

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

**UNIVERSITÉ MOHAMED KHIDER, BISKRA**

FACULTÉ des SCIENCES EXACTES et des SCIENCES de la NATURE et de la VIE

**DÉPARTEMENT DE MATHÉMATIQUES**



Mémoire présenté en vue de l'obtention du Diplôme :

**MASTER en Mathématiques**

Option : **Analyse**

Par

**BOUCHERITTE Ahlem**

Titre :

**Algorithme d'optimisation Sous Contraintes**

Membres du Comité d'Examen :

Dr. MENACER Tidjani	UMKB	Président
Dr. BELLAGOUN Abed Ghani	UMKB	Encadreur
Dr. ADOUANE Saida	UMKB	Examineur

Juin 2019

## DÉDICACE

Avec tout respect et amour, je dédie ce modeste travail après "**Dieu**" aux deux personnes qui ont sacrifié leur vie pour ma réussite ;

★ A la lumière de ma vie "**ma mère**".

★ A la source de joie et de bonheur "**mon père**".

Je dédie aussi à mes chères sœurs "**Amira, Malak, Meriem**" et à mon seul frère, la joie de nos vies "**Abed el Azize**" pour leurs conseils, aides et encouragements.

A toute ma famille, je souhaite la bonne santé, la longue vie et la bonheur.

A la famille "cherih" sur tout mon fiancé "**Mounir**".

A toutes mes amies du primaire jusqu'à l'université.

Sans oublier le grand remerciement à mes amies de travail pour leurs encouragements et leurs aides.

A tous le monde je sais.

**Ahlem**

## REMERCIEMENTS

Je tiens tout d'abord à remercier "*allah*" le tout puissant et miséricordieux, qui m'a donné la force et la patience d'accomplir ce modeste travail.

Mes chaleureux remerciements à ceux qui m'a énormément aidé et de me faire profiter de ses conseils, En Particulier à mon encadreur Monsieur le *Dr.BELLAGOUN.A*

Mes vifs remerciements vont également aux membres de jury pour l'intérêt qu'ils ont accepté d'examiner mon travail et de l'enrichir par leurs propositions.

Je tiens également a remercié toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

Enfin ; à tous mes enseignants de primaire jusqu'à l'université a tout pour tout.

**Merci à tout**

# Table des matières

<b>Remerciements</b>	<b>ii</b>
<b>Table des matières</b>	<b>iii</b>
<b>Table des figures</b>	<b>vi</b>
<b>Introduction</b>	<b>1</b>
<b>1 Notion d'Optimisation Sous Contraintes</b>	<b>2</b>
1.1 Rappel de calcul différentiel . . . . .	3
1.1.1 Définition de la différentiabilité . . . . .	3
1.1.2 Le gradient d'une fonction . . . . .	3
1.1.3 La Hessienne d'une fonction . . . . .	3
1.1.4 Matrice Jacobienne . . . . .	4
1.2 Forme quadratique . . . . .	4
1.3 Notion de convexité . . . . .	5
1.3.1 Ensemble convexe . . . . .	5
1.3.2 Fonctions concaves, Fonction convexes . . . . .	5
1.4 Extremas locaux et globaux sous contraintes . . . . .	7
1.4.1 Optimum local . . . . .	8
1.4.2 Optimum global . . . . .	8
1.5 Conditions d'optimalités . . . . .	9
1.5.1 Les condition nécessaires et suffisantes d'existence d'un optimum . . . . .	9

1.6	Les conditions de lagrange . . . . .	9
1.6.1	Théorème de Lagrange . . . . .	10
1.6.2	Définition du Lagrangien . . . . .	11
1.7	Conditions de Karush Kuhn et Tucker . . . . .	16
1.7.1	Problèm avec contraintes d'inégalités . . . . .	16
1.7.2	Conditions de qualification des contraintes . . . . .	17
1.7.3	Le lagrangien Modifié . . . . .	20
<b>2</b>	<b>Les Algorithmes D'Optimisation Sous Contraintes</b>	<b>21</b>
2.1	Méthodes Admissibles . . . . .	21
2.1.1	La méthode du Gradient Projeté . . . . .	22
2.1.2	Méthode de Newton projeté . . . . .	24
2.1.3	Méthode de Directions Réalisables . . . . .	27
2.1.4	Méthode de Pénalisation . . . . .	29
2.1.5	Méthode d'Uzawa . . . . .	33
2.1.6	Méthode de Gradients Conjugués . . . . .	34
2.2	Convergence des Méthodes d'Optimisation . . . . .	39
<b>3</b>	<b>Problèmes Variationnels</b>	<b>41</b>
3.1	Exemples de fonctionnels . . . . .	42
3.2	Constructions mathématiques . . . . .	43
3.3	Fonction $y : \mathbb{R} \rightarrow \mathbb{R}$ . . . . .	43
3.3.1	Cas général . . . . .	43
3.3.2	L'astuce de Beltrami . . . . .	44
3.3.3	Minimisation avec contraintes d'égalité . . . . .	44
3.4	Fonction $q : \mathbb{R} \rightarrow \mathbb{R}^n$ . . . . .	45
3.5	Fonction $u : \mathbb{R}^n \rightarrow \mathbb{R}$ . . . . .	45
3.6	Dérivées d'ordre supérieur . . . . .	46

3.7	Condition d'Euler-Lagrange pour les exemples antérieurs . . . . .	47
3.7.1	La longueur d'une courbe paramétrée . . . . .	47
3.7.2	La longueur d'une courbe paramétrée . . . . .	47
3.7.3	L'action d'une trajectoire . . . . .	47
3.7.4	L'action dans un moyen non-homgène . . . . .	47
3.7.5	L'énergie potentielle d'une chaîne suspendue . . . . .	48
3.7.6	La tension d'une membrane par petites déformations verticales . . .	48
3.7.7	La tension d'une plaque par petites déformations verticales . . . . .	49
3.7.8	L'aire d'une surface définie par graphe . . . . .	49
3.7.9	La variation totale d'une fonction . . . . .	49
3.7.10	Le $p$ laplacien . . . . .	49
3.7.11	La pente maximale d'une fonction . . . . .	50
3.7.12	Le débruitage par lissage d'une image $I(x)$ . . . . .	50
	<b>Conclusion</b>	<b>51</b>
	<b>Bibliographie</b>	<b>52</b>

# Table des figures

1.1	ensemble convexe et ensemble non convexe . . . . .	5
1.2	Fonction Concave . . . . .	7
1.3	Fonction convexe . . . . .	7
1.4	Extremum local et extremum global . . . . .	8
1.5	Multiplicateurs de Lagrange . . . . .	15
1.6	Multiplicateurs de Lagrange . . . . .	16
1.7	Condition du Kuhn et Tucker . . . . .	19
2.1	Méthode de Pénalisation Externe . . . . .	32
2.2	Méthode de Pénalisation Externe . . . . .	33

# Introduction

L'objectif de ce mémoire est de fournir un aperçu général sur l'optimisation sous contraintes et les principales méthodes utilisées pour les résoudre. Ce n'est pas un document approfondi proprement dit mais une illustrations sur les bases des algorithmes qui seront par la suite un pivot pour la recherche scientifique et numérique des optimums.

L'optimisation sous contraintes se trouve pratiquement dans toutes les branches de la science moderne, une approximation des solutions par des algorithmes simples et convergents est le souhait de tout chercheur pour mieux économiser le temps et l'argent, la qualité de l'optimisation est un but dans toute recherche scientifique qui s'intéresse à l'optimisation.

Après une introduction générale et les principaux résultats d'existence et d'unicité, le deuxième chapitre est basé essentiellement sur les méthodes et algorithmes d'optimisation, nous nous bornerons à citer les plus célèbres dans la littérature et quelques résultats de convergence.

3<sup>ème</sup> chapitre est une extension aux problèmes variationnels et leurs relations avec l'optimisation sous contraintes pour plus de détail nous renvoyons les intéressés à la bibliographie, car ce chapitre est beaucoup plus explicatif par rapport à un cours approfondi.



# Chapitre 1

## Notion d'Optimisation Sous Contraintes

### Introduction

On s'intéresse à des problèmes d'optimisation de la forme :

$$\left\{ \begin{array}{l} \min_{x \in \mathbb{R}^n} f(x) \\ \text{sous les contraintes} \\ g(x) \leq 0 \\ h(x) = 0 \end{array} \right. \quad (1.1)$$

où les fonctions  $f$ ,  $g$  et  $h$  sont différentiables au moins une fois, et  $f$  est non-linéaire.

Cependant nous étudierons le cas où  $g$  et  $h$  sont linéaires avec un intérêt tout particulier.

Ce chapitre est consacré aux définitions et aux notions principales de l'optimisation avec contraintes, on décrit dans la notion de la différentiabilité d'une fonction, le gradient et la hessienne, puis nous allons montrer comment les conditions d'optimalité (condition de lagrange et condition de Kuhn et Tucker) associées au problème (1.1)

## 1.1 Rappel de calcul différentiel

### 1.1.1 Définition de la différentiabilité

Soit  $V$  un ouvert de  $\mathbb{R}^n$ , et  $f : V \rightarrow \mathbb{R}$  une fonction. On dit que la fonction  $f$  définie sur un voisinage de  $u \in V$  à valeurs dans  $\mathbb{R}$ , est dérivable (ou différentiable) au sens de *Fréchet* en  $u$  s'il existe une forme linéaire continue sur  $V$ ,  $L \in V'$  telle que :

$$f(u + \omega) = f(u) + L(\omega) + \theta(\omega), \quad \text{avec} \quad \lim_{\omega \rightarrow 0} \frac{\theta(\omega)}{\|\omega\|} = 0.$$

On appelle  $L$  la dérivée (ou la différentielle, ou le gradient) de  $f$  en  $u$  et on note  $L = f'(u)$ .

### 1.1.2 Le gradient d'une fonction

Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$  différentiable en un point  $a$ .

**Définition 1.1.1** On appelle gradient de  $f$  en  $(x_1, x_2, \dots, x_n)$  la matrice ligne  $Df(x_1, x_2, \dots, x_n)$  telle que :

$$Df(x_1, x_2, \dots, x_n) = \left( \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right) (x_1, x_2, \dots, x_n).$$

### 1.1.3 La Hessienne d'une fonction

On appelle matrice hessienne  $H(x_1, x_2, \dots, x_n)$  de  $f$  la matrice des dérivées secondes de  $f$  évaluées au point  $(x_1, x_2, \dots, x_n)$

$$H(x_1, x_2, \dots, x_n) = \begin{pmatrix} f''_{11} & f''_{12} & \dots & f''_{1n} \\ f''_{21} & f''_{22} & \dots & f''_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ f''_{n1} & f''_{n2} & \dots & f''_{nn} \end{pmatrix}$$

Comme  $f''_{ij} = f''_{ji} \forall (i, j)$ , la matrice hessienne de  $f$  est une matrice symétrique d'ordre  $n$ .

### 1.1.4 Matrice Jacobienne

Soit  $G = (g_1, g_2, \dots, g_m)$  une fonction définie de  $\mathbb{R}^n$  dans  $\mathbb{R}^m$ . A tout vecteur  $x = (x_1, x_2, \dots, x_n)$ , la fonction  $G$  associée le vecteur de fonction  $(g_1(x), g_2(x), \dots, g_m(x))$ .

On appelle matrice jacobienne de  $G$  la matrice de dimension  $(m, n)$ ;  $J_G(x_1, x_2, \dots, x_n)$  des dérivées partielles des  $m$  fonctions qui composent  $G$  :

$$J_G(x, \dots, x) = \begin{pmatrix} \frac{\partial g_1}{\partial x_1}(x) & \frac{\partial g_1}{\partial x_2}(x) & \dots & \frac{\partial g_1}{\partial x_n}(x) \\ \frac{\partial g_2}{\partial x_1}(x) & \frac{\partial g_2}{\partial x_2}(x) & \dots & \frac{\partial g_2}{\partial x_n}(x) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_m}{\partial x_1}(x) & \frac{\partial g_m}{\partial x_2}(x) & \dots & \frac{\partial g_m}{\partial x_n}(x) \end{pmatrix}$$

## 1.2 Forme quadratique

**Définition 1.2.1** Soit  $A_{(n \times n)}$  une matrice symétrique et  $b \in \mathbb{R}^n$ . On appelle forme quadratique la fonction.

$f : \mathbb{R}^n \rightarrow \mathbb{R}$  définie par :

$$f(x) = \frac{1}{2}x^t Ax - b^t x.$$

**Propriété 1.2.1** Soit  $A$  une matrice carrée symétrique. Soit  $x$  un vecteur colonne quelconque. On note  $x^t$  sa transposée.

1. /Si  $x^t Ax \geq 0$  ,  $\forall x \in \mathbb{R}^n$  :  $A$  est semi-définie positive.
2. /Si  $x^t Ax > 0$  ,  $\forall x \in \mathbb{R}^n$  :  $A$  est définie positive.
3. /Si  $x^t Ax \leq 0$  ,  $\forall x \in \mathbb{R}^n$  :  $A$  est semi-définie négative.
4. /Si  $x^t Ax < 0$  ,  $\forall x \in \mathbb{R}^n$  :  $A$  est définie négative.

**Remarque 1.2.1** Soit  $A$  une matrice carré symétrique d'ordre  $n$ .

- \*  $A$  définie positive  $\iff$  ses  $n$  mineures principaux diagonaux  $D_k$  sont  $> 0$ .
- \*  $A$  semi-définie positive  $\iff$  tous ses mineures principaux  $D_k$  sont  $\geq 0$ .
- \*  $A$  définie négative  $\iff$  ses  $n$  mineures principaux diagonaux  $D_k$  sont alternativement  $< 0$  ( $k$  impair) et  $> 0$  ( $k$  pair).

