

Introduction au Compressed sensing.

Méthodes de points intérieurs¹

Guillaume Lécué²

Résumé

Dans ce chapitre, nous présentons deux algorithmes de points intérieurs pour résoudre des problèmes de programmation linéaires. Les deux algorithmes sont des méthodes de barrière qui s'autorisent à traverser le polytope des contraintes de l'intérieur pour donner une solution approchante.

Le premier algorithme est une méthode dite primale qui choisit les directions de descentes uniquement à partir du problème primal. Le deuxième algorithme utilise le problème dual pour choisir les directions de descente.

La présentation de ces deux algorithmes donne l'occasion de rappeler des outils très utiles pour l'étude et la construction d'algorithmes pour des problèmes d'optimisation : dualité Lagrangienne, conditions KKT, directions de descentes et méthodes de barrière.

1 Introduction

L'algorithme du simplexe se "promène" sur les sommets du polytope des contraintes de proche en proche tout en assurant la décroissance de la fonction objectif. Quand il atteint un sommet pour lequel la fonction objectif ne peut plus être minimisé sur les voisins alors l'algorithme renvoie le sommet solution du problème.

Cet algorithme a d'excellentes performances en pratique ; en général, il tourne en temps linéaire en le nombre de contraintes. Cependant, il existe un exemple appelé le "twisted cube" ou le cube de Klee-Minty pour lequel il a été démontré que l'algorithme du simplexe nécessite un temps exponentiel en le nombre de contraintes pour se terminer. Pour le cas du cube de Klee-Minty, l'algorithme du simplexe visite tous les sommets de ce cube qui en a 2^m en dimension m où m est le nombre de contraintes du problème de programmation linéaire.

Le comportement en temps exponentiel de l'algorithme du simplexe a longtemps perturbé les chercheurs car il contredisait son comportement en pratique qui est lui très bon. Beaucoup de recherches ont alors tenté de trouver une manière de choisir le pivot à chaque itération de telle sorte que l'algorithme en résultant ait un comportement en temps linéaire. Ce problème est toujours ouvert de nos jours (cf. Problème 9 de la liste "Mathematical problems for the next century" de Steve Smale). Néanmoins, une explication du l'excellent comportement en pratique de cet algorithme peut être trouvé dans [6]).

Neuf ans après la construction du cube de Klee-Minty, [2] a proposé l'algorithme de l'ellipsoïde qui permet de résoudre en temps polynomial tout problème de programmation linéaire. La clef de cette solution était d'autoriser l'algorithme à évoluer à *l'intérieur du polytope des contraintes* et non seulement à sa surface comme le fait l'algorithme du simplexe. L'algorithme de l'ellipsoïde est le premier exemple d'algorithme de points intérieurs.

1. Notes du cours du 1 mars 2016

2. CNRS, CREST, ENSAE. Bureau E31, 3 avenue Pierre Larousse. 92 245 Malakof. Email : guillaume.lecue@ensae.fr.

Cependant l'algorithme du simplexe reste beaucoup plus performant en pratique que l'algorithme de l'ellipsoïde. Le problème de trouver un algorithme aussi performant en pratique que l'algorithme du simplexe mais ayant une assurance théorique d'être résoluble en temps polynomial n'a été résolu qu'en 1984 par l'algorithme de projection de [3]. Dans [1], les auteurs ont montré que l'algorithme de projection était en fait un **algorithme de barrière**. C'est cet algorithme que nous présentons dans ce chapitre sous l'angle "moderne" des **méthode de barrière primal**. Puis, nous utiliserons des informations sur le problème dual pour construire un deuxième algorithme de point intérieur de la famille des **méthode de barrière dual-primal**.

Avant de présenter ces deux algorithmes, nous introduisons quelques outils de l'optimisation convexe qui vont nous servir à comprendre les méthodes de points intérieurs.

2 Outils d'optimisations

Dans cette section, nous rappelons quelques outils de la théorie de l'optimisation qui vont nous servir à la construction de deux algorithmes de points intérieurs.

2.1 Dualité Lagrangienne

La dualité Lagrangienne joue un rôle centrale pour les problèmes d'optimisation sous contraintes. On considère donc un tel problème ici :

$$\min_{x \in K} J(x) \text{ où } K = \left\{ x \in \mathbb{R}^n : \begin{array}{ll} c_i(x) \leq 0 & \text{pour } i \in \mathcal{I} \\ e_i(x) = 0 & \text{pour } i \in \mathcal{E} \end{array} \right\} \quad (\text{P})$$

On suppose que K est non vide et que J atteint son minimum sur K .

Définition 2.1. La **fonction de Lagrange** associée au problème d'optimisation sous contraintes (P) est définie par

$$L(x, \alpha, \beta) = J(x) + \sum_{i \in \mathcal{I}} \alpha_i c_i(x) + \sum_{i \in \mathcal{E}} \beta_i e_i(x)$$

où $\alpha_i \geq 0$ et $\beta_i \in \mathbb{R}$. Les vecteurs $\alpha = (\alpha_i)_{i \in \mathcal{I}}$ et $(\beta_i)_{i \in \mathcal{E}}$ sont appelés **multiplicateurs de Lagrange**.

La fonction de Lagrange permet de "transformer" le problème d'optimisation sous contraintes (P) en un problème d'optimisation sous des contraintes plus simples que celles données par K grâce au résultat suivant.

Proposition 2.2. On a :

$$\max_{\alpha \geq 0, \beta} L(x, \alpha, \beta) = \begin{cases} J(x) & \text{si } x \in K \\ +\infty & \text{sinon.} \end{cases}$$

Démonstration. Si $x \in K$ alors $c_i(x) \leq 0$ pour tout $i \in \mathcal{I}$ et $e_i(x) = 0$ pour tout $i \in \mathcal{E}$ et donc pour tout $\alpha \geq 0$ et β

$$\sum_{i \in \mathcal{I}} \alpha_i c_i(x) + \sum_{i \in \mathcal{E}} \beta_i e_i(x) = \sum_{i \in \mathcal{I}} \alpha_i c_i(x) \leq 0$$

qui est atteint en $\alpha = 0$ et n'importe quel β . On a alors $\max_{\alpha \geq 0, \beta} L(x, \alpha, \beta) = J(x)$.

Réciproquement, si $x \notin K$ alors une des conditions de K n'est pas satisfaite par x : soit $c_i(x) > 0$ ou $e_i(x) \neq 0$. Dans les deux cas

$$\max_{\alpha \geq 0, \beta} \sum_{i \in \mathcal{I}} \alpha_i c_i(x) + \sum_{i \in \mathcal{E}} \beta_i e_i(x) = +\infty.$$

Ce qui conclut la preuve. ■

D'après Proposition 2.2, on a

$$\min_{x \in K} J(x) = \min_{x \in \mathbb{R}^n} \max_{\alpha \geq 0, \beta} L(x, \alpha, \beta). \quad (2.1)$$

On remarque que dans la terme de droite le minimum n'est plus contraint : on minimise une fonction sur tout \mathbb{R}^n et non plus sur $K \subset \mathbb{R}^n$.

La dualité de Lagrange consiste à identifier les situations pour lesquelles, on peut intervertir $\min_{x \in \mathbb{R}^n}$ et $\max_{\alpha \geq 0, \beta}$ dans le membre de droite de (2.1). On peut déjà identifier un résultat immédiat qui est vrai sans aucune condition.

Proposition 2.3 (dualité faible). *On a :*

$$\max_{\alpha \geq 0, \beta} \min_{x \in \mathbb{R}^n} L(x, \alpha, \beta) \leq \min_{x \in \mathbb{R}^n} \max_{\alpha \geq 0, \beta} L(x, \alpha, \beta). \quad (2.2)$$

Démonstration. Pour tout $x \in \mathbb{R}^n, \alpha \geq 0$ et β , on a

$$\min_{x \in \mathbb{R}^n} L(x, \alpha, \beta) \leq L(x, \alpha, \beta).$$

Alors, en passant au maximum sur tout $\alpha \geq 0$ et β et en prenant ensuite le minimum sur $x \in \mathbb{R}^n$, on obtient le résultat. ■

Le résultat de la Proposition 2.3 est un résultat de dualité faible puisqu'il est toujours vrai. L'inégalité inverse ne l'est pas toujours et nécessite des hypothèses sur les fonctions J, c_i, e_i . On peut réécrire cette dualité faible en fonction de la fonction duale de Lagrange.

