

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE



UNIVERSITE MOHAMED KHIDER, BISKRA



FACULTE DES SCIENCES EXACTES ET DES SCIENCES DE LA NATURE ET DE LA VIE

DEPARTEMENT DE MATHEMATIQUES

THESE

Présentée par :

LEULMI Assma

Pour obtenir le titre de Doctorat Es-Sciences en mathématiques

OPTION

MATHEMATIQUES APPLIQUEES

THEME

*ETUDE D'UNE METHODE BARRIERE LOGARITHMIQUE VIA LES
FONCTIONS MINORANTES POUR LA PROGRAMMATION SEMI-DEFINIE*

Soutenu Le : 02/07/2018

Devant le jury composé de :

Président : *Mokhtari Zohir* Professeur Université Mohamed Khider Biskra

Rapporteur : *Merikhi Bachir* M. C. (A) Université Ferhat Abbas Sétif 1

Examineur : *Bensalem Nacerddine* Professeur Université Ferhat Abbas Sétif 1

Examineur : *Khelil Nacer* M. C. (A) Université Mohamed Khider Biskra

Année 2017–2018



Dédicaces



Toutes les lettres ne sauraient trouver les mots qu'il faut...

Tous les mots ne sauraient exprimer la gratitude,

L'amour, le respect, la reconnaissance...

Aussi, c'est tout simplement que



Je dédie cette

Thèse...

À MES CHERS PARENTS

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être.

Ma mère, « Fatima » qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mon père, « Salah » qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Dieu faire en sorte que ce travail porte son fruit ; Merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.

Mon mari, « Abdelgani » et mes enfants, « Wail », « Basma » et « Ayoub » pour son soutien, sa compréhension et son sacrifice durant l'élaboration de cette thèse.

A MES CHERS ET ADORABLE SŒURS ET FRÈRES

Mes sœurs et mes frères qui n'ont cessé d'être pour moi des exemples de persévérance, de courage et de générosité. En témoignage de mon affection fraternelle, de ma profonde tendresse et reconnaissance, je vous souhaite une vie pleine de bonheur et de succès et que Dieu, le tout puissant, vous protégez et vous garde.

A les familles « Leulmi », « Abed » et « Maache ».

A tous ceux collègue et amies pendant tout ma vie.

A tous ceux qui m'aiment, A tous ceux que j'aime, Je dédie le fruit de ma thèse de DOCTORAT.

Assma Leulmi



Remerciements



*Avant tout, je voudrais remercier **Allah** qui nous permette d'arriver là où nous sommes, et de conclure avec succès ce modeste travail et pour mes parents qui m'ont offert aussi affectifs que soucieux de mon avenir.*

*Je tiens plus particulièrement à exprimer ma gratitude envers Monsieur **Merikhi Bachir** professeur à l'**université Ferhat Abbas de Sétif-1** qui a eu la lourde tâche d'être mon encadreur.*

Ce travail n'aurait jamais abouti sans son indéfectible dynamisme et inébranlable conviction. Dans l'intérêt de mes travaux, Ses constants encouragements m'ont toujours permis de dépasser les différentes périodes de découragement qui ont pu jaloner cette période de la thèse... en toute franchise je ne pense pas que ces petits mots lui accordent tous ce qu'il mérite.

*Je remercie Monsieur **Z. Mokhtari**, Professeur à l'Université de Biskra, de m'avoir fait l'honneur de faire partie de ce jury et d'en être le président.*

*Je remercie aussi Monsieur **N. Bensalem** Professeur à l'Université de Sétif-1 et Monsieur **N. Khelil** Professeur à l'Université de Biskra pour avoir accepté de participer au jury de cette thèse.*

*Mes plus vifs remerciements vont à Monsieur **Djamel Benterki** professeur à Sétif-1, pour sa disponibilité et pour toute l'aide qu'il m'a apportée.*

Je remercie également, Madame Sara Leulmi Professeur à l'Université Constantine-1, et Madame Soumaya Leulmi Professeur à l'Université 20 Aout 1955 Skikda, pour ses encouragements, disponibilités et aides.

Nous remercions mes professeurs qui doivent voir dans ce travail la fierté d'un savoir bien acquis.

Merci

Table des matières

Introduction	4
Formulation ou position du problème	6
Objectifs contributions souhaités	7
Présentation du mémoire	8
1 Analyse convexe et notion de base	9
1.1 Analyse convexe	9
1.1.1 Ensemble et application affine	9
1.1.2 Ensembles convexes	10
1.1.3 Cônes convexes	11
1.1.4 Fonctions convexes	13
1.1.5 Semi-continuité	14
1.2 Notion de base	15
1.2.1 Définitions basiques	15
1.3 Programmation mathématique	17
1.3.1 Définitions	17
1.3.2 Classification d'un programme mathématique	19
1.3.3 Qualification des contraintes	19
1.3.4 Existence et unicité d'une solution d'un (PM)	20
1.3.5 Conditions d'optimalités	20
1.3.6 Dualité lagrangienne	21

1.3.7	Fonction barrière	21
2	Programmation semi-définie linéaire	23
2.1	Préliminaires matriciels	23
2.1.1	Matrices semi-définies positives	23
2.1.2	Le cône S_n	25
2.2	Formulation du problème	27
2.2.1	Problème primal	27
2.2.2	Problème dual	29
2.2.3	Dualité en programmation semi-définie	30
2.3	Complémentarité en SDP	33
2.4	Exemples de problèmes convertibles en SDP	34
2.4.1	Problème de programmation non linéaire	35
2.4.2	Programmation quadratique avec des contraintes quadratiques	35
2.4.3	Le problème min-max des valeurs propres	36
2.4.4	Approximation logarithmique de Tchebychev	38
2.5	Méthode de points intérieurs pour résoudre (SDP)	40
2.5.1	Méthodes affines	41
2.5.2	Méthodes de réduction du potentiel	41
2.5.3	Méthodes de trajectoire centrale	44
3	Méthode barrière logarithmique via les fonctions minorantes	47
3.1	Introduction	47
3.2	Existence et unicité de solution optimale de problème $(SDP)_\eta$ et sa convergence vers le problème (1)	50
3.2.1	Existence de solution optimale de problème $(SDP)_\eta$	50
3.2.2	Problème $(SDP)_\eta$ a une solution optimale unique	51
3.2.3	Comportement de la solution lorsque $\eta \rightarrow 0$	52
3.3	Direction de descente de Newton et recherche linéaire	54

3.4	Calcul de pas de déplacement	60
3.4.1	Première fonction minorante	60
3.4.2	Deuxième fonction minorante	62
3.4.3	Troisième fonction minorante	64
3.5	Description de l'algorithme	67
3.6	Tests numériques	68
3.6.1	Exemples à taille fixe	69
3.6.2	Exemples à taille variable	71
3.6.3	Commentaires	73
	Conclusion	74
	Bibliographie	75

Introduction générale

La programmation semi-définie (SDP) est l'un des problèmes d'optimisation qui a connu un fantastique regain d'intérêt depuis les années 90, entre autres parce que l'on a disposé depuis, d'algorithmes efficaces permettant de les résoudre : il s'agit des algorithmes de méthodes de points intérieurs.

L'évolution rapide et le succès des méthodes de points intérieurs depuis leur relance par Karmarkar (1984) [19] dans le domaine de la programmation linéaire, ont incité les chercheurs du monde entier à développer tout un arsenal de méthodes permettant de traiter convenablement plusieurs classes de problèmes considérées jadis difficiles à résoudre, parmi lesquelles la programmation semi-définie. Grâce à ces méthodes, la programmation semi-définie a connue une évolution considérable sur tous les aspects : théorique, algorithmique et numérique.

Un tel succès constitue un bon stimulant pour d'autres développements. On parle déjà de programmation semi-définie quadratique, non linéaire et programmation semi-définie linéaire.

On désigne par méthode de points intérieurs, toute procédure de résolution générant une suite de points appartenant à l'intérieur (relatif) du domaine réalisable (admissible) et convergeant vers une solution optimale du programme considéré. Il y a principalement trois grandes catégories des méthodes de points intérieurs : les méthodes affines, les méthodes de réduction du potentiel et les méthodes de trajectoire centrale. On résume la chronologie de l'évolution de l'étude des problèmes de programmation semi-définie comme suit :

- K. Bellman et S. Fan (1963) : première formulation de SDP, sous la nomination LMI (Linear Matrix Inequality),
- N. Karmarkar (1984) [19] : fameux algorithme de points intérieurs,
- Nesterov, Nemirovski, Alizadeh (1990) : ont pu étendre les méthodes de points intérieurs à la résolution de problèmes SDP.

Dés lors, plusieurs recherches ont été réalisées. On cite par exemple :

- F. Alizadeh (1995) [2], est proposé une méthode projective primale-duale.
- Vanderbeghe et al. (1996) [43], ont proposé un algorithme primal-dual de la méthode de réduction du potentiel.
- M. Kojima et al. [23], Monteiro [31] et Y. Zhang [47], ont présenté des algorithmes primaux-duaux de points intérieurs, qui sont généralisés à partir des algorithmes similaires conçus pour la programmation linéaire (ou pour le problème de complémentarité linéaire (PCL)).
- R. D. C. Monteiro (1998) [32], est proposé une méthode de type trajectoire centrale pour résoudre le problème de complémentarité linéaire.
- M. Halicka et al. (2002) [16], ont étudié la convergence de la méthode de trajectoire centrale.
- De même en (2002), J. Peng et al. [37], ont donné des fonctions auto-régulières et nouvelles directions de recherche pour l'optimisation linéaire et semi-définie.
- D. Benterki et al. (2003) [8], ont fait une étude aménagée théorique et numérique de l'algorithme proposé par Alizadeh [3].
- Koulaei et al (2007) [24], qui se sont intéressés à l'extension de l'algorithme de type Mehrotra pour la programmation semi-définie.
- De même en (2007), B. Merikhi et al. [7], ont proposé une méthode de points intérieurs réalisable pour résoudre le problème de programmation semi-définie.
- B. Merikhi et al. (2008) [11], ont proposé une méthode barrière logarithmique.
- C. Liu et al. (2012) [26], ont présenté une nouvelle analyse de la complexité d'un algorithme prédicteur-correcteur de type Mehrotra, et un nouveau algorithme de

- type correcteur de second ordre [27].
- S. Kettab et al. (2015) [20], ont proposé une relaxation de la méthode barrière logarithmique.
- A. Dehghani et al. (2017) [13], ont proposé une méthode primal-dual régularisé de point intérieur.
- I. Touil et al. (2017) [42] ont proposé une méthode de type trajectoire centrale primale-duale.

Leurs directions de recherche sont obtenues à partir d’une équation modifiée (en utilisant un facteur de paramétrisation) de Newton pour approcher une solution réalisable sur le chemin central. La convergence polynômiale de ces algorithmes a été étudiée et prouvée par plusieurs chercheurs [16, 17, 21, 25, 32, 33, 36, 40].

Formulation ou position du problème

Le sujet traité dans cette thèse concerne le problème de programmation dit d’optimisation sous contraintes semi-définie, encore appelé problème SDP. Cette appellation est une conséquence de la terminologie anglaise Semi Definite Programming.

L’objectif est de minimiser le produit scalaire de deux matrices symétriques, à savoir une matrice constante C et une matrice variable X semi-définie positive ($X \succeq 0$), soumis à un ensemble de contraintes

$$(SDP) \left\{ \begin{array}{l} \min [\langle C, X \rangle = \text{trace}(CX)] \\ \text{trace}(A_i X) = b_i, \quad i = 1, \dots, m \\ X \succeq 0 \end{array} \right.$$

où trace , désigne la trace d’une matrice carrée, $A_i, b_i, i = 1, \dots, m$ sont respectivement des matrices symétriques réelles et des scalaires réels.

Ce problème est une extension naturelle de la programmation linéaire dans le sens où les variables sont des matrices symétriques semi-définies positives au lieu d’être des

vecteurs. Ce qui explique d'ailleurs le transport du savoir faire de la programmation linéaire à la programmation semi-définie. L'étude de ce problème a commencé dès le début des années 60 en tant que problème mathématique non linéaire, il existe alors très peu de résultats publiés et ils sont principalement d'ordre théorique.

Objectifs & contributions souhaités

Actuellement, le problème (SDP) constitue l'un des sujets de recherche le plus convoité dans le domaine de l'optimisation numérique. Le but étant de développer une méthodologie adéquate. A ce propos, il est impératif de traiter des questions ouvertes, déjà étudiées au niveau de la programmation linéaire : Initialisation, pas de déplacement, coût excessif de l'itération, qui sont ici plus complexes en raison de la structure complexe du problème.

Notre travail est basé sur une méthode barrière logarithmique des méthodes de points intérieurs.

Nous proposons un algorithme de point intérieur basé sur l'approche de Newton qui nous permet de résoudre le système non linéaire résultant des conditions d'optimalité. L'itération de cet algorithme est de type descente, définie par $x^{k+1} = x^k + \alpha_k d^k$ où d^k est la direction de descente et α_k est le pas de déplacement. Le principal obstacle pour construire une itération est : La détermination et le calcul du pas de déplacement. Plusieurs alternatives sont proposées pour résoudre ce problème parmi lesquelles les méthodes de recherche linéaire [39], malheureusement le calcul du pas de déplacement par ces méthodes est coûteux et encore plus délicat dans le cas des problèmes de programmation semi-définie linéaire.

L'objectif de notre travail est de remédier à cette difficulté. Pour cela, nous proposons des procédures originelles efficaces et moins coûteuses dans la programmation semi-définie linéaire basées sur les fonctions minorantes, non seulement pour éviter les méthodes de recherche linéaire, mais aussi pour accélérer la convergence de l'algorithme, et ceci en s'inspirant des travaux de J.P. Crouzeix et B. Merikhi en programmation semi-définie

linéaire [11]. Ce point sera abordé, en détail, au chapitre 3.

Présentation de la thèse

Une recherche bibliographique a été, brièvement, survolée tout en présentant la formulation du problème, les objectifs ainsi que le plan sommaire de la thèse sont, eux-aussi, évoqués dans cette partie, qualifiée sous l'intitulé : Introduction générale.

Le premier chapitre caractérise un bref rappel relatif aux l'analyse convexe ainsi que la programmation mathématique qui servira d'appui pédagogique dans tout le contenu de la thèse.

Le deuxième chapitre, est consacré à la programmation semi-définie. On donne alors la formulation générale de ce type de problème et sa dualité. On précise ainsi quelques domaines d'application et on termine par les méthodes de résolution d'un problème (SDP).

Le dernier chapitre est le vif de notre travail. Sur le plan théorique, nous montrons que le problème perturbé $(SDP)_\eta$ de notre problème (SDP) converge vers la solution optimale du problème originel.

Dans une deuxième partie, on présente d'une manière détaillée une méthode barrière logarithmique pour résoudre le problème $(SDP)_\eta$ dans laquelle, on construit trois nouvelles fonctions minorantes donnant lieu au calcul effectif du pas de déplacement. Cette nouvelle technique a contribué d'une manière efficace à la réduction de la complexité de notre algorithme. L'ensemble de ses résultats constituent l'originalité de notre travail. Ce chapitre est étayé par des expérimentations numériques importantes d'ordre comparatif sur quelques exemples connus dans la littérature.

Notre étude porte des conclusions et ouvre des perspectives intéressantes pour l'extension de notre approche à d'autres classes de problèmes d'optimisation.

Chapitre 1

Analyse convexe et notion de base

Dans ce chapitre nous allons introduire certaines notions et résultats bien connus sur les matrices symétriques et les matrices semi-définies positives, ainsi que des notions de base de l'analyse convexe.

1.1 Analyse convexe

1.1.1 Ensemble et application affine

Définition 1.1.1 *Un sous-ensemble de \mathbb{R}^n est dit affine si :*

$$\forall x, y \in F, \forall \lambda \in \mathbb{R}, (1 - \lambda)x + \lambda y \in F$$

autrement dit, un sous-ensemble affine F contient toujours la “droite” passant par deux ses points x et y .

Les ensembles affines élémentaires sont : \emptyset , $\{x\}$ ($x \in \mathbb{R}^n$), et chaque sous-espace vectoriel de \mathbb{R}^n .

Définition 1.1.2 *On appelle hyperplan de \mathbb{R}^n toute partie affine de dimension $(n - 1)$.*

Définition 1.1.3 Une application

$$\begin{aligned} T : \mathbb{R}^n &\longrightarrow \mathbb{R} \\ x &\longmapsto Tx \end{aligned}$$

est dite affine si :

$$T[(1 - \lambda)x + \lambda y] = (1 - \lambda)Tx + \lambda Ty, \quad \forall x, y \in \mathbb{R}^n, \quad \forall \lambda \in \mathbb{R}$$

1.1.2 Ensembles convexes

Définition 1.1.4 Un sous-ensemble C de \mathbb{R}^n est dit convexe si :

$$\forall x, y \in C, \quad \forall \lambda \in [0, 1], \quad (1 - \lambda)x + \lambda y \in C$$

autrement dit, un sous-ensemble convexe contient toujours le “segment” joignant deux de ses points x et y . Sachant que :

$$[x, y] = \{(1 - \lambda)x + \lambda y : 0 \leq \lambda \leq 1\}$$

Remarque 1.1.1 Tout ensemble affine est convexe. La réciproque, est fausse en général.

