vare en base} & x_1 & x_2 & x_3 & x_4 & SM \\
 -z & 1 & -3 & 0 & 0 & 0 \\
 x_3 & -1 & 2 & 1 & 0 & 6 \\
 x_4 & 1 & 1 & 0 & 1 & 5
 \end{array} \right|$$

($k = 0$)

x_2 rentre en base

x_3 quitte la base

Itération 02 :

$$\left| \begin{array}{cccccc}
 \text{vare en base} & x_1 & x_2 & x_3 & x_4 & SM \\
 -z & -\frac{1}{2} & 0 & \frac{3}{2} & 0 & 9 \\
 x_2 & -\frac{1}{2} & 1 & \frac{1}{2} & 0 & 3 \\
 x_4 & \frac{3}{2} & 0 & -\frac{1}{2} & 1 & 2
 \end{array} \right|$$

($k = 1$)

x_1 rentre en base

x_4 quitte la base

Itération 03 :

$$\left| \begin{array}{l} \text{var en base} \\ -z \\ x_2 \\ x_1 \end{array} \begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & SM \\ 0 & 0 & \frac{4}{3} & \frac{1}{3} & \frac{29}{3} \\ 0 & 1 & \frac{1}{3} & \frac{1}{3} & \frac{11}{3} \\ 1 & 0 & -\frac{1}{3} & \frac{2}{3} & \frac{4}{3} \end{array} \right|$$

$$(c_N > 0)$$

stop , optimalité ($k = 2$)

ou trouver ainsi la solution optimale au bout de deux itérations

$$\bar{x} = \left(\begin{array}{c} \frac{4}{3} \\ \frac{11}{3} \end{array} \right), \text{ la valeur optimale est } z = c^T \bar{x} = \frac{-29}{3}$$

2.2.2 Algorithme du Simplexe

Phase 01 : (Initialisation) :

Chercher une base réalisable de départ B_0 et un sommet initial

$$x^0 = (B_0^{-1}b, 0)^T \text{ correspondant}$$

Prendre $k = 0$ (compteur)

Phase 02 : (Itération)

(1) B étant la base réalisable à l'itération k et $x = (\bar{b} = B^{-1}b, 0)^T$ le sommet associé à B , on calcule $\bar{p} = B^{-T}c_B$ (mult ^ plicateurs du simplexe), puis

$$\bar{c}_N = c_N - N^T \bar{p} \quad (\text{ couts réduits })$$

(2) si $\bar{c} \geq 0$ on termine avec \bar{x} solution optimale (\bar{p} est alors solution optimale duale)

sinon : déterminer $\bar{c}_s < 0$ (une composante de \bar{c}_N)

(3) Calculer $y = B^{-1}A_s$, (A_s étant la colonne d'ordre s de la matrice A)

Si $y \leq 0$ on termine avec $c^T x$ non borné inférieurement sur S . et le probleme n'admet pas de solution optimale.

sinon : calculer :

$$\alpha = \min_{i=1, n} \left\{ \frac{\bar{b}_i}{y_i}, y_i > 0 \right\} = \frac{\bar{b}_r}{y_r} \geq 0, \text{ puis un nouveau sommet}$$

$$\begin{cases} x_i = \bar{b}_i - \alpha y_i, & i \in \{1 : m\} - \{r\} \\ x_r = 0 \\ x_s = \alpha \\ x_i = 0, & i \in \{m + 1 : n\} - \{s\}. \end{cases}$$

La variable x_r quitte alors la base et x_s devient une variable de base. Une nouvelle base est mise en oeuvre $B^+ = B + \{A_s\} - \{A\}$
 on pose $k := k + 1$ et on revient en (1)...

2.3 Méthode de Karmarkar

Malgré la complexité exponentielle de la méthode simplexe, il s'agissait de l'algorithme de référence pour la programmation linéaire. En 1984, cette méthode, sans aucun concurrent, a été proposée à Narendra Karmarkar et utilisait un nouvel algorithme à la limite proposée par AT & T Bell Labs aux Etatsunis à l'époque. Les réglages et les transformations nécessaires pour résoudre le problème de la programmation linéaire standard par la méthode de Karmarkar ont été gardés secrets jusqu'en 1985, année de leur publication notamment par Tomlin, Shanno et Marsten.

Pour ce qui est de la méthode Karmarkar, nous avançons tout en restant dans la fourchette avec une précision pouvant être atteinte, et donc une indication de l'intérieur, grace à quoi une recherche intensive a été lancée. Parmi les méthodes de point internes et ont abouti à l'émergence d'un grand nombre de groupes différents de ces algorithmes.

2.3.1 Algorithme de Karmarkar

Karmarkar traite le problème linéaire (PK) sous la forme réduite suivante :

$$(PK) \begin{cases} \min c^T x \\ Ax = 0 \\ x \in S_n = \{x \in R_+^n, e^T x = 1\} \end{cases}$$

Considérons le problème linéaire suivant :

$$(PL) \begin{cases} \min c^T x \\ A^T x = b \\ x \geq 0 \end{cases}$$

tell que $A \in R^{m \times n}$, $c \in R^n$ et $\text{rang}(A) = m < n$, ou $e = (1, 1, \dots, 1)^T$

Hypothèses

Le problème est sous forme homogène

$$(PH) \begin{cases} \min c^T x \\ Ax = 0 \\ e^T x = 1 \\ x \geq 0 \end{cases}$$

La valeur optimale de la fonction objectif est 0

Cette forme n'est pas restrictive puisque tout programme linéaire peut se ramener à cette forme (en classe).

On pose les trois hypothèses suivantes :

$$B_1 : \text{rang}(A) = m$$

$B_2 : \text{int}(S) \neq \emptyset$ et l'ensemble des optimal est borné

$$B_3 : \min_{x \in S} c^T x = 0.$$

La troisième hypothèse n'est que technique. Il est possible d'obtenir une convergence polynomiale sans cette hypothèse par l'utilisation d'une borne inférieure de la valeur optimale (obtenue par une solution duale)

Soit S le simplexe défini par

$$S = \{x \in \mathbb{R}^n : e^T x = 1, x \geq 0\}$$

L'algorithme de Karmarkar suit les étapes suivantes :

– Etant donné le point courant x_k , on effectue une transformation projective (TP) afin de centrer le point courant $x_k \rightarrow \frac{1}{n}e$ et conserver le problème dans le simplexe unité S .

– On se déplace, dans le nouvel espace, dans la direction opposée au gradient projeté sur le noyau de $\begin{bmatrix} AX^k \\ e^T \end{bmatrix}$ avec un pas relié au rayon de la plus grande sphère inscrite dans le simplexe S .

Idée :

-Utilisez la transformation projective pour déplacer un point intérieur vers le centre de la région réalisable .

-Sa déplacer le long de la direction de descente la plus raide prévue .

