Optimisation Multidisciplinaire

Jean-Antoine Désidéri

DR INRIA, émérite
Équipe-Projet INRIA Acumes
Centre INRIA Université Côte d'Azur (France)
http ://www-sop.inria.fr/acumes

Présentation générale du cours
12 Janvier 2023

Institut Supérieur de l'Aéronautique et de l'Espace
Modélisation Systèmes Complexes et Simulation
Optimisation Multidisciplinaire et ProblÃ¨mes Inverses

Université Paul Sabatier
Master Recherche & Innovation

Année universitaire 2023 - 2024

Résumé

Ce cours fait suite au cours d'"Introduction à l'optimisation numérique" dans lequel l'étudiant s'est familiarisé

— aux notions fondamentales en optimisation (calcul différentiel, rôle de la convexité, conditions d'optimalité),
— aux algorithmes de résolution avec gradient,
— et à quelques algorithmes d'optimisation sans dérivée.

Il s'adresse donc à des étudiants de niveau "Master" qui ont assimilé ces notions. Alors que le propos était jusqu'ici centré sur la convergence des algorithmes en dimension finie, on s'intéresse désormais aux problèmes dont la formulation, au moins sur une base conceptuelle, fait intervenir une inconnue fonctionnellle. C'est le cas notamment en optimisation de forme, où l'inconnue est une courbe ou une surface.

Les sytèmes complexes en ingénierie sont le plus souvent régis par des équations différentielles ordinaires (EDO) ou des équations aux dérivées partielles (EDP) qui modélisent les disciplines fondamentales : équations de l'élasticité en mécanique du solide, de Navier-Stokes en mécanique des fluides, de Maxwell en électromagnétisme, équation des ondes en acoustique, de la chaleur en thermique, etc. L'analyse d'un tel système passe généralement par simulation numérique qui fournit le champ d'une variable d'état fonctionnelle, dont dépendent par post-traitement les critères de performance, ou fonctions de coût considérées dans le processus d'optimisation. L'objectif premier de ce cours est d'examiner les conséquences algorithmiques de cette dépendance fonctionnelle, en particulier et surtout, pour ce qui est du calcul du gradient de fonctionnelle par une équation adjointe. On aborde cette question en introduisant la difficulté graduellement par le traitement d'exemples types de

— calcul des variations,
— contrôle optimal,
— et systèmes distribués.

Dans la deuxième partie du cours, on se penche sur une problématique très importance en ingénierie des systèmes complexes : l'optimisation multiobjectif et/ou multidisciplinaire. Après avoir formulé la notion classique d'optimalité au sens de Pareto, on cite les stratégies évolutionnaires très fréquemment utilisées pour identifier le front de Pareto et on brosse très brièvement les problématiques industrielles d'"ingénierie concourante". On aborde ensuite deux approches déterministes en optimisation différentiable :

— la construction de méthodes de descente multiobjectifs, et
— l'optimisation priorisée par jeux de Nash.

Tout au long, on donne des exemples en aéronautique, notamment en aérodynamique compressible.

Avant-propos : notion de fonctionnelle

Rappelons la terminologie classique :

- fonction : application de $[a, b]$ dans $\mathbb{R}$
- fonction de n variables : application de $\Omega \subset \mathbb{R}^n$ dans $\mathbb{R}$
- espace fonctionnel : espace vectoriel dont les éléments sont des fonctions, généralement caractérisées par leur régularité ; principaux exemples :
  $\mathcal{C}^n([a,b]) = \{ f : [a,b] \to \mathbb{R}, \text{continue sur } [a,b] \text{ et } n \text{ fois continûment dérivables sur }]a,b[\}$
  $L^p([a,b]) = \left\{ f : [a,b] \to \mathbb{R}, \int_a^b |f(x)|^p \, dx < \infty \right\}$
  $H^1([a,b]) = \{ f : [a,b] \to \mathbb{R}, \ f \in L^2([a,b]), \ f' \in L^2([a,b]) \}$
  $H_0^1(\Omega) = \{ f \in H^1(\Omega), \ f_{/\partial\Omega} = 0 \}$
- **fonctionnelle** : "fonction de fonction", i.e. une application dont l'espace objet est un espace fonctionnel, et dont la valeur est dans $\mathbb{R}$.

Dans cette partie, on s'intéresse aux principales formulations classiques de problèmes d'optimisation dans lesquels on cherche à extrémaliser une fonctionnelle, sans ou sous contraintes. Ces problèmes se classent par complexité croissante comme suit :

1. **Fonctionnelles explicites : calcul des variations**
   Problème type : optimisation d'une forme dont un critère en dépend explicitement
   Quelle courbe extrémalise tel ou tel critère dont le calcul peut être fait explicitement en fonction de l'équation de la courbe (ex. : plus court chemin, aire maximale à périmètre donné, etc).
2. **Fonctionnelles soumises à une dynamique : contrôle optimal**
   Problème type : optimisation de la trajectoire d'un système "contrôlé" ou "commandé"
   La variable indépendante est ici le temps. L'état d'un système (- il s'agit souvent d'un système mécanique : avion, fusée, etc -) est décrit par un vecteur d'état $\mathbf{x}(t) \in \mathbb{R}^n$ solution d'une équation différentielle ordinaire (EDO) dans laquelle intervient une fonction de commande (ou "contrôle"), telle que, pour une fusée, la manette des gaz, ou l'orientation de la poussée ; le plus souvent cette dynamique est l'expression de la loi fondamentale de la dynamique. Il s'agit alors d'optimiser le contrôle pour extrémaliser une fonctionnelle de la trajectoire complète $\mathbf{x}(t)$ $(0 \leq t \leq T)$, avec éventuellement une pénalisation sur $\mathbf{u}(t)$.
3. **Optimisation de systèmes distribués**
   On appelle ici "système distribué", un système régi dans un certain domaine d'espace Ω par une équation aux dérivées partielles (EDP). Le contrôle peut alors s'exercer sur une portion de la frontière $\Gamma = \partial\Omega$, par une condition aux limites particulière imposée à l'EDP, ou être lui-même distribué (dans le domaine) et apparaître par un ou plusieurs termes dans l'EDP. Nous étudierons des exemples de ces deux situations.
   Exemple : conception optimale de forme en aérodynamique.

En pratique, les problèmes d'optimisation fonctionnelle, lorsqu'ils sont traités par voie numérique, se ramènent au plan algorithmique à des problèmes d'optimisation paramétrique, dans $\mathbb{R}^N$, avec, en principe, N grand. Cependant le caractère fonctionnel de la formulation conceptuelle a d'importantes conséquences sur la nature de la solution optimale et le calcul pratique du gradient.

Chapitre 1

Calcul des variations

En calcul des variations, on minimise une "fonctionnelle explicite", ayant souvent la forme d'une intégrale dont l'intégrande est une expression formelle de la variable indépendante x ainsi que de la fonction inconnue $y(x)$ et de ses dérivées.

1.1 Problème type, exemples

Trouver la fonction $y:\ [a,b] \to \mathbb{R}$ qui résout le problème de minimisation suivant :

$$\boxed{\min_y J(y) = \int_a^b F\Big(x; y(x), y'(x), ..., y^{(n)}(x)\Big)\, dx} \tag{1.1}$$

dans lequel la fonction $y(x)$ admet la régularité suivante :

$$\boxed{y \in \mathcal{C}^{2n}([a,b])} \tag{1.2}$$

et satisfait les conditions de Dirichlet suivantes :

$$\boxed{i = 0, 1, ..., n-1 : \quad y^{(i)}(a) = \alpha_i\ ,\ y^{(i)}(b) = \beta_i} \tag{1.3}$$

et F est une fonction régulière de tous ses arguments.

L'exemple le plus simple d'un tel problème correspond à la recherche du plus court chemin entre les points $A = (a, \alpha_0)$ et $B = (b, \beta_0)$. En supposant que l'arc $\overset{\frown}{AB}$ cherché est un graphe qui admet l'équation analytique $y = y(x)$, il vient :

$$J(y) = \int_a^b \sqrt{1 + y'(x)^2}\, dx \tag{1.4}$$

qui est bien de la forme posée, avec ici, $n = 1$ et $F = (1+y'^2)^{\frac{1}{2}}$ qui ne dépend pas explicitement de x ni de $y(x)$, mais seulement de $y'(x)$. Nous allons retrouver que la solution optimale correspond bien au segment rectiligne AB, pour lequel $y(x) = \alpha_0 + [(\beta_0 - \alpha_0)/(b-a)](x-a)$, et $y''(x) = 0$.

La fondation de Carthage par Didon

Par Didier Müller, mardi 2 septembre 2008 à 08:04 - Il y a des maths là ? - #1068 - rss

Selon la légende, Carthage a été fondée au IXe siècle av. J.C. (en -814) par Didon, princesse de Tyr en Phénicie, soeur de Pygmalion. Celui-ci a assassiné l'époux de Didon, Sichée, pour prendre le pouvoir. Elle s'enfuit, entourée de Phéniciens, et accoste en Afrique du Nord, à l'emplacement de Carthage. Didon demande au roi de Numibie, Iarbas, une terre pour s'y établir. Iarbas, réticent, consent à ce qu'elle ne prenne que la grandeur de terre délimitée par une peau de boeuf. Didon fait alors découper la peau de boeuf en lanières très fines.

Dido Purchases Land for the Foundation of Carthage. Engraving by Matthäus Merian the Elder, in *Historische Chronica*, Frankfurt a.M., 1630. Dido's people cut the hide of an ox into thin strips and try to enclose a maximal domain.

On a montré que la plus grande surface limitée par une longueur fixée est un disque. La démonstration, dite de la propriété isopérimétrique du cercle, est due à Zénodore (Grèce, seconde moitié du IIe siècle avant J.-C. ; ceci est cité par Théon dans l'Almageste de Ptolémée et par Pappus), et sera complétée par Weierstrass à la fin du XIXe siècle. En prenant astucieusement un terrain en forme de demi-disque au bord de la mer (ce bord étant supposé rectiligne), Didon multiplia encore par $\frac{\pi}{2}$ la surface acquise.

FIGURE 1.1 – Illustration du problème de Zénodore (origine : *Le blog-notes mathématique du coyote*)

Un autre exemple est fourni par le fameux **problème de Zénodore**[1]. Comme nous l'apprend Wikipedia : "Zénodore a étudié la détermination de la surface d'une figure géométrique et celle du volume d'un objet avec une surface donnée [... et a démontré que :] Un polygone régulier a la plus grande surface parmi les polygones de même périmètre et de même nombre de côtés." Par passage à la limite, on voit que la solution est un arc de cercle.

Dans le cas plan, le problème revient à chercher la fonction $y(x)$ qui satisfait des conditions de Dirichlet comme précédemment, disons $y(a) = y(b) = 0$, et maximise la fonctionelle

$$\mathcal{A}(y) = \int_a^b y(x)\, dx \tag{1.5}$$

sous la contrainte

$$P(y) = \int_a^b \sqrt{1 + y'(x)^2}\, dx = C \tag{1.6}$$

On sait que dans un problème d'optimisation sous contrainte, on extrémalise le lagrangien (voir rappels en Annexe A). Ici, on doit minimiser par rapport à y (à λ donné)

$$\mathcal{L}(y, \lambda) = -\mathcal{A}(y) + \lambda[P(y) - C] = \int_a^b F_\lambda(y, y')\, dx - \lambda C \tag{1.7}$$

où $F_\lambda(y, y') = -y(x) + \lambda\sqrt{1 + y'(x)^2}$ et ajuster λ pour satisfaire la contrainte. Dans la minimisation du lagrangien $\mathcal{L}$ par rapport à y, le terme λC ne joue aucun rôle. On est donc bien ramené à un problème du même type, où ici, l'intégrande F_λ ne dépend pas explicitement de x.

1.2 Première variation, gradient

On revient au problème général défini par les équations (1.1)-(1.2)-(1.3). La fonction $y(x)$ appartient à un espace affine E sous-tendu par l'espace vectoriel E' de ses perturbations admissibles :

$$\begin{aligned} &y \in E,\ y + \delta y \in E, \\ &\delta y \in E' := \left\{ \delta y \in \mathcal{C}^{2n}([a,b])\ /\ i = 0, 1, ..., n-1 :\ \delta y^{(i)}(a) = \delta y^{(i)}(b) = 0 \right\}. \end{aligned} \tag{1.8}$$

Il est important de comprendre que **l'opérateur de variation-première,** δ, opérant sur $y(x)$ ou sa dérivée i-ème, **commute avec l'opérateur de dérivation** par rapport à x, ou ses puissances. Autrement dit, si l'on donne une variation fonctionnelle $\delta y(x)$ à la fonction $y(x)$, la variation induite sur sa dérivée i-ème, à savoir la fonction $\delta\left[y^{(i)}(x)\right]$, sera précisément égale à la dérivée i-ème de la perturbation $\delta y(x)$:

$$\delta\left[y^{(i)}(x)\right] = \frac{d^i}{dx^i}\delta y(x) \tag{1.9}$$

Dès lors, omettant la variable x pour alléger la notation, cette fonction sera notée sans ambigüité : $\delta y^{(i)}$, puisqu'il est indifférent que la dérivée i-ème s'applique à y préalablement à δ, ou globalement à δy.

1. Grec, seconde moitié du IIe siècle av. J.-C., voir : `https://fr.wikipedia.org/wiki/Zenodore_(mathematicien)`

Cette remarque importante étant faite, on suppose connue la notion générale de "différentielle" d'une fonction régulière. Ici, on s'intéresse à la forme linéaire d'argument $\delta y(x)$, d'abord notée $\delta J(y;\delta y)$, pour laquelle, la fonction $y(x)$ étant fixée, on a :

$$J(y+\delta y) - J(y) = \delta J(y;\delta y) + o(\|\delta y\|) \tag{1.10}$$

où le dernier terme est un "petit o". La valeur de la différentielle, $\delta J(y;\delta y)$, est également appelée **variation-première de la fonctionnelle** $J(y)$ induite par la perturbation $\delta y(x)$ de la fonction $y(x)$, et sera notée simplement δJ.

La variation première de la fonctionnelle $J(y)$ s'obtient en dérivant l'intégrande par rapport à ses arguments fonctionnels :

$$\delta J = \int_a^b \sum_{i=0}^{n} \frac{\partial F}{\partial y^{(i)}}\, \delta y^{(i)}\, dx = \sum_{i=0}^{n} \delta_i J \tag{1.11}$$

où l'on a introduit la notation :

$$\delta_i J = \int_a^b \frac{\partial F}{\partial y^{(i)}}\, \delta y^{(i)}\, dx \tag{1.12}$$

Pour $i \geq 1$, on a donc :

$$\delta_i J = \int_a^b \frac{\partial F}{\partial y^{(i)}} \frac{d}{dx}\left(\delta y^{(i-1)}\right)\, dx \tag{1.13}$$

et une intégration par parties donne :

$$\delta_i J = \left[\frac{\partial F}{\partial y^{(i)}}\, \delta y^{(i-1)}\right]_a^b - \int_a^b \frac{d}{dx}\left(\frac{\partial F}{\partial y^{(i)}}\right)\, \delta y^{(i-1)}\, dx \tag{1.14}$$

Or, le terme intégré, $[...]_a^b$ est nul en vertu des conditions aux limites satisfaites par la perturbation δy. Il vient donc :

$$\delta_i J = -\int_a^b \frac{d}{dx}\left(\frac{\partial F}{\partial y^{(i)}}\right)\, \delta y^{(i-1)}\, dx \tag{1.15}$$

En comparant (1.15) à (1.12), il est facile de voir que i intégrations par parties conduisent à l'expression :

$$\delta_i J = (-1)^i \int_a^b \frac{d^i}{dx^i}\left(\frac{\partial F}{\partial y^{(i)}}\right)\, \delta y(x)\, dx \tag{1.16}$$

qui s'applique également au cas $i = 0$.

En regroupant ces résultats, il vient

$$\boxed{\delta J = \int_a^b G\, \delta y(x)\, dx\,, \quad G := \sum_{i=0}^{n} (-1)^i\, \frac{d^i}{dx^i}\left(\frac{\partial F}{\partial y^{(i)}}\right)} \tag{1.17}$$

Cette équation fournit le **gradient de fonctionnelle** puisqu'elle définit explicitement la forme linéaire qui exprime la première variation de la fonctionnelle en fonction de la variation δy de son argument.

1.3 Stationnarité, équation d'Euler-Lagrange

Une condition nécessaire pour que la fonctionnelle $J(y)$ atteigne un extremum est que son gradient s'annule. Cette condition fournit l'EDO d'ordre $2n$ suivante :

$$G := \sum_{i=0}^{n} (-1)^i \, \frac{d^i}{dx^i} \left(\frac{\partial F}{\partial y^{(i)}} \right) = \frac{\partial F}{\partial y} - \frac{d}{dx} \left(\frac{\partial F}{\partial y'} \right) + ... + (-1)^n \, \frac{d^n}{dx^n} \left(\frac{\partial F}{\partial y^{(n)}} \right) = 0 \qquad (1.18)$$

Cette EDO est soumise aux $2n$ conditions aux limites de Dirichlet, (1.3). On obtient donc, en principe, un problème fermé admettant une solution unique. Cette question devra être examinée dans le cas particulier étudié.

Dans le cas particulier de $n = 1$, l'EDO prend la forme classique suivante :

$$\boxed{\frac{\partial F}{\partial y} - \frac{d}{dx} \left(\frac{\partial F}{\partial y'} \right) = 0} \qquad (1.19)$$

et porte le nom d'**équation d'Euler-Lagrange**.

En mécanique rationnelle, dans le formalisme de Lagrange, certains systèmes mécaniques complexes sont représentés par un ensemble de coordonnées généralisées y. Lorsque le système est conservatif, l'équation d'Euler-Lagrange dans laquelle F représente l'énergie totale (cinétique+potentielle) et x le temps, donne l'équation différentielle du mouvement.

Exercice 1
Résoudre le problème de Zénodore avec l'équation d'Euler-Lagrange.

Exercice 2
On considère la fonctionnelle suivante :

$$J(y) = \frac{P^2}{\mathcal{A}} \qquad (1.20)$$

où $y(x)$ satisfait les conditions de Dirichlet homogènes :

$$y(0) = y(1) = 0 \qquad (1.21)$$

et

$$\mathcal{A} = \int_0^1 y(x)\, dx\,, \quad P = \int_0^1 \sqrt{1 + y'(x)^2}\, dx \qquad (1.22)$$

a) Montrer que dans une classe d'équivalence de fonctions proportionnelles entre elles, il existe un minimum de $J(y)$.
b) Montrer que l'arc de cercle correspond à un minimum de la fonctionnelle.

1.4 Intégrale première d'Euler

On considère maintenant le cas particulier important où **la fonction F ne dépend pas explicitement de x** :

$$F = F\Big(\cancel{x}; y(x), y'(x), ..., y^{(n)}(x)\Big) \qquad (1.23)$$

Cependant, pour un choix donné de la fonction $y(x)$, F est une fonction de x, par l'intermédiaire de $y(x)$ et de ses dérivées. On note alors :

$$F = F\Big(y(x), y'(x), ..., y^{(n)}(x)\Big) = \mathsf{F}(x) \tag{1.24}$$

dont la dérivée ("dérivée totale de F par rapport à x) se simplifie :

$$\mathsf{F}'(x) = \frac{d\mathsf{F}}{dx} = \cancel{\frac{\partial F}{\partial x}} + \sum_{i=0}^{n} \frac{\partial F}{\partial y^{(i)}}\, y^{(i+1)}(x) \tag{1.25}$$

En intégrant cette équation de a à x, il vient :

$$\mathsf{F}(x) - \mathsf{F}(a) = F - \text{const.} = \sum_{i=0}^{n} \phi_i(x)\,; \quad \phi_i(x) := \int_a^x \frac{\partial F}{\partial y^{(i)}}\, y^{(i+1)}(x)\, dx \tag{1.26}$$

Or, pour $i \geq 1$, une intégration par parties donne :

$$\begin{aligned} \phi_i(x) &= \int_a^x \frac{\partial F}{\partial y^{(i)}}\, dy^{(i)}(x) = \left[\frac{\partial F}{\partial y^{(i)}}\, y^{(i)}(x)\right]_a^x - \int_a^x \frac{d}{dx}\Big(\frac{\partial F}{\partial y^{(i)}}\Big)\, y^{(i)}(x)\, dx \\ &= \frac{\partial F}{\partial y^{(i)}}\, y^{(i)}(x) - \int_a^x \frac{d}{dx}\Big(\frac{\partial F}{\partial y^{(i)}}\Big)\, y^{(i)}(x)\, dx + \text{const.} \end{aligned} \tag{1.27}$$

de sorte que i intégrations par parties donnent :

$$\begin{aligned} \phi_i(x) = {} & \frac{\partial F}{\partial y^{(i)}}\, y^{(i)}(x) - \frac{d}{dx}\Big(\frac{\partial F}{\partial y^{(i)}}\Big)\, y^{(i-1)}(x) + ... + (-1)^{i-1} \frac{d^{i-1}}{dx^{i-1}}\Big(\frac{\partial F}{\partial y^{(i)}}\Big)\, y'(x) \\ & + (-1)^i \int_a^x \frac{d^i}{dx^i}\Big(\frac{\partial F}{\partial y^{(i)}}\Big)\, y'(x)\, dx + \text{const.} \end{aligned} \tag{1.28}$$

Supposons que $y(x)$ satisfasse l'équation d'Euler-Lagrange. Alors, la somme de $\phi_0(x)$ et des intégrales $\int_a^x [...]\, dx$ qui apparaissent dans $\sum_{i=1}^n \phi_i(x)$ est nulle. Restent les termes intégrés qui donnent :

$$F - \sum_{i=1}^{n} \left[\frac{\partial F}{\partial y^{(i)}}\, y^{(i)}(x) - \frac{d}{dx}\Big(\frac{\partial F}{\partial y^{(i)}}\Big)\, y^{(i-1)}(x) + ... + (-1)^{i-1} \frac{d^{i-1}}{dx^{i-1}}\Big(\frac{\partial F}{\partial y^{(i)}}\Big)\, y'(x)\right] = \text{const.} \tag{1.29}$$

Cette EDO d'ordre $2n-1$ constitue une intégrale première de l'équation d'Euler-Lagrange. Dans le cas où $n = 1$, elle prend la forme classique :

$$\boxed{F - y' \frac{\partial F}{\partial y'} = \text{const.}} \tag{1.30}$$

et porte le nom d'**intégrale première d'Euler**.

Exercice 3
Résoudre à nouveau le problème de Zénodore en utilisant l'intégrale première d'Euler.

Exercice 4
Soit un point matériel de masse m accroché à un ressort de raideur k. On repère la position verticale du point par rapport à sa position d'équilibre que l'on note $y(x)$ où $x = t$ est le temps. On note $F = T - U$, où T est l'energie cinétique du point matériel et U son énergie potentielle. A quelles lois de la mécanique correspondent l'équation d'Euler-Lagrange et l'intégrale première d'Euler ?

1.5 Le fameux problème du brachistochrone

Citons Wikipedia (`http://en.wikipedia.org/wiki/Brachistochrone`) :

"Le mot brachistochrone désigne une courbe dans un plan vertical sur laquelle un point matériel pesant placé dans un champ de pesanteur uniforme, glissant sans frottement et sans vitesse initiale, présente un temps de parcours minimal parmi toutes les courbes joignant deux points fixés : on parle de problème de la courbe brachistochrone.

[...]

*La résolution du problème de la courbe brachistochrone passionna les mathématiciens. Il apparaît en 1633 chez Galilée qui crut que la solution consistait en un arc de cercle. Jean Bernoulli pose clairement le problème en 1696 dans les Acta Eruditorum, et des solutions furent apportées par lui-même ainsi que par son frère Jacques Bernoulli, Newton, Leibniz, L'Hôpital et Tschirnhaus : il s'agit d'***un arc de cycloïde** *commençant avec une tangente verticale.*

Les méthodes imaginées pour sa résolution amenèrent à développer la branche des mathématiques qu'on appelle le calcul des variations.

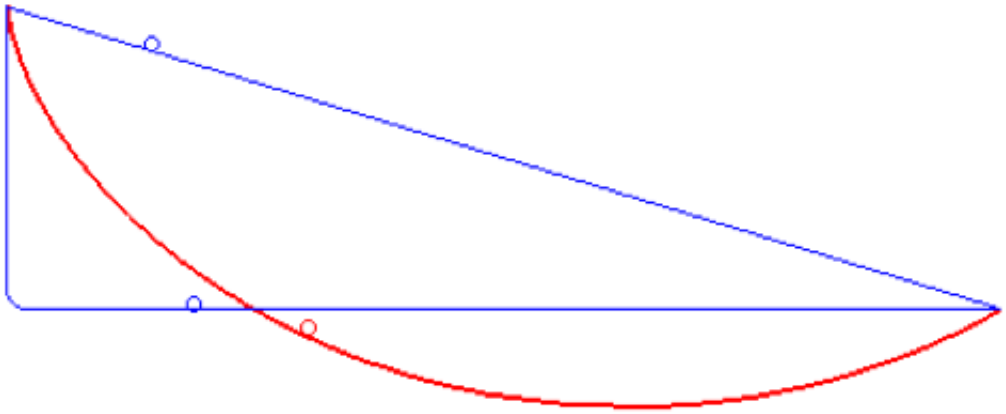


FIGURE 1.2 – Illustration de la courbe brachistochrone : les 3 billes ont été lâchées sans vitesse du coin supérieur gauche au même instant ; à l'instant ultérieur de la figure, la bille qui glisse sur la cycloïde a pris de l'avance sur les deux autres et arrivera la première au coin droit.