Les opérations algébriques suivantes conservent la convexité :

- L’intersection quelconque.
- Le produit cartésien.
- Les transformations affines.
- Les combinaisons linéaires $\sum_{i=1}^m \alpha_i C_i$, ($\alpha_i \in \mathbb{R}$) et $m \in \mathbb{N}$.
- La translation $C + a$ avec $a \in \mathbb{R}^n$.

Définition 1.1.5 On appelle combinaison convexe de m -vecteurs de \mathbb{R}^n , x_1, \dots, x_m , toute

combinaison linéaire

$$\sum_{i=1}^m \lambda_i x_i, \text{ où } \lambda_i \geq 0 \text{ et } \sum_{i=1}^m \lambda_i = 1$$

Définition 1.1.6 L'enveloppe convexe de l'ensemble $S \subset \mathbb{R}^n$ est le plus petit convexe de \mathbb{R}^n contenant S .

$$\text{conv}(S) = \bigcap_i (C_i)$$

avec C_i convexe contenant S , et on a

$$(S \text{ convexe}) \Leftrightarrow (S = \text{conv}(S))$$

Définition 1.1.7 Soit C un convexe non vide de \mathbb{R}^n et $x \in C$. Le point x est dit extrémal (ou sommet de C) s'il n'est pas à l'intérieur d'un segment de droite contenu dans C . Autrement dit si :

$$\forall x_1, x_2 \in C, \forall \lambda \in]0, 1[: (x = \lambda x_1 + (1 - \lambda)x_2) \Rightarrow (x = x_1 = x_2)$$

Remarque 1.1.2 La définition d'un point extrémal x d'un convexe C , est équivalente à chacune des deux propriétés suivantes :

1. $(x = \frac{1}{2}x_1 + \frac{1}{2}x_2) \Rightarrow (x = x_1 = x_2, \forall x_1, x_2 \in C)$.
2. $C - \{x\} = \{y \in C : y \neq x\}$ est convexe.

Remarque 1.1.3 Tout point extrémal d'un convexe C est un point de la frontière de C .

1.1.3 Cônes convexes

Définition 1.1.8 Un sous-ensemble K est un cône si et seulement si :

$$\forall x \in K, \forall \lambda \geq 0, \lambda x \in K$$

Un cône est donc une union de demi-droites fermées positives issues de l'origine, ce

dernier peut (ou ne peut pas) appartenir à K .

Un cône est dit pointé ou saillant si et seulement si :

$$K \cap (-K) = \{0\}$$

Définition 1.1.9 1- Tout ensemble de la forme

$$K = \{x \in \mathbb{R}^n : b_i^T x \leq 0, i \in I \subseteq \mathbb{N}, b_i \in \mathbb{R}^n\}$$

est un cône convexe.

2- De même l'ensemble des solutions du système d'inégalités homogènes

$$x \in \mathbb{R}^n : Ax \leq 0$$

A est une $m \times n$ -matrice.

Cône de récession

Soit C un convexe non vide de \mathbb{R}^n , et $a \in C$, on pose :

$$C_\infty(a) = \{d \in \mathbb{R}^n : a + \lambda d \in C, \forall \lambda > 0\}$$

Alors, $C_\infty(a)$ est un cône convexe non vide.

Définition 1.1.10 On appelle cône de récession (ou asymptote) de C l'ensemble

$$C_\infty = \bigcap_{a \in C} C_\infty(a)$$

Un élément $d \in C_\infty$ est appelé direction de récession.

Proposition 1.1.1 Soit C un convexe fermé non vide de \mathbb{R}^n , alors

$$(C \text{ est borné}) \iff (C_\infty = \{0\})$$

1.1.4 Fonctions convexes

Etant donné f est une fonction définie sur un sous-ensemble, $\emptyset \neq C \subset \mathbb{R}^n$, on la remplace par l'extension suivante :

$$\tilde{f} = \begin{cases} f(x) & \text{si } x \in C \\ +\infty & \text{sinon} \end{cases}$$

Soit alors

$$\tilde{f} : \mathbb{R}^n \rightarrow]-\infty, +\infty] = \bar{\mathbb{R}} = \mathbb{R} \cup \{+\infty\}$$

Remarque 1.1.4 $f : C \subset \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ est convexe sur C si et seulement si son extension \tilde{f} (définie précédemment) est convexe sur \mathbb{R}^n .

Proposition 1 $f : \mathbb{R}^n \rightarrow]-\infty, +\infty]$ est convexe si l'une des deux propriétés équivalentes suivantes est vérifiée

1. $f[(1-\lambda)x + \lambda y] \leq (1-\lambda)f(x) + \lambda f(y), \forall \lambda \in [0, 1], \forall x, y \in \mathbb{R}^n.$

2. $f\left(\sum_{i=1}^m \lambda_i x_i\right) \leq \sum_{i=1}^m \lambda_i f(x_i), \forall m \in \mathbb{N}, \forall \lambda_i \geq 0 : \sum_{i=1}^m \lambda_i = 1, \forall x_i \in \mathbb{R}^n.$

- Si l'inégalité (1) est stricte pour $x \neq y$ et $\lambda \in]0, 1[$ f est dite **strictement convexe**.

- L'inégalité (2) est appelée **inégalité de Jensen**.

Lemme 1.1.1 Soit $f : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ alors, f est convexe sur \mathbb{R}^n si et seulement si $\forall x, d \in \mathbb{R}^n$ la fonction d'une variable réelle,

$$\varphi_{x,d}(t) = f(x + td)$$

est convexe sur \mathbb{R} .

1.1.5 Semi-continuité

Définition 1.1.11 Une fonction $f : \mathbb{R}^n \rightarrow \bar{\mathbb{R}}$ est dite **semi-continue inférieurement (s.c.i.)** en $x^0 \in \mathbb{R}^n$ si :

$$\forall \varepsilon > 0, \exists \delta > 0 : f(x) \geq f(x^0) - \varepsilon, \text{ dès que } \|x - x^0\| \leq \delta$$

Définition 1.1.12 f est dite **semi-continue supérieurement (s.c.s.)** en x^0 si $-f$ est (s.c.i.) en x^0 .

Evidemment, f est continue en x^0 si et seulement si elle est à la fois (s.c.i.) et (s.c.s.) en x^0 .

Enfin, on dit que f est (s.c.i.), si elle est (s.c.i.) en tout point $x \in \mathbb{R}^n$.

Remarque 1.1.5 1- On peut remplacer \mathbb{R}^n dans la définition précédente par un sous-ensemble quelconque $C \subset \mathbb{R}^n$.

2- La semi-continuité inférieure de f en x^0 peut être exprimée par :

$$f(x^0) \leq \liminf_{k \rightarrow \infty} f(x^k)$$

pour chaque suite $\{x^k\}$ convergeant vers x^0 et telle que :

$$\lim_{k \rightarrow \infty} f(x^k) \in \mathbb{R}$$

Autrement dit : f est (s.c.i.) en x^0 si et seulement si :

$$f(x^0) = \liminf_{x \rightarrow x^0} f(x) = \lim_{\delta \downarrow 0} \left(\inf_{\|x - x^0\| \rightarrow \delta} f(x) \right)$$

3- On peut également exprimer la (s.c.i.) en termes de voisinages comme c'est le cas pour la continuité. A ce propos, on dit que :

f est (**s.c.i.**) en x^0 si $\forall \lambda \in \mathbb{R}$ vérifiant $\lambda < f(x^0)$, $\exists \vartheta$ (voisinage de x), tel que :

$$\forall x \in \vartheta \text{ on ait } \lambda < f(x)$$

1.2 Notion de base

Notation 2 1. $M_n(\mathbb{R}) = M_n$ l'ensemble des matrices carrées de taille n à coefficients dans \mathbb{R} .

2. $S_n(\mathbb{R}) = S_n$ l'ensemble des matrices symétriques réelles de taille n . **i.e.**, Les matrices $A \in M_n$ vérifiant $A = A^T$.

1.2.1 Définitions basiques

1. L'opérateur trace de A , $\text{trace}(\cdot) : M_n \rightarrow \mathbb{R}$ donné par :

$$\text{trace}(A) = \sum_{i=1}^n a_{ii}, \quad \forall A \in M_n$$

Et pour tous $A, B, C \in M_n$ et $\alpha, \beta \in \mathbb{R}$ on a les propriétés suivantes :

$$\text{trace}(\alpha A + \beta B) = \alpha \text{trace}(A) + \beta \text{trace}(B) \quad (\text{Linéarité})$$

$$\text{trace}(A^T) = \text{trace}(A)$$

$$\text{trace}(A^2) \leq \text{trace}(A^T A) \quad (\text{Commutativité})$$

$$\text{trace}(AB) = \text{trace}(BA)$$

$$\text{trace}(AB) \leq \frac{1}{2}(\text{trace}(A^2 + B^2)) \quad \text{avec } A \text{ et } B \text{ symétriques}$$

$$\text{trace}(ABC) = \text{trace}(CAB) = \text{trace}(BCA) \neq \text{trace}(ACB)$$

$$\text{trace}(BAB^{-1}) = \text{trace}(A)$$

2. Le produit scalaire de deux matrices $A, B \in M_n$ est définie par :

$$A \bullet B = \text{trace}(A^T B) = \sum_{i=1}^n \sum_{j=1}^n a_{ij} b_{ij}$$

3. La norme, dite de Frobenius, associée à ce produit scalaire est donnée par :

$$\|A\|_F = \sqrt{\text{trace}(A^T A)}, \forall A \in M_n$$

4. L'inégalité de Cauchy-Schwarz s'écrit alors pour toutes matrices

$$A, B \in M_n : |A \bullet B| \leq \|A\|_F \|B\|_F$$

5. D'autres normes peuvent être définies sur M_n , par exemple la norme spectrale

$$\|A\|_2 = \sqrt{\lambda_{\max}(A^T A)}, \forall A \in M_n$$

Où $\lambda_{\max}(A^T A)$ désigne la plus grande valeur propre de $A^T A$. Et on a

$$\|A\|_2 \leq \|A\|_F \leq \sqrt{n} \|A\|_2$$

6. Le rayon spectral de A , $\rho(A)$ est définie par :

$$\rho(A) = \max |\lambda_i(A)| \text{ tel que } \lambda_i(A) \text{ sont les valeurs propres de } A$$

7. L'ensemble

$$S^n = \{A \in M_n \text{ tel que } A^T = A\}$$

désigne l'espace des matrices carrées symétriques d'ordre n à coefficients dans \mathbb{R} .

8. Soit $A \in S^n$, on a

$$\lambda_{\max}(A) = \max \langle v, Av \rangle \quad (\text{Rayleigh-Ritz})$$

Ici $\|v\|$ est la norme euclidienne du vecteur $v \in \mathbb{R}^n$. De plus on a

$$\begin{aligned}\|A\|_2 &= \max_{i=1,\dots,n} |\lambda_i(A)| = \rho(A) \\ \|A\|_F &= \left(\sum_{i=1}^n a_{ij}^2 \right)^{\frac{1}{2}} = \left(\sum_{i=1}^n \lambda_i^2(A) \right)^{\frac{1}{2}}\end{aligned}$$

Ici les valeurs propres de A notées $\lambda_i(A)$, sont toutes réelles (car $A \in S^n$).

9. Soit A une matrice dans S^n . Alors il existe une base de vecteurs orthonormés dans laquelle A est diagonalisable, **i.e.**, il existe une matrice P orthogonale (dite de passage), telle que $A = PDP^T$ où D est une matrice diagonale. Les colonnes u_i de P sont les vecteurs propres de A et les valeurs propres (λ_i) de A sont les coefficients diagonaux de D .

1.3 Programmation mathématique

1.3.1 Définitions

- Programme mathématique

Un programme mathématique est en général défini comme suit :

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

Où, h_j, g_i sont des fonctions de $\mathbb{R}^n \longrightarrow \mathbb{R}$ et l'ensemble

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

est appelé ensemble des contraintes (ou des solutions admissibles), dit aussi domaine de

faisabilité. La fonction $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ est appelé objectif ou économique.

- **Solution réalisable** : On appelle solution réalisable de (PM) tout point x vérifiant les contraintes **i.e.**, $x \in \mathcal{F}$.

- **Solution optimale globale** : On appelle solution optimale globale de (PM) toute solution réalisable (notée x^*) qui minimise f sur \mathcal{F} .

L'ensemble des solutions optimales globales est noté par : $\arg \min_{\mathcal{F}} f(x)$.

- **Solution optimale locale** : Un point $x^* \in \mathcal{F}$ est une solution optimale locale de (PM) si :

$$\exists \vartheta \text{ (voisinage) de } x^* \text{ tel que } f(x^*) \leq f(x), \forall x \in \vartheta$$

L'ensemble des solutions optimales locales de (PM) est noté par : $loc \min_{\mathcal{F}} f(x)$. Nous avons toujours

$$\arg \min_{\mathcal{F}} f(x) \subseteq loc \min_{\mathcal{F}} f(x)$$

- Une contrainte d'inégalité $h_j(x) \leq 0$, $\forall j$ est dite saturée (active) en $x^* \in \mathcal{F}$, si :

$$h_j(x^*) = 0$$

Une contrainte d'égalité est par définition saturée en tout point x de \mathcal{F} .

Remarque 1.3.1 *Le problème d'optimisation précédent consiste*

- Soit à chercher un point optimal.
- Soit, si un tel point n'existe pas on cherche une borne inférieure à f .
- Soit à établir que f est non bornée inférieurement sur \mathcal{F} , auquel cas on adopte la convention

$$\inf_{\mathcal{F}} f(x) = -\infty$$

Lorsque \mathcal{F} est vide, on pose par convention

$$\inf_{\mathcal{F}} f(x) = +\infty$$

1.3.2 Classification d'un programme mathématique

On classifie un (PM) à partir de deux propriétés fondamentales à savoir la convexité et la différentiabilité de la fonction objectif et les contraintes.

- (PM) est un problème différentiable si les fonctions f , g_i , h_j sont toutes différentiables.

- (PM) est un problème convexe si f et h_j sont convexes et g_i affines.

La classe modèle des (PM) est celle des programmes convexes différentiables, les programmes non convexes ou non différentiables sont difficiles à traiter. Enfin, le cas le plus simple est celui de la programmation linéaire où f , g_i et h_j sont affines.

1.3.3 Qualification des contraintes

Définition 1.3.1 On dit que la contrainte $h_i(x) \leq 0$ est active ou saturée en $\bar{x} \in \mathcal{F}$ si $h_i(\bar{x}) = 0$. On introduit alors l'ensemble

$$I(\bar{x}) = \{i : h_i(\bar{x}) = 0\}$$

Par définition, une contrainte d'égalité est saturée.

Voici trois conditions de qualification classiques

- Karlin (1959) : Si \mathcal{F} est un polyèdre convexe (**i.e.**, h_i , g_j sont affines), alors les contraintes sont qualifiées en tout point réalisable.
- Slater (1950) : Si \mathcal{F} est convexe (**i.e.**, h_i convexe et g_j affine) et $\text{int}(\mathcal{F}) \neq \emptyset$, alors les contraintes sont qualifiées en tout point réalisable.
- Mangasarian-Fromovitz (1967) : Si les gradients de toutes les fonctions des contraintes saturées en $\bar{x} \in \mathcal{F}$ sont linéairement indépendants, alors les contraintes sont qualifiées en \bar{x} .

Théorème 1.3.1 Pour un programme convexe, tout optimum local est un optimum global.

1.3.4 Existence et unicité d'une solution d'un (PM)

Théorème 1.3.2 (Weirstass) Si f est une fonction continue sur $\mathcal{F} \subset \mathbb{R}^n$ et \mathcal{F} est compact (fermé et borné), alors (PM) admet au moins une solution optimale $x^* \in \mathcal{F}$.

Corollaire 1 Si $\mathcal{F} \subset \mathbb{R}^n$ est non vide et fermé et si f est continue et coercive sur \mathcal{F} (au sens que $f(x) \rightarrow +\infty$ lorsque $x \rightarrow +\infty$), alors (PM) admet au moins une solution optimale.

Théorème 1.3.3 Si f est strictement convexe et l'ensemble \mathcal{F} est convexe, alors (PM) admet une solution optimale unique.

Remarque 1.3.2 La stricte convexité n'assure pas l'existence de la solution mais assure l'unicité.

1.3.5 Conditions d'optimalités

Reprenons le problème (PM)

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

Le lagrangien associé à (PM) est défini par :

$$\mathcal{L}(x, \lambda, \mu) = f(x) + \sum_{j=1}^p \lambda h_j(x) + \sum_{i=1}^m \mu g_i(x)$$

La théorie de **Karush-Kuhn-Tucker (K.K.T)** permet d'écrire les conditions nécessaires d'optimalité pour tout problème d'optimisation avec contraintes possèdent une fonction objectif différentiable.