\*  $A$  semi définie négative  $\iff$  tous ses mineures principaux  $D_k$  sont alternativement  $\leq 0$  ( $k$  impair) et  $\geq 0$  ( $k$  pair).

## 1.3 Notion de convexité

La convexité joue un rôle extrêmement important en optimisation.

### 1.3.1 Ensemble convexe

**Définition 1.3.1** – Un ensemble  $S$  de  $R^n$  est convexe ssi :

$$\forall (x, y) \in S^2 : (1 - \lambda)x + \lambda y \in S, \forall \lambda \in [0, 1].$$

– Un ensemble  $S$  de  $R^n$  est strictement convexe ssi :

$$\forall (x, y) \in S^2 : (1 - \lambda)x + \lambda y \in \text{intérieur } S, \forall \lambda \in ]0, 1[.$$

**Remarque 1.3.1** La notion d'ensemble concave n'existe pas.

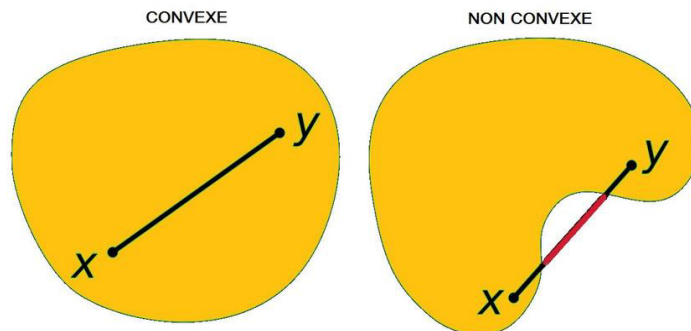


FIG. 1.1 – ensemble convexe et ensemble non convexe

### 1.3.2 Fonctions concaves, Fonction convexes

Soit  $f$  une fonction de plusieurs variables définie sur un ensemble convexe  $S$ .

**Fonction concave :**

- $f$  est concave sur  $S$  ssi,  $\forall (x; y) \in S^2$  et  $\forall \lambda \in [0, 1]$ , on a :

$$f((1 - \lambda)x + \lambda y) \geq (1 - \lambda)f(x) + \lambda f(y).$$

- $f$  est strictement concave sur  $S$  ssi :

$$f((1 - \lambda)x + \lambda y) > (1 - \lambda)f(x) + \lambda f(y).$$

**Fonction convexe :**

- $f$  est convexe sur  $S$  ssi,  $\forall (x, y) \in S^2$  et  $\lambda \in [0, 1]$ , on a :

$$f((1 - \lambda)x + \lambda y) \leq (1 - \lambda)f(x) + \lambda f(y).$$

- $f$  est strictement convexe sur  $S$  ssi :

$$f((1 - \lambda)x + \lambda y) < (1 - \lambda)f(x) + \lambda f(y).$$

**Propriété importantes :**

- \*  $f$  concave  $\Leftrightarrow -f$  convexe.
- \* Si  $f$  et  $g$  sont des fonctions concaves (respectivement convexes), alors  $\forall (a, b) \in \mathbb{R}_+^2$   $(a * f + b * g)$  est une fonction concave (respectivement convexe).
- \* Si  $f$  est une fonction concave et  $g$  est une fonction croissante et concave alors la fonction  $g(f(x))$  est concave.
- \* Si  $f$  est une fonction convexe et  $g$  est une fonction croissante et convexe alors la fonction  $g(f(x))$  est convexe.
- \* Une fonction affine est à la fois concave et convexe.

**Exemple 1.3.1** ■ *Fonction strictement concave*  $f(x, y) = 6 - (x - 1)^2 - (y - 1)^2$ .

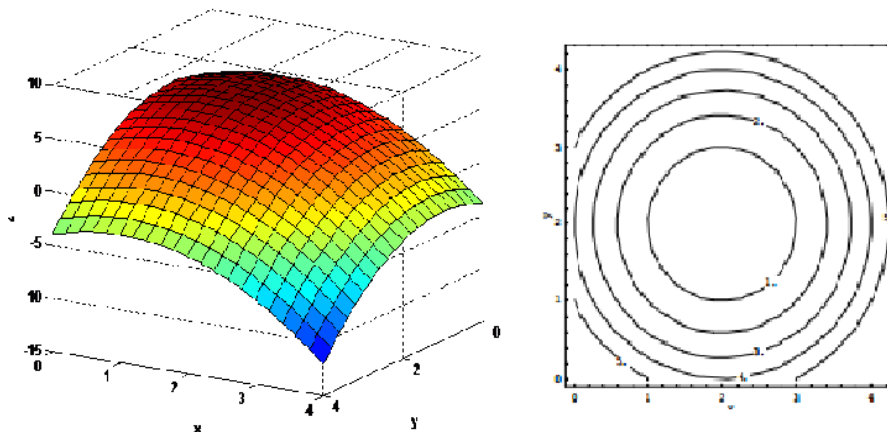


FIG. 1.2 – Fonction Concave

**Exemple 1.3.2** ■ *Fonction strictement convexe*  $f(x, y) = (x - 1)^2 + (y - 1)^2$ .

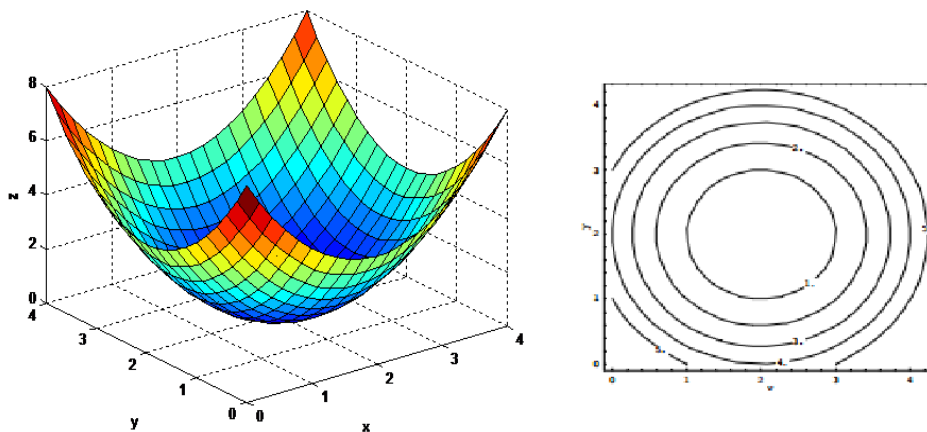


FIG. 1.3 – Fonction convexe

## 1.4 Extremas locaux et globaux sous contraintes

Un problème d'optimisation s'écrit en général sous la forme :

$$\begin{cases} \min_{x_1, x_2, \dots, x_n} f(x_1, x_2, \dots, x_n), \\ \text{s.c} \quad \text{contraintes}(j) \quad \forall j \in 1, \dots, m. \end{cases} \quad (1.2)$$

### Définition 1.4.1

#### 1.4.1 Optimum local

- La variable  $x^*$  est un maximum local d'une fonction  $f$  définie sur l'ensemble convexe  $S \Leftrightarrow \exists \varepsilon > 0$  tel que  $f(x) \leq f(x^*)$ ,  $\forall x \in S$  et  $|x - x^*| \leq \varepsilon$ .
- La variable  $x^*$  est un minimum local d'une fonction  $f$  définie sur l'ensemble convexe  $S \Leftrightarrow \exists \varepsilon > 0$  tel que  $f(x) \geq f(x^*)$ ,  $\forall x \in S$  et  $|x - x^*| \leq \varepsilon$ .

#### 1.4.2 Optimum global

- La variable  $x^*$  est un maximum global d'une fonction  $f$  définie sur l'ensemble convexe  $S \Leftrightarrow f(x) \leq f(x^*)$ ,  $\forall x \in S$ .
- La variable  $x^*$  est un minimum global d'une fonction  $f$  définie sur l'ensemble convexe  $S \Leftrightarrow f(x) \geq f(x^*)$ ,  $\forall x \in S$ .

**Exemple 1.4.1** ★ *Fonction  $f$  admettant un maximum local en  $x^*$  et un maximum global en  $x^{**}$  sur l'ensemble de son domaine de définition.*

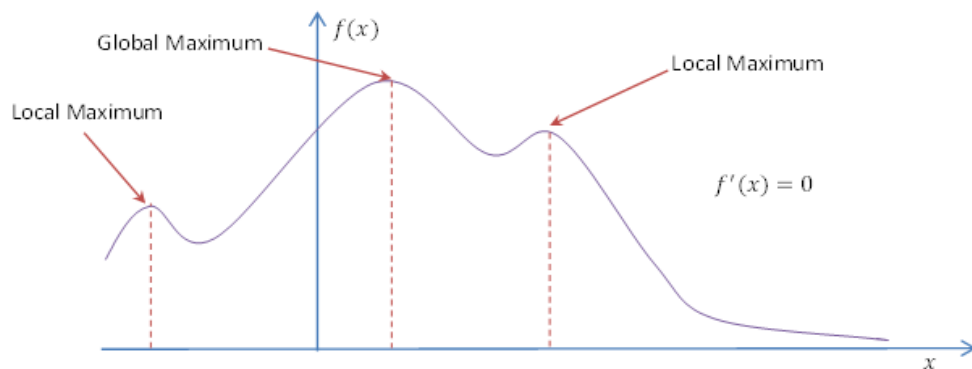


FIG. 1.4 – Extremum local et extremum global

## 1.5 Conditions d'optimalités

### 1.5.1 Les conditions nécessaires et suffisantes d'existence d'un optimum

Pour trouver la solution  $x^*$  d'un problème d'optimisation quelconque, on distingue traditionnellement deux types de conditions :

**Les conditions du première ordre (C P O) :** Sont les conditions nécessaires que doit vérifier un optimum, s'il existe. Ces conditions s'écrivent comme un système d'égalités ou d'inégalités dont la résolution permet de trouver  $x^*$ .

**Les conditions du second ordre (C S O) :** Garantissent que les conditions du premier ordre sont suffisantes pour que  $x^*$  soit bien la solution du problème d'optimisation.

## 1.6 Les conditions de lagrange

On considère le problème d'optimisation suivante :

$$\left\{ \begin{array}{l} \min f(x), \\ \text{sous la contraintes} \quad h(x) = 0. \end{array} \right. \quad (1.3)$$

telle que  $x \in \mathbb{R}^n$ ,  $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ .

Nous entendons par optimiser localement c'est chercher le point  $x^*$  de  $\mathbb{R}^n$  qui minimise ou maximise localement  $f$  tout en satisfaisant la contrainte  $h(x^*) = 0$ .

**Les Contraintes Régulières :**

**Définition 1.6.1** on dit que la contrainte  $h$  est régulière en  $x^* \in S$  si les vecteurs  $Dh_1(x^*), \dots, Dh_n(x^*)$  forment une partie libre.



### 1.6.1 Théorème de Lagrange

**Théorème 1.6.1** Soit  $x^* \in S = \{x \in \mathbb{R}^n, h(x) = 0\}$  un point régulier solution du problème vérifiant donc :

$$f(x^*) \leq f(x) \quad , \quad \forall x \in S$$

Alors il existe nécessairement un vecteur  $\lambda \in \mathbb{R}^m$  unique vérifiant :

$$\nabla f(x^*) + \nabla h(x^*) \lambda = 0.$$

Soit encore :

$$\nabla f(x^*) + \sum_{i=1}^m \lambda_i \nabla h_i(x^*) = 0.$$

Les composantes du vecteur sont appelées **multiplicateurs de Lagrange**

**Preuve.** Considérons une courbe  $x(t)$  définie pour  $t \in [-\alpha, \alpha]$  vérifiant :

$$\begin{cases} x(t) \in S, \forall t \in [-\alpha, \alpha] & , \quad \alpha > 0, \\ x(0) = x^*. \end{cases}$$

On a :

$$f(x(0)) \leq f(x(t)) \quad , \quad \forall t \in [-\alpha, \alpha],$$

donc nécessairement :

$$\frac{d}{dt} f(x(t)) \Big|_{t=0} = \nabla f(x^*)^T \dot{x}(0) = 0.$$

Ce qui signifie que  $\nabla f(x^*)$  se trouve dans l'orthogonal de  $T(x^*)$  et le plan tangent à  $S$  en  $x^*$ . Si l'on utilise l'équivalence :

$$T(x^*) = \text{Ker } \nabla h(x^*)^T \Leftrightarrow T(x^*)^\perp = \text{Im } \nabla h(x^*),$$

il existe donc un vecteur  $\lambda \in \mathbb{R}^m$  tel que :

$$\nabla f(x^*) = -\nabla h(x^*) \lambda.$$

L'unicité résulte du fait que  $\nabla h(x^*)$  est de rang  $m$ . ■

## 1.6.2 Définition du Lagrangien

**Théorème 1.6.2** Soit  $f : \mathbb{R}^n \rightarrow \mathbb{R}$

Soit  $x^*$  un minimiseur ou maximiseur local de  $f$  sous la contrainte  $h(x) = 0$ .

On suppose que la contrainte  $h$  est régulière en  $x^*$ , alors il existe des réels  $\lambda_1^*, \lambda_2^*, \dots, \lambda_m^*$  telle que :

$$Df(x^*) + \lambda_1^* Dh_1(x^*) + \dots + \lambda_m^* Dh_m(x^*) = 0.$$

★ Le point  $x^*$  pour lequel il existe des coefficients  $\lambda_1^*, \lambda_2^*, \dots, \lambda_m^*$  satisfaisantes l'identité sont appelés des points stationnaires du problème d'optimisation.

