



THESIS / THÈSE

MASTER EN SCIENCES MATHÉMATIQUES

Une méthode de point proximal pour la minimisation d'une fonction DC

Simonet, Christophe

Award date:
2006

[Link to publication](#)

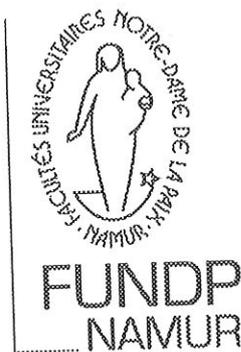
General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

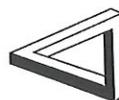
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



Faculté des Sciences
Département de Mathématique

Rempart de la Vierge, 8
B - 5000 Namur (Belgique)

Une méthode de point proximal pour la minimisation d'une fonction DC



Mémoire présenté pour l'obtention
du grade de
Licencié en Sciences Mathématiques
par

SIMONET Christophe

Promoteur : STRODIOT Jean-Jacques

Année Académique 2005-2006

Je remercie tout d'abord Jean-Jacques Strodiot, promoteur de ce mémoire, de m'avoir encadré tout au long de cette année et d'avoir permis la réalisation de ce travail.

Je remercie également toutes les personnes qui, de près ou de loin, m'ont aidé et soutenu durant mes quatre années universitaires.

Une méthode de point proximal pour la minimisation d'une fonction DC

Résumé

Ce mémoire consiste en l'analyse théorique approfondie des méthodes de point proximal pour la résolution de problèmes d'optimisation non convexes et plus particulièrement de problèmes de programmation DC. L'objectif est d'étudier les propriétés de convergence d'un algorithme de point proximal approximé pour minimiser les fonctions DC et de suggérer un schéma algorithmique de calcul concret pour le point proximal. La technique utilisée est celle de la méthode faisceau en optimisation convexe. Notre analyse permet également de traiter la méthode exacte.

A proximal point method for the minimization of a DC function

Abstract

In this work, a theoretical analysis of proximal point methods is done for solving nonconvex optimization problems and more specifically DC programming problems. The aim is to study convergence properties of an approximate proximal point algorithm for minimizing DC functions and to suggest an algorithmic scheme for computing the proximal point. The method used in this work is the bundle method well-known in convex optimization. Our analysis also allows us to consider the exact scheme.

Table des matières

Introduction	1
1 Fonctions convexes	4
1.1 Définitions	4
1.2 Fonctions convexes spéciales	7
1.2.1 Fonctions affines	7
1.2.2 Fonctions convexes fermées	8
1.3 Supremum de fonctions convexes	9
1.4 Fonctions conjuguées	10
2 Sous-différentiabilité	11
2.1 Introduction	11
2.2 Dérivée directionnelle	12
2.3 Sous-différentiel	12
3 Le sous-différentiel approximé	18
3.1 Définition	18
3.2 Propriétés	20
3.3 La dérivée directionnelle approximée	22
3.4 Méthodes de descente à ϵ près	23
4 Programmation DC	25
4.1 Introduction	25
4.2 Fonctions DC	27
4.3 Problèmes de programmation DC	29
4.4 Modèles de problèmes de programmation DC	30
4.4.1 Problème de programmation avec des contraintes de complémentarité	30
4.4.2 Problèmes d'emplacement de pont	31

TABLE DES MATIÈRES

4.4.3	Problème de placement	32
4.4.4	Optimisation sur des ensembles efficaces	32
4.5	Conditions d'optimalité et dualité	33
4.5.1	Définitions et notations	33
4.5.2	Conditions d'optimalité	34
4.5.3	Dualité en programmation DC	37
4.6	Algorithme DC pour les programmes DC généraux	41
4.6.1	Rappels	41
4.6.2	Algorithme DC	42
5	Méthode de point proximal	45
5.1	Introduction	45
5.2	Régularisée Moreau-Yosida	46
5.3	Algorithme de point proximal	54
5.4	Convergence de la méthode de point proximal	56
6	Méthode de point proximal pour la programmation DC	57
6.1	Introduction	57
6.2	Algorithme de point proximal	58
6.3	Convergence de l'algorithme de point proximal	61
6.4	Quelques applications	63
6.4.1	Application à la programmation concave	63
6.4.2	Application à la valeur propre maximale	65
7	Méthode faisceau pour la programmation DC	67
7.1	Méthode de point proximal approximée	67
7.1.1	Un algorithme général	68
7.1.2	Construction des approximations de g	69
7.2	Algorithme de point proximal approximé	76
7.3	Une méthode proximale faisceau pour les fonctions DC	83
	Conclusion	90

Introduction

Les problèmes d'optimisation non convexes que nous allons traiter dans ce mémoire sont du type

$$\begin{cases} \min & f(x) \equiv g(x) - h(x) \\ \text{s.c.} & x \in \mathbb{R}^n, \end{cases}$$

où la fonction objectif $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est une fonction non convexe et les fonctions $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ et $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ sont propres, convexes et semi-continues inférieurement satisfaisant $\text{dom } g \cap \text{dom } h \neq \emptyset$. Dans ce cas, la fonction f est appelée une fonction DC (i.e., différence de deux fonctions convexes) et le problème d'optimisation ci-dessus un problème de programmation DC. La classe des fonctions DC contient notamment les fonctions convexes, concaves et les fonctions de classe \mathcal{C}^2 et constitue une extension de la classe des fonctions convexes. Elle est utilisée dans de nombreuses applications non convexes telles que la finance, la biologie moléculaire, les réseaux de distribution, les processus de restauration d'images et semble particulièrement bien convenir pour modéliser plusieurs problèmes industriels non convexes.

Nous allons étudier des méthodes permettant de résoudre de tels problèmes. Ces techniques de résolution sont appelées méthodes de point proximal. L'algorithme de point proximal a été introduit par Martinet [8] dans les années 1970 pour résoudre les problèmes de minimisation convexes propres semi-continus inférieurement et étudié de plus ample façon par Rockafellar [12] en 1976 dans le contexte des inégalités variationnelles monotones. Il est bien connu que si nous laissons tomber l'hypothèse de convexité sur la fonction objectif, plusieurs problèmes surgissent. L'application proximale n'est pas bien définie et, en général, difficile à calculer même dans des voisinages arbitrairement petits de minima. Jusqu'à présent, seulement quelques recherches ont été proposées

concernant la construction de solutions dans ce cas non convexe (voir par exemple [10]). La méthode présentée dans ce mémoire est basée sur les articles de Sun, Sampaio et Candido [15] et de Moudafi [9]. Pour trouver un point critique de $f := g - h$, elle consiste à augmenter la fonction h le long de la direction d'un sous-gradient et ensuite à diminuer la fonction g grâce à un pas proximal. Nous montrons que si la suite engendrée par cet algorithme est bornée, alors tout point limite de celle-ci est un point critique de f . L'article [9] propose une preuve exacte et élémentaire du résultat de convergence pour la forme approximée de l'algorithme de point proximal et fournit des conditions assurant le caractère borné des suites engendrées. Il suggère et analyse ensuite un algorithme pratique obtenu en couplant la méthode proposée par Sun et al. dans [15] avec les approximations de plan sécant.

La méthode présentée dans ces articles est basée sur l'algorithme de point proximal (voir par exemple [12]). Cette classe d'algorithmes trouve un zéro d'un opérateur monotone maximal T au moyen de l'itération suivante :

$$x^{k+1} = (I + c_k T)^{-1} x^k,$$

où $c_k > c > 0$. L'opérateur $P_k = (I + c_k T)^{-1}$, appelé la résolvente de T , est univoque sur l'espace entier et Lipschitz continu. Quand T est le sous-différentiel d'une fonction g convexe semi-continue inférieurement, i.e., $T = \partial g$, l'itération ci-dessus devient

$$x^{k+1} = \arg \min \left\{ g(x) + \frac{1}{2c_k} \|x - x^k\|^2 \right\}.$$

Rockafellar [12] [13] a développé une étude détaillée de la convergence de l'algorithme de point proximal. En particulier, l'algorithme converge linéairement au moins. Si $c_k \rightarrow \infty$, la convergence est superlinéaire. De plus, des versions approximées de l'algorithme de point proximal sont établies dans [7] et [12].

Utilisant cette stratégie, nous proposons un nouvel algorithme de descente pour trouver un point critique d'une fonction DC qui satisfait les conditions nécessaires d'optimalité. Chaque itération combine un pas sous-gradient ascendant sur la seconde fonction avec un pas proximal sur la première fonction.

Ce mémoire est avant tout théorique. Il a pour objectif principal d'établir un algorithme faisceau pour la résolution de problèmes de programmation DC et d'étudier ses propriétés de convergence. Pour ce faire, nous allons d'abord expliciter la méthode de point proximal dans le cadre général de la programmation convexe et ensuite nous développerons les méthodes proposées par Sun et al. [15] et Moudafi [9] pour la résolution de problèmes de programmation DC.

Chapitre 1

Fonctions convexes

Dans ce premier chapitre, nous allons rappeler toute une série de définitions et de résultats bien connus en analyse convexe. Ces notions seront d'application tout au long de ce mémoire.

1.1 Définitions

Nous commençons par rappeler les définitions d'un ensemble convexe C et d'une fonction convexe f sur C .

Définition 1.1.1.

(1) Soit C un sous-ensemble non vide de \mathbb{R}^n . C est convexe si

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

(2) Soit C un sous-ensemble convexe non vide de \mathbb{R}^n . Une fonction $f : C \rightarrow \mathbb{R}$ est convexe sur C si

$$\forall x, y \in C, \forall \alpha \in]0, 1[\quad f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$

De même, nous donnons les définitions d'une fonction f strictement convexe, fortement convexe et concave sur C .

1.1 Définitions

Définition 1.1.2.

(1) Soit C un sous-ensemble convexe non vide de \mathbb{R}^n . Une fonction $f : C \rightarrow \mathbb{R}$ est strictement convexe sur C si

$$\forall x, y \in C, x \neq y, \forall \alpha \in]0, 1[\quad f(\alpha x + (1 - \alpha)y) < \alpha f(x) + (1 - \alpha)f(y).$$

(2) Soit C un sous-ensemble convexe non vide de \mathbb{R}^n . Une fonction $f : C \rightarrow \mathbb{R}$ est fortement convexe sur C de module $\mu > 0$ si $f - (\mu/2) \|\cdot\|^2$ est convexe sur C .

(3) f est concave (strictement concave, fortement concave) sur C si $-f$ est convexe (strictement convexe, fortement convexe) sur C .

Rappelons à présent quelques propriétés bien connues des fonctions convexes.

Quand f est de classe \mathcal{C}^1 sur un ensemble ouvert contenant C , alors

$$f \text{ est convexe sur } C \iff \forall x, y \in C \quad f(y) \geq f(x) + \nabla f(x)^T(y - x),$$

où $\nabla f(x)$ correspond au gradient de f en x .

Quand f est de classe \mathcal{C}^2 sur un ensemble ouvert contenant C , alors

1. f est convexe sur $C \iff \nabla^2 f$ est semi-définie positive sur C .
2. f est strictement convexe sur $C \iff \nabla^2 f$ est définie positive sur C .
3. f est fortement convexe (de module $\mu > 0$) sur $C \iff \nabla^2 f - \mu I$ est semi-définie positive sur C .

Ici $\nabla^2 f(x)$ dénote la matrice Hessienne de f en x . Rappelons qu'une matrice carrée symétrique A est semi-définie positive (resp. définie positive) si toutes les valeurs propres de A sont positives (resp. strictement positives).

Dans ce travail nous considérerons une classe spéciale de problèmes d'optimisation non convexes :

$$\begin{cases} \min & f(x) \\ \text{s.c.} & x \in \mathbb{R}^n \end{cases} \quad (1.1.1)$$

1.1 Définitions

où $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est une fonction non convexe. Dans de nombreuses situations, la fonction non convexe f peut être traitée comme une différence de deux fonctions convexes

$$f(x) = g(x) - h(x) \quad \forall x \in \mathbb{R}^n, \quad (1.1.2)$$

où $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ et $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ sont propres, convexes et semi-continues inférieurement. Dans ce cas, la fonction f est appelée une fonction DC. Nous définirons les termes utilisés pour décrire ce problème dans la suite de ce chapitre.

Par la suite, nous considérerons des fonctions f avec de possibles valeurs infinies. Pour ce type de fonctions, nous introduisons les définitions suivantes :

Définition 1.1.3. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$. Le domaine de f est l'ensemble

$$\text{dom } f = \{x \in \mathbb{R}^n \mid f(x) < +\infty\}.$$

La fonction f est propre si $\text{dom } f$ est non vide.

Définition 1.1.4. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ une fonction propre. L'épigraphe de f est l'ensemble non vide

$$\text{epi } f = \{(x, r) \in \mathbb{R}^n \times \mathbb{R} \mid r \geq f(x)\}.$$

f est dite convexe si son épigraphe est un sous-ensemble convexe de $\mathbb{R}^n \times \mathbb{R}$. Nous notons par $\text{Conv } \mathbb{R}^n$ l'ensemble des fonctions convexes propres définies sur \mathbb{R}^n .

Nous avons immédiatement que

$$\begin{aligned} f \in \text{Conv } \mathbb{R}^n &\implies \text{dom } f \text{ est non vide et convexe;} \\ f \in \text{Conv } \mathbb{R}^n &\iff f \text{ est convexe sur } \text{dom } f. \end{aligned}$$

1.2 Fonctions convexes spéciales

Exemple 1.1.1. La fonction indicatrice

Soit C un sous-ensemble non vide de \mathbb{R}^n . La fonction indicatrice de C est définie par

$$I_C : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\} \quad I_C(x) = \begin{cases} 0 & \text{si } x \in C \\ +\infty & \text{sinon.} \end{cases}$$

La fonction indicatrice I_C est une fonction propre dont le domaine est C . Puisque $\text{epi } I_C = C \times \mathbb{R}^+$, nous avons que I_C est convexe si et seulement si C est convexe. De plus,

$$\left. \begin{array}{l} f \in \text{Conv } \mathbb{R}^n \\ C \subseteq \mathbb{R}^n \text{ non vide et convexe} \\ \text{dom } f \cap C \neq \emptyset \end{array} \right\} \implies f + I_C \in \text{Conv } \mathbb{R}^n.$$

1.2 Fonctions convexes spéciales

Dans cette section, nous définissons deux classes importantes de fonctions convexes : les fonctions affines et les fonctions convexes fermées.

1.2.1 Fonctions affines

Définition 1.2.1. Soit $a \in \mathbb{R}^n$ et $b \in \mathbb{R}$. La fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ définie par $f(x) = \langle a, x \rangle - b$ pour tout $x \in \mathbb{R}^n$, est appelée une fonction affine sur \mathbb{R}^n . Si $b = 0$, alors $f(x) = \langle a, x \rangle$ est dite linéaire.

Les fonctions affines jouent un rôle important en analyse convexe comme le montre le résultat suivant.

Proposition 1.2.1. Toute fonction $f \in \text{Conv } \mathbb{R}^n$ est minorée par une fonction affine.

Notons que f est minorée par une fonction affine signifie qu'il existe $a \in \mathbb{R}^n$ et $r \in \mathbb{R}$ tels que

$$f(x) \geq \langle a, x \rangle - r \quad \forall x \in \mathbb{R}^n.$$

1.2 Fonctions convexes spéciales

Quand f est convexe et différentiable sur \mathbb{R}^n , alors $\text{dom } f = \mathbb{R}^n$ et nous savons dès lors que f est minorée par une fonction affine :

$$f(x) \geq \langle \nabla f(x_0), x - x_0 \rangle + f(x_0) \quad \forall x \in \mathbb{R}^n,$$

où x_0 est un point dans \mathbb{R}^n .

1.2.2 Fonctions convexes fermées

Nous savons que pour qu'une fonction ait un minimum, nous avons besoin qu'elle soit semi-continue inférieurement (s.c.i.).

Définition 1.2.2. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ une fonction propre. f est dite s.c.i. en $x \in \mathbb{R}^n$ si

$$\liminf_{y \rightarrow x} f(y) \geq f(x) \quad [\iff \forall \{y_k\} \rightarrow x \quad \liminf_{k \rightarrow \infty} f(y_k) \geq f(x)]$$

En termes d'épigraphe, nous avons la caractérisation suivante d'une fonction s.c.i. :

Proposition 1.2.2. Pour $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, les deux propriétés suivantes sont équivalentes :

- (i) f est s.c.i. sur \mathbb{R}^n (i.e. en chaque point de \mathbb{R}^n);
- (ii) $\text{epi } f$ est un sous-ensemble fermé de $\mathbb{R}^n \times \mathbb{R}$.

Le résultat suivant est bien connu.

Proposition 1.2.3. (Théorème de Weierstrass) Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ une fonction propre s.c.i. et soit C un sous-ensemble fermé borné de \mathbb{R}^n tels que $\text{dom } f \cap C \neq \emptyset$. Alors f atteint sa borne inférieure sur C , i.e., il existe $u \in C$ tel que $f(u) = \inf_{x \in C} f(x)$.

1.3 Supremum de fonctions convexes

Définition 1.2.3. Une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est dite fermée si son épigraphe est un sous-ensemble fermé de $\mathbb{R}^n \times \mathbb{R}$.

Par la proposition 1.2.2, nous avons immédiatement

$$f \text{ est fermée} \iff f \text{ est s.c.i.}$$

Par la suite, nous noterons par $\Gamma_0(\mathbb{R}^n)$ l'ensemble des fonctions convexes propres fermées sur \mathbb{R}^n .

Exemple 1.2.1.

1. Toute fonction affine f appartient à $\Gamma_0(\mathbb{R}^n)$ parce que $\text{epi } f$ est fermé et convexe et $\text{dom } f$ est non vide.
2. Soit C un sous-ensemble non vide de \mathbb{R}^n . Puisque $\text{epi } I_C = C \times \mathbb{R}^+$, nous avons

$$I_C \in \Gamma_0(\mathbb{R}^n) \iff C \text{ est fermé et convexe.}$$

1.3 Supremum de fonctions convexes

Nous donnons dans cette section un exemple de fonction supremum qui sera utilisée dans le chapitre 7 lors de la description de la méthode de point proximal approximée.

Définition 1.3.1. Fonctions affines par morceaux

Soit $(s_1, b_1), \dots, (s_m, b_m)$ m éléments de $\mathbb{R}^n \times \mathbb{R}$. La fonction $\hat{f} : \mathbb{R}^n \rightarrow \mathbb{R}$ définie par

$$\hat{f}(x) = \max_{j=1, \dots, m} \{ \langle s_j, x \rangle - b_j \}$$

est appelée affine par morceau. Puisque les m fonctions $\langle s_j, \cdot \rangle - b_j$ sont affines, il suit par la proposition 2.3.1 dans [14] que $\hat{f} \in \Gamma_0(\mathbb{R}^n)$.

1.4 Fonctions conjuguées

Définition 1.4.1. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ une fonction propre minorée par une fonction affine. Alors la fonction $f^* : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ définie par

$$f^*(x) = \sup_{y \in \mathbb{R}^n} \{\langle x, y \rangle - f(y)\}$$

est appelée la fonction conjuguée de f .

La proposition suivante donne les propriétés d'une fonction conjuguée de f .

Proposition 1.4.1. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ une fonction propre minorée par une fonction affine. Alors la fonction conjuguée de f est bien définie et est propre, fermée et convexe.

Observons que f^* est fermée et convexe même si f n'est pas convexe.

Puisque f^* est propre et minorée par une fonction affine (car f^* est convexe), nous pouvons considérer la conjuguée de f^* , notée f^{**} et appelée la fonction biconjuguée de f . Nous pouvons à présent énoncer les propriétés de la fonction biconjuguée de f .

Proposition 1.4.2. Les propriétés suivantes ont lieu :

- (i) Si $f \in \text{Conv } \mathbb{R}^n$, alors $f^{**} \leq f$;
- (ii) Si $f \in \Gamma_0(\mathbb{R}^n)$, alors $f^{**} = f$.

Chapitre 2

Sous-différentiabilité

2.1 Introduction

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ et $x \in \text{int}(\text{dom } f)$. Le but du calcul différentiel est d'approximer f dans un voisinage de x par une fonction linéaire. Rappelons ce qui est déjà connu.

Définition 2.1.1. *La fonction f est différentiable en x si il existe $l_x : \mathbb{R}^n \rightarrow \mathbb{R}$ linéaire telle que*

$$f(x+h) = f(x) + l_x(h) + o(\|h\|).$$

(ici $g(h) = o(\|h\|)$ quand $h \rightarrow 0$ signifie que $\lim_{h \rightarrow 0} \frac{g(h)}{\|h\|} = 0$).

En outre, dans ce cas, l_x est unique et peut être exprimé comme

$$l_x(h) = \langle \nabla f(x), h \rangle,$$

où le vecteur $\nabla f(x)$ est le gradient de f en x .

Quand f est convexe, il n'est pas toujours vrai que f est différentiable. Par exemple, $f(x) = |x|$ n'est pas différentiable en 0.

2.2 Dérivée directionnelle

Définition 2.2.1. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, $x_0 \in \text{dom } f$ et $d \in \mathbb{R}^n$. La dérivée directionnelle de f en x_0 dans la direction d est

$$f'(x_0, d) = \lim_{t \searrow 0} \frac{f(x_0 + td) - f(x_0)}{t}$$

si la limite existe.

La proposition suivante nous donne la propriété principale de la dérivée directionnelle.

Proposition 2.2.1. Soit f une fonction convexe propre et soit $x_0 \in \mathbb{R}^n$. Si $x_0 \in \text{dom } f$, alors $f'(x_0, d) \in \mathbb{R}$ pour tout $d \in \mathbb{R}^n$. En outre,

$$f'(x_0, d) = \inf_{t > 0} \frac{f(x_0 + td) - f(x_0)}{t}.$$

Quand f est différentiable, nous avons le résultat suivant :

Proposition 2.2.2. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ et $x_0 \in \text{int}(\text{dom } f)$. Si f est différentiable en x_0 , alors

- (i) $d \mapsto f'(x_0, d)$ est linéaire ;
- (ii) $f'(x_0, d) = \langle \nabla f(x_0), d \rangle$ pour tout $d \in \mathbb{R}^n$;
- (iii) $f'(x_0, d) = \lim_{t \rightarrow 0} \frac{f(x_0 + td) - f(x_0)}{t}$.