L'étudiant est invité à consulter le site internet cité pour la démonstration d'origine de Bernoulli et celle par le calcul des variations, ainsi que d'amusantes animations du résultat.

1.6 Deux exercices tirés de Bryson and Ho

Exercice 5 (Problème 3 p. 64 du Bryson and Ho)

Problem 3. *Minimum surface of revolution connecting two coaxial circular loops.* Given two coaxial circular loops, of radius a which are a distance 2ℓ apart, *find* the surface of revolution containing the two loops with minimum area. (This is the shape a soap film would take if stretched between two rings.) [HINT: Choose cylindrical coordinates r,x as shown in Figure 2.4.6. The annular element of area is

$$dA = 2\pi r\sqrt{(dr)^2 + (dx)^2}\,,$$

so the problem is to find $u(x)$ to minimize the integral.

$$A = 2\pi \int_{-\ell}^{\ell} r\sqrt{1 + u^2}\, dx\,,$$

where

$$\frac{dr}{dx} = u \qquad \text{and} \qquad r(\ell) = a\,, r(-\ell) = a\,.$$

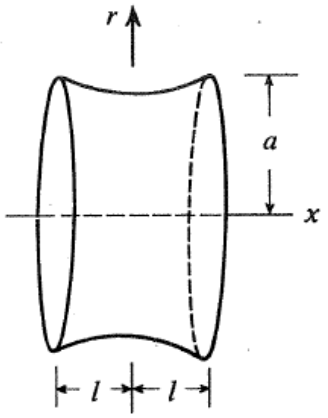


Figure 2.4.6. Minimum-area surface of revolution connecting two coaxial circular loops.

Exercice 6 (Problème 4 p. 65 du Bryson and Ho)

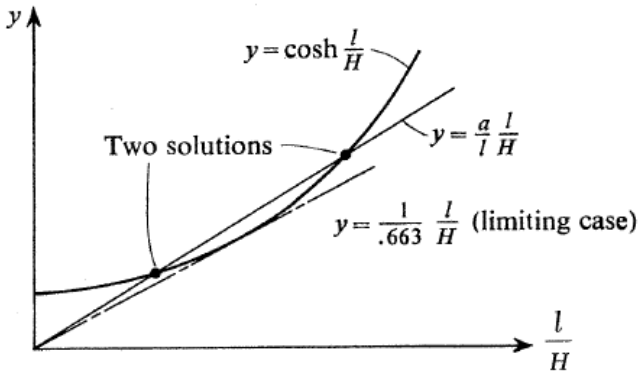


Figure 2.4.7. Solution of the minimum surface of revolution problem.

Problem 4. Find the minimum surface of revolution connecting two coaxial circular loops a distance ℓ apart, where one loop has radius a and the other loop has radius $b < a$. For each given value of b/a, show that a limiting value of $\ell/a < (\ell/a)_{\text{lim}}$ exists beyond which the minimum surface is $r = 0$; that is, two flat discs within the circular loops.

Chapitre 2

Contrôle optimal

En contrôle optimal, le temps t est souvent la variable indépendante. On minimise une "fonctionnelle implicite", ayant souvent la forme d'une intégrale dont l'intégrande dépend du temps, de la fonction de contrôle, inconnue du problème, et de la trajectoire $\{\mathbf{x}(t)\}$ $(t \geq 0)$ d'un système soumis à une équation différentielle ordinaire où intervient le contrôle.

2.1 Problème type (en temps fixé), exemples

On considère un système évolutif dont l'état est décrit par la variable $\mathbf{x}(t)$ à valeurs dans $\mathbb{R}^n$. Ce système évolue à partir d'un état initial $\mathbf{x}_0$, ici entièrement fixé, selon une "dynamique contrôlée" :

$$\boxed{\begin{cases} \dot{\mathbf{x}}(t) = f\Big(t; \mathbf{x}(t), \mathbf{u}(t)\Big) \\ \mathbf{x}(0) = \mathbf{x}_0 \end{cases}} \tag{2.1}$$

où t est le temps et $\mathbf{u}(t)$ est la "fonction de contrôle", ou "commande", à valeurs dans $\mathbb{R}^p$. Aucune hypothèse n'est faite sur les dimensions n et p. La fonction $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^n$ est supposée régulière par rapport à ses trois arguments.

On s'intéresse ici à la "**trajectoire**" $\{\mathbf{x}(t)\}$ $(0 \leq t \leq T)$ du système avec l'hypothèse d'un temps final T fixé.

De très nombreux exemples de tels systèmes sont fournis par la dynamique des véhicules de transport (voiture, avion, fusée, etc), mais pas uniquement.

Exercice 7
Donner des exemples d'origines différentes (biologie, dynamique des populations, etc).

A chaque choix fait du contrôle $\mathbf{u}(t)$ correspond une trajectoire, $\{\mathbf{x}(t)\}$ $(0 \leq t \leq T)$, dont on peut mesurer la performance par une fonctionelle-coût :

$$\boxed{J(\mathbf{u}) = \Phi\Big(\mathbf{x}(T)\Big) + \int_0^T \mathcal{L}\Big(t; \mathbf{x}(t), \mathbf{u}(t)\Big)\, dt} \tag{2.2}$$

Commentons cette expression. Le terme en Φ permet d'inclure dans la formulation des problèmes de tir dans lesquels on cherche à atteindre en temps T un certain état final, en pénalisant l'écart à cette cible. L'intégrale traduit le coût accumulé tout au long du trajet. La fonction $\mathcal{L}$ tire sa notation de l'anglais "loss" (coût) ou du terme "lagrangien"... La dépendance de cette fonction de la variable d'état permet de traduire le souhait que la trajectoire s'écarte peu d'une trajectoire de référence, ou passe à proximité de points donnés. Par exemple, pour remplir sa mission, un satellite d'observation doit couvrir une zone donnée, et peut-être aussi en éviter une autre. La dépendance de $\mathcal{L}$ de la variable de contrôle $\mathbf{u}(t)$ marque généralement la pénalisation du coût effectif pour réaliser l'objectif ; typiquement, pour un véhicule, la dépense énergétique (carburant) pour réaliser la trajectoire.

Pour simplifier, on considère ici, à titre générique, un problème dans lequel aucune contrainte de bornes n'est imposée à la variable d'état ou à la variable de contrôle. Les formulations soumises à de telles contraintes se traitent peu différemment.

Notons enfin que dans les cours spécialisés, on traite de nombreuses variantes, dans lesquelles, par exemple, la variable d'état, $\mathbf{x}(t)$, est seulement partiellement spécifiée en $t = 0$, mais aussi en $t = T$. Nous nous limitons à considérer une dynamique formulée comme un problème de Cauchy pur.

Le problème d'optimisation consiste à trouver le contrôle $\mathbf{u}(t)$ qui minimise la fonctionnelle $J(\mathbf{u})$, (2.2), sous la contrainte dynamique "EDO", (2.1).

2.2 Première variation, gradient

Soit $\{\mathbf{x}(t)\}$ $(0 \leq t \leq T)$ la trajectoire réalisée par le contrôle $\mathbf{u}(t)$, donnant la valeur $J(\mathbf{u})$ à la fonctionnelle-coût.

Soit $\{\mathbf{x}(t) + \delta\mathbf{x}(t)\}$ $(0 \leq t \leq T)$ la trajectoire voisine réalisée par le contrôle $\mathbf{u}(t) + \delta\mathbf{u}(t)$, donnant la valeur $J(\mathbf{u} + \delta\mathbf{u})$ à la fonctionnelle-coût.

La **première variation de la fonctionnelle-coût**, δJ, est la partie de l'écart effectif, $J(\mathbf{u} + \delta\mathbf{u}) - J(\mathbf{u})$ qui dépend linéairement de la variation $\delta\mathbf{u}(t)$. On l'obtient en effectuant la somme de différentielles partielles associées aux variations $\delta\mathbf{x}(t)$ et $\delta\mathbf{u}(t)$:

$$\delta J = \underbrace{\Phi'\Big(\mathbf{x}(T)\Big)\,\delta\mathbf{x}(T) + \int_0^T \Big(\mathcal{L}_{\mathbf{x}}\delta\mathbf{x}(t)}_{\text{implicite}} + \underbrace{\mathcal{L}_{\mathbf{u}}\delta\mathbf{u}(t)}_{\text{expl.}}\Big)\,dt \tag{2.3}$$

En notation matricielle :

- # $\Phi'\Big(\mathbf{x}(T)\Big)$ est un vecteur ligne $1 \times n$, dont les composantes sont les dérivées partielles $\partial\Phi/\partial x_i(T)$ $(1 \leq i \leq n)$;
- # $\mathcal{L}_{\mathbf{x}} = \mathcal{L}_{\mathbf{x}}\big(t; \mathbf{x}(t), \mathbf{u}(t)\big)$ est un vecteur-ligne $1 \times n$ dont les composantes sont les dérivées partielles formelles $\partial\mathcal{L}/\partial x_i(t)$ $(1 \leq i \leq n)$;
- # $\mathcal{L}_{\mathbf{u}} = \mathcal{L}_{\mathbf{u}}\big(t; \mathbf{x}(t), \mathbf{u}(t)\big)$ est un vecteur-ligne $1 \times p$ dont les composantes sont les dérivées partielles formelles $\partial\mathcal{L}/\partial u_j(t)$ $(1 \leq j \leq p)$.

Ici, la variation fonctionnelle $\delta\mathbf{x}(t)$ dépend implicitement de la variation du contrôle, $\delta\mathbf{u}(t)$. Cette dépendance linéaire se traduit par la linéarisation du système de Cauchy, (2.1) :

$$\begin{cases} \delta\dot{\mathbf{x}}(t) = f_{\mathbf{x}}\delta\mathbf{x}(t) + f_{\mathbf{u}}\delta\mathbf{u}(t) \\ \delta\mathbf{x}(0) = 0 \end{cases} \tag{2.4}$$

A nouveau, on remarque que les opérateurs de variation première et de dérivation par rapport à t commutent :

$$\delta\left[\dot{\mathbf{x}}(t)\right] = \frac{d}{dt}\delta\mathbf{x}(t) \tag{2.5}$$

Cette remarque justifie que ces deux quantités soient notées simplement $\delta\dot{\mathbf{x}}(t)$ sans risque d'ambigüité. Noter aussi qu'à la condition initiale de Dirichlet sur $\mathbf{x}(0)$ correspond la condition initiale de Dirichlet homogène sur $\delta\mathbf{x}(0)$.

Au vu de (2.3), la première variation de fonctionnelle-coût, $\delta J(\mathbf{u})$, contient donc un terme intégral qui dépend explicitement de la variation de contrôle, $\delta\mathbf{u}(t)$. Les autres termes font intervenir la variation de la variable d'état, $\delta\mathbf{x}(t)$, ce qui correspond à une dépendance implicite sur $\delta\mathbf{u}(t)$.

On introduit une variable duale de l'état, $\lambda(t)$, dite état-adjoint (en anglais, *costate variable*), à valeurs dans le dual de l'espace de l'état, ici $\mathbb{R}^n$ pour l'état et le dual. On fait le produit scalaire de $\lambda(t)$ avec le système linéarisé (2.4) : pour cela, on convient de **noter la transposition des vecteurs et des matrices par l'indice supérieur** ^{t} (sans lien avec la variable temps, t) ; on prémultiplie (2.4) par $\lambda(t)^t$ et on intègre sur l'intervalle $[0, T]$:

$$\int_0^T \lambda(t)^t\,\delta\dot{\mathbf{x}}(t)\,dt = \int_0^T \lambda(t)^t\Big(f_{\mathbf{x}}\delta\mathbf{x}(t) + f_{\mathbf{u}}\delta\mathbf{u}(t)\Big)\,dt \tag{2.6}$$

On transforme le terme implicite de dérivée temporelle par intégration par parties :

$$\begin{aligned}\int_0^T \lambda(t)^t\,\delta\dot{\mathbf{x}}(t)\,dt &= \left[\lambda(t)^t\,\delta\mathbf{x}(t)\right]_0^T - \int_0^T \dot{\lambda}(t)^t\,\delta\mathbf{x}(t)\,dt \\ &= \lambda(T)^t\,\delta\mathbf{x}(T) - \int_0^T \dot{\lambda}(t)^t\,\delta\mathbf{x}(t)\,dt\end{aligned} \tag{2.7}$$

où la condition initiale $\delta\mathbf{x}(0) = 0$ a été utilisée. Il vient donc, quel que ce soit le choix à venir de $\lambda(t)$:

$$\lambda(T)^t\,\delta\mathbf{x}(T) - \int_0^T \dot{\lambda}(t)^t\,\delta\mathbf{x}(t)\,dt - \int_0^T \lambda(t)^t\Big(f_{\mathbf{x}}\delta\mathbf{x}(t) + f_{\mathbf{u}}\delta\mathbf{u}(t)\Big)\,dt = 0 \tag{2.8}$$

Enfin, on pose des conditions sur l'état-djoint pour imposer que les termes implicites de la première variation de fonctionnelle-coût qui apparaissent dans (2.3) soient identiques à ceux de l'identité (2.8), à savoir :

$$\begin{cases} -\dot{\lambda}(t)^t - \lambda(t)^t\,f_{\mathbf{x}} = \mathcal{L}_{\mathbf{x}} \\ \lambda(T)^t = \Phi'\Big(\mathbf{x}(T)\Big) \end{cases} \tag{2.9}$$

Ces équations constituent un système linéaire, mais à coefficients variables, adjoint au système de Cauchy de la dynamique. Ce **système adjoint est soumis à une condition finale** sur l'état-adjoint et s'intègre de manière **rétrograde en temps**.

En retranchant (2.8) à (2.3), seuls restent les termes explicites :

$$\delta J = \int_0^T \Big(\mathcal{L}_{\mathbf{u}} + \lambda(t)^t\,f_{\mathbf{u}}\Big)\,\delta\mathbf{u}(t)\,dt \tag{2.10}$$

Ces équations peuvent s'écrire de manière plus compacte en introduisant le hamiltonien :

$$\boxed{H = \mathcal{L} + \lambda^t\,f := H\Big(t; \mathbf{x}(t), \lambda(t), \mathbf{u}(t)\Big)} \tag{2.11}$$

En rassemblant en un seul système la dynamique et le système adjoint, on obtient un **problème aux 2 limites** :

$$\boxed{\begin{aligned} &\mathbf{x}(0) = \mathbf{x}_0\,; \quad \dot{\mathbf{x}}(t) = f\Big(t; \mathbf{x}(t), \mathbf{u}(t)\Big) = H_\lambda^t \\ &\qquad\qquad \dot{\lambda}(t) = -\left(\mathcal{L}_{\mathbf{x}}^t + f_{\mathbf{x}}^t\,\lambda\right) = -H_{\mathbf{x}}^t\,; \quad \lambda(T) = \Phi'\Big(\mathbf{x}(T)\Big)^t \end{aligned}} \tag{2.12}$$

La fonction de contrôle $\mathbf{u}(t)$ étant donnée, ce système constitue un jeu de $2n$ EDO pour les $2n$ composantes inconnues de $\mathbf{x}(t)$ et $\lambda(t)$. En général il admet une solution, et le **gradient de fonctionnelle-coût** qui lui est associé est donné par :

$$\boxed{\delta J = \int_0^T H_{\mathbf{u}}\;\delta\mathbf{u}(t)\,dt\,, \quad H_{\mathbf{u}} = \mathcal{L}_{\mathbf{u}} + \lambda(t)^t\,f_{\mathbf{u}}} \tag{2.13}$$

2.3 Condition de stationnarité, optimalité

Une condition nécessaire (et souvent suffisante) pour que la fonctionnelle-coût soit extrémale, est qu'elle soit **stationnaire**, ce qui donne la **condition nécessaire d'optimalité** suivante :

$$\boxed{H_{\mathbf{u}} = 0} \tag{2.14}$$

La condition d'optimalité constitue un jeu de p relations non-différentielles liant les inconnues $\mathbf{x}(t)$ et $\lambda(t)$ aux p variables de contrôle, composantes du vecteur $\mathbf{u}(t)$.

Au vu des dimensions, pour les problèmes bien posés simples, on conçoit que l'on peut résoudre la condition d'optimalité par rapport à $\mathbf{u}(t)$, pour l'exprimer en fonction de $\mathbf{x}(t)$ et $\lambda(t)$. En reportant l'expression correspondante dans le système aux deux limites, on obtient un système d'EDO d'ordre $2n$, clos, qui définit la solution optimale.

2.4 Cas particulier important des sytèmes autonomes

Lorsque la fonction-coût, $\mathcal{L}$, et la dynamique, par la fonction f, ne dépendent pas explicitement du temps, on dit que **le système est autonome** :

$$\mathcal{L} = \mathcal{L}\Big(\not{t}; \mathbf{x}(t), \mathbf{u}(t)\Big)\,, \quad f = f\Big(\not{t}; \mathbf{x}(t), \mathbf{u}(t)\Big) \tag{2.15}$$

Dans ce cas, le hamiltonien lui-même ne dépend pas explicitement du temps :

$$H = \mathcal{L} + \lambda^t f = H\Big(\not{t}; \mathbf{x}(t), \lambda(t), \mathbf{u}(t)\Big) \Longrightarrow H_t = \frac{\partial H}{\partial t} = 0 \tag{2.16}$$

Au cours du temps, si la trajectoire est optimale, et si $\mathsf{H}(t) = H\big(\mathbf{x}^\star(t), \lambda^\star(t), \mathbf{u}^\star(t)\big)$:

$$\begin{aligned} \dot{\mathsf{H}}(t) &= H_t + H_{\mathbf{x}}\,\dot{\mathbf{x}}(t) + H_\lambda\,\dot{\lambda}(t) + H_{\mathbf{u}}\,\dot{\mathbf{u}}(t) \\ &= 0 + H_{\mathbf{x}}\,H_\lambda^t - H_\lambda\,H_{\mathbf{x}}^t + 0 \\ &= 0 \end{aligned} \tag{2.17}$$

Par conséquent :

$$\boxed{\mathsf{H}(t) = \text{const.}} \tag{2.18}$$

Le hamiltonien d'un système autonome contrôlé optimalement reste constant au cours du temps.

En pratique, ce résultat scalaire s'adjoint (de manière redondante) ou se substitue à la condition d'optimilaté $H_{\mathbf{u}} = 0$.

2.5 Exemple : traînée minimale en écoulement hypersonique

Cet exemple correspond à l'Exemple 3, pp. 52-55 du Bryson and Ho.

On considère un corps de révolution en mouvement par rapport à l'atmosphère à une vitesse correspondant au régime "hypersonique" (froid), c'est-à-dire à un nombre de Mach M_∞ entre 5 et 10, pour fixer les idées. [1]

On cherche la forme cylindrique optimale correspondant à la traînée minimale (voir FIG. 2.1). On suppose que le nez de la forme correspond aux abscisses

$$0 \leq x \leq \ell \tag{2.19}$$

et se définit en coordonnées cylindrique (x, r) par l'équation

$$r = r(x) \quad (0 \leq x \leq \ell) \tag{2.20}$$

En $x = 0$, $r(0) = a$; en $x = \ell$ précisément, le bord d'attaque est vertical sur un rayon $r(\ell)$ dont la détermination est à préciser.

On rappellle que la traînée est la force de résistance à la pénétration du corps dans l'atmosphère exercée par l'air. Cette force est donnée par l'expression suivante :

$$D = -2\pi q \int_{r=a}^{r=0} C_p \, r dr = -2\pi q \left[\int_{r=a}^{r=r(\ell)} C_p \, r dr + \int_{r=r(\ell)}^{r=0} C_p \, r dr \right] \tag{2.21}$$

où :

- q est la pression dynamique ; $q = \frac{1}{2}\rho_\infty V_\infty^2 = \frac{1}{2}\gamma p_\infty M_\infty^2$ ($\gamma = C_p/C_v = 7/5$ pour un mélange de gaz parfaits diatomiques inertes, conformément à la modélisation de l'atmosphère), est une donnée du problème combinant les caractéristiques de l'atmosphère standard à une altitude donnée, à la vitesse d'avancement V_∞ supposée de l'engin, ou au nombre de Mach, $M_\infty = V_\infty/c_\infty$ ($c_\infty = \sqrt{\gamma p_\infty/\rho_\infty} = \sqrt{\gamma \mathsf{R} T_\infty}$ est la vitesse du son, c'est-à-dire une propriété locale thermodynamique de l'atmosphère) ;
- C_p est le coefficient adimensionné de pression ; $C_p = \frac{p - p_\infty}{q}$; en approximation hypersonique newtonnienne, ce coefficient est fonction uniquement de l'angle d'incidence θ : $C_p = 2\sin^2\theta$.

1. Ecoulement supersonique : $M_\infty > 1$; écoulement "hypersonique" : $M_\infty \gg 1$. L'approximation d'"écoulement "newtonnien" est justifiée lorsque le nombre de Mach est suffisamment grand, disons, $M_\infty > 5$ (cf. *Hypersonic Flow Theory*, Second Edition, Volume I Inviscid Flows, W. D. Hayes and R. F. Probstein, Academic Press, New York, London, 1966 ; chapter III). Cependant, si $M_\infty > 10$, d'autres phénomènes physiques apparaissent, notamment la dissociation chimique de l'atmosphère au passage du corps ("hypersonique chaud").

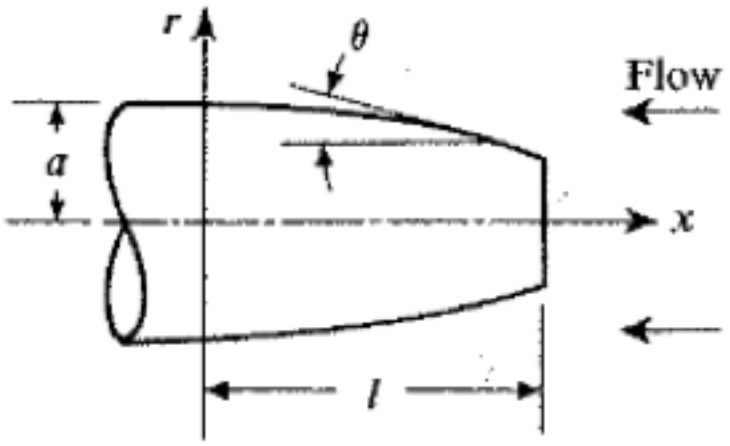


FIGURE 2.1 – Nomenclature pour le problème de traînée minimale d'un corps cylindrique en écoulement hypersonique froid

Il vient :

$$J := \frac{D}{4\pi q} = \tfrac{1}{2} r(\ell)^2 + \int_{r(\ell)}^{a} \sin^2\theta \, r dr$$

L'optimisation d'une forme plane équivaut à contrôler sa tangente : il convient donc de minimiser J par le contrôle

$$\mathbf{u} = u = \tan\theta = -\frac{dr}{dx} \Longleftrightarrow \dot{r}(x) = -u \tag{2.22}$$

agissant, par cette dynamique triviale, sur la variable d'état $r(x)$.