2.3.2 Transformation projective

Soit X^k la solution courante. La transformation est définie par

$$y = T(x) = \frac{X^{k-1} x}{e^T X^{k-1} x}; \quad x \neq 0,$$

$$x \longrightarrow y \quad \left(y = \left(\frac{1}{n}, \dots, \frac{1}{n} \right)^T, e^T y = 1 \right),$$

et son inverse par

$$x = \left(e^T X^{k-1} x \right) \left(X^{k-1} y \right) \implies e^T x = \left(e^T X^{k-1} x \right) e^T (X^k y),$$

pour chaque faisable

$$e^T x = 1 \implies e^T X^{k-1} x = \frac{1}{e^T X^k y},$$

la transformation T est inversible et on a

$$T^{-1}(y) = x = \frac{X^k \bar{x}}{e^T X^k \bar{x}}.$$

En utilisant la transformation

$$x = \frac{X^k y}{e^T X^k y},$$

$$\left\{ \begin{array}{l} \min c^T x \\ Ax = 0 \\ e^T x = 1 \\ x \geq 0 \end{array} \right. \iff \left\{ \begin{array}{l} \min \frac{cX^k y}{e^T X^k y} \\ AX^k y = 0 \\ e^T y = 1 \\ y \geq 0 \end{array} \right. \iff \left\{ \begin{array}{l} \min cX^k y \\ AX^k y = 0 \\ e^T y = 1 \\ y \geq 0 \end{array} \right.$$

Equivalence basée sur l'hypothèse : objectif optimal la valeur de la fonction est 0 et $eX^k y > 0$.

Le problème transformé de (PK) par la transformation T est le problème linéaire (PKT) suivant :

$$(PKT) \left\{ \begin{array}{l} \min cX^k y \\ AX^k y = 0 \\ e^T y = 1 \\ y \geq 0, \end{array} \right.$$

qui peut s'écrire aussi sous la forme

$$(PKT) \left\{ \begin{array}{l} \min cX^k y \\ \left[\begin{array}{c} AX^K \\ e^T \end{array} \right] y = \left[\begin{array}{c} 0 \\ 1 \end{array} \right] \\ y \geq 0. \end{array} \right.$$

Le problème transformé (PKT) est aussi mis sous la forme réduite de Karmarkar.

De même, cette transformation permet de ramener à chaque itération, l'itéré x^k au centre du simplexe S_n , i.e, $T^k(x^k) = \frac{e_n}{n}$

Remarque 2.1 Les deux transformations utilisées par Karmarkar ramènent les itérés au centre du simplexe. D'où deux stratégies sont envisagées :

L'une est de ramener le problème général (PL) à la forme réduite (PKT) puis revenir à chaque itération au problème initial (PL) pour tester l'optimalité des itérés. La deuxième, on suggère de travailler dans S_n jusqu'à l'optimalité (sans revenir à la forme initiale).

La première stratégie est la plus commode du point de vue numérique, traduite par la variante de Ye-Lustig dans le cas général (cas où z^* est inconnu).

Avant d'appliquer les conditions d'optimalité, Karmarkar relaxe le problème (PKT) en remplaçant la condition $y \geq 0$ par la sphère $s\left(\frac{e_n}{n}, \alpha r\right)$ (on montre facilement que si $y \in s\left(\frac{e_n}{n}, \alpha r\right)$ alors $y \geq 0$; voir [9]) où $r = \frac{1}{\sqrt{n(n-1)}}$ et $0 < \alpha < 1$.

Le problème (PKT) devient le problème $(PKT)_r$ alors

$$(PKT)_r \begin{cases} \min cX^k y \\ \begin{bmatrix} AX^K \\ e^T \end{bmatrix} y = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \|y - \frac{e_n}{n}\|^2 \leq (\alpha r)^2. \end{cases}$$

En utilisant les conditions d'optimalité, la solution optimale du problème $(PKT)_r$ est donnée par :

$$y = \frac{e_n}{n} - \alpha r d_k,$$

ou $d_k = \frac{p_k}{\|p_k\|}$; avec p_k est la projection du vecteur $X^k c$ sur la noyau de la matrice des pour le problème initial (PK) tel que :

$$x^k = T_k^{-1}(y^k) = \frac{X^k y^k}{e^T X^k y^k}.$$

Remarque 2.2 Cet algorithme généralisé l'algorithme de Karmarkar, cet algorithme donner la solution dans le cas général la valeur optimal z inconnu, mais l'algorithme de Karmarkar utiliser pour z connu et égal 0

L'algorithme correspondant de Ye-Lustig est le suivant [9] :

Début algorithme

Initialisation

$k = 0, x^0 > 0$ strictement réalisable pour (PL)

Un paramètre de précision $\epsilon > 0$;

$\|P_0\| + (1 + \epsilon) |c^T x^0|$;

Tant que $\frac{\|P_k\|}{\|c^T x^0\|} > \epsilon$ faire :

Etape 0 : (Construire la matrice des contraintes)

$B_k = [A_k, -b]$ ou $A^k = AX^k$ et $X^k = \text{diag}(x^k)$

Etape 1 : (Calculer la projection P_k)

$$P_k = \left[I - B^{kT} \left(B^k B^{kT} \right)^{-1} B^k \right] \begin{pmatrix} X^k c \\ -c^T x^k \end{pmatrix};$$

Et poser $d^k = \frac{P_k}{\|P_k\|}$;

Etape 2 : Calculer l'itéré suivant :

$$y^k = \frac{e_{n+1}}{n+1} - \alpha r d^k, \quad r = \frac{1}{\sqrt{n(n+1)}} \text{ avec } 0 < \alpha < 1;$$

α est le pas de déplacement ;

Etape 3 : (Revenir au problème initial (PL) par T_k^{-1})

$$x^{k+1} = T_k^{-1} (y^k) = \frac{X^k y^k [n]}{y_{n+1}^k}, \quad k = k + 1;$$

Fin tant que ;

Fin algorithme.

Chapitre 3

Méthodes de points intérieurs

Les méthodes de points intérieurs forment une classe d'algorithmes, ces derniers travaillant à résoudre des problèmes d'amélioration mathématique. Lorsqu'ils sont appliqués à ces problèmes, ils sont également multi-frontières lorsqu'ils sont appliqués au chargeur carré et généralement aux problèmes d'amélioration convexes, à condition que nous ayons une barrière autocohérente qui représente le groupe accepté et calculable à une époque de nombreuses limites.

Depuis la découverte de Karmarkar, de nombreuses variantes de méthode de point intérieur ont été développées. Il est important de réaliser qu'il existe un véritable arsenal de méthodes, basées sur les memes principes fondamentaux mais dont les caractéristiques individuelles peuvent varier assez fortement.

3.1 les méthodes du potentiel

La méthode de potentiel est une autre variante de l'algorithme de Karmarkar, qui distingue cet algorithme conservant ses propriétés.