2.3 Sous-différentiel

Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe et soit $x_0 \in \mathbb{R}^n$. Rappelons que pour un sous-ensemble non vide S de \mathbb{R}^n , la fonction de support de

2.3 Sous-différentiel

S est définie par

$$\sigma_S : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\} \quad \sigma_S(x) = \sup_{s \in S} \langle x, s \rangle.$$

Nous savons que la dérivée directionnelle $f'(x_0, \cdot)$ est une fonction sous-linéaire et à valeurs finies. Donc, $f'(x_0, \cdot)$ est le support d'un ensemble convexe compact non vide

$$S = \{s \in \mathbb{R}^n \mid \langle s, d \rangle \leq f'(x_0, d) \quad \forall d \in \mathbb{R}^n\}.$$

Définition 2.3.1. *Le sous-différentiel $\partial f(x)$ de f en x est l'ensemble convexe compact non vide de \mathbb{R}^n dont la fonction de support est $f'(x, \cdot)$, i.e.,*

$$\partial f(x) = \{s \in \mathbb{R}^n \mid \langle s, d \rangle \leq f'(x, d) \quad \forall d \in \mathbb{R}^n\}.$$

Tout vecteur $s \in \partial f(x)$ est appelé un sous-gradient de f en x .

En d'autres termes, nous avons l'importante égalité

$$f'(x, d) = \sup_{s \in \partial f(x)} \langle s, d \rangle.$$

Le sous-différentiel peut aussi être caractérisé par des inégalités. L'équivalence suivante est très importante et sera utilisée à de nombreuses reprises dans la suite de ce travail.

Proposition 2.3.1. *La propriété suivante a lieu :*

$$s \in \partial f(x_0) \iff \forall x \in \mathbb{R}^n \quad f(x) \geq f(x_0) + \langle s, x - x_0 \rangle.$$

Preuve. Nous avons successivement :

$$\begin{aligned} s \in \partial f(x_0) &\iff \langle s, d \rangle \leq f'(x_0, d) \quad \forall d \in \mathbb{R}^n \\ &\iff \langle s, d \rangle \leq \inf_{t>0} \frac{f(x_0 + td) - f(x_0)}{t} \quad \forall d \in \mathbb{R}^n \\ &\iff \langle s, d \rangle \leq \frac{f(x_0 + td) - f(x_0)}{t} \quad \forall t > 0, \forall d \in \mathbb{R}^n \\ &\iff f(x_0 + td) \geq f(x_0) + \langle s, td \rangle \quad \forall t > 0, \forall d \in \mathbb{R}^n \\ &\iff f(x) \geq f(x_0) + \langle s, x - x_0 \rangle \quad \forall x \in \mathbb{R}^n. \end{aligned}$$

□

2.3 Sous-différentiel

Quand f est différentiable, le sous-différentiel est réduit au gradient.

Proposition 2.3.2. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe et soit $x_0 \in \mathbb{R}^n$. Alors

(i) Si f est différentiable en x_0 , alors $\partial f(x_0) = \{\nabla f(x_0)\}$;

(ii) Si $\partial f(x_0) = \{s\}$, alors f est différentiable en x_0 et $\nabla f(x_0) = s$.

Interprétation géométrique

L'inégalité définissant le $s \in \partial f(x_0)$:

$$\forall x \in \mathbb{R}^n \quad f(x) \geq f(x_0) + \langle s, x - x_0 \rangle$$

signifie que s est la pente d'une fonction affine qui minore f et qui passe par le point $(x_0, f(x_0))$.

Pour illustrer cette interprétation, nous considérons la fonction $f(x) = |x|$.

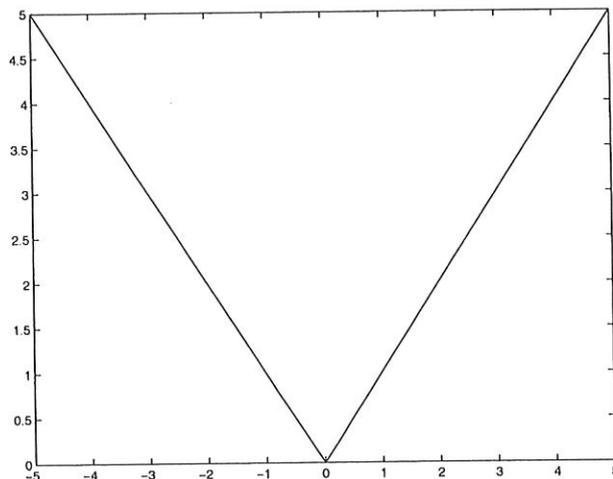


FIG. 2.1 – Graphe de $f(x) = |x|$.

2.3 Sous-différentiel

Il est facile de voir sur le graphe de cette fonction que

$$\partial f(0) = [-1, 1], \quad \partial f(x_0) = \{1\} \text{ si } x_0 > 0 \text{ et } \partial f(x_0) = \{-1\} \text{ si } x_0 < 0.$$

Pour la fonction

$$f(x) = \begin{cases} e^x - 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0, \end{cases}$$

nous obtenons

$$\partial f(0) = [0, 1], \quad \partial f(x_0) = \{e^{x_0}\} \text{ si } x_0 > 0 \text{ et } \partial f(x_0) = \{0\} \text{ si } x_0 < 0.$$

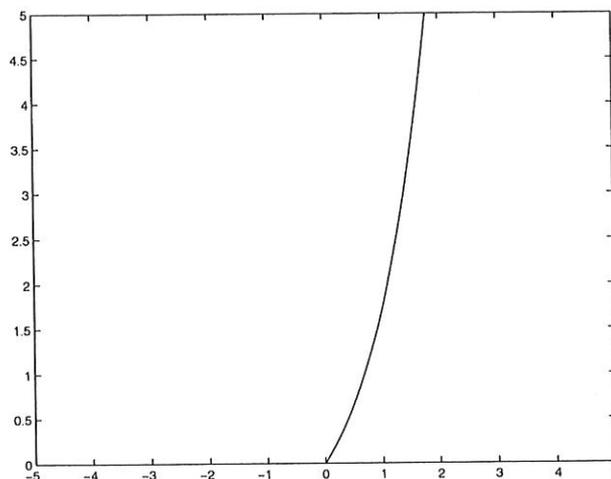


FIG. 2.2 – Graphe de $f(x) = e^x - 1$ si $x \geq 0$.

Les conditions d'optimalité sont présentées dans la proposition suivante.

Proposition 2.3.3. *Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe et soit $x_0 \in \mathbb{R}^n$. Les trois conditions suivantes sont équivalentes :*

- (i) x_0 minimise f sur \mathbb{R}^n ;
- (ii) $0 \in \partial f(x_0)$;
- (iii) $f'(x_0, d) \geq 0$ pour tout $d \in \mathbb{R}^n$.

2.3 Sous-différentiel

Preuve. Par définition du sous-différentiel, il est immédiat que (i) et (ii) sont équivalents. Pour prouver que (i) et (iii) sont équivalents, il suffit d'observer que

$$\begin{aligned} f'(x_0, d) \geq 0 \quad \forall d \in \mathbb{R}^n &\iff \frac{f(x_0 + td) - f(x_0)}{t} \geq 0 \quad \forall d \in \mathbb{R}^n, \forall t > 0 \\ &\iff f(x_0 + td) \geq f(x_0) \quad \forall d \in \mathbb{R}^n, \forall t > 0 \\ &\iff f(x) \geq f(x_0) \quad \forall x \in \mathbb{R}^n. \end{aligned}$$

□

Observons que quand f est différentiable, la condition d'optimalité devient $\nabla f(x_0) = 0$. Pour finir ce chapitre, nous présentons d'autres caractérisations du sous-différentiel basées sur la fonction conjuguée.

Proposition 2.3.4. *Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe et soit $x, y \in \mathbb{R}^n$. Alors les conditions suivantes sont équivalentes :*

- (i) $y \in \partial f(x)$;
- (ii) $f(x) + f^*(y) \leq \langle x, y \rangle$;
- (iii) $f(x) + f^*(y) = \langle x, y \rangle$.

Preuve. Tout d'abord, rappelons que, par définition, $f^*(y) = \sup_z \{ \langle y, z \rangle - f(z) \}$ et donc que

$$\langle x, y \rangle \leq f(x) + f^*(y).$$

Cette inégalité est appelée l'Inégalité de Young-Fenchel.

Alors nous avons, en utilisant successivement la définition du sous-différentiel, la définition de la fonction conjuguée et l'Inégalité de Young-Fenchel,

$$\begin{aligned} y \in \partial f(x) &\iff f(z) \geq f(x) + \langle y, z - x \rangle \quad \forall z \in \mathbb{R}^n \\ &\iff \langle y, x \rangle - f(x) \geq \langle y, z \rangle - f(z) \quad \forall z \in \mathbb{R}^n \\ &\iff \langle y, x \rangle - f(x) \geq f^*(y) \\ &\iff \langle y, x \rangle - f(x) = f^*(y). \end{aligned}$$

□

2.3 Sous-différentiel

Proposition 2.3.5. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe et soit $x, y \in \mathbb{R}^n$. Alors

$$y \in \partial f(x) \iff x \in \partial f^*(y).$$

Preuve. Puisque $f^{**} = f$, nous avons, en utilisant la proposition 2.3.4,

$$\begin{aligned} y \in \partial f(x) &\iff f(x) + f^*(y) \leq \langle x, y \rangle \\ &\iff f^{**}(x) + f^*(y) \leq \langle x, y \rangle \\ &\iff x \in \partial f^*(y). \end{aligned}$$

□

Chapitre 3

Le sous-différentiel approximé

3.1 Définition

Définition 3.1.1. Soit $f \in \Gamma_0(\mathbb{R}^n)$, $\epsilon \geq 0$ et $x \in \text{dom } f$. Un vecteur $s \in \mathbb{R}^n$ est appelé un ϵ -sous-gradient de f en x si

$$\forall y \in \text{dom } f \quad f(y) \geq f(x) + \langle s, y - x \rangle - \epsilon.$$

L'ensemble des ϵ -sous-gradients de f en x est noté $\partial_\epsilon f(x)$ et appelé le ϵ -sous-différentiel de f en x . Par convention, $\partial_\epsilon f(x) = \emptyset$ quand $x \notin \text{dom } f$.

Nous avons immédiatement que

$$\partial_\epsilon f(x) \subseteq \partial_{\epsilon'} f(x) \text{ pour } \epsilon \leq \epsilon'$$

et

$$\partial f(x) = \partial_0 f(x) = \bigcap_{\epsilon > 0} \partial_\epsilon f(x).$$

Exemple 3.1.1. 1. $f(x) = x^2$. Le ϵ -sous-différentiel de f en $x = 0$ est

$$\partial_\epsilon f(0) = [-2\sqrt{\epsilon}, 2\sqrt{\epsilon}].$$

Cet ensemble est réduit au gradient de f en 0 quand $\epsilon = 0$.

3.1 Définition

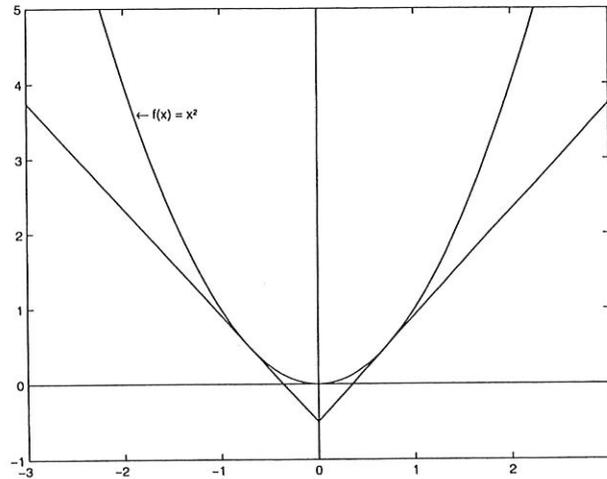


FIG. 3.1 – Graphes de $f(x) = x^2$ et de $\partial_{0,5}f(0)$.

2. $f(x) = |x|$. Le ϵ -sous-différentiel de f en $x = 0$ est

$$\partial_{\epsilon}f(0) = \partial f(0) = [-1, 1].$$

Plus généralement, pour $\epsilon > 0$, nous avons

$$\partial_{\epsilon}f(x) = \begin{cases} [-1, -1 - \epsilon/x] & \text{si } x < -\epsilon/2 \\ [-1, 1] & \text{si } -\epsilon/2 \leq x \leq \epsilon/2 \\ [1 - \epsilon/x, 1] & \text{si } x > \epsilon/2. \end{cases}$$

Observons que $\partial_{\epsilon}f(0) = \partial f(0)$ pour tout $\epsilon > 0$ et que $\partial_{\epsilon}f(x)$ est continu quand $\epsilon > 0$ et que $x \rightarrow 0$ (pour le voir, il est utile de dessiner le graphe de $\partial_{\epsilon}f$).

Exemple 3.1.2. Directions ϵ -normales et cône normal

Soit C un sous-ensemble convexe fermé non vide de \mathbb{R}^n et soit I_C la fonction indicatrice de C . Le domaine de I_C est C et pour chaque $x \in C$, nous avons

$$\begin{aligned} \partial_{\epsilon}I_C(x) &= \{s \in \mathbb{R}^n \mid \forall y \in C \ I_C(y) \geq I_C(x) + \langle s, y - x \rangle - \epsilon\} \\ &= \{s \in \mathbb{R}^n \mid \forall y \in C \ \langle s, y - x \rangle \leq \epsilon\}. \end{aligned}$$

Cet ensemble est appelé l'ensemble des directions ϵ -normales à C en $x \in C$. Quand $\epsilon = 0$, cet ensemble devient

$$\partial I_C(x) = \{s \in \mathbb{R}^n \mid \forall y \in C \ \langle s, y - x \rangle \leq 0\}.$$

3.2 Propriétés

Preuve. En utilisant les définitions du ϵ -sous-différentiel et de la fonction conjuguée, nous avons successivement

$$\begin{aligned} s \in \partial_\epsilon f(x) &\iff \forall y \in \text{dom } f \quad f(y) \geq f(x) + \langle s, y - x \rangle - \epsilon \\ &\iff \forall y \in \text{dom } f \quad f(x) + \{\langle s, y \rangle - f(y)\} - \langle s, x \rangle \leq \epsilon \\ &\iff f(x) + \sup_{y \in \text{dom } f} \{\langle s, y \rangle - f(y)\} - \langle s, x \rangle \leq \epsilon \\ &\iff f(x) + f^*(s) - \langle s, x \rangle \leq \epsilon. \end{aligned}$$

De façon similaire, nous obtenons

$$x \in \partial_\epsilon f^*(s) \iff f^{**}(x) + f^*(s) - \langle s, x \rangle \leq \epsilon.$$

Puisque $f \in \Gamma_0(\mathbb{R}^n)$, nous avons $f^{**} = f$ et la conclusion suit directement. \square

Enfin, le ϵ -sous-différentiel peut être utilisé pour caractériser les solutions optimales à ϵ près (solutions ϵ -optimales).

Proposition 3.2.3. Soit $\epsilon \geq 0$. La condition d'optimalité suivante a lieu :

$$0 \in \partial_\epsilon f(x) \iff \forall y \in \mathbb{R}^n \quad f(x) \leq f(y) + \epsilon.$$

Preuve. Il suffit d'utiliser la définition du ϵ -sous-différentiel pour obtenir le résultat. En effet :

$$\begin{aligned} 0 \in \partial_\epsilon f(x) &\iff f(y) \geq f(x) + \langle 0, y - x \rangle - \epsilon \quad \forall y \in \mathbb{R}^n \\ &\iff f(x) \leq f(y) + \epsilon \quad \forall y \in \mathbb{R}^n. \end{aligned}$$

\square

Une autre façon de considérer l'optimalité approximative est donnée par la proposition suivante.

Proposition 3.2.4. Soit $\epsilon \geq 0$. Si $x \in \mathbb{R}^n$ est tel que

$$\exists s \in \partial_\epsilon f(x) \text{ t.q. } \|s\| \leq \epsilon, \tag{3.2.1}$$

alors

$$\forall y \in \mathbb{R}^n \quad f(y) \geq f(x) - \epsilon \|y - x\| - \epsilon.$$

3.3 La dérivée directionnelle approximée

Preuve. Soit $y \in \mathbb{R}^n$. Puisque $s \in \partial_\epsilon f(x)$, nous avons

$$\begin{aligned} f(y) &\geq f(x) + \langle s, y - x \rangle - \epsilon \\ &\geq f(x) - \|s\| \|y - x\| - \epsilon \end{aligned}$$

où la dernière inégalité a été obtenue en utilisant l'inégalité de Cauchy-Schwarz. Le reste de la preuve est évident. \square

Un point x qui satisfait la propriété (3.2.1) est appelé un ϵ -point stationnaire. Un tel point peut être arbitrairement loin de l'optimum à la fois dans la valeur de la fonction que dans la distance à l'optimum. Cependant, ce concept est utile car il fournit un critère d'arrêt pour les méthodes faisceaux (voir chapitre 7).

3.3 La dérivée directionnelle approximée

Définition 3.3.1. Soit $f \in \Gamma_0(\mathbb{R}^n)$, $x \in \text{dom } f$ et $\epsilon \geq 0$. La fonction de support du ϵ -sous-différentiel de f en x est appelée la ϵ -dérivée directionnelle de f en x et est notée par

$$f'_\epsilon(x, d) = \sup_{s \in \partial_\epsilon f(x)} \langle s, d \rangle.$$

Nous avons immédiatement que la fonction $f'_\epsilon(x, \cdot)$ est fermée et sous-linéaire (et donc convexe). Quand $\epsilon = 0$, nous savons que

$$f'_0(x, d) = f'(x, d) = \lim_{t \searrow 0} \frac{f(x + td) - f(x)}{t} = \inf_{t > 0} \frac{f(x + td) - f(x)}{t}.$$

Quand $\epsilon > 0$, nous avons seulement le résultat suivant.

Proposition 3.3.1. Soit $x \in \text{dom } f$ et $\epsilon > 0$. Alors

$$f'_\epsilon(x, d) = \inf_{t > 0} \frac{f(x + td) - f(x) + \epsilon}{t}.$$

3.4 Méthodes de descente à ϵ près

Géométriquement, le quotient

$$\frac{f(x+td) - f(x) + \epsilon}{t}$$

peut être interprété comme la pente de la ligne droite joignant le point $(0, f(x) - \epsilon)$ au point $(t, f(x+td))$. Donc, par la proposition 3.3.1, $f'_\epsilon(x, d)$ est la plus petite pente de ces lignes droites.

3.4 Méthodes de descente à ϵ près

Dans la dernière section de ce chapitre, nous supposons que $f : \mathbb{R}^n \rightarrow \mathbb{R}$ est une fonction convexe à valeurs finies. Pour présenter la méthode de descente à ϵ près, nous avons besoin de la définition suivante.

Définition 3.4.1. Une direction $d \in \mathbb{R}^n$ est une ϵ -direction de descente en x pour f si

$$\exists t > 0 \text{ t.q. } f(x+td) < f(x) - \epsilon.$$

La proposition suivante donne certaines caractérisations d'une ϵ -direction de descente.

Proposition 3.4.1. Les propriétés suivantes sont équivalentes :

- (i) d est une ϵ -direction de descente en x pour f ;
- (ii) $f'_\epsilon(x, d) < 0$;
- (iii) $\langle s, d \rangle < 0$ pour tout $s \in \partial_\epsilon f(x)$.

Preuve.

(i) \implies (ii). Puisque d est une ϵ -direction de descente, il existe une longueur de pas $\bar{t} > 0$ telle que $f(x + \bar{t}d) < f(x) - \epsilon$. Donc

$$f'_\epsilon(x, d) = \inf_{t>0} \frac{f(x+td) - f(x) + \epsilon}{t} \leq \frac{f(x + \bar{t}d) - f(x) + \epsilon}{\bar{t}} < 0.$$

3.4 Méthodes de descente à ϵ près

D'où (ii) est satisfait.

(ii) \implies (i). Puisque $f'_\epsilon(x, d) < 0$, il existe $\bar{t} > 0$ tel que

$$\frac{f(x + \bar{t}d) - f(x) + \epsilon}{\bar{t}} < 0,$$

i.e., d est une ϵ -direction de descente.

(ii) \iff (iii). Evident car $f'_\epsilon(x, d)$ est la fonction de support de $\partial_\epsilon f(x)$ et cet ensemble est compact. \square

L'existence d'une ϵ -direction de descente est donnée par la proposition suivante.

Proposition 3.4.2. *La propriété suivante a lieu :*

$$0 \notin \partial_\epsilon f(x) \iff \exists d \text{ t.q. } f'_\epsilon(x, d) < 0.$$

Chapitre 4

Programmation DC

4.1 Introduction

Dans le domaine de l'optimisation non convexe, la programmation DC joue un rôle intéressant et important de par ses aspects théoriques ainsi que de son large champ d'applications. Une fonction est appelée DC si elle peut être représentée comme la différence de deux fonctions convexes¹. Les problèmes de programmation mathématique traitant des fonctions DC sont appelés problèmes de programmation DC. Nous savons que l'ensemble des fonctions DC définies sur un ensemble convexe compact de \mathbb{R}^n est dense dans l'ensemble des fonctions continues sur cet ensemble. Par conséquent, en principe, toute fonction continue peut être approximée par une fonction DC avec n'importe quelle précision désirée. En outre, toute fonction de classe \mathcal{C}^2 est une fonction DC. Bien que les représentations DC soient disponibles pour les classes importantes de fonctions, trouver une telle représentation pour une fonction DC arbitraire est un problème en cours assez difficile.

La programmation DC et son algorithme DC (DCA²) abordent le problème de minimiser une fonction $f = g - h$ sur l'espace entier :

$$\alpha = \inf \{f(x) := g(x) - h(x) \mid x \in X\}$$

où $X = \mathbb{R}^n$ est l'espace Euclidien usuel et g, h sont deux fonctions convexes propres s.c.i. sur X (i.e. $g, h \in \Gamma_0(X)$). En réalité, nous vou-

¹DC = Difference of two Convex functions

²DCA = DC Algorithm

4.1 Introduction

drions obtenir une extension de la programmation convexe, pas trop large pour nous permettre d'encore utiliser l'arsenal d'outils puissants en analyse convexe et en optimisation convexe, mais suffisamment large pour couvrir la plupart des problèmes d'optimisation non convexes du monde réel.

Par rapport à l'approche combinatoire où de nombreux algorithmes globaux ont été étudiés, il y a très peu d'algorithmes pour résoudre les programmes DC dans l'approche d'analyse convexe. Ici nous nous intéressons aux conditions d'optimalité locale et globale, aux rapports entre les solutions locale et globale du programme DC primal et de son dual

$$\alpha = \inf \{h^*(y) - g^*(y) \mid y \in Y\}$$

(où Y est l'espace dual de X , qui peut être identifié à X lui-même, et g^* , h^* représentent les fonctions conjuguées de g et h , respectivement) et les algorithmes de solution.