L'intégration de J donne :

$$J(\mathbf{u}) = \tfrac{1}{2}\left[r(\ell)\right]^2 + \int_0^{\ell} \frac{r u^3}{1+u^2}\, dx \tag{2.23}$$

Cette optimisation a bien la forme du problème type à condition de convenir des substitutions symboliques suivantes :

$$t \to x\,, \quad \mathbf{x}(t) \to r(x)\,, \quad \dot{r}(x) = f\Big(x; r(x), u(x)\Big) = -u(x) \tag{2.24}$$

Il s'agit donc d'un système autonome dont le hamiltonien, à l'optimalité, est constant :

$$H = \frac{r u^3}{1+u^2} + \lambda(-u) = \text{const.} \tag{2.25}$$

Le problème aux deux limites combiné à la condition d'optimalité s'écrit donc :

$$\begin{cases} r(0) = a\,, \quad \dot{r}(x) = \dfrac{\partial H}{\partial \lambda} = -u \\ \\ \qquad\qquad \dot{\lambda}(x) = -\dfrac{\partial H}{\partial r} = -\dfrac{u^3}{1+u^2}\,, \quad \lambda(l) = \dfrac{\partial \Phi\Big(r(\ell)\Big)}{\partial r(\ell)} = r(\ell) \\ \\ \qquad\qquad 0 = \dfrac{\partial H}{\partial u} = \dfrac{r u^2 (3+u^2)}{(1+u^2)^2} - \lambda \end{cases} \tag{2.26}$$

Il vient :

$$\lambda = \frac{ru^2(3+u^2)}{(1+u^2)^2} \tag{2.27}$$

D'où :

$$H = \frac{ru^3}{1+u^2} - \frac{ru^3(3+u^2)}{(1+u^2)^2} = -\frac{2ru^3}{(1+u^2)^2} = -\frac{r(\ell)}{2} \tag{2.28}$$

où la constante H a été évaluée en $x = \ell$ sachant qu'à cette limite la condition $\lambda(\ell) = r(\ell)$ exige que $u(\ell) = 1$, ce qui équivaut à $\theta(\ell) = \pi/4$. Il vient :

$$\frac{r}{r(\ell)} = \frac{(1+u^2)^2}{4u^3} \tag{2.29}$$

Par ailleurs, en intégrant

$$\frac{dx}{dr} = -\frac{1}{u} \tag{2.30}$$

il vient :

$$\frac{\ell - x}{r(\ell)} = \frac{1}{4}\left(\frac{3}{4u^4} + \frac{1}{u^2} - \frac{7}{4} - \ln\frac{1}{u}\right) \tag{2.31}$$

Les équations (2.29) et (2.31) constituent une représentation paramétrique de la forme optimale. Si les données spécifiées sont les dimensions a et ℓ, il convient de résoudre les deux équations transcendantales suivantes faisant intervenir les inconnues auxiliaires $r(\ell)$ et la pente $u_0 = u(0)$ en $x = 0$:

$$\frac{a}{r(\ell)} = \frac{(1+u_0^2)^2}{4u_0^3} \tag{2.32}$$

$$\frac{\ell}{r(\ell)} = \frac{1}{4}\left(\frac{3}{4u_0^4} + \frac{1}{u_0^2} - \frac{7}{4} - \ln\frac{1}{u_0}\right) \tag{2.33}$$

Les formes correspondantes sont representées à la FIG. 2.2, pour différentes valeurs du facteur de forme a/ℓ.

On peut noter que dans la limite $a/\ell \to 0$, la forme optimale est donnée par une loi en puissance :

$$\frac{r}{a} \sim \left(\frac{x}{\ell}\right)^{\frac{3}{4}} \tag{2.34}$$

2.6 Relation avec le calcul des variations

Nous allons maintenant montrer qu'on peut donner au problème type du calcul des variations, (1.1), la formulation d'un problème de contrôle optimal.

Pour cela, on convient d'utiliser la dérivée d'ordre le plus élévé, $y^{(n)}(x)$, comme variable de contrôle agissant sur un vecteur d'état constitué de la fonction $y(x)$ et de ses dérivées d'ordre $< n$. Pour se ramener aux notations habituelles, on convient également de faire les substitutions symboliques suivantes :

$$x \to t\,, \quad \left(y(x), y'(x), ..., y^{(n-1)}(x)\right)^t \to \mathbf{x}(t)\,, \quad y^{(n)}(x) \to \mathbf{u}(t)\,, \quad F \to \mathcal{L} \tag{2.35}$$

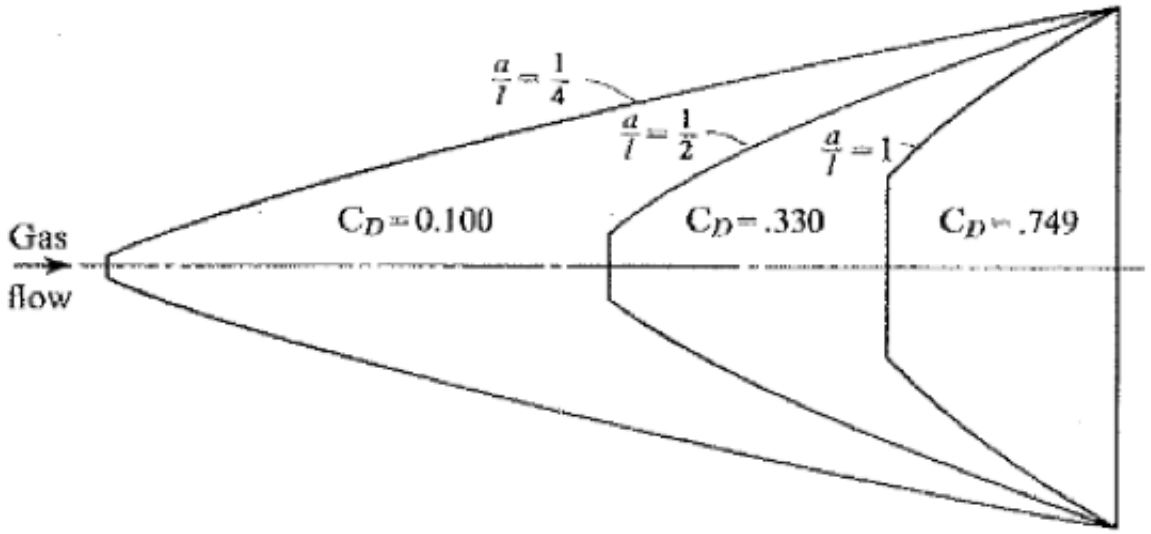


FIGURE 2.2 – Formes cylindriques optimales correspondant à une traînée minimale en écoulement hypersonique (froid), pour différentes valeurs du facteur de forme a/ℓ

La fonctionnelle prend alors la forme standard, ici sans terme dépendant explicitement de l'état final :

$$J(\mathbf{u}) = \int_0^T \mathcal{L}\Big(t; \mathbf{x}(t), \mathbf{u}(t)\Big)\, dt \tag{2.36}$$

où l'on a supposé $[a, b] = [0, T]$. La dynamique est constituée des équations triviales suivantes :

$$\begin{cases} \dfrac{d}{dt}\Big(y(t)\Big) = y'(t) \\ \dfrac{d}{dt}\Big(y'(t)\Big) = y''(t) \\ \quad \vdots \\ \dfrac{d}{dt}\Big(y^{(n-2)}(t)\Big) = y^{(n-1)}(t) \\ \dfrac{d}{dt}\Big(y^{(n-1)}(t)\Big) = y^{(n)}(t) = \mathbf{u}(t) \end{cases} \tag{2.37}$$

Ce système est bien de la forme :

$$\dot{\mathbf{x}}(t) = f\Big(t; \mathbf{x}(t), \mathbf{u}(t)\Big) \tag{2.38}$$

La seule différence formelle avec le problème type de contrôle optimal que nous avons étudié réside dans le fait qu'ici, la variable d'état, $\mathbf{x}(t)$ est spécifiée aux deux limites, et logiquement, dans l'expression de la fonctionnelle-coût aucun terme de pénalisation qui serait fonction explicite de l'état final n'est introduit. En adaptant le développement des conditions

d'optimalité, on pourra vérifier que celles-ci prennent la forme :

$$\mathbf{x}(0) = \mathbf{x}_0\,, \quad \dot{\mathbf{x}}(t) = H_\lambda^t = f\Big(t; \mathbf{x}(t), \mathbf{u}(t)\Big)\,, \quad \mathbf{x}(T) = \mathbf{x}_T \tag{2.39}$$

$$\dot{\lambda}(t) = -H_{\mathbf{x}}^t \tag{2.40}$$

$$H_{\mathbf{u}} = 0 \tag{2.41}$$

où le hamiltonien est défini par :

$$\begin{aligned} H &= \mathcal{L}\Big(t; \mathbf{x}(t), \mathbf{u}(t)\Big) + \lambda^t\, f\Big(\mathbf{x}(t), \mathbf{u}(t)\Big) \\ &= \mathcal{L}\Big(t; \mathbf{x}(t), \mathbf{u}(t)\Big) + \lambda_1(t)x_2(t) + \lambda_2(t)x_3(t) + ... + \lambda_{n-1}(t)x_n(t) + \lambda_n(t)\mathbf{u}(t) \end{aligned} \tag{2.42}$$

Par conséquent :

$$\dot{\lambda}_1(t) = -\mathcal{L}_{x_1}\,, \quad \dot{\lambda}_i(t) = -\mathcal{L}_{x_i} - \lambda_{i-1}(t) \quad (2 \leq i \leq n) \tag{2.43}$$

et

$$\mathcal{L}_{\mathbf{u}} + \lambda_n(t) = 0 \tag{2.44}$$

En effectuant les substitutions symboliques inverses de celles de (2.35), il vient :

$$\begin{aligned} \lambda_1'(x) + \frac{\partial F}{\partial y} &= 0 \\ \lambda_2'(x) + \lambda_1(x) + \frac{\partial F}{\partial y'} &= 0 \\ &\vdots \\ \lambda_n'(x) + \lambda_{n-1}(x) + \frac{\partial F}{\partial y^{(n-1)}} &= 0 \\ \lambda_n(x) + \frac{\partial F}{\partial y^{(n)}} &= 0 \end{aligned} \tag{2.45}$$

On applique à ces équations respectivement les opérateurs

$$1\,,\ -\frac{d}{dx}\,,\ +\frac{d^2}{dx^2}, ..., (-1)^n \frac{d^n}{dx^n} \tag{2.46}$$

et on additionne les résultats. On retrouve alors précisément l'équation d'Euler-Lagrange :

$$\frac{\partial F}{\partial y} - \frac{d}{dx}\Big(\frac{\partial F}{\partial y'}\Big) + \frac{d^2}{dx^2}\Big(\frac{\partial F}{\partial y''}\Big) + ... + (-1)^n \frac{d^n}{dx^n}\Big(\frac{\partial F}{\partial y^{(n)}}\Big) = 0 \tag{2.47}$$

En conclusion, le formalisme du contrôle optimal contient donc celui du calcul des variations comme cas particulier.

Exercice 8
Dans le cas où de plus F ne dépend pas explicitement de s on a l'intégrale première d'Euler. Retrouver ce résultat à partir du formalisme du contrôle optimal.

2.7 Equation de Riccati

Nous allons maintenant étudier le cas particulier très important où **la fonctionnelle-coût est quadratique et la dynamique linéaire**. Plus précisément, on considère un système dont la dynamique s'écrit :

$$\boxed{\mathbf{x}(0) = \mathbf{x}_0\,, \quad \dot{\mathbf{x}}(t) = F(t)\mathbf{x}(t) + G(t)\mathbf{u}(t) \quad (0 \leq t \leq T)} \tag{2.48}$$

A nouveau, $\mathbf{x}(t)$ est à valeurs dans $\mathbb{R}^n$, et $\mathbf{u}(t)$ dans $\mathbb{R}^p$: les matrices $F(t)$ et $G(t)$ sont donc de dimensions $n \times n$ et $n \times p$ respectivement. Le temps final T est ici fixé.

On souhaite amener le système au temps final $\mathbf{x}(T)$ aussi près que possible d'un état donné pris conventionnellement égal à 0 ; pour cela, on pénalise le terme $\mathbf{x}(T)^t S_T \mathbf{x}(T)$. On souhaite également éviter des niveaux prohibitifs de contrôle et d'état au cours du trajet. Par exemple, pour un véhicule de transport on souhaite éviter une consommation de carburant excessive, ou une trajectoire trop extrême. A cette fin, on décide de minimiser la fonctionnelle quadratique suivante :

$$\boxed{J = \tfrac{1}{2}\mathbf{x}(T)^t S_T \mathbf{x}(T) + \tfrac{1}{2}\int_0^T \Big(\mathbf{x}(t)^t A \mathbf{x}(t) + \mathbf{u}(t)^t B \mathbf{u}(t)\Big)\, dt} \tag{2.49}$$

où S_T , A et B sont des matrices constantes symétriques définies-positives (S_T et A sont $n \times n$; B est $p \times p$) et on rappelle que l'indice supérieur ^{t} indique la transposition.

On introduit la variable adjointe $\lambda(t)$, à valeurs dans $\mathbb{R}^n$, et le hamiltonien :

$$\begin{aligned} H &= H\Big(t; \mathbf{x}(t), \lambda(t), \mathbf{u}(t)\Big) \\ &= \mathcal{L} + \lambda^t f \\ &= \tfrac{1}{2}\Big(\mathbf{x}(t)^t A \mathbf{x}(t) + \mathbf{u}(t)^t B \mathbf{u}(t)\Big) + \lambda^t \Big(F(t)\mathbf{x}(t) + G(t)\mathbf{u}(t)\Big) \end{aligned} \tag{2.50}$$

Le problème aux deux limites s'écrit alors :

$$\begin{cases} \mathbf{x}(0) = \mathbf{x}_0\,, & \dot{\mathbf{x}}(t) = H^t_\lambda = F(t)\mathbf{x}(t) + G(t)\mathbf{u}(t) \\ & \dot{\lambda}(t) = -H^t_{\mathbf{x}} = -A\mathbf{x}(t) - F(t)^t \lambda(t)\,, \quad \lambda(T) = S_T \mathbf{x}(T) \quad (0 \leq t \leq T) \end{cases} \tag{2.51}$$

Par ailleurs, la condition d'optimalité s'écrit :

$$0 = H_{\mathbf{u}} = \mathbf{u}(t)^t B + \lambda^t G(t) \Longrightarrow \mathbf{u}(t) = -B^{-1} G(t)^t \lambda(t) \tag{2.52}$$

En reportant ce résultat dans la formulation aux limites ci-dessus, il vient :

$$\mathbf{x}(0) = \mathbf{x}_0\,, \quad \begin{pmatrix} \dot{\mathbf{x}}(t) \\ \dot{\lambda}(t) \end{pmatrix} = \begin{pmatrix} F(t) & -G(t)B^{-1}G(t)^t \\ -A & -F(t)^t \end{pmatrix} \begin{pmatrix} \mathbf{x}(t) \\ \lambda(t) \end{pmatrix}, \quad \lambda(T) = S_T \mathbf{x}(T) \tag{2.53}$$

Noter que le membre de droite provient de dérivations du hamiltonien, ici quadratique ; d'où la linéarité du système. Cette observation nous conduit à chercher une solution pour laquelle il existe une relation fonctionnelle linéaire entre les variables d'état et adjointe. On pose :

$$\lambda(t) = S(t)\mathbf{x}(t) \tag{2.54}$$

où $S(t)$ est une matrice $n \times n$ inconnue.

Par substitution, il vient :

$$\begin{aligned}\dot{\lambda}(t) &= \dot{S}(t)\mathbf{x}(t) + S(t)\dot{\mathbf{x}}(t) \\ &= \dot{S}(t)\mathbf{x}(t) + S(t)\Big(F(t)\mathbf{x}(t) - G(t)B^{-1}G(t)^t S(t)\mathbf{x}(t)\Big) \\ &= -A\mathbf{x}(t) - F(t)^t S(t)\mathbf{x}(t) \end{aligned} \tag{2.55}$$

Pour satisfaire les équations, il suffit donc que la matrice $S(t)$ soit solution de l'**équation de Riccati** suivante :

$$\boxed{\dot{S}(t) = -A - S(t)F(t) - F(t)^t S(t) + S(t)G(t)B^{-1}G(t)^t S(t)} \tag{2.56}$$

Le membre de droite de cette **EDO matricielle** est une matrice symétrique qui dépend quadratiquement de la matrice $S(t)$. L'EDO est soumise à la condition finale :

$$\boxed{S(T) = S_T} \tag{2.57}$$

afin de satisfaire la condition $\lambda(T) = S_T\mathbf{x}(T)$. L'élimination de la variable adjointe $\lambda(t)$ conduit à la **loi de commande optimale** suivante :

$$\boxed{\mathbf{u}^\star(t) = -B^{-1}G(t)^t S(t)\,\mathbf{x}(t)} \tag{2.58}$$

En conclusion, au prix de la **résolution, une fois pour toutes, de l'équation matricielle de Riccati**, qui en général fait intervenir $n(n+1)/2$ EDO indépendantes, on peut **piloter optimalement le système en fonction seulement de la valeur instantannée du vecteur d'état $\mathbf{x}(t)$** (cf. systèmes embarqués).

Chapitre 3

Optimisation de systèmes distribués

On appelle ici **"système distribué"** un système régi par une EDP satisfaite dans un certain domaine d'espace Ω.

Un système distribué peut être soumis à un **contrôle à la frontière** par une condition à une limite particulière, ou à un **contrôle distribué** intervenant dans tout le domaine par un terme de l'EDP. Nous donnerons des exemples de ces deux cas. L'optimisation de forme sous contrainte d'EDP est un cas particulier de système distribué contrôlé à la frontière.

3.1 Exemples d'optimisation géométrique en ingénierie

3.1.1 Exemple en mécanique

Il est fortement recommandé à l'étudiant de consulter les ouvrages de G. Allaire sur ces questions, notamment [3].

Optimisation de l'épaisseur d'une membrane Ω : domaine plan occupé par la membrane au repos. On suppose que la membrane tendue est fixée par son bord $\partial\Omega$. La membrane est soumise a un chargement correspondant à une force verticale de mesure f (par exemple son poids). Le déplacement vertical est noté u. Il satisfait l'équation classique des membranes :

$$\begin{cases} -\operatorname{div}(A\nabla u) = f \quad (\Omega) \\ u = 0 \quad (\partial\Omega) \end{cases} \tag{3.1}$$

Dans le cas le plus général, le coefficient A est un tenseur qui représente la résistance mécanique du matériau aux déformations.

Considérons le cas d'une membrane constituée d'un matériau homogène d'épaisseur variable $h(\mathbf{x})$ bornée par les **contraintes suivantes sur la fonction à optimiser** :

$$0 < h_{\min} \leq h(\mathbf{x}) \leq h_{\max} < \infty \tag{3.2}$$

Il vient alors :

$$A = A(\mathbf{x}) = \mu h(\mathbf{x}) I \tag{3.3}$$

où μ est un coefficient constant, caractéristique du matériau. Souvent, le problème est soumis à une **contrainte fonctionnelle** sur le poids

$$W = \rho L \int_\Omega h(\mathbf{x})\, dx = \text{const.} \tag{3.4}$$

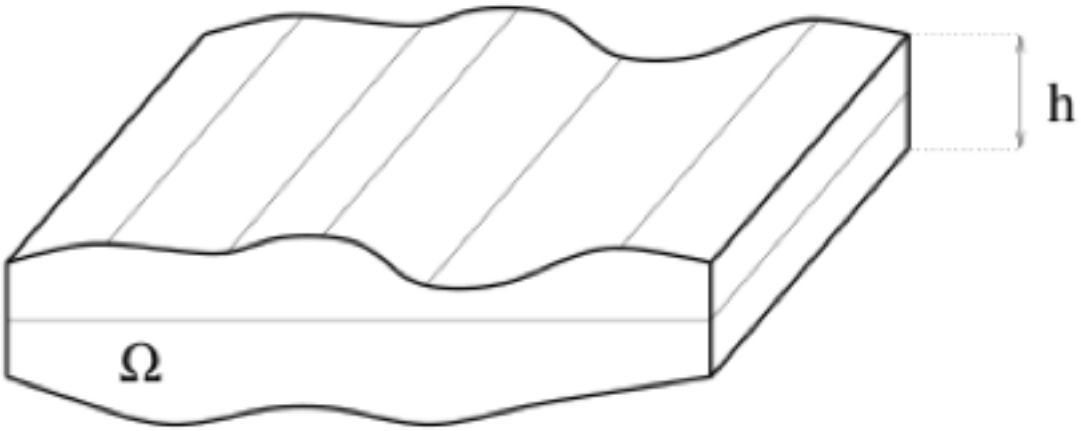


FIGURE 3.1 – La membrane d'épaisseur variable $h(\mathbf{x})$ (exemple tiré de Allaire (2005), [2])

L'optimisation de $h(\mathbf{x})$ consiste généralement à minimiser une **fonctionnelle-coût** de la forme :

$$J(h) = \int_\Omega j(u)\, dx \tag{3.5}$$

Plusieurs choix de $j(u)$ peuvent être faits :

– Minimisation de la **compliance** :

$$\boxed{j(u) = fu} \tag{3.6}$$

La "compliance" est le produit scalaire de la force appliquée f avec la déformation u, c'est-à-dire le travail de cette force associé à ce déplacement. À charge donnée, plus la structure est rigide, et moins elle se déforme, et moindre est la "compliance". En minimisant la compliance, on vise donc à augmenter la rigidité.

– Ciblage d'une déformation donnée (objectif de fabrication), ce qui constitue un "problème inverse" :

$$\boxed{j(u) = |u - u_0|^2} \tag{3.7}$$

– Etc.

Dans le cas de deux matériaux homogènes, on aura :

$$\begin{cases} A(\mathbf{x}) = \chi(\mathbf{x})A_1 + \Big(1 - \chi(\mathbf{x})\Big)A_2 \\ W \sim \displaystyle\int_\Omega \Big[\chi(\mathbf{x})\rho_1 + \Big(1 - \chi(\mathbf{x})\Big)\rho_2\Big]\, dx \end{cases} \tag{3.8}$$

Pour ces différents problèmes, on optimise une variable distribuée $h(\mathbf{x})$ soumise à des bornes inférieure et supérieure, et une contrainte intégrale (poids). Cette variable intervient

comme un coefficient variable d'une EDP, et l'objectif est de minimiser une fonctionnelle-coût $J(h)$, qui s'exprime au moyen de la solution u de l'EDP.

Enfin, notons que des problèmes de même type sont issus de phénomènes physiques différents, notamment ceux liés à la conduction thermique ou électrique.

Optimisation de forme d'un élément encastré ˜

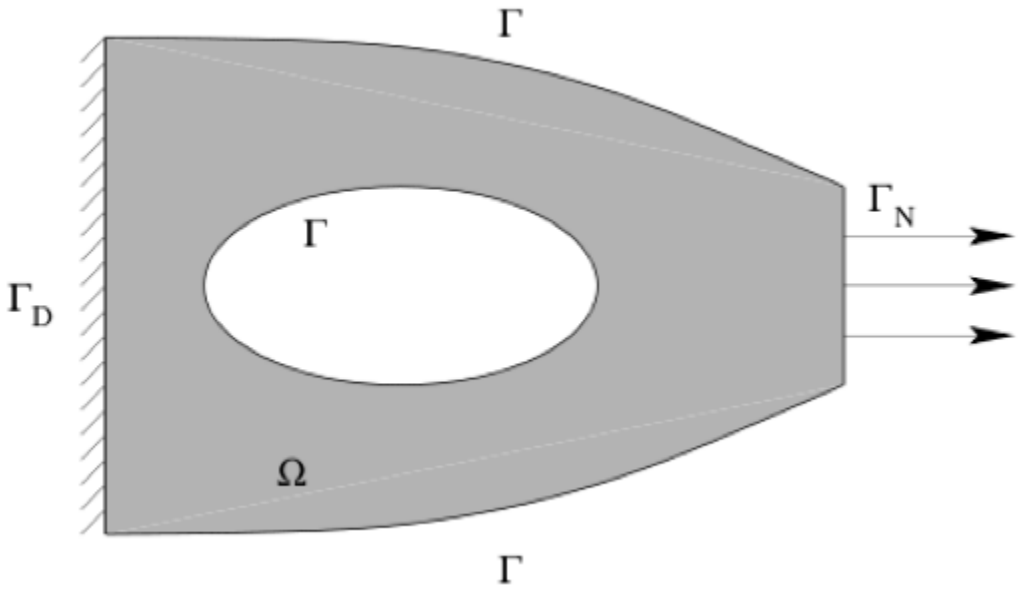


FIGURE 3.2 – Elément encastré et conditions aux limites (exemple tiré de Allaire (2005), [2])

$$\begin{cases} -\Delta u = 0 \quad (\Omega) \\ u = 0 \quad (\Gamma_D) \\ \dfrac{\partial u}{\partial n} = g \quad (\Gamma_N) \\ \dfrac{\partial u}{\partial n} = 0 \quad (\Gamma) \end{cases} \tag{3.9}$$

Il s'agit d'optimiser la forme du bord libre et du trou.

Fonctionnelle-coût ; au choix :

$$J(\Omega) = \begin{cases} \displaystyle\int_{\Gamma_N} gu \, d\gamma \quad \text{(compliance)} \\ \displaystyle\int_{\Omega} |u - u_0|^2 \, d\omega \\ \text{fréquence propre, contrainte moyennne ou maximale, etc.} \end{cases} \tag{3.10}$$

Comment procéder pour représenter la forme ?

– par une paramétrisation particulière des formes s'appuyant sur quelques paramètres ex. : trou ellipsoïdal ($\frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} = 1$: 4 paramètres : $x_0,\ y_0,\ a,\ b$) et bord Γ parabolique... pas très général : ne permet pas d'approcher aussi finement que l'on pourrait souhaiter la solution continue.