Définition 2.4. *La fonction duale de Lagrange est*

$$D : \alpha \geq 0, \beta \longrightarrow \min_{x \in \mathbb{R}^n} L(x, \alpha, \beta).$$

Le problème duale de Lagrange est le problème d'optimisation

$$\boxed{\max D(\alpha, \beta) \text{ tel que } \alpha \geq 0.} \quad (\text{D})$$

Il est important de noter que le problème d'optimisation qui définit la fonction duale de Lagrange est un problème non-contraint qui peut donc être résolu de manière classique (par exemple, avec une condition de premier ordre $\nabla_x L(x, \alpha, \beta) = 0$ quand L est \mathcal{C}^1 et convexe). De plus pour tout $x \in \mathbb{R}^n$, la fonction $(\alpha, \beta) \rightarrow L(x, \alpha, \beta)$ est linéaire. Alors $D(\alpha, \beta)$ est le minimum d'une famille de fonction concave donc elle est aussi concave. Alors le problème dual de Lagrange (D) est un problème de maximisation d'une fonction concave sur un ensemble convexe. C'est donc un problème d'optimisation convexe qui est en général plus facile à résoudre que le problème (P) qu'on appellera dorénavant **problème primal**.

L'idée centrale de la dualité Lagrangienne est d'identifier des situations pour lesquelles, résoudre le problème duale permet de résoudre la problème primal. Ces situations sont identifiées par la dualité forte. Pour le moment, on peut réécrire la propriété de dualité faible par

$$D^* := \max_{\alpha \geq 0, \beta} D(\alpha, \beta) \leq \min_{x \in K} J(x) := J^*. \quad (2.3)$$

On a donc que D^* est toujours une borne inférieure pour J^* .

Définition 2.5. La quantité $J^* - D^*$ est appelée **duality gap**.

D'après la Proposition 2.3, le *duality gap* est toujours positif. On va dans la suite identifier des situations de dualité forte càd quand $D^* = J^*$ autrement dit quand le *duality gap* est nul.

On va identifier ces situations pour des problèmes d'optimisation sous contraintes de la forme

$$\boxed{\min_{x \in K} J(x) \text{ où } K = \left\{ x \in \mathbb{R}^n : \begin{array}{ll} c_i(x) \leq 0 & \text{pour } i \in \mathcal{I} \end{array} \right\} \text{ et } \begin{array}{l} J, c_i, i \in \mathcal{I} \text{ sont convexes} \\ A \text{ est de rang } m. \end{array}} \quad (\text{P1})$$

Il y a beaucoup de résultats établissant la dualité forte pour les problèmes de la forme (P1) sous certaines conditions appelées **qualification de contraintes**. Une qualification de contrainte simple est la condition de Slater.

Définition 2.6. On dit que la contrainte K définie par les fonctions $(c_i)_{i \in \mathcal{I}}$ satisfait les **conditions de Slater** quand il existe $x \in \mathbb{R}^n$ tel que

$$Ax = b \text{ et } \begin{cases} c_i(x) \leq 0 & \text{pour tout } i \in \mathcal{I}_a \\ c_i(x) < 0 & \text{pour tout } i \in \mathcal{I} \setminus \mathcal{I}_a \end{cases}$$

où $\mathcal{I}_a \subset \mathcal{I}$ est l'ensemble de toutes les contraintes affines de K (càd, pour tout $i \in \mathcal{I}_a$, c_i est une fonction affine).

Théorème 2.7 (dualité forte). On suppose que la contrainte K satisfait les conditions de Slater pour le problème (P1) et que (P1) admet au moins une solution. Alors

$$\min_{x \in K} J(x) = \max_{\alpha \geq 0, \beta} D(\alpha, \beta).$$

De plus, le problème dual admet au moins une solution (α^*, β^*) et le problème $\inf_{x \in \mathbb{R}^n} L(x, \alpha^*, \beta^*)$ admet une solution x^* qui est aussi solution de (P1).

Les conditions de Slater assurent donc la dualité forte pour des problèmes d'optimisation convexe pour des contraintes d'égalité affines.

Une manière géométrique de voir la dualité forte est de remarquer que pour des problème convexe,

1. pour tout x , $(\alpha, \beta) \longrightarrow L(x, \alpha, \beta)$ est concave
2. pour tout $(\alpha \geq 0, \beta)$, $x \longrightarrow L(x, \alpha, \beta)$ est convexe

alors si (x^*, α^*, β^*) est tel que

$$\min_x \max_{\alpha \geq 0, \beta} L(x, \alpha, \beta) = L(x^*, \alpha^*, \beta^*) = \max_{\alpha \geq 0, \beta} \min_x L(x, \alpha, \beta)$$

ce point est appelé **point selle**.

Remarque 2.8. La dualité forte permet aussi de définir un critère d'arrêt pour les algorithmes itératifs. En effet, si x^* est une solution primale et (α^*, β^*) est une solution duale alors $J(x^*) - D(\alpha^*, \beta^*) = 0$. On peut alors se servir de la quantité $J(x) - D(\alpha, \beta)$ comme d'un critère d'arrêt pour un algorithme itératif : si à l'étape k on a $J(x^{(k)}) - D(\alpha^{(k)}, \beta^{(k)}) \leq \epsilon$ alors on stoppe les itérations pour un niveau de tolérance ϵ donné en entrée.

Remarque 2.9 (Qualification de contraintes). *Les conditions de Slater sont un exemple de qualification de contraintes. D'une manière générale, on dit qu'une contrainte*

$$K = \left\{ x \in \mathbb{R}^n : \begin{array}{ll} c_i(x) \leq 0 & \text{pour } i \in \mathcal{I} \\ e_i(x) = 0 & \text{pour } i \in \mathcal{E} \end{array} \right\}$$

est qualifiée en un point x de K quand :

1. *la famille $(\nabla e_i(x))_{i \in \mathcal{E}}$ est libre ;*
2. *il existe un vecteur $v \in \mathbb{R}^n$ tel que $\langle \nabla e_i(x), v \rangle = 0$ pour tout $i \in \mathcal{E}$ et pour tout $i \in \mathcal{I}$ tel que $c_i(x) = 0$, $\langle \nabla c_i(x), v \rangle < 0$.*

Quand une contrainte est qualifiée en un point x_0 de K , on dit que la contrainte est qualifiée. Si tel est le cas et que le problème (P) admet au moins une solution alors le duality gap est nul.

2.2 Conditions KKT

On considère à nouveau le problème général d'optimisation sous contraintes (P). Dans le cas convexe (càd quand J et les c_i, e_i sont convexes) et quand le dualité gap est nul, càd quand

$$\min_{x \in K} J(x) = \max_{\alpha \geq 0, \beta} D(\alpha, \beta).$$

Dans cette situation, il existe des conditions nécessaires et suffisantes d'optimalité pour les deux problèmes primal et dual. Ces conditions sont appelées conditions KKT du nom de leurs auteurs Karush, Kuhn et Tucker.

Définition 2.10. *Soit $J, (c_i)_{i \in \mathcal{I}}, (e_i)_{i \in \mathcal{E}}$ des fonctions différentiables. On dit que (x^*, α^*, β^*) satisfait les **conditions KKT** quand, pour $K = \{x \in \mathbb{R}^n : c_i(x) \leq 0, \forall i \in \mathcal{I}, e_i(x) = 0, \forall i \in \mathcal{E}\}$,*

$$x^* \in K \text{ et } \alpha^* \geq 0 \tag{KKT1}$$

$$\alpha_i^* c_i(x^*) = 0, \forall i \in \mathcal{I} \tag{KKT2}$$

$$\nabla_x L(x^*, \alpha^*, \beta^*) = 0 \tag{KKT3}$$

*La condition (KKT2) est appelée **complementary slackness condition**.*

La première conditions (KKT1) dit simplement que les éléments x^* et (α^*, β^*) sont dans leur ensemble de contraintes respectifs (pour le primal et le dual). La condition (KKT2) signifie que soit $c_i(x^*) = 0$, càd la i -ième contrainte est saturée, soit le i -ième multiplicateur de Lagrange α_i^* est nul. Finalement, la troisième condition (KKT3) signifie que x^* est un point optimal de $L(\cdot, \alpha^*, \beta^*)$.