En pratique, l'algorithme DC est appliqué avec succès à un grand nombre de problèmes différents et variés d'optimisation non convexe auxquels il donne assez souvent des solutions globales et prouve qu'il est plus robuste et plus efficace que les méthodes standard rapportées, spécialement pour les problèmes de grande taille. L'algorithme DC est actuellement l'un des rares algorithmes pour la programmation non convexe non différentiable qui permette de résoudre des programmes DC de grande taille.

Nous pouvons affirmer que la plupart des problèmes d'optimisation du monde réel peuvent être formulés comme des programmes DC. Nous pouvons expliquer la richesse de l'ensemble des fonctions DC sur $X = \mathbb{R}^n$, noté par $\mathcal{DC}(X)$:

- $\mathcal{DC}(X)$ est un sous-espace contenant les fonctions de classe \mathcal{C}^2 . En particulier, $\mathcal{DC}(X)$ contient l'espace $\mathcal{C}^{1,1}(X)$ des fonctions dont le gradient est localement Lipschitzien sur X .
- $\mathcal{DC}(X)$ est fermé sous toutes les opérations habituellement considérées en optimisation. En particulier, une combinaison linéaire de fonctions DC appartient à $\mathcal{DC}(X)$, un supremum fini de fonctions DC est DC. De même, le produit de deux fonctions DC reste DC.

4.2 Fonctions DC

Dans cette section, nous rappelons sans les démontrer certaines propriétés clés des fonctions DC.

Définition 4.2.1. Soit C un sous-ensemble convexe de \mathbb{R}^n . Une fonction $f : C \rightarrow \mathbb{R}$ est appelée DC sur C s'il existe deux fonctions convexes $g, h : C \rightarrow \mathbb{R}$ telles que f peut être exprimée sous la forme

$$f(x) = g(x) - h(x). \quad (4.2.1)$$

Si $C = \mathbb{R}^n$, alors f est simplement appelée une fonction DC. Chaque représentation de la forme (4.2.1) est une décomposition DC de f . Nous appelons une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ localement DC si

$\forall x_0 \in \mathbb{R}^n \quad \exists \epsilon > 0$ t.q. f est DC sur la boule

$$B(x_0, \epsilon) = \{x \in \mathbb{R}^n \mid \|x - x_0\| \leq \epsilon\}.$$

Des exemples de fonctions DC incluent les fonctions quadratiques indéfinies sous la forme xQx (pour chaque matrice Q de dimension $n \times n$, il existe deux matrices A et B semi-définies positives de dimension $n \times n$ telles que $xQx = xAx - xBx$), les produits scalaires

$$xy = (1/4)(\|x + y\|^2 - \|x - y\|^2),$$

et le carré de la fonction distance $d_M : \mathbb{R}^n \rightarrow \mathbb{R}$ définie par

$$d_M(x) = \inf \{\|x - y\| \mid y \in M\},$$

où M est un sous-ensemble fermé non vide de \mathbb{R}^n ,

$$\begin{aligned} d_M^2(x) &= \inf \{\|x - y\|^2 \mid y \in M\} \\ &= \|x\|^2 + \inf \{-\|x\|^2 + \|x - y\|^2 \mid y \in M\} \\ &= \|x\|^2 - \sup \{2xy - \|y\|^2 \mid y \in M\}. \end{aligned}$$

La proposition suivante montre que la classe des fonctions DC est fermée sous de nombreuses opérations fréquemment rencontrées en optimisation.

Proposition 4.2.1. *Soit f et f_i , $i = 1, \dots, m$, des fonctions DC. Alors les fonctions suivantes sont aussi DC :*

- (i) $\sum_{i=1}^m \lambda_i f_i(x)$, $\lambda_i \in \mathbb{R}$, $i = 1, \dots, m$,
- (ii) $\max_{i=1, \dots, m} f_i(x)$, $\min_{i=1, \dots, m} f_i(x)$,
- (iii) $|f(x)|$, $f^+(x) := \max\{0, f(x)\}$, $f^-(x) := \min\{0, f(x)\}$,
- (iv) $\prod_{i=1}^m f_i(x)$.

Il est intéressant de noter que les résultats ci-dessus peuvent être prouvés par induction, i.e., pour chacune des fonctions définies ci-dessus, une décomposition DC explicite peut être construite. Par exemple, considérons la fonction

$$f(x) = \max \{f_i(x) = g_i(x) - h_i(x) \mid i = 1, \dots, m\},$$

g_i et h_i étant des fonctions convexes, $i = 1, \dots, m$. Puisque, pour chaque $i = 1, \dots, m$,

$$f_i = g_i + \sum_{\substack{j=1 \\ j \neq i}}^m h_j - \sum_{j=1}^m h_j,$$

nous obtenons la décomposition DC suivante de f :

$$f = g - h, \quad g = \max_{i=1, \dots, m} \left\{ g_i + \sum_{\substack{j=1 \\ j \neq i}}^m h_j \right\}, \quad h = \sum_{j=1}^m h_j. \quad (4.2.2)$$

Un résultat principal concernant la reconnaissance des fonctions DC est donné ci-dessous.

Proposition 4.2.2. *Toute fonction localement DC est DC.*

4.3 Problèmes de programmation DC

Cette proposition mène aux conséquences importantes suivantes.

Proposition 4.2.3.

(i) Toute fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}$ de classe \mathcal{C}^2 est DC.

(ii) Soit C un sous-ensemble convexe compact de \mathbb{R}^n . Alors toute fonction continue sur C est la limite d'une suite de fonctions DC qui converge uniformément sur C ; i.e., pour toute fonction continue $c : C \rightarrow \mathbb{R}$ et pour tout $\epsilon > 0$, il existe une fonction DC $f : C \rightarrow \mathbb{R}$ telle que

$$|c(x) - f(x)| \leq \epsilon \quad \forall x \in C.$$

(iii) Soit $f : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction DC et soit $g : \mathbb{R} \rightarrow \mathbb{R}$ une fonction convexe. Alors la fonction composée

$$(g \circ f)(x) = g(f(x))$$

est DC.

4.3 Problèmes de programmation DC

Les problèmes de programmation traitant de fonctions DC sont appelés *problèmes de programmation DC*. Nous donnons dans cette section une forme générale ainsi que certains cas particuliers des problèmes de programmation DC. La forme générale des problèmes de programmation DC considérés ici est donnée par

$$\min \{f_0(x) \mid x \in X, f_i(x) \leq 0, i = 1, \dots, m\}, \quad (4.3.3)$$

où les $f_i = g_i - h_i$, $i = 0, 1, \dots, m$, sont des fonctions DC et X est un sous-ensemble convexe fermé de \mathbb{R}^n . Le problème

$$\min \{c(x) \mid x \in X, \psi(x) \leq 0\}, \quad (4.3.4)$$

où c est une fonction linéaire, $X \subset \mathbb{R}^n$ est convexe fermé, et ψ est une fonction concave, est souvent appelé un *programme DC canonique*. Tout problème de programmation DC de la forme (4.3.3) peut être transformé en une forme canonique (4.3.4) comme suit.

4.4 Modèles de problèmes de programmation DC

En utilisant une variable additionnelle x_{n+1} , le problème (4.3.3) est réécrit comme

$$\min \{x_{n+1} \mid x \in X, x_{n+1} \in \mathbb{R}, f_0(x) - x_{n+1} \leq 0, f_i(x) \leq 0, i = 1, \dots, m\}.$$

Ensuite, définissons une fonction DC $f : \mathbb{R}^{n+1} \rightarrow \mathbb{R}$ par

$$f(x, x_{n+1}) = \max \{(f_0(x) - x_{n+1}), f_1(x), \dots, f_m(x)\},$$

et considérons $f = g - h$ une décomposition DC déterminée comme en (4.2.2). Finalement, en utilisant une autre variable additionnelle x_{n+2} , et en posant

$$\begin{aligned} z &= \{x, x_{n+1}, x_{n+2}\} \in \mathbb{R}^{n+2}, \\ Z &= \{z \in \mathbb{R}^{n+2} \mid x \in X, g(x, x_{n+1}) - x_{n+2} \leq 0\}, \\ \psi(z) &= x_{n+2} - h(x, x_{n+1}), \end{aligned}$$

nous obtenons le programme DC canonique

$$\min \{c(z) = x_{n+1} \mid z \in Z, \psi(z) \leq 0\}.$$

Pour l'établissement des conditions d'optimalité et pour le développement des méthodes de solution pour le problème (4.3.4), la fonction linéaire c peut être remplacée, dans de nombreux cas, par une fonction convexe. Dans ce qui suit, nous appellerons le problème (4.3.4) un *programme DC canonique généralisé*, si la fonction c est convexe.

4.4 Modèles de problèmes de programmation DC

La programmation DC possède un large champ d'applications. Nous allons ici discuter de problèmes typiques qui peuvent être modélisés comme des problèmes de programmation DC.

4.4.1 Problème de programmation avec des contraintes de complémentarité

Considérons le problème de programmation

$$\min \{c(x) \mid x \in C\}, \tag{4.4.5}$$

4.4 Modèles de problèmes de programmation DC

où $c : \mathbb{R}^n \rightarrow \mathbb{R}$ est une fonction continue et où C est un sous-ensemble convexe fermé de \mathbb{R}^n , avec les contraintes additionnelles suivantes :

$$x \geq 0, \quad h(x) \geq 0, \quad \langle x, h(x) \rangle = 0, \quad (4.4.6)$$

où $h : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est une fonction vecteur continue. Le problème (4.4.6) est habituellement appelé le *problème de complémentarité*; comme il est bien connu, les problèmes (4.4.5) et (4.4.6) sont utilisés pour étudier de nombreux problèmes incluant les modèles d'équilibre économique ou encore les problèmes de jeux théoriques à plusieurs niveaux.

Supposons que les composantes h_i de h , $i = 1, \dots, n$, soient des fonctions concaves, et définissons une fonction

$$\psi(x) = \sum_{i=1}^n \min_{x_i} h_i(x).$$

Alors il est évident que ψ est une fonction concave, et que le système (4.4.6) est équivalent à

$$x \geq 0, \quad h(x) \geq 0, \quad \psi(x) \leq 0.$$

Donc, en posant

$$X = \{x \in \mathbb{R}^n \mid x \in C, x \geq 0, h(x) \geq 0\},$$

nous pouvons formuler le problème (4.4.5) avec les contraintes additionnelles (4.4.6) de manière équivalente sous la forme

$$\min \{c(x) \mid x \in X, \psi(x) \leq 0\},$$

lequel est un problème de programmation DC ou un programme DC canonique généralisé, lorsque c est DC ou convexe, respectivement.

4.4.2 Problèmes d'emplacement de pont

Un pont de longueur minimale doit être construit d'une part, entre deux îles convexes M et D (problème P1) et, d'autre part, entre une île convexe M et le bord d'un lac convexe L (problème P2), respectivement. Les modèles de programmation DC relatifs à ces problèmes sont

$$(P1) \quad \min \{d_M^2(x) \mid x \in D\}$$

et

$$(P2) \quad \min \{d_M^2(x) \mid x \in R \setminus \text{int } L\},$$

respectivement, où R est une région convexe autour du lac L . Remarquons que d_M^2 est une fonction DC, comme nous l'avons vu à la section 2 de ce chapitre.

4.4.3 Problème de placement

Un service d'après vente doit être placé dans un ensemble convexe X du plan \mathbb{R}^2 afin de servir k clients situés aux points $c_j \in X$, $j = 1, \dots, k$. Lorsque le service est placé en $x \in X$, son attraction au client j est décrite par une fonction $g_j(d_j(x))$, où $d_j(x) = \|x - c_j\|$ est la distance de c_j à x . Chaque fonction g_j est supposée convexe. Ainsi choisir l'emplacement avec une attraction totale maximale revient à résoudre le problème

$$\max_{x \in X} f(x) = \sum_{j=1}^k g_j(d_j(x)). \quad (4.4.7)$$

Dès lors, par la proposition 4.2.3 (iii), la fonction objectif f dans (4.4.7) est une fonction DC, puisque chaque fonction $g_j(d_j(x))$ est DC.

4.4.4 Optimisation sur des ensembles efficaces

Considérons le problème de programmation linéaire à fonction objectif multiple suivant :

$$\max_{x \in X} c^i x, \quad i = 1, \dots, p \quad (4.4.8)$$

où X est un polytope (ensemble polyédrique borné) dans \mathbb{R}^n et $c^i \in \mathbb{R}^n \setminus \{0\}$, $i = 1, \dots, p$.

Soit C une matrice de dimension $p \times n$ ayant les lignes c^1, \dots, c^p . Un point $y \in X$ est appelé une solution efficace du problème (4.4.8) s'il n'existe pas de point $x \in X$ tel que $Cx \geq Cy$ et $Cx \neq Cy$. En notant E_x l'ensemble de toutes les solutions efficaces du problème (4.4.8), et en prenant $c : \mathbb{R}^n \rightarrow \mathbb{R}$ une fonction convexe, nous considérons le problème d'optimisation suivant sous l'ensemble efficace E_x :

$$\min_{y \in E_x} c(y). \quad (4.4.9)$$

L'optimisation sur un ensemble efficace est une des plus importantes et des plus intéressantes approches dans l'optimisation à critère multiple.

4.5 Conditions d'optimalité et dualité

Soit $e = (1, \dots, 1)^T \in \mathbb{R}^n$ et soit $b : X \rightarrow \mathbb{R}$ une fonction définie par

$$b(y) = \max \{ eC(x - y) \mid Cx \geq Cy, x \in X \}. \quad (4.4.10)$$

Alors il est évident que b est concave sur X satisfaisant

$$b(y) \geq 0, \quad \forall y \in X,$$

et y est une solution efficace si et seulement si $b(y) = 0$. Donc le problème (4.4.9) peut être formulé comme

$$\min \{ c(y) \mid b(y) \leq 0, y \in X \}, \quad (4.4.11)$$

qui est un problème de programmation DC canonique généralisé.

4.5 Conditions d'optimalité et dualité

4.5.1 Définitions et notations

Nous rappelons dans cette section certaines définitions et notations souvent utilisées en analyse convexe et en optimisation, et que nous utiliserons par après pour établir les conditions d'optimalité.

Epigraphe d'une fonction

Rappelons la définition d'épigraphe d'une fonction vue dans le premier chapitre et définissons par la même occasion l'épigraphe d'une fonction sur un sous-ensemble de \mathbb{R}^n .

Définition 4.5.1. Soit $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ une fonction propre. L'épigraphe de f est l'ensemble non vide

$$\text{epi } f = \{(x, r) \in \mathbb{R}^n \times \mathbb{R} \mid r \geq f(x)\}.$$

Définition 4.5.2. Soit X un sous-ensemble de \mathbb{R}^n . L'épigraphe de f sur X est l'ensemble non vide

$$\text{epi } f|X = \{(x, r) \in \mathbb{R}^n \times \mathbb{R} \mid x \in X, r \geq f(x)\}. \quad (4.5.12)$$

4.5 Conditions d'optimalité et dualité

ϵ -sous-différentiel d'une fonction

Rappelons à présent la définition de ϵ -sous-différentiel de f en x que nous avons déjà vue au début du troisième chapitre.

Définition 4.5.3. Soit $f \in \Gamma_0(\mathbb{R}^n)$, $\epsilon \geq 0$ et $x \in \text{dom } f$. Un vecteur $s \in \mathbb{R}^n$ est appelé un ϵ -sous-gradient de f en x si

$$\forall y \in \text{dom } f \quad f(y) \geq f(x) + \langle s, y - x \rangle - \epsilon.$$

L'ensemble des ϵ -sous-gradients de f en x est noté $\partial_\epsilon f(x)$ et appelé le ϵ -sous-différentiel de f en x .

Si $\epsilon = 0$, alors s et $\partial f(x)$ sont simplement appelés le sous-gradient et le sous-différentiel de f en x , respectivement. Nous définissons alors le ϵ -sous-différentiel de f en x sur un sous-ensemble de \mathbb{R}^n .

Définition 4.5.4. Soit X un sous-ensemble de \mathbb{R}^n . Le ϵ -sous-différentiel de f en x sur X est défini par

$$\partial_{X,\epsilon} f(x) = \{s \in \mathbb{R}^n \mid f(y) \geq f(x) + \langle s, y - x \rangle - \epsilon \quad \forall y \in X\}. \quad (4.5.13)$$

4.5.2 Conditions d'optimalité

Une classe importante des problèmes d'optimisation DC est la suivante :

$$\omega^* = \inf_{x \in X} \{g(x) - h(x)\}, \quad (4.5.14)$$

où g et h sont deux fonctions convexes sur \mathbb{R}^n , et X est un sous-ensemble convexe fermé de \mathbb{R}^n . La proposition suivante donne une condition d'optimalité pour le problème (4.5.14).

Proposition 4.5.1. *Supposons que le problème (4.5.14) soit résoluble. Alors un point $x^* \in X$ est une solution optimale de (4.5.14) si et seulement si il existe $t^* \in \mathbb{R}$ tel que*

$$0 = \inf \{-h(x) + t \mid x \in X, t \in \mathbb{R}, g(x) - t \leq g(x^*) - t^*\}. \quad (4.5.15)$$

Preuve. En utilisant une variable additionnelle t , nous définissons le problème

$$\omega^* = \inf \{g(x) - t \mid x \in X, t \in \mathbb{R}, -h(x) + t \leq 0\}. \quad (4.5.16)$$

Il est évident que $x^* \in X$ est une solution optimale de (4.5.14) si et seulement si il existe $t^* \in \mathbb{R}$ tel que (x^*, t^*) est une solution optimale de (4.5.16). Soit

$$p = n + 1, \quad Z = X \times \mathbb{R}, \quad \omega(z) = g(x) - t, \quad \psi(z) = -h(x) + t.$$

Alors il est facile de voir que les hypothèses de la proposition 4.3 dans [5] sont remplies, donc que les problèmes (4.5.16) et (4.5.15) sont réciproques. \square

En utilisant la notation (4.5.12), nous obtenons la condition d'optimalité géométrique suivante pour le problème (4.5.14).

Proposition 4.5.2. *Un point $x^* \in X$ est une solution optimale du problème (4.5.14) si et seulement si*

$$\text{epi } \bar{g}|_X \subset \text{epi } h|_X, \quad (4.5.17)$$

$$\text{où } \bar{g}(x) = g(x) - (g(x^*) - h(x^*)).$$

Preuve. Nous montrons que la condition (4.5.17) est équivalente à la condition (4.5.15). Comme nous l'avons vu dans la preuve de la proposition 4.5.1, les problèmes (4.5.15) et (4.5.16) sont réciproques. Par conséquent, en prenant (x^*, t^*) une solution optimale du problème (4.5.16), nous avons $h(x^*) = t^*$ dans (4.5.15). Dès lors, par (4.5.15), il suit que

$$h(x) - t \leq 0,$$

4.5 Conditions d'optimalité et dualité

pour chaque $(x, t) \in X \times \mathbb{R}$ satisfaisant

$$\bar{g}(x) - t = g(x) - (g(x^*) - t^*) - t \leq 0,$$

ce qui implique que la condition (4.5.17) tient. D'autre part, par (4.5.17), il suit que

$$0 \leq \inf \{-h(x) + t \mid x \in X, t \in \mathbb{R}, g(x) - t \leq g(x^*) - h(x^*)\}.$$

Puisque le point (x^*, t^*) , avec $x^* \in X, t^* = h(x^*)$, est une solution admissible du problème ci-dessus, la condition (4.5.15) suit. \square

Une autre forme de la condition (4.5.17) est la condition suivante, où nous utilisons la notation (4.5.13).

Proposition 4.5.3. *Un point $x^* \in X$ est une solution optimale du problème (4.5.14) si et seulement si*

$$\partial X_\epsilon h(x^*) \subset \partial X_\epsilon g(x^*) \quad \forall \epsilon \geq 0. \quad (4.5.18)$$

Preuve. Supposons que x^* ne soit pas une solution optimale de (4.5.14). Par la proposition 4.5.2, cela revient à dire que la condition (4.5.17) ne tient pas, i.e., il existe $\bar{x} \in X$ tel que $(\bar{x}, \bar{g}(\bar{x})) \notin \text{epi } h|_X$. Puisque $\text{epi } h|_X$ est un ensemble convexe fermé, il existe un hyperplan non vertical séparant le point $(\bar{x}, \bar{g}(\bar{x}))$ de $\text{epi } h|_X$; i.e., il existe $s \in \mathbb{R}^n, \gamma \in \mathbb{R}$ tels que

$$h(x) \geq \langle s, x - x^* \rangle - \gamma \quad \forall x \in X, \quad (4.5.19)$$

$$\bar{g}(\bar{x}) = g(\bar{x}) - (g(x^*) - h(x^*)) < \langle s, \bar{x} - x^* \rangle - \gamma. \quad (4.5.20)$$

Soit $\epsilon = \gamma + h(x^*)$. Alors par (4.5.19) et (4.5.20), nous avons

$$\epsilon \geq 0,$$

$$h(x) \geq h(x^*) + \langle s, x - x^* \rangle - \epsilon \quad \forall x \in X,$$

$$g(\bar{x}) < g(x^*) + \langle s, \bar{x} - x^* \rangle - \epsilon,$$

ce qui implique que la condition (4.5.18) ne tient pas. \square

Remarque 4.5.1. 1. Pour le cas $X = \mathbb{R}^n$, la condition (4.5.18) donne

$$\partial_\epsilon h(x^*) \subset \partial_\epsilon g(x^*) \quad \forall \epsilon > 0. \quad (4.5.21)$$

C'est un résultat bien connu en optimisation DC qui fut prouvé en premier par Hiriart-Urruty [3].