– par une représentation de type CAO (courbes de Bézier, B-splines, Nurbs) cf. Partie I (Optimisation paramétrique, Régis Duvigneau)

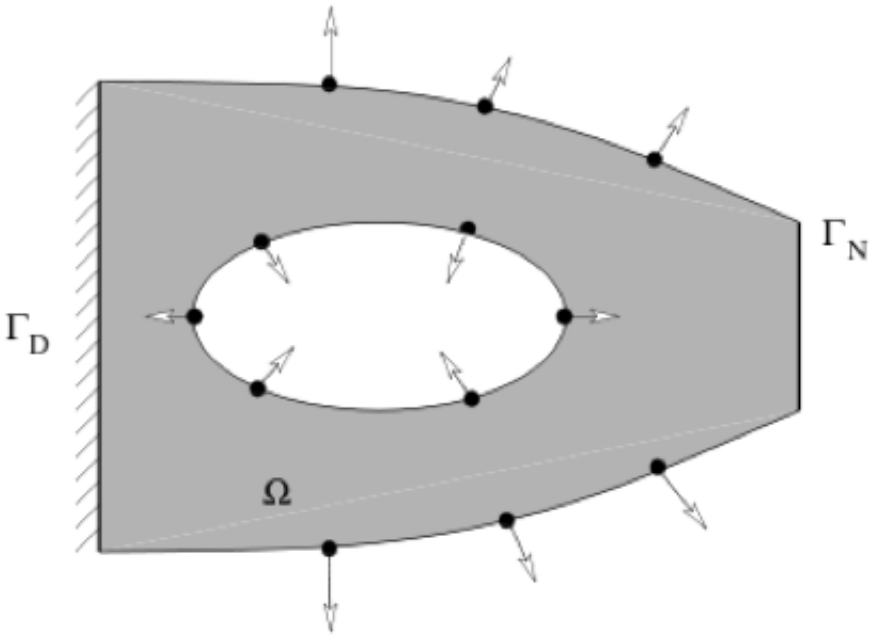


FIGURE 3.3 – Représentation d'un bord par des NURBS (exemple tiré de Allaire (2005), [2])

Ici encore on est ramené à optimiser dans un espace de dimension finie ; cependant le nombre de degrés de liberté (ddl) peut ici être grand, comme doit l'être le nombre de points de discrétisation d'un bord de maillage de type éléments-finis.

Optimisation topologique Exemple : optimiser la répartition de la masse d'une pièce mécanique sous chargement pour en maximiser la rigidité sous contrainte de masse totale imposée. La solution optimale peut comporter un ou plusieurs trous.

Cf. thèse de Beuzit (sous la direction d'A. Habbal). Dans une première phase (optimisation topologique), on optimise une variable continue à valeurs dans [0,1] et représentant la répartition moyenne locale de la masse. Lorsque cette variable est < 0.5 on assimile à un trou, et sinon à la présence de matière. Après cette identification approchée des trous, dans une deuxième phase, on optimise leurs formes au moyen de représentations de type B-splines (optimisation de forme).

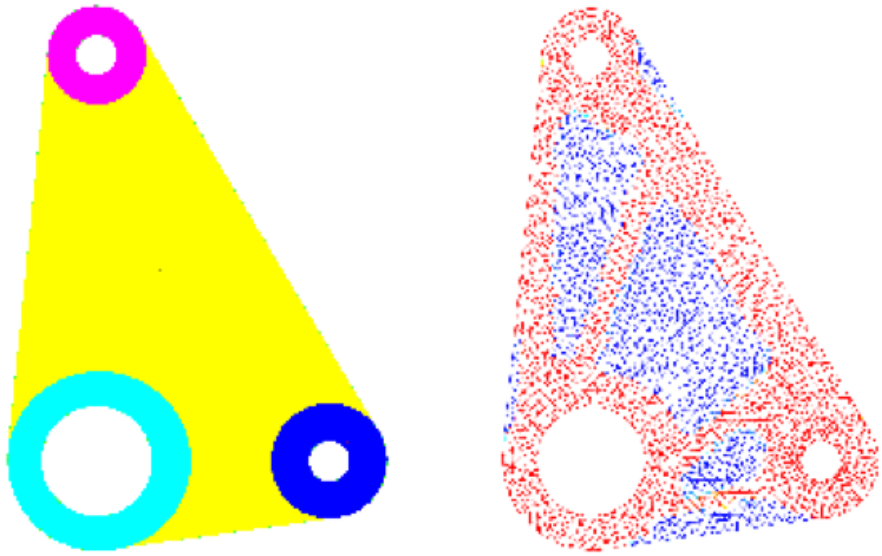


FIGURE 3.4 – Identification de la topologie (Beuzit-Habbal).

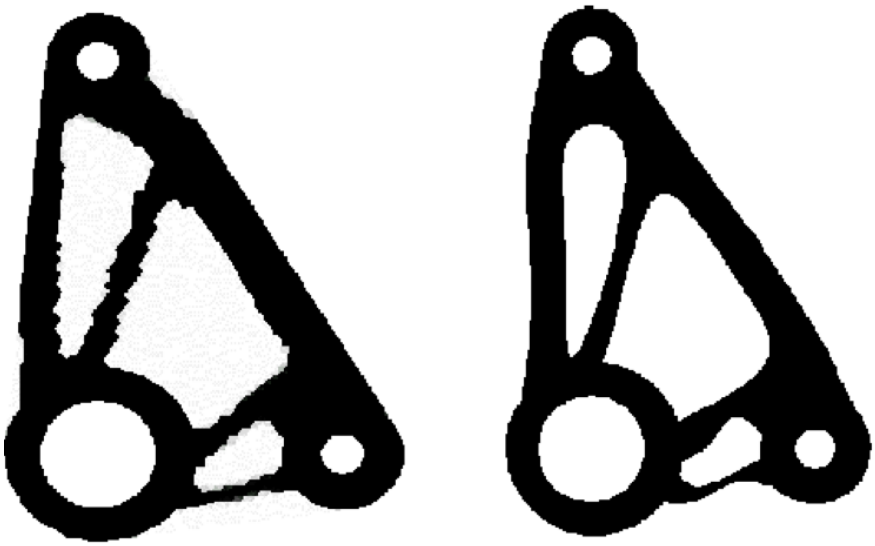
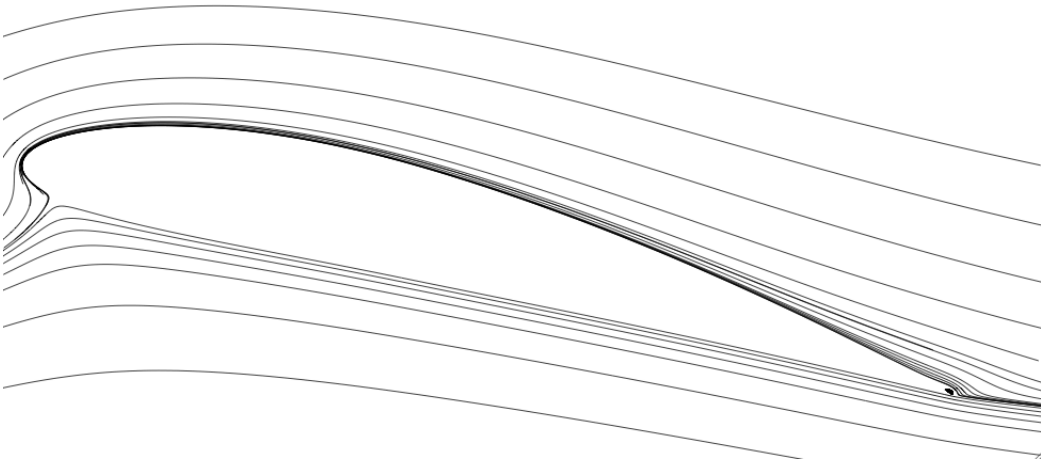


FIGURE 3.5 – Détermination précise de la forme, une fois la topologie fixée (Beuzit-Habbal)

3.1.2 Exemples en aérodynamique

En aérodynamique externe, on cherche à optimiser la forme d'une voilure vis-à-vis de critères principalement liés aux forces de portance et de traînée.

(a) Visualisation des lignes de courant dans un écoulement bidimensionnel (Ici, la modélisation correspond aux équations de Navier-Stokes, en écoulement incompressible turbulent ; tiré de la thèse de R. Duvigneau.)



(b) Repère xyz lié au corps et repère aérodynamique XYZ lié à la vitesse à l'infini amont $\overrightarrow{V}_\infty$; incidence $\alpha = AoA$

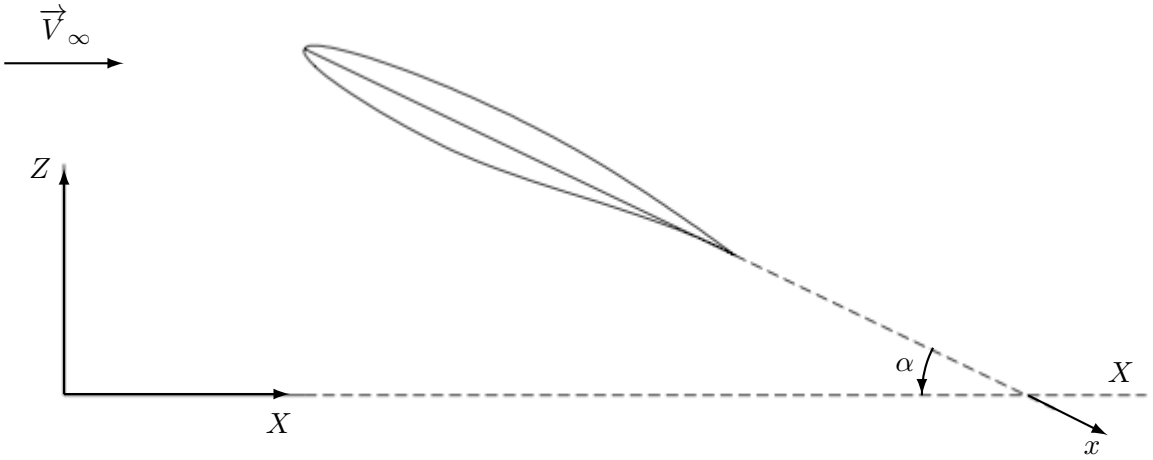


FIGURE 3.6 – Configuration d'aérodynamique externe autour de la voilure d'un aéronef

L'engin se déplace à la vitesse $\overrightarrow{V}_S$ par rapport à un repère fixe (galiléen) lié au sol. Dans la représentation eulérienne, on observe dans un repère relatif lié à l'engin (x, y, z). Usuellement l'axe des x est longitudinal, l'axe des y transversal, et l'axe des z orthogonal à x dans un plan vertical. On s'intéresse au mouvement de l'air relatif à l'avion : on parle donc d'écoulement d'air autour de l'avion. Le principe de l'action et de la réaction nous permet de comprendre que cette situation est semblable à celle de la soufflerie où un modèle est fixé et un écoulement

d'air est créé permettant la réalisation de mesures, notamment de pression, vitesse, etc.

Dans le cas le plus général, l'écoulement est décrit par les fonctions

$$W = W(x,y,z,t) = \begin{cases} \rho & \text{masse volumique} \\ \rho u & \text{quantité de mouvement volumique (composante-}x\text{)} \\ \rho v & \text{quantité de mouvement volumique (composante-}y\text{)} \\ \rho w & \text{quantité de mouvement volumique (composante-}z\text{)} \\ E & \text{énergie totale (cinétique+interne) volumique} \end{cases} \tag{3.11}$$

Le vecteur $W(x,y,z,t)$ est une grandeur physique attachée à la particule matérielle fluide qui à l'instant t se situe en (x,y,z). Il s'agit d'une grandeur physique par unité de volume ; on dit aussi "grandeur spécifique". Dans ce qui suit, on s'intéresse principalement au cas de l'écoulement permanent (ou stationnaire) :

$$\frac{\partial W}{\partial t} = 0 \quad (\forall (x,y,z,t)) \quad \text{(écoulement stationnaire)} \tag{3.12}$$

A l'infini, en amont ou en aval de l'avion, l'atmosphère est supposée au repos, c'est-à-dire en écoulement uniforme à vitesse

$$\overrightarrow{V}_\infty = -\overrightarrow{V}_S \tag{3.13}$$

dans les conditions thermodynamiques de l'atmosphère standard à l'altitude considérée (p_∞, ρ_∞, T_∞), pour lesquelles la célérité du son est donnée, pour un mélange de gaz parfaits diatomiques, par la relation

$$c_\infty = \sqrt{\frac{\gamma p_\infty}{\rho_\infty}} = \sqrt{\gamma \mathsf{R} T_\infty} \tag{3.14}$$

Pour des raisons déjà un peu esquissées, le premier paramètre caractéristique de l'écoulement est le nombre de Mach, local, ou ici à l'infini

$$M_\infty = \frac{V_\infty}{c_\infty} \tag{3.15}$$

C'est le rapport de la vitesse matérielle de la particule fluide locale à la vitesse locale de propagation des ondes thermodynamiques (notamment de pression) générées par le déplacement de la particule fluide elle-même.

Lorsque localement l'écoulement est **subsonique**, $M < 1$, l'information thermodynamique, notamment les variations de pression engendrées par le passage de l'engin, se propage sphériquement plus vite que l'engin progresse, et l'amont autant que l'aval perçoit la perturbation. On dit un peu trivialement que "tout influence/dépende de tout". Le problème mathématique est de **type elliptique** et requiert la résolution globale d'un grand système couplé aux limites.

A l'inverse, en **écoulement supersonique**, $M_\infty > 1$, le repère d'observation qui se déplace avec l'avion progresse plus rapidement que les ondes qui ont été générées par le passage de l'engin. L'espace entier n'est pas influencé, mais seulement une zone attachée à l'engin qui dans le cas où l'engin serait remplacé par un point matériel (ou une aigüille) serait un "cône de Mach"[1]. Le problème mathématique est de **type hyperbolique**. En principe la région

1. `http://fr.wikipedia.org/wiki/Supersonique`

hyperbolique peut être résolue par une technique d'avancement en espace, telle que la méthode des caractéristiques, en respectant le sens de propagation des ondes.

En pratique, pour calculer un écoulement permanent partiellement subsonique et partiellement supersonique, aujourd'hui, la méthode la plus courante consiste à simuler numériquement, typiquement par volumes finis, un écoulement pseudo-instationnaire. A partir d'une condition initiale assez arbitraire, on procède par avancement en temps jusqu'à atteindre l'état stationnaire compatible avec les conditions aux limites. Dans ce qui suit on se place dans ce cas.

Les équations de l'écoulement en fluide compressible peuvent être établies et utilisées suivant différents niveaux de complexité croissante :

- équation du potentiel complet ; modélisation utile en exploitation intensive en avant-projet ;
- équations d'Euler : modèle de fluide parfait (non visqueux) compressible ; permet la résolution correcte de l'onde, mais pas le calcul des forces de frottement et transferts thermiques pariétaux ;
- équations de Navier-Stokes : incluent en plus la modélisation des forces visqueuses et du tenseur de diffusion ;
- équations RANS (*Reynolds-Averaged Navier-Stokes*) : incluent en plus un modèle de turbulence.

Aujourd'hui, les modèles 3D Euler et RANS, sont couramment utilisés dans les laboratoires et l'industrie, au niveau de l'**analyse de "haute-fidélité"** et de l'**optimisation**.

On s'intéresse ici à ces modèles de haute fidélité, et plus spécifiquement aux équations d'Euler, qui expriment sous **forme conservative**, ou **forme divergence** les lois de bilan, ou de conservation de la masse (continuité), quantité de mouvement (loi de Newton pour un fluide), et de l'énergie pour un fluide parfait. Sous forme différentielle, elles s'expriment dans le cas général d'un écoulement instationnaire comme suit :

$$\frac{\partial W}{\partial t} + \frac{\partial F(W)}{\partial x} + \frac{\partial G(W)}{\partial y} + \frac{\partial H(W)}{\partial z} = 0 \tag{3.16}$$

où W est le **vecteur des variables conservatives**, et $F(W)$, $G(W)$ et $H(W)$ sont les **vecteurs de flux** suivant x, y et z :

$$W = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{pmatrix}, \; F(W) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho v u \\ \rho w u \\ (E+p)u \end{pmatrix}, \; G(W) = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ \rho w v \\ (E+p)v \end{pmatrix}, \; H(W) = \begin{pmatrix} \rho w \\ \rho u w \\ \rho v w \\ \rho w^2 + p \\ (E+p)w \end{pmatrix} \tag{3.17}$$

Exercice 9

Etablir la première de ces équations, celle de conservation de la masse, ou équation de continuité :

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} = \frac{\partial \rho}{\partial t} + div\left(\rho \overrightarrow{V}\right) = 0 \tag{3.18}$$

La pression n'est pas une composante du vecteur W, et pour clore le système, il convient de l'exprimer en fonction des ces variables. Modélisant l'air comme un mélange à composition constante de gaz parfaits diatomiques (O^2 et N^2) :

$$p = \rho \mathsf{R} T = (\gamma - 1)\left(E - \frac{(\rho u)^2 + (\rho v)^2 + (\rho w)^2}{2\rho}\right) \tag{3.19}$$

Exercice 10
Etablir cette relation.

Les vecteurs de flux $F(W)$, $G(W)$ et $H(W)$ admettent des matrices jacobiennes 5×5 notées $A(W)$, $B(W)$ et $C(W)$ dont les expressions et la diagonalisation sont bien connues [2]. En particulier, quel que soit le vecteur d'onde $\overrightarrow{k} = (k_1, k_2, k_3)$, la matrice $k_1 A(W) + k_2 B(W) + k_3 C(W)$ admet les **valeurs propres réelles** suivantes : $\overrightarrow{k}.\overrightarrow{V}$ (triple), $\overrightarrow{k}.\overrightarrow{V} - c\,\|k\|$ et $\overrightarrow{k}.\overrightarrow{V} + c\,\|k\|$ où c est la vitesse locale du son, $c = \sqrt{\gamma p/\rho}$.

Exercice 11
Que devient ce résultat dans le cas d'un écoulement monodimensionnel ? Vérifier la diagonalisation.

En conséquence, linéarisons les équations d'Euler instationnaires autour d'un état local constant W_0 ; on pose :

$$W(x, y, z, t) = W_0 + W'(x, y, z, t) \tag{3.20}$$

Il vient :

$$\frac{\partial W'}{\partial t} + A_0 \frac{\partial W'}{\partial x} + B_0 \frac{\partial W'}{\partial y} + C_0 \frac{\partial W'}{\partial z} + ... = 0 \tag{3.21}$$

où $A_0 = A(W_0)$, $B_0 = B(W_0)$, et $C_0 = C(W_0)$. En décomposant la condition initiale dans une base de Fourier,

$$W'(x, y, z, 0) = \sum_{(k_1, k_2, k_3)} \exp\left(ik_1 x + ik_2 y + ik_3 z\right) \tilde{W}_k \tag{3.22}$$

où $\tilde{W}_k$ est un **vecteur propre réel** associé à la **valeur propre réelle** $\mathbf{p}_k$,

$$(k_1 A_0 + k_2 B_0 + k_3 C_0)\tilde{W}_k = \mathbf{p}_k \tilde{W}_k \tag{3.23}$$

il vient :

$$W'(x, y, z, t) = \sum_{(k_1, k_2, k_3)} \exp\left(-i\mathbf{p}_k t\right) \exp\left(ik_1 x + ik_2 y + ik_3 z\right) \tilde{W}_k \tag{3.24}$$

ce qui correspond à un **train d'ondes simples**, caractéristique d'un système linéarisé **hyperbolique**. L'hyperbolicité justifie qu'on puisse **intégrer les équations d'Euler instationnaires par avancement en temps**, quelles que soient les conditions initiales.

Un autre trait mathématique essentiel des équations d'Euler est la **nonlinéarité**. On apprend dans les cours sur les EDP hyperboliques nonlinéaires que ces équations n'ont de solutions qu'au sens faible, qu'elles présentent des **discontinuité**. Ce phénomène a été compris et correctement évalué par les aérodynamiciens, notamment Rankine-Hugoniot, bien avant la

2. Warming, R. F., Beam, R., and Hyett, B. J. : Diagonalization and Simultaneous Symmetrization of Gas-Dynamics Matrices , Math. Comput. 29, 1975, 1037-1045

théorie des distributions. Ces discontinuités peuvent être de différentes natures : ondes de choc, ondes de détente, surfaces de glissement. Cette difficulté majeure est aujourd'hui très bien prise en compte par les calculs par **Volumes Finis** (VF) qui, par leur formulation même discrétisent de manière consistante [3] les lois de conservation écrites sous **forme intégrale**. Ces aspects font l'objet de cours spécialisés que l'on ne peut aborder ici. (Pour un aperçu des schémas VF en mécanique des fluides, voir par exemple : Modèles discrets et Schémas itératifs, JAD, chapitre 6).

Revenons aux aspects physiques. Les moyens-courriers civils actuels (par ex. Paris-Nice), volent en croisière en **régime transsonique**. Pour fixer les idées : $M_\infty \sim 0.8$. Sur l'extrados, surface supérieure de la voilure, l'écoulement accélère, devient supersonique et redevient subsonique (comme nécessaire) au travers d'une **onde de choc** (FIG. 3.7). Cette surface de discontinuité correspond à un saut positif de pression qui dégrade la performance aérodynamique de la voilure et le problème principal du concepteur est celui de la réduction de l'intensité de ce choc.

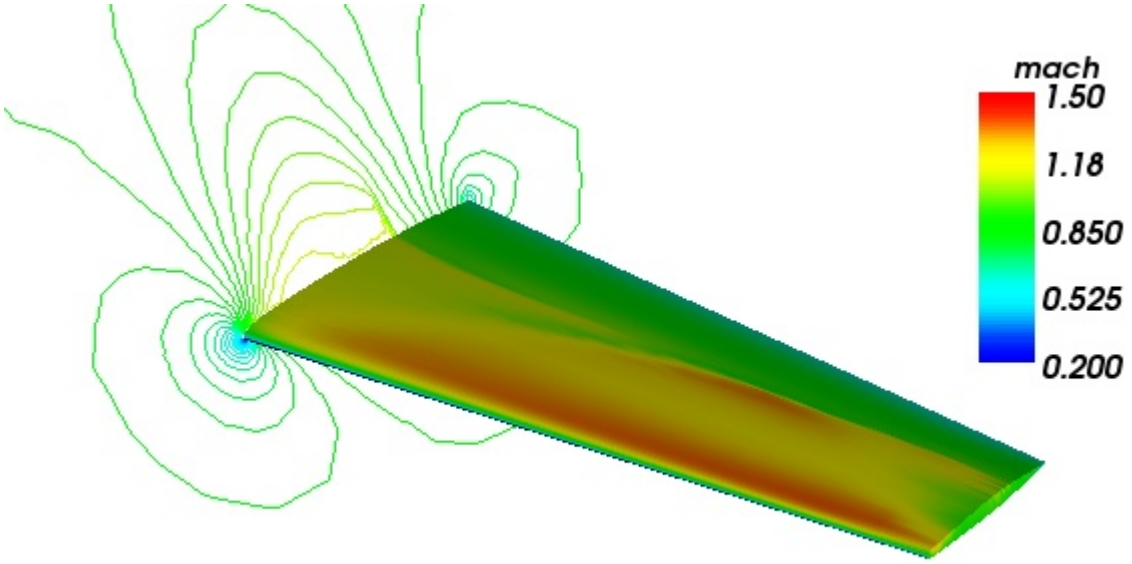


FIGURE 3.7 – Visualisation des niveaux de nombre de Mach local sur une voilure en écoulement eulérien transsonique ; onde de choc

Sur un élément de surface de normale extérieure $\overrightarrow{n}dS$ s'exerce un force de pression $-p\overrightarrow{n}dS$. En écoulement visqueux, on prendrait en compte en plus de la force de frottement proportionnelle au gradient normal de vitesse, $-K\frac{\partial \overrightarrow{V}}{\partial n}$.

L'intégrale à la surface de la voilure de la force de pression fournit la composante d'onde de la résultante des forces aérodynamiques exercées par l'écoulement sur la voilure :

$$\overrightarrow{R} = -\iint_S p\overrightarrow{n}dS \tag{3.25}$$

Le praticien conventionnellement adimensionne cette force en normalisant la pression à la

3. En hyperbolique nonlinéaire, la notion de consistance est très spécifique.

pression dynamique, $q_\infty = \frac{1}{2}\rho_\infty V_\infty^2$, ce qui permet de définir le coefficient de pression

$$C_p = \frac{p - p_\infty}{q_\infty} = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty V_\infty^2} \tag{3.26}$$

et en normalisant la surface à la surface en plan S :

$$\frac{\overrightarrow{R}}{q_\infty S} = -\iint C_p \overrightarrow{n} \frac{dS}{S} := \begin{pmatrix} C_x \\ C_y \\ C_z \end{pmatrix} \tag{3.27}$$

Cette force, et les coefficients aérodynamiques associés, sont habituellement exprimées dans le **repère aérodynamique** (X, Y, Z) dont l'axe des X est porté par le vecteur vitesse $\overrightarrow{V}_\infty$, et l'axe des Y (transersal) est confondu à celui des y. L'angle

$$\alpha = (Ox, OX) = AoA \tag{3.28}$$

est appelé **angle d'incidence** par les aérodynamiciens, ou **angle d'attaque** par les mécaniciens du vol, souvent noté à l'anglaise *AoA (angle of attack)*. Par projection dans le repère aérodynamique, il vient :

$$\begin{cases} C_D = C_X = C_x \cos\alpha + C_y \sin\alpha & \textbf{“coefficient de traînée”} \textit{ (drag)} \\ C_L = C_Z = C_y \cos\alpha - C_x \sin\alpha & \textbf{“coefficient de portance”} \textit{ (lift)} \end{cases} \tag{3.29}$$

On peut définir de manière analogue un coefficient C_Y, et 3 coefficients de moment. Du point de vue de la mécanique du vol, les 3 coefficients de force régissent le mouvement du centre de masse de l'engin dans l'espace, et les 3 coefficients de moment le mouvement de l'engin autour de son centre de masse.