★ Les coefficients  $\lambda_1^*, \lambda_2^*, \dots, \lambda_m^*$  satisfaisantes l'identité sont appelés **Les Multiplicateurs de Lagrange** en  $x^*$ .

On note  $\lambda^* = (\lambda_1^*, \lambda_2^*, \dots, \lambda_m^*)$  Les Multiplicateurs de Lagrange.

★ On introduit la fonction suivante :

$$L(x, \lambda) = f(x) + \lambda_1 h_1(x) + \dots + \lambda_m h_m(x) \quad , \quad (x, \lambda) \in \mathbb{R}^n \times \mathbb{R}^m$$

On appelle Lagrangien du problème d'optimisation, l'expression ci-dessus.

### Proposition 1.6.1

◆ le points  $x^*$  satisfait  $h(x^*) = 0$  si et seulement si :

$$DL(x^*, \lambda^*) = 0 \in \mathbb{R}^n.$$

◆ Si  $x^*$  est une solution du problème d'optimisation local, alors il existe  $\lambda^* \in \mathbb{R}^m$  tel que :

$$DL(x^*, \lambda^*) = 0.$$

**Preuve.** En effet

$$\frac{\partial L}{\partial x_j}(x^*, \lambda^*) = \frac{\partial f}{\partial x_j}(x^*) + \lambda_1^* \frac{\partial h_1}{\partial x_j}(x^*) + \dots + \lambda_m^* \frac{\partial h_m}{\partial x_j}(x^*).$$

Pour tout  $1 \leq j \leq n$  et  $\frac{\partial L}{\partial \lambda_i}(x^*, \lambda^*) = h_i(x^*)$  pour tout  $1 \leq i \leq m$  donc:

$$\left\{ \begin{array}{l} Df(x^*) + \lambda_1^* Dh(x^*) + \dots + \lambda_m^* Dh(x^*) = 0. \\ \qquad \qquad \qquad et \\ \qquad \qquad \qquad h(x^*) = 0. \end{array} \right.$$

On a donc transformer le problème d'optimisation sous contraintes dans  $\mathbb{R}^n$  en un problème d'optimisation sans contraintes.

$$\text{Optimiser localement } L(x, \lambda), \quad (x, \lambda) \in \mathbb{R}^{n+m}.$$

■

### Condition de Qualification des Contraintes

Pour pouvoir utiliser le Lagrangien dans la résolution d'un problème d'optimisation sous plusieurs contraintes en équations, il suffit que l'une des conditions suivantes soit vérifiée :

- La matrice Jacobienne des fonctions contraintes  $h_i$ ;  $i = 1, \dots, m$  notée  $\mathbf{G}_H$  taille  $(m, n)$  est de rang  $m$  lorsque elle est évaluée à l'optimum  $x^*$ .
- Les fonction contrainte  $h_i$  sont toutes linéaires.

### Condition du premier ordre :

**Théorème 1.6.3** *On suppose que la contrainte de qualification est vérifiée. Si le vecteur  $x^* = (x_1^*, x_2^*, \dots, x_n^*)$  est une solution du problème de maximisation (1.3), alors il existe un unique vecteur  $\lambda^*$  tel que  $x^*$  vérifie les  $n + m$  conditions suivantes :*

$$\left\{ \begin{array}{l} \frac{\partial L(x^*, \lambda^*)}{\partial x_j} = 0 \Leftrightarrow \frac{\partial f(x^*)}{\partial x_j} - \sum_{i=1}^m \lambda_i^* \frac{\partial h_i(x^*)}{\partial x_j} = 0 \quad \forall j = 1, \dots, n \\ \frac{\partial L(x^*, \lambda^*)}{\partial \lambda_j} = 0 \Leftrightarrow h_j(x^*) = 0 \quad \forall j = 1, \dots, m \end{array} \right.$$

### Condition nécessaire du second ordre :

**Théorème 1.6.4** Soit  $x^*$  un point régulier solution de (1.3). Alors il existe  $\lambda^*$  tel que :

$$\nabla_x L(x^*, \lambda^*) = 0,$$

et de plus pour tout  $y \in T(x^*)$ ,  $y \neq 0$ , on a :

$$y^\top \nabla_{xx}^2 L(x^*, \lambda^*) y \geq 0.$$

**Preuve.** Soit  $y \in T(x^*)$ . On sait qu'il existe une courbe  $x(t)$  définie pour  $t \in [-\alpha, \alpha]$  vérifiant :

$$\begin{cases} x(t) \in S & , \quad \forall t \in [-\alpha, \alpha] & , \quad \alpha > 0, \\ & x(0) = x^*. \\ & x'(0) = y. \end{cases}$$

Puisque  $x^*$  est optimal on a :

$$f(x(0)) \leq f(x(t)) \quad , \quad \forall t$$

et puisque la fonction  $f$  est deux fois différentiable, on a nécessairement :

$$\frac{d^2}{dt^2} f(x(t)) \Big|_{t=0} \geq 0.$$

On a ici d'une part :

$$\frac{d^2}{dt^2} f(x(t)) = \nabla f(x(t))^\top \ddot{x}(t),$$

et donc :

$$\frac{d^2}{dt^2} f(x(t)) = \dot{x}(t)^\top \nabla^2 f(x(t)) \dot{x}(t) + \nabla f(x(t))^\top \ddot{x}(t).$$

$$\frac{d^2}{dt^2} f(x(t)) \Big|_{t=0} = y^\top \nabla^2 f(x^*) y + \nabla f(x^*)^\top \ddot{x}(0) \geq 0. \quad (1.4)$$

D'autre part on a  $h_j(x(t)) = 0$  donc :

$$\frac{d^2}{dt^2} h(x(t)) \Big|_{t=0} = y^\top \nabla^2 h_i(x^*) y + \nabla h_i(x^*)^\top \ddot{x}(0) = 0 \quad , \quad i = 0, \dots, m.$$

On peut multiplier chacune de ces égalités par  $\lambda_i^*$  et en faire la somme, ce qui donne :

$$y^\top \left( \sum_{i=1}^m \lambda_i^* \nabla^2 h_i(x^*) y \right) + \left( \sum_{i=1}^m \lambda_i^* \nabla h_i(x^*)^\top \right) \ddot{x}(0) = 0.$$

En additionnant cette dernière égalité à (1.4) on obtient

$$y^\top \left( \nabla^2 f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla^2 h_i(x^*) \right) y + \left( \nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla h_i(x^*) \right)^\top \ddot{x}(0) \geq 0,$$

et puisque le deuxième terme est nul (**condition de lagrange**) on obtient bien l'inégalité annoncée. Le résultat suivant est une généralisation du théorème précédent. ■

**Théorème 1.6.5** Soit  $x^* \in R^n$  et  $\lambda^* \in R^m$  vérifiant les condition :

$$\left\{ \begin{array}{l} h(x^*) = 0. \\ \nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla h_i(x^*) = 0. \\ y^\top \nabla_{xx}^2 L(x^*, \lambda^*) y \geq 0 \quad , \quad \forall y \in T(x^*) \quad , \quad y \neq 0. \end{array} \right.$$

alors  $x^*$  est un minimum local du problème (1.3).

**Condition suffisant du second order :**

**Théorème 1.6.6** Soit  $x^* \in R^n$  et  $\lambda^* \in R^m$  vérifiant les condition :

$$\left\{ \begin{array}{l} h(x^*) \leq 0. \\ \nabla f(x^*) + \sum_{i=1}^m \lambda_i^* \nabla h_i(x^*) = 0. \\ \lambda_i^* \geq 0 \quad , \quad i = 1, \dots, m \\ \lambda_i^* h_i = 0 \quad , \quad i = 1, \dots, m \\ y^\top \nabla_{xx}^2 L(x^*, \lambda^*) y \geq 0 \quad , \quad \forall y \in T^+(x^*) \quad , \quad y \neq 0. \end{array} \right.$$

où on a noté  $T^+(x^*)$  le plan tangent en  $x^*$  à la surface

$$S^+ = \{ x \in R^n \quad , \quad h_i(x^*) = 0 \quad , \quad i \in I(x^*) \text{ et } \lambda_i > 0. \}$$

Alors  $x^*$  est un minimum local du problème (1.3).

**Exemple 1.6.1** (1) : Soit  $f(x, y) = x^2 + 2y^2 - 2x$ .

On remarque que  $f(x, y) = (x - 1)^2 + 2y^2 - 1$  donc  $\min f(x, y) = (1, 0)$ .

considérons  $f$  la contrainte  $g(x, y) = y - x^2 = 0$ .

Minimiser  $f$  revient à minimiser  $f(x) = (x - 1)^2 - 2x^4 - 1$ .

$f'(x) = 2(x - 1) + 8x^3 = (2x - 1)(2x^2 + x + 1)$ .

donc  $f'(x) = 0 \Leftrightarrow x = \frac{1}{2}$ .

On remarque que la solution du problème est la projection de  $(1, 0)$  sur  $C$ , la courbe représentative de la contrainte.

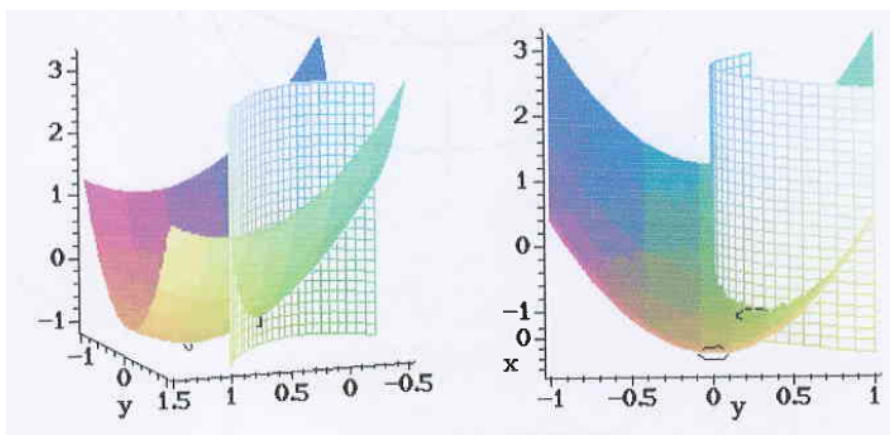


FIG. 1.5 – Multiplicateurs de Lagrange

**Exemple 1.6.2** (2) :  $\min f(x) = x_1^2 + x_2^2$ .

avec  $h(x) = x_1^2 + 2x_2^2 - 1 = 0$ .

$$\nabla f(x) = \begin{bmatrix} 2x_1 \\ 2x_2 \end{bmatrix}, \quad \nabla h(x) = \begin{bmatrix} 2x_1 \\ 4x_2 \end{bmatrix}.$$

$$\begin{cases} \nabla f(x^*) + \nabla h^\top(x^*) \lambda^* = 0 \\ h(x^*) = 0 \end{cases} \Leftrightarrow \begin{cases} x_1^* + \lambda^* x_1^* = 0 \Leftrightarrow x_1^* = 0 \quad \text{ou} \quad \lambda^* = -1 \\ x_2^* + 2\lambda^* x_2^* = 0 \Leftrightarrow x_2^* = 0 \quad \text{ou} \quad \lambda^* = -\frac{1}{2} \\ (x_1^*)^2 + 2(x_2^*)^2 - 1 = 0 \end{cases}$$

$$\Leftrightarrow \begin{cases} x_1^* = 0 \text{ et } x_2^* = \pm \frac{1}{\sqrt{2}} \text{ et } \lambda^* = -\frac{1}{2} \\ \text{ou bien} \\ x_2^* = 0 \text{ et } x_1^* = +1 \text{ et } \lambda^* = -1 \end{cases}$$

$$\Rightarrow 2 \text{ minimasesen } \begin{bmatrix} 0 \\ \frac{1}{\sqrt{2}} \end{bmatrix} \text{ et } \begin{bmatrix} 0 \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \text{ et } 2 \text{ minimasesen } \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{ et } \begin{bmatrix} -1 \\ 0 \end{bmatrix}$$

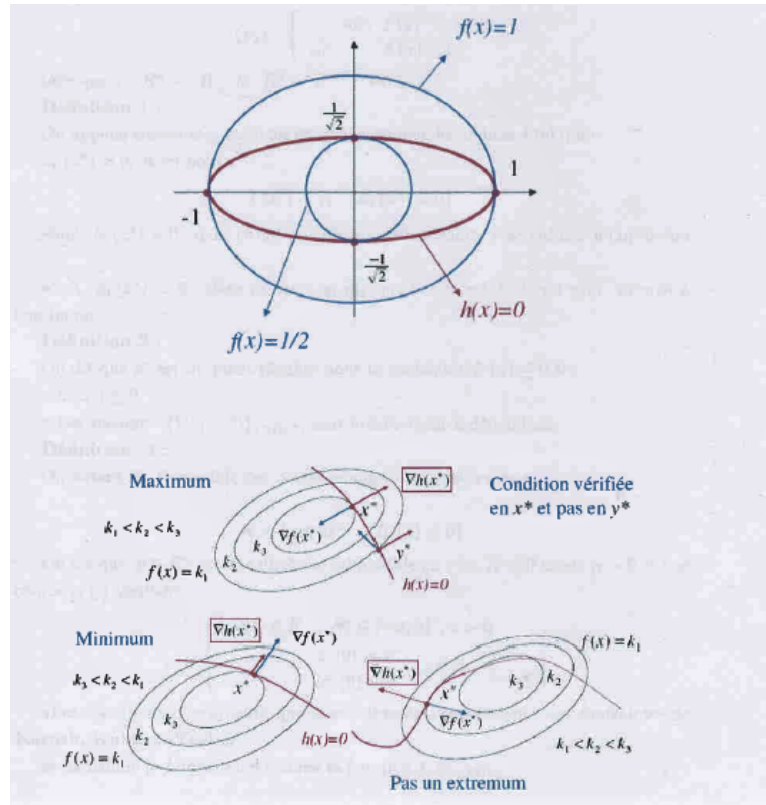


FIG. 1.6 – Multiplicateurs de Lagrange

## 1.7 Conditions de Karush Kuhn et Tucker

### 1.7.1 Problème avec contraintes d'inégalités

On considère le problème d'optimisation :

$$\begin{cases} \min f(x), \\ \text{s.c} \quad h(x) \leq 0. \end{cases} \quad (1.5)$$

telle que  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $h : \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $m < n$ .