2. Si $g \equiv 0$, alors le problème (4.5.14) est habituellement appelé un problème de programmation concave. En utilisant la fonction indicatrice de C , ce problème peut être réécrit comme

$$\inf_{x \in \mathbb{R}^n} \{I_C(x) - h(x)\}. \quad (4.5.22)$$

4.5.3 Dualité en programmation DC

Considérons le problème de programmation DC donné sous la forme suivante :

$$\inf_{x \in \mathbb{R}^n} \{g(x) - h(x)\}, \quad (4.5.23)$$

où $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ et $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ sont deux fonctions convexes. Notons que les problèmes sous la forme

$$\inf_{x \in X} \{g(x) - h(x)\},$$

où X est un sous-ensemble convexe de \mathbb{R}^n , peuvent être réécrits sous la forme (4.5.23) en posant

$$g = g + I_X,$$

où I_X est la fonction indicatrice de X . Prenons g^* et h^* les fonctions conjuguées de g et h respectivement, et posons

$$Y = \{y \in \mathbb{R}^n \mid h^*(y) < +\infty\}. \quad (4.5.24)$$

Le problème de programmation

$$\inf_{y \in Y} \{h^*(y) - g^*(y)\} \quad (4.5.25)$$

est appelé le *problème dual de Fenchel-Rockafellar* de (4.5.23). Avant d'établir le rapport entre les problèmes (4.5.23) et (4.5.25), nous donnons certaines formules utiles pour calculer les valeurs des fonctions conjuguées.

Lemme 4.5.1.

(i) Soit $\bar{x} \in \mathbb{R}^n$ et $\bar{y} \in \partial h(\bar{x})$. Alors

$$h^*(\bar{y}) = \langle \bar{y}, \bar{x} \rangle - h(\bar{x}). \quad (4.5.26)$$

(ii) Soit $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ une fonction arbitraire et $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ une fonction convexe. De plus, considérons $f = g - h$. Alors la fonction conjuguée f^* de f est donnée par

$$f^*(z) = \sup_{y \in Y} \{g^*(y+z) - h^*(y)\}, \quad (4.5.27)$$

où Y est défini en (4.5.24).

Preuve. (i) Puisque

$$h(x) \geq h(\bar{x}) + \langle \bar{y}, x - \bar{x} \rangle \quad \forall x \in \mathbb{R}^n,$$

i.e.,

$$\langle \bar{y}, x \rangle - h(x) \leq \langle \bar{y}, \bar{x} \rangle - h(\bar{x}) \quad \forall x \in \mathbb{R}^n,$$

il suit que

$$h^*(\bar{y}) \triangleq \sup_{x \in \mathbb{R}^n} \{\langle \bar{y}, x \rangle - h(x)\} \leq \langle \bar{y}, \bar{x} \rangle - h(\bar{x}).$$

Ceci implique que

$$h^*(\bar{y}) = \langle \bar{y}, \bar{x} \rangle - h(\bar{x}).$$

(ii) Par la définition des fonctions conjuguées, nous avons

$$\begin{aligned} f^*(z) &\triangleq \sup_{x \in \mathbb{R}^n} \{\langle z, x \rangle - (g(x) - h(x))\} \\ &= \sup_{x \in \mathbb{R}^n} \{(\langle y+z, x \rangle - g(x)) - (\langle y, x \rangle - h(x))\} \quad \forall y \in \mathbb{R}^n \\ &\geq \sup_{x \in \mathbb{R}^n} \{\langle y+z, x \rangle - g(x)\} - \sup_{x \in \mathbb{R}^n} \{\langle y, x \rangle - h(x)\} \quad \forall y \in \mathbb{R}^n \\ &\geq \sup_{y \in Y} \{g^*(y+z) - h^*(y)\}. \end{aligned}$$

4.5 Conditions d'optimalité et dualité

Ensuite, supposons par l'absurde que

$$f^*(z) > \sup_{y \in Y} \{g^*(y+z) - h^*(y)\},$$

i.e., il existe $\bar{x} \in \mathbb{R}^n$ tel que

$$\langle z, \bar{x} \rangle - (g(\bar{x}) - h(\bar{x})) > \sup_{y \in Y} \{g^*(y+z) - h^*(y)\}. \quad (4.5.28)$$

Soit alors \bar{y} un sous-gradient de la fonction convexe finie h en \bar{x} . En vue de (4.5.26), nous avons

$$h^*(\bar{y}) = \langle \bar{y}, \bar{x} \rangle - h(\bar{x}).$$

De ceci, il suit que

$$\begin{aligned} \langle z, \bar{x} \rangle - (g(\bar{x}) - h(\bar{x})) &= (\langle \bar{y} + z, \bar{x} \rangle - g(\bar{x})) - (\langle \bar{y}, \bar{x} \rangle - h(\bar{x})) \\ &= (\langle \bar{y} + z, \bar{x} \rangle - g(\bar{x})) - h^*(\bar{y}) \\ &\leq g^*(\bar{y} + z) - h^*(\bar{y}), \end{aligned}$$

ce qui contredit (4.5.28). \square

En utilisant le lemme 4.5.1, nous obtenons les résultats suivants concernant le rapport entre les problèmes (4.5.23) et (4.5.25).

Théorème 4.5.1.

(i) Si le problème (4.5.23) a une solution optimale, alors on a que

$$\inf_{x \in \mathbb{R}^n} \{g(x) - h(x)\} = \inf_{y \in Y} \{h^*(y) - g^*(y)\}. \quad (4.5.29)$$

(ii) Le problème dual de Fenchel-Rockafellar de (4.5.25) est le problème (4.5.23).

Preuve. (i) Par définition³ de f^* , nous avons

$$f^*(0) = \sup_{x \in \mathbb{R}^n} -f(x),$$

³Rappel : $f^*(y) = \sup_{x \in \mathbb{R}^n} \{\langle y, x \rangle - f(x)\}$

i.e.,

$$\inf_{x \in \mathbb{R}^n} f(x) = -f^*(0).$$

Par (4.5.27), il suit que

$$\begin{aligned} \inf_{x \in \mathbb{R}^n} f(x) &= -f^*(0) \\ &= -\sup_{y \in Y} \{g^*(y) - h^*(y)\} \\ &= \inf_{y \in Y} \{h^*(y) - g^*(y)\}. \end{aligned}$$

(ii) Puisque les fonctions h et g sont convexes, la thèse suit directement du fait que $h^{**} = h$ et $g^{**} = g$. \square

Puisque h^* et g^* sont des fonctions convexes, le problème (4.5.25) est exactement un problème de programmation DC de type (4.5.23). De plus, la propriété de symétrie apparaît également dans le rapport entre les solutions optimales de ces problèmes dans le sens suivant.

Proposition 4.5.4.

(i) Si x^* est une solution optimale du problème (4.5.23), alors chaque $y^* \in \partial h(x^*)$ est une solution optimale du problème (4.5.25).

(ii) Si y^* est une solution optimale du problème (4.5.25), alors chaque $x^* \in \partial g^*(y^*)$ est une solution optimale du problème (4.5.23).

Preuve. Nous allons montrer (i). L'assertion (ii) se démontre de manière analogue. Soit x^* une solution optimale de (4.5.23), et soit $y^* \in \partial h(x^*)$. Alors il suit par (4.5.18) que $y^* \in \partial g(x^*)$. Par (4.5.26) nous avons

$$h^*(y^*) = \langle y^*, x^* \rangle - h(x^*)$$

et

$$g^*(y^*) = \langle y^*, x^* \rangle - g(x^*).$$

Par conséquent,

$$h^*(y^*) - g^*(y^*) = g(x^*) - h(x^*),$$

ce qui implique par (4.5.29) que y^* est une solution optimale du problème (4.5.25). \square

4.6 Algorithme DC pour les programmes DC généraux

Avant de donner un aperçu plus détaillé de l'algorithme DC qui permettra de concevoir une nouvelle vision de celui-ci et qui pourrait en partie expliquer son efficacité, nous allons rappeler le contexte de la programmation DC.

4.6.1 Rappels

Nous travaillons avec l'espace $X = \mathbb{R}^n$, celui-ci étant muni du produit scalaire canonique $\langle \cdot, \cdot \rangle$ et de la norme Euclidienne correspondante $\|\cdot\|$. Ainsi l'espace dual Y de X peut être identifié à X lui-même.

Pour deux fonctions $g, h \in \Gamma_0(X)$, un programme DC général est de la forme

$$(P_{dc}) \quad \alpha = \inf_{x \in X} \{f(x) = g(x) - h(x)\}.$$

Notons que le caractère fini de α implique simplement que

$$\text{dom } g \subset \text{dom } h \quad \text{et} \quad \text{dom } h^* \subset \text{dom } g^*. \quad (4.6.30)$$

Un point x^* est un *minimum local* de $g - h$ si $g(x^*) - h(x^*)$ est fini (i.e., $x^* \in \text{dom } g \cap \text{dom } h$) et s'il existe un voisinage \mathcal{U} de x^* tel que

$$g(x^*) - h(x^*) \leq g(x) - h(x) \quad \forall x \in \mathcal{U}. \quad (4.6.31)$$

Un point x^* est un *point critique* de $g - h$ si $\partial g(x^*) \cap \partial h(x^*) \neq \emptyset$.

Nous avons obtenu le programme dual de (P_{dc}) :

$$(D_{dc}) \quad \alpha = \inf_{y \in Y} \{h^*(y) - g^*(y)\}.$$

Nous observons la symétrie parfaite entre les programmes DC primal et dual : le dual de (D_{dc}) est exactement (P_{dc}) .

Soit \mathcal{P} et \mathcal{D} les ensembles de solution des problèmes (P_{dc}) et (D_{dc}) , respectivement, et soit

$$\mathcal{P}_l = \{x^* \in X \mid \partial h(x^*) \subset \partial g(x^*)\}, \quad \mathcal{D}_l = \{y^* \in Y \mid \partial g^*(y^*) \subset \partial h^*(y^*)\}.$$

4.6.2 Algorithme DC

L'algorithme DC (DCA) consiste en la construction de deux suites $\{x^k\}$ et $\{y^k\}$ — candidates pour être les solutions primales et duales, respectivement — que nous améliorons à chaque itération⁴ de telle façon que leurs valeurs d'adhérence correspondantes x^∞ et y^∞ satisfassent la condition d'optimalité locale

$$\partial h(x^\infty) \subset \partial g(x^\infty) \quad \text{et} \quad \partial g^*(y^\infty) \subset \partial h^*(y^\infty),$$

i.e.,

$$(x^\infty, y^\infty) \in \mathcal{P}_l \times \mathcal{D}_l$$

ou soient des points critiques de $g - h$ et $h^* - g^*$, respectivement.

Ces suites sont engendrées comme suit : x^{k+1} (resp. y^k) est une solution du programme convexe (P_k) (resp. (D_k)) définis par

$$(P_k) \quad \begin{cases} \alpha_k = \inf \{g(x) - [h(x^k) + \langle x - x^k, y^k \rangle]\} \\ \text{s.c. } x \in X, \end{cases}$$

$$(D_k) \quad \begin{cases} \inf \{h^*(y) - [g^*(y^{k-1}) + \langle x^k, y - y^{k-1} \rangle]\} \\ \text{s.c. } y \in Y. \end{cases}$$

(P_k) (resp. (D_k)) est obtenu par (P_{dc}) (resp. (D_{dc})) en remplaçant h (resp. g^*) par sa minorisation affine définie par $y^k \in \partial h(x^k)$ (resp. $x^k \in \partial g^*(y^{k-1})$). L'algorithme DC donne ainsi le schéma suivant.

Algorithme DC

- Choisir $x^0 \in \text{dom } g$ arbitraire.
- Pour $k = 0, 1, \dots$ calculer :

$$\begin{aligned} y^k &\in \partial h(x^k), \\ x^{k+1} &\in \partial g^*(y^k). \end{aligned}$$

⁴donc les suites $\{g(x^k) - h(x^k)\}$ et $\{h^*(y^k) - g^*(y^k)\}$ sont décroissantes

4.6 Algorithme DC pour les programmes DC généraux

Cet algorithme correspond en réalité à l'algorithme DC simplifié où x^{k+1} (resp. y^k) est choisi de façon arbitraire dans $\partial g^*(y^k)$ (resp. $\partial h(x^k)$).

Dans la forme complète de l'algorithme DC, nous imposons le choix naturel suivant :

$$x^{k+1} \in \arg \min \{g(x) - h(x) \mid x \in \partial g^*(y^k)\} \quad (4.6.32)$$

et

$$y^k \in \arg \min \{h^*(y) - g^*(y) \mid y \in \partial h(x^k)\}. \quad (4.6.33)$$

Les problèmes (4.6.32) et (4.6.33) sont équivalents aux problèmes de maximisation convexe (4.6.34) et (4.6.35), respectivement :

$$x^{k+1} \in \arg \min \{\langle x, y^k \rangle - h(x) \mid x \in \partial g^*(y^k)\}, \quad (4.6.34)$$

$$y^k \in \arg \min \{\langle x^k, y \rangle - g^*(y) \mid y \in \partial h(x^k)\}. \quad (4.6.35)$$

L'algorithme DC complet assure que $(x^\infty, y^\infty) \in \mathcal{P}_l \times \mathcal{D}_l$. Il peut être vu comme une sorte d'approche de décomposition des problèmes primal (P_{dc}) et dual (D_{dc}). D'un point de vue pratique, bien que les problèmes (4.6.32) et (4.6.33) sont plus simples que les programmes (P_{dc}) et (D_{dc}) (car nous travaillons dans $\partial h(x^{k+1})$ et $\partial g^*(y^k)$ avec des problèmes de maximisation convexe), ils restent cependant des programmes non convexes et sont donc toujours difficiles à résoudre. En pratique, excepté les cas où les problèmes de maximisation convexe (4.6.34) et (4.6.35) sont faciles à traiter, on utilise généralement l'algorithme DC simplifié pour résoudre les programmes DC.

L'algorithme DC a été introduit par Pham Dinh Tao [10] en 1986 comme une extension des algorithmes de sous-gradient pour la programmation DC. Cependant, ce domaine a réellement été développé ces dernières années par les travaux de Le Thi Hoai An et Pham Dinh Tao pour résoudre des problèmes d'optimisation non convexes non différentiables. Aujourd'hui, l'algorithme DC est un des rares algorithmes (dans l'approche de l'analyse convexe pour la programmation DC) qui permet de résoudre des programmes DC de grande taille.

Pham Dinh Tao et Le Thi Hoai An ont prouvé que pour l'algorithme DC simplifié nous avons que :

4.6 Algorithme DC pour les programmes DC généraux

1. Les suites $\{g(x^k) - h(x^k)\}$ et $\{h^*(y^k) - g^*(y^k)\}$ sont décroissantes.
2. Si la valeur optimale α du problème (P_{dc}) est finie et si les suites $\{x^k\}$ et $\{y^k\}$ sont bornées, alors toute valeur d'adhérence x^∞ (resp. y^∞) de la suite $\{x^k\}$ (resp. $\{y^k\}$) est un point critique de $g - h$ (resp. $h^* - g^*$).
3. L'algorithme DC converge linéairement pour les programmes DC généraux.

Remarque 4.6.1. *Le choix de la décomposition DC de la fonction objectif dans un programme DC et le choix du point initial pour l'algorithme DC sont des questions ouvertes qui sont encore étudiées. Bien sûr, cela dépend fortement de la forme très spécifique du problème que l'on considère. En pratique, pour résoudre un programme DC donné, nous essayons de choisir les fonctions g et h de telle façon que les suites $\{x^k\}$ et $\{y^k\}$ peuvent être facilement calculées, c'est-à-dire qu'elles sont soit sous une forme explicite ou que leurs calculs sont peu coûteux.*

Chapitre 5

Méthode de point proximal

5.1 Introduction

En général, la stratégie pour minimiser une fonction convexe non différentiable est la suivante : nous construisons des directions de descente à chaque itération grâce à l'information collectée dans un voisinage du point actuel et nous pouvons ainsi imiter la méthode de la plus forte pente en optimisation différentiable pour obtenir l'itérée suivante. Ici notre démarche est différente. Nous voulons construire une fonction convexe différentiable approximant la fonction convexe g non nécessairement différentiable de telle façon que les minima de g et de son approximation coïncident. Puisque la fonction approximante, appelée régularisée Moreau-Yosida de g , est convexe et différentiable, nous pouvons utiliser, pour la minimiser, les méthodes classiques telles que la méthode du gradient ou la méthode BFGS. Cependant ces méthodes sont souvent non implémentables et pour les rendre implémentables, des approximations comme dans les méthodes faisceaux doivent être utilisées.

Dans ce chapitre, nous allons nous intéresser au problème d'optimisation général dans lequel nous tenterons de minimiser la fonction convexe $g \in \Gamma_0(\mathbb{R}^n)$:

$$\begin{cases} \min & g(x) \\ \text{s.c.} & x \in \mathbb{R}^n. \end{cases} \quad (5.1.1)$$

Dans le chapitre suivant, nous aborderons alors le problème particulier

5.2 Régularisée Moreau-Yosida

de programmation DC suivant

$$\begin{cases} \min & f(x) \equiv g(x) - h(x) \\ \text{s.c.} & x \in \mathbb{R}^n, \end{cases} \quad (5.1.2)$$

et nous terminerons, dans le dernier chapitre, par la description d'une méthode faisceau pour résoudre ce problème de programmation DC (5.1.2).

5.2 Régularisée Moreau-Yosida

Définition 5.2.1. Soit $g \in \Gamma_0(\mathbb{R}^n)$ et $c > 0$. La fonction $G : \mathbb{R}^n \rightarrow \mathbb{R}$ définie par

$$G(x) = \min_{y \in \mathbb{R}^n} \left\{ g(y) + \frac{1}{2c} \|y - x\|^2 \right\}$$

est appelée la régularisée Moreau-Yosida de g .

Ici nous considérons une fonction g avec de possibles valeurs égales à $+\infty$. Ceci nous permet d'examiner les problèmes de la forme :

$$\min_{x \in C} h(x)$$

où $h : \mathbb{R}^n \rightarrow \mathbb{R}$ est convexe et $C \subseteq \mathbb{R}^n$ est un ensemble convexe fermé. Ce problème revient à minimiser $g(x)$ sur \mathbb{R}^n où $g = h + I_C$.

Tout d'abord, nous prouvons que G est bien définie, i.e., que $\text{dom } G = \mathbb{R}^n$.

Proposition 5.2.1. Pour chaque $x \in \mathbb{R}^n$, le problème

$$\min_{y \in \mathbb{R}^n} \left\{ g(y) + \frac{1}{2c} \|y - x\|^2 \right\}$$

a une solution unique notée $\text{prox}_{cg}(x)$. De plus, $\text{prox}_{cg}(x)$ est l'unique point $y \in \mathbb{R}^n$ tel que $\frac{1}{c}(x - y) \in \partial g(y)$.

5.2 Régularisée Moreau-Yosida

Preuve. Soit $x \in \mathbb{R}^n$ et considérons la fonction

$$\tilde{g} : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\} \quad \tilde{g}(y) = g(y) + \frac{1}{2c} \|y - x\|^2.$$

Puisque \tilde{g} est strictement convexe, il existe au plus un minimum pour \tilde{g} . D'autre côté, puisque $g \in \Gamma_0(\mathbb{R}^n)$, g est fermée et minorée par une fonction affine. Donc

$$\exists s \neq 0 \exists r \in \mathbb{R} \text{ t.q. } \forall y \quad \tilde{g}(y) \geq \langle s, y \rangle + r + \frac{1}{2c} \|y - x\|^2.$$

D'où \tilde{g} est fermée et coercive¹, ce qui implique que \tilde{g} admet au moins un minimum.

Soit $\text{prox}_{cg}(x)$ cet unique minimum. En utilisant les conditions d'optimalité, ce minimum est caractérisé par

$$0 \in \partial \left(g + \frac{1}{2c} \|\cdot - x\|^2 \right) (\text{prox}_{cg}(x))$$

ou encore, puisque $\frac{1}{2c} \|\cdot - x\|^2$ a \mathbb{R}^n pour domaine, par

$$0 \in \partial g(\text{prox}_{cg}(x)) + \frac{1}{c} [\text{prox}_{cg}(x) - x],$$

i.e.,

$$\frac{1}{c} [x - \text{prox}_{cg}(x)] \in \partial g(\text{prox}_{cg}(x)).$$

□

Le vecteur $\text{prox}_{cg}(x)$ est appelé le **point proximal** de x . Il satisfait la propriété suivante

$$\forall x \in \mathbb{R}^n \quad G(x) = g(\text{prox}_{cg}(x)) + \frac{1}{2c} \|\text{prox}_{cg}(x) - x\|^2.$$

En outre, si nous prenons $s_g(x) \equiv \frac{1}{c} [x - \text{prox}_{cg}(x)]$, alors $s_g(x) \in \partial g(\text{prox}_{cg}(x))$.

Lorsque $g = I_S$ la fonction indicatrice associée à S , un sous-ensemble convexe fermé de \mathbb{R}^n , la régularisée Moreau-Yosida de g est

$$G(x) = \min_{y \in S} \left\{ I_S(y) + \frac{1}{2c} \|y - x\|^2 \right\} = \min_{y \in S} \frac{1}{2c} \|y - x\|^2.$$

Dans ce cas, $\text{prox}_{cg}(x)$ est la projection orthogonale de x sur S (d'où le nom **point proximal** de x).

¹ \tilde{g} est coercive si $\tilde{g}(y) \rightarrow +\infty$ quand $\|y\| \rightarrow +\infty$

Interprétation géométrique

Considérons l'ensemble niveau $S = \{y \mid g(y) \leq g(\text{prox}_{cg}(x))\}$. Alors, pour tout $y \in S$, nous avons (puisque $s_g(x) \in \partial g(\text{prox}_{cg}(x))$) que

$$g(\text{prox}_{cg}(x)) \geq g(y) \geq g(\text{prox}_{cg}(x)) + \langle s_g(x), y - \text{prox}_{cg}(x) \rangle,$$

i.e.,

$$\langle s_g(x), y - \text{prox}_{cg}(x) \rangle \leq 0$$

ou encore, en utilisant l'expression de $s_g(x)$,

$$\langle x - \text{prox}_{cg}(x), y - \text{prox}_{cg}(x) \rangle \leq 0.$$

Puisque $\text{prox}_{cg}(x) \in S$, cela signifie que $\text{prox}_{cg}(x) = \text{Proj}_S(x)$.

Le théorème suivant donne les propriétés de la régularisée Moreau-Yosida G .

Théorème 5.2.1. *La régularisée Moreau-Yosida G est finie partout, convexe et différentiable. Son gradient est*

$$\nabla G(x) = s_g(x) = \frac{1}{c}[x - \text{prox}_{cg}(x)] \in \partial g(\text{prox}_{cg}(x)).$$

Sa conjuguée est $G^* : \mathbb{R}^n \rightarrow \mathbb{R}$ $G^*(s) = g^*(s) + \frac{1}{2c} \|s\|^2$.