Projection : une station de complexité multi-frontière est également la possibilité de faire de grands pas en fonction potentielle sans utiliser de variable de projection variable, nous choisirons la méthode du potentiel primal.

3.1.1 Principe général de la méthode

On considère le problème (P) ainsi que son dual (D). les algorithmes de cette méthode sont simplement des algorithmes de descente de plus forte pente avec conditionnement affine appliqué à une fonction de potentiel.

Les fonctions de potentiel

Karmarkar a été le premier à introduire une fonction de potentiel en programmation linéaire. La fonction de potentiel primale qu'il a définie est :

$$f_{KP}(x, v_L) = q \log(c^T x - v_L) - \sum_{j=1}^n \log x_j,$$

où v_L est une borne inférieure de la valeur optimale v^* et $q = n + 1$ et on a sa fonction duale

$$f_{KD}(y, v_U) = q \log(v_U - b^T y) - \sum_{j=1}^n \log s_j,$$

où v_U est une borne supérieure de v^* . Ye et Todd, à leur tour, ont proposé la fonction de potentiel primale-duale suivante

$$f_{YT}(x, s) = q \log(x^T s) - \sum_{j=1}^n \log s_j - \sum_{j=1}^n \log s_j,$$

où $q = 2n$ et $q = n + \sqrt{n}$ dans . Cette fonction a été à la base de l'algorithme de Ye, elle a été utilisée par Tanabe. Ces fonctions ont pour rôle primordial de mesurer le progrès de l'algorithme (analyse et complexité), elles peuvent aussi être utilisées dans la recherche des directions. Les fonctions de potentiel ne sont pas convexes ; cependant, elles admettent un minimum unique pour n'importe quelle borne v ($v = v_L$ ou $v = z_U$) et quel que soit $q \geq n$. Ceci est une conséquence de la convexité stricte de la fonction de potentiel multiplicative comme l'ont montré Iri et Imai.

Pour définir les fonctions de potentiel primales, on a besoin des bornes inférieures de la valeur optimale ; donc, il sera intéressant de donner une procédure pour la recherche de telles bornes.

Les bornes inférieures

Todd [3] considère le problème donné par

$$(P_\xi) \begin{cases} \min c^T x \\ Ax = b \\ \xi^T x = 0 \\ x \geq 0, \end{cases}$$

avec A est une matrice ($m \times n$) de rang maximal , $0 < m < n$; c, ξ et $x \in \mathbb{R}^n$ et $b \in \mathbb{R}^m$, pour lequel il est plus facile de générer une borne inférieure que pour le problème standard Cependant tout problème standard peut se transformer facilement en (P_ξ) . Soit alors le problème (P_ξ) . On définit :

$$F_{P_\xi}^\circ = \{x \in \mathbb{R}^n, Ax = b, \xi^T x = 0, x \geq 0\};$$

$$F'_{(P_\xi)} = \{x \in \mathbb{R}^n, Ax = b, x \geq 0\}.$$

On suppose qu'on connaît un point de $F'_{(P_\xi)}$. Par changement de variable, on peut supposer que c'est le point $e = (1, \dots, 1)^T$. Sans perte de généralité, on suppose que $\xi^T e \geq 0$, sinon on peut changer le signe de ξ . Si $\xi^T e = 0$, alors $e \in F'_{p_\xi}$, et on peut continuer avec l'algorithme de potentiel primal donné dans la prochaine section.

Dans ce qui suit, on suppose que $\xi^T e > 0$.

Remarque 3.1 *Étant donné le problème suivant :*

$$(P) \begin{cases} \min \tilde{c}^T \tilde{x} \\ \tilde{A} \tilde{x} = \tilde{b} \\ \tilde{x} \geq 0 \end{cases}$$

On définit : $x^T = (\tilde{x}^T, \tilde{x}_+)$, $c^T = (\tilde{c}^T, 0)$, $\xi^T = (0, 1)$ et $A = (\tilde{A}, \tilde{b} - \tilde{A}\tilde{e})$ ou \tilde{e} est un vecteur dont les composantes sont tous égales à 1 ayant la même dimension que \tilde{x} . Alors, le problème (P_ξ) est équivalent au problème (P) , puisque la contrainte $\xi^T x = 0$ oblige la dernière variable à être nulle, $e \in F'_{p_\xi}$ et $\xi^T e = 1 > 0$.

On suppose que l'itération courante est e et la borne inférieure est v_L (v_L peut être égale à $-\infty$). On pose $\Delta_I = \xi^T e$, par h-yppothèse, on a $\Delta_I > 0$.

En écrivant $x = x' + e$, le problème (P_ξ) devient :

$$(P'_\xi) \begin{cases} \min c^T x' + c^T e \\ Ax' = 0 \\ \xi^T x' = -\Delta_I \\ x' \geq -e. \end{cases}$$

Donc $x' \in N(A)$ ou $N(A)$ désigne le noyau de la matrice A . Pour tout vecteur u , on désigne par u_p sa projection sur $N(A)$, donc $x'_p = x'$. En utilisant les propriétés de la projection on a $u^T x' = u^T x'_p = u_p^T x'$ pour tout vecteur u . Donc, en remplaçant c et ξ par c_p et ξ_p dans (P'_ξ) et en ajoutant la contrainte $(e - e_p)^T x' = 0$, on aura le problème équivalent suivant :

$$(P''_\xi) \begin{cases} \min c_p^T x' + c^T e \\ Ax' = 0 \\ \xi_p^T x' = -\Delta_I \\ (e - e_p)^T x' = 0 \\ x' \geq -e. \end{cases}$$

En utilisant la relaxation de Fraley [3], on obtient :

$$(FRP) \begin{cases} \min c_p^T x' + c^T e \\ \xi_p^T x' = -\Delta_I \\ (e - e_p)^T x' = 0 \\ x' \geq -e. \end{cases}$$

Son dual est :

$$(FRD) \begin{cases} \max -\Delta_I \lambda - e^T s + c^T e \\ \xi_p \lambda + (e - e_p) \tau + s = c_p \\ s \geq 0. \end{cases}$$

Ce problème est lié au dual (D_ξ) de (P_ξ) qui est :

$$(D_\xi) \begin{cases} \max b^T y \\ A^T y + \xi \lambda + s = c \\ s \geq 0. \end{cases}$$

En effet, puisque $b = Ae$, la fonction objectif du dual peut s'écrire : $e^T A^T y = c^T e - \Delta_I \lambda - e^T s$, comme dans (FRD) . Aussi, pour tout vecteur u , $u - u_p \in R(A^T)$. En écrivant $u - u_p = A^T y_u$, alors le problème (FRD) peut s'écrire :

$$(FRD') \begin{cases} \max -\Delta_I \lambda - e^T s + c^T e \\ A^T (y_c - \lambda y_\xi + \tau y_e) + \xi \lambda + s = c \\ s \geq 0. \end{cases}$$