Théorème 1.3.4 Si x^* est une solution optimale locale de (PM) satisfaisant l'une des conditions de qualifications précédentes, alors il existe des multiplicateurs $\lambda \in \mathbb{R}_+^p$ et $\mu \in \mathbb{R}^m$ tel que :

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

Remarque 1.3.3 1- Si (PM) est convexe, les conditions de **(K.K.T)** sont à la fois nécessaire et suffisantes pour que x^* soit un minimum global.

2- Si les contraintes ne sont pas qualifiées en x^* , les conditions de **(K.K.T)** ne s'appliquent pas (x^* peut être optimal sans vérifier ces conditions).

1.3.6 Dualité lagrangienne

Le dual de (PM) est le programme mathématique (DM) suivant :

$$(DM) \begin{cases} \sup_{\lambda, \mu} \inf_{x \in \mathcal{F}} \mathcal{L}(x, \lambda, \mu) \\ \nabla_x \mathcal{L}(x, \lambda, \mu) = 0 \\ \lambda \in \mathbb{R}_+^p, \quad \mu \in \mathbb{R}^m \end{cases}$$

La dualité joue un rôle fondamental dans l'étude et la résolution de (PM) . Entre autre, elle fournit des informations supplémentaires très utiles.

1.3.7 Fonction barrière

Définition 1.3.2 Soit $C \subset \mathbb{R}^n$. On appelle fonction barrière associée à C toute fonction B_r définie sur $\text{int}(C)$ telle que :

1. B_r est continue.
2. $B_r(x) \geq 0, \quad \forall x \in \text{int}(C)$.
3. $\lim_{x \rightarrow x^*} B_r(x) = +\infty$, quand $x \rightarrow x^* \in \text{Fr}(C)$.

Les fonctions barrières les plus utilisées sont les fonctions logarithmiques définies par :

$$B_r(x) = -\sum_{i=1}^n \ln(h_i(x))$$

Exemple 1.3.1 Pour

$$C = \{x \in \mathbb{R}^n : h_i(x) \leq 0, i = 1, \dots, m\}$$

avec $h_i \in C^1(\mathbb{R}^n)$ et

$$\text{int}(C) = \{x \in \mathbb{R}^n : h_i(x) < 0, i = 1, \dots, m\}$$

On peut prendre

$$B_{r1}(x) = -\sum_{i=1}^m \frac{1}{h_i(x)}$$

et

$$B_{r2}(x) = -\sum_{i=1}^m \log(-h_i(x))$$

Le problème pénalisé correspondant est alors

$$\left\{ \min(f(x) + \frac{1}{\mu} B_r(x)), \mu > 0 \right\}$$

avec $\mu > 0$ paramètre de pénalisation.

Chapitre 2

Programmation semi-définie linéaire

Dans ce chapitre, nous aborderons l'étude des problèmes de minimisation de fonctions linéaires à contraintes linéaires, où la variable, est une matrice carrée symétrique $X \in S_n$ semi-définie positive. Dans la suite, on va donner un préliminaire matriciel.

2.1 Préliminaires matriciels

2.1.1 Matrices semi-définies positives

Définition 2.1.1 Soit $A \in S_n$

· A est dite semi-définie positive ou $A \succeq 0$ si :

$$\forall x \in \mathbb{R}^n, x^T A x \geq 0$$

· A est dite définie positive ou $A \succ 0$ si :

$$\forall x \in \mathbb{R}^n, x \neq 0 \text{ et } x^T A x > 0$$

Toute sous-matrice principale d'une matrice symétrique semi-définie (resp. définie) positive est aussi semi-définie (resp. définie) positive.

En particulier, tous les éléments diagonaux d'une matrice symétrique semi-définie (resp. définie) positive sont positifs (resp. strictement positifs).

Notation 3 1. $S_n^+(\mathbb{R}) = S_n^+ = \{A \in S_n : A \succeq 0\}$.

2. $S_n^{++}(\mathbb{R}) = S_n^{++} = \{A \in S_n : A \succ 0\}$.

Proposition 4 Soit $B \in M_n$ inversible, alors

$$(A \in S_n^+) \Leftrightarrow (B^T A B \in S_n^+)$$

et

$$(A \in S_n^{++}) \Leftrightarrow (B^T A B \in S_n^{++})$$

Lemme 2.1.1 Soient $A, B \in S_n^+$. Alors $\langle A, B \rangle \geq 0$ et en plus $\langle A, B \rangle = 0$ si et seulement si $AB = 0$.

Théorème 2.1.1 Pour $A \in S_n$ les propriétés suivantes sont équivalentes

1. $A \in S_n^{++}$.
2. $\lambda_i > 0, \forall i = 1, \dots, n$.
3. Il existe $B \in M_n$ avec $\text{rang}(B) = n$ tel que $A = B^T B$.
4. Pour une suite arbitraire $A_i \in S_i, i = 1, \dots, n$, de sous-matrices principales de A

$$\det(A_i) > 0 \text{ pour } i = 1, \dots, n$$

On a aussi, pour une matrice semi-définie positive, l'existence d'une racine carrée.

Proposition 2.1.1 Soit $A \in S_n^+$. Alors il existe une matrice unique $B \in S_n^+$ telle que :

$$A = B^2$$

On a de plus

$$\text{rg}(A) = \text{rg}(B)$$

et on notera $B = A^{\frac{1}{2}}$.

Théorème 2.1.2 (Complément de Schur)

Soient $A \in S_m^{++}$, $C \in S_n$ et $B \in M_{m,n}$. On a alors les équivalences suivantes :

$$\begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \succeq 0 \Leftrightarrow C - B^T A^{-1} B \succeq 0$$

et

$$\begin{bmatrix} A & B \\ B^T & C \end{bmatrix} \succ 0 \Leftrightarrow C - B^T A^{-1} B \succ 0$$

2.1.2 Le cône S_n

Nous étudions quelques propriétés spécifiques à S_n et sa structure conique.

Proposition 5 1. S_n^+ est un cône fermé pointé dans S_n . (car $S_n \cap (-S_n) = \{0_n\}$, 0_n est la matrice nulle).

2. S_n^{++} n'est pas un cône (car $0_n \notin S_n^{++}$).

3. S_n^{++} est l'intérieur de S_n^+ , de plus on a

$$S_n^{++} = \text{ri}(S_n^+)$$

(à rappeler que : Soit $C \subset \mathbb{R}^n$, $C \neq \emptyset$, l'intérieur relatif de C est :

$$\text{ri}(C) = \{ \text{aff}(C) : \exists \varepsilon > 0, (x + \varepsilon B_1) \cap \text{aff}(C) \subseteq C \}$$

et tel que :

$$\text{aff}(C) = \left\{ x = \sum_{i=1}^n \lambda_i x_i, x_i \in C, \sum_{i=1}^n \lambda_i = 1 \right\}$$

et B_1 est la boule unité euclidienne de \mathbb{R}^n tel que :

$$B_1 = \{x \in \mathbb{R}^n : \|x\| \leq 1\}$$

On a aussi

$$(aff(C) = \mathbb{R}^n) \Rightarrow (ri(C) = int(C))$$

tel que :

$$int(C) = \{x \in C : \exists \varepsilon > 0, (x + \varepsilon B_1)\} \subseteq C$$

avec C un convexe de \mathbb{R}^n .

4. La structure du cône de S_n^+ induit un ordre partiel sur S_n dit "Ordre de Löwner":

$$(A \succeq B) \Leftrightarrow (A - B) \in S_n^+(\mathbb{R})$$

$$(A \succ B) \Leftrightarrow (A - B) \in S_n^{++}(\mathbb{R})$$

(\succ, \succeq : relation d'ordre partiel).

Proposition 6 Soient $A, B \in S_n^+$, alors

$$1. A + B \succeq B.$$

$$2. A^{\frac{1}{2}} B A^{\frac{1}{2}} \succeq 0.$$

(à noter que si $A \in S_n$, alors il existe une unique matrice $M \in S_n^+$, telle que $A = M^2$, de plus M commute avec A , et vérifie

$$rg(A) = rg(M)$$

et on note souvent $M = A^{\frac{1}{2}}$).

$$3. trace(AB) \leq trace(A)trace(B).$$

4. Les valeurs propres de la matrice AB sont positives.

Remarque 2.1.1 Le produit de deux matrices semi-définies positives n'est pas nécessai-

rement une matrice semi-définie positive.

2.2 Formulation du problème

2.2.1 Problème primal

Un programme semi-défini linéaire de forme standard est défini par :

$$\begin{cases} \min (C \bullet X = \text{trace}(CX)) \\ A_i \bullet X = b_i, \quad i = 1, \dots, m \\ X \in S_n^+(\mathbb{R}) \end{cases} \quad (\text{SDP})$$

où $C, A_i \in M_n$, $b = (b_1, \dots, b_m)^T \in \mathbb{R}^m$. Sans perte de généralité on peut supposer que C et les A_i sont symétrique.

Remarque 2.2.1 1. Dans le cas, où C n'est pas symétrique, on la symétrise en remplaçant C par $\frac{1}{2}(C + C^T)$ de même pour les A_i où on prend $\frac{1}{2}(A_i + A_i^T)$ à la place de A_i .

2. On peut supposer également (et sans perte de généralité) que les matrices A_i , $i = 1, \dots, m$, sont linéairement indépendantes pour des besoins théoriques.

3. La programmation semi-définie est un cas particulier de la programmation conique, et une généralité de la programmation linéaire, ici on manipule des matrices au lieu des vecteurs. Effectivement on peut ramener un problème de programmation linéaire en un problème de programmation semi-définie. Pour ce fait, soit le programme linéaire suivant :

$$\begin{cases} \min_x c^T x \\ Ax = b \\ x \geq 0, \quad x \in \mathbb{R}^n \end{cases} \quad (\text{PL})$$

En pose

$$C = \begin{pmatrix} c_1 & 0 & \dots & 0 \\ 0 & c_2 & \dots & 0 \\ 0 & 0 \dots & 0 & c_n \end{pmatrix}, X = \begin{pmatrix} x_1 & 0 & \dots & 0 \\ 0 & x_2 & \dots & 0 \\ 0 & 0 \dots & 0 & x_n \end{pmatrix}$$

$$A = \begin{pmatrix} a_{i1} & 0 & \dots & 0 \\ 0 & a_{i2} & \dots & 0 \\ 0 & 0 \dots & 0 & a_{in} \end{pmatrix} \text{ et } b = (b_i)_{i=1, \dots, n}$$

On a donc

$$\begin{cases} c^T x = \langle C, X \rangle = \text{trace}(C^T X) \\ (Ax = b) \Leftrightarrow (A_i \bullet X = b_i), \quad i = 1, \dots, m \\ (x \geq 0) \Leftrightarrow (x_i \geq 0, \forall i) \Rightarrow (X \succeq 0) \end{cases}$$

d'où

$$(PL) \Leftrightarrow \begin{cases} \min c^T x \\ Ax = b \\ x \geq 0, x \in \mathbb{R}^n \end{cases} \Rightarrow \begin{cases} \min \langle C, X \rangle \\ A_i \bullet X = b_i, \quad i = 1, \dots, m \\ (x_i \geq 0, \forall i) \Rightarrow (X \succeq 0) \end{cases} \Leftrightarrow (SDP)$$

Définition 2.2.1 Une matrice $X \in S_n$ est dite réalisable (resp. strictement réalisable) pour SDP si :

$$\langle A_i, X \rangle = b_i, \quad i = 1, \dots, m \text{ et } X \in S_n^+$$

(resp. $\langle A_i, X \rangle = b_i, \quad i = 1, \dots, m$ et $X \in S_n^{++}$). On note Y (resp. \hat{Y}) l'ensemble des solutions réalisables (resp. l'ensemble des solutions strictement réalisable) pour (SDP).

Définition 2.2.2 La valeur optimale primale de SDP est définie par :

$$p^* = \inf \{ \langle C, X \rangle, \langle A_i, X \rangle = b_i, \quad i = 1, \dots, m, X \in S_n^+ \}$$

X^* est une solution optimale primale de (SDP) si :

$$X^* \in Y \text{ et } \langle C, X^* \rangle = p^*$$

2.2.2 Problème dual

La dualité en SDP est très similaire à la dualité classique en programmation linéaire à quelques différences près. Soit le problème (SDP) linéaire primal sous la forme standard précédemment donné.

Pour obtenir le problème dual de (SDP), on considère la fonction Lagrangienne

$$q(y) = \min_{x \in S_n^+} \left[\langle C, X \rangle + \sum_{i=1}^m (b_i - \langle A_i, X \rangle) y_i, y \in \mathbb{R}^m \right]$$

d'où

$$\max_{y \in \mathbb{R}^m} q(y) = \max_{y \in \mathbb{R}^m} \min_{x \in S_n^+} \left[\left\langle \left(C - \sum_{i=1}^m y_i A_i \right), X \right\rangle + \sum_{i=1}^m b_i y_i, y \in \mathbb{R}^m \right]$$

donc

$$\max_{y \in \mathbb{R}^m} q(y) = \begin{cases} \max_{y \in \mathbb{R}^m} \sum_{i=1}^m b_i y_i & \text{si } C - \sum_{i=1}^m y_i A_i \in S_n^+ \\ -\infty & \text{ailleurs} \end{cases}$$

D'où par convention le dual du problème (SDP) est un problème SDP défini par :

$$m_d = \begin{cases} \max b^T y \\ C - \sum_{i=1}^m y_i A_i \in S_n^+ \\ y \in \mathbb{R}^m \end{cases} \Leftrightarrow \begin{cases} \max b^T y \\ C - \sum_{i=1}^m y_i A_i = S \\ y \in \mathbb{R}^m, S \in S_n^+ \end{cases} \quad (DSDP)$$

Définition 2.2.3 Une solution réalisable de (DSDP) est le couple $(y, S) \in \mathbb{R}^m \times S_n^+$, tel que :

$$C - \sum_{i=1}^m y_i A_i = S$$

De même le couple (y, S) est dit strictement réalisable pour (DSDP) si :

$$C - \sum_{i=1}^m y_i A_i \in S_n^{++}$$

On note F (resp. \widehat{F}) l'ensemble des solutions réalisables pour (DSDP) (resp. l'ensemble des solutions strictement réalisables pour (DSDP)).

Définition 2.2.4 La valeur optimale de (DSDP) est définie par :

$$d^* = m_d = \sup \left\{ b^T y : C - \sum_{i=1}^m y_i A_i \in S_n^+, y \in \mathbb{R}^m \right\}$$

avec

$$(y^*, S^*) \in F, by^* = d^*$$

et

$$S^* = C - \sum_{i=1}^m y_i^* A_i$$

2.2.3 Dualité en programmation semi-définie

Définition 2.2.5 (Saut de dualité) Soit $(X, y, S) \in F$, la quantité

$$\langle C, X \rangle - b^T y = \langle S, X \rangle = X \bullet S$$

est appelée saut de dualité de (SDP) et (DSDP) en (X, y, S) .

Dualité faible

Proposition 7 Soit $X \in Y$ et $(y, S) \in F$, alors

$$C \bullet X - b^T y = X \bullet S \geq 0$$

Remarque 2.2.2 Contrairement à la programmation linéaire, il n'est pas toujours vrai que l'optimalité de (SDP) et (DSDP) implique que $\langle S, X \rangle = 0$, comme le montrent le exemple suivant :

Exemple 2.2.1 $E = S_3$ l'espace des matrice symétriques 3×3 :

$$m_p = \min_x [C \bullet X, X \in S_3^+, \langle A_i, X_i \rangle = b_i, i = 1, \dots, m] \quad (\text{SDP})$$

où

$$C = \begin{pmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, A_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, A_2 = \begin{pmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$
$$X = \begin{pmatrix} x_{11} & x_{12} & 0 \\ x_{12} & x_{22} & 0 \\ 0 & 0 & x_{33} \end{pmatrix} \text{ et } b = (0, -1)^T$$

Ce problème s'écrit aussi sous la forme suivante :

$$m_p = \min_x \left[x_{12}, X = \begin{pmatrix} 0 & x_{12} & 0 \\ x_{12} & x_{22} & 0 \\ 0 & 0 & 1 + x_{12} \end{pmatrix} \succeq 0 \right] \quad (\text{SDP})$$

L'ensemble des solutions réalisables correspond à l'ensemble

$$Y = \{(x_{12}, x_{22}) \in \mathbb{R}^2 : x_{12} = 0, x_{22} \geq 0\}$$

En outre l'ensemble des solutions optimales est

$$X^* = \left\{ \left(\begin{array}{ccc} 0 & x_{12} & 0 \\ x_{12} & x_{22} & 0 \\ 0 & 0 & 1 + x_{12} \end{array} \right), x_{22} \geq 0 \right\}$$

Il s'ensuit que, $m_p = 0$.