De plus, pour tout x et $x' \in \mathbb{R}^n$,

$$\|\nabla G(x) - \nabla G(x')\|^2 \leq \frac{1}{c} \langle \nabla G(x) - \nabla G(x'), x - x' \rangle$$

et

$$\|\nabla G(x) - \nabla G(x')\| \leq \frac{1}{c} \|x - x'\|.$$

En d'autres termes, ∇G est Lipschitz continu sur \mathbb{R}^n de constante $(1/c)$.

Preuve. Nous montrons uniquement que G est différentiable et que

$$\nabla G(x) = s_g(x) = \frac{1}{c}[x - \text{prox}_{cg}(x)] \in \partial g(\text{prox}_{cg}(x)).$$

5.2 Régularisée Moreau-Yosida

Pour cela, il suffit de prouver que $G'(x, d) = (1/c)[x - \text{prox}_{cg}(x)]^T d$ pour tout $d \in \mathbb{R}^n$. Soit $d \in \mathbb{R}^n$, $d \neq 0$ et $t > 0$. Alors

$$\begin{aligned} \frac{G(x+td) - G(x)}{t} &\leq \frac{g(\text{prox}_{cg}(x)) + (1/2c) \|\text{prox}_{cg}(x) - x - td\|^2 - G(x)}{t} \\ &= \frac{g(\text{prox}_{cg}(x)) + (1/2c) \|\text{prox}_{cg}(x) - x\|^2 + (t^2/2c) \|d\|^2 + (t/c) \langle x - \text{prox}_{cg}(x), d \rangle - G(x)}{t} \\ &= (t/2c) \|d\|^2 + (1/c) \langle x - \text{prox}_{cg}(x), d \rangle. \end{aligned}$$

D'où $G'(x, d) \leq (1/c) \langle x - \text{prox}_{cg}(x), d \rangle$ pour tout $d \in \mathbb{R}^n$. Puisque $G'(x, \cdot)$ est convexe, nous avons

$$0 = G'(x, \frac{1}{2}d + \frac{1}{2}(-d)) \leq \frac{1}{2}G'(x, d) + \frac{1}{2}G'(x, -d)$$

et donc,

$$G'(x, d) \geq -G'(x, -d) \geq -\frac{1}{c} \langle x - \text{prox}_{cg}(x), -d \rangle = \frac{1}{c} \langle x - \text{prox}_{cg}(x), d \rangle.$$

Par conséquent, $G'(x, d) = (1/c)[x - \text{prox}_{cg}(x)]^T d$ pour tout $d \in \mathbb{R}^n$. Donc $\nabla G(x) = s_g(x) = (1/c)[x - \text{prox}_{cg}(x)]$ et par la proposition 5.2.1, $\nabla G(x) \in \partial g(\text{prox}_{cg}(x))$. \square

Exemple 5.2.1. 1. $g(x) = |x|$. La régularisée Moreau-Yosida de g est

$$G(x) = \begin{cases} (c/2)x^2 & \text{si } |x| \leq 1/c, \\ |x| - (1/2c) & \text{si } |x| > 1/c. \end{cases}$$

Expliquons le détail qui nous permet d'obtenir $G(x)$.

Considérons la fonction

$$\tilde{g}(y) = |y| + \frac{c}{2} |y - x|^2.$$

Tout d'abord, commençons par regarder si l'équivalence suivante est correcte :

$$0 \text{ est un minimum de } \tilde{g} \iff |x| \leq \frac{1}{c}.$$

Par définition de \tilde{g} , nous avons

$$\partial \tilde{g}(0) = [-1, 1] - cx.$$

5.2 Régularisée Moreau-Yosida

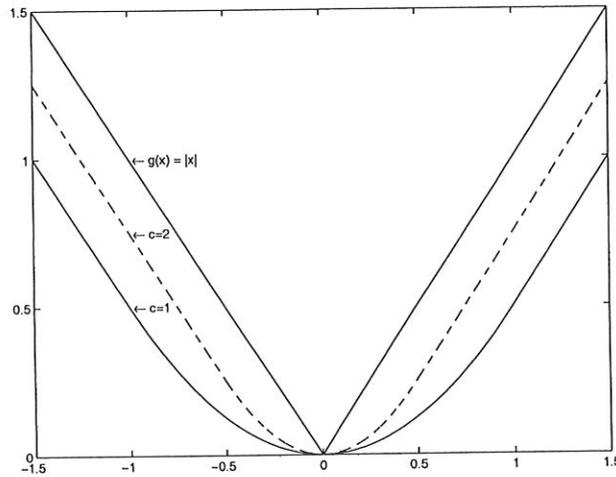


FIG. 5.1 – Régularisée Moreau-Yosida de $g(x) = |x|$.

Ainsi

$$\begin{aligned} 0 \in \partial \tilde{g}(0) &\iff cx \in [-1, 1] \\ &\iff -\frac{1}{c} \leq x \leq \frac{1}{c} \\ &\iff |x| \leq \frac{1}{c}. \end{aligned}$$

Dès lors 0 est un minimum de \tilde{g} et nous avons

$$G(x) = \tilde{g}(0) = \frac{c}{2} |x|^2 = (c/2)x^2 \text{ si } |x| \leq (1/c).$$

D'autre part, nous devons montrer que

$$G(x) = |x| - (1/2c) \text{ si } |x| > (1/c).$$

• $x > \frac{1}{c} \stackrel{?}{\Rightarrow} x - \frac{1}{c}$ est un minimum de \tilde{g} . Regardons si

$$0 \in \partial \tilde{g}\left(x - \frac{1}{c}\right).$$

$$\begin{aligned} x - \frac{1}{c} > 0 &\Rightarrow \tilde{g} \text{ est dérivable en } x - \frac{1}{c} \\ &\Rightarrow \tilde{g}'\left(x - \frac{1}{c}\right) = 1 + c\left(x - \frac{1}{c} - x\right) = 0. \end{aligned}$$

5.2 Régularisée Moreau-Yosida

Ainsi, lorsque $x > (1/c)$,

$$\begin{aligned} G(x) &= \tilde{g}\left(x - \frac{1}{c}\right) = x - \frac{1}{c} + \frac{c}{2}\left(x - \frac{1}{c} - x\right)^2 \\ &= x - \frac{1}{c} + \frac{1}{2c} \\ &= x - \frac{1}{2c}. \end{aligned}$$

• $x < -\frac{1}{c}$. De manière analogue, nous pouvons montrer que $x + \frac{1}{c}$ est un minimum de \tilde{g} et que $G(x) = -x - \frac{1}{2c}$. Dès lors nous obtenons bien que

$$G(x) = |x| - \frac{1}{2c} \text{ lorsque } |x| > (1/c).$$

2. $g = I_C$ la fonction indicatrice associée à $C = [-1, 1]$.

La régularisée Moreau-Yosida de g est

$$G(x) = \begin{cases} 0 & \text{si } x \in [-1, 1], \\ (c/2)(x-1)^2 & \text{si } x > 1, \\ (c/2)(x+1)^2 & \text{si } x < -1. \end{cases}$$

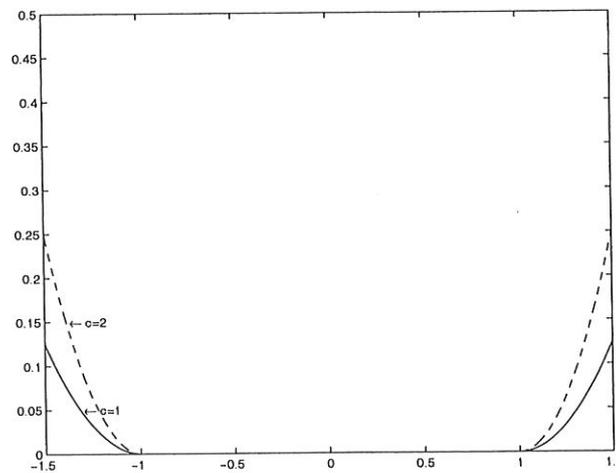


FIG. 5.2 – Régularisée Moreau-Yosida de $g(x) = I_C(x)$.

3. $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ définie par

$$g(x) = \begin{cases} x & \text{si } -1 \leq x \leq 2, \\ +\infty & \text{sinon.} \end{cases}$$

5.2 Régularisée Moreau-Yosida

La régularisée Moreau-Yosida de g est

$$G(x) = \begin{cases} -1 + (1/2c)(1+x)^2 & \text{si } x \leq c-1, \\ x - (c/2) & \text{si } c-1 < x < c+2, \\ 2 + (1/2c)(2-x)^2 & \text{si } x \geq c+2. \end{cases}$$

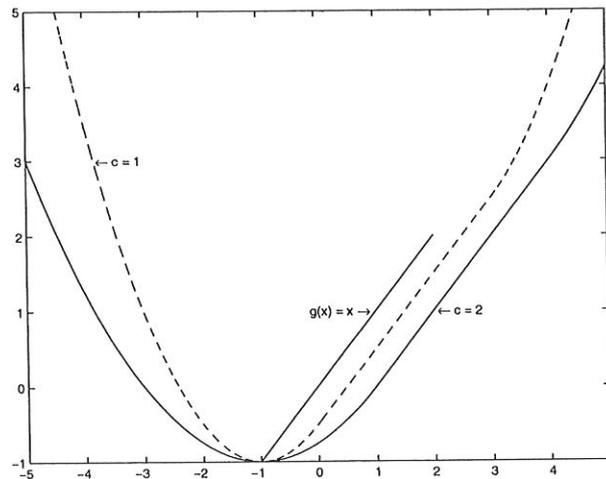


FIG. 5.3 – Régularisée Moreau-Yosida de $g(x) = x$ si $-1 \leq x \leq 2$.

Dans tous ces exemples, nous pouvons observer que les minima de g et de G sont les mêmes. Ce résultat est vrai pour toute fonction convexe. Ceci est prouvé dans le théorème qui suit. Cependant, afin d'obtenir ce résultat, nous avons besoin de prouver le lemme suivant.

Lemme 5.2.1. Pour tout $x \in \mathbb{R}^n$,

$$G(x) + (c/2) \|s_g(x)\|^2 = g(\text{prox}_{cg}(x)) + c \|s_g(x)\|^2 \leq g(x).$$

Preuve. Puisque

$$G(x) = g(\text{prox}_{cg}(x)) + \frac{1}{2c} \|\text{prox}_{cg}(x) - x\|^2$$

5.2 Régularisée Moreau-Yosida

et puisque

$$s_g(x) = \frac{1}{c}[x - \text{prox}_{cg}(x)],$$

nous avons

$$\begin{aligned} G(x) + \frac{c}{2} \|s_g(x)\|^2 &= g(\text{prox}_{cg}(x)) + \frac{1}{2c} \|\text{prox}_{cg}(x) - x\|^2 + \frac{c}{2} \|s_g(x)\|^2 \\ &= g(\text{prox}_{cg}(x)) + \frac{c^2}{2c} \|s_g(x)\|^2 + \frac{c}{2} \|s_g(x)\|^2 \\ &= g(\text{prox}_{cg}(x)) + \frac{c}{2} \|s_g(x)\|^2 + \frac{c}{2} \|s_g(x)\|^2 \\ &= g(\text{prox}_{cg}(x)) + c \|s_g(x)\|^2. \end{aligned}$$

D'un autre côté, puisque $s_g(x) \in \partial g(\text{prox}_{cg}(x))$, nous avons

$$\begin{aligned} g(x) &\geq g(\text{prox}_{cg}(x)) + \langle s_g(x), x - \text{prox}_{cg}(x) \rangle \\ &= g(\text{prox}_{cg}(x)) + c \|s_g(x)\|^2. \end{aligned}$$

Donc la preuve est complète. \square

Théorème 5.2.2.

(1) $\inf_{x \in \mathbb{R}^n} G(x) = \inf_{x \in \mathbb{R}^n} g(x)$ (égalité dans $\mathbb{R} \cup \{+\infty\}$).

(2) Les relations suivantes sont équivalentes :

- (i) x minimise g ,
- (ii) $\text{prox}_{cg}(x) = x$,
- (iii) $s_g(x) = 0$,
- (iv) x minimise G ,
- (v) $g(\text{prox}_{cg}(x)) = g(x)$,
- (vi) $G(x) = g(x)$.

Preuve. (1) Par le théorème 5.2.1, nous avons $G^*(0) = g^*(0)$, i.e.,

$$\sup_{x \in \mathbb{R}^n} \{\langle 0, x \rangle - G(x)\} = \sup_{x \in \mathbb{R}^n} \{\langle 0, x \rangle - g(x)\}.$$

5.3 Algorithme de point proximal

Mais ceci signifie que

$$\inf_{x \in \mathbb{R}^n} G(x) = \inf_{x \in \mathbb{R}^n} g(x).$$

- (2) (i) \Rightarrow (ii). Si x minimise g , alors $y = x$ minimise la fonction $\tilde{g}(y) = g(y) + (1/2c) \|y - x\|^2$ et donc $x = \text{prox}_{cg}(x)$ car $\text{prox}_{cg}(x)$ est l'unique minimum de \tilde{g} (voir proposition 5.2.1).
(ii) \Leftrightarrow (iii) car $s_g(x) = (1/c)[x - \text{prox}_{cg}(x)]$.
(iii) \Leftrightarrow (iv) car $\nabla G(x) = s_g(x)$.
(iv) \Rightarrow (v) car (iv) \Rightarrow (iii) \Rightarrow (ii) \Rightarrow (v).
(v) \Rightarrow (iii) \Rightarrow (iv). Par le lemme 5.2.1, nous avons

$$(v) \Rightarrow \|s_g(x)\|^2 \leq 0$$

et donc (iii) et alors (iv).

(v) \Rightarrow (vi). Puisque (v) \Rightarrow (iii), il suit par le lemme 5.2.1 que $G(x) = g(\text{prox}_{cg}(x))$ et donc, en utilisant (v) à nouveau, il suit que $G(x) = g(x)$.

(vi) \Rightarrow (i). Par le lemme 5.2.1, nous avons

$$(vi) \Rightarrow \|s_g(x)\|^2 \leq 0$$

et donc (iii) et alors (ii). D'où

$$0 = s_g(x) \in \partial g(\text{prox}_{cg}(x)) = \partial g(x).$$

□

5.3 Algorithme de point proximal

Par le théorème 5.2.2, minimiser g revient à trouver un point fixe du **prox-opérateur** $\text{prox}_{cg}(x)$. D'où l'itération $x^{k+1} = \text{prox}_{cg}(x^k)$ pour trouver un minimum de g . Cet algorithme est connu sous le nom d'**Algorithme de Point Proximal**.

5.3 Algorithme de point proximal

Algorithme de Point Proximal

Pas 1 : Choisir un point de départ $x^0 \in \mathbb{R}^n$ et $c_0 > 0$. Poser $k = 0$.

Pas 2 : Calculer $x^{k+1} = \text{prox}_{c_k g}(x^k)$ en résolvant le problème

$$\min_{y \in \mathbb{R}^n} \left\{ g(y) + (1/2c_k) \|y - x^k\|^2 \right\}.$$

Pas 3 : Si $x^{k+1} = x^k$: STOP, x^{k+1} est un minimum de g .

Pas 4 : Choisir $c_{k+1} > 0$. Augmenter k de 1 et retourner au Pas 2.

Lorsque $c_k = 1/c$ pour tout k , le gradient $\nabla G(x^k)$ est égal à $c[x^k - \text{prox}_{c_k g}(x^k)]$. Alors $x^{k+1} = \text{prox}_{c_k g}(x^k)$ devient

$$x^{k+1} = x^k - (1/c)\nabla G(x^k).$$

Donc, dans ce cas, la méthode de point proximal est équivalente à la méthode du gradient appliquée à G avec un pas constant $1/c$.

Interprétation géométrique

Puisque

$$x^{k+1} = \arg \min_{y \in \mathbb{R}^n} \left\{ g(y) + \frac{1}{2c_k} \|y - x^k\|^2 \right\},$$

nous obtenons, en utilisant les conditions d'optimalité, que

$$0 \in \partial g(x^{k+1}) + \nabla \left((1/2c_k) \|\cdot - x^k\|^2 \right) (x^{k+1}),$$

i.e.,

$$\nabla \left(- (1/2c_k) \|\cdot - x^k\|^2 \right) (x^{k+1}) \in \partial g(x^{k+1}).$$

Donc en x^{k+1} la pente de la tangente de $-(1/2c_k) \|\cdot - x^k\|^2$ coïncide avec la pente d'un certain sous-gradient de g . En conséquence, x^{k+1} est l'unique point auquel le graphe de la fonction quadratique $-(1/2c_k) \|\cdot - x^k\|^2$

augmente ou diminue jusqu'à ce qu'il touche le graphe de $g(y)$. Quand c_k est grand, le graphe de la fonction quadratique est obtus et la progression vers le minimum de g est importante; tandis que la méthode donne une progression lente lorsque c_k est petit.

5.4 Convergence de la méthode de point proximal

Soit $\{x^k\}_k$ la suite engendrée par la méthode de point proximal. Par définition de $x^{k+1} = \arg \min_y \left\{ g(y) + (1/2c_k) \|y - x^k\|^2 \right\}$, nous avons

$$\gamma^k \equiv \frac{1}{c_k} (x^k - x^{k+1}) \in \partial g(x^{k+1}).$$

Donc la prox-itération peut être écrite

$$x^{k+1} = x^k - c_k \gamma^k \text{ avec } \gamma^k \in \partial g(x^{k+1}).$$

Nous énonçons à présent le théorème de convergence de la méthode de point proximal. La preuve de la convergence sera donnée dans le chapitre suivant dans le cadre particulier de la méthode de point proximal pour les fonctions DC.

Théorème 5.4.1. *Soit $\{x^k\}_{k \in \mathbb{N}}$ la suite engendrée par l'algorithme de point proximal. Si $\sum_{k=0}^{+\infty} c_k = +\infty$, alors*

(i) $\lim_{k \rightarrow \infty} g(x^k) = g^* = \inf_{x \in \mathbb{R}^n} g(x)$,

(ii) la suite $\{x^k\}$ converge vers un minimum de g (s'il y en a un).

Remarque 5.4.1. *En particulier, si $c_k = 1/c$ pour tout k avec $c > 0$, la suite engendrée par l'algorithme de point proximal converge vers un minimum de g (s'il existe un minimum).*

Chapitre 6

Méthode de point proximal pour la programmation DC

6.1 Introduction

Dans ce chapitre, nous allons résoudre une classe spéciale de problèmes d'optimisation non convexes :

$$\min_{x \in \mathbb{R}^n} f(x), \quad (6.1.1)$$

où $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ est une fonction non convexe. Dans de nombreux cas, par exemple en contrôle optimal ou en design ingénierie, la fonction non convexe f peut être traitée comme une différence de deux fonctions convexes

$$f(x) = g(x) - h(x) \quad \forall x \in \mathbb{R}^n, \quad (6.1.2)$$

où $g : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ et $h : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ sont propres, convexes et s.c.i. Dans ce cas, nous avons vu que la fonction f est appelée une fonction DC.

L'intérêt d'étudier les fonctions DC est motivé par la possibilité d'utiliser deux fois la structure convexe d'une telle représentation lorsqu'on traite des problèmes non convexes. C'est même particulièrement intéressant lorsque l'une des deux fonctions convexes — ou les deux — est non différentiable. Bien qu'il existe beaucoup d'articles consacrés à la théorie des fonctions DC dans la littérature (voir par exemple, [1] [2] [4]), seulement quelques uns proposent des algorithmes spécifiques et rapportent

6.2 Algorithme de point proximal

des résultats numériques. Ici nous allons expliquer l'algorithme de point proximal pour la programmation DC dans le détail.

On a vu dans le chapitre précédent que l'algorithme de point proximal est une méthode efficace pour résoudre des problèmes d'optimisation convexes non différentiables. Sa remarquable particularité est qu'un problème d'optimisation convexe non différentiable peut être transformé en un problème d'optimisation convexe continuellement différentiable. En conséquence, nous pouvons utiliser certaines méthodes d'optimisation différentiable pour le traiter. Ce chapitre a pour but d'étudier l'utilité de l'algorithme de point proximal pour minimiser une fonction DC.

Soit donc f une fonction DC, i.e., $f = g - h$ où g et h sont deux fonctions propres, convexes et s.c.i. sur \mathbb{R}^n satisfaisant $\text{dom } g \cap \text{dom } h \neq \emptyset$. Nous considérons le problème

$$\min_{x \in \mathbb{R}^n} \{g(x) - h(x)\} \quad (6.1.3)$$

et son dual associé

$$\min_{y \in \mathbb{R}^n} \{h^*(y) - g^*(y)\}. \quad (6.1.4)$$

Nous savons déjà que

$$\min_{x \in \mathbb{R}^n} \{g(x) - h(x)\} = \min_{y \in \mathbb{R}^n} \{h^*(y) - g^*(y)\},$$

et qu'une condition nécessaire pour que $x \in \text{dom } f$ soit un *minimum local* de f est que

$$\partial h(x) \subset \partial g(x). \quad (6.1.5)$$

Comme en général cette condition nécessaire est difficile à atteindre, nous nous concentrerons sur le fait de trouver des *points critiques de f* , à savoir des points satisfaisant la condition moins exigeante

$$\partial g(x) \cap \partial h(x) \neq \emptyset. \quad (6.1.6)$$

Nous dirons donc que x^* est un *point critique de f* s'il satisfait (6.1.6).

6.2 Algorithme de point proximal

Dans cette section, nous démontrons notre méthode en établissant deux lemmes, en décrivant l'algorithme et en faisant plusieurs remarques.

6.2 Algorithme de point proximal

La clé de cette méthode est que chaque itération sera décomposée en deux pas distincts et chacun d'entre eux prendra en compte la convexité de la fonction. Grosso modo, cette méthode consiste à augmenter la fonction h le long de la direction d'un sous-gradient et ensuite à diminuer la fonction g par un pas proximal. En effet, toute direction du sous-gradient d'une fonction convexe est une direction ascendante de la fonction.

Lemme 6.2.1. Soit $h \in \Gamma_0$ et $x \in \mathbb{R}^n$. Alors $\forall w \in \partial h(x)$ avec $w \neq 0$ et $\forall c_k > c > 0$, nous avons

$$h(x + c_k w) > h(x).$$

Preuve. C'est une conséquence immédiate de la définition du sous-gradient :

$$h(x + c_k w) \geq h(x) + \underbrace{\langle w, c_k w \rangle}_{>0} \quad \forall w \in \partial h(x),$$

i.e.,

$$h(x + c_k w) > h(x) \quad \forall w \in \partial h(x).$$

□

Le lemme suivant donne une condition nécessaire et suffisante pour que x soit un point critique de la fonction DC f . Il n'est pas difficile de voir que tout point critique d'une fonction DC f peut être considéré comme un point fixe d'un certain opérateur.