Théorème 2.11. *On considère le problème d'optimisation sous contraintes (P) où J, c_i, e_i sont différentiables. On a les résultats suivant :*

1. *si le duality gap est nul, si x^* est une solution primale et si (α^*, β^*) est une solution duale, alors (x^*, α^*, β^*) satisfait les conditions KKT.*
2. *de plus, si le problème est convexe (càd quand J, c_i, e_i sont convexes) alors les conditions KKT sont suffisantes : si (x^*, α^*, β^*) satisfait KKT alors x^* est solution du problème primal, (α^*, β^*) est solution du problème duale et le duality gap est nul.*

Démonstration. On démontre le premier point. Soit x^* une solution primale et (α^*, β^*) une solution duale. On a

$$J(x^*) \stackrel{(1)}{=} D(\alpha^*, \beta^*) \stackrel{(2)}{\leq} J(x^*) + \sum_{i \in \mathcal{I}} \alpha_i^* c_i(x^*) + \sum_{i \in \mathcal{E}} \beta_i^* e_i(x^*) \stackrel{(3)}{\leq} J(x^*)$$

où (1) est obtenue car le duality gap est nul, (2) est obtenue par définition de D et (3) est obtenue car $x^* \in K$. On en déduit que toutes les inégalités sont des égalités et que donc

$$\sum_{i \in \mathcal{I}} \alpha_i^* c_i(x^*) + \sum_{i \in \mathcal{E}} \beta_i^* e_i(x^*) = 0.$$

Par ailleurs, comme $e_i(x^*) = 0$ et les termes de la somme $\sum_{i \in \mathcal{I}} \alpha_i^* c_i(x^*)$ sont négatifs ou nuls (car $\alpha_i^* \geq 0$ et $c_i(x^*) \leq 0$), on obtient que chaque terme doit être nul. Ce qui est bien la complementary slackness condition.

Par ailleurs, comme $x^* \in K$ et $\alpha^* \geq 0$, on a

$$L(x^*, \alpha^*, \beta^*) \leq J(x^*) = D(\alpha^*, \beta^*) = \min_{x \in \mathbb{R}^n} L(x, \alpha^*, \beta^*).$$

Donc, $L(\cdot, \alpha^*, \beta^*)$ est minimale en x^* . Comme $L(\cdot, \alpha^*, \beta^*)$ est différentiable, on a nécessairement que $\nabla_x L(x^*, \alpha^*, \beta^*) = 0$. Ce qui démontre la troisième et dernière condition KKT.

Réciproquement, on démontre le deuxième point sous l'hypothèse supplémentaire que le problème est convexe. Soit (x^*, α^*, β^*) satisfaisant les conditions KKT. Condition (KKT1) assure que x^* et (α^*, β^*) sont dans les ensembles de contraintes respectifs du problème primal et dual. Par ailleurs, comme $\alpha^* \geq 0$, $L(\cdot, \alpha^*, \beta^*) = J(\cdot) + \sum_{i \in \mathcal{I}} \alpha_i^* c_i(\cdot)$ est convexe et comme $\nabla_x L(x^*, \alpha^*, \beta^*) = 0$, x^* est un minimum de $L(\cdot, \alpha^*, \beta^*)$. On a donc

$$\begin{aligned} D(\alpha^*, \beta^*) &= \min_x L(x, \alpha^*, \beta^*) = L(x^*, \alpha^*, \beta^*) \\ &= J(x^*) + \sum_{i \in \mathcal{I}} \alpha_i^* c_i(x^*) + \sum_{i \in \mathcal{E}} \beta_i^* e_i(x^*) = J(x^*) \end{aligned}$$

où la dernière inégalité est obtenue d'après la complementary slackness condition $\alpha_i^* c_i(x^*) = 0$ et $e_i(x^*) = 0$ car $x^* \in K$. Donc, le duality gap est nul. Par ailleurs, pour tout $\alpha \geq 0, \beta$, $D(\alpha, \beta) \leq J(x)$, alors, comme le duality gap est nul, on a $J(x^*) \leq J(x)$ pour tout $x \in \mathbb{R}^n$. Donc x^* est solution du problème primal. De même, $D(\alpha, \beta) \leq J(x^*) = D(\alpha^*, \beta^*)$ donc (α^*, β^*) est solution du problème dual.

On a donc bien montré que le duality gap est nul, que x^* est une solution primale et que (α^*, β^*) est une solution duale. ■

Conclusion : Pour les problèmes convexes différentiables (càd quand J, c_i, e_i sont différentiables et convexes) les conditions KKT sont nécessaires et suffisantes : il y a équivalence entre

1. le duality gap est nul, x^* est solution de (P) et (α^*, β^*) est solution de (D)
2. (x^*, α^*, β^*) vérifie les conditions KKT.

De plus, pour les problèmes de la forme (P1) vérifiant les conditions de Slater, si x^* est une solution de (P1) alors il existe (α, β) tel que (x^*, α, β) vérifie les conditions KKT et dans ce cas (α, β) est solution du problème dual.

2.3 Dualité Lagrangienne et conditions KKT en programmation linéaire

Dans cette section, on applique les idées des deux dernières sous-section (à savoir la dualité Lagrangienne et les conditions KKT) au cas particulier des problèmes d'optimisation de programmation linéaire sous forme standard

$$\min_{x \in \mathbb{R}^n} \langle c, x \rangle \text{ tel que } Ax = b, x \geq 0 \quad (\text{PPL})$$

où $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ et $b \in \mathbb{R}^m$. On suppose que A est de rang m (ceci pour se placer dans le cadre des conditions de Slater).

La fonction objectif est convexe de classe \mathcal{C}^1 et les contraintes sont affines. On peut appliquer tous les résultats de la partie précédente.

La fonction de Lagrange est ici

$$L(x, \alpha, \beta) = \langle c, x \rangle - \langle \alpha, x \rangle + \langle \beta, b - Ax \rangle$$

définie pour tout $x \in \mathbb{R}^n, \alpha \in (\mathbb{R}_+)^n$ et $\beta \in \mathbb{R}^m$. La fonction duale de Lagrange est définie pour tout $\alpha \in (\mathbb{R}_+)^n$ et $\beta \in \mathbb{R}^m$ par

$$D(\alpha, \beta) = \min_{x \in \mathbb{R}^n} L(x, \alpha, \beta) = \begin{cases} -\infty & \text{quand } A^\top \beta + \alpha \neq c \\ \langle \beta, b \rangle & \text{quand } A^\top \beta + \alpha = c. \end{cases}$$

Le problème dual $\max_{\alpha \geq 0, \beta} D(\alpha, \beta)$ est donc ici

$$\max_{\alpha, \beta} \langle b, \beta \rangle \text{ tel que } A^\top \beta + \alpha = c, \alpha \geq 0 \quad (\text{DPL})$$

C'est aussi un problème de programmation linéaire.

Les conditions de Slater sont satisfaites quand il existe un $x \in \mathbb{R}^n$ tel que $Ax = b$ et $c_i(x) = -x_i \leq 0, i = 1, \dots, n$. Les conditions de Slater sont donc ici équivalentes à l'existence d'un élément dans l'espace des contraintes de (PPL). Dans ce cas, on a la dualité forte (cf. Théorème 2.7) et d'après les conditions KKT (cf. Théorème 2.11) il y a équivalence entre :

1. x^* est solution de (PPL) et (α^*, β^*) est solution de (DPL)
2. (x^*, α^*, β^*) vérifie les conditions de KKT :

$$(KKT1) : Ax^* = b, x^* \geq 0 \text{ et } \alpha^* \geq 0$$

$$(KKT2) : \alpha_i^* x_i^* = 0, \forall i = 1, \dots, n$$

$$(KKT3) : 0 = \nabla_x L(x^*, \alpha^*, \beta^*) = A^\top \beta^* - \alpha^* + c$$

On peut vérifier que le dual du problème dual (DPL) est ici le problème primal (PPL). En effet, le lagrangien de (DPL) est

$$\tilde{L}(\alpha, \beta, x, y) = -\langle b, \beta \rangle + \langle x, A^\top \beta + \alpha - c \rangle + \langle y, \alpha \rangle = \langle Ax - b, \beta \rangle + \langle x - y, \alpha \rangle - \langle x, c \rangle$$

où $(x, y) \in \mathbb{R}^n \times (\mathbb{R}_+)^m$ est ici le multiplicateur de Lagrange. Puis, la fonction duale de Lagrange est ici définie pour tout $x, y \geq 0$ par

$$\tilde{D}(x, y) = \min_{\alpha, \beta} \tilde{L}(\alpha, \beta, x, y) = \begin{cases} -\infty & \text{quand } Ax \neq b \text{ ou } x \neq y \\ -\langle c, x \rangle & \text{quand } Ax = b \text{ et } x = y. \end{cases}$$

Le problème duale $\max_{y \geq 0, x} \tilde{D}(x, y)$ est alors

$$\max \left(-\langle c, x \rangle : Ax = b \text{ et } x = y \text{ et } y \geq 0 \right)$$

qui est bien équivalent à (PPL).