Ecrivons le problème dual

$$m_d = \max_{y \in \mathbb{R}^2} \left[b^t y : C - \sum_{i=1}^2 y_i A_i \in S_3^+ \right] \quad (\text{DSDP})$$

Qui correspond à la forme

$$m_d = \max_{y \in \mathbb{R}^2} \left[-y : \begin{pmatrix} -y_1 & \frac{1-y_2}{2} & 0 \\ \frac{1-y_2}{2} & 0 & 0 \\ 0 & 0 & y_2 \end{pmatrix} \in S_3^+ \right] \quad (\text{DSDP})$$

L'ensemble des solutions réalisables correspond à l'ensemble

$$\{(y_1, y_2) \in \mathbb{R}^2 : y_1 \leq 0, y_2 = 1\}$$

En outre l'ensemble des solutions optimales est

$$Y^* = \{(y_1, 1) : y_1 \leq 0\}$$

Il s'ensuit que, $m_d = -1 \neq 0 = m_p$. Les problèmes (SDP) et (DSDP) sont tous deux réalisables, X et Y sont tous deux non vides mais, $-\infty < m_d < m_p < +\infty$.

Ainsi la réalisabilité des deux problèmes ne suffit pas pour avoir les mêmes valeurs optimales.

En conséquence, l'existence d'une solution optimale pour l'un des deux problèmes (SDP) ou son dual ($DSDP$) n'implique pas nécessairement l'optimalité de l'autre même si leurs valeurs optimales sont identiques et finis. Pour conserver le résultat de forte dualité pour (SDP), on exige la condition de stricte réalisabilité de l'un des deux problèmes comme le montre le théorème suivant.

Dualité forte

Théorème 2.2.1 [35]

1. Si le problème (SDP) primal est strictement réalisable, c-à-d :

$$\exists X \in S_n^{++}(\mathbb{R}) : \langle A_i, X \rangle = b_i, \quad i = 1, \dots, m$$

Alors, $m_d = m_p$.

2. Si en outre m_p est fini, alors l'ensemble des solutions optimales du problème dual ($DSDP$) est compact non vide.

3. Si le problème ($DSDP$) est strictement réalisable, c-à-d :

$$\exists y \in \mathbb{R}^m : C - \sum_{i=1}^m y_i A_i \in S_n^{++}$$

alors, $m_d = m_p$.

4. Si en outre m_d est fini, alors l'ensemble des solutions optimales du problème primal (SDP) est compact non vide.

2.3 Complémentarité en SDP

A l'image de ce qui se passe en programmation linéaire, on peut exprimer la condition pour X^* et y^* d'être des solutions optimales de (SDP) et ($DSDP$) sous la forme d'une

condition dite de complémentarité.

Théorème 2.3.1 *Soient $X^* \in Y$ et $(y^*, S^*) \in F$ de saut de dualité*

$$\langle C, X^* \rangle - b^t y^* = \langle S^*, X^* \rangle$$

Alors X^ et (y^*, S^*) sont des solutions optimales pour (SDP) et $(DSDP)$ respectivement si et seulement si $X^* S^* = 0$.*

Le problème de complémentarité s'écrit comme suit :

$$(PC) \begin{cases} \langle A_i, X \rangle & = b_i, \quad i = 1, \dots, m, \quad X \succeq 0 \\ C - \sum_{i=1}^m y_i A_i & = S, \quad S \succeq 0 \\ XS & = 0 \end{cases}$$

Remarque 2.3.1 *Comme (SDP) est un problème convexe, la solution du problème de complémentarité est un optimum global pour (SDP) .*

2.4 Exemples de problèmes convertibles en SDP

L'intérêt pour la programmation (SDP) s'est encore accru ces dernières années lorsque de nombreuses applications ont été identifiées dans des domaines variés tels que le contrôle, les statistiques, la finance, la localisation, l'optimisation robuste, l'ingénierie, etc. Pour ne citer que quelques exemples, les problèmes (SDP) apparaissent naturellement en contrôle (stabilité), analyse des séries temporelles (complétion de covariances), graphes (mélange de chaînes de Markov), etc. (voir [22, 44]). Quelques-unes de ces applications sont présentées dans la partie suivante :

2.4.1 Problème de programmation non linéaire

Considérons le problème non linéaire suivant :

$$\begin{cases} \min(x_1^3 + x_2) \\ x_1^3 x_2 \geq 1 \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

Ce problème s'écrit sous forme d'un programme semi-défini (*SDP*) comme suit :

$$\begin{cases} \min \text{trace}(CX) \\ \text{trace}(A_1 X) = b \\ X \in K \end{cases}$$

avec

$$C = I, X = \begin{pmatrix} x_1^3 & x_3 \\ x_3 & x_2 \end{pmatrix}, A_1 = \begin{pmatrix} 0 & 0.5 \\ 0.5 & 0 \end{pmatrix} \text{ et } b = 1$$

2.4.2 Programmation quadratique avec des contraintes quadratiques

La contrainte quadratique convexe

$$(Ax + b)^T (Ax + b) - c^T x - d \leq 0$$

avec $x \in \mathbb{R}^k$, peut s'écrire comme suit :

$$\begin{bmatrix} I & Ax + b \\ (Ax + b)^T & c^T x + d \end{bmatrix} \succeq 0$$

Cette dernière inégalité peut être exprimée par :

$$F(x) = F_0 + x_1 F_1 + \dots + x_k F_k \succeq 0$$

avec

$$F_0 = \begin{bmatrix} I & b \\ b^T & d \end{bmatrix}, \quad F_i = \begin{bmatrix} 0 & a_i \\ a_i^T & c_i^T \end{bmatrix}, \quad i = 1, \dots, k$$

où $A = [a_1, \dots, a_k]$.

Par conséquent, un programme convexe quadratique avec des contraintes quadratiques de type

$$(QCQP) \begin{cases} \min f_0(x) \\ f_i(x) \leq 0, \quad i = 1, \dots, L \end{cases}$$

où chaque f_i est une fonction quadratique convexe

$$f_i(x) = (A_i x + b)^T (A_i x + b) - c_i^T x - d_i$$

Peut-être formuler comme suit :

$$(QCQP) \begin{cases} \min t \\ \begin{pmatrix} I & A_0 x + b \\ (A_0 x + b)^T & c_0^T x + d_0 + t \end{pmatrix} \succeq 0 \\ \begin{pmatrix} I & A_i x + b \\ (A_i x + b)^T & c_i^T x + d_i + t \end{pmatrix} \succeq 0, \quad i = 1, \dots, L \end{cases}$$

$(QCQP)$ est un programme semi-défini avec les variables $x \in \mathbb{R}^k$ et $t \in \mathbb{R}$.

2.4.3 Le problème min-max des valeurs propres

Ce problème a été étudié depuis des années (vers les années 1950) en Algèbre linéaire comme suit :

Chercher une valeur optimale

$$(VP) : \lambda^* = \min_{y \in \mathbb{R}^n} \lambda_{\max}(C + A(y))$$

où $C \in M_n$ et

$$A_{\text{opérateur}} : \mathbb{R}^n \longrightarrow M_n$$

$$y \longmapsto A(y) = \sum_{i=1}^n y_i A_i, \quad A_i \in M_n$$

En fait trouver la plus grande valeur propre d'une matrice symétrique A correspond au problème suivant :

$$\max_x [x^T A x : \|x\| = 1]$$

où $x \in \mathbb{R}^n - \{0\}$ est un vecteur propre de la matrice A .

Soit encore en considérant sur les matrices symétriques le produit scalaire

$$\langle A, B \rangle = \text{trace}(AB)$$

$$\max_x [\langle A, x x^T \rangle : \langle I, x x^T \rangle = 1]$$

Ou encore, X étant symétrique

$$(P1) : \max_X [\langle A, X \rangle : \langle I, X \rangle = 1, X \text{ est un SDP de rang } 1]$$

Si on Considère le problème semi-défini linéaire suivant :

$$(P1) : \max_X [\langle A, X \rangle : \langle I, X \rangle = 1, X \succeq 0]$$

Ce problème admet des solutions optimales car l'ensemble des solutions réalisables est compact. La fonction objective étant linéaire et l'ensemble des solutions réalisables étant convexe, l'optimum est atteint au moins en un point extrémal, donc en un X matrice de rang 1. X est aussi solution optimale du problème $(P1)$. Bien sur toute solution optimale de $(P1)$ est solution optimale de (VP) .

Le problème (VP) s'écrit sous forme d'un problème (SDP) comme suit :

$$\left(\lambda^* = \min_{y \in \mathbb{R}^n} \lambda_{\max}(C + A(y)) \right) \iff \begin{cases} \min_{y \in \mathbb{R}^n} \lambda \\ \lambda I \succeq C + A(y) \\ y \in \mathbb{R}^n \end{cases}$$

$$\iff \begin{cases} \min_{y \in \mathbb{R}^n} \lambda \\ \lambda I - C - A(y) \succeq 0 \\ y \in \mathbb{R}^n \end{cases}$$

Le dual de ce problème est un problème (SDP) formulé par :

$$\begin{cases} \max \langle C, X \rangle \\ A^T(X) = 0 \\ X \succeq 0 \\ \text{trace}(X) = 1 \end{cases}$$

2.4.4 Approximation logarithmique de Tchebychev

Supposons qu'on cherche à résoudre approximativement le système d'équations

$$a_i^T x = b_i \text{ pour } i = 1, \dots, m$$

Notons que la recherche des paramètres d'une régression peut être solution d'un tel système.

Le terme approximativement a été utilisé parce que la résolution exacte est très difficile (trop de contraintes). On peut dès lors choisir plusieurs critères pour déterminer la meilleure solution. Le plus courant est la minimisation de la somme des carrés des écarts $(a_i^T x - b_i)$, cependant, dans l'approximation de Tchebychev, on minimise la norme du

résidu l_∞ , **i.e.**, on minimise l'écart maximal en valeur absolue, selon

$$\min \max_i |a_i^T x - b_i|$$

Ce problème peut se mettre sous la forme du programme linéaire suivant :

$$\begin{cases} \min t \\ -t \leq a_i^T x - b_i \leq t, \quad i = 1, \dots, m \end{cases}$$

où x est une variable et t est une variable auxiliaire.

Dans certaines applications les b_i sont des quantités s'exprimant dans une unité de puissance ou d'intensité et généralement de logarithme. Dans ce cas, il est préférable d'un point de vue purement physique de minimiser le maximum des écarts entre les logarithmes de $a_i^T x$ et b_i , ce qui se traduit par :

$$\min \max_i |\log a_i^T x - \log b_i| \tag{2.1}$$

On suppose que $b_i > 0$ et

$$a_i^T x > 0, \forall i = 1, \dots, m$$

Ce problème est appelé Problème d'approximation logarithmique de Tchebychev et peut être transformé en un programme semi-défini. On transforme alors les contraintes en

$$-\log u \leq \log \frac{a_i^T x}{b_i} \leq \log u$$

d'où

$$\frac{1}{t} \leq \frac{a_i^T x}{b_i} \leq t$$

Le problème (2.1) est équivalent au problème semi-défini suivant :

$$\begin{cases} \min t \\ \frac{1}{t} \leq \frac{a_i^T x}{b_i} \leq t, \quad i = 1, \dots, m \end{cases}$$

où

$$\left\{ \begin{array}{l} \min t \\ \left(\begin{array}{ccc} t - \frac{a_i^T x}{b_i} & 0 & 0 \\ 0 & \frac{a_i^T x}{b_i} & 1 \\ 0 & 1 & t \end{array} \right) \succeq 0, \quad i = 1, \dots, m \end{array} \right.$$

Le premier terme de la diagonale est responsable de l'inégalité

$$\frac{a_i^T x}{b_i} \leq t$$

tandis que

$$\frac{a_i^T x}{b_i} \geq \frac{1}{t}$$

se traduit par la matrice 2×2 inférieure droite.

En effet, pour une matrice 2×2 , le fait d'être semi-définie positive implique la non-négativité du déterminant, d'où on déduit

$$\frac{a_i^T x}{b_i} t - 1 \geq 0$$

et finalement l'inégalité recherchée.

Donc, cet exemple illustre deux points importants. Le premier, il montre que la programmation semi-définie inclut beaucoup de problèmes d'optimisation qui ne semblent pas pour la première vue. Et, le second, il montre que le problème est plus général qu'un programme linéaire, malgré la proche analogie.

2.5 Méthode de points intérieurs pour résoudre (*SDP*)

La méthode de points intérieurs est l'une des méthodes les plus utilisées et les plus efficaces pour la résolution des problèmes (*SDP*), elle est relativement nouvelle et s'apparente à la méthode projective de Karmarkar pour la programmation linéaire. Derrière le terme points intérieurs découle trois différents types de méthodes :

1. Les méthodes affines.
2. Les méthodes de réduction du potentiel.
3. Les méthodes de trajectoire centrale.

2.5.1 Méthodes affines

L'idée remonte en 1967 par Dikin [14], Puis reprise et développée par plusieurs chercheurs au milieu des années 80. Il s'agit pratiquement de l'algorithme de Karmarkar sans fonction potentiel et sans transformation projective, on utilise une transformation affine et on remplace la contrainte de non négativité par un ellipsoïde qui contient le nouvel itéré.

L'algorithme correspondant à cette méthode est d'une structure simple, malheureusement, il n'est pas facile de démontrer la polynomialité. À cet égard, on trouve, en 1967 Dikin [14] à prouvé la convergence de la méthode affine primale sous des hypothèses de non dégénérescence. Actuellement la convergence pour les méthodes affines primales ou duales n'est pas prouvée. Par contre, en 1990 Monteiro et al. [30] ont démontré que la méthode affine primale-duale est de complexité polynomiale ($O(nL^2)$), où L représente le nombre de bits requis pour stocker (et traiter) les données.

2.5.2 Méthodes de réduction du potentiel

Ces méthodes sont le fruit direct d'une grande partie des études acharnées menées par plusieurs chercheurs vers la fin des années 80. En 1995, Alizadeh [2] a proposé une méthode projective primale-duale pour résoudre les problèmes de SDP. En 2003, Merikhi et al. [8] ont fait l'étude théorique et numérique de l'algorithme proposé par Alizadeh [3]. En 2007, Merikhi et al. [7] ont proposé une méthode primale réalisable de réduction du potentiel. L'algorithme de ces méthodes est basé sur les deux éléments principaux suivant :

1. La transformation projective.

2. La réduction de la fonction potentielle primale-duale.

Transformation projective

Dans ces méthodes à chaque itération on utilise la transformation projective T_k qui nous permet de ramener le problème (SDP) à une forme réduite plus maniable. En effet, supposons à l'itération k , qu'on dispose d'une solution strictement réalisable $X_k \in S_{++}^n$. La factorisation de X_k donne une matrice triangulaire inférieure L_k telle que $X_k = L_k L_k^T$. Il existe plusieurs choix de L_k par exemple L_k peut être trouvée par la factorisation de Cholesky de X_k ou bien L_k est la racine carrée de X_k , **i.e.**, $L_k = X_k^{\frac{1}{2}}$. Soit r un entier fixé, on définit la transformation projective comme suit :

$$\begin{aligned} T_k : S^n &\rightarrow S^n \times \mathbb{R}^r \\ X &\mapsto T_k(X) = (\bar{X}, \bar{x}) \end{aligned}$$

où

$$\bar{X} = \frac{(n+r)L_k^{-1}XL_k^{-T}}{r + \langle X_k^{-1}, X \rangle}, \quad \bar{x} = \frac{(n+r)}{r + \langle X_k^{-1}, X \rangle} e_r, \quad e_r = (1, \dots, 1)^T \in \mathbb{R}^r$$

Aussi, la transformation inverse est donnée par :

$$X = T_k^{-1}(\bar{X}, \bar{x}) = r \frac{L_k \bar{X} L_k^T}{e_r^T \bar{x}}$$

Sous la transformation T_k , le problème (SDP) est transformé en un problème de programmation semi-définie linéaire ($TSDP$). Puis, comme dans la programmation linéaire, on remplace les contraintes des inégalités $\bar{X} \succeq 0$ et $\bar{x} \geq 0$ de ($TSDP$) par la boule du centre (I, e_r) et de rayon β avec $0 < \beta < 1$. Donc le problème ($TSDP$) devient

$$\begin{cases} \min[\langle \bar{C}, \bar{X} \rangle - \frac{z_k e_r^T \bar{x}}{r}] \\ \langle A_i^k, \bar{X} \rangle - \frac{b_i e_r^T \bar{x}}{r} = 0, \quad i = 1, \dots, m \\ \text{trace}(\bar{X}) + e_r^T \bar{x} = n + r \\ \|\bar{X} - I\|^2 + \|\bar{x} - e_r\|^2 \leq \beta^2 < 1 \end{cases}$$

où

$$\bar{C} = L_k^T C L_k, z_k = \langle C, X \rangle \text{ et } A_i^k = L_k^T A_i L_k$$

Ce problème est un programme mathématique convexe et différentiable, donc les conditions de **K.K.T** sont nécessaires et suffisantes. La solution optimale obtenue est

$$\begin{cases} \bar{X}^* = I - \beta P_k \\ \bar{x}^* = (1 - \beta p_k) e_r \end{cases}$$

où

$$P_k = V_k / (\|V_k\|^2 + r\alpha_k^2)^{\frac{1}{2}}, p_k = \alpha_k / (\|V_k\|^2 + r\alpha_k^2)^{\frac{1}{2}}$$

avec

$$V_k = C_k + \sum_{i=1}^m y_i A_i^k, \alpha_k = -\frac{1}{r} \left(\sum_{i=1}^m b_i y_i + z_k \right)$$

$y \in \mathbb{R}^m$ est la solution du système linéaire symétrique défini positif suivant :

$$My = d, M_{i,j} = \langle A_i^k, A_j^k \rangle + \frac{b_i b_j}{r}, d_i = -\frac{b_i z_k}{r} - \langle C_k, A_i^k \rangle, i, j = 1, \dots, m$$

Le nouvel itéré est donc

$$X_{k+1} = T_k^{-1}(\bar{X}, \bar{x})$$

Fonction potentiel primale-duale

Pour tout $X, S \in S_{++}^n$ la fonction potentiel primale-duale est définie par :

$$\psi(X, S) = (n + r) \ln \langle X, S \rangle - \ln \det(XS)$$

Suite aux travaux fondateurs de Nesterov et al. [34], la fonction potentiel a reconnu un regain d'intérêt et joue un rôle très important dans le développement des algorithmes de réduction du potentiel. Les algorithmes correspondants à ces méthodes possèdent une complexité polynomiale, ils nécessitent $O(\sqrt{n} |\ln \varepsilon|)$ itérations pour réduire le saut de

dualité ($\langle X, S \rangle \leq \varepsilon$ où ε est une précision donnée). (Voir Alizadeh [2]).