Ce problème est clairement une restriction de (D_ξ) , puisque y doit être une fonction spécifiée en λ et τ . Ainsi, la valeur optimale du problème (FRD) est inférieure ou égale à celle du problème (D_ξ) . Simplifions encore (FRD') . Tout d'abord, on élimine la constante $c^T e$ et on met au négatif la fonction objectif pour avoir $\min (\Delta_I \lambda + e^T s)$. Ceci représente la marge au-dessous de la valeur courante $c^T e$. Notons que cette marge doit être négative puisque e n'est pas réalisable. Après, on élimine s de la fonction objectif en utilisant la première contrainte de (FRD') , on aura alors

$$(FRD'') \begin{cases} \min (-\xi_p^T + \Delta_I) \lambda - (e_p^T e - n) + c_p^T e \\ \xi_p \lambda + (e - e_p) \tau \leq c_p. \end{cases}$$

Ceci est un problème de programmation linéaire à deux dimensions qui peut être résolu facilement.

- Supposons que (FRD'') a une solution optimale avec une valeur δ , alors (FRD) a une valeur optimale $c^T e - \delta$. Il existe donc une solution réalisable à (D_ξ) avec une valeur de l'objectif égale à $c^T e - \delta$. Alors, $v_L^+ = \max(v_L, c^T e - \delta)$ est notre nouvelle borne inférieure ;
- Si le problème (FRD'') est non borné, alors (FRD) et (D_ξ) le sont aussi. Dans ce cas, (P_ξ) est non réalisable ;
- Si (FRD'') est non réalisable, alors (FRD) l'est aussi et donc on ne peut rien conclure sur (D_ξ) , Dans ce cas, on se contente de $v_L^+ = v_L$.

3.1.2 Méthode du potentiel primale

Gonzaga est le premier qui a décrit un algorithme potentiel primal. Son algorithme est simplement la méthode de descente de plus forte pente avec conditionnement affine appliqué à la fonction de potentiel primale de Karmarkar :

$$f_{kp}(x, v_L) = q \log(c^T x - v_L) - \sum_{j=1}^n \log x_j,$$

il a supposé que la valeur optimale v^* de (P) est nulle. Freund a proposé une extension de l'algorithme de Gonzaga. Cependant, il a utilisé une fonction de potentiel particulière pour mesurer la performance de son algorithme :

$$f(x) = q \log(c^T s) - \sum_{j=1}^n \log x_j.$$

Dans cette $j = L$ section, on va proposer un algorithme de potentiel primal et étudier sa polynomialité. Pour faciliter l'analyse, on suppose que la valeur optimale du problème (P) est nulle et on considère la fonction de potentiel suivante :

$$f(x) = q \log(c^T x) - \sum_{j=1}^n \log x_j,$$

ou $x > 0$ et $q > 0$. Cette fonction est différentiable et on a : $\forall x > 0, \nabla f(x) = \frac{q}{c^T x} c - X^{-1} e$. En particulier

$$\nabla f(e) = \frac{q}{c^T e} c - e.$$

3.1.3 Algorithme potentiel primal

Données :

- x^0 strictement \bar{A} réalisable pour (P) ;

- ε un paramètre de tolérance strictement positif ;

Pour $k = 0, 1, \dots$

- Répéter :

- Conditionnement : $\bar{A} = AX^k, \bar{c} = X^k c, y = (X^k)^{-1} x, X^k = \text{diag}(x_1^k, \dots, x_n^k)$,

$$\bar{f}(y) = q \log(\bar{c}^T y) - \sum_{j=1}^n \log y_j$$

- Direction : $h = -P_{\bar{A}} \nabla \bar{f}(y)$ où $P_{\bar{A}} = I - \bar{A}^T (\bar{A} \bar{A}^T)^{-1} \bar{A}$

- Pas : chercher approximativement $\bar{\lambda} = \arg \min \{ \bar{f}(y + \lambda h) ; \lambda > 0 ; y + \lambda h > 0 \}$

- $y^* = y + \bar{\lambda} h$

- $x^{k+1} = X^k y^*$

- $k = k + 1$

- Jusqu'à : $c^T x^k \leq \varepsilon$

Remarque 3.2 Pour $x = X^k y, f(x) = \bar{f}(y) - \sum_{j=1}^n \log X_j^k$; donc le conditionnement affine n'a pas d'effet sur la variation de la fonction de potentiel.

Notre but est de prouver que cette fonction de potentiel diminue à chaque itération par un facteur constant indépendant de l'itération. Il suffit alors de montrer qu'il existe une constante α indépendante de k telle que $\bar{f}(y^*) \leq \bar{f}(e) - \alpha$. Pour cela, on considère le problème :

$$(\bar{P}) \begin{cases} \min \bar{c}^T x \\ \bar{A}x = b \\ x \geq 0. \end{cases}$$

Lemme 3.1 [3]

On considère le problème (\bar{P}) et soit $h = -P_{\bar{A}} \nabla \bar{f}(e)$, alors

- Si $q \geq n + \sqrt{n}$, alors $\|h\| \geq 1$

- Si $q = n$, alors $h \geq \frac{1}{2\sqrt{n}}$

Lemme 3.2 [3]

On considère le problème (\bar{P}) et soit $h = -P_{\bar{A}} \nabla \bar{f}(e)$, alors :

- Si $q \geq n + \sqrt{n}$, alors $\bar{f}(y^*) - \bar{f}(e) \leq -0.1$

- Si $q = n$, alors $\bar{f}(y^*) - \bar{f}(e) \leq \frac{-0.03}{n}$

3.1.4 Complexité de l'algorithme

Théorème 3.1 [3]

On suppose que $q = O(n)$, alors l'algorithme de potentiel primal appliqué au problème (P) possède la propriété suivante :

- Si $q \geq n + \sqrt{n}$, alors l'algorithme se termine en au plus $O(nL)$ itérations ;
- Si $q = n$, alors l'algorithme se termine dans au plus $O(n^2L)$ itérations .

Preuve. (voir [3]). ■

3.2 Les méthodes Affines

Les méthodes affines ont été introduites pour la première fois par Dikin en 1967 (algorithme d'affine à petits pas) puis par Barnes, Vanderbei et al. et Kortanek et Shi. Il a ensuite été prouvé que les algorithmes convergent sous l'hypothèse de non dégénérescence. Récemment, Tsuchiya, Terlaky et Tsuchiya, Tsuchiya et Muramatsu et Tseng et Luo ont pu montrer que la méthode converge même dans le cas de la dégénérescence, Cependant, aucune borne de polynomialité n'a été prouvée jusqu'à maintenant ces méthodes ont été mises en œuvre par de nombreux chercheurs, notamment Monna.