On peut donc de nouveau appliquer la dualité Lagrangienne et obtenir la caractérisation des solutions aux problèmes primal et dual par les conditions KKT qui sont identiques à celles obtenues précédemment.

Théorème 2.12. *Si le problème primal (PPL) a une solution ou si le problème dual (DPL) a une solution alors c'est aussi le cas pour l'autre et les fonctions objectifs sont égales. Si un des problèmes n'est pas borné alors l'ensemble des contraintes de l'autre est vide. Par ailleurs, on a l'équivalence entre :*

1. x^* est solution de (PPL) et (α^*, β^*) est solution de (DPL)
2. (x^*, α^*, β^*) vérifie les conditions de KKT :

$$Ax^* = b, x^* \geq 0 \text{ et } \alpha^* \geq 0; \quad \alpha_i^* x_i^* = 0, \forall i = 1, \dots, n; \quad A^\top \beta^* - \alpha^* + c = 0.$$

2.4 Directions de descentes (de Newton)

2.4.1 Méthode de Newton pour "F(x) = 0"

Soit $F : \mathbb{R} \rightarrow \mathbb{R}$ une fonction différentiable sur \mathbb{R} . On souhaite trouver $x^* \in \mathbb{R}$ tel que $F(x^*) = 0$. La méthode de Newton consiste à approcher la fonction F par son développement du premier ordre de Taylor : pour tout $x \in \mathbb{R}$ et $p \in \mathbb{R}$

$$F(x + p) = F(x) + F'(x)p + o(p).$$

Ainsi, si on choisit p^* tel que

$$F(x) + F'(x)p^* = 0 \tag{2.4}$$

on aura $F(x + p^*) = o(p^*)$ et donc si p^* est petit, on aura $F(x + p^*)$ qui sera proche de zéro, et donc $x + p^*$ sera une solution approchante au problème qu'on souhaite résoudre ici.

L'algorithme de Newton pour résoudre "F(x) = 0" est un algorithme itératif qui commence d'un point initial donné et s'arrête lorsqu'un critère d'arrêt est rencontré :

input : x_0 : un point initial; $\epsilon > 0$: un paramètre de critère d'arrêt

output: une approximation x_k de x^* tel que $F(x^*) = 0$

```

1 while  $|x_{k+1} - x_k| \geq \epsilon$  do
2   | Calcul de  $p_k$ , une direction de descente solution de  $F(x_k) + F'(x_k)p_k = 0$ 
3   |  $x_k = x_k + p_k$ 
4 end
```

Algorithm 1: Algorithme de Newton pour $F(x) = 0$.

On voit ici que le choix de la direction de descente p_k au point p_k est donnée par l'équation 2.4 qui consiste à annuler le développement du premier ordre de F en x_k . Quand $F'(x_k) \neq 0$ alors la direction de descente est donnée par $p_k = -F(x_k)/F'(x_k)$ qui est la forme classique de l'algorithme de Newton :

$$x_{k+1} = x_k - \frac{F(x_k)}{F'(x_k)}.$$

2.4.2 Descente de gradient et algorithme de Newton

Soit $f : \mathbb{R} \rightarrow \mathbb{R}$ une fonction convexe deux fois différentiable. On cherche à trouver une solution au problème $\min_{x \in \mathbb{R}} f(x)$.

Par convexité et différentiabilité de f , résoudre ce problème est équivalent à résoudre le problème $f'(x) = 0$. On est donc amené à résoudre un problème du type $F(x) = 0$ pour $F = f'$ qu'on peut résoudre de manière approchée par la méthode de Newton vue dans la section précédente.

Dans le cadre de la minimisation d'une fonction convexe, la méthode de Newton est un exemple particulier de **méthode de descente de gradient**.

Pour le problème de minimisation $\min_{x \in \mathbb{R}} f(x)$ d'une fonction convexe, la méthode de Newton consiste à minimiser une approximation de Taylor de second ordre de f : pour tout $x, p \in \mathbb{R}$,

$$f(x+p) = f(x) + f'(x)p + \frac{p^2 f''(x)}{2} + o(p^2).$$

Minimiser $p \rightarrow f(x) + f'(x)p + (1/2)p^2 f''(x)$ revient à trouver un zéro de son gradient : p^* tel que

$$f'(x) + f''(x)p^* = 0$$

qui est exactement la même équation que celle définissant la direction de descente pour la méthode de Newton dans (2.4) pour $F = f'$.

input : $x_0 \in \mathbb{R}^n$: un point initial ; $\epsilon > 0$: un paramètre de critère d'arrêt
output: une approximation $x_k \in \mathbb{R}^n$ de x^* tel que $f(x^*) = \min_x f(x)$

```

1 while  $|x_{k+1} - x_k| \geq \epsilon$  do
2   | Calcul de  $p_k$  est une direction de descente solution de  $f'(x_k) + f''(x_k)p_k = 0$ 
3   |  $x_k = x_k + p_k$ 
4 end
```

Algorithm 2: Algorithme de Newton pour le problème $\min_x f(x)$.

Ainsi quand $f''(x_k) \neq 0$, la direction de descente est donnée par $p_k = -f'(x_k)/f''(x_k)$.

Dans certain cas, il est coûteux, voire impossible de calculer $f''(x_k)$. On a donc recours à une suite donnée a priori, ou calculée au fil des itérations, qu'on appelle les **steps size** : $(\eta_k)_k$ où $\eta_k > 0$. La direction de descente est obtenue à l'étape k comme étant solution de l'équation

$$f'(x) + \frac{p_k}{\eta_k} = 0.$$

On a donc "remplacer" la quantité difficile à obtenir $f''(x_k)$ par $1/\eta_k$. La direction de descente ainsi obtenue est $p_k = -\eta_k f'(x_k)$ et l'algorithme associé est

$$x_{k+1} = x_k - \eta_k f'(x_k)$$

qui l'algorithme de **descente de gradient**. L'algorithme de Newton est donc un algorithme de descente de gradient pour un step size $\eta_k = 1/f''(x_k)$.

2.4.3 Algorithme de Newton pour des fonctions de plusieurs variables

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe deux fois différentiable. On souhaite résoudre le problème d'optimisation $\min_{x \in \mathbb{R}^n} f(x)$.

La méthode de Newton consiste à approcher f au point courant x_k par son développement d'ordre 2 de Taylor :

$$f(x_k + p) = f(x_k) + \langle \nabla f(x_k), p \rangle + \frac{1}{2} p^\top \nabla^2 f(x_k) p + o(\|p\|_2^2)$$

et à minimiser en $p \in \mathbb{R}^n$ cette approximation. L'approximation étant convexe et différentiable, cela revient à annuler son gradient. On obtient donc une direction de descente p_k solution de l'équation :

$$\nabla f(x_k) + \nabla^2 f(x_k) p_k = 0 \quad (2.5)$$

où $\nabla f(x) = (\partial_i f(x))_{i=1}^n$ est le **gradient** de f en x et $\nabla^2 f(x) = (\partial_i \partial_j f(x))_{1 \leq i, j \leq n}$ est la matrice **Hessienne** de f en x .

Ainsi, quand $\nabla^2 f(x_k)$ est inversible, la direction de descente est donnée par

$$p_k = -(\nabla^2 f(x_k))^{-1} \nabla f(x_k)$$

et l'algorithme de Newton associé est décrit dans Algorithme 3.

input : $x_0 \in \mathbb{R}^n$: un point initial; $\epsilon > 0$: un paramètre de critère d'arrêt
output: une approximation $x_k \in \mathbb{R}^n$ de x^* tel que $f(x^*) = \min_x f(x)$

1 **while** $\|x_{k+1} - x_k\|_2 \geq \epsilon$ **do**
2 $x_k = x_k - (\nabla^2 f(x_k))^{-1} \nabla f(x_k)$
3 **end**

Algorithme 3: Algorithme de Newton pour la construction d'une solution approchante au problème $\min_{x \in \mathbb{R}^n} f(x)$.