**Définition 1.7.1** On appelle contraintes saturées en  $x^*$  l'ensemble des indices  $i$  tel que :

$h_i(x^*) = 0$ , et on note :

$$I(x^*) = \{i \mid h_i(x^*) = 0\}$$

1. Soit  $h_i(x^*) = 0$  : dans ce cas, on dit que la contrainte  $i$  est saturée à l'optimum.
2. Soit  $h_i(x^*) < 0$  : dans ce cas, on dit que la contrainte  $i$  est non saturée à l'optimum.

**Définition 1.7.2** On dit que  $x^*$  est un point régulier pour la contrainte  $h(x) \leq 0$  si:

- $h(x^*) \leq 0$ .
- Les vecteurs  $\{\nabla h_i(x^*)\}_{i \in I(x^*)}$  sont linéairement indépendants.

**Définition 1.7.3** On notera  $K$  l'ensemble des points admissibles, c'est à dire :

$$K = \{x \in \mathbb{R}^n, \quad h(x) \leq 0\}.$$

On dit que  $y \in \mathbb{R}^n$  est une direction admissible en  $x^* \in K$  s'il existe  $\alpha > 0$  et une courbe  $x(t)$  vérifiant :

$$\begin{cases} x(t) \in K & , & \forall t \in [-\alpha, \alpha] & , & \alpha > 0 \\ & & x(0) = x^* \\ & & \dot{x}(0) = y. \end{cases}$$

★ Les conditions d'optimalité que nous allons obtenir s'appelleront Les contraintes de **Karush, Kuhnet Tucher**, et on définit le Lagrangien comme la fonction  $L$  suivante :

$$L(x, \lambda) = f(x) + \sum_{i=1}^m \lambda_i h_i(x).$$

où les variables  $\lambda_i$  sont les multiplicateurs de **Lagrange** associés à chaque contrainte  $i$  et  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)$ .

## 1.7.2 Conditions de qualification des contraintes

Pour pouvoir utiliser le **lagrangien** dans la résolution d'un programme d'optimisation sous contraintes prenant la forme d'inéquations, il suffit que l'une des conditions suivantes soit vérifiée :

- Soit  $s \leq m$  le nombre de contraintes saturées à l'optimum  $x^*$ . On suppose sans perte de généralité qu'il s'agit des  $s$  fonctions contraintes, notée  $J_G$  et de taille  $(s, n)$ , est de



rang  $s$  lorsqu'elle est évaluée à l'optimum  $x^*$ , alors la condition de qualification des contraintes est vérifiée.

- Les fonctions contraintes  $h_i$ ,  $i = 1, \dots, s$  sont toutes linéaires.

### Conditions du premier ordre (Kuhn et Tucker)

On suppose que la contrainte de qualification est vérifiée. Si le vecteur  $x^* = (x_1^*, x_2^*, \dots, x_n^*)$  est une solution du problème de maximisation (1.2), alors il existe un unique vecteur  $\lambda^*$  tel que  $x^*$  vérifie les  $n + 3m$  conditions suivantes :

$$\left\{ \begin{array}{l} \frac{\partial L(x^*, \lambda^*)}{\partial x_j} = 0 \quad \Leftrightarrow \quad \frac{\partial f(x^*)}{\partial x_j} - \sum_{i=1}^m \lambda_i^* \frac{\partial h_i(x^*)}{\partial x_j} = 0 \quad , \forall j = 1, \dots, n \\ \frac{\partial L(x^*, \lambda^*)}{\partial \lambda_i} \geq 0 \quad \Leftrightarrow \quad h_i(x^*) \leq 0 \quad , \forall i = 1, \dots, m \\ \lambda_i^* \geq 0 \quad \Leftrightarrow \quad \lambda_i^* \geq 0 \quad , \forall i = 1, \dots, m \\ \lambda_i^* (h_i(x^*)) = 0 \quad \Leftrightarrow \quad \lambda_i^* = 0 \text{ ou } h_i(x^*) = 0 \quad , \forall i = 1, \dots, m \end{array} \right.$$

**Remarque 1.7.1** Ces conditions n'excluent pas la possibilité que  $\lambda_i^* = 0$  et  $h_i = 0$  simultanément.

### Conditions suffisantes du second ordre pour un optimum local :

Supposons qu'il existe un  $x^*$  qui vérifie les CPO. Soit  $s$  le nombre de contraintes saturées à l'optimum. Sans perte de généralité, on suppose qu'il s'agit des  $s$  premières. On désigne par  $H_L$  la matrice hessienne du **Lagrangien** bordée par les dérivées premières des seules contraintes saturées. Cette matrice est symétrique d'ordre  $s + n$ .

- ◆ Les  $n - s$  derniers mineurs principaux diagonaux de la matrice  $H_L(x^*, \lambda^*)$  évaluée à l'optimum sont alternativement  $> 0$  et  $< 0$ , le dernier d'entre eux ( $D_{n+s}$ ) étant du même signe que  $(-1)^n \implies x^*$  est un maximum local.
- ◆ Les  $n - s$  derniers mineurs principaux diagonaux de la matrice  $H_L(x^*, \lambda^*)$  évaluée à l'optimum sont tous du même signe (strictement) que  $(-1)^m \implies x^*$  est un minimum local.

### Théorème 1.7.1

### Condition suffisantes du second ordre pour un optimum global :

Supposons qu'il existe un  $x^*$  qui vérifie les CPO.

- ◆ Si  $f$  est concave et les  $h_i$  sont convexes  $\forall i = 1, \dots, m \implies x^*$  est un maximum global.
- ◆ Si  $f$  est convexe et les  $h_i$  sont concaves  $\forall i = 1, \dots, m \implies x^*$  est un minimum global.

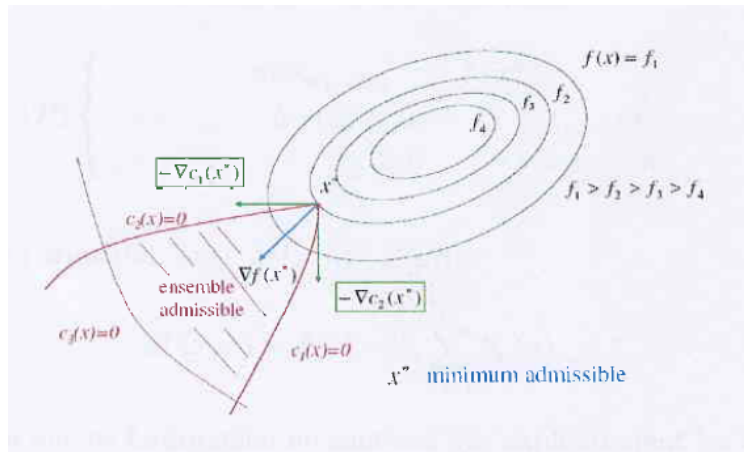


FIG. 1.7 – Condition du Kuhn et Tucker

**Remarque 1.7.2** *Il est très facile de passer d'un problème de minimisation sous contraintes prenant la forme d'inégalité à un problème de maximisation.*

Soit (1.6) le problème de minimisation suivant :

$$\begin{cases} \min f(x), \\ \text{s.c. } h_i(x) \geq 0 \quad \forall i = 1, \dots, m. \end{cases} \quad (1.6)$$

Pour pouvoir appliquer les conditions de Kuhn et Tucker évoquées ci-dessus, il suffit de transformer le problème (1.6) en un problème de maximisation (1.7) :

$$\begin{cases} \max -f(x), \\ \text{s.c. } -h(x) \leq 0. \end{cases} \quad (1.7)$$

Le Lagrangien s'écrit donc :

$$\begin{aligned} L(x, \lambda) &= -f(x) - \sum_{i=1}^m \lambda_i h_i(x) \\ &= -f(x) + \sum_{i=1}^m \lambda_i h_i(x) \end{aligned}$$

### 1.7.3 Le lagrangien Modifié

Dans de nombreuses applications économiques, les problèmes d'optimisation considérés imposent que les variables considérées ne puissent pas être négatives.

On considère le problème de maximisation (1.8) suivant :

$$\begin{cases} \max_{x_1, \dots, x_n} & f(x). \\ \text{s.c} & h_i(x) \leq 0 \quad \forall i = 1, \dots, m \\ & x_j \geq 0 \quad \forall j = 1, \dots, n \end{cases} \quad (1.8)$$

**Définition 1.7.4** *Le Lagrangien modifié, noté  $M(x, \lambda)$ , s'écrit :*

$$M(x, \lambda) = f(x) - \sum_{i=1}^m \lambda_i h_i(x).$$

On remarquera que ce Lagrangien ne contient pas explicitement les contraintes de non négativité  $x_j \geq 0$ ,  $j = 1, \dots, n$ .

**Condition du premier ordre(Kuhn et Tucker) :**

On suppose que la contrainte de qualification est vérifiée. Si le vecteur  $x^* = (x_1^*, \dots, x_n^*)$  est une solution du problème de maximisation (1.8), alors il existe un unique vecteur  $\lambda^*$  tel que  $x^*$  vérifie les 3  $(n + m)$  conditions suivantes :

$$\begin{cases} \frac{\partial M(x^*, \lambda^*)}{\partial x_j} \leq 0 & \Leftrightarrow & \frac{\partial f(x^*)}{\partial x_j} - \sum_{i=1}^m \lambda_i^* \frac{\partial h_i(x^*)}{\partial x_j} \leq 0 & \forall j = 1, \dots, n \\ x_j^* \geq 0 & & & \forall j = 1, \dots, n \\ x_j^* \frac{\partial M(x^*, \lambda^*)}{\partial x_j} = 0 & \Leftrightarrow & x_j^* = 0 \quad \text{ou} \quad \frac{\partial M(x^*, \lambda^*)}{\partial x_j} = 0 & \forall j = 1, \dots, n \\ \frac{\partial M(x^*, \lambda^*)}{\partial \lambda_i} \leq 0 & \Leftrightarrow & h_i(x^*) \leq 0 & \forall i = 1, \dots, m \\ \lambda_i^* \geq 0 & & & \forall i = 1, \dots, m \\ \lambda_i^* h_i(x^*) = 0 & \Leftrightarrow & \lambda_i^* = 0 \quad \text{ou} \quad h_i(x^*) = 0 & \forall i = 1, \dots, m \end{cases}$$

# Chapitre 2

## Les Algorithmes D'Optimisation Sous Contraintes

### Introduction

**N**ous consacrons ce chapitre aux principes des méthodes les plus importantes de recherche de minimum d'une fonction que ce soit convexe ou quelconque dans le cas de l'optimisation sous contraintes.

Ces méthodes sont : La méthode du gradient projeté, méthode des directions réalisables, méthodes de pénalisation, méthode d'Uzawa et méthode des gradients conjugués.

### 2.1 Méthodes Admissibles

Les méthodes admissibles consistent à trouver un point de l'ensemble admissible

$$X^{ad} = \{v \mid \theta(v) \leq 0\}.$$

Puis à se déplacer exclusivement dans cet ensemble, à la recherche du point  $x^*$ , solution de  $\min_{x \in X^{ad}} f(x)$ .

**Les avantages principaux :**

- A chaque étape de l'algorithme, l'approximation de la solution trouvée est admissible.
- On peut en général prouver que la suite de valeur engendrée converge vers au moins

un minimum local du problème considéré.

■ Ces méthode ne reposent sur aucune forme spéciale (convexité) du problème.

Elle sont donc très générales.

**Les inconvénients :**

◆ Il n'est pas toujours facile d'obtenir des pointes admissibles, ni même de rester admissible d'une itération à la suivante.

### 2.1.1 La méthode du Gradient Projeté

On s'intéresse à un problème avec contraintes d'égalité linéaires :

$$\begin{cases} \min_{x \in \mathbb{R}^n} f(x), \\ Ax - b = 0. \end{cases}$$

et nous ferons l'hypothèse que  $A \in M_{pn}$  est de rang maximal. Une idée assez naturelle consiste à appliquer une méthode de descente qui prenne en compte la contrainte  $Ax - b = 0$ . Supposons que nous disposons d'un point  $x_0 \in K = \{x \in \mathbb{R}^n, Ax - b = 0\}$ . On sait qu'une direction admissible doit vérifier

$$Ad = 0. \tag{2.1}$$

On peut chercher la meilleure direction de descente respectant en résolvant le problème

$$\begin{cases} \min \nabla f(x)^\top d, \\ Ad = 0, \\ \|d\| = 1. \end{cases} \tag{2.2}$$

**Proposition 2.1.1** *Le vecteur  $d$  solution du problème (2.2) est donné par  $d = \frac{y}{\|y\|}$  où  $y$  est la projection orthogonale de*

★  $\nabla f(x)$  sur  $\ker A$ .

**Preuve.** On peut écrire que :

$$-\nabla f(x) = y + z,$$

où  $y \in \ker A$  et  $z \in (\ker A)^\perp$ , ces deux sous-espaces étant complémentaires dans  $R^n$ . On a donc

$$-\nabla f(x)^\top d = -y^\top d.$$

Comme  $d$  est un vecteur unitaire quelconque  $y^\top d$  sera maximal pour  $d = \frac{y}{\|y\|}$ , d'où le résultat.