Lemme 6.2.2. Une condition nécessaire et suffisante pour que x soit un point critique de f est que

$$x = (I + c_k \partial g)^{-1}(x + c_k w) \stackrel{\Delta}{=} \text{prox}_{c_k g}(x + c_k w) \quad (6.2.7)$$

pour tout $c_k > c > 0$ et $w \in \partial h(x)$.

Preuve. Soit x un point critique de f , i.e., tel que x satisfait $\partial g(x) \cap \partial h(x) \neq \emptyset$. D'où

$$\exists w \neq 0 \text{ t.q. } w \in \partial g(x) \cap \partial h(x).$$

6.2 Algorithme de point proximal

Ainsi

$$\begin{aligned} w \in \partial g(x) &\iff c_k w \in c_k \partial g(x) \\ &\iff x + c_k w \in x + c_k \partial g(x) \\ &\iff x + c_k w \in (I + c_k \partial g)(x). \end{aligned}$$

Puisque • ∂g est un opérateur monotone maximal,
 • $(I + c_k \partial g)^{-1}$ est univoque,

on a

$$\begin{aligned} x + c_k w &\in (I + c_k \partial g)(x) \\ \iff x &= (I + c_k \partial g)^{-1}(x + c_k w) \quad \forall c_k > c > 0, \quad \forall w \in \partial h(x). \end{aligned}$$

Et vice versa. □

Soit l'opérateur

$$P_k = (I + c_k \partial g)^{-1}.$$

Le lemme 6.2.2 implique que $x = P_k(x + c_k w)$ si et seulement si x est un point critique de f . Notons que P_k est une application proximale et par conséquent on obtient immédiatement une itération du type point proximal :

$$x^{k+1} = P_k(x^k + c_k w^k), \text{ où } P_k = (I + c_k \partial g)^{-1}. \quad (6.2.8)$$

Nous pouvons à présent énoncer l'algorithme de point proximal pour la minimisation de fonctions DC.

Algorithme de Point Proximal pour les fonctions DC

Pas 1 : Choisir un point de départ x^0 et $c_0 > c > 0$. Poser $k = 0$.

Pas 2 : Calculer $w^k \in \partial h(x^k)$ et $y^k = x^k + c_k w^k$.

Pas 3 : Calculer $x^{k+1} = \text{prox}_{c_k g}(y^k) = (I + c_k \partial g)^{-1}(y^k)$.
 (Pas Proximal)

Pas 4 : Si $x^{k+1} = x^k$: STOP.
 Sinon augmenter k de 1 et retourner au Pas 2.

6.3 Convergence de l'algorithme de point proximal

Remarque 6.2.1. 1. Dans le cas convexe ($h \equiv 0$), cet algorithme est l'algorithme de point proximal classique.

2. Comme toute méthode de type proximal, le comportement numérique de la fonction dépend de la complexité relative du calcul du point proximal. Ici, le pas 3 signifie que nous devons résoudre

$$\min_{x \in \mathbb{R}^n} \left\{ g(x) + \frac{1}{2c_k} \|x - y^k\|^2 \right\},$$

qui peut être résolu, par exemple, par un algorithme de plan sécant ou une méthode faisceau. Remarquons que c_k est modifié à chaque itération.

6.3 Convergence de l'algorithme de point proximal

Dans cette section, nous établissons la convergence de l'algorithme de point proximal. Nous commençons par montrer que l'algorithme de point proximal est un algorithme de descente.

Théorème 6.3.1. La suite $\{x^k\}$ engendrée par l'algorithme de point proximal satisfait

- soit l'algorithme s'arrête en un point critique de f ;
- soit f décroît strictement, i.e., $f(x^{k+1}) < f(x^k)$.

Preuve. Si $x^{k+1} = x^k$, alors par le lemme 6.2.2, x^k est un point critique de f . En effet, par l'algorithme de point proximal, on a :

$$\begin{aligned} x^{k+1} &= (I + c_k \partial g)^{-1}(x^k + c_k w^k) \\ \iff x^k &= (I + c_k \partial g)^{-1}(x^k + c_k w^k) \\ \iff x^k + c_k w^k &\in (I + c_k \partial g)(x^k) \\ \iff x^k + c_k w^k &\in x^k + c_k \partial g(x^k) \\ \iff w^k &\in \partial g(x^k). \end{aligned}$$

6.3 Convergence de l'algorithme de point proximal

Or

$$w^k \in \partial h(x^k) \quad (\text{par l'algorithme de point proximal}),$$

d'où

$$w^k \in \partial g(x^k) \cap \partial h(x^k)$$

implique que

$$\partial g(x^k) \cap \partial h(x^k) \neq \emptyset$$

et donc que x^k est un point critique de f .

Supposons maintenant que $x^{k+1} \neq x^k$. En utilisant la définition du sous-gradient, nous pouvons réécrire l'itération (6.2.8) de la façon suivante :

$$\begin{aligned} (6.2.8) \quad &\iff x^{k+1} = (I + c_k \partial g)^{-1}(x^k + c_k w^k) \\ &\iff x^k + c_k w^k \in x^{k+1} + c_k \partial g(x^{k+1}) \\ &\iff c_k^{-1}(x^k - x^{k+1}) + w^k \in \partial g(x^{k+1}) \\ &\iff g(x^k) \geq g(x^{k+1}) + \langle c_k^{-1}(x^k - x^{k+1}) + w^k, x^k - x^{k+1} \rangle. \end{aligned} \quad (6.3.9)$$

D'un autre côté, puisque $w^k \in \partial h(x^k)$, nous avons

$$h(x^{k+1}) \geq h(x^k) + \langle w^k, x^{k+1} - x^k \rangle. \quad (6.3.10)$$

En soustrayant (6.3.10) de (6.3.9), nous obtenons

$$\begin{aligned} &f(x^k) \geq f(x^{k+1}) + \langle c_k^{-1}(x^k - x^{k+1}) + w^k, x^k - x^{k+1} \rangle + \langle w^k, x^{k+1} - x^k \rangle \\ \iff &f(x^k) \geq f(x^{k+1}) + \langle c_k^{-1}(x^k - x^{k+1}), x^k - x^{k+1} \rangle + \langle w^k, x^k - x^{k+1} \rangle \\ &\quad - \langle w^k, x^k - x^{k+1} \rangle \\ \iff &f(x^{k+1}) \leq f(x^k) - \underbrace{c_k^{-1}}_{>0} \underbrace{\|x^k - x^{k+1}\|^2}_{>0 \text{ car } x^{k+1} \neq x^k}. \end{aligned}$$

Par conséquent, nous concluons que $f(x^{k+1}) < f(x^k)$. □

Dans le théorème qui suit, nous utilisons le théorème de convergence de Zangwill [18] pour prouver la convergence de notre algorithme.

Théorème 6.3.2. *Supposons que les suites $\{x^k\}$ et $\{y^k\}$ engendrées par l'algorithme de point proximal soient bornées. Alors toute sous-suite convergente de $\{x^k\}$ converge vers un point critique de f .*

6.4 Quelques applications

Preuve. D'après Zangwill, la convergence globale d'un algorithme dépend de trois propriétés de la suite itérative : la descente, le caractère fermé et le caractère borné. Nous regardons à présent en détail ces trois propriétés. Soit S l'ensemble des points critiques de f .

1. f est une fonction de descente en dehors de S . En effet, le théorème 6.3.1 garantit que $f(x^{k+1}) < f(x^k)$ pour tout x tel que $x^k \neq x^{k+1}$. D'où, par le lemme 6.2.2, $x^k \notin S$. Evidemment, si $x^k \in S$, alors $f(x^k) = f(x^{k+1})$.
2. L'application algorithmique est fermée. En fait, l'algorithme (le pas proximal) peut être écrit comme $x^{k+1} \in B \circ C(x^k)$ avec $B = (I + c_k \partial g)^{-1}$ et $C = I + c_k \partial h$. Notons que B est l'opérateur obtenu à partir de ∂g , d'où son graphe est fermé. En outre, puisque h est une fonction propre, convexe, s.c.i., le graphe de C est aussi fermé. Par conséquent, la suite $\{y^k\}$ avec $y^k \in C(x^k)$ étant bornée par hypothèse, possède une sous-suite convergente $\{y^{k_i}\}$, et alors l'application $B \circ C$ est fermée (cf. le théorème sur la composition d'applications fermées multivoques dans Zangwill [18]).
3. La suite $\{x^k\}$ est bornée par hypothèse.

Par conséquent, en utilisant le théorème de Zangwill, toute sous-suite convergente de $\{x^k\}$ converge vers un point critique de f dans S . \square

6.4 Quelques applications

6.4.1 Application à la programmation concave

Nous considérons le problème de maximisation d'une fonction h convexe s.c.i. sur un ensemble convexe fermé C de \mathbb{R}^n , i.e.,

$$\max_{x \in C} h(x).$$

Ce problème a reçu une grande attention durant les deux dernières décennies et peut être réécrit facilement comme un problème de programmation DC en introduisant une fonction indicatrice I_C sur l'ensemble C :

$$\min_{x \in \mathbb{R}^n} \{I_C(x) - h(x)\}.$$

L'algorithme de point proximal pour les fonctions DC prend la forme suivante :

$$x^{k+1} = Proj_C(x^k + c_k w^k), \quad \text{où } w^k \in \partial h(x^k).$$

6.4 Quelques applications

Plus précisément x^{k+1} résoud l'inégalité variationnelle :

$$\langle c_k^{-1}(x^k - x^{k+1}) + w^k, u - x^{k+1} \rangle \leq 0 \quad \forall u \in C.$$

En effet :

$$\begin{aligned} x^{k+1} &= Proj_C(x^k + c_k w^k) = prox_{I_C}(x^k + c_k w^k) \\ \iff x^{k+1} &= (I + \partial I_C)^{-1}(x^k + c_k w^k) \\ \iff x^k + c_k w^k - x^{k+1} &\in \partial I_C(x^{k+1}) \\ \iff \underbrace{I_C(u)}_{=0} &\geq \underbrace{I_C(x^{k+1})}_{=0} + \langle x^k + c_k w^k - x^{k+1}, u - x^{k+1} \rangle \quad \forall u \in C \\ \iff \langle c_k^{-1}(x^k - x^{k+1}) + w^k, u - x^{k+1} \rangle &\leq 0 \quad \forall u \in C. \end{aligned}$$

Considérons maintenant le cas où la fonction h est différentiable. Dans ce contexte l'algorithme de point proximal se réduit à

$$x^{k+1} = prox_{c_k g}(x^k + c_k \nabla h(x^k))$$

et nous avons le résultat suivant :

Proposition 6.4.1. *Si l'algorithme de point proximal converge vers un point x^∞ qui admet un voisinage, $\mathcal{V}(x^\infty)$, dans lequel la fonction f est convexe et la fonction h est différentiable avec un gradient Lipschitzien, et si nous supposons que la suite $\{c_k\}$ est bornée, alors x^∞ est un minimum local de f .*

Preuve. Par le théorème 7.2.2¹, nous avons

$$\lim_{k \rightarrow \infty} c_k^{-1} \|x^k - x^{k+1}\| = 0,$$

i.e.,

$$\underbrace{c_k^{-1} \|x^k - x^{k+1}\|}_{\rightarrow 0} \cdot \underbrace{c_k}_{\text{borné}} \rightarrow 0,$$

d'où $\lim_{k \rightarrow \infty} \|x^k - x^{k+1}\| = 0$ qui, avec le fait que ∇h est continuellement Lipschitzien sur $\mathcal{V}(x^\infty)$, donne, pour k suffisamment grand, que

$$\lim_{k \rightarrow \infty} \|\nabla h(x^k) - \nabla h(x^{k+1})\| = 0. \quad (*)$$

¹cf. chapitre 7

6.4 Quelques applications

Nous avons aussi

$$c_k^{-1}(x^k - x^{k+1}) + \nabla h(x^k) \in \partial g(x^{k+1}),$$

qui peut être réécrit comme²

$$c_k^{-1}(x^k - x^{k+1}) + \nabla h(x^k) - \nabla h(x^{k+1}) \in \partial g(x^{k+1}) - \nabla h(x^{k+1}) \subset \partial(g-h)(x^{k+1}),$$

i.e., en d'autres termes, $\forall x \in \mathcal{V}(x^\infty)$

$$f(x) \geq f(x^{k+1}) + \langle c_k^{-1}(x^k - x^{k+1}) + \nabla h(x^k) - \nabla h(x^{k+1}), x - x^{k+1} \rangle. \quad (6.4.11)$$

D'après ce que nous avons obtenu avant, nous avons

$$\lim_{k \rightarrow \infty} \underbrace{(c_k^{-1}(x^k - x^{k+1}))}_{\rightarrow 0 \text{ quand } k \rightarrow \infty} + \underbrace{(\nabla h(x^k) - \nabla h(x^{k+1}))}_{\rightarrow 0 \text{ quand } k \rightarrow \infty \text{ par } (*)} = 0,$$

et en prenant en compte le fait que f est s.c.i., ce qui assure que

$$\liminf_{k \rightarrow \infty} f(x^k) \geq f(x^\infty),$$

nous obtenons, en passant à la limite inférieure dans (6.4.11), que

$$\forall x \in \mathcal{V}(x^\infty) \quad f(x) \geq \liminf_{k \rightarrow \infty} f(x^{k+1}) \geq f(x^\infty),$$

d'où x^∞ est un minimum local de f . □

6.4.2 Application à la valeur propre maximale

Soit Q une matrice symétrique définie positive. Chaque valeur propre de Q est un nombre réel positif satisfaisant la condition suivante :

$$\exists x \in \mathbb{R}^n, x \neq 0, \text{ et un nombre réel positif } \lambda, \text{ tels que } Qx = \lambda x.$$

Alors nous avons

$$\langle x, \lambda x \rangle = \langle x, Qx \rangle \iff \lambda = \frac{x^T}{\|x\|} Q \frac{x}{\|x\|},$$

et donc, la valeur propre maximale de Q peut être atteinte en résolvant le problème

$$\max_{\|x\| \leq 1} \langle x, Qx \rangle.$$

²On admet ici que $\partial g - \nabla h \subset \partial(g-h)$

6.4 Quelques applications

En utilisant la même raison et la même stratégie que celles expliquées dans l'application précédente, nous obtenons comment l'algorithme de point proximal s'applique et

$$x^{k+1} = \frac{(I + cQ)x^k}{\|(I + cQ)x^k\|},$$

qui souligne que, dans ce cas, l'algorithme de point proximal se comporte comme la méthode itérative pour déterminer la valeur propre maximale de la matrice Q , où c est un nombre positif petit qui convient.

Chapitre 7

Méthode faisceau pour la programmation DC

7.1 Méthode de point proximal approximée

Très souvent le problème de trouver $\text{prox}_{c_k g}(y^k)$, i.e., de résoudre

$$\min_{y \in \mathbb{R}^n} \left\{ g(y) + \frac{1}{2c_k} \|y - y^k\|^2 \right\} \quad (7.1.1)$$

est aussi difficile que de résoudre le problème initial. Ici nous présentons une implémentation de ce pas basée sur l'hypothèse qu'au point y^k , seules la valeur $g(y^k)$ et le sous-gradient $s(y^k) \in \partial g(y^k)$ sont disponibles grâce à une procédure. Dans ce chapitre, nous supposons que la fonction g est une fonction convexe à valeurs finies. La stratégie utilisée est la suivante : à chaque itération, la fonction g est remplacée par une fonction convexe plus simple \hat{g}_k de telle façon que les sous-problèmes

$$\min_{y \in \mathbb{R}^n} \left\{ \hat{g}_k(y) + \frac{1}{2c_k} \|y - y^k\|^2 \right\} \quad (7.1.2)$$

soient plus faciles à résoudre et que la convergence de la suite des minima soit préservée. Par exemple, si la fonction \hat{g}_k est une fonction convexe linéaire par morceaux de la forme

$$\hat{g}_k(y) = \max_{1 \leq j \leq m} \{ a_j^T y + b_j \}$$

7.1 Méthode de point proximal approximée

où, pour tout j , $a_j \in \mathbb{R}^n$ et $b_j \in \mathbb{R}$, alors le sous-problème (7.1.2) est équivalent au problème quadratique convexe

$$\begin{cases} \min & v + (1/2c_k) \|y - y^k\|^2 \\ \text{s.c.} & a_j^T y + b_j \leq v. \end{cases}$$

Il existe des méthodes très efficaces pour résoudre un tel problème. Maintenant prenons y^{k+1} la solution du sous-problème (7.1.2). Si la décroissance $g(y^k) - g(y^{k+1})$ est suffisante, i.e., si

$$g(y^k) - g(y^{k+1}) \geq m[g(y^k) - \hat{g}_k(y^{k+1})], \quad (7.1.3)$$

où $0 < m < 1$, alors x^{k+1} est mis à y^{k+1} . Sinon, la fonction modèle \hat{g}_k est "améliorée". La condition (7.1.3) est similaire à celle utilisée dans les algorithmes de région de confiance. La décroissance réelle sur g , i.e., $g(y^k) - g(y^{k+1})$ est comparée avec la décroissance prédite par le modèle, à savoir $g(y^k) - \hat{g}_k(y^{k+1})$.

Dans une première sous-section, nous décrivons un algorithme général dans lequel une approximation de g est utilisée à chaque itération de l'algorithme de point proximal. Dans une seconde sous-section, nous présentons une méthode pour construire les approximations de g .

7.1.1 Un algorithme général

Dans cette sous-section, nous considérons l'algorithme suivant.

7.1 Méthode de point proximal approximée

Algorithme Général

Soit $\{c_k\}_{k \in \mathbb{N}_0}$ une suite de nombres positifs.

Choisir un point de départ x^0 et poser $k = 0$.

Pas 1 : Calculer $w^k \in \partial h(x^k)$ et $y^k = x^k + c_k w^k$.

Pas 2 : Choisir \hat{g}_k une approximation de g en y^k et noter x^{k+1} l'unique solution du problème

$$\min_{y \in \mathbb{R}^n} \left\{ \hat{g}_k(y) + \frac{1}{2c_k} \|y - y^k\|^2 \right\}.$$

Pas 3 : Augmenter k de 1 et retourner au Pas 1.

Par optimalité de x^{k+1} , nous avons

$$0 \in \partial \hat{g}_k(x^{k+1}) + \frac{1}{c_k} (x^{k+1} - y^k),$$

i.e.,

$$\gamma^k \equiv \frac{1}{c_k} (y^k - x^{k+1}) \in \partial \hat{g}_k(x^{k+1}).$$

Donc la pente prédite par le modèle \hat{g}_k , à savoir $g(y^k) - \hat{g}_k(x^{k+1})$, est toujours positive. En effet, puisque $g \geq \hat{g}_k$ et $\gamma^k \in \partial \hat{g}_k(x^{k+1})$, nous avons

$$g(y^k) - \hat{g}_k(x^{k+1}) \geq \hat{g}_k(y^k) - \hat{g}_k(x^{k+1}) \geq \langle \gamma^k, y^k - x^{k+1} \rangle = c_k \|\gamma^k\|^2 > 0$$

si l'on suppose que $x^{k+1} \neq y^k$.

7.1.2 Construction des approximations de g

Afin d'obtenir un algorithme implémentable, nous devons maintenant regarder comment construire une approximation \hat{g}_k de g en y^k telle que le sous-problème (7.1.2) soit plus facile à résoudre que le problème (7.1.1). Nous savons déjà que si \hat{g}_k est une fonction convexe linéaire par morceaux, alors le sous-problème (7.1.2) est équivalent à un problème quadratique convexe et des méthodes numériques efficaces existent pour le résoudre. Donc, si nous voulons obtenir une fonction convexe linéaire par morceaux

7.1 Méthode de point proximal approximée

pour la fonction modèle \hat{g}_k , il est judicieux de la construire morceau par morceau en générant les modèles successifs

$$\hat{g}_i^k, \quad i = 1, 2, \dots$$

jusqu'à ce que (si possible) $\hat{g}_{i_k}^k$ soit une approximation de g en y^k pour un certain $i_k \geq 1$. Pour $i = 1, 2, \dots$, nous notons par y_i^k l'unique solution du problème

$$(P_i^k) \quad \min_y \left\{ \hat{g}_i^k(y) + (1/2c_k) \|y - y^k\|^2 \right\},$$

et nous prenons $\hat{g}_k = \hat{g}_{i_k}^k$ et $x^{k+1} = y_{i_k}^k$.

Afin d'obtenir une approximation $\hat{g}_{i_k}^k$ de g en y^k , nous devons imposer certaines conditions sur les modèles successifs \hat{g}_i^k , $i = 1, 2, \dots$. Cependant, avant de les présenter, nous avons besoin de définir les fonctions affines l_i^k , $i = 1, 2, \dots$ par

$$l_i^k(y) = \hat{g}_i^k(y_i^k) + \langle \gamma_i^k, y - y_i^k \rangle \quad \forall y \in \mathbb{R}^n,$$

où $\gamma_i^k = \frac{1}{c_k}(y^k - y_i^k)$. Par optimalité de y_i^k , nous avons

$$\gamma_i^k \in \partial \hat{g}_i^k(y_i^k). \quad (7.1.4)$$

Il est alors facile d'observer que

$$l_i^k(y_i^k) = \hat{g}_i^k(y_i^k) \quad \text{et} \quad l_i^k(y) \leq \hat{g}_i^k(y) \quad \text{pour tout } y \in \mathbb{R}^n. \quad (7.1.5)$$

Maintenant, nous supposons que les conditions suivantes sont satisfaites par les modèles convexes \hat{g}_i^k :

- (C1) $\hat{g}_i^k \leq g$ sur \mathbb{R}^n pour $i = 1, 2, \dots$
 - (C2) $\hat{g}_{i+1}^k \geq g(y_i^k) + \langle s(y_i^k), \cdot - y_i^k \rangle$ sur \mathbb{R}^n pour $i = 1, 2, \dots$
 - (C3) $\hat{g}_{i+1}^k \geq l_i^k$ sur \mathbb{R}^n pour $i = 1, 2, \dots$,
- où $s(y_i^k)$ désigne le sous-gradient de g disponible en y_i^k .