L'algorithme de Newton converge très rapidement vers la solution du problème d'optimisation $\min_x f(x)$ quand $x \rightarrow \nabla^2 f(x)$ est Lipschitz et $\nabla^2 f(x)$ a une plus petite valeur singulière plus grande que c uniformément en x . On rappelle la définition de la norme d'opérateur d'une matrice :

$$\|M\| = \max_{\|x\|_2=1} \|Mx\|_2.$$

Théorème 2.13. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe deux fois différentiable. On suppose qu'il existe $x^* \in \mathbb{R}^n$ tel que $f(x^*) = \min_{x \in \mathbb{R}^n} f(x)$. On suppose que :

i) $\nabla^2 f$ est L -Lipschitz pour la norme d'opérateur : pour tout $x, y \in \mathbb{R}^n$,

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \leq L \|x - y\|_2$$

ii) pour tout $x \in \mathbb{R}^n$, $\nabla^2 f(x) \succeq cI_n$

Soit $x_0 \in \mathbb{R}^n$. Alors, l'algorithme de descente de gradient $(x_k)_k$ défini par x_0 et $x_{k+1} = x_k - (\nabla^2 f(x_k))^{-1} \nabla f(x_k)$ vérifie $\|x_{k+1} - x^*\|_2 \leq [L/(2c)] \|x_k - x^*\|_2^2$ pour tout $k \in \mathbb{N}$.

Démonstration. On a $\nabla f(x^*) = 0$, alors d'après le théorème fondamental de l'analyse,

$$\nabla f(x_k) = \nabla f(x_k) - \nabla f(x^*) = \int_0^1 \nabla^2 f(x^* + \tau(x_k - x^*))(x_k - x^*) d\tau.$$

On a donc

$$\begin{aligned}
x_{k+1} - x^* &= x_k - x^* - (\nabla^2 f(x_k))^{-1} \nabla f(x_k) \\
&= x_k - x^* - (\nabla^2 f(x_k))^{-1} \int_0^1 \nabla^2 f(x^* + \tau(x_k - x^*)) (x_k - x^*) d\tau \\
&= (\nabla^2 f(x_k))^{-1} \int_0^1 [\nabla^2 f(x_k) - \nabla^2 f(x^* + \tau(x_k - x^*))] (x_k - x^*) d\tau.
\end{aligned}$$

On en déduit que

$$\|x_{k+1} - x^*\|_2 \leq \|\nabla f(x_k)^{-1}\| \left\| \int_0^1 [\nabla^2 f(x_k) - \nabla^2 f(x^* + \tau(x_k - x^*))] d\tau \right\| \|x_k - x^*\|_2.$$

On conclut en observant que $\|\nabla f(x_k)^{-1}\| = \|\nabla f(x_k)\|^{-1} \leq (1/c)$ et

$$\begin{aligned}
\left\| \int_0^1 [\nabla^2 f(x_k) - \nabla^2 f(x^* + \tau(x_k - x^*))] d\tau \right\| &\leq \int_0^1 \|\nabla^2 f(x_k) - \nabla^2 f(x^* + \tau(x_k - x^*))\| d\tau \\
&\leq \int_0^1 L(1 - \tau) \|x_k - x^*\|_2 d\tau = (L/2) \|x_k - x^*\|_2.
\end{aligned}$$

■

Ainsi l'algorithme de Newton converge de manière quadratique quand ∇f satisfait certaines conditions. C'est donc une convergence très rapide. La condition ii) est en particulier vérifiée pour les fonctions deux fois différentiables et **fortement convexes**, càd qui vérifient pour tout $x, y \in \mathbb{R}^n$ et $0 \leq t \leq 1$,

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y) - (c/2)t(1 - t) \|x - y\|_2^2$$

2.4.4 Directions de descente pour les problèmes sous contraintes

Dans les deux sous-sections précédentes, les directions de descente p étaient obtenues par minimisation d'un développement de Taylor du second ordre de la fonction convexe f à minimiser. Le choix de p n'était pas contraint et la minimisation de ce critère était le seul objectif.

Pour un problème d'optimisation sous contraintes, le choix de la direction de descente ne peut plus uniquement se faire dans le seul objectif de minimiser une approximation de f : au point courant x_k , on doit s'assurer que l'itération suivante $x_{k+1} = x_k + p_k$ est dans l'ensemble des contraintes. On s'assure que la direction de descente n'amène pas l'algorithme à produire des itérations qui sortent de l'ensemble de contrainte.

Soit le problème d'optimisation suivant

$$\boxed{\min_{x \in K} f(x)} \tag{2.6}$$

où $K \subset \mathbb{R}^n$ et $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est une fonction convexe deux fois différentiable.

L'algorithme de descente de gradient peut aussi être utilisé pour les problèmes d'optimisation sous contraintes à condition de s'assurer que les itérés de l'algorithme restent dans K .

Comme précédemment, au point courant x_k de l'algorithme, on fait une approximation de Taylor du second ordre de f :

$$f(x_k + p) = f(x_k) + \langle \nabla f(x_k), p \rangle + \frac{1}{2} p^\top \nabla^2 f(x_k) p + o(\|p\|_2^2).$$

et on minimise cette approximation tout en s'assurant que $x + p$ est bien dans l'ensemble de contraintes K . Ceci définit la direction de descente pour un problème d'optimisation sous contrainte.

Définition 2.14. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ convexe deux fois différentiable et $K \subset \mathbb{R}^n$ non vide. Soit $x \in K$, la **direction de descente (de Newton)** en x de f est une solution au problème d'optimisation

$$\min_{p \in \mathbb{R}^n} \left(\langle \nabla f(x), p \rangle + \frac{1}{2} p^\top \nabla^2 f(x) p : x + p \in K \right). \quad (2.7)$$

Remarque 2.15. Ici on s'assure que les itérations successives appartiennent bien à l'ensemble des contraintes K en cherchant les directions de descente p telles que $x_k + p \in K$. Une autre stratégie consiste à considérer la direction de descente comme dans les algorithmes de descentes de gradient, càd $p_k = -\eta_k \nabla f(x_k)$ et ensuite à projeter $x_k + p_k$ sur K : $x_{k+1} = \text{proj}_K(x_k + p_k)$ où proj_K est un opérateur de projection sur K (par exemple, $\text{proj}_K(y) \in \text{argmin}_{z \in K} \|y - z\|_2$). Ces procédures sont appelées **méthodes de descente de gradient projetée**; elles sont très populaires en optimisation convexe.

2.5 Les méthodes de barrières

Les méthodes de barrières sont aussi connues sous le nom de méthodes de points intérieurs, au-delà du cadre des problèmes de programmation linéaire. Elles sont utilisées pour construire des solutions approchantes aux problèmes d'optimisation sous contraintes. On considère donc un tel problème où les contraintes d'égalité sont affines comme considérées dans (P1) :

$$\boxed{\min_{x \in K} J(x) \text{ où } K = \left\{ x \in \mathbb{R}^n : \begin{array}{ll} c_i(x) \leq 0 & \text{pour } i \in \mathcal{I} \end{array} \right\} \text{ et } \begin{array}{l} J, c_i, i \in \mathcal{I} \text{ sont convexes} \\ A \text{ est de rang } m. \end{array}} \quad (\text{P1})$$

Les méthodes de barrières se proposent de trouver une solution approchante au problème (P1) en résolvant une suite de problèmes d'optimisation sans contraintes d'inégalités. Pour cela, on ajoute à la fonction objectif $J(\cdot)$ une fonction de coût qui pénalise les points de K qui s'approche de la frontière de K (vu comme sous-ensemble de $\{x : Ax = b\}$). C'est ce coût additionnel qui crée une sorte de barrière interdisant la sortie de K .

Définition 2.16. Une **fonction barrière pour (P1)** est une fonction continue $b : \overset{\circ}{K} \rightarrow \mathbb{R}$ (où $\overset{\circ}{K}$ est l'intérieur de K dans $\{x : Ax = b\}$) telle que $b(x) \rightarrow +\infty$ quand $\max_{i \in \mathcal{I}} c_i(x) \rightarrow 0^-$.

Les deux exemples classiques de fonctions barrières sont

$$b(x) = -\sum_{i \in \mathcal{I}} \log(-c_i(x)) \text{ et } b(x) = -\sum_{i \in \mathcal{I}} \frac{1}{c_i(x)}.$$

L'objectif est de dissuader les points de K de s'approcher de la frontière de K . On introduit alors une famille de fonctions objectif indexée par un paramètre $\mu > 0$:

$$B(x, \mu) = J(x) + \mu b(x)$$

où $b(\cdot)$ est une fonction barrière. Le problème d'optimisation associé est

$$\boxed{\min_{x: Ax=b} B(x, \mu)} \quad (\text{PB}\mu)$$

On obtient ainsi une suite de problèmes d'optimisation ((PB μ)) indexée par un paramètre $\mu > 0$. L'idée maintenant est de résoudre ces problèmes et de faire tendre μ vers 0 pour construire une solution approchante au problème d'origine (P1).