On remarquera que si  $y \neq 0$ , le vecteur  $d$  est bien une direction de descente car on a :

$$\nabla f(x)^\top = -y^\top (y + z) = -y^\top y < 0.$$

Pour former la matrice de projection sur  $\ker A$  on utilise en général la factorisation  $QR$  de la matrice  $A^\top$ , qui s'exprime sous la forme.

$$A^\top = Q \begin{pmatrix} \mathbb{R} \\ 0 \end{pmatrix}$$

où  $\mathbb{R} \in M_{pp}$  est triangulaire supérieure et  $Q \in M_{n,p}$  est orthogonale, et se décompose en  $Q = [U \ V]$  où les colonnes de  $U \in M_{n,p}$  forment une base orthogonale de  $\text{Im } A^\top$  et les colonnes de  $V \in M_{n,n-p}$  une base orthogonale de  $(\text{Im } A^\top)^\perp = \ker A$ . Dans ce cas la matrice de la projection orthogonale sur  $\ker A$  s'écrit :

$$P = I - UU^\top = VV^\top.$$

■

**Remarque 2.1.1** *Dans l'algorithme que nous allons étudier, la matrice de projection peut être calculée une fois pour toutes puisque  $A$  donnée. Cependant, pour les problèmes avec contraintes d'inégalité linéaire, on sera amené à considérer une succession de problème avec contraintes d'égalité, et la matrice  $A$  pourra évoluer à chaque itération, par ajout ou suppression d'une ligne. Le choix de la factorisation  $QR$  est tout indiqué car il existe des*

techniques de mise à jour particulièrement économiques, ce qui n'est pas le cas quand on exprime la matrice  $P$  sous la forme classique :

$$P = I - A^\top [AA^\top]^{-1} A.$$

La méthode du gradient projeté consiste tout simplement à mettre en œuvre une méthode de descente utilisant à chaque pas la direction  $d_k = -VV^\top \nabla f(x_k)$ . Les itérations sont poursuivies jusqu'à ce que  $d_k = 0$ .

Cela signifie alors que  $\nabla f(x) \in \text{Im } A^\top$  et donc qu'il existe  $\lambda$  tel que :

$$\nabla f(x_k) = -A^\top \lambda.$$

On peut utiliser la factorisation de  $A^\top$  pour obtenir  $\lambda$  par résolution du système linéaire :

$$R\lambda = -U^\top \nabla f(x).$$

### Algorithme du gradient projeté :

- Poser  $K = 0$  et choisir  $x_0$  admissible.
- Calculer la projection  $d_k = -VV^\top \nabla f(x)$ .
- Si  $d_k = 0$ 
  - o Calculer  $\lambda = R^{-1}U^\top \nabla f(x)$ .
  - o Arrêter les itérations.
- Déterminer  $\rho_k > 0$  réalisant le minimum de  $f(x_k + \rho d_k)$ .
- Poser  $x_{k+1} = x_k + \rho_k d_k$  faire  $k \leftarrow k+1$  et retourner en 2.

### 2.1.2 Méthode de Newton projeté

La méthode du **gradient projeté** souffrant des mêmes problèmes que la méthode du gradient (vitesse de convergence très sensible au conditionnement), on lui préfère souvent les méthodes de *Quasi-newton* adaptées au cas des contraintes linéaires. Il est plus facile de comprendre comment fonctionnent ces méthodes en faisant l'analyse suivante :

Supposons que l'on dispose d'un point  $x_0$  admissible. l'idée est de poser  $x = x_0 + V_z$  et de considérer une nouvelle fonction  $\tilde{f}$  définie par :

$$\tilde{f}(z) = f(x_0 + V_z),$$

où les colonnes de  $V$  forment une base orthogonale de  $\ker A$  (on a vu comment obtenir une telle matrice).

Alors par construction le problème est équivalent au problème sans contraintes :

$$\min_{x \in \mathbb{R}^p} \tilde{f}(z). \tag{2.3}$$

puisque :

$$A(x_0 + V_z) - b = Ax_0 - b + AV_z = 0.$$

On peut donc appliquer n'importe quelle méthode de descente pour résoudre de (2.3).

Notons que l'on a :

$$\nabla \tilde{f}(z) = V^\top \nabla f(x_0 + V_z),$$

donc la méthode du gradient appliquée à la minimisation de  $\tilde{f}(z)$  s'écrit :

$$z_{k+1} = z_k - \rho V^\top \nabla f(x_0 + V_{z_k}),$$

et si on pose  $x_k = x_0 + V_{z_k}$ , les itérations précédentes s'écrivent :

$$x_{k+1} = x_k - \rho k V V^\top \nabla f(x_k),$$

ce qui redonne exactement la méthode du *gradient projeté*. On peut de la même manière écrire la méthode de **Newton** appliquée à la résolution de (2.3), le hessien de  $\tilde{f}$  s'écrit :

$$\nabla^2 \tilde{f}(z) = V^\top \nabla^2 f(x_0 + V_z) V,$$



si on note  $G_k = \nabla^2 \tilde{f}(z_k)$  la direction de **Newton** en  $z_k$  s'écrit :

$$p_k = -G_k^{-1} \nabla f(z_k).$$

Si la matrice  $G_k$  est définie positive alors  $p_k$  sera une direction de descente pour  $\tilde{f}$  et le vecteur  $Vp_k$  sera une direction de descente puisque :

$$\nabla f(x_k)^\top Vp_k = \nabla \tilde{f}(z_k)^\top p_k < 0.$$

**Remarque 2.1.2** *On sait que dans le cas général un optimum local du problème (1.2) est caractérisé par :*

$$y^\top \nabla_{xx}^2 L(x^*, \lambda^*) y \geq 0 \quad , \quad \forall y \in T(x^*) \quad , \quad y \neq 0.$$

Où dans le cas des contraintes linéaires on a :

$$\nabla_{xx}^2 L(x^*, \lambda^*) = \nabla^2 f(x), \tag{2.4}$$

et le sous espace  $T(x^*)$  n'est autre que  $\ker A$ . Et donc si l'on dispose d'une matrice  $V$  dont les colonnes forment une base orthogonale de  $\ker A$ , tout vecteur  $y \in T(x^*)$  s'exprime sous la forme  $y = Vz$  et la condition (2.4) s'écrit :

$$zV^\top \nabla^2 f(x^*) Vz > 0 \quad , \quad \forall z.$$

On est donc assuré que le hessien projeté est défini positif à l'optimum, ce qui justifie l'utilisation des méthodes **quasi-Newton**.

On peut donc envisager une méthode de **quasi-Newton** ou la mise à jour à opère non pas sur le hessien de  $f$  mais sur le hessien projeté. Voici l'algorithme correspondant pour la méthode **BFGS** :

**Algorithme de la méthode BGFS projetée :**

– Poser  $k = 0$ , choisir  $x_0$  admissible est poser  $H_0 = I$ .

- Poser  $g_k = V^\top \nabla f(x_k)$ .
- Si  $g_k = 0$
- Calculer  $\lambda = -\mathbb{R}^{-1}U^\top \nabla f(x_k)$ .
- Arrêter les itérations.
- Calculer la direction  $p_k = -H_k^{-1}g_k$ .
- Déterminer  $\rho_k > 0$  réalisant le minimum de  $f(x_k + \rho V p_k)$ .
- Poser  $x_{k+1} = x_k + \rho V p_k$ .
- Calculer  $g_{k+1} = V^\top \nabla f(x_{k+1})$  et  $y_k = g_{k+1} - g_k$ .
- Mise à jour du hessien projeté  $H_{k+1} = H_k + \frac{y_k y_k^\top}{\rho_k y_k^\top p_k} + \frac{g_k g_k^\top}{p_k^\top g_k}$ .
- Faire  $k \leftarrow k + 1$  et retourner en 2.

### 2.1.3 Méthode de Directions Réalisables

On s'intéresse maintenant à un problème avec contraintes d'inégalités linéaires :

$$\begin{cases} \min_{x \in R} f(x), \\ \text{s.c} \quad Ax - b \leq 0. \end{cases}$$

On peut essayer de voir comment adapter la stratégie de l'algorithme du **gradient projeté**. Supposons que nous disposons d'un point initial admissible  $x_0 \in K = \{x \in R_n, Ax - b \leq 0\}$ .

Notons  $I_0$  l'ensemble des indices des contraintes saturées, soit :

$$I_0 = \{i \mid A_i x_0 - b_i = 0\}.$$

On peut chercher une direction de descente  $d$  qui permette, au moins pour un petit déplacement, de rester dans  $K$ . Si on note  $A_0 \in M_{pn}$  la matrice composée des lignes  $i \in I_0$  on doit donc avoir

$$A_{I_0} d = 0.$$

Après calcul de la factorisation  $\begin{pmatrix} U & V \end{pmatrix} \begin{pmatrix} \mathbb{R} \\ 0 \end{pmatrix}$  de  $A_{I_0}^\top$  une direction admissible  $d$  peut être obtenue par :

$$d = -VV^\top \nabla f(x_0).$$

Il y a ensuite deux cas à envisager :

- ★ Si  $d \neq 0$ , il faut déterminer le déplacement maximal autorisé par les contraintes non saturées, c'est à dire

$$\rho_{\max} = \{\rho \mid \rho \geq 0, A_i(x_0 + \rho d) - b_i \leq 0, i \notin I_0\}.$$

Ensuite, on cherche le pas optimal  $\rho_{opt}$  dans la direction  $d$ , Ce pas pouvant faire sortir du domaine admissible, on prendra donc toujours

$$\rho = \min(\rho_{opt}, \rho_{\max}),$$

en notant bien que lorsque  $\rho = \rho_{\max}$ , Cela signifie qu'une nouvelle contrainte sera saturée.

- Si  $d = 0$  cela signifie que  $\nabla f(x) \in \text{Im } A_{I_0}^\top$  et donc qu'il existe  $\lambda$  tel que :

$$\nabla f(x) = -A_{I_0}^\top \lambda$$

et qui s'obtient par résolution du système linéaire

$$\mathbb{R}\lambda = -\mathbb{U}^\top \nabla f(x)$$

et il faut ensuite considérer deux cas :

- Si  $\lambda \geq 0$ , alors  $x$  satisfait les condition de **Kuhn et Tucker**. Le point  $x$  est donc un optimum local du problème.
- Sinon, on supprime dans  $I_0$  une des contraintes pour les quelles  $\lambda_i < 0$  (par exemple la plus négative).

On obtient alors une nouvelle matrice  $A_1$  qui permet de déterminer une nouvelle direction de descente en  $x_0$ . On peut ensuite poursuivre les itérations.

On peut donc résumer l'algorithme de la façon suivant :

**Algorithme du gradient projeté (contraintes d'inégalité):**

1. Poser  $k = 0$  et choisir  $x_0$ .
2. Déterminer  $I_k = \{i \mid A_i x_k - b_i = 0\}$ .

3. Former la matrice  $A_{I_k} = \{A_i\}_i \in I_k$ .
4. Calculer ou mettre à jour la factorisation  $A_{I_0}^\top = \begin{bmatrix} U_k & V_k \end{bmatrix} \begin{pmatrix} \mathbb{R}_k \\ 0 \end{pmatrix}$ .
5. Calculer la projection  $d_k = -V_k V_k^\top \nabla f(x_k)$ .
6. Si  $d_k = 0$ 
  - Calculer la  $\lambda = -(R_k)^{-1} U_k^\top \nabla f(x_k)$
  - Si  $\lambda \geq 0$ , alors on s'arrête
  - Sinon, choisir  $j$  tel que  $\lambda_j \leq \lambda_i, \forall i$ , faire  $I_k = I_k - \{j\}$  et retourner en **3**.
7. Calculer  $\rho_{\max} = \{\rho \mid \rho \geq 0, A_i(x_k + \rho d_k) a - b_i \leq 0, i \notin I_k\}$ .
8. Déterminer  $\rho_k$  réalisant le minimum de  $f(x_k + \rho d_k)$  sur  $[0, \rho_{\max}]$ .
9. Poser  $x_{k+1} = x_k + \rho_k d_k$ , faire  $k \leftarrow k + 1$  et retourner en **2**.

## 2.1.4 Méthode de Pénalisation

### Méthode de Pénalisation Externe :

On considère un problème avec contraintes d'inégalité non-linéaires :

$$\begin{cases} \min_{x \in \mathbb{R}^n} f(x), \\ s.c \quad g(x) \leq 0. \end{cases} \quad (2.5)$$

Le but des méthodes de pénalisation est de résoudre (2.5) de façon approchée de la façon suivante : on définit la fonction  $\varphi(x)$  par :

$$\varphi(x) = \sum_{i=1}^m (g_i^+(x))^2,$$

où  $[\cdot]^+$  est la fonction partie positive définie par :

$$y^+ = \max(0, y).$$

Si on note  $K = \{x \in R^n, g(x) \leq 0\}$ , la fonction  $\varphi$  vérifie par construction :

$$\begin{cases} \varphi(x) = 0, & \text{pour } x \in R. \\ \varphi(x) > 0, & \text{pour } x \notin R. \end{cases}$$

On introduit alors le problème (2.6) :

$$\begin{cases} \min_{x \in R^n} f_\varepsilon(x), \\ f_\varepsilon(x) = f(x) + \frac{1}{\varepsilon}\varphi(x). \end{cases} \quad (2.6)$$

dont on notera  $x_\varepsilon$  la solution, vérifiant

$$f_\varepsilon(x_\varepsilon) \leq f_\varepsilon(x) \quad , \forall x \in R^n$$

Le nom de pénalité extérieure provient du fait que  $x_\varepsilon$  est toujours à l'extérieur (au sens large) de  $K$  comme le montre le résultat suivant :

**Proposition 2.1.2** *S'il existe au moins une contrainte saturée à l'optimum  $x^*$  du problème (2.5) alors le vecteur solution du problème pénalisé (2.6) vérifie nécessairement*

$$\exists i_0, g_{i_0}(x_\varepsilon) \geq 0.$$

**Preuve.** Utilisant l'absurde : Si  $g_i(x_\varepsilon) < 0, \forall i$  on a par définition  $x_\varepsilon \in K$ . Puisque

$$f_\varepsilon(x_\varepsilon) \leq f_\varepsilon(x) \quad , \forall x \in R^n$$

donc en particulier pour  $x = x^*$ , on a :

$$f_\varepsilon(x_\varepsilon) \leq f_\varepsilon(x^*),$$

mais comme  $x_\varepsilon \in K$  et  $x^* \in K$  on a :

$$\varphi(x_\varepsilon) = \varphi(x^*) = 0,$$

et donc :

$$f(x_\varepsilon) \leq f(x^*).$$

D'où  $x_\varepsilon = x^*$ . On a donc  $g_i(x^*) < 0, \forall i$  et aucune contrainte n'est saturée en  $x^*$ .