Plusieurs modèles remplissent ces conditions. Par exemple, pour le premier modèle \hat{g}_1^k , nous pouvons prendre la fonction linéaire

$$\hat{g}_1^k(y) = g(y^k) + \langle s(y^k), y - y^k \rangle \quad \forall y \in \mathbb{R}^n.$$

7.1 Méthode de point proximal approximée

Puisque $s(y^k) \in \partial g(y^k)$, la condition (C1) est satisfaite pour $i = 1$. Pour les modèles suivants \hat{g}_i^k , $i = 2, \dots$, il existe plusieurs possibilités. Un premier exemple est de prendre pour $i = 1, 2, \dots$

$$\hat{g}_{i+1}^k(y) = \max \{l_i^k(y), g(y_i^k) + \langle s(y_i^k), y - y_i^k \rangle\} \quad \forall y \in \mathbb{R}^n. \quad (7.1.6)$$

Les conditions (C2), (C3) sont évidemment satisfaites et la condition (C1) est également satisfaite parce que chaque morceau linéaire de ces fonctions est en-dessous de g .

Un autre exemple consiste à prendre pour $i = 1, 2, \dots$

$$\hat{g}_{i+1}^k(y) = \max_{0 \leq j \leq i} \{g(y_j^k) + \langle s(y_j^k), y - y_j^k \rangle\} \quad \forall y \in \mathbb{R}^n, \quad (7.1.7)$$

où $y_0^k = y^k$. Puisque $s(y_j^k) \in \partial g_k(y_j^k)$ pour $j = 0, \dots, i$ et puisque $\hat{g}_{i+1}^k \geq \hat{g}_i^k \geq l_i^k$, il est facile de voir que les conditions (C1)-(C3) sont satisfaites. Soulignons que la fonction \hat{g}_k , définie en (7.1.7), est appelée la fonction approximante de plan sécant classique de g . En comparant (7.1.6) et (7.1.7), nous pouvons dire que l_i^k joue le même rôle que les i fonctions linéaires $g_k(y_j^k) + \langle s(y_j^k), y - y_j^k \rangle$, $j = 0, \dots, i - 1$. C'est la raison pour laquelle cette fonction l_i^k est appelée fonction affine totale. A présent, l'algorithme permettant de passer de x^k à x^{k+1} , i.e., de faire ce que l'on appelle un pas sérieux, peut être exprimé comme suit :

Algorithme de Pas Sérieux

Soit $y^k \in \mathbb{R}^n$ et une tolérance $\epsilon_k > 0$. Poser $i = 1$.

Pas 1 : Choisir \hat{g}_i^k une fonction convexe qui satisfait (C1)-(C3) et résoudre (P_i^k) pour obtenir y_i^k .

Pas 2 : Si $g(y_i^k) - \hat{g}_i^k(y_i^k) \leq \epsilon_k$, alors mettre $x^{k+1} = y_i^k$, $i_k = i$ et STOP ; x^{k+1} est un pas sérieux.

Pas 3 : Augmenter i de 1 et retourner au Pas 1.

Précisons que \hat{g}_i^k satisfait (C1)-(C3) signifie que \hat{g}_i^k satisfait (C1) et, si $i \geq 2$, \hat{g}_i^k satisfait (C2) et (C3) avec $i + 1$ remplacé par i .

7.1 Méthode de point proximal approximée

Preuve. En utilisant les définitions de \tilde{l}_i^k , l_i^k et γ_i^k ,
i.e.,

$$\tilde{l}_i^k(y) = l_i^k(y) + \frac{1}{2c_k} \|y - y^k\|^2 \quad (7.1.8)$$

$$l_i^k(y) = \hat{g}_i^k(y_i^k) + \langle \gamma_i^k, y - y_i^k \rangle \quad (7.1.9)$$

$$\gamma_i^k = \frac{1}{c_k} (y^k - y_i^k), \quad (7.1.10)$$

nous avons

$$\begin{aligned} \tilde{l}_i^k(y_i^k) + \frac{1}{2c_k} \|y - y_i^k\|^2 &\stackrel{(7.1.8)}{=} l_i^k(y_i^k) + \frac{1}{2c_k} \|y_i^k - y^k\|^2 + \frac{1}{2c_k} \|y - y_i^k\|^2 \\ &\stackrel{(7.1.5)}{=} \hat{g}_i^k(y_i^k) + \frac{1}{2c_k} \|y_i^k - y^k\|^2 + \frac{1}{2c_k} \|y - y_i^k\|^2 \\ &= \hat{g}_i^k(y_i^k) + \frac{1}{2c_k} \|y - y^k\|^2 - \frac{1}{c_k} \langle y_i^k - y^k, y - y_i^k \rangle \\ &\stackrel{(7.1.10)}{=} \hat{g}_i^k(y_i^k) + \frac{1}{2c_k} \|y - y^k\|^2 + \langle \gamma_i^k, y - y_i^k \rangle \\ &\stackrel{(7.1.9)}{=} l_i^k(y) + \frac{1}{2c_k} \|y - y^k\|^2 \\ &\stackrel{(7.1.8)}{=} \tilde{l}_i^k(y). \end{aligned}$$

□

Nous démontrons à présent la proposition 7.1.1.

Preuve.

1. $\{\tilde{l}_i^k(y_i^k)\}$ est convergente et $y_{i+1}^k - y_i^k \rightarrow 0$ quand $i \rightarrow +\infty$.

Pour tout $i \geq 1$, nous avons

$$\begin{aligned} g(y^k) &\geq \hat{g}_{i+1}^k(y^k) = \tilde{g}_{i+1}^k(y^k) && \text{(car } g \geq \hat{g}_{i+1}^k) \\ &\geq \tilde{g}_{i+1}^k(y_{i+1}^k) && \text{(définition de } y_{i+1}^k) \\ &= \tilde{l}_{i+1}^k(y_{i+1}^k) && \text{(définition de } \tilde{l}_{i+1}^k) \\ &\geq \tilde{l}_i^k(y_{i+1}^k) && \text{(car par (C3) } \hat{g}_{i+1}^k \geq l_i^k) \\ &= \tilde{l}_i^k(y_i^k) + \frac{1}{2c_k} \|y_{i+1}^k - y_i^k\|^2 && \text{(par le lemme avec } y = y_{i+1}^k). \end{aligned}$$

7.1 Méthode de point proximal approximée

De ces relations, nous déduisons que

- la suite $\{\tilde{l}_i^k(y_i^k)\}$ est croissante (car $\tilde{l}_i^k(y_i^k) \leq \tilde{l}_{i+1}^k(y_i^k)$) et bornée supérieurement par $g(y^k)$; elle est donc convergente.
- $\tilde{l}_{i+1}^k(y_{i+1}^k) - \tilde{l}_i^k(y_i^k) \geq (1/2c_k) \|y_{i+1}^k - y_i^k\|^2 \geq 0$, ce qui implique que $y_{i+1}^k - y_i^k \rightarrow 0$ car le membre gauche de l'inégalité précédente tend vers 0.

2. La suite $\{y_i^k\}_i$ est bornée.

Soit $y \in \mathbb{R}^n$. Alors, puisque $g \geq l_i^k$, il suit par le lemme que

$$\underbrace{g(y)}_{\geq l_i^k(y)} + \frac{1}{2c_k} \|y - y^k\|^2 \geq \tilde{l}_i^k(y) \stackrel{\text{lemme}}{=} \tilde{l}_i^k(y_i^k) + \frac{1}{2c_k} \|y - y_i^k\|^2.$$

Puisque $\{\tilde{l}_i^k(y_i^k)\}$ est convergente, la suite $\{y - y_i^k\}$ doit être bornée et donc aussi la suite $\{y_i^k\}$.

3. $g(y_i^k) - \hat{g}_i^k(y_i^k) \rightarrow 0$ quand $i \rightarrow +\infty$.

Nous avons successivement

$$\begin{aligned} g(y_{i+1}^k) - g(y_i^k) &\geq \hat{g}_{i+1}^k(y_{i+1}^k) - g(y_i^k) \quad (\text{car } g \geq \hat{g}_{i+1}^k) \\ &\geq \langle s(y_i^k), y_{i+1}^k - y_i^k \rangle \quad (\text{condition (C2)}). \end{aligned}$$

Aussi, puisque g est continuellement Lipschitzienne sur des ensembles bornés, nous obtenons

$$|g(y_{i+1}^k) - g(y_i^k)| \leq L \underbrace{\|y_{i+1}^k - y_i^k\|}_{\rightarrow 0}.$$

Donc

$$g(y_{i+1}^k) - g(y_i^k) \rightarrow 0 \quad (\text{car } y_{i+1}^k - y_i^k \rightarrow 0). \quad (7.1.11)$$

D'un autre côté, la suite $\{s(y_i^k)\}_i$ est bornée parce que $\{y_i^k\}_i$ est bornée et le sous-différentiel est borné sur des ensembles bornés.

Donc, puisque $y_{i+1}^k - y_i^k \rightarrow 0$, nous avons

$$\langle s(y_i^k), y_{i+1}^k - y_i^k \rangle \rightarrow 0. \quad (7.1.12)$$

7.1 Méthode de point proximal approximée

D'où nous avons

$$\underbrace{g(y_{i+1}^k) - g(y_i^k)}_{\rightarrow 0 \text{ par (7.1.11)}} \geq \hat{g}_{i+1}^k(y_{i+1}^k) - g(y_i^k) \geq \underbrace{\langle s(y_i^k), y_{i+1}^k - y_i^k \rangle}_{\rightarrow 0 \text{ par (7.1.12)}}$$

i.e.,

$$\hat{g}_{i+1}^k(y_{i+1}^k) - g(y_i^k) \longrightarrow 0 \quad (7.1.13)$$

et

$$\hat{g}_{i+1}^k(y_{i+1}^k) - g(y_{i+1}^k) = \underbrace{\hat{g}_{i+1}^k(y_{i+1}^k) - g(y_i^k)}_{\rightarrow 0 \text{ par (7.1.13)}} + \underbrace{g(y_i^k) - g(y_{i+1}^k)}_{\rightarrow 0 \text{ par (7.1.11)}} \longrightarrow 0.$$

4. $y_i^k \longrightarrow \text{prox}_{c_k g}(y^k)$ quand $i \rightarrow +\infty$.

Puisque la suite $\{y_i^k\}_i$ est bornée, il suffit de prouver que chaque valeur d'adhérence \bar{y}^k de $\{y_i^k\}$ est égale à $\text{prox}_{c_k g}(y^k)$, i.e., par la caractérisation de $\text{prox}_{c_k g}(y^k)$ (voir proposition 5.2.1), que

$$\frac{y^k - \bar{y}^k}{c_k} \in \partial g(\bar{y}^k).$$

Nous avons $y_i^k \rightarrow \bar{y}^k$ pour $i \in K \subseteq \mathbb{N}$ (avec K infini).

Donc, pour tout $i \in K$ et pour tout $y \in \mathbb{R}^n$, il suit de la définition du sous-différentiel que

$$g(y) \stackrel{(C1)}{\geq} \hat{g}_i^k(y) \stackrel{(7.1.5)}{\geq} l_i^k(y) = \hat{g}_i^k(y_i^k) + \langle \gamma_i^k, y - y_i^k \rangle. \quad (\star)$$

Puisque $g(y_i^k) \rightarrow g(\bar{y}^k)$ (car $y_i^k \rightarrow \bar{y}^k$) et $\hat{g}_i^k(y_i^k) - g(y_i^k) \rightarrow 0$ (par le premier résultat de cette proposition), nous avons que

$$\hat{g}_i^k(y_i^k) \rightarrow g(\bar{y}^k).$$

En outre,

$$\gamma_i^k = \frac{1}{c_k}(y^k - y_i^k) \rightarrow \frac{1}{c_k}(y^k - \bar{y}^k) \equiv \bar{\gamma}.$$

D'où, en passant à la limite dans (\star) , nous obtenons

$$\forall y \in \mathbb{R}^n \quad g(y) \geq \lim_{i \rightarrow \infty} \hat{g}_i^k(y_i^k) + \lim_{i \rightarrow \infty} \langle \gamma_i^k, y - y_i^k \rangle,$$

7.2 Algorithme de point proximal approximé

i.e.,

$$\forall y \in \mathbb{R}^n \quad g(y) \geq g(\bar{y}^k) + \langle \bar{\gamma}, y - \bar{y}^k \rangle,$$

i.e.,

$$\bar{\gamma} \in \partial g(\bar{y}^k) \text{ ou encore } \frac{y^k - \bar{y}^k}{c_k} \in \partial g(\bar{y}^k).$$

□

Théorème 7.1.1. *Si y^k n'est pas un minimum de g , alors l'algorithme de pas sérieux s'arrête après un nombre fini d'itérations i_k avec $\hat{g}_{i_k}^k$ une approximation de g en y^k et avec $x^{k+1} = y_{i_k}^k$.*

Preuve. Supposons — dans le but d'obtenir une contradiction — que le STOP ne se produit jamais dans l'algorithme de pas sérieux. Alors

$$\forall i \geq 1 \quad g(y_i^k) - \hat{g}_i^k(y_i^k) > \epsilon_k. \quad (7.1.14)$$

Par la proposition 7.1.1, nous obtenons que $y_i^k \rightarrow \text{prox}_{c_k g}(y^k)$ et $\hat{g}_i^k(y_i^k) - g(y_i^k) \rightarrow 0$ quand $i \rightarrow +\infty$. D'où, en passant à la limite dans (7.1.14), nous avons

$$g(\text{prox}_{c_k g}(y^k)) - g(\text{prox}_{c_k g}(y^k)) \geq \epsilon_k,$$

i.e., $0 \geq \epsilon_k$, ce qui contredit l'hypothèse que $\epsilon_k > 0$. Donc l'algorithme de pas sérieux s'arrête après un nombre fini d'itérations. □

7.2 Algorithme de point proximal approximé

A l'aide de la définition de sous-gradient à ϵ près, nous établissons, dans cette section, une version approximée de l'algorithme de point proximal pour la minimisation d'une fonction DC et analysons sa propriété de convergence. Notre analyse ressemble à celle proposée au chapitre 6 pour trouver un point critique d'une fonction DC, à la grande différence que nous traitons des ϵ -sous-gradients. En effet, nous considérons ici la version approximée obtenue en remplaçant le sous-différentiel exact par le sous-différentiel approximé. Puisque la fonction h (respectivement g) est supposée convexe, propre et s.c.i., $\partial_0 h(x) = \partial h(x)$ pour tout x . De plus, il suit directement de la définition de sous-différentiel

7.2 Algorithme de point proximal approximé

que $0 \leq \epsilon_1 \leq \epsilon_2 \Rightarrow \partial_{\epsilon_1} h(x) = \partial_{\epsilon_2} h(x)$. Donc $\partial_\epsilon h(x)$ est une extension de $\partial h(x)$. L'utilisation d'éléments dans $\partial_\epsilon h(x)$ au lieu de $\partial h(x)$ permet un plus large degré de liberté qui est très utile dans des applications diverses. D'un autre côté, en posant $\epsilon = 0$, on retrouve le sous-différentiel exact; ainsi la méthode exacte peut aussi être traitée.

Nous donnons en premier lieu la définition d' ϵ -point critique d'une fonction DC f . Soit $\epsilon \geq 0$, le point x est un ϵ -point critique de f s'il satisfait

$$\partial_\epsilon g(x) \cap \partial_\epsilon h(x) \neq \emptyset. \quad (7.2.15)$$

Evidemment, lorsque $\epsilon = 0$, (7.2.15) se réduit à (6.1.6).

Lemme 7.2.1. Soit $h \in \Gamma_0$ et $x \in \mathbb{R}^n$. Alors $\forall w \in \partial_\epsilon h(x)$ avec $w \neq 0$ et $\forall c_k > c > 0$, nous avons

$$h(x + c_k w) > h(x) - \epsilon.$$

Preuve. Il suit directement de la définition du ϵ -sous-gradient que

$$h(x + c_k w) \geq h(x) + \underbrace{\langle w, c_k w \rangle}_{>0} - \epsilon \quad \forall w \in \partial_\epsilon h(x),$$

i.e.,

$$h(x + c_k w) > h(x) - \epsilon \quad \forall w \in \partial_\epsilon h(x).$$

□

Le résultat suivant donne une condition nécessaire et suffisante pour que x soit un ϵ -point critique de f .

Lemme 7.2.2. Une condition nécessaire et suffisante pour que x soit un ϵ -point critique de f est que

$$x \in (I + c_k \partial_\epsilon g)^{-1}(x + c_k w) \stackrel{\Delta}{=} \text{prox}_{c_k g}^\epsilon(x + c_k w) \quad (7.2.16)$$

pour tout $c_k > c > 0$ et $w \in \partial_\epsilon h(x)$.

7.2 Algorithme de point proximal approximé

Preuve. Soit x un ϵ -point critique de f , i.e., tel que x satisfait $\partial_\epsilon g(x) \cap \partial_\epsilon h(x) \neq \emptyset$. D'où

$$\exists w \neq 0 \text{ t.q. } w \in \partial_\epsilon g(x) \cap \partial_\epsilon h(x).$$

Ainsi

$$\begin{aligned} w \in \partial_\epsilon g(x) &\iff c_k w \in c_k \partial_\epsilon g(x) \\ &\iff x + c_k w \in x + c_k \partial_\epsilon g(x) \\ &\iff x + c_k w \in (I + c_k \partial_\epsilon g)(x). \end{aligned}$$

Par conséquent

$$x \in (I + c_k \partial_\epsilon g)^{-1}(x + c_k w).$$

Et vice versa. □

Par le lemme ci-dessus, nous obtenons une itération du type ϵ -point proximal :

$$x^{k+1} \in P_{k,\epsilon}(x^k + c_k w^k), \text{ où } P_{k,\epsilon} = (I + c_k \partial_\epsilon g)^{-1}. \quad (7.2.17)$$

A présent, nous pouvons résumer ces idées dans le schéma suivant :

Algorithme de ϵ -Point Proximal pour les fonctions DC

Pas 1 : Choisir un point de départ x^0 et $c_0 \geq c > 0$. Poser $k = 0$.

Pas 2 : Calculer $w^k \in \partial_{\epsilon_k} h(x^k)$ et $y^k = x^k + c_k w^k$.

Pas 3 : Calculer $x^{k+1} = \text{prox}_{c_k g}^{\epsilon_k}(y^k) = (I + c_k \partial_{\epsilon_k} g)^{-1}(y^k)$.
(Pas Proximal)

Pas 4 : Si $x^{k+1} = x^k$: STOP.
Sinon augmenter k de 1 et retourner au Pas 2.

Le théorème suivant indique que l'algorithme de point proximal approximé produit une descente à 2ϵ près de la fonction objectif à chaque itération.

Théorème 7.2.2. *Supposons que $f := g - h$ soit bornée inférieurement, $c_k \geq c > 0 \quad \forall k \in \mathbb{N}$ et supposons que $\sum_{k=0}^{\infty} \epsilon_k < +\infty$. Alors la suite $\{f(x^k)\}_{k \in \mathbb{N}}$ est convergente et la suite $\{x^k\}$ est asymptotiquement régulière dans le sens suivant : $\lim_{k \rightarrow \infty} c_k^{-1} \|x^k - x^{k+1}\| = 0$. De plus, si les suites $\{x^k\}$ et $\{w^k\}$ sont bornées, alors toutes valeurs d'adhérence x^∞ et w^∞ des suites $\{x^k\}$ et $\{w^k\}$ sont des points critiques des fonctions $g - h$ et $h^* - g^*$ respectivement.*

Preuve. Par le Pas 3 de l'algorithme de point proximal à ϵ près pour les fonctions DC, nous avons

$$\begin{aligned} x^{k+1} &= \text{prox}_{c_k g}^{\epsilon_k}(y^k) = (I + c_k \partial_{\epsilon_k} g)^{-1}(x^k + c_k w^k) \\ \iff x^k + c_k w^k &\in (I + c_k \partial_{\epsilon_k} g)(x^{k+1}) \\ \iff c_k^{-1} x^k + w^k &\in c_k^{-1} x^{k+1} + \partial_{\epsilon_k} g(x^{k+1}) \\ \iff c_k^{-1} (x^k - x^{k+1}) + w^k &\in \partial_{\epsilon_k} g(x^{k+1}), \end{aligned}$$

ce qui implique que

$$g(x^k) \geq g(x^{k+1}) + \langle w^k + c_k^{-1}(x^k - x^{k+1}), x^k - x^{k+1} \rangle - \epsilon_k. \quad (7.2.21)$$

Par ailleurs, puisque $w^k \in \partial_{\epsilon_k} h(x^k)$, nous avons

$$h(x^{k+1}) \geq h(x^k) + \langle w^k, x^{k+1} - x^k \rangle - \epsilon_k. \quad (7.2.22)$$

En soustrayant (7.2.22) de (7.2.21), nous obtenons

$$\begin{aligned} f(x^k) &\geq f(x^{k+1}) + \langle c_k^{-1}(x^k - x^{k+1}), x^k - x^{k+1} \rangle - 2\epsilon_k \\ \iff f(x^{k+1}) &\leq f(x^k) - \underbrace{c_k^{-1} \|x^k - x^{k+1}\|^2}_{\geq 0 \text{ car } c_k > 0 \text{ et } \|x^k - x^{k+1}\|^2 \geq 0} + 2\epsilon_k \end{aligned} \quad (7.2.23)$$

qui à son tour implique que

$$f(x^{k+1}) \leq f(x^k) + 2\epsilon_k. \quad (7.2.24)$$

Par conséquent nous avons

$$\begin{aligned} f(x^{k+1}) &\leq f(x^k) + 2\epsilon_k \\ &\leq f(x^{k-1}) + 2\epsilon_{k-1} + 2\epsilon_k \\ &\leq f(x^{k-2}) + 2\epsilon_{k-2} + 2\epsilon_{k-1} + 2\epsilon_k \\ &\vdots \end{aligned}$$

7.2 Algorithme de point proximal approximé

i.e.,

$$f(x^k) \leq f(x^0) + 2 \sum_{i=0}^{k-1} \epsilon_i$$

qui, en considérant l'hypothèse que la série $\sum_{k=0}^{\infty} \epsilon_k < +\infty$ implique que la suite $\{f(x^k)\}$ est bornée supérieurement. D'autre part, f est bornée inférieurement par hypothèse, d'où la suite $\{f(x^k)\}$ est bornée et a donc au moins une valeur d'adhérence (par le théorème de Bolzano-Weierstrass¹). En fait, toute la suite converge. En effet, supposons que $\{f(x^k)\}$ admette deux valeurs d'adhérence f_1^∞ et f_2^∞ telles que $f_1^\infty < f_2^\infty$ et considérons $\{f(x^{\nu_1})\}$ et $\{f(x^{\nu_2})\}$ deux sous-suites convergeant vers f_1^∞ et f_2^∞ respectivement. Posons

$$\tilde{\epsilon} = \frac{f_2^\infty - f_1^\infty}{4} > 0.$$

Alors il existe deux entiers $\tilde{\nu}_1, \tilde{\nu}_2$ tels que

$$f_1^\infty - \tilde{\epsilon} < f(x^{\nu_1}) < f_1^\infty + \tilde{\epsilon} \quad \forall \nu_1 > \tilde{\nu}_1$$

et

$$f_2^\infty - \tilde{\epsilon} < f(x^{\nu_2}) < f_2^\infty + \tilde{\epsilon} \quad \forall \nu_2 > \tilde{\nu}_2.$$

De plus, puisque $\sum_{k=0}^{\infty} \epsilon_k < +\infty$,

$$\exists \tilde{\nu}_3 \in \mathbb{N} \text{ t.q. } \sum_{k=\tilde{\nu}_3}^{+\infty} \epsilon_k < \tilde{\epsilon}.$$

A présent choisissons un entier $\bar{\nu}_1$ tel que $\bar{\nu}_1 \geq \max\{\tilde{\nu}_1, \tilde{\nu}_2, \tilde{\nu}_3\}$. Alors, en vertu de (7.2.24), nous pouvons écrire

$$f(x^{\nu_1}) < f(x^{\bar{\nu}_1}) + 2 \sum_{i=\bar{\nu}_1}^{\nu_1-1} \epsilon_i \quad \forall \nu_1 > \bar{\nu}_1.$$

Or • $f(x^{\bar{\nu}_1}) < f_1^\infty + \tilde{\epsilon}$ car $\nu_1 > \bar{\nu}_1 \geq \tilde{\nu}_1$ et on a que $f(x^{\nu_1}) < f_1^\infty + \tilde{\epsilon}$
 $\forall \nu_1 > \tilde{\nu}_1,$

•

$$\sum_{i=\bar{\nu}_1}^{\nu_1-1} \epsilon_i < \tilde{\epsilon} \text{ car } \sum_{k=\tilde{\nu}_3}^{+\infty} \epsilon_k < \tilde{\epsilon} \text{ et } \bar{\nu}_1 \geq \tilde{\nu}_3.$$

¹Théorème de Bolzano-Weierstrass : Toute suite bornée admet au moins une valeur d'adhérence.