En fait, certains des codes basés sur dans cette section, nous avons choisi la méthode affine primal, ou nous allons donner l'algorithme primal ainsi sa variante ..

3.2.1 Méthode affine primale

Principe de ces méthodes

Considérons le problème de programmation linéaire suivant

$$(P) \begin{cases} \min c^T x \\ Ax = b \\ x \geq 0, \end{cases}$$

où A est une matrice $(m \times n)$, $c, x \in \mathbb{R}^n$ et $b \in \mathbb{R}^m$

On suppose que l'on dispose d'un point x^0 strictement réalisable de (P) i.e $x^0 > 0$ et $Ax^0 = b$

Le principe des méthodes affines consiste à générer une suite $\{x^k\}$ de points strictement réalisables vérifiant :

$$c^T x^k < c^T x^{k-1} < \dots < c^T x^0$$

et convergeant vers une solution optimale de (P) .

Détermination du point x^{k+1}

L'idée de base d'une itération est la suivante : soit x^k le point obtenu à l'itération k (le point x^k est strictement réalisable). On considère la transformation affine : $x \xrightarrow{T_\alpha} u = (X^k)^{-1} x$ ou X^k est la matrice diagonale dont les composantes sont les $x_i^k, i = 1, \dots, n$.

Alors x^k se transforme en $u = e = (1, \dots, 1)^T$ et le problème (P) se transforme en (\tilde{P}) défini par :

$$(\tilde{P}) \begin{cases} \min \tilde{c}^T u \\ \tilde{A}u = b \\ u \geq 0, \end{cases}$$

où $\tilde{c} = X^k c$ et $\tilde{A} = AX^k$. Le point courant pour (\tilde{P}) étant $u = e$, on construit alors un nouveau point $\tilde{u} = e + \bar{\alpha} \tilde{d}$ ou \tilde{d} est une direction réalisable de descente pour (P) et $\bar{\alpha}$, un pas de déplacement. x^{k+1} est alors obtenu en utilisant la transformation inverse :

$$\tilde{u} \xrightarrow{(T\alpha)^{-1}} x^{k+1} = X^k \tilde{u}, \text{ qui s'écrit aussi, } x^{k+1} = X^k \tilde{u} = X^k (e + \bar{\alpha} \tilde{d}) = x^k + \bar{\alpha} X^k \tilde{d} = x^k + \bar{\alpha} d^k$$

ou $d^k = X^k \tilde{d}$

Choix de la direction

La direction de descente généralement utilisée dans le nouvel espace est $\tilde{d} = -\tilde{c}_p$, où \tilde{c}_p est la projection de \tilde{c} sur le noyau de la matrice \tilde{A} . Ainsi, on a :

$$\begin{aligned} \tilde{d} &= - \left[I - \tilde{A}^T (\tilde{A} \tilde{A}^T)^{-1} \tilde{A} \right] \tilde{c} \\ &= - \left[I - (AX^k)^T (AX^k (AX^k)^T)^{-1} AX^k \right] X^k c \\ &= - \left[I - X^k A^T (A (X^k)^2 A^T)^{-1} AX^k \right] X^k c \\ &= -X^k c + X^k A^T (A (X^k)^2 A^T)^{-1} A (X^k)^2 c. \end{aligned}$$

En posant

$$w^k = \left(A (X^k)^2 A^T \right)^{-1} A (X^k)^2 c, \quad (3.1)$$

\tilde{d} s'écrit :

$$\tilde{d} = -X^k (c - A^T w^k), \quad (3.2)$$

où w^k est la solution du système :

$$\left(A (X^k)^2 A^T \right) w = A (X^k)^2 c. \quad (3.3)$$

Choix du pas

Nous supposons que $\tilde{d} \neq 0$ (car sinon la solution courante est optimale). Le pas est $\bar{\alpha}$ choisi le plus grand possible de façon à ce que le nouveau point \tilde{u} demeure réalisable.

Ainsi,

$$\bar{\alpha} = \max \left\{ \alpha : e + \alpha \tilde{d} \geq 0 \right\} = \begin{cases} +\infty & \text{Si } \tilde{d} \geq 0 \\ \min \left\{ \frac{-1}{\tilde{d}_j}; \tilde{d}_j < 0 \right\} & \text{Sinon.} \end{cases}$$

Notons que si $\tilde{d} > 0$ alors $\bar{\alpha} = \infty$, par conséquent le problème (\tilde{P}) est non borné et par la suite le problème (P) est aussi non borné. D'autre part, afin d'assurer la stricte réalisabilité, on prend : $x^{k+1} = x^k + \gamma \bar{\alpha} \tilde{d}^k$ avec $\gamma \in (0, 1)$

Proposition 3.1 Pour tout $k \geq 0$, on a : $c^T x^k - c^T x^{k+1} > 0$

3.2.2 Algorithme affine primal

Algorithme 1

-Etape 1 :

- Poser $k = 0$ et prendre $x^k > 0$ tel que $Ax^k = b$. (Une méthode de recherche de la solution initiale sera exposée ultérieurement)

- Choisir $\gamma \in (0, 1)$

-Etape 2 :

- Poser $x^k = \text{diag}(x_1^k, \dots, x_n^k)$.

-Etape 3 :

- Calculer la solution w^k de l'équation (3)

-Etape 4 :

- Calculer \tilde{d} d'après l'équation (2)

- Poser $d^k = X^k \tilde{d}$

-Etape 5 :

- Si $d^k > 0$, stop. Le problème (P) n'est pas borné.

- Sinon : $\bar{\alpha} = \min \left\{ \frac{-1}{\tilde{d}_j}; \tilde{d}_j < 0 \right\}$ et $x^{k+1} = x^k + \gamma \bar{\alpha} d^k$

-Étape 6 :

- Si le critère d'arrêt est satisfait. stop. Sinon $k = k + 1$ et aller à l'étape 2.

Plusieurs critères d'arrêt peuvent être considérés. Nous citons les plus utilisés. Les deux premiers critères s'appliquent lorsque la valeur de l'objectif est non nulle alors que le troisième critère s'applique lorsque cette valeur est nulle.

Critère 1 : Ce critère est basé sur la variation relative de la valeur de la fonction objectif.

Soit $\varepsilon > 0$ un paramètre de tolérance, alors l'algorithme s'arrête lorsque : $\frac{c^T x^k - c^T x^{k+1}}{c^T x^k} < \varepsilon$, avec une solution optimale $x^* = x^{k+1}$.