2.5.3 Méthodes de trajectoire centrale

Les méthodes de trajectoire centrale primale-duale ont été introduites à la même époque que les méthodes de réduction du potentiel et pleinement développées au début des années 90. Elles ont attiré une grande attention de la part des chercheurs dans le monde entier et elles montrent en général un excellent comportement pratique et théorique (une complexité polynomiale et une convergence super linéaire). On trouve en 1996, les travaux de Helmberg et al. [17] qui ont proposé un algorithme primal-dual de trajectoire centrale pour (SDP). En 1997, Monteiro [32] a proposé une méthode primale-duale de trajectoire centrale et a montré la convergence polynomiale de l'algorithme à court et long pas. En 1999, Ji et al. [18] ont étudié la convergence de la méthode de trajectoire centrale de type prédicteur-correcteur. Aussi, Monteiro et al. [33] ont proposé une méthode primale-duale de trajectoire centrale et ont montré la convergence polynomiale vers une solution optimale. Ces travaux se poursuivent à ce jour, on trouve ainsi ceux de Halicka et al. en 2002 [16], qui ont proposé une étude sur la convergence de la méthode de trajectoire centrale en optimisation semi-définie. En 2007 Koulaei et al. [24] ont proposé une extension de l'algorithme de Mehrotra prédicteur-correcteur basé sur la direction de Nesterov et Todd. En 2012, Liu et al. [27] ont présenté un nouveau algorithme de type correcteur de second ordre pour (SDP) et ont prouvé la convergence polynomiale de ce dernier pour la direction de Nesterov et Todd.

Les méthodes de trajectoire centrale reposent sur les principes suivants, on associe au problème initial (SDP) un problème perturbé (SDP) $_{\mu}$, puis on applique les conditions de (KKT) sur ce problème, on obtient le système non linéaire suivant :

$$\begin{cases} \text{trace}(A_i X) = b_i, \quad i = 1, \dots, m, \quad X \succ 0 \\ \sum_{i=1}^m y_i A_i + S = C, \quad S \succ 0 \\ XS = \mu I, \quad \mu > 0 \end{cases} \quad (2.2)$$

où, on suppose

- (H_1) $\widehat{F}(SDP) \times \widehat{F}(DSDP) \neq \phi$.
- (H_2) Les matrices A_i , $i = 1, \dots, m$, sont linéairement indépendantes.

Tels que $\widehat{F}(SDP)$ et $\widehat{F}(DSDP)$ désigne les ensembles des solutions strictement réalisables de (SDP) et $(DSDP)$ respectivement, **i.e.**,

$$\begin{aligned}\widehat{F}(SDP) &= \{X \in S_{++}^n : AX = b\} \\ \widehat{F}(DSDP) &= \{(y, S) \in \mathbb{R}^m \times S_{++}^n : A^*y + S = C\}\end{aligned}$$

Sous les hypothèses (H_1) et (H_2) , le système (2.2) admet une solution unique

$$(X(\mu), y(\mu), S(\mu)) \text{ pour chaque } \mu > 0$$

Cette solution s'appelle μ -centre de (SDP) et $(DSDP)$. L'ensemble des μ -centres, avec

$$\Lambda = \{X(\mu), y(\mu), S(\mu), \mu > 0\}$$

construit la trajectoire centrale qui converge vers la solution optimale primale-duale de (SDP) et $(DSDP)$ quand μ tend vers zéro.

Le système (2.2) doit être résolu par la méthode de Newton, cette dernière consiste à trouver la direction $\Delta W = (\Delta X, \Delta y, \Delta S)$, en résolvant le système linéaire suivant :

$$\begin{cases} A\Delta X = b, X \succ 0 \\ A^*\Delta y + \Delta S = C, S \succ 0 \\ S\Delta X + X\Delta S = \mu I - XS, \mu > 0 \end{cases} \quad (2.3)$$

Le nouvel itéré est donné par :

$$X^+ = X + \Delta X$$

$$y^+ = y + \Delta y$$

$$S^+ = S + \Delta S$$

Signalant que la symétrie des itérés X^+ et S^+ n'est pas préservée pour toutes les directions trouvées dans (2.3), pour cela, plusieurs aménagements ont été proposés par des chercheurs afin de résoudre ce problème. On cite par exemple les travaux de Zhang [47], Helmberg et al, Kojima et al et Monteiro [4, 17, 23], Alizadeh-Heaberly-Overton [3] et Nesterov-Todd [41].

Actuellement, plusieurs chercheurs bai et autres [5, 9, 10, 15, 45] ont introduit la notion des fonctions noyaux pour trouver une classe de directions, à travers laquelle les auteurs ont pu améliorer la complexité algorithmique de méthodes à court et long-pas.

Dans le chapitre suivant, on présente d'une manière détaillée une étude théorique et numérique sur une méthode de type barrière. Et on mis en œuvre d'une manière effective l'algorithme correspondant pour la résolution d'un programme semi-défini (*SDP*).

Chapitre 3

Méthode barrière logarithmique via les fonctions minorantes

3.1 Introduction

L'idée de cette méthode consiste à résoudre SDP avec une nouvelle approche de barrière logarithmique. Nous utilisons la méthode de Newton pour traiter les équations perturbées associées pour obtenir une direction de descente. Pour le calcul du pas de déplacement, on propose une nouvelle variante basée sur la technique des fonctions minorantes, cette idée contribue à la réduction maximale de la complexité de notre algorithme relativement aux méthodes de recherche linéaire classique (Wolfe, Armijo-Goldstein).

Nous considérons le problème de programmation semi-définie suivant :

$$\begin{cases} \min b^T x \\ \sum_{i=1}^m x_i A_i - C \in S_n^+ \\ x \in \mathbb{R}^m \end{cases} \quad (1)$$

où S_n^+ désigne le cône des matrices $n \times n$ symétriques semi-définies positives et les matrices C, A_i , avec $i = 1, \dots, m$, sont des matrices symétriques données et $b \in \mathbb{R}^m$.

Le problème (1) est le dual du problème suivant :

$$\begin{cases} \max \langle C, Y \rangle \\ \langle A_i, Y \rangle = b_i, \forall i = 1, \dots, m \\ Y \in S_n^+ \end{cases} \quad (2)$$

Nous désignons par $\langle C, Y \rangle$, la trace de la matrice $(C^T Y)$. Il est rappelé que $\langle \cdot, \cdot \rangle$ correspond à un produit interne sur l'espace des matrices M_n .

Leurs ensembles réalisables impliquant un cône convexe non polyèdre, des matrices semi-définis positives, s'appellent des programmes semi-définie linéaires.

A priori, un des avantages du problème (1) par rapport au problème (2) est que l'argument de la fonction que l'on minimise est un vecteur, alors qu'il est une matrice dans (2). D'autre part sous des hypothèses convenables, la résolution de (1) est équivalente à celle de (2) au sens que la solution optimale de l'un des deux problèmes se déduit directement de l'autre via le théorème des écarts complémentaires.

Dans tout ce qui suit, nous désignons par :

1. $X = \left\{ x \in \mathbb{R}^m : \sum_{i=1}^m x_i A_i - C \in S_n^+ \right\}$, l'ensemble des solutions réalisables de (1).
2. $\hat{X} = \left\{ x \in \mathbb{R}^m : \sum_{i=1}^m x_i A_i - C \in \text{int}(S_n^+) \right\}$, l'ensemble des solutions strictement réalisables de (1).
3. $F = \{ Y \in S_n^+ : \langle A_i, Y \rangle = b_i, \forall i = 1, \dots, m \}$, l'ensemble des solutions réalisables de (2).
4. $\hat{F} = \{ Y \in F : Y \in \text{int}(S_n^+) \}$, l'ensemble des solutions strictement réalisables de (2).

Avec $\text{int}(S_n^+)$ est l'ensemble des matrices symétriques définies positives ($n \times n$).

Le problème (1) est approximé par le problème $(SDP)_\eta$ suivant :

$$\begin{cases} \min f_\eta(x) \\ x \in \mathbb{R}^m \end{cases} \quad (SDP)_\eta$$

avec $\eta > 0$ paramètre de pénalisation, et $f_\eta : \mathbb{R}^m \rightarrow]-\infty, +\infty]$ la fonction barrière définie par :

$$f_\eta(x) = \begin{cases} b^T x + n\eta \ln \eta - \eta \ln \left[\det \left(\sum_{i=1}^m x_i A_i - C \right) \right] & \text{si } x \in \widehat{X} \\ +\infty & \text{sinon} \end{cases}$$

Pour résoudre $(SDP)_\eta$, on utilise une méthode de descente classique de type Newton. La difficulté au niveau de la recherche linéaire est la présence du déterminant dans la définition de f_η entraîne un coût très élevé dans les procédures classiques de la recherche linéaire. Dans notre approche, au lieu de minimiser f_η , le long de la direction de descente d au point courant x , on va minimiser une fonction G telle que :

$$\frac{1}{\eta} [f_\eta(x + \alpha d) - f_\eta(x)] = G(\alpha) \geq \check{G}(\alpha), \forall \alpha > 0$$

avec

$$G(0) = \check{G}(0) = 0, \quad G'(0) = \check{G}'(0) < 0$$

\check{G} doit être choisie de telle manière que le calcul de α soit facile, et \check{G} suffisamment proche de G pour donner à chaque itération une décroissance significative à f_η . On propose à ce propos des fonctions \check{G} pour lesquelles la solution optimale de α est obtenue explicitement, la bonne qualité des approximations de G est assurée par la condition

$$G''(0) = \check{G}''(0)$$

Information brève en programmation linéaire semi-définie

Dans le reste de notre travail, on admet les hypothèses suivantes :

(A1) Le système d'équations $\langle A_i, Y \rangle = b_i, i = 1, \dots, m$ est de rang m .

(A2) Les ensembles \widehat{X} et \widehat{F} sont non vides.

On sait alors que (voir [3, 8])

1. Les ensembles de solutions optimales de (2) et (1) sont convexes compacts non vides.
2. Si \bar{x} est une solution optimale de (1), alors \bar{Y} est une solution optimale de (2) si et seulement si :

$$\bar{Y} \in F \text{ et } \left(\sum_{i=1}^m \bar{x}_i A_i - C \right) \bar{Y} = 0$$

3. Si \bar{Y} est une solution optimale de (2), alors \bar{x} est une solution optimale de (1) si et seulement si :

$$\bar{x} \in X \text{ et } \left(\sum_{i=1}^m \bar{x}_i A_i - C \right) \bar{Y} = 0$$

Selon les hypothèses **(A1)** et **(A2)**, la résolution du problème (1) permet d'obtenir celle de (2) et vice versa.

L'avantage principal de $(SDP)_\eta$ réside dans la convexité stricte de sa fonction objective et son domaine faisable. Par conséquent, les conditions d'optimalité sont nécessaires et suffisantes. Cela favorise les études théoriques et numériques du problème.

Tout d'abord, il est nécessaire de montrer que $(SDP)_\eta$ a au moins une solution optimale.

3.2 Existence et unicité de solution optimale de problème $(SDP)_\eta$ et sa convergence vers le problème (1)

3.2.1 Existence de solution optimale de problème $(SDP)_\eta$

Premièrement, nous commençons par la définition suivante :

Définition 3.2.1 Soit f une fonction définie de \mathbb{R}^m dans $\mathbb{R} \cup \{\infty\}$, f s'appelle inf-compact si pour tout $\eta > 0$, l'ensemble $S_\eta(f) = \{x \in \mathbb{R}^m : f(x) \leq \eta\}$ est compact, ce qui vient notamment dire que son cône de récession est réduit à zéro.

Comme la fonction f_η la valeur $+\infty$ sur la limite de X et est différentiable sur \widehat{X} , elle est inférieure semi-continue. Afin de prouver que $(SDP)_\eta$ a une solution optimale, il suffit de prouver que le cône de récession de f_η

$$S_0((f_\eta)_\infty) = \{d \in \mathbb{R}^m, (f_\eta)_\infty(d) \leq 0\}$$

est réduit à zéro **i.e.**,

$$d = 0 \text{ si } (f_\eta)_\infty(d) \leq 0$$

où $(f_\eta)_\infty$ est défini pour $x \in \widehat{X}$ comme

$$(f_\eta)_\infty(d) = \lim_{\alpha \rightarrow +\infty} \left[\xi(\alpha) = \frac{f_\eta(x + \alpha d) - f_\eta(x)}{\alpha} \right]$$

Cela nécessite la proposition suivante :

Proposition 8 [11] *Si $b^T d \leq 0$ et $\sum_{i=1}^m d_i A_i \in \widehat{X}$ alors $d = 0$.*

Comme f_η est inf-compact, alors il existe au moins une solution optimale du problème $(SDP)_\eta$.

3.2.2 Problème $(SDP)_\eta$ a une solution optimale unique

Propriétés fondamentales de f_η

Pour $x \in \widehat{X}$, nous introduisons la matrice positive définie symétrique $B(x)$ de rang m , et la matrice triangulaire inférieure $L(x)$, telle que :

$$B(x) = \sum_{i=1}^m x_i A_i - C = L(x)L^T(x)$$

et définissons, pour $i, j = 1, \dots, m$

$$\widehat{A}_i(x) = [L(x)]^{-1} A_i [L^T(x)]^{-1}$$

et

$$b_i(x) = \text{trace}(\widehat{A}_i(x)) = \text{trace}(A_i B^{-1}(x))$$

$$\Delta_{ij}(x) = \text{trace}(B^{-1}(x) A_i B^{-1}(x) A_j) = \text{trace}(\widehat{A}_i(x) \widehat{A}_j(x))$$

ainsi, $b(x) = (b_i(x))_{i=1,\dots,m}$ est un vecteur de \mathbb{R}^m et $\Delta(x) = (\Delta_{ij}(x))_{i,j=1,\dots,m}$ est une matrice symétrique de rang m .

Théorème 3.2.1 [11] *La fonction f_η est deux fois continûment différentiables sur \widehat{X} . Plus précisément, pour tout $x \in \widehat{X}$ on a*

- (a) $\nabla f_\eta(x) = b - \eta b(x)$.
- (b) $H = \nabla^2 f_\eta(x) = \eta \Delta(x)$.
- (c) *La matrice $\Delta(x)$ est définie positive.*

Puisque f_η est strictement convexe, $(SDP)_\eta$ a au plus une solution optimale.

Nous désignons par x_η ou $x(\eta)$ la solution optimale unique de $(SDP)_\eta$.

3.2.3 Comportement de la solution lorsque $\eta \rightarrow 0$

Dans ce qui suit, nous serons intéressés par le comportement de la valeur optimale et la solution optimale $x(\eta)$ du problème $(SDP)_\eta$. Pour cela, introduisons la fonction

$$f : \mathbb{R}^m \times \mathbb{R} \rightarrow]-\infty, +\infty]$$

définie par :

$$f(x, \eta) = \begin{cases} f_\eta(x) & \text{si } \eta > 0 \\ b^T x & \text{si } \eta = 0, x \in X \\ +\infty & \text{sinon} \end{cases}$$

Il est facile de vérifier que la fonction f est convexe et semi-continue inférieure sur $\mathbb{R}^m \times \mathbb{R}$, voir par exemple R. T. Rockafellar [38].