Considérons le cas particulier du vol de croisière stabilisé. Pour calculer les coefficients aérodynamiques, il faut connaître tous les éléments géométriques (forme Γ de la voilure et angle d'incidence), et l'écoulement qui dépend de ces éléments ainsi que des conditions de vol (altitude H et atmosphère standard, nombre de Mach, éventuellement nombre de Reynolds en écoulement visqueux, éventuellemnt modèle de turbulence en RANS) :

$$C_D,\ C_L = C_D,\ C_L\left(H, M_\infty, ...; \Gamma, \alpha\right) \tag{3.30}$$

Dans ces conditions un problème fondamental pour le concepteur de voilure est celui de la réduction de traînée sous contrainte de portance :

$$\boxed{\begin{array}{l} \min\limits_{\Gamma,\alpha} C_D \\ \text{Sous contrainte : } C_L \geq C_{L_0} \end{array}} \tag{3.31}$$

Ce problème conditionne directement la consommation en kérosène (donc le coût commercial du vol), la pollution atmosphérique qu'elle engendre (donc le coût écologique), ou inversement le rayon d'action (distance franchissable) à masse au décollage donnée.

Il n'est pas obligatoire d'inclure l'angle d'incidence dans les variables d'optimisation. Si on ne le fait pas, le problème est un problème d'optimisation de la forme Γ. Dans la réalité

industrielle, ce problème est soumis à de très nombreuses contraintes, y compris des contraintes de fabrication.

Ce cas correspond à un système distribué régi par un jeu approprié d'EDP en 3D modélisant la mécanique des fluides. Ces EDP s'appliquent à un domaine dont un bord est optimisé. Sur ce bord se mesure également la fonctionnelle coût, et les contraintes. Notons qu'ici que les fonctionnelles optimisées (coefficients aérodynamiques) sont des intégrales sur Γ, et Γ est également la variable d'optimisation. Pour cette raison, le terme principal du gradient de forme est un terme explicite de déformation géométrique. Cette observation a conduit certains auteurs à promouvoir l'utilisation de gradients incomplets.

Enfin, notons que la voilure d'un avion est un **système très complexe**. Des disciplines autres que l'aérodynamiques, en particulier le calcul de structure et la thermique jouent également un rôle important dans sa conception. La règlementation aérienne impose aux concepteurs de prendre en compte également des objectifs écoligiques de réduction de bruit (externe et interne), de pollution. La conception d'un avion furtif repose sur une reduction de la trace électromagnétique. Le coût (en euros ou en dollars) et le temps de fabrication, introduisent une autre discipline encore : l'économie de marché. Le "vrai problème" du concepteur industriel d'avion est donc pleinement un problème d'**optimisation multidisciplinaire**, un domaine qui mobilise actuellement un très gros effort de recherche et développement, notamment dans le domaine aéronautique.

3.1.3 Exemples en électromagnétisme

Optimisation de structures rayonnantes : La propagation d'ondes est un problème distribué dans l'espace régi par les équations de Maxwell en régime permanent et formulation harmonique. Orange Labs, anciennement *France Telecom R & D*, installés notamment à la Turbie [4]. ont développé des codes par Eléments Finis pour la simulation de ces équations et leur utilisation en conception d'antennes.

La forme Γ doit être optimisée de manière à ce que le spectre réfléchi en fréquences soit aussi conforme que possible à un gabarit souhaité et que les pertes énergétiques soient aussi faibles que possible,

On pourra examiner quelques images de la thèse de Benoît C
(`https://tel.archives-ouvertes.fr/tel-00429366/`).

Le problème du radar : Le problème du radar est un problème inverse géométrique. Il s'agit de reconstituer la forme Γ de manière à ce que l'onde réfléchie calculée par résolution des équations de Maxwell correspondant à une certaine forme supposée, soit aussi conforme que possible à l'onde réfléchie effectivement mesurée sur le site d'observation.

Remarque importante : notons que dans ces deux problèmes l'optimisation est pilotée par la forme Γ qui est distante de l'observation à partir de laquelle se calcule la fonctionnelle-coût. Il s'agit là d'une différence assez fondamentale avec le cas de l'aérodynamique externe de la sous-section précédente.

3.2 Système distribué soumis à un contrôle frontière

4. `http://www.eurecom.fr/~nikaeinn/files/sympa/Orange_Labs_La_Turbie.pdf`

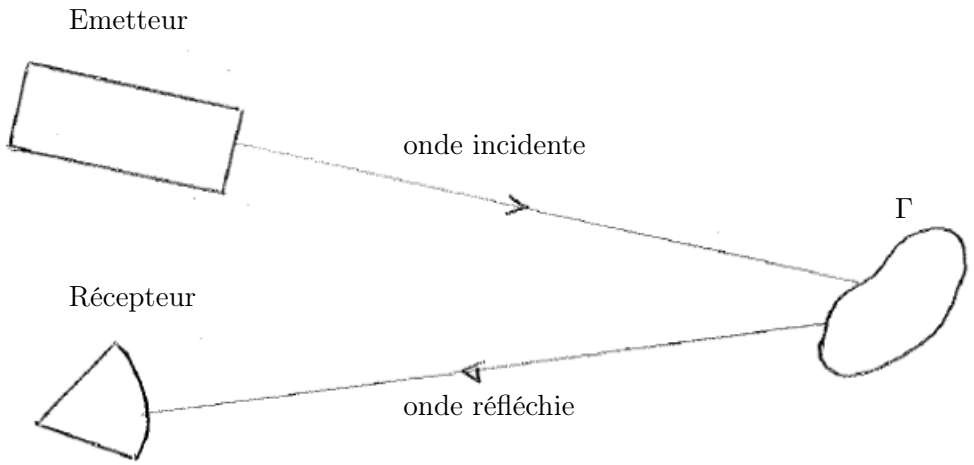


FIGURE 3.8 – Configuration d'émission/réception d'onde électromagnétique : source (onde incidente), observation (onde réfléchie), réflecteur de forme Γ

3.2.1 Exemple en décomposition de domaine

Dans ce paragraphe, on traite un problème de décomposition de domaine [5] en partition parce qu'il fournit un exemple de problème aux limites distribué soumis à un contrôle fonctionnel à une frontière, pour lequel on calcule un gradient de fonctionnelle par la technique de l'**"équation adjointe"**.

On considère une équation modèle de convection-diffusion-production ici non linéaire :

$$u_t + auu_x + buu_y - \varepsilon(u_{xx} + u_{yy}) = \varpi(u)\,,\ (x,y) \in]-1,1[^2 \tag{3.32}$$

L'intégration en temps s'effectue simplement par la méthode d'Euler. Pour un traitement implicite en temps, on suppose connue la solution $u^n(x,y)$ correspondant au temps $t^n = n\Delta t$, et on cherche $u(x,y) = u^{n+1}(x,y)$. Pour cela, on remplace la dérivée en temps par une simple différence finie :

$$u_t \doteq \frac{u - u^n}{\Delta t} \tag{3.33}$$

et on linéarise le terme source de production :

$$\varpi(u) = \varpi\big(u^n(x,y)\big) + \varpi'\big(u^n(x,y)\big)\,(u - u^n) + \dots \tag{3.34}$$

de sorte que la fonction $u(x,y)$ est solution du problème :

$$\boxed{\begin{array}{l} \alpha u + auu_x + buu_y - \varepsilon(u_{xx} + u_{yy}) = f\,,\ (x,y) \in]-1,1[^2 \\ \\ \qquad u(-1,y) = g(y)\,,\ \forall y \in]-1,1[\\ \qquad u_y(x,-1) = u_y(x,1) = 0\,,\ \forall x \in]-1,1[\\ \qquad u_x(1,y) = 0\,,\ \forall y \in]-1,1[\end{array}} \tag{3.35}$$

5. en anglais "DDM" : Domain Decomposition Method

où l'on a posé

$$\alpha = \frac{1}{\Delta t} - \varpi'\big(u^n(x,y)\big)\,,\ f = \frac{u^n(x,y)}{\Delta t} + \varpi\big(u^n(x,y)\big) - \varpi'\big(u^n(x,y)\big)u^n(x,y)\,. \tag{3.36}$$

On se place dans le cas où ce système est bien posé (e.g. $a > 0$; $g > 0$; $\varepsilon > 0$, petit mais suffisamment grand).

Une "méthode de gradient" en partition pour le problème de base, (3.35), consiste à adopter les options suivantes :

- Partition du domaine
$$\Omega =]-1,1[\times]-1,1[\tag{3.37}$$
en deux sous-domaines, par exemple
$$\Omega_1 =]-1,0[\times]-1,1[\,,\ \Omega_2 =]0,1[\times]-1,1[\tag{3.38}$$
et introduction d'une fonction de contrôle $v(y)$ des valeurs de u à l'interface $\gamma_{12} = \{(0,y)\ /\ y \in [-1,1]\}$ de l'axe des y.
- Définition d'une fonctionnelle-coût :
$$\boxed{J(v) = \frac{1}{2}\int_{-1}^{1} \left(u_{1,x} - u_{2,x}\right)^2 (0,y)\,\omega(y)\,dy} \tag{3.39}$$
mesurant la violation de régularité à l'interface suivant le schéma fonctionnel de la FIG. 3.9,

$$\begin{array}{ccccc} v(y) & \longrightarrow & \left\{\begin{array}{l} u_1(x,y) \text{ dans } \Omega_1 \\ u_2(x,y) \text{ dans } \Omega_2 \end{array}\right. & \longrightarrow & J(v) \\ & \searrow & & \nearrow & \\ & & \text{fonctionnelle } J & & \end{array}$$

FIGURE 3.9 – Schéma fonctionnel du critère J

- Calcul de la fonctionnelle J et de son gradient.
- Réactualisation du contrôle $v(y)$ aur la base du gradient et test d'arrêt :
$$J(v) < \epsilon \tag{3.40}$$
où ϵ mesure la tolérance en précision.

Noter que pour une EDP du second-ordre, il est naturel d'exprimer le défaut de raccord à l'interface en terme de sauts de la fonction et de son gradient, et ici seulement du saut de gradient, puisque le problème de départ, ici (3.35), doit plus généralement et plus proprement s'exprimer par une formulation variationnelle dans H^1, omise ici, faisant intervenir u et ∇u. L'utilisation de la fonction de pondération $\omega(y)$ (positive) n'est pas indispensable, mais quelquefois utile au plan numérique.

Par conséquent, les sous-problèmes se formulent respectivement comme suit :

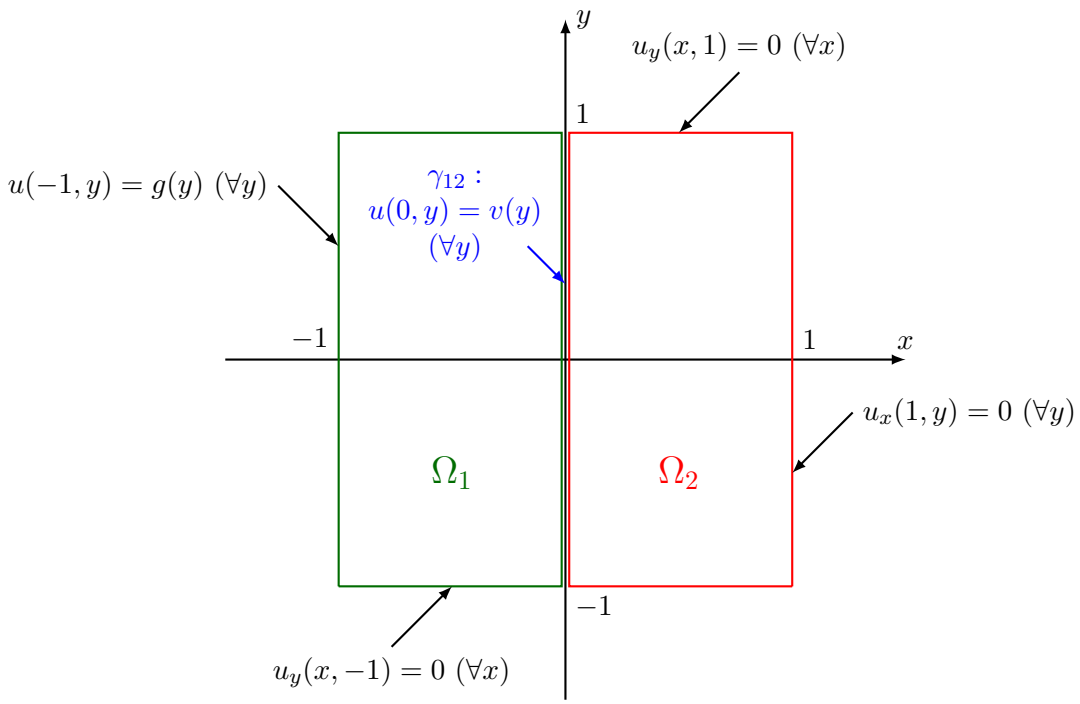


FIGURE 3.10 – Partition de domaine et fonction de contrôle à l'interface

Sous-domaine Ω_1 :

$$\begin{array}{l} \alpha u_1 + a u_1 u_{1,x} + b u_1 u_{1,y} - \varepsilon(u_{1,xx} + u_{1,yy}) = f, \\ (x,y) \in \Omega_1 =]-1,0[\times]-1,1[\\ \\ u_1(-1,y) = g(y)\,,\ \forall y \in]-1,1[\\ u_{1,y}(x,-1) = u_{1,y}(x,1) = 0\,,\ \forall x \in]-1,0[\\ u_1(0,y) = v(y)\,,\ \forall y \in]-1,1[\end{array} \tag{3.41}$$

Sous-domaine Ω_2 :

$$\begin{array}{l} \alpha u_2 + a u_2 u_{2,x} + b u_2 u_{2,y} - \varepsilon(u_{2,xx} + u_{2,yy}) = f, \\ (x,y) \in \Omega_2 =]0,1[\times]-1,1[\\ \\ u_2(0,y) = v(y)\,,\ \forall y \in]-1,1[\\ u_{2,y}(x,-1) = u_{2,y}(x,1) = 0\,,\ \forall x \in]0,1[\\ u_{2,x}(1,y) = 0\,,\ \forall y \in]-1,1[\end{array} \tag{3.42}$$

Dans cet exemple, en x le problème dans le sous-domaine Ω_1 est de type Dirichlet-Dirichlet, et dans le sous-domaine Ω_2 de type Dirichlet-Neumann ; en y, le problème est de type Neumann-Neumann dans les deux sous=domaines.

A l'issue de la résolution (séparée) des sous-problèmes posés dans Ω_1 et Ω_2, on évalue la fonctionnelle-coût $J(v)$ définie en (3.39). On souhaite maintenant identifier le gradient de cette fonctionnelle par rapport à $v(y)$ dans le but de modifier $v(y)$ dans un sens qui réduise cette fonctionnelle. Pour l'identification du gradient, on note δu_1 et δu_2 les variations que subissent les solutions partielles u_1 et u_2 lorsqu'on perturbe le contrôle $v(y)$ de $\delta v(y)$ selon le schéma fonctionnel de la FIG. 3.11.

$$\delta v \longrightarrow \begin{cases} \delta u_1 \text{ dans } \Omega_1 \\ \delta u_2 \text{ dans } \Omega_2 \end{cases} \longrightarrow \delta J(v;\delta v)$$

gradient de fonctionnelle

FIGURE 3.11 – Schéma fonctionnel du gradient

Les variations δu_1 et δu_2 sont les solutions de sous-problèmes linéarisés, qui sont les suivants :

Sous-domaine Ω_1 :

$$\begin{gathered} (\alpha + a u_{1,x} + b u_{1,y})\,\delta u_1 + a u_1 \delta u_{1,x} + b u_1 \delta u_{1,y} = \varepsilon(\delta u_{1,xx} + \delta u_{1,yy})\,, \\ (x,y) \in \Omega_1 =]-1,0[\times]-1,1[\\ \\ \delta u_1(-1,y) = 0\,,\ \forall y \in]-1,1[\\ \delta u_{1,y}(x,-1) = \delta u_{1,y}(x,1) = 0\,,\ \forall x \in]-1,0[\\ \delta u_1(0,y) = \delta v(y)\,,\ \forall y \in]-1,1[\end{gathered} \tag{3.43}$$

Sous-domaine Ω_2 :

$$\begin{gathered} (\alpha + a u_{2,x} + b u_{2,y})\,\delta u_2 + a u_2 \delta u_{2,x} + b u_2 \delta u_{2,y} = \varepsilon(\delta u_{2,xx} + \delta u_{2,yy}\,, \\ (x,y) \in \Omega_2 =]0,1[\times]-1,1[\\ \\ \delta u_2(0,y) = \delta v(y)\,,\ \forall y \in]-1,1[\\ \delta u_{2,y}(x,-1) = \delta u_{2,y}(x,1) = 0\,,\ \forall x \in]0,1[\\ \delta u_{2,x}(1,y) = 0\,,\ \forall y \in]-1,1[\end{gathered} \tag{3.44}$$

Ces deux systèmes constituent une représentation **implicite** des fonctions $\delta u_1(x,y)$ et $\delta u_2(x,y)$ en tant que fonctionnelles de $\delta v(y)$. On en déduit l'expression implicite suivante de la variation de fonctionnelle coût :

$$\delta J = \delta J(u_1,u_2;\delta u_1,\delta u_2) = \int_{-1}^{1} (u_{1,x} - u_{2,x})\ (\delta u_{1,x} - \delta u_{2,x})\,(0,y)\,\omega(y)\,dy \tag{3.45}$$

dont on sépare les contributions des deux sous-domaines :

$$\boxed{\delta J = \int_{-1}^{1} (u_{1,x} - u_{2,x})\ \delta u_{1,x}\, \omega(y)\, dy - \int_{-1}^{1} (u_{1,x} - u_{2,x})\ \delta u_{2,x}\, \omega(y)\, dy} \tag{3.46}$$

Il s'agit maintenant de transformer cette relation en une expression **explicite** de $\delta v(y)$. Pour cela, on utilise la technique classique de l'**"équation adjointe"** constituée des étapes suivantes :

1. Définir une variable adjointe $\mathbf{p}(x, y)$ dont la restriction à Ω_1 (resp. Ω_2) est notée $\mathbf{p}_1$ (resp. $\mathbf{p}_2$).
2. À partir de l'EDP linéaire satisfaite par δu_1 (resp. δu_2), (3.43) (resp. (3.44)), écrite sous la forme $E_1 = 0$ (resp. $E_2 = 0$), effectuer le produit scalaire, $(\mathbf{p}_1, E_1) = 0$ (resp. $(\mathbf{p}_2, E_2) = 0$) (c'est-à-dire multiplier et intégrer dans le sous-domaine). On obtient une identité valable quel que soit le choix ultérieur de $\mathbf{p}_1$ (resp. $\mathbf{p}_2$).
3. Transformer l'identité obtenue par intégrations par parties successives ; simplifier en vertu des conditions aux limites homogènes satisfaites par δu_1 (resp. δu_2).
4. Au vu de (3.46), poser le système EDP (EDP linéaire + conditions aux limites) en $\mathbf{p}_1$ (resp. $\mathbf{p}_2$) de manière à reproduire dans l'identité exactement les mêmes termes implicites que ceux qui apparaissent dans la contribution à δJ du sous-domaine Ω_1 (resp. Ω_2).
5. Soustraire l'identité à δJ de manière à retenir seulement les termes explicites en δv.

Nous allons d'abord réaliser ces étapes pour le sous-domaine Ω_1 en omettant momentanément l'indice inférieur $_1$ pour alléger l'écriture et en adoptant la notation

$$\iint_{\Omega_1} (\dots) \stackrel{\text{def}}{=} \int_{-1}^{0}\int_{-1}^{1} (\dots)\, dx\, dy \tag{3.47}$$

Pour le sous-domaine Ω_1, on part donc de l'équation suivante :

$$\boxed{\iint_{\Omega_1} \mathbf{p}(x, y) \Big[(\alpha + a u_x + b u_y)\, \delta u + a u \delta u_x + b u \delta u_y - \varepsilon\, (\delta u_{xx} + \delta u_{yy}) \Big] = 0} \tag{3.48}$$

qui est une identité car elle est satisfaite quel que soit le choix que l'on fera ultérieurement (étape 4) de la fonction adjointe $\mathbf{p}(x, y)$. Avant ce choix, simplifions par des intégrations par parties pour éliminer les dérivées partielles par rapport à x ou y portant sur la variable d'état u. On a :

$$\begin{aligned}
\iint_{\Omega_1} \mathbf{p} a u \delta u_x &= \int_{-1}^{1} \left(\int_{-1}^{0} a u \mathbf{p} \delta u_x\, dx \right) dy \\
&= \int_{-1}^{1} [a u \mathbf{p} \delta u]_{x=-1}^{x=0}\, dy - \iint_{\Omega_1} (a u \mathbf{p})_x\, \delta u \\
&= \int_{-1}^{1} a u \mathbf{p}(0, y) \delta v(y)\, dy - \iint_{\Omega_1} (a u \mathbf{p})_x\, \delta u
\end{aligned} \tag{3.49}$$

où l'on a utilisé le fait que δu est nul sur le bord où u est imposé ($x = -1$) et égal à δv sur celui où la fonction u est "contrôlée" ($x = 0$). Symétriquement,

$$\begin{aligned}
\iint_{\Omega_1} \mathbf{p} b u \delta u_y &= \int_{-1}^{0} \left(\int_{-1}^{1} b u \mathbf{p} \delta u_y \, dy \right) dx \\
&= \int_{-1}^{0} \left[b u \mathbf{p} \delta u \right]_{y=-1}^{y=1} dx - \iint_{\Omega_1} \left(b u \mathbf{p} \right)_y \delta u \\
&= \int_{-1}^{0} \left[b u \mathbf{p} \delta u(x,1) - b u \mathbf{p} \delta u(x,-1) \right] dx - \iint_{\Omega_1} \left(b u \mathbf{p} \right)_y \delta u
\end{aligned} \tag{3.50}$$

Pour les termes de diffusion, il faut faire deux intégrations par parties. En particulier :

$$\begin{aligned}
\iint_{\Omega_1} \mathbf{p} \delta u_{xx} &= \int_{-1}^{1} \left(\int_{-1}^{0} \mathbf{p} \delta u_{xx} \, dx \right) dy \\
&= \int_{-1}^{1} \left[\mathbf{p} \delta u_x \right]_{x=-1}^{x=0} dy - \iint_{\Omega_1} \mathbf{p}_x \delta u_x \\
&= \int_{-1}^{1} \left[\mathbf{p} \delta u_x(0,y) - \mathbf{p} \delta u_x(-1,y) \right] dy - \int_{-1}^{1} \left(\int_{-1}^{0} \mathbf{p}_x \delta u_x \, dx \right) dy \\
&= \int_{-1}^{1} \left[\mathbf{p} \delta u_x(0,y) - \mathbf{p} \delta u_x(-1,y) \right] dy - \int_{-1}^{1} \left[\mathbf{p}_x \delta u \right]_{x=-1}^{x=0} dy \\
&+ \iint_{\Omega_1} \mathbf{p}_{xx} \delta u \\
&= \int_{-1}^{1} \left[\mathbf{p} \delta u_x(0,y) - \mathbf{p} \delta u_x(-1,y) - \mathbf{p}_x(0,y) \delta v(y) \right] dy \\
&+ \iint_{\Omega_1} \mathbf{p}_{xx} \delta u
\end{aligned} \tag{3.51}$$

Symétriquement,

$$\begin{aligned}
\iint_{\Omega_1} \mathbf{p} \delta u_{yy} &= \int_{-1}^{0} \left(\int_{-1}^{1} \mathbf{p} \delta u_{yy} \, dy \right) dx \\
&= \int_{-1}^{0} \left[\mathbf{p} \delta u_y \right]_{y=-1}^{y=1} dx - \iint_{\Omega_1} \mathbf{p}_y \delta u_y \\
&= - \int_{-1}^{0} \left(\int_{-1}^{1} \mathbf{p}_y \delta u_y \, dy \right) dx \\
&= - \int_{-1}^{0} \left[\mathbf{p}_y \delta u \right]_{y=-1}^{y=1} dx + \iint_{\Omega_1} \mathbf{p}_{yy} \delta u \\
&= \int_{-1}^{0} \left[-\mathbf{p}_y \delta u(x,1) + \mathbf{p}_y \delta u(x,-1) \right] dx + \iint_{\Omega_1} \mathbf{p}_{yy} \delta u
\end{aligned} \tag{3.52}$$

Noter que ces intégrations par parties font apparaître les opérateurs adjoints de ceux qui apparaissaient dans le système linéarisé (3.43).