Critère 2 : Celui-ci est basé sur l'utilisation de la marge duale. Todd et Burrell (1986) ont montré que le vecteur w^k obtenu à l'étape 3 de l'algorithme primal est une solution approximative du problème dual et que la suite $\{w^k\}$ engendrée par cet algorithme converge vers la solution optimale du dual. Ainsi, si $\varepsilon > 0$ alors l'algorithme s'arrête lorsque : $\left| \frac{b^T w^k - c^T x^{k+1}}{c^T x^k} \right| < \varepsilon$

Critère 3 : Soit $\varepsilon > 0$, l'algorithme s'arrête lorsque : $\frac{c^T x^k - c^T x^{k+1}}{\max\{1, |c^T x^k|\}} < \varepsilon$

3.3 Méthodes duale de chemin centrale (barrière logarithmique)

Dans cette section, nous allons développer la méthode duale de chemin central, cette dernière faisant partie des méthodes de réduction en série des programmes restreints non standard (SUMT). Nous utilisons ici la barrière logarithmique, nous allons également définir le concept de la chemin central car il est fondamental d'analyser la complexité de méthodes de points internes, il est possible de montre cela, la méthode de Newton a une certaine voisinage du chemin central une convergence globale et quadratique vers le point central du chemin.

3.3.1 Barrière logarithmique

Considérons le programme convexe suivant :

$$\begin{cases} \max f_0(y) \\ f_i(y) \leq 0, i = 1, \dots, n, \end{cases}$$

ou $-f_0(y)$ et $f(y)$ sont de fonction convexes au moins deux fois continuellement différentiables de $R^m \rightarrow R$.

On pose les hypothèses suivantes :

$A_1 : \text{int}(S) = S^0 \neq \emptyset$ (condition de Slater)

$A_2 : S$ est borné

Le conditions d'optimalité du premier ordre (*KKT*), nécessaires et suffisantes sous les hypothèses posées, sont l'existence d'un point y et d'un vecteur de multiplicateurs x qui satisfont

$$\nabla f_0(y) - \sum_{i=1}^n x_i \nabla f_i(y) = 0 \tag{3.4}$$

$$f_i(y) \leq 0, \quad i = 1, \dots, n \tag{3.5}$$

$$x_i f_i(y) = 0, \quad i = 1, \dots, n \tag{3.6}$$

$$x_i \geq 0, \quad i = 1, \dots, n. \tag{3.7}$$

Considérons maintenant la fonction barrière suivante

$$B(y) = - \sum_{i=1}^n \ln(-f_i(y)),$$

et le programme non contraint

$$\min_y \phi_B(y, \mu) = - \frac{f_0(y)}{\mu} + B(y)$$

$\mu > 0$ est le paramètre de pénalité.

Rappel :

Une fonction barrière $B(x)$ pour le programme mathématique

$$\begin{cases} \min f(x) \\ g_i(x) \leq 0, i = 1 \dots k, \end{cases}$$

est une fonction non négative et continue sur le domaine réalisable qui tend vers ∞ lorsque x tend vers la frontière à partir de l'intérieur du domaine. On définit souvent une fonction barrière sous la forme

$$B(x) = \sum_{i=1}^k \theta [g_i(x)],$$

tell que la fonction μ vérifie :

- $\theta(y) \geq 0$ si $y < 0$
- $\lim_{y \rightarrow 0^-} -\theta(y) = \infty$

Remarque 3.3 La barrière logarithmique utilisée ne vérifie ces conditions que dans un voisinage de 0.

On définit le gradient de la fonction ϕ_B par

$$\nabla \phi_B(y, \mu) = g(y, \mu) = -\frac{\nabla f_0(y)}{\mu} + \sum_{i=1}^n \frac{\nabla f_i(y)}{-f_i(y)},$$

et la matrice hessienne

$$\nabla^2 \phi_B(y, \mu) = H(y, \mu) = -\frac{\nabla^2 f_0(y)}{\mu} + \sum_{i=1}^n \left(\frac{\nabla^2 f_i(y)}{-f_i(y)} + \frac{\nabla f_i(y)^T \nabla f_i(y)}{(f_i(y))^2} \right).$$

Il est facile de voir que la matrice hessienne est semi-définie positive pour $y \in S^0$ sous les hypothèses A_1 et A_2 posées. Elle est définie positive sous des conditions un peu plus fortes.

Proposition 3.2 Sous les hypothèses A_1 et A_2 , la matrice hessienne $H(y, \mu)$, $y \in S^0$ et $\mu > 0$, est définie positive dans les deux cas suivants :

- Les fonctions $f_i(y)$, $i = 0, \dots, n$ sont linéaires (affines)
- La fonction $-f_0(y)$ est convexe et les fonctions $f_i(y)$, $i = 1, \dots, n$ sont quadratiques convexes.

La fonction barrière $\phi_B(y, \mu)$ est donc strictement convexe sur le domaine borné $\text{int}(S)$ et prend des valeurs infinies sur la frontière de S . Elle atteint alors son minimum, pour $\mu > 0$ fixé, en un point unique de $\text{int}(S)$ noté $y(\mu)$. Les conditions nécessaires et suffisantes d'optimalité d'un point y du problème non contraint sont données par les équations suivantes :

$$\begin{cases} \nabla \phi_B(y, \mu) = 0 \\ f_i(y) \leq 0, \quad i = 1, \dots, n. \end{cases}$$

Ces conditions sont satisfaites si et seulement si

$$\begin{cases} \nabla f_0(y) + \mu \sum_{i=1}^n \frac{\nabla f_i(y)}{f_i(y)} = 0 \\ f_i(y) \leq 0, \quad i = 1, \dots, n, \end{cases}$$

ou par un changement de variables

$$\begin{cases} \nabla f_0(y) - \sum_{i=1}^n x_i \nabla f_i(y) = 0 \\ f_i(y) \leq 0, \quad i = 1, \dots, n \\ -x_i f_i(y) = \mu, \quad i = 1, \dots, n \\ x_i \geq 0, \quad i = 1, \dots, n. \end{cases}$$

Ces conditions sont semblables à celles présentées plus haut dans cette section (3.4) – (3.7) pour le programme convexe. En fait, seule la contrainte de complémentarité est perturbée.

3.3.2 Chemin central et centre analytique

On peut maintenant donner la définition du chemin central.

Définition 3.1 On définit le chemin central comme le chemin paramètre formé de la suite de points $y(\mu)$ lorsque l'on fait varier μ de $+\infty$ à 0^+ .

Définition 3.2 Soit $S = \{y : f_i(y) \leq 0, i = 1, \dots, n\}$ un ensemble borné et tel que $\text{int}(S) \neq \emptyset$. Le centre analytique de S est le point qui est solution du programme

$$\max \sum_{i=1}^n \ln(-f_i(y)).$$

Le centre analytique est en fait le point de départ du chemin central et est solution du programme

$$\min \phi_B(y, \mu),$$

lorsque $\mu \rightarrow \infty$. Il est facile de voir que le centre analytique est le point intérieur qui maximise le produit des variables d'écart. Ce point est donc dépendant de la représentation algébrique de l'ensemble réalisable. Dédoubler une contrainte éloignera le centre analytique de la frontière définie par cette contrainte, sans que la géométrie de l'ensemble soit modifiée.