On définit la fonction $m : \mathbb{R} \rightarrow]-\infty, +\infty]$ par :

$$m(\eta) = \inf_x [f(x, \eta) : x \in \mathbb{R}^m]$$

Cette fonction est convexe. En outre, nous avons $m(0)$ et $m(\eta)$ est la valeur optimale de $(SDP)_\eta$ pour $\eta > 0$. Il est clair que pour $\eta > 0$, nous obtenons

$$m(\eta) = f_\eta(x(\eta)) = f(x(\eta), \eta)$$

et

$$0 = \nabla f_\eta(x(\eta)) = \nabla_x f(x(\eta), \eta) = b - \eta b(x_\eta)$$

Nous sommes maintenant intéressés par la différentiabilité des fonctions $m(\eta)$ et $x(\eta)$ sur $(0, +\infty)$.

Proposition 9 [11] *Les fonctions m et x sont continûment différentiables sur $(0, +\infty)$. On a pour tout $\eta > 0$*

$$\begin{aligned} \eta \Delta(x_\eta) x'(\eta) - b(x_\eta) &= 0 \\ m'(\eta) &= n + n \ln(\eta) - \ln \det(B(x_\eta)) \end{aligned}$$

En outre

$$m(0) \leq b^T x(\eta) \leq m(0) + n\eta$$

Désignons par S_D l'ensemble des solutions optimales de (1), on sait que cet ensemble est un convexe compact non vide. La distance du point x à S_D est définie par :

$$d(x, S_D) = \inf_z [\|x - z\| : z \in S_D]$$

Le résultat suivant concerne le comportement de $x(\eta)$ et $m(\eta)$ lorsque $\eta \rightarrow 0$.

Théorème 3.2.2 [11] *Lorsque $\eta \rightarrow 0$, $d(x, S_D) \rightarrow 0$ et $m(\eta) \rightarrow m(0)$.*

Remarque 3.2.1 *Nous savons que si l'un des problèmes (1) et (2) a une solution optimale et les valeurs de leurs fonctions objectives sont égales et finies, l'autre problème a une solution optimale.*

3.3 Direction de descente de Newton et recherche linéaire

De par La présence de la fonction barrière, le problème $(SDP)_\eta$ peut être considéré comme sans contraintes. On peut donc le résoudre par une méthode de descente classique. Comme f_η prend la valeur $+\infty$ sur la frontière de X , alors les itérés x sont dans \widehat{X} . Ainsi la méthode que nous proposons est une méthode de points intérieurs.

Soit $x \in \widehat{X}$ l'itéré en cours. Comme direction de descente en x , prenons la direction de Newton d solution du système linéaire :

$$\nabla^2 f_\eta(x)d = -\nabla f_\eta(x)$$

En vertu du théorème 3.2.1, le système linéaire est équivalent au système suivant :

$$\Delta(x)d = b(x) - \frac{1}{\eta}b \tag{3}$$

où $b(x)$ et $\Delta(x)$ sont définies en 3.2.2.

La matrice $\Delta(x)$ étant symétrique, définie positive, le système linéaire (3) peut être efficacement résolu via la décomposition de Cholesky. Bien évidemment, on admet que $\nabla f(x) \neq 0$ (sinon l'optimum est atteint). Il s'ensuit que $d \neq 0$. La direction d étant calculée, on cherche $\bar{\alpha} > 0$ donnant une décroissance significative à f_η sur la demi-droite $x + \alpha d$, $\alpha > 0$, tout en conservant la définie positivité de la matrice $B(x + \bar{\alpha}d)$. Ensuite, la prochaine itération sera prise égale à $x + \bar{\alpha}d$. Pour se faire, on considère la fonction

$$\begin{aligned} G(\alpha) &= \frac{1}{\eta}[f_\eta(x + \alpha d) - f_\eta(x)], \quad x + \alpha d \in \widehat{X} \\ G(\alpha) &= \frac{1}{\eta}b^T d\alpha - \ln \det(B(x + \alpha d)) + \ln \det(B(x)) \end{aligned}$$

Puisque $\nabla^2[f_\eta(x)]d = -\nabla f_\eta(x)$, on a

$$d^T \nabla^2 f_\eta(x) d = -d^T \nabla f_\eta(x) = d^T b(x) - \eta d^T b$$

Pour simplifier les notations, on prend

$$B = B(x) = \sum_{i=1}^m x_i A_i - C \text{ et } H = \sum_{i=1}^m d_i A_i$$

B étant symétrique définie positive, il existe une matrice triangulaire inférieure L telle que $B = LL^t$.

Par suite, on pose

$$E = L^{-1}H(L^{-1})^t$$

Puisque $d \neq 0$, l'hypothèse **(A1)** implique que $H \neq 0$ et alors $E \neq 0$ (où 0 désigne la matrice nulle).

Lemme 3.3.1 *Soit*

$$\hat{t} = \sup \{t : 1 + t\lambda_i > 0, i = 1, \dots, n\}$$

Pour tout $\alpha \in [0, \hat{\alpha}[$, la fonction suivante $G(\alpha)$ est bien définie

$$G(\alpha) = \sum_{i=1}^n [\alpha(\lambda_i - \lambda_i^2) - \ln(1 + \alpha\lambda_i)] \quad (3.1)$$

Preuve :

On a

$$\begin{aligned} G(\alpha) &= \frac{1}{\eta} [f_\eta(x + \alpha d) - f_\eta(x)] \\ &= \frac{1}{\eta} \alpha b^T d - \ln \det \left(\sum_{i=1}^m (x_i + \alpha d) A_i - C \right) + \ln \det \left(\sum_{i=1}^m x_i A_i - C \right) \end{aligned}$$

mais

$$\begin{aligned}
\sum_{i=1}^m (x_i + \alpha d) A_i - C &= \left(\sum_{i=1}^m x_i A_i - C \right) + \alpha \sum_{i=1}^m d_i A_i \\
&= B + \alpha H \\
&= B (I + \alpha B^{-1} H)
\end{aligned}$$

donc

$$\begin{aligned}
G(\alpha) &= \frac{\alpha}{\eta} b^t d - \ln \det B(x) - \ln \det (I + \alpha B^{-1} H) + \ln \det (B(x)) & (3.2) \\
G(\alpha) &= \frac{\alpha}{\eta} b^t d - \ln \det (I + \alpha B(x)^{-1} H) \\
&= \frac{\alpha}{\eta} b^t d - \ln \det (I + \alpha E)
\end{aligned}$$

mais $B = LL^t$, alors $B^{-1} = (L^t)^{-1} L^{-1}$. Puisque

$$\nabla f_\eta(x) = b - \eta b(x)$$

on a

$$b_i(x) = \text{trace}(\widehat{A}_i(x)) = \text{trace}(A_i B^{-1}(x))$$

et

$$[\nabla^2 f_\eta(x_\eta)] d = -\nabla f_\eta(x_\eta)$$

donc

$$d^t b = d^t \nabla f_\eta(x) + \eta d^t b(x) \quad (3.3)$$

En raison du fait que la direction d satisfait

$$[\nabla^2 f_\eta(x_\eta)] d = -\nabla f_\eta(x_\eta)$$

alors, on a

$$d^t [\nabla^2 f_\eta(x_\eta)] d = -d^t \nabla f_\eta(x_\eta) \quad (3.4)$$

En remplaçant (3.4) dans (3.3), on obtient

$$\begin{aligned} d^t b &= d^t \nabla f_\eta(x) + \eta d^t b(x) \\ &= -d^t [\nabla^2 f_\eta(x_\eta)] d + \eta d^t b(x) \\ &= -d^t [\nabla^2 f_\eta(x_\eta)] d + \eta \sum d^t \text{trace}(A_i B^{-1}(x_\eta)) \\ &= -d^t [\nabla^2 f_\eta(x_\eta)] d + \eta \text{trace}(E) \end{aligned} \quad (3.5)$$

Puisque $d \neq 0$, l'hypothèse **(A1)** implique que $H \neq 0$ et par suite $E \neq 0$. On a

$$\begin{aligned} d^t [\nabla^2 f_\eta(x_\eta)] d &= d^t [\eta \Delta(x_\eta)] d \\ &= \eta d^t [\text{trace}(B^{-1}(x_\eta) A_i B^{-1}(x_\eta) A_j)] d \\ &= \eta \sum_{i=1}^n d_i^2 \text{trace}(B^{-1}(x_\eta) A_i B^{-1}(x_\eta) A_j) \\ &= \eta \text{trace}(E^2) \end{aligned}$$

Puis, en remplaçant dans (3.5), on trouve

$$d^t b = \eta \text{trace}(E) - \eta \text{trace}(E^2) \quad (3.6)$$

En prenant compte de cette notation, pour tout $\alpha > 0$, tel que $I + \alpha E$ est définie positive, et avec le remplacement de (3.6) dans (3.2), nous donne

$$G(\alpha) = \alpha (\text{trace}(E) - \text{trace}(E^2)) - \ln \det(I + \alpha E) \quad (3.7)$$

Désignons par λ_i , les valeurs propres de la matrice symétrique E , donc

$$G(\alpha) = \sum_{i=1}^n [\alpha (\lambda_i - \lambda_i^2) - \ln(1 + \alpha \lambda_i)], \quad \alpha \in [0, \hat{\alpha}[\quad (3.8)$$

avec

$$\hat{\alpha} = \sup[\alpha : 1 + \alpha\lambda_i > 0 \text{ pour tout } i = 1, \dots, n] = \sup[\alpha : x + \alpha d \in \hat{X}] \quad (3.9)$$

La preuve est complète. \square

Notons que $\hat{\alpha} = +\infty$ si E est une semi-définie positive et $0 < \alpha < \infty$ sinon. Il est clair que G est convexe sur $[0, \hat{\alpha}[$, $G(0) = 0$ et

$$0 < \sum_i \lambda_i^2 = G''(0) = -G'(0)$$

En outre, $G(\alpha) \rightarrow +\infty$ lorsque $\alpha \rightarrow \hat{\alpha}$. Il s'ensuit, qu'il existe un point unique α_{opt} tel que $G'(\alpha_{opt}) = 0$, G atteint son minimum en ce point.

Malheureusement, il n'existe pas de formule explicite donnant α_{opt} , et la résolution de l'équation $G'(\alpha_{opt}) = 0$ par des méthodes itératives nécessite à chaque itération le calcul de G et G' . Ces calculs sont trop coûteux du fait que l'expression de G en (3.7) contient le déterminant qui n'est pas facile à calculer et (3.8) nécessite la connaissance des valeurs propres de E , c'est un problème numérique de taille. Ces difficultés nous ont conduits à chercher d'autres alternatives. À partir de la donnée de la matrice E , il est facile d'obtenir les quantités suivantes :

$$\text{trace}(E) = \sum_i e_{ii} = \sum_i \lambda_i \quad \text{et} \quad \text{trace}(E^2) = \sum_{i,j} e_{ij}^2 = \sum_i \lambda_i^2$$

Dans la section 3.4, on profitera de ces données pour proposer des bornes inférieures pour $\hat{\alpha}$ et des minorantes pour la fonction G . Avant cela, on introduit quelques inégalités utiles concernant un ensemble de nombres réels positifs dont on connaît la somme et la somme des carrés.

Le résultat suivant est causé par H. Wolkowicz et al. [46], voir aussi J. P. Crouzeix et al. [12] pour des résultats supplémentaires.

Proposition 10 [46]

$$\begin{aligned}\bar{x} - \sigma_x \sqrt{n-1} &\leq \min_i x_i \leq \bar{x} - \frac{\sigma_x}{\sqrt{n-1}} \\ \bar{x} + \frac{\sigma_x}{\sqrt{n-1}} &\leq \max_i x_i \leq \bar{x} + \sigma_x \sqrt{n-1}\end{aligned}$$

Rappelons que, B. Merikhi et al. [11] a proposé quelques inégalités utiles liées au maximum et au minimum de $x_i > 0$ pour tout $i = 1, \dots, n$.

$$n \ln(\bar{x} - \sigma_x \sqrt{n-1}) \leq A \leq \sum_{i=1}^n \ln(x_i) \leq B \leq n \ln(\bar{x}) \quad (7)$$

avec

$$\begin{aligned}A &= (n-1) \ln\left(\bar{x} + \frac{\sigma_x}{\sqrt{n-1}}\right) + \ln(\bar{x} - \sigma_x \sqrt{n-1}) \\ B &= \ln(\bar{x} + \sigma_x \sqrt{n-1}) + (n-1) \ln\left(\bar{x} - \frac{\sigma_x}{\sqrt{n-1}}\right)\end{aligned}$$

Tels que \bar{x} et σ_x sont respectivement, la moyenne et l'écart type d'une série statistique $\{x_1, x_2, \dots, x_n\}$ de n nombres réels. Ces quantités sont définies comme suit :

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad \text{et} \quad \sigma_x^2 = \frac{1}{n} \sum_{i=1}^n x_i^2 - \bar{x}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$$

Le calcul de pas de déplacement par des méthodes classiques de recherche linéaire n'est pas souhaitable et en général impossible.

Sur la base de cette proposition, nous donnons dans la section suivante, une nouvelle notion des fonctions minorantes pour G , qui offrent des pas de déplacement à chaque itération avec une technique simple.

L'efficacité de l'un à l'autre peut être traduite par des tests numériques que nous présenterons à la fin de ce travail.

3.4 Calcul de pas de déplacement

Revenons aux équations (3.8) et (3.9), notons par $\bar{\lambda}$ et σ_λ la moyenne et l'écart type des λ_i et par $\|\lambda\|$ la norme euclidienne du vecteur λ . Alors :

$$\|\lambda\|^2 = n(\bar{\lambda}^2 + \sigma_\lambda^2) = G''(0) = -G'(0)$$

et

$$G(\alpha) = n\bar{\lambda}\alpha - \|\lambda\|^2\alpha - \sum_{i=1}^n \ln(1 + \alpha\lambda_i) \quad (8)$$

Notre problème consiste à chercher $\bar{\alpha} \in]0, \hat{\alpha}[$ donnant une décroissance significative à la fonction convexe G . Nous avons souligné que le choix le plus naturel, $\bar{\alpha} = \alpha_{opt}$ où $G'(\alpha_{opt}) = 0$, présente des complications numériques. On peut en revanche trouver approximativement $\bar{\alpha}$, mais cette procédure nécessite également plusieurs évaluations de G et G' . Toutefois, si on utilise une recherche linéaire, il convient de connaître la borne supérieure $\hat{\alpha}$ du domaine de G , qui est numériquement difficile à obtenir. On prendra une borne inférieure de $\hat{\alpha}$ fournie par la proposition 10.

$$\hat{\alpha} = \sup[\alpha : 1 + \alpha\beta_1 > 0]$$

et

$$\beta_1 = \bar{\lambda} - \frac{\sigma_\lambda}{\sqrt{n-1}}$$

3.4.1 Première fonction minorante

Cette stratégie, consiste à minimiser des approximations minorantes \check{G} de G au lieu de minimiser G sur $[0, \hat{\alpha})$. Pour être efficace, cette approximation doit être simple et suffisamment proche de G . Dans notre cas on exige

$$0 = \check{G}(0), \quad \|\lambda\|^2 = \check{G}''(0) = -\check{G}'(0)$$

Donc, pour tout $x_i = 1 + \alpha\lambda_i$, $i = 1, \dots, m$, on a $\bar{x} = 1 + \alpha\bar{\lambda}$ et $\sigma_x = \alpha\sigma_\lambda$.