Les équations (3.49)-(3.50)-(3.51)-(3.52) permettent de transformer (3.48) en l'expression

suivante :

$$\begin{aligned}\iint_{\Omega_1} \left[(\alpha + au_x + bu_y)\,\mathbf{p} - (au\mathbf{p})_x - (bu\mathbf{p})_y - \varepsilon\,(\mathbf{p}_{xx} + \mathbf{p}_{yy})\right]\,\delta u \\ + \int_{-1}^{1} (au\mathbf{p} + \varepsilon\mathbf{p}_x)\,(0,y)\,\delta v(y)\,dy \\ + \int_{-1}^{0} \left[(bu\mathbf{p} + \varepsilon\mathbf{p}_y)\,\delta u(x,1) - (bu\mathbf{p} + \varepsilon\mathbf{p}_y)\,\delta u(x,-1)\right]\,dx \\ + \int_{-1}^{1} \left[-\varepsilon\mathbf{p}\delta u_x(0,y) + \varepsilon\mathbf{p}\delta u_x(-1,y)\right]\,dy \\ = 0\,,\ \forall\,\mathbf{p}.\end{aligned} \tag{3.53}$$

Cette équation fait intervenir trois types de termes : une intégrale (la première) étendue au domaine dont l'intégrande fait apparaître le facteur δu et aucune de ses dérivées partielles, une intégrale de bord (la seconde) faisant intervenir explicitement la perturbation de contrôle, et deux intégrales de bord faisant intervenir des perturbations *a priori* inconnues de l'état. Puisqu'il s'agit d'une identité, on choisit $\mathbf{p} = \mathbf{p}_1$ (étape 4) de manière à éliminer la première et la troisième intégrales, et à donner à la quatrième la forme de la partie de δJ qui est proportionnelle à $\delta u_x(0,y) = \delta u_{1,x}(0,y)$. Plus précisément, on pose l'**"équation adjointe"** suivante :

$$\boxed{\begin{array}{c}(\alpha + au_x + bu_y)\,\mathbf{p}_1 - (au\mathbf{p}_1)_x - (bu\mathbf{p}_1)_y = \varepsilon(\mathbf{p}_{1,xx} + \mathbf{p}_{1,yy})\,, \\ (x,y) \in \Omega_1 =]-1,0[\times]-1,1[\\ \\ \begin{array}{l}\mathbf{p}_1(-1,y) = 0\,,\ \forall y \in]-1,1[\\ \left[bu_1\mathbf{p}_1 + \varepsilon\mathbf{p}_{1,y}\right](x,-1) = 0\,,\ \forall x \in]-1,0[\\ \left[bu_1\mathbf{p}_1 + \varepsilon\mathbf{p}_{1,y}\right](x,1) = 0\,,\ \forall x \in]-1,0[\\ \mathbf{p}_1(0,y) = (u_{1,x} - u_{2,x})\,(0,y)\,\omega(y)\,,\ \forall y \in]-1,1[\end{array}\end{array}} \tag{3.54}$$

de sorte que (3.53) fournit le résultat partiel suivant :

$$\boxed{\begin{array}{l}\varepsilon \displaystyle\int_{-1}^{1} (u_{1,x} - u_{2,x})\ \delta u_{1,x}(0,y)\,\omega(y)\,dy \\ = \displaystyle\int_{-1}^{1} (au\mathbf{p}_1 + \varepsilon\mathbf{p}_{1,x})\,(0,y)\,\delta v(y)\,dy\end{array}} \tag{3.55}$$

On procède ensuite de manière analogue pour le sous-domaine Ω_2 pour lequel les conditions aux limites ne sont pas exactement les mêmes. Après quelques calculs de même nature, on obtient les résultats suivants (en omettant, pour alléger l'écriture, l'indice inférieur $_2$ portant sur $\mathbf{p}$ dans les 5 équations suivantes) :

$$\iint_{\Omega_2} \mathbf{p}au\delta u_x = \int_{-1}^{1} \left[(au\mathbf{p}\delta u(1,y)) - au\mathbf{p}(0,y)\,\delta v(y)\right]\,dy - \iint_{\Omega_2} (au\mathbf{p})_x\ \delta u. \tag{3.56}$$

Symétriquement,

$$\iint_{\Omega_2} \mathbf{p} bu \delta u_y = \int_0^1 \left[bu\mathbf{p}\delta u(x,1) - bu\mathbf{p}\delta u(x,-1) \right] dx - \iint_{\Omega_2} (bu\mathbf{p})_y \, \delta u. \tag{3.57}$$

En outre,

$$\iint_{\Omega_2} \mathbf{p}\delta u_{xx} = \int_{-1}^1 \left[-\mathbf{p}\delta u_x(0,y) - \mathbf{p}_x \delta u(1,y) + \mathbf{p}_x(0,y)\delta v(y) \right] dy + \iint_{\Omega_2} \mathbf{p}_{xx}\delta u. \tag{3.58}$$

Symétriquement,

$$\iint_{\Omega_2} \mathbf{p}\delta u_{yy} = \int_0^1 \left[-\mathbf{p}_y \delta u(x,1) + \mathbf{p}_y \delta u(x,-1) \right] dx + \iint_{\Omega_1} \mathbf{p}_{yy}\delta u. \tag{3.59}$$

L'identité satisfaite par $\mathbf{p}$ est donc la suivante :

$$\begin{aligned} \iint_{\Omega_2} \left[(\alpha + au_x + bu_y)\,\mathbf{p} - (au\mathbf{p})_x - (bu\mathbf{p})_y - \varepsilon\left(\mathbf{p}_{xx} + \mathbf{p}_{yy}\right) \right] \delta u \\ - \int_{-1}^1 \left(au\mathbf{p} + \varepsilon \mathbf{p}_x\right)(0,y)\,\delta v(y)\,dy \\ + \int_0^1 \left[\left(bu\mathbf{p} + \varepsilon\mathbf{p}_y\right)\delta u(x,1) - \left(bu\mathbf{p} + \varepsilon\mathbf{p}_y\right)\delta u(x,-1) \right] dx \\ + \int_{-1}^1 \left[\left(au\mathbf{p} + \varepsilon\mathbf{p}_x\right)\delta u(1,y) + \varepsilon\mathbf{p}\delta u_x(0,y) \right] dy \\ = 0\,,\ \forall\, \mathbf{p} \end{aligned} \tag{3.60}$$

ce qui conduit à poser l'**"équation adjointe"** suivante pour $\mathbf{p} = \mathbf{p}_2$:

$$\boxed{\begin{array}{c} (\alpha + au_x + bu_y)\,\mathbf{p}_2 - (au\mathbf{p}_2)_x - (bu\mathbf{p}_2)_y = \varepsilon(\mathbf{p}_{2,xx} + \mathbf{p}_{2,yy})\,, \\ (x,y) \in \Omega_2 =]0,1[\times]-1,1[\\ \\ \left[au_2\mathbf{p}_2 + \varepsilon\mathbf{p}_{2,x}\right](1,y) = 0\,,\ \forall y \in]-1,1[\\ \left[bu_2\mathbf{p}_2 + \varepsilon\mathbf{p}_{2,y}\right](x,-1) = 0\,,\ \forall x \in]0,1[\\ \left[bu_2\mathbf{p}_2 + \varepsilon\mathbf{p}_{2,y}\right](x,1) = 0\,,\ \forall x \in]0,1[\\ \mathbf{p}_2(0,y) = (u_{1,x} - u_{2,x})\,(0,y)\,\omega(y)\,,\ \forall y \in]-1,1[\end{array}} \tag{3.61}$$

de sorte que (3.60) fournit le résultat suivant :

$$\boxed{\begin{aligned} \varepsilon \int_{-1}^1 (u_{1,x} - u_{2,x})\,(0,y)\,\delta u_{2,x}(0,y)\,\omega(y)\,dy \\ = \int_{-1}^1 \left(au\mathbf{p}_2 + \varepsilon\mathbf{p}_{2,x}\right)(0,y)\,\delta v(y)\,dy \end{aligned}} \tag{3.62}$$

Finalement, en combinant (3.55) et (3.62), on aboutit au gradient recherché :

$$\boxed{\delta J = \int_{-1}^{1} G(y)\,\delta v(y)\,dy} \tag{3.63}$$

où l'on a posé :

$$\boxed{G(y) = (\mathbf{p}_{1,x} - \mathbf{p}_{2,x})\,(0,y)} \tag{3.64}$$

En résumé, les étapes de la méthode sont les suivantes :

1. *Choix initial de la fonction d'interface $v(y)$*
2. *Résolution de l'équation d'état sur les différents sous-domaines, (3.41)-(3.42), et calcul de la fonctionnelle de coût $J(v)$: arrêt si $J(v) < \epsilon$, sinon :*
3. *Résolution de l'équation adjointe sur les différents sous-domaines, (3.54)-(3.61), et calcul du gradient de la fonctionnelle de coût $G(y)$*
4. *Modification de la fonction d'interface*

$$v(y) \longrightarrow v(y) + \delta v(y) \tag{3.65}$$

et retour à l'étape 2.

En ce qui concerne le choix de la perturbation $\delta v(y)$, plusieurs options sont possibles :

4.1 Un pas dans la direction opposée au gradient :

$$\delta v(y) = -\rho\,\frac{J}{\int_{-1}^{1} G^2(y)\,dy}\,G(y) \quad (0 < \rho < 1) \tag{3.66}$$

de sorte que si ρ est suffisamment petit : $\delta J \approx -\rho J$.

4.2 Une optimisation unidimensionnelle :
Une fois l'état u, le coût J_0 et le gradient $G(y)$ calculés, on perturbe le contrôle dans la direction opposée au gradient, comme ci-dessus, avec deux valeurs différentes ρ_1 et ρ_2 du pas, ce qui fournit deux valeurs supplémentaires du coût J_1 et J_2 (après deux nouvelles résolutions de l'équation d'état). On choisit ρ comme l'abscisse du minimum de la parabole qui passe par les points $(0, J_0)$, (ρ_1, J_1) et (ρ_2, J_2).

4.3 Gradients conjugués :
Dans la méthode précédente on remplace le gradient par une direction de gradient conjugué [6]

4.4 Autres méthodes à base de gradient : méthodes de quasi-Newton [7], GMRES [8], Optimisation quadratique successive (SQP) [9], méthode de Broyden-Fletcher-Goldfarb-Shanno (BFGS) [10], etc.

6. `https://en.wikipedia.org/wiki/Conjugate_gradient_method` (existe en fançais)
7. `https://en.wikipedia.org/wiki/Quasi-Newton_method` (existe en français)
8. `http://en.wikipedia.org/wiki/Gmres` (existe en français)
9. `https://fr.wikipedia.org/wiki/Optimisation_quadratique_successive` (existe en anglais)
10. `https://fr.wikipedia.org/wiki/BFGS` (existe en anglais)

REMARQUES FINALES :

(1) On a choisi dans l'exemple, de régulariser la dérivée normale à l'interface par le choix de l'état u comme fonction de contrôle. En conséquence, on a obtenu un gradient égal au saut de cette dérivée à travers l'interface. On peut à l'inverse, régulariser la fonction d'état (continuité) par le choix de la dérivée normale comme fonction de contrôle (pour autant que les problèmes correspondants aux sous-domaines restent bien posés, voir exercice 12). Dans ce cas, l'expression du gradient contient le saut de u à travers l'interface comme facteur.

Exercice 12 (Contrôle de la continuité par la dérivée normale)
On considère le problème suivant qui diffère du précédent par le remplacement de la condition de Neumann sur le bord $y = -1$ par une condition de type Dirichlet :

$$\begin{array}{l} \alpha u + auu_x + buu_y - \varepsilon(u_{xx} + u_{yy}) = f\,,\ (x,y) \in]-1,1[^2 \\ \\ u(-1,y) = g(y)\,,\ \forall y \in]-1,1[\\ u(x,-1) = h(\mathbf{x})\,,\ \forall x \in]-1,1[\\ u_y(x,1) = 0\,,\ \forall x \in]-1,1[\\ u_x(1,y) = 0\,,\ \forall y \in]-1,1[\end{array} \tag{3.67}$$

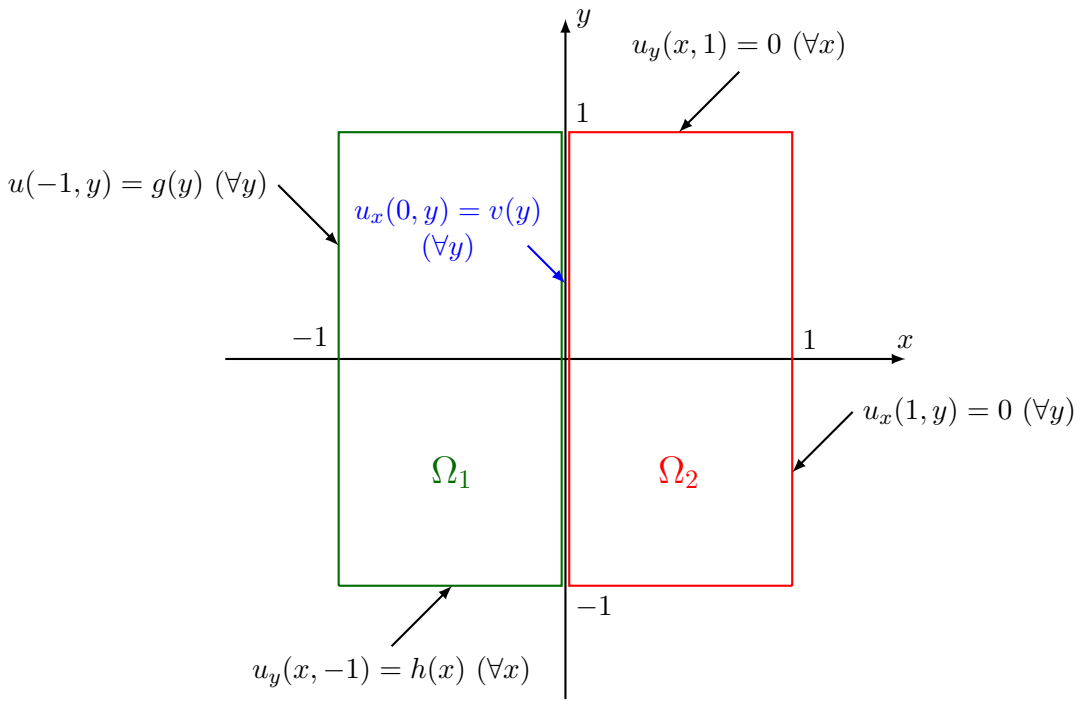


FIGURE 3.12 – Partition et contrôle par la dérivée normale (exercice 12)

On souhaite résoudre ce problème par partition du domaine conformément à la FIG. *3.12*

et contrôle par la dérivée normale à l'interface ($x = 0$) :

$$v(y) = u_x(0, y) \quad (-1 \leq y \leq 1) \tag{3.68}$$

Reformuler les sous-problèmes non linéaires, la fonctionnelle-coût, sa première variation, les sous-problèmes linéarisés et les équations adjointes. En déduire l'expression du gradient.

On peut imaginer des "méthodes adaptatives" dans lesquelles on choisirait localement le long de l'interface la "bonne variable" (ou son gradient) à contrôler.

(2) **Moindres carrés**

Dans le cas d'une décomposition en sous-domaines ayant un recouvrement dont on note γ_1 et γ_2 les bords, une approche alternative consiste à régulariser à convergence en contrôlant u (ou toute autre variable plus pertinente) sur l'interface γ_1 (par la fonction v_1) et sur l'interface γ_2 (par la fonction v_2) et en minimisant la fonctionnelle-coût suivante :

$$J(v_1, v_2) = \tfrac{1}{2} \iint_{\Omega_1 \cap \Omega_2} (u_1 - u_2)^2 \, \omega(x, y) \, dx \, dy \quad (\omega(x, y) \geq 0) \tag{3.69}$$

(voir FIG. 3.13).

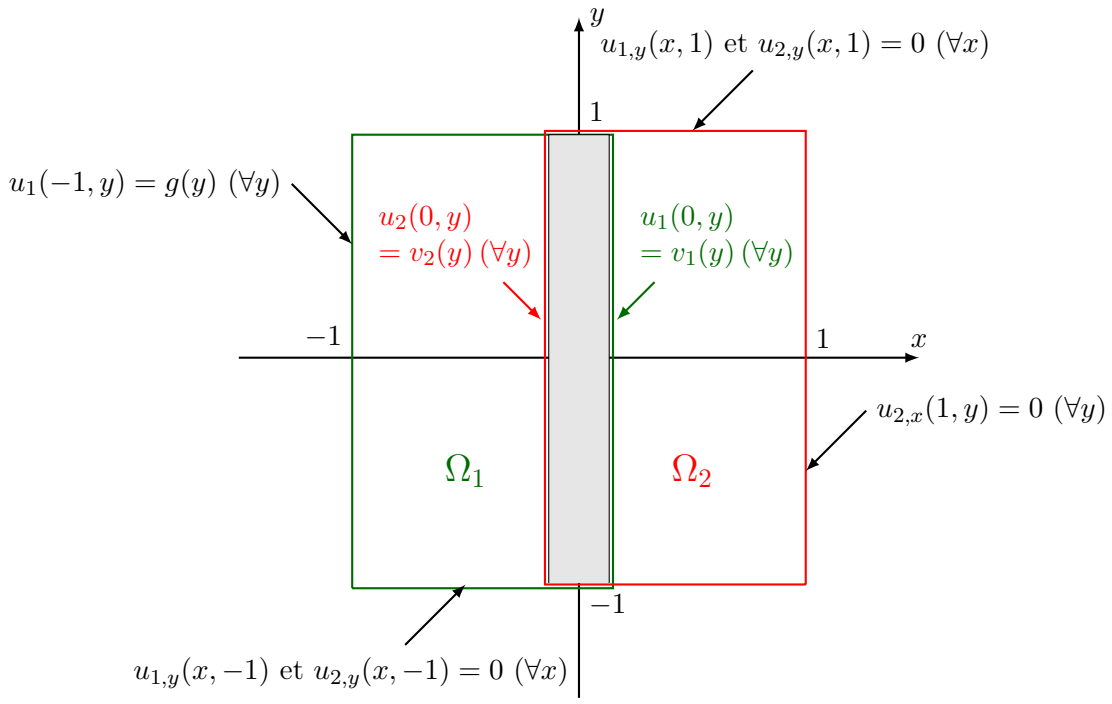


FIGURE 3.13 – Décomposition de domaine avec recouvrement $\Omega_1 \cap \Omega_2 \neq \emptyset$ (en grisé) et fonctions de contrôle au bord du recouvrement : $v_1(y)$ portant sur $u_1 = u_{/\Omega_1}$ et $v_2(y)$ portant sur $u_2 = u_{/\Omega_2}$.

Exercice 13 (Moindres carrés)
Identifier le gradient de la fonctionnelle-coût de la méthode des moindres carrés.

(3) Le mérite principal de ces méthodes de contrôle semble résider davantage dans leur approche rationnelle pour identifier des conditions d'interface viables et dans le fait qu'elles ont permis de résoudre effectivement certains problèmes de grande taille inaccessibles globalement, que dans leur potentialité à réduire le coût global de la résolution par le biais notamment du parallélisme.

3.2.2 Alternative spécifique au laplacien : formule de Green

Pour le cas du laplacien certaines formules peuvent s'établir rapidement à partir d'une formule de Green que l'on rappelle ici d'abord.

Soit $\vec{\phi}$ est un champ de vecteurs continûment différentiable. Le théorème de la divergence s'écrit :

$$\iint_{\Omega} \nabla.\vec{\phi} = \int_{\Gamma} \vec{\phi}.\vec{n}\, d\Gamma \tag{3.70}$$

où $\vec{n}$ est le vecteur unitaire normal à Γ orienté vers l'extérieur. Soit p et q deux fonctions deux fois continûment différentiables. En appliquant le théorème de la divergence au champ $\vec{\phi} = q\overrightarrow{\nabla p}$, il vient :

$$\iint_{\Omega} \nabla.\left(q\overrightarrow{\nabla p}\right)\, d\Omega = \int_{\Gamma} qp_n\, d\Gamma \tag{3.71}$$

où $p_n = \partial p/\partial n$ est la dérivée normale. En développant, il vient :

$$\iint_{\Omega} \left(\overrightarrow{\nabla q}.\overrightarrow{\nabla p} + q\Delta p\right)\, d\Omega = \int_{\Gamma} qp_n\, d\Gamma \tag{3.72}$$

Symétriquement :

$$\iint_{\Omega} \left(\overrightarrow{\nabla p}.\overrightarrow{\nabla q} + p\Delta q\right)\, d\Omega = \int_{\Gamma} pq_n\, d\Gamma \tag{3.73}$$

D'où :

$$\iint_{\Omega} \big(q\Delta p - p\Delta q\big)\, d\Omega = \int_{\Gamma} \big(qp_n - pq_n\big)\, d\Gamma \tag{3.74}$$

(Deuxième identité de Green ; cf : `https://en.wikipedia.org/wiki/Green%27s_identities`.)

Considérons le cas d'un laplacien contrôlé sur tout le bord :

$$\begin{cases} -\Delta u = f & (\Omega) \\ u = v & (\Gamma = \partial\Omega) \end{cases} \tag{3.75}$$

(u et v scalaires), et supposons pour simplifier que la fonctionnelle à minimiser ne dépende que de l'état $u(\mathbf{x})$:

$$J(v) = \iint_{\Omega} \mathcal{L}\big(x; u(\mathbf{x})\big)\, d\Omega + \int_{\Gamma} \Phi\big(x; u(\mathbf{x})\big)\, d\Gamma \tag{3.76}$$

Il vient :

$$\delta J = \iint_{\Omega} \mathcal{L}_u \delta u\, d\Omega + \int_{\Gamma} \Phi_u \delta u\, d\Gamma = \iint_{\Omega} \underbrace{\mathcal{L}_u \delta u}_{\text{implicite}}\, d\Omega + \int_{\Gamma} \underbrace{\Phi_u \delta v}_{\text{explicite}}\, d\Gamma \tag{3.77}$$

où δu est la solution du système linéarisé :

$$\begin{cases} -\Delta(\delta u) = 0 & (\Omega) \\ \delta u = \delta v & (\Gamma = \partial\Omega) \end{cases} \tag{3.78}$$

En appliquant la formule de Green au cas où $p = \mathbf{p}$ et $q = \delta u$, il vient :

$$\iint_\Omega \big(\delta u\, \Delta \mathbf{p} - \mathbf{p}\, \underbrace{\Delta(\delta u)}_{=0}\big)\, d\Omega = \int_\Gamma \big(\underbrace{\delta u}_{\delta v}\, \mathbf{p}_n - \mathbf{p}(\delta u)_n\big)\, d\Gamma \tag{3.79}$$

On pose donc :

$$\begin{cases} \Delta \mathbf{p} = \mathcal{L}_u & (\Omega) \\ \mathbf{p} = 0 & (\Gamma) \end{cases} \tag{3.80}$$

et on obtient :

$$\delta J = \int_\Gamma \big(\mathbf{p}_n + \Phi_u\big)\delta v\, d\Gamma\,. \tag{3.81}$$

3.3 Système distribué soumis à un contrôle distribué

Dans cette section, on traite un exemple d'application de la technique de l'équation adjointe à un problème distribué hyperbolique soumis à un contrôle distribué.

3.3.1 Exemple : Génération d'un maillage plan structuré orthogonal par le solveur hyperbolique d'orthogonalité-volume

On considère le contexte de la génération d'un "maillage en O" plan par la construction d'une carte

$$\mathbf{x}(\xi, \eta) = \begin{pmatrix} x(\xi, \eta) \\ y(\xi, \eta) \end{pmatrix} \tag{3.82}$$

qui transforme un quadrillage uniforme ($\Delta\xi = \Delta\eta = 1$) du carré $D = \{\, (\xi, \eta) \in [0, \xi_{\max}] \times [0, \eta_{\max}] \,\}$ en un maillage Ω_h dans une couronne dont le bord intérieur Γ_0 est spécifié et le bord extérieur Γ_1 est calculé (voir FIG. 3.14).

Pour cela, dans une procédure due à Steger [11], on impose à la carte de satisfaire le système suivant d'"orthogonalité-volume" :

$$\begin{cases} x_\xi x_\eta + y_\xi y_\eta = 0 \\ x_\xi y_\eta - y_\xi x_\eta = \mathcal{A}(\xi, \eta) \end{cases} \tag{3.83}$$

où les quantités indicées par ξ ou η sont des dérivées premières. La première équation traduit l'orthogonalité des vecteurs $\frac{\partial \mathbf{x}(\xi,\eta)}{\partial \xi}$ et $\frac{\partial \mathbf{x}(\xi,\eta)}{\partial \eta}$. La deuxième traduit la spécification de l'aire du quadrangle dont ces vecteurs sont des côtés adjacents. La terminologie provient de la version tridimensionnelle de la méthode dans la laquelle on spécifie le volume des cellules.