Remarque 2.17. Une manière naturelle de construire une barrière (non-continue) du bord de K dans $\{x : Ax = b\}$ est de prendre la fonction indicatrice de $\{x : c_i(x) \leq 0, i \in \mathcal{I}\}$, càd

$$b(x) = \sum_{i \in \mathcal{I}} i_{\mathbb{R}_-}(c_i(x)) \text{ où } i_{\Omega}(t) = \begin{cases} 0 & \text{si } t \in \Omega \\ +\infty & \text{si } t \notin \Omega \end{cases}$$

pour tout $t \in \mathbb{R}$ et $\Omega \subset \mathbb{R}$. On peut voir les fonctions barrières de base $t \in \mathbb{R}_-^* \rightarrow -\mu \log(-t)$ et $t \in \mathbb{R}_-^* \rightarrow -\mu/t$ comme des fonctions continues convergeant simplement vers $i_{\mathbb{R}_-}$ sur \mathbb{R}_-^* quand $\mu \rightarrow 0^-$.

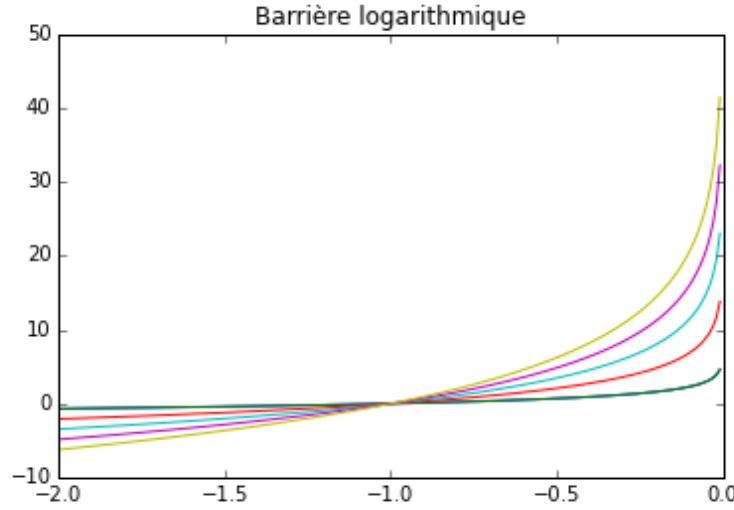


FIGURE 1 – Approximation de la barrière idéale $i_{\mathbb{R}_-}$ par les barrières logarithmiques.

Théorème 2.18. (Convergence des méthodes de barrière) On suppose que J et $(c_i)_{i \in \mathcal{I}}$ sont continues et que b est une fonction barrière (continue) pour (P1). Pour tout $\mu > 0$, on suppose que (PB μ) admet au moins une solution. On suppose aussi que K est sans point isolé.

Soit $(\mu_k)_k$ une suite décroissante de réels strictement positifs et pour tout $k \in \mathbb{N}$, on note par x_k une solution de (PB μ_k) pour $\mu = \mu_k$. Tout point d'adhérence de $(x_k)_k$ est solution de (P1).

Démonstration. Soit \bar{x} un point d'adhérence de $(x_k)_k$. Pour ne pas charger les notations, on note encore $(x_k)_k$ la sous-suite convergeante vers \bar{x} .

Par continuité de $x \rightarrow Ax$, on a $b = Ax_k \rightarrow A\bar{x}$ quand $k \rightarrow +\infty$. En particulier, $A\bar{x} = b$. De même, par continuité de c_i pour $i \in \mathcal{I}$, on a $c_i(\bar{x}) \leq 0$ pour tout $i \in \mathcal{I}$. On a donc $\bar{x} \in K$.

Si pour tout $i \in \mathcal{I}$, $c_i(\bar{x}) < 0$ alors $\mu_k c_i(x_k) \rightarrow 0$ quand $k \rightarrow +\infty$ et ceci pour tout $i \in \mathcal{I}$. Si \bar{x} est sur le bord de K alors par construction $b(x_k) \rightarrow +\infty$ quand $k \rightarrow +\infty$. Dans les deux cas, on a $\liminf_k \mu_k b(x_k) \geq 0$. On en déduit que

$$\liminf_k (J(x_k) + \mu_k b(x_k)) = J(\bar{x}) + \liminf_k \mu_k b(x_k) \geq J(\bar{x}).$$

Par définition de l'optimalité de x_k , on a pour tout x tel que $Ax = b$ et $b(x) < \infty$

$$J(x) + \mu_k b(x) \geq J(x_k) + \mu_k b(x_k)$$

et en passant à la limite inférieure pour $k \rightarrow \infty$, on a $J(x) \geq J(\bar{x})$. Ceci étant vrai pour tout $x \in \overset{\circ}{K}$, on en déduit, par continuité de J et comme K est sans point isolé, que pour tout $x \in K$, $J(x) \geq J(\hat{x})$. Donc \hat{x} est bien solution au problème (P1). ■

Un algorithme de barrière consiste à résoudre une suite de problème (PB μ) pour $\mu = \mu_k$ où $\mu_k \downarrow 0$ et à utiliser cette suite de solution comme une approximation de la solution du problème initial (P1).

Concrètement, on se fixe un nombre fini de problèmes du type (PB μ) pour $\mu = \mu_k$ où $\mu_k \downarrow 0$ et $k \in \{1, \dots, k_{max}\}$. La dernière solution $x_{k_{max}}$ est utilisé comme solution approchante de (P1).

On peut alors se demander pourquoi on ne résout pas seulement le dernier problème (PB μ) pour $\mu = \mu_{k_{max}}$? Cela est dû au fait que la Hessienne de $B(\cdot, \mu)$ varie très rapidement en s'approchant du bord de K . La méthode de Newton est donc très instable : un choix légèrement différent du point initial dans la méthode de Newton produira de grands changements dans la suite des itérations de l'algorithme. On va alors résoudre une suite de problèmes (PB μ) en décroissant le paramètre μ à chaque itération et utiliser la solution de dernier problème comme point initial du problème suivant.

Les solutions $x^*(\mu)$ de (PB μ) joue un rôle central pour les méthodes de barrière. L'ensemble de ces points décrit un "chemin" dans l'intérieur des contraintes.

Définition 2.19. *On suppose que pour tout $\mu > 0$, le problème (PB μ) admet une unique solution notée $x^*(\mu)$. L'ensemble $\{x^*(\mu) : \mu > 0\}$ est appelé **central path**.*

3 Un algorithme de barrière primal en programmation linéaire

On se place maintenant dans le cadre des problèmes de programmation linéaire. On considère alors le problème sous forme standard

$$\min_{x \in \mathbb{R}^n} \langle c, x \rangle \text{ tel que } Ax = b, x \geq 0 \quad (\text{PPL})$$

où $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ et $b \in \mathbb{R}^m$. On suppose que A est de rang m .

On utilise une méthode de barrière logarithmique associé à (PPL) : pour tout $\mu > 0$, on introduit la fonction objectif

$$B(x, \mu) = \langle x, c \rangle - \mu \sum_{i=1}^n \log(x_i)$$

(où on étend la définition du log à \mathbb{R} en posant $\log t = -\infty$ quand $t \leq 0$) et on considère le problème d'optimisation

$$\min_{x \in \mathbb{R}^n} B(x, \mu) \text{ tel que } Ax = b. \quad (\text{PPL}\mu)$$

La fonction barrière nous a donc permis de retirer la contrainte $x \geq 0$.