En général, on a toujours  $x^* \notin K$ .

$x_\varepsilon$  tend vers une solution du problème (2.5) quand  $\varepsilon$  tend vers 0. ■

**Théorème 2.1.1** Soit  $\varphi : R^n \rightarrow R$  une fonction de pénalisation extérieure vérifiant :

- $\varphi(x) \geq 0$ .
- $\varphi(x) = 0 \Leftrightarrow x \in K$ .
- $\varphi$  continue.

On suppose  $f$  continue,  $K$  fermé et que l'une des deux conditions suivantes est vérifiée :

- $f(x) \rightarrow +\infty$  quand  $\|x\| \rightarrow +\infty$ .
- $K$  est borné et  $\varphi(x) \rightarrow +\infty$  quand  $\|x\| \rightarrow +\infty$ .
- $\varphi$  continue.

Alors, quand  $\varepsilon_k$  tend vers 0, la suite  $x_{\varepsilon_k}$  admet au moins d'accumulation qui est alors une solution optimale du problème (2.5).

Lorsqu'on met en œuvre cette méthode de façon pratique, on ne peut pas prendre tout de suite  $\varepsilon_k$  très petite, à cause des problèmes de conditionnement que cela peut causer. On commence donc avec une valeur du type  $\varepsilon_0 = 1$ , et chaque solution  $x_{\varepsilon_k}$  est prise comme vecteur initial pour résoudre le problème avec  $\varepsilon_{k+1} = \frac{\varepsilon_k}{100}$  (par exemple)

**Exemple 2.1.1**  $f(x) = 2x$       $c(x) = 1 - x \leq 0$

$$p(x) = \max(1 - x, 0)^2$$

$$g(x, \alpha_k) = 2x + \alpha_k \max(1 - x, 0)^2 \quad \alpha_1 < \alpha_2 < \alpha_3$$

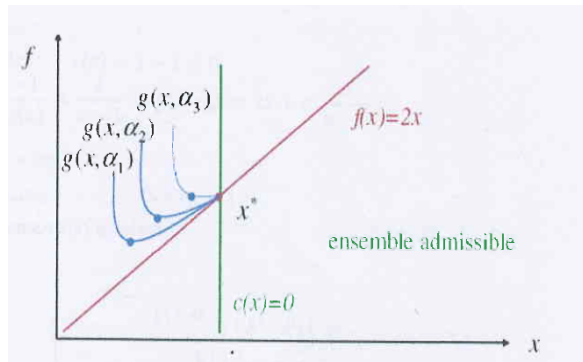


FIG. 2.1 – Méthode de Pénalisation Externe

### Algorithme de la méthode de pénalisation :

- ◆ Choisir  $x_0, \varepsilon_1$  et poser  $k = 1$ .
- ◆ Trouver  $x_k$  solution du problème  $\min_{x \in R} f_{\varepsilon_k}(x)$  en partant de  $x_{k-1}$ .
- ◆ Poser  $\varepsilon_{k+1} = \frac{\varepsilon_k}{100}$ .
- ◆ Faire  $k \leftarrow k + 1$  et retourner en **2**.

### Méthode de Pénalisation Interne :

Dans le cas des méthodes internes, en général,  $x_\varepsilon$  n'est jamais dans  $K$  (sauf cas particulier) : Cela peut poser de problème si par exemple la fonction  $f$  n'est pas définie hors de  $K$ . Les méthodes internes permettent d'éviter cet inconvénient. Leur principe est le même que les méthodes externes : On considère une fonction

$$f_\varepsilon(x) = f(x) + \varepsilon \Psi(x),$$

mais ici la fonction  $\Psi(x)$  est définie pour  $x \in K$  et est du type :

$$\Psi(x) = \sum_{i=1}^m \frac{1}{g_i(x)^2}.$$

Puisque l'on a  $\Psi(x) \rightarrow \infty$  quand on s'approche de la frontière de  $K$ , on qualifie souvent de  $\Psi$  fonction barrière. Les propriétés de convergence sont les mêmes que pour les méthodes externes, mais il faut ici disposer d'un  $x_0 \in K$ , ce qui peut être difficile dans certains cas.

**Exemple 2.1.2**  $f(x) = 2x \quad c(x) = 1 - x \leq 0$

$$P(x) = \frac{-1}{c(x)} = \frac{1}{x-1} \Rightarrow g(x, \alpha_k) = 2x + \alpha_k \frac{1}{x-1}$$

$$\alpha_1 > \alpha_2 > \alpha_3$$

$$\alpha_{k+1} = c_\alpha \alpha_k \quad 0 < c_\alpha < 1$$

$\Rightarrow$  *Extrapolation à la limite.*

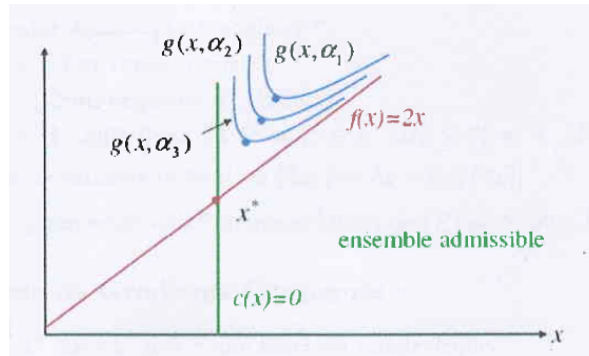


FIG. 2.2 – Méthode de Pénalisation Externe

### 2.1.5 Méthode d'Uzawa

Le principe de la méthode d'*Uzawa* est d'utiliser la méthode du gradient pour maximiser la fonction duale, tout en tenant compte de la contrainte  $\lambda \geq 0$  : Cela donne la méthode :

$$\lambda_{k+1} = [\lambda_k + \rho_k \nabla \omega(\lambda_k)]^+.$$

L'utilisation de cette méthode suppose que la fonction duale est différentiable (au moins à l'optimum). Ce sera le cas si le minimum en  $x$  de  $L(x, \lambda^*)$  est unique. Dans ce cas si on note  $x(\lambda)$  le vecteur tel que :

$$\omega(\lambda) = L(x(\lambda), \lambda)$$

on peut écrire que :

$$\begin{aligned} \nabla \omega(\lambda) &= \nabla_x L(x(\lambda), \lambda) \frac{dx(\lambda)}{d\lambda} + \nabla_\lambda L(x(\lambda), \lambda) \\ &= g(x(\lambda)) \end{aligned}$$

puisque  $x(\lambda)$  est par définition le minimum en  $x$  de  $L(x, \lambda)$ . L'algorithme de la méthode est donc le suivant :



**Algorithme d'Uzawa :**

1. Poser  $k = 0$  et  $\lambda_0 = 0$ .
2. Déterminer  $x_k$  solution du problème  $\min_{x \in R^n} f(x) + \lambda_k^\top g(x)$ .
3. Si  $\max_i g_i(x_k) < \varepsilon$  alors on s'arrête.
4. Sinon, calculer  $\lambda_{k+1} = [\lambda_k + \rho_k g(x_k)]^+$ .
5. Faire  $k \leftarrow k + 1$  et retourner en **2**.

**Théorème 2.1.2** (*Convergence d'Uzawa*) Soit  $J : R^n \rightarrow R$ , elliptique, et  $U = \{v \in R^n; Bv \leq d\} \neq \emptyset$  ( $B \in R^m * R^n$ ) si  $0 < \rho < \frac{2\alpha}{\|B\|^2}$  ou  $\alpha$  satisfait la relation  $[\lambda_{k+1} = \lambda_k + \rho_k g(x_k)]^+$  la suite  $(U_k)_{k \in N}$  converge vers l'unique solution de (P) si de plus  $B$  est de rang  $m$ .

### 2.1.6 Méthode de Gradients Conjugués

Soit  $q(x) = \frac{1}{2}x^\top Ax + b^\top x + c$  une fonction quadratique.

**Définition 2.1.1** Soit  $A_{(n \times n)}$  une matrice symétrique, on dit que deux vecteur  $U$  et  $V$  de  $R^n$  sont conjugués par rapport à  $A$  si  $U^\top AV = 0$ .

**L'algorithme du Gradient Conjugué :**

Après avoir transformé le problème d'optimisation avec contraintes en le problème sans contraintes, on peut imaginer l'introduction de la méthode du gradient conjugué. L'idée de la méthode est de construire itérativement des directions  $d_0, \dots, d_k$  mutuellement conjuguées. A chaque étape  $k$  la direction  $d_k$  est obtenue comme combinaison linéaire du gradient en  $x_k$  et de la direction précédente  $d_{k-1}$ , les coefficients étant choisis de telle manière que  $d_k$  soit conjuguée avec toutes les directions précédentes.

Si l'on note  $g_k = \nabla f(x_k)$ , l'algorithme prend la forme suivante :

On se donne  $x_0$  et on pose  $d_0 = -g_0$ .

$$x_{k+1} = x_k + \rho_k d_k. \quad (2.7)$$

$$\rho_k = \frac{g_k^\top d_k}{d_k^\top A d_k}. \quad (2.8)$$

$$d_{k+1} = -g_{k+1} + B_k d_k. \quad (2.9)$$

$$B_k = \frac{g_{k+1}^\top A_k d_k}{d_k^\top A_k d_k}. \quad (2.10)$$

Notons d'une part que la formule (2.8) définit bien le pas optimal : en effet on a bien

$$\nabla f(x_{k+1})^\top d_k = g_k^\top d_k + \rho_k d_k^\top A_k d_k = 0.$$

On va maintenant montrer que l'algorithme ci-dessus définit bien une méthode de direction conjuguées.

**Théorème 2.1.3** *A une itération  $k$  quelconque de l'algorithme où l'optimum n'est pas encore atteint, c'est  $g_k \neq 0$ , on a :*

$$\rho_k = \frac{g_k^\top g_k}{d_k^\top A_k d_k}. \quad (2.11)$$

$$B_k = \frac{g_{k+1}^\top (g_{k+1} - g_k)}{g_k^\top g_k}. \quad (2.12)$$

$$B_k = \frac{g_{k+1}^\top g_{k+1}}{g_k^\top g_k}. \quad (2.13)$$

et les directions  $d_0, \dots, d_{k+1}$  sont mutuellement conjuguées.

**Preuve.** On raisonne par récurrence sur  $k$  en supposant  $d_0, \dots, d_k$  sont mutuellement conjuguées. ■

► Montrons d'abord l'équivalence de (2.8) et (2.11) Comme  $d_0, \dots, d_{k+1}$  sont mutuellement conjuguées

$x_k$  réalise le minimum de  $f$  sur  $x_0 + E_k$ , on a  $g_k^\top d_{k-1} = 0$  d'où

$$g_k^\top d_k = g_k^\top (-g_k + B_k d_{k-1}) = -g_k^\top g_k.$$

► Pour montrer (2.12) on note que :

$$g_{k+1} - g_k = A(x_{k+1} - x_k) = \rho A d_k. \quad (2.14)$$

on a alors :

$$g_{k+1}^\top Ad_k = \frac{1}{\rho_k} g_{k+1}^\top (g_{k+1} - g_k),$$

et en utilisant (2.11) il vient bien :

$$B_k = \frac{g_{k+1}^\top (g_{k+1} - g_k)}{g_k^\top g_k}.$$

ce qui démontre (2.12). On a de plus  $g_{k+1}^\top g_k = 0$  car  $g_k = d_k - B_{k-1}d_{k-1}$  appartient à  $E_{k+1}$  et que  $g_{k+1}$  est orthogonal au sous-espace (les directions  $d_0, \dots, d_{k+1}$  sont conjuguées, par hypothèse de récurrence), ceci démontre (2.13).

► Montrons maintenant que  $d_{k+1}^\top Ad_i = 0$ , pour  $i = 0, \dots, k$ . On a d'une part

$$d_{k+1}^\top Ad_k = (-g_{k+1} + B_k d_k)^\top Ad_k = 0,$$

par définition de  $B_k$ . D'autre part, on a pour  $i < k$

$$d_{k+1}^\top Ad_i = -g_{k+1}^\top Ad_i + B_k d_k^\top Ad_i,$$

avec  $d_k^\top Ad_i = 0$  en vertu de l'hypothèse de récurrence. On a ensuite, en utilisant la formule (2.14)

$$g_{k+1}^\top Ad_i = \frac{1}{\rho_i} g_{k+1}^\top (g_{i+1} - g_i),$$

et si l'on note que :

$$g_{i+1} - g_i = -d_{i+1} + (B_i + 1)d_i - B_{i-1}d_{i-1},$$

on a bien

$$g_{k+1}^\top (g_{i-1} - g_i) = 0,$$

car  $g_{k+1}^\top d_{i+1} = g_{k+1}^\top d_i = g_{k+1}^\top d_{i-1} = 0$ , en vertu du fait que  $g_{k+1}$  est orthogonal à  $E_{k+1}$  et que  $i < k$ . On a donc bien  $d_{k+1}^\top Ad_i = 0$ , ce qui achève la démonstration.