La coordonnée curviligne η est telle que $\eta = 0$ correspond à la frontière Γ_0, et $\eta = \eta_{\max}$ au bord extérieur Γ_1 du maillage. La coordonnée orthogonale ξ est de type "angle-polaire" de sorte que $\xi = 0$ et $\xi = \xi_{\max}$ correspondent à une même ligne de coupure, généralement non rectiligne, entre Γ_0 et Γ_1.

En résolvant (3.83) par rapport à x_η et y_η, il vient :

$$x_\eta = \mathcal{A}\frac{v}{u^2 + v^2}\,, \quad y_\eta = -\mathcal{A}\frac{u}{u^2 + v^2} \tag{3.84}$$

11. Generation of Body-Fitted Coordinates Using Hyperbolic Partial Differential Equations, Joseph L. Steger and Denny S. Chaussee, SIAM J. Sci. and Stat. Comput. 1, pp. 431-437, 1980

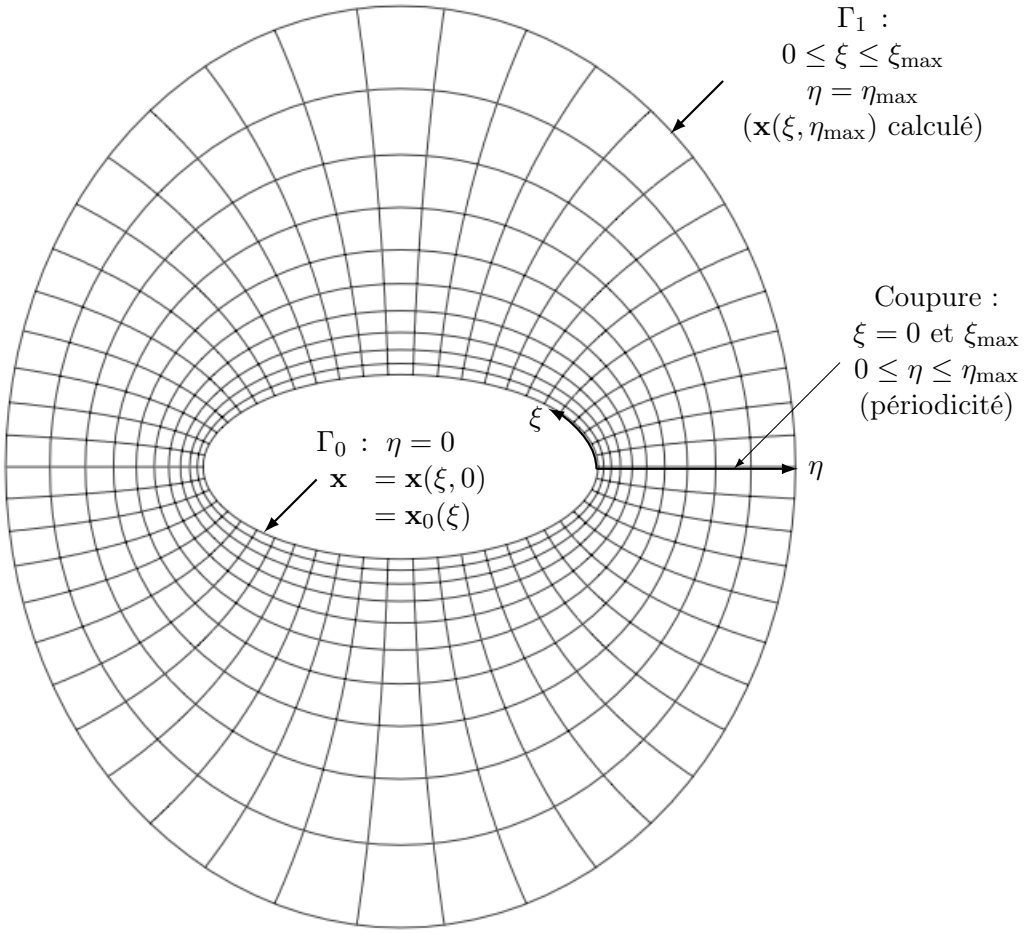


FIGURE 3.14 – Maillage orthogonal généré par intégration du système hyperbolique d'orthogonalité-volume, à partir du bord Γ_0 ($\eta = 0$; $\mathbf{x}(\xi, 0) = \mathbf{x}_0(\xi)$) ; les aires des cellules sont spécifiées par la donnée de la fonction $\mathcal{A}(\xi, \eta)$; le bord extérieur atteint, Γ_1, résulte de cette intégration.

où l'on a posé :

$$u = x_\xi \,, \quad v = y_\xi. \tag{3.85}$$

Ceci permet de réécrire le système (3.83) comme suit :

$$\boxed{\mathbf{x}_\eta = \mathcal{A}(\xi, \eta)\phi(\mathbf{x}_\xi)} \tag{3.86}$$

où la fonction "vecteur de flux", ϕ, admet la définition suivante :

$$\phi(\mathbf{x}_\xi) = \phi(u, v) = \frac{1}{u^2 + v^2} \begin{pmatrix} v \\ -u \end{pmatrix} \tag{3.87}$$

Cette fonction admet la jacobienne suivante :

$$\phi'(\mathbf{x}_\xi) = \frac{\partial \phi(\mathbf{x}_\xi)}{\partial \mathbf{x}_\xi} = \begin{pmatrix} \phi_{1,u} & \phi_{1,v} \\ \phi_{2,u} & \phi_{2,v} \end{pmatrix} = \frac{1}{(u^2+v^2)^2} \begin{pmatrix} 2uv & v^2-u^2 \\ v^2-u^2 & -2uv \end{pmatrix} \tag{3.88}$$

Cette matrice 2×2 est réelle-symétrique (et de trace nulle). Ses valeurs propres sont donc réelles (et opposées l'une de l'autre, $\pm 1/(u^2+v^2)$). **Le système (3.86) est donc hyperbolique**, ce qui légitime sa possible résolution par avancement dans la direction de η croissant, par empilement de couches successives à partir de la frontière Γ_0 ($\eta = 0$) où les points sont spécifiés, jusqu'à la frontière Γ_1 ($\eta = \eta_{\max}$) qui résulte du calcul. La spécification prend donc la forme d'une condition initiale :

$$\boxed{\mathbf{x}(\xi, 0) = \mathbf{x}_0(\xi) \quad (\forall \xi)} \tag{3.89}$$

On s'est placé dans un cas de maillage en forme de couronne qui entoure la frontière fermée Γ_0. On dit qu'il s'agit d'un "maillage en O". Dans ce cas, le système (3.86) est soumis à la condition suivante de périodicité en ξ :

$$\boxed{\mathbf{x}(0, \eta) = \mathbf{x}(\xi_{\max}, \eta) \quad (\forall \eta)} \tag{3.90}$$

En conclusion, dans cette procédure, la carte de maillage $\mathbf{x}(\xi, \eta)$ est la solution du **système hyperbolique** défini par :

- — équation d'état : (3.86)
- — condition initiale : (3.89)
- — conditions aux limites (de périodicité) : (3.90).

Ce système d'EDP est soumis au **contrôle distribué** $\mathcal{A}(\xi, \eta)$. Dans ce qui suit, on notera :

$$\mathcal{B}(\xi, \eta) = \mathcal{A}(\xi, \eta)\phi'(\mathbf{x}_\xi) = \frac{\mathcal{A}(\xi, \eta)}{(u^2+v^2)^2} \begin{pmatrix} 2uv & v^2-u^2 \\ v^2-u^2 & -2uv \end{pmatrix} \tag{3.91}$$

la jacobienne de ce système.

3.3.2 Adaptation du maillage par contrôle de la distribution des aires

Dans l'approche de Steger, la spécification initiale des aires est un problème technique délicat. Steger a notamment proposé de procéder en deux étapes. Dans la première étape, on construit par un moyen *ad hoc* un maillage qui n'est pas orthogonal mais possède la structure souhaitée et recouvre la zone d'intérêt. On en calcule les aires, puis on se sert de cette distribution pour intégrer le système hyperbolique dans la deuxième étape.

La technique de Steger présente notamment les avantages suivants :

Régularité et orthogonalité : si la fonction spécifiée $\mathcal{A}(\xi, \eta)$ est régulière, le maillage généré possède la régularité de la solution d'une EDP hyperbolique, ici sous forme quasi-linéaire, à coefficients réguliers. Cette régularité se vérifie en pratique même à proximité d'une singularité de bord, comme un point anguleux, grâce à la nécessaire viscosité artificielle du schéma numérique. Les caractéristiques des cellules (aire, angles, etc) évoluent continûment en espace. Par ailleurs, l'orthogonalité introduit toujours

des simplifications algorithmiques dans le traitement des éléments de métrique des schémas de type différences finies, et presque toujours des gains de précision formels ou quantitatifs.

Contrôle des mailles : le maillage a pour fonction de permettre le calcul d'un phénomène physique dans le domaine extérieur au bord Γ_0, dont le voisinage constitue la zone physiquement primordiale. Le choix d'un solveur hyperbolique, par opposition à la génération par solveur elliptique, permet le contrôle des mailles à proximité de ce bord. A l'inverse, loin de ce bord, on peut aisément imposer une expansion rapide de la taille des cellules à mesure que η croît (typiquement exponentielle).

Extension au 3D efficace : l'extension au 3D de cette technique est très efficace en temps calcul parce que le maillage est construit en un seul passage, par avancement en couches successives de $\eta = 0$ à $\eta_{\max}$ ("méthode de front"). Par cette caractéristique, la procédure contraste avec d'autres, fournissant également un maillage régulier par résolution d'une EDP définissant une carte, lorsque cette EDP est elliptique. En effet, dans la variante elliptique, si la carte a la régularité des fonctions harmoniques, elle s'obtient par résolution itérative d'un grand système couplé dont le nombres d'inconnues est égal au nombre de cellules quadrangulaires du maillage.

L'inconvénient principal de cette méthode réside dans le fait que le bord extérieur Γ_1 ne peut être spécifié, puisqu'il résulte de l'intégration du système. Cet inconvénient est plus ou moins gênant suivant le type des simulation physique envisagée. En principe, la méthode de Steger est conçue pour résoudre des problèmes extérieurs où la frontière Γ_1 n'est qu'une troncature de l'infini, où s'applique une condition de bord connue. C'est le cas notamment en aérodynamique externe, en écoulement transsonique autour d'une configuration, où à l'infini l'écoulement est constant. Par contre, la qualité du maillage près du bord extérieur du domaine de calcul est critique dans certaines simulations ; par exemple :

— en simulation d'écoulement hypersonique de rentrée, où la présence d'un choc très fort a pour conséquence l'exigence de grande régularité du maillage local ;
— en calcul des ondes : les équations de Helmholtz peuvent être soumises à une condition dite de Sommerfeld [12] modélisant le rayonnement d'un corps à l'infini ; cette condition complexe s'applique généralement sur une coupure circulaire ce qui conduit à une exigence encore plus stricte sur la construction du maillage.

D'une manière générale, l'analyse numérique du schéma d'approximation au bord peut révéler une plus ou moins grande propension à créer, amplifier ou au contraire, absorber les ondes réfléchies artificielles. Cette question difficile dépasse très largement le cadre de ce cours.

On admet donc ici qu'il peut être intéressant de chercher à améliorer la qualité du maillage, notamment vis-à-vis de son bord extérieur, par une meilleure définition de la fonction $\mathcal{A}(\xi, \eta)$.

Des techniques ont été développées dans lesquelles on itère sur la fonction $\mathcal{A}(\xi, \eta)$ dans le but de réaliser une condition supplémentaire, ou du moins, de réduire un critère mesurant son degré de non respect [13]. Plusieurs critères peuvent être pris en considération pour faire en sorte que :

— la frontière Γ_1 approche une courbe cible $\mathcal{C}$;
— les variations de courbure de la frontière Γ_1 soient réduites ;

12. `https://en.wikipedia.org/wiki/Sommerfeld~radiation~condition`
13. *Control of hyperbolic grid-generation systems*, J.-A. Désidéri, Proc. International Conference on Finite Elements in Fluids – New trends and applications, Venice, 15-21 October, 1995, M. Morandi Cecchi et al Eds.

— les lignes de coordonnées iso-η loin de Γ_0 soient aussi alignées que possible avec les iso-valeurs d'une fonction donnée ψ ; en particulier la fonction ψ pour laquelle on construit le mailllage ("adaptation de maillage") ;

etc. A titre d'exemple, on reprend ici cette approche dans le premier cas, où l'on souhaite itérer sur $\mathcal{A}(\xi,\eta)$ afin que le bord extérieur calculé Γ_1 soit aussi proche que possible d'une courbe cible $\mathcal{C}$, dont l'équation analytique est la suivante :

$$g(x,y) = g(\mathbf{x}) = 0 \tag{3.92}$$

A cette fin, on considère la fonctionnelle-coût suivante :

$$\boxed{J(\mathcal{A}) = \int_0^{\xi_{\max}} \tfrac{1}{2} g^2\Big(\mathbf{x}(\xi,\eta_{\max})\Big)\, d\xi} \tag{3.93}$$

Il s'agit alors de minimiser, ou seulement de réduire, le critère de coût $J(\mathcal{A})$, fonctionnelle du contrôle distribué $\mathcal{A}(\xi,\eta)$, dont dépend la fonction d'état, $\mathbf{x}(\xi,\eta)$, par le système hyperbolique, (3.86), soumis à la condition initiale, (3.89), et à la condition de périodicité, (3.90). Dans le but d'appliquer une méthode d'optimisation avec gradient, on souhaite établir l'expression formelle du gradient de fonctionnelle par la technique de l'équation adjointe.

Exercice 14
Définir des fonctionnelles-coûts que l'on pourrait associer aux autres critères cités ci-dessus.

Exercice 15
Imaginer d'autres critères de qualité de maillage et définir des fonctionnelles-coûts corrrespondantes.

Pour cela, on considère une perturbation fonctionnelle de la fonction de contrôle, ici $\delta\mathcal{A}(\xi,\eta)$, perturbation de la distribution des aires, à partir d'un cas de référence supposé connu ou préalablement calculé, et on en examine les conséquences dans un protocole standard constitué des étapes systématiques A-B-C-D suivantes.

A : Première variation de la fonction-état. La perturbation fonctionnelle $\delta\mathcal{A}(\xi,\eta)$ entraîne la première variation $\delta\mathbf{x}(\xi,\eta)$, à valeurs dans $\mathbb{R}^2$, de la fonction-état. Cette fonction est définie par la linéarisation de l'équation d'état, (3.86), et des conditions auxquelles elle est soumise : condition initiale, (3.89), et conditions aux limites, ici de périodicité, (3.90). Il vient :

$$\begin{cases} \delta\mathbf{x}_\eta = \mathcal{B}\delta\mathbf{x}_\xi + \phi\delta\mathcal{A} \quad (\forall(\xi,\eta)\in D) \\ \delta\mathbf{x}(\xi,0) = 0 \quad (\forall\xi) \\ \delta\mathbf{x}(0,\eta) = \delta\mathbf{x}(\xi_{\max},\eta) \quad (\forall\eta) \end{cases} \tag{3.94}$$

Ce système, dit *système linéarisé*, définit implicitement la première variation $\delta\mathbf{x}(\xi,\eta)$ en fonction de la perturbation fonctionnelle $\delta\mathcal{A}(\xi,\eta)$.

B : Première variation de la fonctionnelle-coût. Les perturbations $\delta\mathcal{A}(\xi,\eta)$ et $\delta\mathbf{x}(\xi,\eta)$ entraînent la première variation suivante de la fonctionnelle-coût :

$$\delta J = \int_0^{\xi_{\max}} \gamma(\xi)^t \, \delta\mathbf{x}(\xi,\eta_{\max}) \, d\xi \tag{3.95}$$

où la fonction $\gamma : \Gamma_1 \to \mathbb{R}^2$ est définie par

$$\gamma(\xi) = g\Big(\mathbf{x}(\xi,\eta_{\max})\Big) \nabla g\Big(\mathbf{x}(\xi,\eta_{\max})\Big) \in \mathbb{R}^2 \tag{3.96}$$

et l'indice supérieur ^{t} correspond à la transposition que l'on introduit pour former un produit scalaire. Dans cette notation, on confond les vecteurs du plan aux vecteurs-colonnes de leurs composantes, et les "matrices" 1×1 à des scalaires. Cette notation est ici préférée à celle où le produit scalaire est noté "." dans le but de mieux faire ressortir le rôle de la transposition dans la construction de l'opérateur adjoint.

L'équation (3.95) fait apparaître un terme implicite en $\delta\mathbf{x}$, mais en fait cache également un terme explicite. En effet, il vient :

$$\delta\mathbf{x}(\xi,\eta_{\max}) = \int_0^{\eta_{\max}} \delta\mathbf{x}_\eta \, d\eta = \int_0^{\eta_{\max}} \left(\mathcal{B}\delta\mathbf{x}_\xi + \phi\delta\mathcal{A}\right) \, d\eta \tag{3.97}$$

On introduit la notation :

$$\iint_D [...] = \int_0^{\xi_{\max}} \int_0^{\eta_{\max}} [...] \, d\xi \, d\eta \tag{3.98}$$

L'équation (3.95) devient alors :

$$\delta J = \iint_D \left(\underbrace{\gamma^t \phi \, \delta\mathcal{A}}_{\text{expl.}} + \underbrace{\gamma^t \mathcal{B} \, \delta\mathbf{x}_\xi}_{\text{impl.}} \right) \tag{3.99}$$

Dans cette expression, le terme "implicite" est une fonction de $\delta\mathcal{A}$ par la contrainte du système linéarisé, (3.94). L'équation adjointe, une fois résolue, nous permettra de l'exprimer "explicitement". Avant cela, on transforme le terme implicite par intégration par parties :

$$\begin{aligned}
\iint_D \gamma^t \mathcal{B} \, \delta\mathbf{x}_\xi &= \int_0^{\eta_{\max}} \left(\int_0^{\xi_{\max}} \gamma^t \mathcal{B} \, \delta\mathbf{x}_\xi \, d\xi \right) \, d\eta \\
&= \int_0^{\eta_{\max}} \left(\left[\gamma^t \mathcal{B} \, \delta\mathbf{x}\right]_0^{\xi_{\max}} - \int_0^{\xi_{\max}} \left(\gamma^t \mathcal{B}\right)_\xi \, \delta\mathbf{x} \, d\xi \right) \, d\eta \\
&= - \iint_D \left(\gamma^t \mathcal{B}\right)_\xi \, \delta\mathbf{x}
\end{aligned} \tag{3.100}$$

où le terme de bord, $[...]_0^{\xi_{\max}}$ est nul en vertu de la périodicité en ξ.

Finalement :

$$\delta J = \iint_D \gamma^t \phi \, \delta\mathcal{A} - \iint_D \left(\gamma^t \mathcal{B}\right)_\xi \, \delta\mathbf{x} \tag{3.101}$$

C : Dualité. On introduit une variable $\mathbf{p}(\xi,\eta)$, duale de la variable d'état, $\mathbf{x}(\xi,\eta)$. On dit aussi variable adjointe, ou état-adjoint. D'une manière générale, l'état-adjoint prend ses valeurs dans l'espace dual de l'espace des valeurs de l'équation d'état linéarisée, pour laquelle elle joue le rôle de fonction test. Ici, il s'agit de $\mathbb{R}^2$.

On exprime formellement le produit scalaire de la variable adjointe avec le système linéarisé. Avec la notation adoptée pour la transposition, cela revient à prémultiplier chaque terme de l'EDP linéarisée par $\mathbf{p}^t$ et à intégrer dans tout le domaine. Quel que soit le choix à venir de l'état adjoint, il en résulte l'identité suivante :

$$\iint_D \underbrace{\mathbf{p}^t \delta\mathbf{x}_\eta}_{\text{impl.}} = \iint_D \underbrace{\mathbf{p}^t \mathcal{B}\, \delta\mathbf{x}_\xi}_{\text{impl.}} + \iint_D \underbrace{\mathbf{p}^t \phi\, d\mathcal{A}}_{\text{expl.}} \tag{3.102}$$

Ensuite, on transforme les termes implicites par intégration par parties pour aboutir à des termes analogues à ceux de (3.99). Il vient :

$$\begin{aligned}
\iint_D \mathbf{p}^t \delta\mathbf{x}_\eta &= \int_0^{\xi_{\max}} \left(\int_0^{\eta_{\max}} \mathbf{p}^t \delta\mathbf{x}_\eta \, d\eta \right) d\xi \\
&= \int_0^{\xi_{\max}} \left(\left[\mathbf{p}^t \delta\mathbf{x}\right]_0^{\eta_{\max}} - \int_0^{\eta_{\max}} \mathbf{p}_\eta^t \, \delta\mathbf{x} \, d\eta \right) d\xi \\
&= \int_0^{\xi_{\max}} \mathbf{p}^t \delta\mathbf{x}(\xi,\eta_{\max}) \, d\xi - \iint_D \mathbf{p}_\eta^t \, \delta\mathbf{x}
\end{aligned} \tag{3.103}$$

où le terme de bord, $[...]_0^{\eta_{\max}}$, a été simplifié en utilisant le linéarisé de la condition initiale : $\delta\mathbf{x}(\xi,0) = 0, \forall\xi$; par ailleurs :

$$\iint_D \mathbf{p}^t \mathcal{B}\, \delta\mathbf{x}_\xi = -\iint_D \left(\mathbf{p}^t \mathcal{B}\right)_\xi \, \delta\mathbf{x} \tag{3.104}$$

de manière analogue à (3.100), ce qui suppose, bien naturelllemnt, que l'état-adjoint $\mathbf{p}(\xi,\eta)$ est également périodique en ξ :

$$\mathbf{p}(0,\eta) = \mathbf{p}(\xi_{\max},\eta) \quad (\forall\eta) \tag{3.105}$$

En reportant les résultats de (3.103) et (3.104) dans (3.102), on obtient la relation suivante, vraie quel que soit le choix de $\mathbf{p}$:

$$\int_0^{\xi_{\max}} \mathbf{p}^t \delta\mathbf{x}(\xi,\eta_{\max}) \, d\xi = \iint_D \left(\mathbf{p}_\eta^t - \left(\mathbf{p}^t \mathcal{B}\right)_\xi \right) \delta\mathbf{x} + \iint_D \mathbf{p}^t \phi \, \delta\mathcal{A} \tag{3.106}$$

D : Système adjoint et gradient de fonctionnelle. Il convient maintenant de poser des conditions sur l'état-adjoint $\mathbf{p}(\xi,\eta)$ de manière à ce que les termes implicites qui apparaissent dans l'expression de la première variation de la fonctionnelle-coût d'une part, (3.101), et dans l'identité obtenue par dualité d'autre part, (3.106), soient identiques. Pour cela, on pose :

$$\begin{cases} \mathbf{p}_\eta^t - \left(\mathbf{p}^t \mathcal{B}\right)_\xi = -\left(\gamma^t \mathcal{B}\right)_\xi \quad (\forall(\xi,\eta) \in D) \\ \mathbf{p}^t(\xi,\eta_{\max}) = 0 \quad (\forall\xi) \end{cases} \tag{3.107}$$

En conséquence :

$$\begin{aligned}\delta J &= \iint_D \gamma^t \phi \, \delta\mathcal{A} + \iint_D \left(\mathbf{p}_\eta^t - \left(\mathbf{p}^t \mathcal{B}\right)_\xi\right) \delta\mathbf{x} \\ &= \iint_D \gamma^t \phi \, \delta\mathcal{A} - \iint_D \mathbf{p}^t \phi \, d\mathcal{A}\end{aligned} \tag{3.108}$$

et finalement :

$$\boxed{\delta J = \iint_D G \, \delta\mathcal{A}\,, \quad G = (\gamma - \mathbf{p})^t \, \phi} \tag{3.109}$$

où $\mathbf{p}(\xi, \eta)$ est la solution du "système adjoint" :

$$\begin{cases} \mathbf{p}_\eta = \Big(\mathcal{B}\,(\mathbf{p} - \gamma)\Big)_\xi \quad (\forall (\xi,\eta) \in D) \\ \mathbf{p}(\xi, \eta_{\max}) = 0 \quad (\forall \xi) \\ \mathbf{p}(0, \eta) = \mathbf{p}(\xi_{\max}, \eta) \quad (\forall \eta) \end{cases} \tag{3.110}$$

où la symétrie de $\mathcal{B}$ a été utilisée, et la périodicité de $\mathbf{p}$ en ξ rappelée.

On constate qu'à une équation d'état hyperbolique sous forme quasi-linéaire correspond une équation adjointe hyperbolique sous forme divergence [14] associées à des matrices jacobiennes transposées l'une de l'autre (ici symétriques donc égales). En d'autres termes, **le linéarisé de l'équation d'état et l'équation adjointe font intervenir des opérateurs adjoints l'un de l'autre**. A la condition initiale sur l'état correspond une condition finale de Dirichlet homogène sur l'état adjoint : **l'équation adjointe constitue un système rétrograde**. Enfin, à la condition de périodicité en ξ sur l'état correspond la même condition sur l'état adjoint.