En appliquant les inégalités (7), on obtient

$$\sum_{i=1}^n \ln(1 + \lambda_i \alpha) \leq (n-1) \ln(1 + \beta_1 \alpha) + \ln(1 + \gamma_1 \alpha)$$

avec

$$\beta_1 = \bar{\lambda} - \frac{\sigma_\lambda}{\sqrt{n-1}} \text{ et } \gamma_1 = \bar{\lambda} + \sigma_\lambda \sqrt{n-1}.$$

alors

$$-\sum_{i=1}^n \ln(1 + \lambda_i \alpha) - \|\lambda\|^2 \alpha \geq -(n-1) \ln(1 + \beta_1 \alpha) - \ln(1 + \gamma_1 \alpha) - \|\lambda\|^2 \alpha$$

et

$$n\bar{\lambda}\alpha - \|\lambda\|^2 \alpha - \sum_{i=1}^n \ln(1 + \lambda_i \alpha) \geq n\bar{\lambda}\alpha - \|\lambda\|^2 \alpha - (n-1) \ln(1 + \beta_1 \alpha) - \ln(1 + \gamma_1 \alpha)$$

Les logarithmes sont bien définis lorsque $\alpha < \hat{\alpha}$ avec

$$\hat{\alpha}_1 = \begin{cases} -\frac{1}{\beta_1} & \text{si } \beta_1 < 0 \\ +\infty & \text{sinon} \end{cases}$$

Par substitution de l'inégalité du précédent dans (8), on en déduit la fonction de minoration suivante :

$$\check{G}_1(\alpha) \leq G(\alpha) \text{ pour tout } \alpha \in [0, \hat{\alpha}_1[$$

tel que :

$$\check{G}_1(\alpha) = \delta_1 \alpha - (n-1) \ln(1 + \beta_1 \alpha) - \ln(1 + \gamma_1 \alpha)$$

avec

$$\delta_1 = n\bar{\lambda} - \|\lambda\|^2$$

\check{G}_1 vérifie les propriétés suivantes :

$$\check{G}_1''(0) = -\check{G}_1'(0) = \text{trace}(E^2) \text{ et } \check{G}_1(0) = 0$$

En outre

$$\check{G}_1(\alpha) < 0, \forall \alpha \in [0, \hat{\alpha}_1[$$

\check{G}_1 est convexe et admet un minimum unique sur $[0, \hat{\alpha}_1[$, qui est obtenu en résolvant l'équation

$$\check{G}_1'(\alpha) = 0$$

Chercher les racines α revient à résoudre une équation du deuxième degré de la forme

$$\alpha^2 + b\alpha + c = 0$$

avec

$$b = \frac{1}{2} \left(\frac{n}{\delta_1} - \frac{1}{\beta_1} - \frac{1}{\gamma_1} \right) \text{ et } c = \frac{-\|\lambda\|^2}{\beta_1 \gamma_1 \delta_1}$$

On obtient

$$\bar{\alpha}_{1i} = b \pm \sqrt{b^2 - c}, \quad i = 1, 2$$

On prend la racine positive.

3.4.2 Deuxième fonction minorante

On peut également penser à d'autres fonctions plus simples que \check{G}_1 , qui implique un seul logarithme.

Pour cela, nous considérons les fonctions du type suivant :

$$\check{G}(\alpha) = \check{\delta}\alpha - \check{\gamma} \ln(1 + \check{\beta}\alpha), \quad \alpha \in [0, \check{\alpha}[$$

Les logarithmes sont bien définis sur $\alpha \in [0, \check{\alpha}[$, avec

$$\check{\alpha} = \sup[\alpha : 1 + \alpha\check{\beta} > 0]$$

Alors, on a la fonction de minoration suivante :

$$\check{G}_2(\alpha) \leq G(\alpha) \text{ pour tout } \alpha \in [0, \hat{\alpha}_1[$$

tel que :

$$\begin{aligned} \check{G}_2(\alpha) &= \delta_2\alpha - \gamma_2 \ln(1 + \beta_1\alpha) \\ \check{G}_2(\alpha) &= \left(\frac{\|\lambda\|^2}{\beta_1} - \|\lambda\|^2 \right) \alpha - \frac{\|\lambda\|^2}{\beta_1^2} \ln(1 + \beta_1\alpha) \end{aligned}$$

avec

$$\beta_1 = \bar{\lambda} - \frac{\sigma_\lambda}{\sqrt{n-1}}, \quad \delta_2 = \gamma_2\beta_1 - \|\lambda\|^2$$

et on prend

$$\gamma_2 = \frac{\|\lambda\|^2}{\beta_1^2}$$

qui vérifie la condition suivante :

$$\|\lambda\|^2 = \gamma_2\beta_1^2 = \gamma_2\beta_1 - \delta_2 \tag{9}$$

\check{G}_2 vérifie les propriétés suivantes :

$$\check{G}_2''(0) = -\check{G}_2'(0) = \text{trace}(E^2) \text{ et } \check{G}_2(0) = 0$$

En outre

$$\check{G}_2(\alpha) < 0, \quad \forall \alpha \in [0, \hat{\alpha}_1[$$

\check{G}_2 est convexe et admet un minimum unique sur $[0, \hat{\alpha}_1[$, qui est obtenu en résolvant

l'équation

$$\check{G}'_2(\alpha) = 0$$

puis on obtient

$$\bar{\alpha}_2 = \frac{\gamma_2}{\delta_2} - \frac{1}{\beta_1}$$

3.4.3 Troisième fonction minorante

Une autre fonction minorante plus simple que \check{G}_1 peut être extraite de l'inégalité connue suivante :

$$\left(\|\lambda\| - \sum_{i=1}^n \lambda_i \right) \alpha - \ln(1 + \alpha \|\lambda\|) + \sum_{i=1}^n \ln(1 + \alpha \lambda_i) \leq 0$$

Par remplacement de l'inégalité précédent dans (8), on obtient

$$\check{G}_3(\alpha) \leq G(\alpha)$$

donc

$$\check{G}_3(\alpha) = \delta_3 \alpha - \ln(1 + \beta_2 \alpha)$$

avec

$$\delta_3 = -\|\lambda\|(\|\lambda\| - 1) \text{ et } \beta_2 = \|\lambda\|$$

\check{G}_3 vérifie les propriétés suivantes :

$$\check{G}_3''(0) = -\check{G}_3'(0) = \text{trace}(E^2) \text{ et } \check{G}_3(0) = 0$$

En outre

$$\check{G}_3(\alpha) < 0$$

\check{G}_3 est convexe et admet un minimum unique, qui est obtenu en résolvant l'équation

$$\check{G}'_3(\alpha) = 0$$

qui donne

$$\bar{\alpha}_3 = -(\|\lambda\| - 1)^{-1}$$

Proposition 11 G_i , $i = 1, \dots, 3$, strictement convexe sur $\alpha \in [0, \alpha^*[$, avec $\alpha^* = \min(\hat{\alpha}, \hat{\alpha}_1)$. Donc, on a

$$\check{G}_3(\alpha) \leq \check{G}_2(\alpha) \leq \check{G}_1(\alpha) \leq G(\alpha), \quad \forall \alpha \in [0, \alpha^*[$$

Preuve :

La première inégalité est évidente. L'inégalité $G(\alpha) \geq \check{G}_1(\alpha)$ est directement donnée par l'inégalité (7).

Maintenant, prouvez que

$$\check{G}_2(\alpha) \leq \check{G}_1(\alpha)$$

Considérons la fonction

$$v(\alpha) = \check{G}_2(\alpha) - \check{G}_1(\alpha)$$

Puisque $\beta_1 = \beta_1$ et $\beta_1 \leq \gamma_1$, on a pour tout $\alpha > 0$

$$v''(\alpha) = \frac{\gamma_2 \beta_1^2 - (n-1) \beta_1^2}{(1 + \beta_1 \alpha)^2} - \frac{\gamma_1^2}{(1 + \gamma_1 \alpha)^2} = \frac{\gamma_1^2}{(1 + \gamma_1 \alpha)^2} - \frac{\gamma_1^2}{(1 + \beta_1 \alpha)^2} \leq 0$$

Parce que $v''(\alpha) \leq 0$ et $v'(0) = 0$, on a $v'(\alpha) \leq 0$ pour tout $\alpha \in [0, \hat{\alpha}[$. Car $v'(\alpha) \leq 0$ et $v(0) = 0$, on a aussi $v(\alpha) \leq 0$ pour tout $\alpha \in [0, \alpha^*[$. Donc

$$\check{G}_2(\alpha) \leq \check{G}_1(\alpha), \quad \forall [0, \alpha^*[$$

Ensuite, on prouve que

$$\check{G}_3(\alpha) \leq \check{G}_2(\alpha)$$

De même, considérons la fonction

$$w(\alpha) = \check{G}_3(\alpha) - \check{G}_2(\alpha)$$

Donc, $w(0) = w'(0) = 0$ et

$$\begin{aligned} w''(\alpha) &= -\frac{\beta_2^2}{(1 + \beta_2\alpha)^2} + \frac{\gamma_2\beta_1^2}{(1 + \beta_1\alpha)^2} \\ &= \|\lambda\|^2 \left(\frac{1}{(1 + \beta_1\alpha)^2} - \frac{1}{(1 + \beta_2\alpha)^2} \right) \leq 0 \end{aligned}$$

Parce que $w''(\alpha) \leq 0$ et $w(0) = w'(0) = 0$, on a $w(\alpha) \leq 0$ pour tout $\alpha \in [0, \alpha^*]$. Alors $\check{G}_3(\alpha) \leq \check{G}_2(\alpha)$ sur $\alpha \in [0, \alpha^*]$. La preuve est complète. \square

Rappelons que les fonctions \check{G}_i atteignent leur minimum sur un point unique $\bar{\alpha}_i$ qui sont les racines de $\check{G}'_i(\alpha) = 0$. Ainsi, les trois racines sont explicitement calculées, pour $i = 1, \dots, 3$. Donc, on a

$$\begin{aligned} \bar{\alpha}_1 &= b - \sqrt{b^2 - c} \\ \bar{\alpha}_2 &= \frac{\gamma_2}{\delta_2} - \frac{1}{\beta_1} \\ \bar{\alpha}_3 &= -(\|\lambda\| - 1)^{-1} \end{aligned}$$

avec $b = \frac{1}{2} \left(\frac{n}{\delta_1} - \frac{1}{\beta_1} - \frac{1}{\gamma_1} \right)$ et $c = \frac{-\|\lambda\|^2}{\beta_1\gamma_1\delta_1}$. Ainsi, les trois valeurs $\bar{\alpha}_i$, $i = 1, \dots, 3$ sont explicitement calculées.

Le calcul de $\bar{\alpha}$ est approximativement calculé par une procédure dichotomique.

Si $\bar{\alpha} \notin [0, \alpha^*]$ alors, prendre $\bar{\alpha} = \alpha^* - \varepsilon$, $a = 0$ et $b = \alpha^*$.

Tant que $|b - a| > 10^{-4}$ faire

Si $G'(\bar{\alpha}) < 0$ alors

$$a = 0, \quad b = \bar{\alpha} \text{ et } \bar{\alpha} = \frac{a+b}{2}$$

Sinon

$$a = 0, \quad b = b \text{ et } \bar{\alpha} = \bar{\alpha}$$

Fin tant que.

Ce calcul garantit une meilleure approximation de la minimisation de $G(\alpha)$ tout en restant dans le domaine de G .

3.5 Description de l'algorithme

Dans cette section, nous présentons l'algorithme de notre approche pour obtenir une solution optimale \bar{x} de notre problème.

Début d'algorithme

Initialisation

Nous devons décider de la stratégie de pas de déplacement. $\varepsilon > 0$ est une précision donnée, $\eta > 0$, $\rho > 0$ et $\sigma \in]0, 1[$ sont des paramètres fixes. On démarre avec $x^k \in \hat{X}$ et $k = 0$.

Itération

1. Prendre $B = B(x^k) = \sum_{i=1}^m x_i^k A_i - C$ et L tel que $B = LL^T$.
2. Calculer
$$\left\{ \begin{array}{l} \hat{A}_i(x^k) = [L(x^k)]^{-1} A_i [L^T(x^k)]^{-1} \\ b(x^k) = \text{trace}(\hat{A}_i(x^k)) \\ \Delta_{ij}(x^k) = \text{trace}(\hat{A}_i(x^k) \hat{A}_j(x^k)) \\ H = \eta \Delta(x^k) \end{array} \right.$$
3. Résoudre le système linéaire $Hd = \eta b(x) - b$.
4. Calculer $E = L^{-1} H (L^{-1})^T$, $\text{trace}(E)$ et $\text{trace}(E^2)$.

5. Prendre la nouvelle itération $x^{k+1} = x^k + \bar{\alpha}d$. Telle que $\bar{\alpha}$ est obtenue par l'utilisation de la stratégie de pas de déplacement de \check{G}_i , $i = 1, \dots, 3$.
6. Si $n\eta > \varepsilon$, alors $x^k = x^{k+1}$, $\eta = \sigma\eta$ et aller en (1).
7. Si $|b^T x^{k+1} - b^T x^k| > n\rho\eta$, alors $x^k = x^{k+1}$ et aller en (1).
8. Prendre $k = k + 1$.
9. **Stop** : x^{k+1} est une solution optimale approchée de notre problème.

Fin d'algorithme

On sait de ce qui précède, que la solution optimale de $(SDP)_\eta$ est une approximation de la solution de (1), plus η est proche de zéro plus l'approximation est bonne. Malheureusement quand η s'approche de zéro, le problème $(SDP)_\eta$ devient mal conditionné. C'est la raison pour laquelle on utilisera au début de l'itération des valeurs de η pas très proches de zéro, vérifiant le test $n\eta < \varepsilon$. On peut expliquer la mise à jour de η comme suit : si $x(\eta)$ est une solution exacte de $(SDP)_\eta$, alors

$$b^T x(\eta) \in [m(0), m(0) + n\eta]$$

Il est donc inutile de continuer le calcul des itérés quand $|b^T x - b^T \bar{x}| \leq \rho n\eta$. Pour ρ , on peut considérer les valeurs 1, 2, $\sigma = 0.125$.

Le pas de déplacement $\bar{\alpha}$ sera déterminé par la recherche linéaire classique du type Armijo-Goldstein-Price ou par l'une des trois précédentes stratégies.

Dans la section suivante, nous présentons des tests numériques comparatifs pour prouver l'efficacité de notre approche par rapport à la méthode de la recherche linéaire classique.

3.6 Tests numériques

Les exemples suivants sont tirés de la littérature voir par exemple [7, 11, 20] et mis en œuvre dans MATLAB R2013a sur Pentium (R). Nous avons pris $\varepsilon = 1.0e-006$, $\sigma = 0.125$

et deux valeurs de ρ , $\rho = 1$ ou $\rho = 2$.

Dans le tableau des résultats, (exp (m, n)) représente la taille de l'exemple, (Itrat) représente le nombre d'itérations nécessaires pour obtenir une solution optimale, (Temp) représente le temps de calcul en secondes (s), (LS) représente la recherche de la ligne classique de la méthode Armijo-Goldstein et (St i) représente la stratégie qui utilise les fonctions de minoration, \check{G}_i , avec $i = 1, \dots, 3$.

Rappelons que le problème considéré est

$$\begin{cases} \min b^T x \\ \sum_{i=1}^m x_i A_i - C \in S_n^+ \\ x \in \mathbb{R}^m \end{cases}$$

3.6.1 Exemples à taille fixe

Dans les exemples suivants, $diag(x)$ est la matrice diagonale $n \times n$ avec les composantes de x entrées diagonales.

Exemple 1 :

$$C = diag(-5, -8, -8, -5),$$

$A_4 = diag(1, 1, 1, 1)$ et les matrices A_k , $k = 1, \dots, 3$, sont définis comme suit :

$$A_k[i, j] = \begin{cases} 1 & \text{si } i = j = k \quad \text{ou } i = j = k + 1 \\ -1 & \text{si } i = k, j = k + 1 \quad \text{ou } i = k + 1, j = k \\ 0 & \text{sinon} \end{cases}$$

$$b = (1, 1, 1, 2)^T.$$

Nous commençons par un point initial $x^0 = (1.5, 1.5, 1.5, 1.5)^T$.

Exemple 2 :

$$C = diag(-4, -2, -2, 0, 0, 0),$$

$$A_1 = diag(1, -1, 1, 1, 0, 0),$$

$$A_2 = diag(1, 1, 1, 0, 1, 0),$$

$$A_3 = \text{diag}(2, 2, 1, 0, 0, 1),$$

$$b = (6, 2, 4)^T.$$

Nous commençons par un point initial $x^0 = (0.5, 1, 1)^T$.

Exemple 3 :

$$C = \text{diag}(-4, -5, 0, 0, 0)^T,$$

$$A_1 = \text{diag}(2, 1, 1, 0, 0)^T,$$

$$A_2 = \text{diag}(1, 2, 0, 1, 0)^T,$$

$$A_3 = \text{diag}(0, 1, 0, 0, 1)^T,$$

$$b = (8, 7, 3)^T.$$

Nous commençons par un point initial $x^0 = (-2, -1, -2)^T$.

Exemple 4 :

$$C = \text{diag}(-4, -5, -1, -3, 5, -8, 0, 0, 0, 0, 0, 0)^T,$$

$$A_1 = \text{diag}(1, 0, -4, 3, 1, 1, 1, 0, 0, 0, 0, 0)^T,$$

$$A_2 = \text{diag}(5, 3, 1, 0, -1, 3, 0, 1, 0, 0, 0, 0)^T,$$

$$A_3 = \text{diag}(4, 5, 3, 3, 4, 1, 0, 0, 1, 0, 0, 0)^T,$$

$$A_4 = \text{diag}(0, -1, 0, 2, 1, -5, 0, 0, 0, 1, 0, 0)^T,$$

$$A_5 = \text{diag}(-2, 1, 1, 1, 2, 2, 0, 0, 0, 0, 1, 0)^T,$$

$$A_6 = \text{diag}(2, -3, 2, -1, 4, 5, 0, 0, 0, 0, 0, 1)^T.$$

$$b = (1, 4, 4, 5, 7, 5)^T.$$

Nous commençons par un point initial $x^0 = (\frac{1}{16}, \frac{1}{16}, 1, \frac{1}{16}, \frac{1}{8}, \frac{1}{16})^T$.