Remarque 3.1. *On définit les fonctions $B(\cdot, \mu)$ sur tout l'espace \mathbb{R}^n à condition d'autoriser la valeur $+\infty$ due à l'extension de la définition de $t \rightarrow \log t$ à tout \mathbb{R} . Le formalisme de la dualité Lagrangienne et des conditions KKT s'applique encore dans ce cadre si on restreint l'étude à leur domaine, càd à l'ensemble des points où la fonction est finie. On renvoie le lecteur intéressé au livre [5] pour une exposition de ces résultats.*

La fonction de Lagrange associée au problème (PPL μ) est

$$L_\mu(x, \lambda) = B(x, \mu) + \langle \lambda, b - Ax \rangle = \langle x, c \rangle - \mu \sum_{i=1}^n \log(x_i) + \langle \lambda, b - Ax \rangle \quad (3.1)$$

où $\lambda \in \mathbb{R}^m$ est le multiplicateur de Lagrange associé à la contrainte " $Ax = b$ ". On a pour tout $x \in \mathbb{R}^n$ et $\lambda \in \mathbb{R}^m$,

$$\nabla_x L_\mu(x, \lambda) = c - A^\top \lambda - \mu X^{-1} e$$

où pour tout $x \in \mathbb{R}^n$, $X = \text{diag}(x_1, \dots, x_n)$ et $e = (1, \dots, 1)^\top \in \mathbb{R}^n$. En écrivant les conditions KKT, on a d'après le Théorème 2.11 (étendu au cas des fonctions convexes prenant la valeur $+\infty$) que x_μ^* est solution de (PPL μ) si et seulement si, il existe $\lambda \in \mathbb{R}^m$ tel que

$$\begin{aligned} Ax_\mu^* &= b \\ 0 &= \partial_x L_\mu(x_\mu^*, \lambda) \text{ c\`ad } A^\top \lambda + \mu(X_\mu^*)^{-1}e = c \text{ et } x_\mu^* > 0 \end{aligned}$$

où $X_\mu^* = \text{diag}((x_\mu^*)_1, \dots, (x_\mu^*)_n)$. En posant $s = \mu(X_\mu^*)^{-1}e$, on peut réécrire les conditions KKT précédentes comme :

$$\begin{aligned} Ax_\mu^* &= b, x_\mu^* > 0 \\ A^\top \lambda + s &= c \\ X_\mu^* s &= \mu e. \end{aligned} \quad (3.2)$$

On peut mettre ces conditions en parallèle avec les conditions KKT du problème initial (PPL) (cf. Section 2.3) : il existe s, λ tels que

$$\begin{aligned} Ax^* &= b, x^* \geq 0, s \geq 0 \\ A^\top \lambda + s &= c \\ X^* s &= 0 \end{aligned} \quad (3.3)$$

qui sont identiques aux précédentes quand $\mu \downarrow 0$ sauf que la condition " $x \geq 0$ " remplace " $x > 0$ " (solution intérieure). On remarque que la complementary slackness condition (3.3) est la seule condition KKT impactée par l'introduction du paramètre μ dans les problèmes annexes (PPL μ).

L'écriture des conditions KKT permet aussi de voir que quand $\mu \downarrow 0$, une solution de (PPL μ) converge vers une solution de (PPL) et que les solutions de (PPL μ) pour $\mu > 0$ se situe bien à l'intérieur de l'ensemble des contraintes contrairement aux solutions de (PPL) qui vérifient la complementary slackness condition (3.3) pour lesquelles soit le multiplicateur de Lagrange est nul soit la contrainte est active.

On souhaite maintenant résoudre le sous-problème de barrière (PPL μ). On va utiliser la méthode de Newton pour les problèmes sous contraintes. En tout point x , on fait un développement de second ordre de la fonction objectif $B(\cdot, \mu)$:

$$B(x + p, \mu) = B(x, \mu) + \langle \nabla_x B(x, \mu), p \rangle + \frac{1}{2} p^\top \nabla_{x,x}^2 B(x, \mu) p + o(\|p\|_2^2)$$

où on calcul le gradient et la Hessienne de $x \rightarrow B(x, \mu)$ en $x > 0$:

$$\nabla_x B(x, \mu) = c - \mu X^{-1} e := g_x \text{ et } \nabla_{x,x}^2 B(x, \mu) = \mu X^{-2} := H_x \quad (3.4)$$

Les directions de descente en x pour la méthode de Newton sont donc solution du problème d'optimisation :

$$\min_p \left(\langle g_x, p \rangle + \frac{1}{2} p^\top H_x p : A(x + p) = b \right). \quad (3.5)$$

On ne cherchera les directions de descente que pour des points x qui sont dans l'ensemble de contraintes. On aura alors $Ax = b$ et on cherchera alors les directions p pour lesquelles $Ap = 0$. On souhaite alors résoudre le problème d'optimisation :

$$\min_p \left(\langle g_x, p \rangle + \frac{1}{2} p^\top H_x p : Ap = 0 \right) \quad (3.6)$$

pour lequel les conditions de Slater sont vérifiées (car A est de rang m), et donc par les conditions KKT : p_x^* est solution de (3.6) si et seulement si il existe $\lambda \in \mathbb{R}^m$ tel que

1. $Ap_x^* = 0$
2. $\nabla_p L(p_x^*, \lambda) = 0$ càd $g_x + H_x p_x^* + A^\top \lambda = 0$ càd $p_x^* = H_x^{-1} [-g_x - A^\top \lambda] = x + \frac{X^2}{\mu} [-A^\top \lambda - c]$.

De plus, A est de rang m et X est définie positive car $x > 0$ donc AX^2A^\top est une matrice inversible. L'équation suivante (obtenue en multipliant la deuxième équation KKT par μA), en $\lambda \in \mathbb{R}^m$,

$$AX^2A^\top \lambda = A[\mu x - X^2 c] \quad (3.7)$$

admet une unique solution, notée λ^* . Pour cette solution $p_x^* = x + \mu^{-1} X^2 [-A^\top \lambda^* - c]$ est une direction de descente solution du problème (3.5).

On en déduit l'algorithme de Newton suivant.

input : $x_0 \in (\mathbb{R}_+)^n$ tel que $Ax_0 = b$; μ_0 et $0 < \rho < 1$.

output: une solution approchante au problème (PPL)

1 **while** critère d'arrêt **do**

2 $\mu_k = \rho \mu_k$

3 $X_k = \text{diag}(x_k)$

4 Résoudre le système : $\lambda_k \in \mathbb{R}^n$ tel que

$$AX_k^2 A^\top \lambda_k = A[\mu_k x_k - X_k^2 c]$$

Calculer la direction de descente : $p_k = x_k + \mu_k^{-1} X_k^2 [-A^\top \lambda_k - c]$

5 $x_k = x_k + p_k$

6 **end**

Algorithm 4: Algorithme de point intérieur 1 : algorithme de Newton primal sur les sous-problèmes de barrière.

Remarque 3.2. On peut aussi inclure un step size dans l'étape 5 de l'Algorithme 4. On fait alors la mise à jour $x_k = x_k + \alpha_k p_k$ où α_k minimise $\alpha \rightarrow B(x_k + \alpha p_k, \mu_k)$.

On peut trouver un résultat sur la convergence de l'Algorithme 4 dans [1].

4 Un algorithme de barrière primal-dual en PL

L'algorithme présenté dans la section précédente est une séquence d'algorithme de descente pour des directions de descentes obtenues par une approximation de second ordre de $B(\cdot, \mu_k)$, la fonction objectif du problème de barrière pour le paramètre μ_k . La direction de descente est donc obtenue uniquement à partir du problème primal auquel a été ajouté une barrière.

Dans ce chapitre, nous utilisons la même méthode de barrière que précédemment et une étape de descente pour résoudre les sous-problèmes de barrière associés mais les directions de descentes sont obtenues en tenant compte du problème dual. Cette méthode est celle implémentée dans la méthode `conelp` de CVXOPT.

On rappelle maintenant les problèmes primal et dual :

$$\textbf{Primal} : \min_x \langle c, x \rangle \text{ tel que } Ax = b, x \geq 0 \quad (\text{PPL})$$

$$\textbf{Dual} : \max_{s, \lambda} \langle b, \lambda \rangle \text{ tel que } A^\top \lambda + s = c, s \geq 0 \quad (\text{DPL})$$

où $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ et $b \in \mathbb{R}^m$. On suppose que A est de rang m .

D'après le Théorème 2.12, x^* est solution du problème primal (PPL) et (s^*, λ^*) est solution du problème dual (DPL) si et seulement si les conditions KKT sont satisfaites :

$$\begin{aligned} Ax^* &= b, x^* \geq 0, s^* \geq 0 \\ s_i^* x_i^* &= 0, \forall i = 1, \dots, n \\ A^\top \lambda^* + s^* &= c. \end{aligned}$$

On peut mettre une barrière logarithmique dans le problème primal (PPL) comme dans la section précédente, (optionnellement) calculer le problème dual associé, et les conditions KKT. On obtient alors :

$$\textbf{Primal} : \min_x \left(\langle c, x \rangle - \mu \sum_{i=1}^n \log x_i \text{ tel que } Ax = b \right) \quad (\text{PPL}\mu)$$

$$\textbf{Dual} : \max_{\lambda} \left(\langle b, \lambda \rangle - n\mu \log \left(\frac{e}{\mu} \right) + \mu \sum_{i=1}^n \log (c_i - \langle \lambda, A_{\cdot i} \rangle) \right) \quad (\text{DPL}\mu)$$

où $A_{\cdot i}$ est la i -ième colonne de A .