D'où

$$\begin{aligned}
 f(x^{\nu_1}) &< f(x^{\bar{\nu}_1}) + 2 \sum_{i=\bar{\nu}_1}^{\nu_1-1} \epsilon_i \\
 &< f_1^\infty + 3\tilde{\epsilon} \\
 &= f_1^\infty + \frac{3}{4}(f_2^\infty - f_1^\infty) \\
 &= \frac{1}{4}f_1^\infty + \frac{3}{4}f_2^\infty \\
 &= f_2^\infty - \frac{f_2^\infty - f_1^\infty}{4} \\
 &= f_2^\infty - \tilde{\epsilon} \quad \forall \nu_1 > \bar{\nu}_1.
 \end{aligned}$$

Nous avons dès lors

$$f_1^\infty - \tilde{\epsilon} < f(x^{\nu_1}) < f_1^\infty + \tilde{\epsilon} \quad \forall \nu_1 > \bar{\nu}_1$$

et

$$f(x^{\nu_1}) < f_2^\infty - \tilde{\epsilon} \quad \forall \nu_1 > \bar{\nu}_1,$$

ce qui est impossible ! Nous obtenons une contradiction, et par conséquent $\{f(x^k)\}$ admet au plus une valeur d'adhérence ; elle est donc convergente.

En outre, nous déduisons de (7.2.23) que

$$\lim_{k \rightarrow \infty} c_k^{-1} \|x^k - x^{k+1}\|^2 = 0,$$

qui à son tour, puisque $c_k \geq c > 0$, implique que

$$\lim_{k \rightarrow \infty} c_k^{-1} \|x^k - x^{k+1}\| = 0.$$

A présent, considérons deux sous-suites² $\{x^{k_\nu}\}$ et $\{w^{k_\nu}\}$ de $\{x^k\}$ et $\{w^k\}$ convergeant respectivement vers x^∞ et w^∞ . En passant à la limite dans les relations suivantes

$$c_{k_\nu}^{-1}(x^{k_\nu} - x^{k_\nu+1}) + w^{k_\nu} \in \partial_{\epsilon_{k_\nu}} g(x^{k_\nu+1})$$

et

$$w^{k_\nu} \in \partial_{\epsilon_{k_\nu}} h(x^{k_\nu}),$$

²Nous utiliserons la même notation pour les indices même si nous avons besoin d'extraire d'autres sous-suites.

7.3 Une méthode proximale faisceau pour les fonctions DC

et en prenant en compte le fait que les applications multivoques $\partial_{(\cdot)}g(\cdot)$ et $\partial_{(\cdot)}h(\cdot)$ sont fermées sur $\mathbb{R}_+ \times \mathbb{R}^n$, nous obtenons

$$\underbrace{w^{k\nu}}_{\downarrow w^\infty} \in \partial_{\epsilon_{k\nu}} g(\underbrace{x^{k\nu+1}}_{\downarrow x^\infty}) \quad \text{et} \quad \underbrace{w^{k\nu}}_{\downarrow w^\infty} \in \partial_{\epsilon_{k\nu}} h(\underbrace{x^{k\nu}}_{\downarrow x^\infty}),$$

i.e.,

$$w^\infty \in \partial g(x^\infty) \quad \text{et} \quad w^\infty \in \partial h(x^\infty).$$

D'où nous déduisons que $\partial g(x^\infty) \cap \partial h(x^\infty) \neq \emptyset$, i.e., en d'autres termes, que x^∞ est un point critique de $g - h$.

D'autre part, par dualité, nous obtenons par la proposition 2.3.5 que

$$x^\infty \in \partial g^*(w^\infty) \quad \text{et} \quad x^\infty \in \partial h^*(w^\infty),$$

d'où nous déduisons que w^∞ est un point critique de $h^* - g^*$. □

7.3 Une méthode proximale faisceau pour les fonctions DC

En insérant l'algorithme de pas sérieux dans le Pas 2 de l'algorithme général, nous obtenons l'algorithme suivant.

Algorithme Faisceau I pour les fonctions DC

Pas 1 : Choisir un point de départ x^0 , $c_0 \geq c > 0$ et une tolérance $\epsilon_k > 0$. Poser $k = 0$.

Pas 2 : Calculer $w^k \in \partial h(x^k)$ et $y^k = x^k + c_k w^k$.

Pas 3 : Choisir \hat{g}_k une approximation de g en y^k telle que

- $\hat{g}_k \leq g$
- $g(x^{k+1}) - \hat{g}_k(x^{k+1}) \leq \epsilon_k$

où

$$x^{k+1} = \arg \min_{y \in \mathbb{R}^n} \left\{ \hat{g}_k(y) + \frac{1}{2c_k} \|y - y^k\|^2 \right\}.$$

Pas 4 : Mettre $i_k := k$.

Augmenter k de 1 et retourner au Pas 2.

Cet algorithme (algorithme faisceau I) a été introduit afin d'obtenir un algorithme implémentable. En effet, le pas proximal (Pas 3) de l'algorithme de point proximal approximé pour les fonctions DC n'est pas implémentable tandis que le Pas 3 de ce nouvel algorithme bien. Nous allons dès lors montrer que le Pas 3 de l'algorithme faisceau I implique le Pas 3 de l'algorithme de point proximal approximé et donc que ces deux algorithmes sont équivalents. En effet :

– **Algorithme faisceau I**

$$\begin{aligned}
 x^{k+1} &= \arg \min_{y \in \mathbb{R}^n} \left\{ \hat{g}_k(y) + \frac{1}{2c_k} \|y - y^k\|^2 \right\} \\
 \iff 0 &\in \partial \hat{g}_k(x^{k+1}) + \frac{1}{c_k} (x^{k+1} - y^k) \\
 \iff \frac{y^k - x^{k+1}}{c_k} &\in \partial \hat{g}_k(x^{k+1}) \\
 \iff \forall z \in \mathbb{R}^n \quad \underbrace{\hat{g}_k(z)}_{\leq g(z)} &\geq \underbrace{\hat{g}_k(x^{k+1})}_{\geq g(x^{k+1}) - \epsilon_k} + \left\langle \frac{y^k - x^{k+1}}{c_k}, z - x^{k+1} \right\rangle \\
 \iff \forall z \in \mathbb{R}^n \quad g(z) &\geq g(x^{k+1}) - \epsilon_k + \left\langle \frac{y^k - x^{k+1}}{c_k}, z - x^{k+1} \right\rangle \\
 \iff \frac{y^k - x^{k+1}}{c_k} &\in \partial_{\epsilon_k} g(x^{k+1})
 \end{aligned}$$

– **Algorithme de point proximal approximé**

$$\begin{aligned}
 x^{k+1} &= (I + c_k \partial_{\epsilon_k} g)^{-1} (y^k) \\
 \iff y^k &\in (I + c_k \partial_{\epsilon_k} g)(x^{k+1}) \\
 \iff \frac{y^k - x^{k+1}}{c_k} &\in \partial_{\epsilon_k} g(x^{k+1})
 \end{aligned}$$

De plus, pour prouver la convergence de cet algorithme, nous avons imposé que la fonction approximante \hat{g}_k soit choisie ou construite telle que la propriété suivante ait lieu :

$$g(x^{k+1}) - \hat{g}_k(x^{k+1}) \leq \epsilon_k, \quad (7.3.25)$$

où $\epsilon_k > 0$ est un paramètre de tolérance qui sera défini soit a priori soit implicitement pour prendre en compte le comportement des itérées. Un inconvénient de cette façon de procéder est que le paramètre ϵ_k doit être choisi avant de calculer x^{k+1} et, par conséquent, ϵ_k n'a pas de rapport avec le comportement de l'algorithme. Ainsi il serait intéressant de mettre à jour ϵ_k en suivant justement le comportement de l'algorithme. Puisque le critère d'arrêt usuel pour notre problème est

$$\|x^{k+1} - x^k\| < \epsilon,$$

7.3 Une méthode proximale faisceau pour les fonctions DC

où ϵ est une certaine tolérance positive, une stratégie serait d'unir l'approximation de g à la distance $\|x^{k+1} - x^k\|$, et plus précisément, d'accepter une fonction approximante \hat{g}_k lorsque

$$g(x^{k+1}) - \hat{g}_k(x^{k+1}) \leq \epsilon_k \equiv \frac{\alpha}{c_k} \|x^{k+1} - x^k\|^2, \quad (7.3.26)$$

où $0 < \alpha < 1$ et $c_k > 0$.

Si la condition (7.3.26) doit être satisfaite, alors l'algorithme faisceau prend la forme suivante.

Algorithme Faisceau II pour les fonctions DC

Pas 1 : Choisir un point de départ x^0 , $c_0 \geq c > 0$ et une tolérance $\alpha \in (0, 1)$. Poser $k = 0$.

Pas 2 : Calculer $w^k \in \partial h(x^k)$ et $y^k = x^k + c_k w^k$.

Pas 3 : Choisir \hat{g}_k une approximation de g en y^k telle que

- $\hat{g}_k \leq g$
- $g(x^{k+1}) - \hat{g}_k(x^{k+1}) \leq (\alpha/c_k) \|x^{k+1} - x^k\|^2$

où

$$x^{k+1} = \arg \min_{y \in \mathbb{R}^n} \left\{ \hat{g}_k(y) + \frac{1}{2c_k} \|y - y^k\|^2 \right\}.$$

Pas 4 : Mettre $i_k := k$.

Augmenter k de 1 et retourner au Pas 2.

Nous obtenons les résultats de convergence suivants.

7.3 Une méthode proximale faisceau pour les fonctions DC

Théorème 7.3.1. *Supposons que $f := g - h$ soit bornée inférieurement et que $c_k \geq c > 0 \quad \forall k \in \mathbb{N}$. Alors la suite $\{f(x^k)\}_{k \in \mathbb{N}}$ est convergente et la suite $\{x^k\}$ est asymptotiquement régulière dans le sens suivant : $\lim_{k \rightarrow \infty} c_k^{-1} \|x^k - x^{k+1}\| = 0$. De plus, si les suites $\{x^k\}$ et $\{w^k\}$ sont bornées, alors toutes valeurs d'adhérence x^∞ et w^∞ des suites $\{x^k\}$ et $\{w^k\}$ sont des points critiques des fonctions $g - h$ et $h^* - g^*$ respectivement.*

Preuve. Par le Pas 3 de l'algorithme de point proximal à ϵ près pour les fonctions DC, nous avons, en posant $\epsilon_k = (\alpha/c_k) \|x^{k+1} - x^k\|^2$,

$$\begin{aligned} x^{k+1} &= \text{prox}_{c_k g}^{\epsilon_k}(y^k) = (I + c_k \partial_{\epsilon_k} g)^{-1}(x^k + c_k w^k) \\ \iff x^k + c_k w^k &\in (I + c_k \partial_{\epsilon_k} g)(x^{k+1}) \\ \iff c_k^{-1} x^k + w^k &\in c_k^{-1} x^{k+1} + \partial_{\epsilon_k} g(x^{k+1}) \\ \iff c_k^{-1} (x^k - x^{k+1}) + w^k &\in \partial_{\epsilon_k} g(x^{k+1}), \end{aligned}$$

ce qui implique que

$$g(x^k) \geq g(x^{k+1}) + \langle w^k + c_k^{-1}(x^k - x^{k+1}), x^k - x^{k+1} \rangle - \frac{\alpha}{c_k} \|x^{k+1} - x^k\|^2. \quad (7.3.27)$$

Par ailleurs, puisque $w^k \in \partial h(x^k)$, nous avons

$$h(x^{k+1}) \geq h(x^k) + \langle w^k, x^{k+1} - x^k \rangle. \quad (7.3.28)$$

En soustrayant (7.3.28) de (7.3.27), nous obtenons

$$\begin{aligned} f(x^k) &\geq f(x^{k+1}) + \langle c_k^{-1}(x^k - x^{k+1}), x^k - x^{k+1} \rangle - \frac{\alpha}{c_k} \|x^{k+1} - x^k\|^2 \\ \iff f(x^{k+1}) &\leq f(x^k) - c_k^{-1} \|x^k - x^{k+1}\|^2 + \frac{\alpha}{c_k} \|x^{k+1} - x^k\|^2 \\ \iff f(x^{k+1}) &\leq f(x^k) + \frac{\alpha - 1}{c_k} \|x^{k+1} - x^k\|^2 \end{aligned} \quad (7.3.29)$$

qui à son tour, puisque $\frac{\alpha-1}{c_k} < 0$ (car $0 < \alpha < 1$), implique que

$$f(x^{k+1}) < f(x^k) \quad (7.3.30)$$

7.3 Une méthode proximale faisceau pour les fonctions DC

et par conséquent nous avons

$$\begin{aligned} f(x^{k+1}) &< f(x^k) \\ &< f(x^{k-1}) \\ &< f(x^{k-2}) \\ &\vdots \end{aligned}$$

i.e.,

$$f(x^k) < f(x^0).$$

Ainsi la suite $\{f(x^k)\}$ est bornée supérieurement. D'autre part, f est bornée inférieurement par hypothèse, d'où la suite $\{f(x^k)\}$ est bornée et a donc au moins une valeur d'adhérence. En fait, toute la suite converge. En effet, $\{f(x^k)\}$ est décroissante car $f(x^{k+1}) - f(x^k) < 0$ (par (7.3.30)) et $\{f(x^k)\}$ est bornée. Elle est donc convergente.

En outre, nous déduisons de (7.3.29) que

$$\lim_{k \rightarrow \infty} c_k^{-1} \|x^k - x^{k+1}\|^2 = 0,$$

qui à son tour, puisque $c_k \geq c > 0$, implique que

$$\lim_{k \rightarrow \infty} c_k^{-1} \|x^k - x^{k+1}\| = 0.$$

A présent, considérons deux sous-suites $\{x^{k_\nu}\}$ et $\{w^{k_\nu}\}$ de $\{x^k\}$ et $\{w^k\}$ convergeant respectivement vers x^∞ et w^∞ . En passant à la limite dans les relations suivantes

$$c_{k_\nu}^{-1}(x^{k_\nu} - x^{k_\nu+1}) + w^{k_\nu} \in \partial_{\epsilon_{k_\nu}} g(x^{k_\nu+1})$$

et

$$w^{k_\nu} \in \partial h(x^{k_\nu}),$$

et en prenant en compte le fait que les applications multivoques $\partial_{(\cdot)} g(\cdot)$ et $\partial h(\cdot)$ sont fermées sur $\mathbb{R}_+ \times \mathbb{R}^n$, nous obtenons

$$\underbrace{w^{k_\nu}}_{\downarrow w^\infty} \in \partial_{\epsilon_{k_\nu}} \underbrace{g(x^{k_\nu+1})}_{\downarrow x^\infty} \quad \text{et} \quad \underbrace{w^{k_\nu}}_{\downarrow w^\infty} \in \partial h(\underbrace{x^{k_\nu}}_{\downarrow x^\infty}),$$

7.3 Une méthode proximale faisceau pour les fonctions DC

i.e.,

$$w^\infty \in \partial g(x^\infty) \quad \text{et} \quad w^\infty \in \partial h(x^\infty).$$

D'où nous déduisons que $\partial g(x^\infty) \cap \partial h(x^\infty) \neq \emptyset$, i.e., en d'autres termes, que x^∞ est un point critique de $g - h$.

D'autre part, par dualité, nous obtenons par la proposition 2.3.5 que

$$x^\infty \in \partial g^*(w^\infty) \quad \text{et} \quad x^\infty \in \partial h^*(w^\infty),$$

d'où nous déduisons que w^∞ est un point critique de $h^* - g^*$. □

Conclusion

Dans ce mémoire, nous nous sommes intéressés à une famille d'algorithmes pour résoudre des problèmes de programmation DC définis sur l'espace réel à n dimensions.

Après avoir rappelé quelques propriétés importantes d'analyse convexe dans les premiers chapitres de ce mémoire, il était important, dans un premier temps, de bien définir le cadre de la programmation DC. Une bonne compréhension de ce que sont précisément les fonctions DC, les problèmes de programmation DC et l'algorithme DC était nécessaire pour contribuer à l'élaboration de l'objectif principal de ce mémoire.

Il s'agissait en fait de décrire une méthode faisceau permettant la minimisation d'une fonction DC, forme sous laquelle beaucoup de problèmes du monde réel peuvent s'écrire. Nous sommes arrivés à un algorithme proximal faisceau pour les fonctions DC en deux temps : tout d'abord en augmentant la fonction h le long de la direction d'un sous-gradient et ensuite en diminuant la fonction g grâce à un pas proximal. Malheureusement, nous ne sommes pas parvenus à rendre cet algorithme totalement implémentable car nous n'avons pas réussi à spécifier un critère d'arrêt cohérent avec ce type de fonctions. Il faudrait pour cela que la solution de chacun des sous-problèmes que nous considérons puisse être un ϵ -point critique de f ; cette problématique reste donc une question ouverte !

La démarche de ce travail était avant tout théorique. Cependant, une méthode numérique ne peut présenter d'intérêts que si elle est efficace en pratique. Il serait donc intéressant de tester sur divers exemples de fonctions DC l'algorithme faisceau pour la programmation DC. Ceci pourrait faire l'objet d'un travail futur.

Bibliographie

- [1] J. B. Hiriart-Urruty, *From convex optimization to nonconvex optimization : necessary and sufficient conditions for global optimization*, Nonsmooth Optimization and Related Topics, Plenum Press, 1989, pp. 219-240.
- [2] J. B. Hiriart-Urruty, *Generalized differentiability, duality and optimization for problems dealing with difference of convex functions*, Convexity and Duality in Optimization, Lecture Notes in Economics and Mathematical Systems 256, 1986, pp. 37-70.
- [3] J. B. Hiriart-Urruty & C. Lemaréchal, *Convex analysis and minimization algorithms*, Springer-Verlag, Berlin, 1993.
- [4] J. B. Hiriart-Urruty & H. Tuy, *Essays on nonconvex optimization*, Mathematical Programming, Vol. 41, 1988, pp. 125-126.
- [5] R. Horst & N.V. Thoai, *DC Programming : Overview*, Journal of Optimization Theory and Applications, Vol. 103, No.1, 1999, pp. 1-43.
- [6] T. T. Hue, J.-J. Strodiot & V. H. Nguyen, *Convergence of the approximate auxiliary problem method for solving generalized variational inequalities*, Journal of Optimization Theory and Applications, Vol. 121, No.1, 2004, pp. 119-145.
- [7] K. C. Kiwiel, *Proximity control in bundle methods for convex non-differentiable minimization*, Mathematical Programming, Vol.

BIBLIOGRAPHIE

- 46, 1990, pp. 105-122.
- [8] B. Martinet, *Algorithmes pour la résolution des problèmes d'optimisation et de minmax*, Thèse d'état Université de Grenoble, France, 1972.
- [9] A. Moudafi & P. Maingé, *On the convergence of an approximate proximal method for DC functions*, Journal of Computational Mathematics, Vol. 24, No.2, 2006.
- [10] D. T. Pham & E. B. Souad, *Algorithms for solving a class of nonconvex optimization problems : Methods of subgradient*, Fermat days 85, Mathematics for Optimization, Elsevier, North Holland, 1986, pp. 249-270.
- [11] R. T. Rockafellar, *Convex analysis*, Princeton University Press, NJ, 1970.
- [12] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM Journal on Control and Optimization, Vol. 14, No.5, 1976, pp. 877-898.
- [13] R. T. Rockafellar & R. J.-B. Wets, *Variational analysis*, Springer-Verlag, New York, 1997.
- [14] J.-J. Strodiot, *An introduction to nonsmooth optimization*, Cours donné à l'Université Nationale du Vietnam à Ho Chi Minh-Ville en mars 2005.
- [15] Wen-yu Sun, Raimundo J. B. Sampaio & M. A. B. Candido, *Proximal point algorithm for minimization of DC function*, Journal of Computational Mathematics, Vol. 21, No.4, 2003, pp. 451-462.
- [16] Le Thi Hoai An & Pham Dinh Tao, *The DC (Difference of Convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems*, Annals of Operations

BIBLIOGRAPHIE

Research, Vol. 133, 2005, pp. 23-46.

[17] J. F. Toland, *Duality in nonconvex optimization*, Journal of Mathematical Analysis and Applications, Vol. 66, 1978, pp. 399-415.

[18] W. I. Zangwill, *Nonlinear programming : A unified approach*, Prentice-Hall, 1969.