Le tableau suivant reprend les résultats obtenus :

exp (m, n)	St 1		St 2		St3		LS	
	Itrat	Temp	Itrat	Temp	Itrat	Temp	Itrat	Temp
exp 1(4, 4)	3	0.012	3	0.014	4	0.19	5	0.25
exp 2(3, 6)	1	0.0016	1	0.0022	2	0.0065	7	0.36
exp 3(3, 5)	1	0.0027	1	0.0056	3	0.081	10	0.87
exp 4(6, 12)	1	0.0037	1	0.0078	1	0.0081	10	0.87

3.6.2 Exemples à taille variable

Exemple 1 : (Exemple Cube)

$n = 2m$, C est la matrice d'identité de rang n , $b = (2, \dots, 2)^T \in \mathbb{R}^m$, $a \in \mathbb{R}$ et les entrées du $n \times n$ matrices A_k , $k = 1, \dots, m$, sont données par :

$$A_k[i, j] = \begin{cases} 1 & \text{si } i = j = k & \text{ou } i = j = k + m \\ a^2 & \text{si } i = j = k + 1 & \text{ou } i = j = k + m + 1 \\ -a & \text{si } i = k, j = k + 1 & \text{ou } i = k + m, j = k + m + 1 \\ -a & \text{si } i = k + 1, j = k & \text{ou } i = k + m + 1, j = k + m \\ 0 & \text{sinon} \end{cases}$$

Test 1 : $a = 0$ et $C = -I$.

On sait que le vecteur $x = (1, \dots, 1)^T \in \mathbb{R}^m$ est la solution optimale.

Nous commençons par un point initial $x^0 = (1.5, \dots, 1.5)^T \in \mathbb{R}^m$.

Le tableau suivant reprend les résultats obtenus :

Taille(m, n)	St 1		St 2		St 3		LS	
	Itrat	Temp	Itrat	Temp	Itrat	Temp	Itrat	Temp
(50, 100)	1	16	1	17	1	39	dvg	
(100, 200)	1	89	1	104	1	192	dvg	
(200, 400)	1	473	1	545	1	594	dvg	

dvg signifie que l'algorithme ne se termine pas dans un temps fini.

Les résultats de ces tests montrent que LS ne rivalise pas avec St 1, St 2 et St 3. Dans les prochains tests, nous continuons uniquement avec St 1, St 2 et St 3.

Test 2 : $a = 1$ et $C = -I$.

Nous commençons par un point initial $x^0 = (1.5, 1.5, \dots, 1.5)^T \in \mathbb{R}^m$.

Le tableau suivant reprend les résultats obtenus :

Taille(m, n)	St 1		St 2		St 3	
	Itrat	Temp	Itrat	Time	Itrat	Temp
(50, 100)	1	19	1	19	2	51
(100, 200)	1	91	1	102	2	217
(200, 400)	1	235	1	490	2	612

Test 3 : $a = 2$ et $C = -I$.

Nous commençons par un point initial $x^0 = (0, \frac{3}{2}, \dots, \frac{3}{2})^T \in \mathbb{R}^m$.

Le tableau suivant reprend les résultats obtenus :

Taille(m, n)	St 1		St 2		St 3	
	Itrat	Temp	Itrat	Temp	Itrat	Temp
(50, 100)	1	16	1	18	2	24
(100, 200)	1	72	1	114	2	325
(200, 400)	1	258	1	435	3	723

Test 4 : $a = 5$ et $C = -2I$.

Nous commençons par un point initial $x^0 = (0, 0, \dots, 0)^T \in \mathbb{R}^m$.

Le tableau suivant reprend les résultats obtenus :

Taille (m, n)	St 1		St 2		St 3	
	Itrat	Temp	Itrat	Temp	Itrat	Temp
(50, 100)	1	19	1	23	2	86
(100, 200)	1	83	1	154	3	425
(200, 400)	1	318	1	525	3	793

Exemple 2 :

$C = -I$, $b_K = 1$ et A_k , $k = 1, \dots, m$, sont définis comme suit :

$$A_k[i, j] = \begin{cases} 1 & \text{si } i = j = k \\ 1 & \text{si } i = j \text{ et } i = k + m \\ 0 & \text{sinon} \end{cases}$$

Nous commençons par un point initial $x^0 = (1.5, \dots, 1.5, 0.5)^T$.

Le tableau suivant reprend les résultats obtenus :

Taille(m, n)	St 1		St 2		St 3	
	Itrat	Temp	Itrat	Temp	Itrat	Temp
(50, 100)	1	17	1	17	2	25
(100, 200)	1	35	1	89	2	354
(200, 400)	1	223	1	234	2	565

3.6.3 Commentaires

À travers les tests numériques qu'on a réalisés sur des exemples de différentes dimensions, on remarque que les résultats montrent l'importance de notre approche barrière, exprimée par une réduction significative du nombre d'itérations et du temps de calcul. En général, le choix de St 1 et St 2 est meilleure que celui de St 3 en termes de temps de calcul.

Conclusion

Dans notre travail, on est arrivé aux deux résultats suivants :

1. La difficulté de la présence du $\ln(\det)$ dans la fonction perturbé a été surmonté, et ce, en remplaçant la somme de valeurs propre par la trace de E et la somme des carrées par la trace de E^2 .
2. Réduire la complexité de notre algorithme et ce introduisant la nouvelle technique des fonctions minorantes, les tests numériques effectués montrent clairement l'efficacité de notre approche.

L'aspect numérique peut être poussé à un niveau de performance très appréciable pour la pratique.

La technique des fonctions minorantes pour déterminer le pas de déplacement dans la direction de descente est une alternative très fiable qui est confirmée comme la technique de choix pour la programmation semi-définie (SDP) et d'autres classes de problèmes d'optimisation : programmation linéaire (LP) et programmation non linéaire (PNL)...

Bibliographie

- [1] F. Alizadeh, Combinatorial Optimization with Interior-Point Methods and Semi-Definite Matrices, Ph.D. Thesis, Computer Science Department, University of Minnesota, Minneapolis, MN, 1991.
- [2] F. Alizadeh, Interior point methods in semidefinite programming with applications to combinatorial optimization, *SIAM J. Optim.* 5 (1995) 13–51.
- [3] F. Alizadeh, J. P. Haberly, and M. L. Overton, Primal-dual interior-point methods for semidefinite programming, convergence rates, stability and numerical results, *SIAM Journal on Optimization* 8 (1998), 746–768.
- [4] M. S. Babynin, V.G. Zhadan, A primal interior point method for linear semidefinite programming problem, *Comput. Math. Phy.* 48 (2008) 1746–1767.
- [5] Y. Q. Bai, C. Roos, A primal-dual interior point method based on a new kernel function with linear growth rate, *Proceedings of the 9th Australian Optimization Day*, Perth, Australia, (2002).
- [6] Y. Q. Bai, M. El. Ghami, C. Roos, A new efficient large-update primal-dual interior point method based on a finite barrier. *SIAM. J. Optim.* 13 (3), 766–782 (2003).
- [7] D. Benterki, J.P. Crouzeix, B. Merikhi, A numerical feasible interior point method for linear semidefinite programs, *RAIRO Oper. Res.* 41 (2007) 49–59.
- [8] D. Benterki, J.P. Crouzeix, B. Merikhi, A numerical implementation of an interior point method for semidefinite programming, *Pesquisa Operacional Journal* 23 (2003), 49–59.

- [9] M. Bouafia, D. Benterki, A. Yassine, An efficient primal-dual interior point method for linear programming problems based on a new kernel function with a trigonometric barrier term, *J. Optim. Theory Appl.* 170 (2) (2016) 528—545.
- [10] B.K. Choi, G.M. Lee, On complexity analysis of the primal-dual interior-point method for semidefinite optimization problem based on a proximity function. *Nonlinear Anal.* 71 2628–2640 (2009).
- [11] J. P. Crouzeix, B. Merikhi, A logarithm barrier method for semidefinite programming, *RAIRO-Oper. Res.* 42 (2008), 123–139.
- [12] J. P. Crouzeix, A. Seeger, New bounds for the extreme values of a finite sample of real numbers, *Journal of Mathematical Analysis and Applications* 197 (1996), 411–426.
- [13] A. Dehghani, J.-L. Goffin & D. Orban, A primal–dual regularized interior-point method for semidefinite programming, *Optimization Methods and Software* Volume 32, - Issue 1 (2017).
- [14] I. I. Dikin, Iterative solution of problems of linear and quadratic programming, *Doklady Akademii Nauk SSSR.* 174 (1967) 747–748.
- [15] M. El Ghami, I.D. Ivanov, C. Roos, T. Steihaug, A polynomial time algorithm for LO based on generalized logarithmic barrier functions. *Int. J. Appl. Math.* 21, 99–115 (2008).
- [16] M. Halicka, E. De Klerk and C. Roos, On the convergence of the central path in semidefinite optimization, *SIAM J. Optim.* 12 (2002) 1090–1099.
- [17] C. Helmberg and F. Rendl, A Spectral Bundle Method for Semidefinite Programming, *SIAM Journal on Optimization*, 2000, Vol. 10, No. 3 : pp. 673-696 .
- [18] J. Ji, F. A. Potra, R. Sheng, On the local convergence of a predictor-corrector method for semidefinite programming, *SIAM Journal on Optimization*, 10 (1999), 195–210.
- [19] N. K. Karmarkar, A new polynomial time algorithm for linear programming, *Combinatorica.* 4 (1984) 373–395.

- [20] S. Kettab, D. Benterki, A relaxed logarithmic barrier method for semidefinite programming, *RAIRO-Oper. Res.* 49 (2015) 555–568.
- [21] T. Kim-Chuan, Some new search directions for primal-dual interior point methods in semidefinite programming, *SIAM Journal on Optimization*, (2000).
- [22] E. Klerk, *Aspects of Semidefinite Programming : Interior Point Algorithms and Selected Applications*, Kluwer Academic Publishers, Dordrecht, The Netherlands (2002)
- [23] M. Kojima, S. Shindoh, S. Hara, Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices, *SIAM, J. Optim.* 7 (1997) 86–125.
- [24] M. H. Koulaei, T. Terlaky, On the extension of a Mehrotra-type algorithm for semidefinite optimization, Technical Report 2007/4, Advanced optimization Lab. Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada.
- [25] I. Krim, *Optimisation sous contraintes de semi définie positivité*, Thèse de Magister, Université d’Oran, Algérie, 2014.
- [26] C. Liu, H. Liu & X. Yong, New complexity analysis of a Mehrotra-type predictor–corrector algorithm for semidefinite programming, *Optimization Methods and Software*, Published online : 17 May (2012).
- [27] C. Liu, H. Liu, A new second-order corrector interior-point algorithm for semidefinite programming, *Math Meth Oper Res.* 75 (2012) 165–183.
- [28] B. Merikhi, *Extension de quelques méthodes de points intérieurs pour la programmation semi-définie*, Thèse de Doctorat, Département de Mathématiques, Université Ferhat Abbas, Sétif-1, 2007.
- [29] B. Mohar, S. Poljak, D. F. Shanno, Interior point methods for linear programming : Computational state of the art, *ORSA J. Comput.* 6 (1994) 1–14.

- [30] R. D. C. Monteiro, I. Adler, M. G. C. Resende, A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension, *Mathematics of Operations Research*. 15 (1990) 191–214.
- [31] R. D. C. Monteiro, Primal-dual path-following algorithms for semidefinite programming, *SIAM Journal on Optimization*, 7 (1997), 663–678.
- [32] R.D.C. Monteiro, Polynomial convergence of primal-dual algorithms for semidefinite programming based on the Monteiro and Zhang family of directions, *SIAM J. Optim.* 8 (1998) 797–812.
- [33] R. D. C. Monteiro and T. Tsuchiya, Polynomial convergence of a new family of primal-dual algorithms for semidefinite programming, *SIAM J. Optim.* 9 (1999) 551–577.
- [34] Y. E. Nesterov, A. Nemirovski, Optimization over positive semidefinite matrices : Mathematical background and user’s manual, Technical report, Central economic and mathematical institute, USSR academy of science, Moscow, USSR, 1990.
- [35] Y. E. Nesterov and A. S. Nemirovskii, Interior Point Polynomial Algorithms in Convex Programming : Theory and Applications, SIAM, Philadelphia, PA, (1994).
- [36] A.M. Nunez and R.M. Freund, Condition-measure bounds on the behavior of the central trajectory of semidefinite program. *SIAM J. Optim.*12 (2001) 818–836.
- [37] J. Peng, C. Roos, T. Terlaky Self-regular functions and new search directions for linear and semidefinite optimization, *Math. Program.*, 93 (2002), pp. 129-171.
- [38] R. T. Rockafellar, *Convex analysis*, Princeton University Press, New Jerzy, 1970.
- [39] C. Roos, T. Terlaky, J.P. Vial, *Theory and Algorithms for Linear Optimization : An Interior Point Approach*, W. John & Sons, 1997.
- [40] M. J. Todd, On search directions in interior-point methods for semidefinite programming, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, New York (1997).

- [41] M. J. Todd, K. C. Toh and R. H. Tütüncü, On the Nesterov-Todd direction in semidefinite programming, *SIAM J. Optim.* 8 (1998) 769–796.
- [42] I. Touil, D. Benterki b, A Yassine, A feasible primal–dual interior point method for linear semidefinite programming, *Journal of Computational and Applied Mathematics* 312 (2017) 216–230.
- [43] L. Vandenberghe, S. Boyd, Primal-dual potential reduction method for problems involving matrix inequalities, *Math. Programming, Series B*, 69 (1995) 205-236.
- [44] L. Vandenberghe, S. Boyd, Applications of semidefinite programming, *Applied Numerical Mathematics*, Volume 29, Issue 3, March 1999, Pages 283-299
- [45] G. Q. Wang, Y. Q. Bai, C. Roos, Primal-Dual Interior-Point Algorithms for Semidefinite Optimization Based on a Simple Kernel Function, *Journal of Mathematical Modelling and Algorithms* 4 (2005) 409–433.
- [46] H. Wolkowicz, G. P. H. Styan, Bounds for eigenvalues using traces, *Linear Algebra and Appl.* 29 (1980), 471–506.
- [47] Y. Zhang, On extending some primal–dual interior-point algorithms from linear programming to semidefinite programming, *SIAM J. Optim.* 8 (1998) 365–386.

ملخص

تتناول هذه الأطروحة دراسة مسألة البرمجة نصف المعرفة الخطية مع مشكله البرمجة شبه المعرفة وتحديدًا حلها بطريقة النقاط الداخلية. إن الحساب الاقتصادي لخطوة الانتقال يلعب دورًا هامًا في سلوك الخوارزمية ولهذا نتناول هذه الأطروحة دراسة مسألة البرمجة نصف المعرفة الخطية وتحديدًا طريقة الحاجز اللوغاريتمي حيث تطرقنا بالبحث إلى الجانب النظري والخوارزمي. وكان لإدراج الدوال الحدية لحساب الخطوات نتائج فعالة في تقليل كلفة العمليات الحسابية مقارنة بطرق البحث الخطية الكلاسيكية.

التجارب العددية التي قمنا بها مشجعه وتبين فعالية طريقتنا المقترحة.

الكلمات الرئيسية: البرمجة نصف المعرفة، طريقة النقاط الداخلية، طريقة الحاجز اللوغاريتمي، طرق البحث الخطي.

Résumé

Dans cette thèse, on traite le problème de programmation semi-définie (SDP). En particulier, on s'intéresse aux performances d'une méthode de points intérieurs qui le résout. En effet, le calcul économique du pas de déplacement joue un rôle important dans le comportement de l'algorithme. Dans ce sens, Nous proposons dans cette thèse une approche, barrière logarithmique dans laquelle, on introduit une procédure originale pour le calcul du pas de déplacement basée sur les fonctions minorantes : On obtient une approximation explicite entraînant une décroissance significative de l'objectif, de plus elle est économique et robuste, contrairement aux méthodes classiques de recherche linéaire.

Les expérimentations numériques que nous avons effectués sont encourageantes et mettent en évidence les performances de notre approche.

Mots clés : Programmation semi-définie, méthode de points intérieurs, méthode barrière logarithmique, méthodes de recherche linéaire.

Abstract

In this thesis, we deal with the linear semidefinite programming problem. In particular, we are interested in the performance of a method of interior points, which solves it. Indeed, the economic calculation of the displacement pitch plays an important role in the behavior of the algorithm. In this sense, we propose in this thesis an approach, logarithmic barrier in which, it introduces an original procedure for the calculation of the no displacement based on the minorants functions: one obtains an explicit approximation causing a significant decay of the objective, the more it is economic and robust, unlike conventional methods of linear search.

The numerical experiments we have carried out are encouraging and highlight the performance of our approach.

Key words: Semidefinite programming, interior point methods, logarithmic barrier methods, line search methods.