On a d'après les conditions KKT : pour tout $\mu > 0$, x^* est solution de (PPL μ) et λ^* est solution de (DPL μ) si et seulement si (x^*, λ^*) vérifie les conditions KKT :

$$\begin{aligned} Ax^* &= b, x^* > 0 \\ s_i^* x_i^* &= \mu, \forall i = 1, \dots, n \\ A^\top \lambda^* + s^* &= c. \end{aligned}$$

Ces conditions KKT sont très similaires à celle du problème original sauf pour la complementary slackness condition (et donc x^* est forcément à l'intérieur de l'ensemble de contraintes).

L'originalité de l'approche primal-dual est de regarder les conditions d'égalité de KKT comme un problème de recherche de zéro d'une fonction. On pose

$$F_\mu(x, \lambda, s) = \begin{pmatrix} A^\top \lambda + s - c \\ Ax - b \\ Xs - \mu e \end{pmatrix}$$

On voit donc que (x^*, λ^*) vérifie les conditions KKT si $x^* > 0$ et $F_\mu(x^*, \lambda^*, s^*) = 0$ pour un certain s^* .

Une idée est alors de résoudre l'équation $F(x^*, \lambda^*, s^*) = 0$. Pour cela, on utilise l'algorithme de Newton (cf. Section 2.4.1) ; On calcul les directions de descente p en un point (x, λ, s) solutionnant l'équation

$$F_\mu(x, \lambda, s) + JF_\mu(x, \lambda, s)p = 0 \quad (4.1)$$

où $JF_\mu(x, \lambda, s)$ est la Jacobienne de F_μ en (x, λ, s) :

$$JF_\mu(x, \lambda, s) = \begin{pmatrix} 0 & A^\top & I_n \\ A & 0 & 0 \\ S & 0 & X \end{pmatrix}$$

où $S = \text{diag}(s) \in \mathbb{R}^{n \times n}$.

Une fois les directions de descente calculées – en résolvant le système d'équations linéaires (4.1) – on peut faire une descente de Newton au point courant. Comme précédemment, on fait décroître le paramètre μ vers 0 à chaque itération. On obtient ainsi le deuxième algorithme de point intérieur qui, contrairement au premier, utilise les variables du problème dual pour calculer les directions de descente. En particulier, on ne fait pas d'approximation de Taylor sur $B(\cdot, \mu)$ mais sur F_μ qui ne contient qu'un seul terme non linéaire " Xs ".

input : $x_0 \in (\mathbb{R}_+)^n$ tel que $Ax_0 = b$; $\lambda_0 \in \mathbb{R}^m$, $s_0 \in (\mathbb{R}_+)^n$; μ_0 et $0 < \rho < 1$.

output: une solution approchante au problème (PPL)

1 **while** critère d'arrêt **do**

2 $\mu_k = \rho \mu_k$

3 Calculer une direction de descente : $p_k \in \mathbb{R}^{n+m+n}$ tel que

$$F_{\mu_k}(x_k, \lambda_k, s_k) + JF_{\mu_k}(x_k, \lambda_k, s_k)p_k = 0$$

4 On effectue une étape de descente :

$$\begin{pmatrix} x_k \\ \lambda_k \\ s_k \end{pmatrix} = \begin{pmatrix} x_k \\ \lambda_k \\ s_k \end{pmatrix} + p_k$$

5 **end**

6 **return** x_k

Algorithm 5: Algorithme de point intérieur 2 : algorithme de Newton primal-dual sur les sous-problèmes de barrière.

L'algorithme primal de point intérieur a une complexité algorithmique en $\mathcal{O}(n^{7/2})$ (cf. [1]) qui est bien polynomial (contrairement à la complexité exponentielle (en m) de l'algorithme du simplexe) mais qui reste cependant très grand quand n est grand. L'algorithme de point intérieur primal-dual est lui en $\mathcal{O}(\sqrt{m+n} \log(1/\epsilon))$ pour une précision en ϵ (cf. [4]).

Remarque 4.1. *On voit apparaître un certain compromis entre la méthode du simplexe pour laquelle chaque itération est très rapide mais qui peu nécessiter un grand nombre d'itérations. Alors que les méthodes de points intérieurs peuvent avoir des itérations coûteuses : le calcul des directions de descente ; mais nécessite qu'un petit nombre d'itérations.*

5 Critères d'arrêt

Les méthodes de points intérieur peuvent tourner indéfiniment car elle ne peuvent jamais être égale à la solution exacte au problème (PPL) étant donné que la suite des itérées reste toujours à l'intérieur grâce à l'effet de barrière. Il faut donc choisir un critère de sortie de la boucle. Il y a plusieurs choix possibles.

On peut par exemple, utiliser la distance entre deux itérés successifs : $\|x_{k+1} - x_k\|_2$. Mais dans les deux algorithmes présentés, on utilise de la dualité Lagrangienne et les conditions KKT qui peuvent fournir des critères d'arrêt.

Pour le premier algorithme de Newton primal, on peut utiliser les conditions d'optimalité KKT du problème de barrière (PPL μ) : x_μ^* est solution de (PPL μ) si et seulement si, il existe $\lambda \in \mathbb{R}^m$ tel que

$$\begin{aligned} Ax_\mu^* &= b \\ (X_\mu^*)(A^\top \lambda - c) + \mu e &= 0 \end{aligned}$$

où $X_\mu^* = \text{diag}((x_\mu^*)_1, \dots, (x_\mu^*)_n)$.

On peut alors utiliser le vecteur de résidus

$$r_k = X_k(A^\top \lambda_k - c) + \mu_k e$$

comme base d'un critère d'arrêt : l'algorithme s'arrête quand $\|r_k\|_2$ passe sous un certain seuil donné a priori – et quand μ_k est assez proche de zéro.

Pour le deuxième algorithme primal-dual, on utilise un critère d'arrêt très classique : le duality gap. On sait qu'une solution est optimale pour le primal et le dual en un point d'annulation du duality gap. Pour le problème initial (PPL) le duality gap est

$$\langle c, x \rangle - \langle b, \lambda \rangle.$$

Cette quantité est toujours positive pour des points x et λ dans leurs ensemble de contraintes et elle est nulle si et seulement si x est solution du primal et λ est solution du dual.

Une idée naturelle est alors d'arrêter l'algorithme quand le duality gap est proche de zéro. Dans le cadre du deuxième algorithme, on calcul par descente successive une approximation du zéro de la fonction F_μ qui caractérise les conditions KKT du problème de barrière. A l'optimal, on a

$$\langle s, x \rangle = \langle c - A^\top \lambda, x \rangle = \langle c, x \rangle - \langle \lambda, Ax \rangle = \langle c, x \rangle - \langle b, \lambda \rangle$$

de plus $\langle x, s \rangle = n\mu$. Donc $n\mu$ est égal au duality gap à l'optimum. Un critère d'arrêt est donc donner par $n\mu$ petit et $\langle s, x \rangle - n\mu$ petit.

Références

- [1] Philip E. Gill, Walter Murray, Michael A. Saunders, J. A. Tomlin, and Margaret H. Wright. On projected Newton barrier methods for linear programming and an equivalence to Karmarkar's projective method. *Math. Programming*, 36(2) :183–209, 1986.

- [2] L. G. Hačijan. A polynomial algorithm in linear programming. *Dokl. Akad. Nauk SSSR*, 244(5) :1093–1096, 1979.
- [3] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4) :373–395, 1984.
- [4] James Renegar. A polynomial-time algorithm, based on Newton’s method, for linear programming. *Math. Programming*, 40(1, (Ser. A)) :59–93, 1988.
- [5] R. Tyrrell Rockafellar. *Convex analysis*. Princeton Mathematical Series, No. 28. Princeton University Press, Princeton, N.J., 1970.
- [6] Daniel Spielman and Shang-Hua Teng. Smoothed analysis of algorithms : why the simplex algorithm usually takes polynomial time. In *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, pages 296–305 (electronic). ACM, New York, 2001.