### La Méthode du Gradient Conjugué dans le cas général :

La méthode de **Fletcher** et **Reeves** est une extension directe de la méthode du **Gradient conjugué** pour les fonctions quelconques. Appliquée à une fonction quadratique, elle se

comporte comme cette dernière :

On se donne  $x_0$  et on pose  $d_0 = -\nabla f(x_0)$ .

$$x_{k+1} = x_k + \rho_k d_k, \text{ avec } \rho_k \text{ optimal.} \quad (2.15)$$

$$d_{k+1} = -\nabla f(x_{k+1}) + B_k d_k. \quad (2.16)$$

$$B_k = \frac{\|\nabla f(x_{k+1})\|^2}{\|\nabla f(x_k)\|^2}. \quad (2.17)$$

Cette méthode est intéressante car elle ne nécessite pas de stocker une matrice (contrairement aux méthodes qui seront vues dans les chapitres suivants). Sa vitesse de convergence est très supérieure à celle de la méthode du gradient (ce point sera clarifié pour le cas quadratique dans le prochain chapitre).

La variante dite de **Polak Ribière** consiste à définir  $B_k$  par la formule (2.12). On peut démontrer la convergence de la méthode de **Fletcher Reeves** pour une classe assez large de fonction  $f$ , ce qu'on ne peut pas faire pour la variante de **Polak Ribière**. Par contre on peut montrer que cette dernière converge plus rapidement (quand elle converge effectivement), c'est donc la méthode qui est utilisée en général.

L'efficacité de la méthode du **gradient conjugué** repose essentiellement sur deux points :

- ★ La recherche linéaire (détermination du pas optimal) doit être exacte.
- ★ Les relations de conjugaison doivent être précises.

La recherche du pas optimal doit être réalisée à l'aide d'un algorithme spécifique prochain puisque  $f$  est quelconque. Par contre la notion de conjugaison n'a pas de sens dans le cas non quadratique sauf près de l'optimum, mais on ne le connaît pas. Il faut donc tester au cours des itérations si l'hypothèse d'approximation quadratique est vérifiée. On peut surveiller les indicateurs suivants :

- ★  $|\nabla f(x_{k+1})^\top \nabla f(x_k)|$  doit être petit.

★ On doit avoir :

$$\frac{\nabla f(x_{k+1})^\top d_{k+1}}{\|\nabla f(x_{k+1})\| \|d_{k+1}\|} \leq -\alpha,$$

avec  $0 \leq \alpha \leq 1$  pas trop petit, c'est à dire que  $d_{k+1}$  doit être une direction de descente <<raisonnable>>.

Dans le cas où ces conditions ne sont pas vérifiées, on arrête la conjugaison et on redémarre l'algorithme avec  $d_{k+1} = -\nabla f(x_{k+1})$ . On peut aussi décider de faire ce redémarrage arbitrairement toutes les  $p$  itérations ( $p$  fixé de l'ordre de  $n$  par exemple).

### Convergence de la méthode du Gradient Conjugué :

Le résultat suivant va nous permettre de retrouver d'une autre manière la propriété de convergence finie de l'algorithme du **GC** :

**Proposition 2.1.3** *Soit  $A$  une matrice définie positive et  $x_k$  le vecteur obtenu à l'étape  $k$  de l'algorithme du **GC**. alors on a :*

$$E(x_k) \leq E(x_0) \min_{p \in P_k, p(0)=1} \max_{z \in \sigma(A)} p(z)^2.$$

**Preuve.** Puisque la matrice  $A$  est définie positive il existe une matrice orthogonale  $U$  telle que :

$A = UDU^\top$  avec  $D = \text{diag}(\lambda_1, \dots, \lambda_n)$ , où  $\sigma(A) = \{\lambda_i\}_{i=1\dots n}$  sont les valeurs propres de  $A$ . Si on définit  $A^{1/2} = UD^{1/2}U^\top$  on a

$$\|x\|_A^2 = \|A^{1/2}x\|^2,$$

donc  $\|p(A)(x_0 - x^*)\|_A^2 = \|A^{1/2}p(A)(x_0 - x^*)\|^2 \leq \|p(A)^{1/2}\|^2 \|x_0 - x^*\|^2$ .

où on a utilisé la propriété que  $p(A)$  et  $A^{1/2}$  commutent (ces deux matrices ont les mêmes vecteurs propres).

Puisque l'on a aussi  $A_j = UD_jU^\top$  les valeurs propres de  $p(A)$  sont données par les nombres  $p(\lambda_i)$  pour  $i = 1\dots n$ , et donc

$$\|p(A)\|^2 = \max_{i=1\dots n} p(\lambda_i)^2.$$

On a donc bien

$$E(x_k) \leq E(x_0) \min_{p \in P_k, p(0)=1} \max_{z \in \sigma(A)} p(z)^2.$$

On a le corollaire suivant, qui permet d'exhiber le polynôme optimal  $p(z)$  pour  $k = n$  : ■

**Théorème 2.1.4** *Soit  $A$  une matrice définie positive. L'algorithme du **GC** converge en  $n$  itérations au plus.*

*Plus précisément, si la matrice  $A$  possède  $k \leq n$  valeurs propres distinctes, alors l'algorithme du **GC** converge en  $k$  itérations au plus.*

**Preuve.** Dans les deux cas possibles, notons

$$\bar{p}(z) = \prod_{i=1}^k \frac{\lambda_i - z}{\lambda_i}.$$

On a bien  $\bar{p}(z)$  de degré  $k$ ,  $\bar{p}(0) = 1$  et par construction  $\bar{p}(\lambda_i) = 0$  pour  $i = 1, \dots, k$ .

En vertu du résultat montré dans la proposition précédent, on a donc :

$$E(x_k) = 0.$$

Soit  $x_k = x^*$ . La méthode du **gradient conjugué** étant en général utilisée comme une méthode itérative, il est intéressant de la comparer à la méthode du gradient à pas optimal.

Le résultat suivant sera admis (la démonstration repose sur la détermination d'un polynôme particulier  $p(z)$  solution d'un problème de moindre carré). ■

## 2.2 Convergence des Méthodes d'Optimisation

Nous ne pouvons pas nous attendre à qu'un problème non-linéaire soit résolu de façon exacte en nombre fini d'étape, tout ce que nous pouvons espérer est que la suite d'itérations  $\{x_t\}$  produite par la méthode en question converge vers l'ensemble de solution du problème quand  $t \rightarrow \infty$ . Dans la théorie d'optimisation numérique, la convergence d'une méthode d'optimisation sur certaine famille des problèmes est exactement ce qui donne le droit à la méthode d'être qualifiée comme un outil pour résoudre des problèmes de la famille. La

convergence n'est pas la seule caractéristique d'une méthode, mais c'est la propriété qu'en fait une routine d'optimisation théoriquement valide.

**Conclusion :**

Toutes les méthodes d'optimisation convergent en principe vers la même solution, mais la façon dont elles convergent sont différentes. On remarque que la méthode du gradient conjugué est plus rapide par rapport au les autres méthodes.

# Chapitre 3

## Problèmes Variationnels

### Introduction

Le calcul des variations est un calcul avec une infinités de variables. Ces variable sont souvent codifiés par les valeurs d'une fonction  $f : \mathbb{R} \rightarrow \mathbb{R}$ . Ainsi, les "fonctions" dans ce calcul, dites fonctionnels prennent des fonctions comme arguments et produisent des nombres. L'évaluation d'une fonctionnel  $E$  sur fonction  $f$  est notée  $E[f]$ . comme dans le cas de dimension finie, une condition nécessaire pour que  $E[f]$  soit minimale est que  $E'[f] = 0$ . L'objectif de ce chapitre est de fournir une signification à la notation  $E'$ .

par exemple, pour la fonctionnelle qui mesure la longueur d'un graphe  $y = y(x)$  :

$$E[y] = \int_a^b \sqrt{1 + y'(x)^2} dx$$

la condition  $E'[y] = 0$  est

$$\frac{y''(x)}{\sqrt{1 + y'(x)^2}^3} = 0$$

ce qui implique que la courbe la plus courte entre deux points est une droite ( $y'' = 0$ ).

Typiquement la condition  $E'[f] = 0$  revient à une EDO ou une EDP sur  $f$ . par exemple, nous obtenons le résultat que pour une fonctionnelle  $E[f]$  de la forme

$$E[f] = \int_a^b L(f(x), f'(x), x) dx$$



où  $L$  est une fonction de trois variables, la condition  $E'[f] = 0$  est

$$\frac{\partial L}{\partial f} - \frac{d}{dx} \frac{\partial L}{\partial f'} = 0$$

ce qui est une EDO de premier ordre sur  $f$ , dite équation d'Euler-Lagrange de E.

### 3.1 Exemples de fonctionnels

- ★ La longueur du graphe d'une fonction  $E[y] = \int_a^b \sqrt{1 + y'(x)^2} dx$ .
- ★ La longueur d'une courbe paramétrée  $E[q] = \int_a^b \|\dot{q}(t)\| dt$ .
- ★ L'action d'une trajectoire  $E[q] = \int_a^b \|\dot{q}(t)\|^2 dt$ .
- ★ L'action dans un milieu non-homogène  $E[q] = \int_a^b g(q(t)) \|\dot{q}(t)\|^2 dt$ .
- ★ L'énergie potentielle d'une chaîne suspendue  $E[y] = \int_a^b y(x) \sqrt{1 + y'(x)^2} dx$ .
- ★ La tension d'une membrane par petites déformations verticales :  $E[u] = \int_{\Omega} \|\nabla u(x)\|^2 dx$ .
- ★ La tension d'une plaque par petites déformations verticales :  $E[u] = \int_{\Omega} \|\Delta u(x)\|^2 dx$ .
- ★ L'aire d'une surface définie par un graphe :  $E[u] = \int_{\Omega} \sqrt{1 + \|\nabla u(X)\|^2} dx$ .
- ★ La pente maximale d'une fonction :  $E[u] = \sup_{x \in \Omega} \|\nabla u(X)\|$ .
- ★ La variation totale d'une fonction :  $E[u] = \int_{\Omega} \|\nabla u(X)\| dx$ .
- ★ L'énergie du  $p$ -laplacien :  $E[u] = \int_{\Omega} \|\nabla u(X)\|^p dx$ .
- ★ L'aire d'une surface paramétrée :  $E[u] = \int_{\Omega} \|X_u \times X_v\| du dv$ .
- ★ Le dé bruitage par variation totale d'une image  $I(X)$  :

$$E[u] = \int_{\Omega} (|u(X) - I(X)| + \lambda \|\nabla u(X)\|) dx.$$

- ★ L'erreur de flot optique entre deux frames vidéo :

$$E[u] = \int_{\Omega} [ |B(X + u(X)) - A(X)|^2 + \alpha^2 (u_x^2 + u_y^2 + v_x^2 + v_y^2) ] dx.$$

## 3.2 Constructions mathématiques

**Proposition 3.2.1 (Lemme fondamental du calcul de variations)** soit  $f : [a, b] \rightarrow \mathbb{R}$  une fonction continue telle que

$$\int_a^b v(x) f(x) dx = 0,$$

pour toute fonction  $v$  de classe  $C^1$  sur  $[a, b]$  avec  $v(a) = v(b) = 0$ . Alors la fonction  $f$  est identiquement nulle sur  $[a, b]$ .

**Définition 3.2.1 (Dérivée de Gâteaux)** soit  $V$  un espace vectoriel réel et  $f : V \rightarrow \mathbb{R}$  une fonction. La dérivée directionnelle de  $f$  au point  $x \in V$  en la direction  $y \in V$  est la limite

$$\lim_{\varepsilon \rightarrow 0} \frac{f(x + \varepsilon y) - f(x)}{\varepsilon}.$$

Dans le cas particulier où  $V = \mathbb{R}^n$  et  $f$  est différentiable, cette limite est  $\nabla f(x) \cdot y$ .

## 3.3 Fonction $y : \mathbb{R} \rightarrow \mathbb{R}$

### 3.3.1 Cas général

Soit  $L$  une fonction différentiable  $L : \mathbb{R}^3 \rightarrow \mathbb{R}$  appelée le lagrangien. notre but est de minimiser la fonctionnelle

$$E[y] = \int_a^b L(y(x), y'(x), x) dx,$$

dans l'espace des fonctions différentiables sur l'intervalle  $[a, b]$  telle que  $y(a) = \alpha$  et  $y(b) = \beta$  pour deux constantes  $\alpha$  et  $\beta$ .

Supposons que  $y$  est une fonction où le minimum de  $E$  est atteint. On va retrouver une condition nécessaire qui doit satisfaire la fonction  $y$ .

On prend une fonction différentiable  $v : [a, b] \rightarrow \mathbb{R}$  tel que  $v(a) = v(b) = 0$ . une telle fonction s'appelle variation à extrémités fixes. Considérons la fonction  $f : \mathbb{R} \rightarrow \mathbb{R}$  définie

par

$$f(\varepsilon) = E[y + \varepsilon v] = \int_a^b L(y(x) + \varepsilon v(x), y'(x) + \varepsilon v'(x), x) dx.$$

Observons que  $f'(0)$  est la dérivée directionnelle de  $E$  en  $y$  dans la direction  $v$ .