3.3.3 Algorithme de barrière logarithmique

L'algorithme de barrière logarithmique est une méthode de points intérieurs qui, à partir d'un point initial situé près du chemin central, suit approximativement ce dernier jusqu'à la solution optimale.

Algorithme dual de chemin central :

Soit $\mu > 0$ et un critère d'arrêt ε donné.

Faire (itération majeure)

- Résoudre (approximativement) $\min \phi_B(y, \mu)$ et obtenir $y(\mu)$
- Mettre à jour $\mu = (1 - \theta)\mu$ tel que $0 < \theta < 1$ et $\mu > \theta$

L'analyse de la convergence et de la complexité nécessite plusieurs détails, tels que le choix de l'estimation initiale, le calcul de la solution approchée de $y(\mu)$, la valeur de θ et ε , etc. L'algorithme que nous allons analyser pour le cas linéaire est celui présenté dans den Hertog (1994).

Pour μ fixé, le problème d'optimisation est résolu par la méthode de Newton et la notion de proximité (pour évaluer la qualité de l'approximation) est basée sur la norme induite par la matrice hessienne (définie positive) du pas de Newton.

On présente l'algorithme général de barrière logarithmique

Initialisation

ε est le paramètre de précision ;

τ est le paramètre de proximaté ;

θ est le paramètre de réduction , ($0 < \theta < 1$)

μ_0 est la valeur initiale de la pénalité ;

y^0 est un point intérieur tel que

$$\|p(y^0, \mu_0)\|_{H(y^0, \mu_0)} \leq \tau$$

début

$y := y^0; \mu := \mu_0;$

tant que $\mu > \frac{\varepsilon}{4n}$ faire

début (itération majeure)

$\mu := (1 - \theta) \mu;$

tant que $\|P\|_H \geq \tau$ faire

début (itération mineure)

$\tilde{\alpha} = \arg \min_{\alpha > 0} \{ \phi_B(y + \alpha p, \mu) : y + \alpha p \in S^0 \}$

$y := y + \tilde{\alpha} p$

fin (itération mineure)

fin (itération majeure)

fin

Selon la valeur de θ (facteur de réduction), on parlera de méthode à :

- Long pas : $0 < \mu < 1$, indépendant de n et ε ;
- Pas moyen : $\theta = \frac{v}{\sqrt[n]{n}}$, ou $v > 0$ est indépendant de n et ε
- Petit pas : $\theta = \frac{v}{\sqrt[n]{n}}$, ou $v > 0$ est suffisamment petit de façon à ce qu'un seul pas de Newton soit suffisant pour atteindre le voisinage du nouveau centre.

Nous ferons l'analyse de convergence et de complexité pour la méthode à petit pas appliquée à un programme linéaire.

3.3.4 Cas linéaire

On considère le programme linéaire

$$\begin{cases} \max_y b^T y \\ A^T y < c, \end{cases}$$

qui est le dual de

$$\begin{cases} \min_x c^T x \\ Ax = b \\ x \geq 0. \end{cases}$$

On suppose toujours que $A_{m \times n}$ est de plein rang, que $\text{int}(S) \neq \emptyset$; et que S est borné. La fonction barrière (en posant $s = c - A^T y$) est alors

$$\phi_B(y, \mu) = \frac{-b^T y}{\mu} - \sum_{i=1}^n (\ln(s_i)),$$

et le gradient

$$\nabla \phi_B(y, \mu) = -\frac{b}{\mu} - \nabla \left(\sum_{i=1}^n \ln((c - A^T Y)_i) \right) = -\frac{b}{\mu} + AS^{-1}e,$$

ou S est la matrice diagonale formée à partir du vecteur s .

La matrice hessienne est donnée sous la forme

$$\begin{aligned} H(y, \mu) &= AS^{-1}S^{-1}A^T \\ &= AS^{-2}A^T. \end{aligned}$$

Il est facile de voir que la matrice $H(y, \mu)$ est définie positive puisque $s > 0$ et A est de plein rang.

Les conditions d'optimalités du premier ordre donnent

$$\nabla \phi_B(y, \mu) = 0,$$

d'où

$$\mu AS^{-1}e = b,$$

et Posons $x = \mu S^{-1}e$. Ainsi, on doit résoudre (approximativement) à chaque itération majeure (pour $\mu > 0$ fixé)

$$\begin{cases} Ax = b \\ A^T y + s = c \\ Sx = \mu c \\ s > 0, \quad x > 0. \end{cases}$$

Afin d'approcher la solution unique du chemin central.

Notons que le gap d'optimalité d'un point sur le chemin central est simplement $x^T(\mu)s(\mu) = n\mu$. Une notion de voisinage a été proposée par Roos et Vial (1992).

Définition 3.3 On définit la distance de y à $y(\mu)$ par

$$\delta(y, \mu) = \min_{\{x: Ax=b\}} \left\| \frac{Sx}{\mu} - e \right\|.$$

La distance $\delta(y, \mu)$ est une mesure de la violation de la contrainte $Sx = \mu e$.

Proposition 3.3 [6] La solution de :

$$\begin{cases} \min_x \left\| \frac{Sx}{\mu} - e \right\| \\ Ax = b, \end{cases}$$

est unique et est égale à

$$x(y, \mu) = \mu S^{-1}e - S^{-2}A^T (A^T S^{-2}A^T)^{-1} (\mu AS^{-1}e - b).$$

On a

$$\delta(y, \mu) = 0 \iff y = y(\mu).$$

Notons par $p_y = \eta p$ le déplacement en y dans l'ensemble S^0 lors d'une itération de la boucle intérieure. Ce déplacement induit un déplacement :

$$p_s = -A^T p_y$$

dans l'espace des valeurs que peut prendre le vecteur

$$s = c - A^T y$$

on pose

$$\begin{aligned} y^+ &= y + p_y \\ s^+ &= s + p_s = c - A^T y - A^T p_y. \end{aligned}$$

Le déplacement p_y est donné par la méthode de Newton (avec pas unitaire). Ainsi, on a

$$p_y = - (H(y, \mu))^{-1} \nabla \phi_B(y, \mu) = - (AS^{-2}A^T)^{-1} \left(-\frac{b}{\mu} + AS^{-1}e \right).$$

Alors

$$p_s = A^T (AS^{-2}A^T)^{-1} \left(-\frac{b}{\mu} + AS^{-1}e \right).$$

Convergence

Nous avons tous les éléments pour démontrer la convergence polynomiale de l'algorithme à petit pas donné. Rappelons que dans le cas de la méthode à petits pas, il n'y a qu'une seule itération mineure par itération majeure.