Comme  $y$  est un minimum de  $E$  alors la fonction  $f$  a un minimum en  $\varepsilon = 0$ , ce qui implique  $f'(0) = 0$ . Écrivons cette condition en utilisant la règle de composition :

$$\int_a^b \left( v \frac{\partial L}{\partial y} + v' \frac{\partial L}{\partial y'} \right) dx = 0,$$

en intégrant par parties le second terme de la somme,

$$\int_a^b \left( v \frac{\partial L}{\partial y} - v \frac{d}{dx} \frac{\partial L}{\partial y'} \right) dx = 0,$$

et appliquant le lemme fondamental du calcul de variations

$$\frac{\partial L}{\partial y} - \frac{d}{dx} \frac{\partial L}{\partial y'} dx = 0,$$

ce qui est une EDO de second ordre sur la fonction  $y(x)$ . Cette EDO caractérise les points stationnaires du fonctionnel  $E[y] = 0$ .

### 3.3.2 L'astuce de Beltrami

Dans le cas, courant, où le lagrangien  $L$  ne dépend explicitement de  $x$ , l'équation d'Euler-Lagrange est équivalente à

$$L - y' \frac{\partial L}{\partial y'} = K,$$

où  $K$  est une constante qu'il faut déterminer pour satisfaire les conditions de bord. Cette EDO est de premier ordre, donc plus simple que l'équation d'Euler-Lagrange.

### 3.3.3 Minimisation avec contraintes d'égalité

Plusieurs problèmes d'optimisation ont naturellement de contraintes. Par exemple, une chaîne pendue a une longueur fixe, et une bulle de savon fermée contient un volume d'air

constante. Comme dans le cas de dimension finie, les problèmes avec multiplicateurs de Lagrange.

Les minima de  $E[y]$  sous la contrainte  $G[y] = c$ , satisfont l'équation d'Euler-Lagrange  $(E + \lambda G)'[y] = 0$ . Le paramètre  $\lambda$  est un nombre réel que l'on détermine à posteriori en imposant la condition  $G[y] = c$ .

### 3.4 Fonction $q : \mathbb{R} \rightarrow \mathbb{R}^n$

Soit  $q : [a, b] \rightarrow \mathbb{R}^n$  une courbe paramétrée en  $\mathbb{R}^n$ . les problèmes variationnels sur la courbe  $q$  se traitent en considérant chaque composante  $q^i$  indépendamment. Ainsi, les extrema du fonctionnel

$$E[q] = \int_a^b L(q(t), \dot{q}(t), t) dt,$$

où  $L$  est une fonction de  $n + n + 1$  variables, satisfont les équations d'Euler-Lagrange

$$\frac{\partial L}{\partial q} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} = 0.$$

qui sont un système de  $n$  EDO de second ordre.

### 3.5 Fonction $u : \mathbb{R}^n \rightarrow \mathbb{R}$

Soit  $\Omega \subset \mathbb{R}^n$  un domaine borné et  $u : \Omega \rightarrow \mathbb{R}$ . Les problèmes variationnels sur fonctions à plusieurs variables comme  $u$  se traitent en prenant variations de  $u$  qui sont identiquement zéro sur  $\partial\Omega$ . Ainsi, les extrema d'un fonctionnel

$$E[u] = \int_{\Omega} L(u(X), \nabla u(X), X) dX,$$

où  $L$  est une fonction de  $1 + n + n$  variables, satisfont les équations d'Euler-Lagrange

$$\frac{\partial L}{\partial u} - \sum_{i=1}^n \frac{d}{dx^i} \frac{\partial L}{\partial u_{x^i}} = 0,$$

ce qui est une EDP de second ordre sur la fonction  $u$ . Cette équation peut encore s'écrire, avec beaucoup d'abus de notation, comme  $0 = \frac{\partial L}{\partial u} - \operatorname{div} \left( \frac{\partial L}{\partial \nabla u} \right)$ . Ici,  $\frac{\partial L}{\partial \nabla u}$  dénote le gradient de  $L$  par rapport aux  $n$  variables  $\nabla u$ .

Pour  $n = 2$ , par exemple, pour un lagrangien  $L = L(u, u_x, u_y, x, y)$  l'équation d'Euler-Lagrange est

$$\frac{\partial L}{\partial u} - \frac{d}{dx} \frac{\partial L}{\partial u_x} - \frac{d}{dy} \frac{\partial L}{\partial u_y} = 0.$$

### 3.6 Dérivées d'ordre supérieur

Parfois les lagrangiens contiennent des dérivés secondes ou d'ordre plus élevé. En dimension 1, le cas général est

$$E[y] = \int_a^b L(y(x), y'(x), y''(x), \dots, y^n(x), x) dx.$$

Pour minimiser ce fonctionnel on prend des variations  $v$  telles que ses  $n$  dérivées aux extrémités s'annulent. On obtient alors l'EDO d'ordre  $n + 1$  suivante :

$$0 = \frac{\partial L}{\partial y} - \frac{d}{dx} \frac{\partial L}{\partial y'} + \frac{d^2}{dx^2} \frac{\partial L}{\partial y''} - \dots + (-1)^n \frac{d^n}{dx^n} \frac{\partial L}{\partial y^n}.$$

Des cas encore plus généraux sont aussi possibles : lagrangiens d'ordre supérieur à plusieurs variables..., mais la notation devient trop compliquée et peu pratique. par exemple, pour le lagrangien de second ordre  $L = (\Delta u)^2$ , discuté ci dessus, il vaut mieux calculer la variation directement au lieu d'utiliser une formule générale.

## 3.7 Condition d'Euler-Lagrange pour les exemples antérieurs

### 3.7.1 La longueur d'une courbe paramétrée

$$E[y] = \int_a^b \sqrt{1 + y'(x)^2} dx \quad E'[y] = \frac{y''(x)}{\sqrt{1 + y'(x)^2}}.$$

La condition  $E'[y] = 0$  est équivalente à  $y''(x) = 0$ , c'est à dire, que  $y$  est une droite.

### 3.7.2 La longueur d'une courbe paramétrée

$$E[q] = \int_a^b \|\dot{q}(t)\| dt \quad E'[q] = -\frac{d}{dt} \left( \frac{\dot{q}(t)}{\|\dot{q}(t)\|} \right).$$

La condition  $E'[q] = 0$  dit que la direction du vecteur tangent à la courbe  $\left( \frac{\dot{q}(t)}{\|\dot{q}(t)\|} \right)$  est constante, c'est à dire, la courbe est une droite avec une paramétrisation arbitraire.

### 3.7.3 L'action d'une trajectoire

$$E[q] = \int_a^b \frac{1}{2} \|\dot{q}(t)\|^2 dt \quad E'[q] = -\ddot{q}(t).$$

La condition  $E'[q] = 0$  dit que le vecteur tangent à la courbe est constante (et de longueur constante). C'est à dire, la courbe est une droite paramétrée par un multiple constante du paramètre arc ( une droite parcourue à vitesse)

### 3.7.4 L'action dans un moyen non-homgène

$$E[q] = \int_a^b g(q(t)) \|\dot{q}(t)\|^2 dt \quad E'[q] = g_{q^i} \|q\|^2 - 2(\nabla g \cdot \dot{q}) \dot{q}^i - 2g\ddot{q}^i.$$

La condition  $E'[q] = 0$  est une système semi-linéaire d'EDO de second ordre, (les EDO des géodésique).

### 3.7.5 L'énergie potentielle d'une chaîne suspendue

Voici un problème variationnel avec contraintes. Il faut minimiser l'énergie potentielle  $E[y]$  d'une courbe de longueur  $G[y] = l$ . L'énergie potentielle est

$$E[y] = \int_a^b y(x) \sqrt{1 + y'(x)^2} dx.$$

Pour trouver le minimum sous la contrainte  $G[y] = l$  il faut minimiser la fonctionnelle  $F[y] = E[y] - \lambda G[y]$ , où la valeur du multiplicateur de Lagrange  $\lambda$  sera déterminée à posteriori pour satisfaire la contrainte.

Le Lagrangien est donc  $L = (y - \lambda) \sqrt{1 + y'^2}$ . Vu qu'il ne dépend pas explicitement de  $x$  on peut calculer l'équation d'Euler-Lagrange comme  $L - y \frac{\partial L}{\partial y} = c$ , pour une constante  $c$  :

$$\frac{(y(x) - \lambda)}{\sqrt{1 + y'(x)^2}} = c.$$

La solution de cette équation est une chaînette de la forme  $y(x) = \lambda + \cosh\left(\frac{x}{c} + c_2\right)$ . Les trois constantes  $\lambda$ ,  $c$  et  $c_2$  sont enfin ajustées pour satisfaire les deux conditions de bord et la contrainte.

### 3.7.6 La tension d'une membrane par petites déformations verticales

$$E[u] = \int_{\Omega} \|\nabla u(x)\|^2 dx \quad E'[u] = -\Delta u.$$

La solution est la fonction harmoniques ( $0 = \Delta u$ ) qui satisfait condition de bord L'EDP résultante est l'équation de Laplace.

### 3.7.7 La tension d'une plaque par petites déformations verticales

$$E[u] = \int_{\Omega} |\Delta u(x)|^2 dx \quad E'[u] = \Delta\Delta u.$$

Voici un fonctionnel de second ordre, qui résulte en une EDP de quatrième ordre appelé l'équation biharmonique. Pour avoir solution unique on doit fixer la valeur de  $u$  et sa dérivé première sur  $\partial\Omega$ .

Dans cet exemple, pour calculer  $E'$  il vaut mieux de calculer la variation directement au lieu d'utiliser une formule générale.

### 3.7.8 L'aire d'une surface définie par graphe

Le graphe d'une fonction  $u : \mathbb{R}^2 \rightarrow \mathbb{R}$  définit par une surface  $(x, y, u(x, y))$  en  $\mathbb{R}^3$ . Les surfaces minimales sont celles qui minimisent  $E[u]$  avec une condition de bord fixée (par exemple, un film de savon s'appuyant sur un contour rigide).

$$E[u] = \int_{\Omega} \sqrt{1 + \|\nabla u(x)\|^2} dx \quad E'[u] = -\frac{u_{xx}(1 - u_y^2) - 2u_x u_y u_{xy}(1 + u_x^2)}{\sqrt{1 + u_x^2 + u_y^2}^3}.$$

### 3.7.9 La variation totale d'une fonction

$$E[u] = \int_{\Omega} \|\nabla u(x)\| dx \quad E'[u] = -\frac{1}{\|\nabla u\|^3} \operatorname{div} \left( \frac{\nabla u}{\|\nabla u\|} \right).$$

### 3.7.10 Le $p$ laplacien

$$E[u] = \int_{\Omega} \|\nabla u(x)\|^p dx \quad E'[u] = 0 \Leftrightarrow \operatorname{div} (\|\nabla u\|^{p-2} \nabla u) = 0.$$



### 3.7.11 La pente maximale d'une fonction

La pente maximale d'une fonction  $u : \mathbb{R}^n \rightarrow \mathbb{R}$  est

$$E[u] = \sup_{x \in \Omega} \|\nabla u(x)\|.$$

Cette fonctionnelle n'est pas dérivable avec les méthodes décrites aujourd'hui. Par contre, on peut l'interpréter comme le limite quand  $p \rightarrow \infty$  de l'énergie du  $p$ -laplacien. Ainsi, on obtient l'équation non-linéaire du Laplacien infini :

$$D^2u(\nabla u, \nabla u) = 0.$$

### 3.7.12 Le débruitage par lissage d'une image $I(x)$

$$E[u] = \frac{1}{2} \int_{\Omega} (|u(x) - I(x)|^2 + \alpha^2 \|\nabla u(x)\|^2) dx \quad E'[u] = u - I - \alpha^2 \Delta u.$$

La condition  $E'[u] = 0$  est une EDP linéaire de seconde ordre. Sa solution est une fonction qui est à la fois lisse et approxime bien la fonction  $I$ . Le compromis entre ces deux requis contradictoires est géré par la valeur du paramètre  $\alpha$ .

# Conclusion

Les méthodes du gradients constituent une base importante pour résoudre les problèmes d'optimisation sous contraintes, en particulier si les contraintes sont des systèmes d'équations ou d'inéquations, les conditions de Lagrange et Khun-Tucker sont efficaces pour la suite de la résolution du problème.

Les problèmes variationnels sont aussi basés sur le Lagrangien et peuvent être prolongés par d'autres conditions à savoir Beltrami...etc.

Au jourdhui les algorithmes utilisés pratiquement par les scientifiques pour les problèmes d'optimisation sous contrainte sont basés le gradient conjugué, la plus part des logiciels scientifiques ont codé cet algorithme dans leur langage scientifique ce qui montre le robustesse de ces méthodes.

# Bibliographie

- [1] **Cornelius, Lanczos.** The Variational Principles of Mechanics, 1893.
- [2] **Cristian, Léonard.** Note du Cours de Mathe-2004
- [3] **Jacque, Baranger.** Analyse numérique\_Sous la direction, Collection des Science Hermann éditeur des Science et des Arts, janvier 1991.
- [4] **J,Ch. Gilbert.** Optimisation Différentiable Théorie et Algorithmes\_Résumé du cours\_1décembre 2008.
- [5] **Gregorie,Allaire.**Analyse numérique et optimisation-Une introduction à la modélisation mathématique et la simulation numérique Août 2005.
- [6] **L,C,Evans.**Partial Differential Equations, 1997.
- [7] **Stéphane,Mottelet.**RO04/TI07-Optimisation non-linéaire, Université de Technologie de Compiègne-Printemps-2003.