Algorithme de barrière logarithmique à petits pas

Initialisation :

ε est le paramètre de précision ;

θ est le paramètre de réduction ($0 < \theta < 1$) ;

μ_0 est la valeur initiale de la pénalité ;

y^0 tel que $\delta(y^0, \mu^0) \leq 0$;

début

$y := y^0; \mu := \mu_0$;

ant que $\mu > \frac{\varepsilon}{4n}$ faire

début

$\mu := (1 - \theta) \mu$;

$p_y := -(H(y, \mu))^{-1} \nabla \phi_B(y, \mu)$;

$y := y + p_y$;

fin

fin

Nous devons montrer dans un premier temps que la suite des itérés est strictement réalisable (lemme 3.4) et que μ_k représente une bonne approximation du saut de dualité (lemme 3.5). Le nombre d'itérations majeures est alors facile à montrer (théorème 3.8). Ces résultats sont valides si l'itéré courant se trouve dans un voisinage du chemin central ; ce résultat est montré au lemme 3.7.

Le prochain lemme technique relie le pas de Newton à la notion de proximité.

Lemme 3.3 [6] Pour y et μ donnés, on a

$$\begin{aligned} \delta(y, \mu) &= \|p_y\|_H \\ &= \|S^{-1}A^T p_y\|_I \\ &= \|S^{-1}p_y\|_I. \end{aligned}$$

Lemme 3.4 [6] Soit $y \in S^0$ dual réalisable et tel que $\delta(y, \mu) < 1$. On a

$$y^+ \in S^0,$$

et

$$\delta(y^+, \mu) \leq \delta(y, \mu)^2.$$

Lemme 3.5 [6] Soit $\delta = \delta(y, \mu)$. Si $\delta < 1$, alors

$$|b^T y - b^T y(\mu)| \leq \frac{\delta(1 + \delta)}{1 - \delta} \mu \sqrt{n}.$$

Lemme 3.6 [6] Soit $\mu^+ = (1 - \theta)\mu$. On a

$$\delta(y, \mu^+) \leq \frac{1}{1 - \theta} (\delta(y, \mu) + \theta \sqrt{n}).$$

Lemme 3.7 [6] Soit $\mu^+ = (1 - \theta)\mu$ avec $\theta = \frac{1}{9\sqrt{n}}$. Si $\delta(y, \mu) \leq \frac{1}{2}$, alors $\delta(y^+, \mu^+) \leq \frac{1}{2}$.

Théorème 3.2 [6] Après au plus $\frac{1}{\theta} \ln\left(\frac{4n\mu_0}{\varepsilon}\right)$ itérations majeures, l'algorithme arrête avec une solution duale tel que $z^* - b^T y \leq \varepsilon$.

Preuve. (voir [6]) ■

3.3.5 Deux autres fonctions barrières (Gonzaga, 1992)

La fonction barrière centre est donnée par

$$\phi_D(y, z) = -g \left(\ln(b^T y - z) \right) - \sum_{i=1}^n (\ln(s_i)),$$

ou $g \leq n$ est une constante et z est une borne inférieure sur la valeur optimale.

Une autre fonction barrière est donnée par

$$\phi_P(y, v) = -q \left(\ln(b^T y - v) \right) - \sum_{i=1}^n (\ln(s_i)).$$

Ici, $q \leq n$ est une constante et v est une borne supérieure sur la valeur optimale. Il y a une relation entre les valeurs optimales de ces deux fonctions et celles de la barrière logarithmique duale que nous venons de voir. Le chemin central défini par la solution optimale de la fonction $\phi_D = (y, z)$ (resp. $\phi_P = (y, v)$) paramétrée par la valeur z (resp. v) est le même que celui déterminé par la barrière logarithmique duale.

3.3.6 Développement par l'approche primale

La méthode primale de chemin central s'obtient en considérant la barrière logarithmique appliquée au problème primal. Suivant le même développement que pour l'approche duale, on aurait obtenu

$$\begin{cases} \phi_B(x, \mu) = \frac{c^T x}{\mu} - \sum_{i=1}^n (\ln(x_i)) \\ \nabla \phi_B(x, \mu) = \frac{c}{\mu} - X^{-1}e \\ \nabla^2 \phi_B(x, \mu) = X^{-2}. \end{cases}$$

Les conditions nécessaires et suffisantes d'optimalité pour un minimum de $\phi_B(x, \mu)$ sous les contraintes $Ax = b$ sont

$$\begin{cases} Ax = b \\ x > 0 \\ \frac{c}{\mu} - X^{-1}e - A^T y = 0. \end{cases}$$

En posant $s = \mu X^{-1}e$, on retrouve le système déduit pour le dual .

3.4 conclusion

Les méthodes de points intérieur sont connues par leur efficacité, rapide de convergence, simplicité algorithmique. L'inconvénient principe dans ce type de méthodes est l'initialisation. La plupart des résultats théoriques que le point initial est connu.

Cette thèse est consacrée à l'étude du problème d'optimisation linéaire et à sa résolution par des méthodes de points intérieurs, notre travail est une brève de certaines de ces méthodes avec leur algorithme et quelques exemples numériques.

Bibliographie

- [1] Alain Faye, Dualité en Programmation linéaire Algorithmes primal et dual du simplexe, Option 3A, Optimisation 1.
- [2] Anane Nassima, Méthodes de points intérieurs pour la programmation linéaire basées sur les fonctions noyaux, MASSACHUSETTS INSTITUTE OF TECHNOLOGY, 30 JUIN 2012.
- [3] EMNA KALLEL, Une synthèse sur les méthodes du point intérieur, FACULTÉ DES SCIENCES UNIVERSITÉ DE SHERBROOKE, Sherbrooke, Québec, Canada, janvier 1998.
- [4] Eric Duchêne, Programmation linéaire, Laboratoire LIRIS, Université Lyon 1.
- [5] François Glineur, Etude des méthodes de point intérieur appliquées à la programmation semidéfinie.
- [6] Gilles Savard GERAD, Introduction aux méthodes de points intérieurs, Département de mathématiques et génie industriel École Polytechnique de Montréal, Février 2001.
- [7] Javier Peña, Primal-dual interior-point methods part 1, Convex Optimization 10-725/36-725.
- [8] Michel Minoux, Programmation Mathématique : théorie et algorithmes, tome 1, Dunod, (1983).
- [9] Mousaab Bouafia, Étude asymptotique des méthodes de points intérieurs pour la programmation linéaire, Université du Havre, 2016, Français.
- [10] P. Pesneau, Programmation Linéaire 2, D'après linear Programming, Robert Vanderbei - Kluwer, 1997, Université Bordeaux 1 Bât A33 - Bur 256.
- [11] Roumili Hayet, Méthode de points intérieurs non réalisables en optimisation, théorie, Algorithme et Application, Thèse Présentée à la Faculté des Sciences Département de Mathématiques, Décembre 2007.

- [12] Shirish Shevade, Numerical Optimization, Linear Programming - Interior Point Methods, Computer Science and Automation Indian Institute of Science Bangalore 560 012, India.