L'équation (3.109) fournit l'expression de la première variation de la fonctionnelle-coût, δJ, explicitement en fonction de la perturbation $\delta\mathcal{A}$ de la variable de contrôle distribué. Cette expression est appelée "**gradient de fonctionnelle-coût**".

D'un point de vue algorithmique, on résout le système d'état dans le sens de η croissant à partir de la donnée initiale ; puis le système adjoint de manière rétrograde à partir de la condition finale sur l'état adjoint, et les valeurs locales de l'état qui influent au travers de la fonction γ. A l'issue de ces deux intégrations, on peut calculer la fonction G qui définit le gradient de fonctionnelle.

Afin de réduire le critère, on pourra ensuite faire le choix de perturber le contrôle $\mathcal{A}$ de manière opposée au gradient, en construisant un algorithme itératif où, par exemple :

$$\delta\mathcal{A} = -\frac{\varepsilon G}{\iint G^t G} \tag{3.111}$$

où ε est un petit paramètre positif contrôlant la stabilité de l'itération.

Exercice 16
Au vu de (3.95) et (3.106), il apparait qu'on aurait pu alternativement faire le choix suivant

14. L'inverse serait vrai ; voir Exercice 17.

de système adjoint :

$$\begin{cases} \mathbf{p}_\eta = \Big(\mathcal{B}\mathbf{p}\Big)_\xi \quad (\forall (\xi,\eta) \in D) \\ \mathbf{p}(\xi, \eta_{\max}) = \gamma(\xi) \quad (\forall \xi) \\ \mathbf{p}(0,\eta) = \mathbf{p}(\xi_{\max}, \eta) \quad (\forall \eta) \end{cases} \tag{3.112}$$

Montrer que ce choix conduit au même gradient de fonctionnelle.

Exercice 17
On considère des champs de vecteurs $\mathbf{u}(x)$ *et* $\mathbf{v}(x)$ *(*$x \in \mathbb{R}$, $\mathbf{u}(x) \in \mathbb{R}^n$, $\mathbf{v}(x) \in \mathbb{R}^n$*) périodiques en* x*. Soit* $\mathbf{A}(\mathbf{x})$ *une matrice coefficient* $n \times n$ *donnée, également périodique. Soit l'opérateur* $\mathbf{p}$ *suivant et son adjoint* $\mathbf{q}$ *opérant sur de tels champs :*

$$\mathbf{p} = \mathbf{A}(x)\frac{\partial}{\partial x}[\dots], \quad \mathbf{q} = \mathbf{p}^\star \tag{3.113}$$

où $[\dots]$ *symbolise l'emplacement du champ auquel s'applique l'opérateur.*

1. Montrer que :

$$\mathbf{q} = -\frac{\partial}{\partial x}\left(\mathbf{A}(x)^t[\dots]\right). \tag{3.114}$$

Pour cela, on traduira en intégrales l'égalité suivante entre produits scalaires :

$$\big(\mathbf{v}, \mathbf{pu}\big) = (\mathbf{qv}, \mathbf{u}). \tag{3.115}$$

2. On souhaite maintenant examiner comment se traduit ce résultat lorsqu'on remplace les opérateurs continus par des matrices de discrétisation par différences finies sur un maillage uniforme. Dans ce cas, on introduit p *degrés de liberté par période, et on représente les champs* $\mathbf{u}$ *et* $\mathbf{v}$ *par leurs discrétisés, vecteurs de dimension* pn *:*

$$\mathbf{u}_h = \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_p \end{pmatrix}, \quad \mathbf{v}_h = \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_p \end{pmatrix}$$

où $\mathbf{u}_j$ *et* $\mathbf{v}_j$ *sont les valeurs en* x_j *des champs* $\mathbf{u}$ *et* $\mathbf{v}$ *respectivement. Les opérateurs sont alors représentés par des matrices de dimension* $pn \times pn$*.*

Expliciter la structure par blocs des matrices de représentation des opérateurs $\mathbf{p}$ *et* $\mathbf{q}$ *dans le cas de différences finies centrées ; puis décentrées d'ordre 1 (amont ou aval).*

3.4 Conclusion : formulation "symbolique" type

Le développement présent a pour but de formaliser les étapes invariables dans le calcul d'un gradient de fonctionnelle. Le sens précis des équations symboliques devra être précisédans les cas spécifiques considérés.

— **Variable d'Ã©tat : x** (distribuÃ©e dans le domaine Ω) (en aÃ©rodynamique : $\mathbf{x} = (\rho, \rho\vec{V}, E, ...)$)
— **Variable de contrôle : u** (par exemple un vecteur contrôlant une forme)

— **Équation d'Ã©tat** (une EDP et ses CL) :

$$\begin{cases} \mathbf{E}(\mathbf{x},\mathbf{u}) = 0 \quad ((x,y,z) \in \Omega) \\ C_L(\mathbf{x},\mathbf{u}) = 0 \quad ((x,y,z) \in \Gamma = \partial\Omega) \end{cases} \tag{3.116}$$

— **Fonctionnelle :**

$$\mathbf{J}(\mathbf{u}) = J(\mathbf{x}(\mathbf{u}),\mathbf{u}) = \int_\Omega \mathcal{L}(\mathbf{x},\mathbf{u}) \tag{3.117}$$

— **SchÃ©ma fonctionnel :** $\delta\mathbf{u} \Longrightarrow \delta\mathbf{x} \Longrightarrow \delta\mathbf{J}$: comment relier $\delta\mathbf{J}$ Ã $\delta\mathbf{u}$ seulement ?
— **SystÃ¨me linÃ©arisÃ© :**

$$\begin{cases} \mathbf{E_x}\,\delta\mathbf{x} + \mathbf{E_u}\,\delta\mathbf{u} = 0 \quad ((x,y,z) \in \Omega) \\ C_{L\mathbf{x}}(\mathbf{x},\mathbf{u})\delta\mathbf{x} + C_{L\mathbf{u}}(\mathbf{x},\mathbf{u})\delta\mathbf{u} = 0 \quad ((x,y,z) \in \Gamma = \partial\Omega) \end{cases} \tag{3.118}$$

et

$$\delta\mathbf{J} = \int_\Omega [\underbrace{\mathcal{L}_\mathbf{x}\,\delta\mathbf{x}}_{\text{impl.}} + \underbrace{\mathcal{L}_\mathbf{u}\delta\mathbf{u}}_{\text{expl.}}] = \int_\Omega \left[\mathcal{L}_\mathbf{u}\delta\mathbf{u} + \left(\nabla_\mathbf{x}\mathcal{L}, \delta\mathbf{x}\right)\right] \tag{3.119}$$

— **Introduction d'une variable adjointe distribuÃ©e : p, et intÃ©gration par parties** de l'identitÃ©

$$\begin{aligned} 0 = \int_\Omega \big(\mathbf{p}, \underbrace{\mathbf{E_x}\,\delta\mathbf{x} + \mathbf{E_u}\,\delta\mathbf{u}}_{=0}\big) &= \int_\Omega \left[\left(\mathbf{E}_\mathbf{x}^\star\mathbf{p}, \delta\mathbf{x}\right) + \left(\mathbf{p}, \mathbf{E_u}\,\delta\mathbf{u}\right)\right] + [...] \\ \Longrightarrow \int_\Omega \left(\mathbf{E}_\mathbf{x}^\star\mathbf{p}, \delta\mathbf{x}\right) &= -\int_\Omega \left(\mathbf{p}, \mathbf{E_u}\,\delta\mathbf{u}\right) + [...] \end{aligned} \tag{3.120}$$

où [..] dÃ©signe des termes de bord (distincts chaque fois) non prÃ©cisÃ©s ici.
— **Simplification par :**
– application des CL connues sur $\mathbf{x}$ et $\delta\mathbf{x}$
– imposition de CL spÃ©cifiques Ã la variable adjointe $\mathbf{p}$ (pour simplifier les termes de bord)
– imposition du **systÃ¨me adjoint** (EDP) :

$$\mathbf{E}_\mathbf{x}^\star\,\mathbf{p} = \nabla_\mathbf{x}\mathcal{L} \tag{3.121}$$

soumis Ã ses CL. Ce systÃ¨me fait intervenir l'opÃ©rateur $\mathbf{E}_\mathbf{x}^\star$, c'est-Ã -dire l'adjoint de l'opÃ©rateur de linÃ©arisation du systÃ¨me d'Ã©tat par rapport Ã la variable d'Ã©tat ce qui permet d'identifier le :
— **Gradient de fonctionnelle :**

$$\delta\mathbf{J} = \int_\Omega \left[\mathcal{L}_\mathbf{u}\delta\mathbf{u} - \left(\mathbf{p}, \mathbf{E_u}\,\delta\mathbf{u}\right)\right] \tag{3.122}$$

tous les termes ayant Ã©tÃ© explicitÃ©s en $\delta\mathbf{u}$.

Conclusions : Insistons sur le fait que pour prendre un sens précis, au delà du symbolisme, le développement que nous venons de faire doit être exprimé dans un cas spécifique où les opérateurs et les conditions aux bords sont prÃ©cisÃ©es, et qu'un problème bien posé en résulte. Cette restriction importante étant faite, on fait les constations suivantes :

- L'équation adjointe permet d'exprimer la variation de fonctionnelle en fonction seulement de la variation de la fonction de contrôle.
- La connaissance du gradient de fonctionnelle permet d'exprimer formellement les conditions de stationnarité de la fonctionnelle, et par discrétisation de construire des méthodes numériques de minimisation de $\mathbf{J}(\mathbf{u})$ à base de gradient.
- **L'opérateur qui intervient dans le système adjoint, $\mathbf{E}^{\star}_{\mathbf{x}}$, est l'adjoint du linéarisé par rapport à la fonction d'état x, de l'opérateur E par lequel s'exprime l'équation d'état**, (3.116), ce qui justifie la terminologie.

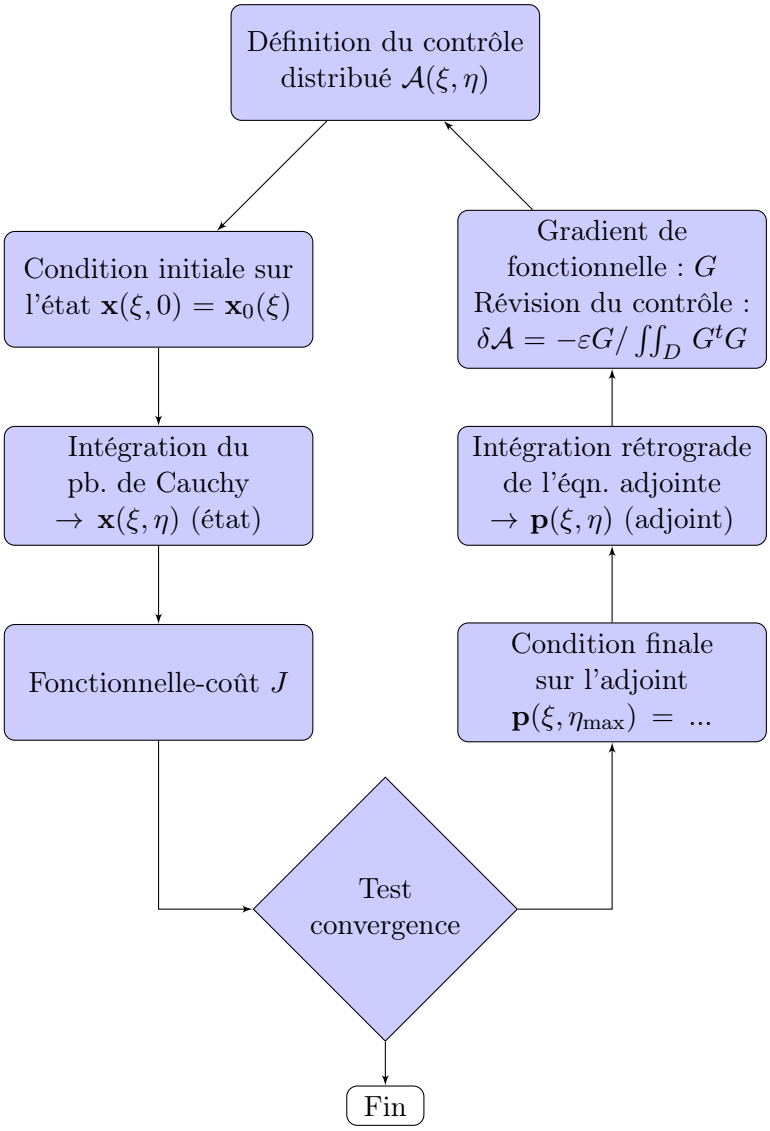


FIGURE 3.15 – Organigramme d'une itération de contrôle d'un problème distribué de Cauchy ; la révision du contrôle est simplement faite ici par une méthode de type *steepest-descent* sans préconditionnement particulier dans le seul but d'illustration.

Chapitre 4

Optimisation multiobjectif et ingénierie concourante

4.1 Notions générales

Mots clés : optimisation multiobjectif, ingénierie concourante, la quête du front de Pareto

4.1.1 Exemples de problèmes d'ingénierie concourante en aérodynamique et ses couplages

(couplages à d'autres disciplines, en particulier à l'analyse structurale et thermique)

- **L'optimisation dans les sciences de l'ingénieur est presque toujours multiobjectif,** car elle revêt l'un des aspects suivants :
  - Multicritère
    plusieurs fonctions d'intérêt associées à un écoulement donné
    exemple : forces et moments en stabilité/maneuvrabilité
  - Multipoint
    plusieurs points de l'enveloppe de vol sont critiques et donnent lieu à l'étude aérodynamique de plusieurs conditions d'écoulement (M_∞, $\alpha = AoA$; écoulement externe/interne, local/de configuration)
    exemple : optimisation concourante à but de :
    – réduction de traînée de croisière en un ou plusieurs points de l'enveloppe ("optimisation robuste") (régime d'écoulement transsonique pour un avion de transport civil actuel)
    – maximisation de portance dans les conditions du décollage ou de l'atterrissage (régime subsonique)
  - Multidisciplinaire
    Aérodynamique concourante à, et concurrente d'autres disciplines
    exemples : performance aérodynamique contre critères liés à l'analyse structurale et thermique, l'acoustique, le calcul des ondes (visibilité, furtivité)
- **En aérodynamique compressible (équations d'Euler ou de Navier-Stokes), les problèmes d'optimisation gardent un caractère fonctionnel,** y compris lorsqu'on représente la variable fonctionnelle (en particulier, une forme) par une paramétrisation contrôlée par un nombre fini de variables.

calcul des gradients : équation adjointe, équations des sensibilités, différences finies, etc

4.1.2 Objectif, notations et convention de minimisation

Les algorithmes d'optimisation multiobjectif visent à apporter des solutions efficaces principalement à deux types de problématique :

- l'optimisation multi-objectif en grande dimension, lorsque le nombre de fonctions objectifs est grand ; les cas où le nombre de variables est également grand soulèvent aussi des difficultés spécifiques ;
- l'ingénierie concourante dans le cas d'un système complexe dont la performance se mesure par plusieurs critères liés à une ou plusieurs disciplines physiques.

Les principales notations sont les suivantes :

- Entier N, dimension d'espace des variables d'optimisation ($N \geq 1$)
- Entier M, nombre de fonctions objectifs ($M \geq 2$)
- Domaine admissible $\Omega_{ad} \subseteq \mathbb{R}^N$ (généralement fermé et convexe)
- Fonctions objectifs $f_j(\mathbf{x})$ ($1 \leq j \leq M$, $\mathbf{x} \in \Omega_{ad}$) (généralement continues et convexes).

On convient que les fonctions-objectifs sont des "fonctions-coûts", et que l'optimisation vise à les minimiser, ou plus modestement, à les réduire. En optimisation sous contrainte d'équation différentielle (EDP ou EDO), la définition précise des fonctions coûts est liée à la notion de fonctionnelle.

4.1.3 Notion d'optimalité au sens de Pareto

En optimisation mono-objectif, lorsque deux éléments admissibles $\mathbf{x}_1$ et $\mathbf{x}_2$ ont une performance différente vis-à-vis de l'unique critère, l'un d'eux domine l'autre en efficacité. Ce n'est plus obligatoirement le cas en optimisation multi-objectif, car les relations d'ordre dans $\mathbb{R}^M$ sont partielles. La notion d'optimalité exige donc d'être revisitée, et précisée.

Dominance en efficacité (Voir par exemple : K. Miettinen : *Nonlinear Multiobjective Optimization*, Kluwer Academic Publishers (1998).)

Définition 1 (Dominance en efficacité)
On dit que le point de conception $\mathbf{x}_1 \in \Omega_{ad}$ domine en efficacité le point de conception $\mathbf{x}_2 \in \Omega_{ad}$ ssi

$$f_j(\mathbf{x}_1) \leq f_j(\mathbf{x}_2) \quad (\forall j = 1, \ldots, m) \tag{4.1}$$

et si au moins l'une de ces inégalités est stricte. (On note alors parfois $\mathbf{x}_1 \succ \mathbf{x}_2$.)

Le "Saint-Graal" du concepteur :

Définition 2 (Ensemble de Pareto (*Pareto set*))
ensemble constitué des points de conception non dominés, dits "solutions Pareto-optimales".

Définition 3 (Front de Pareto (*Pareto front*))
Image de l'ensemble de Pareto dans l'espace des fonctions.

Avant d'examiner les propriétés générales des ensembles et fronts de Pareto dans un contexte de fonctions convexes, nous allons examiner un exemple très simple pour nous familiariser avec ces concepts.

FIGURE 4.1 – Vilfredo Pareto (tiré de Wikipedia)

Un exemple simple

Commençons par une expérience numérique sur un exemple simple d'optimisation bicrière sans contraintes dans lequel les fonctions-coûts sont quadratiques et convexes :

Variables	**Fonctions-coûts**
$N = 2$	$M = 2$
$\mathbf{x} = (x_1, x_2)$	$f_1(\mathbf{x}) = (x_1 - \frac{1}{2})^2 + (x_2 - \frac{1}{2})^2$
$\Omega_{ad} = [-1, 1] \times [-1, 1]$	$f_2(\mathbf{x}) = f_1(-\mathbf{x}) = (x_1 + \frac{1}{2})^2 + (x_2 + \frac{1}{2})^2$
(fermé, convexe)	(quadratiques convexes)

(4.2)

On discrétise très grossièrement le domaine admissible $[-1, 1] \times [-1, 1]$ en un quadrillage de pas $h = \frac{1}{10}$. On évalue les fonctions sur la grille, ce qui fournit un aperçu de l'image $F(\Omega_{ad})$ du domaine par la fonction $F = (f_1, f_2)$. C'est un domaine connexe en forme de chapeau d'arlequin, symétrique par rapport à la bissectrice $f_1 = f_2$.

On identifie l'optimum $\mathbf{x}_1^\star = (\frac{1}{2}, \frac{1}{2})$ réalisant le minimum absolu (ou global) de $f_1(\mathbf{x})$ (resp. $\mathbf{x}_2^\star = (-\frac{1}{2}, -\frac{1}{2})$ réalisant le minimum absolu de $f_2(\mathbf{x})$). Dans l'espace des fonctions, les points $\mathbf{x}_1^\star$ et $\mathbf{x}_2^\star$ ont pour images respectives $F(\mathbf{x}_1^\star) = (0, 2)$ et $F(\mathbf{x}_2^\star) = (2, 0)$.

Par application d'une procédure de **tri global** des valeurs de fonctions, on identifie les points non dominés de l'image $Im\, F$. Ces points (en gras) forment un arc convexe [1] qui relie $F(\mathbf{x}_1^\star)$ à $F(\mathbf{x}_2^\star)$ et constitue la limite atteignable en matière de performances maximales (valeurs minimales des fonctions coûts). C'est le front de Pareto.

Le front de Pareto est constitué des images du segment de droite qui relie $\mathbf{x}_1^\star$ à $\mathbf{x}_2^\star$. Ce segment, $\{(x_1, x_2)$ tel que $x_1 = x_2 = x,\ -\frac{1}{2} \le x \le \frac{1}{2}\}$, constitue l'ensemble de Pareto. En raison de la discrétisation, assez grossière, numériquement, les coordonnées des points non dominés (en gras) ne sont identifiées qu'à $\pm h$ près.

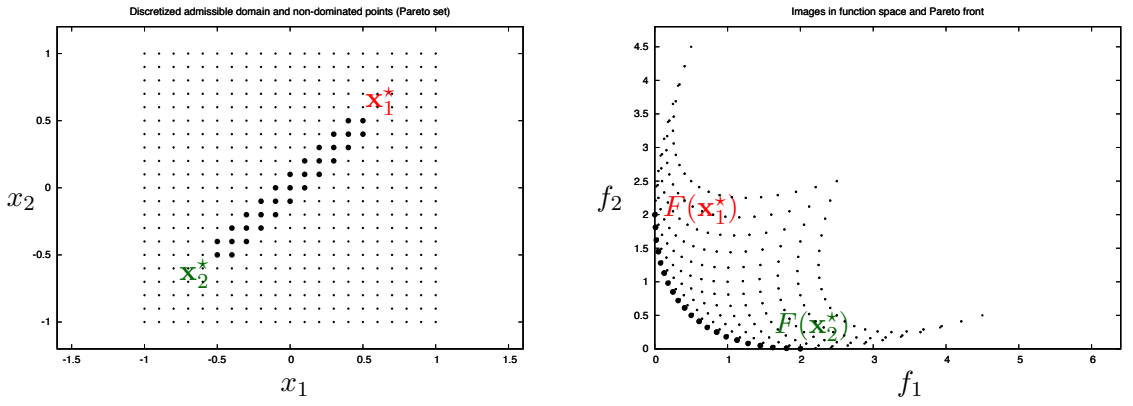


FIGURE 4.2 – Discrétisation des variables (x_1, x_2) et images discrètes de (f_1, f_2) dans l'espace des fonctions

Le front de Pareto admet donc la représentation paramétrique suivante :

$$x_1 = x_2 = x\,, \quad f_1 = 2(x - \tfrac{1}{2})^2\,, \quad f_2 = 2(x + \tfrac{1}{2})^2\,, \quad -\tfrac{1}{2} \le x \le \tfrac{1}{2}\,. \tag{4.3}$$

1. Nous verrons ultérieurement que la convexité de cet arc vient du fait que les fonctions $f_1(\mathbf{x})$ et $f_2(\mathbf{x})$ sont elles-mêmes convexes.

Il vient $\sqrt{f_1} + \sqrt{f_2} = \sqrt{2}$, puis $(f_1 - 2)^2 + (f_2 - 2)^2 = 2^2$: il s'agit d'un quart du cercle de centre (2,2) et de rayon 2.

Nous allons maintenant étudier l'effet de l'introduction d'une perturbation régulière (C^∞) et locale (à support compact) apportée aux fonctions, concentrée aux points $\mathbf{x}_A = \left(-\frac{1}{2}, \frac{1}{2}\right)$ et $\mathbf{x}_B = -\mathbf{x}_A = \left(\frac{1}{2}, -\frac{1}{2}\right)$ qui ne sont pas Pareto-optimaux.

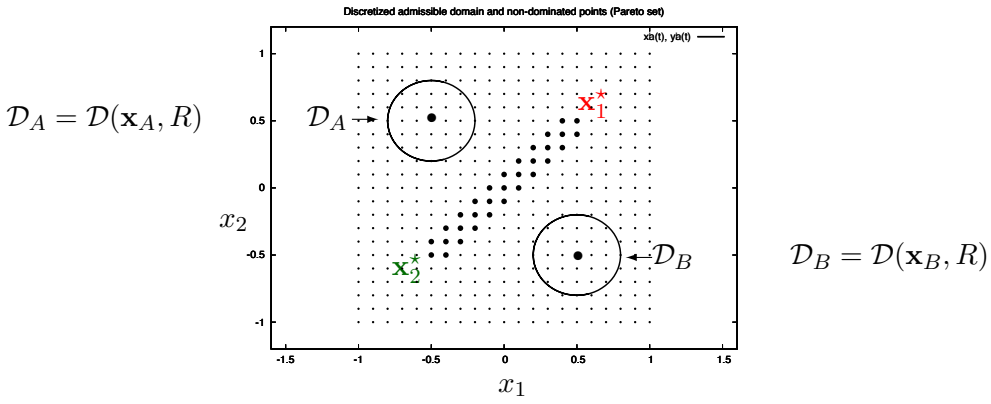


FIGURE 4.3 – Disques $\mathcal{D}_A$ et $\mathcal{D}_B$ dans lesquels on apporte aux fonctions (f_1, f_2) des perturbations C^∞ pour définir $(\tilde{f}_1, \tilde{f}_2)$

Pour cela, on considère la fonction suivante dont on vérifiera qu'elle est C^∞ à support compact $[-1, 1]$:

$$\phi(t) = \begin{cases} \exp\left(\dfrac{t^2}{t^2 - 1}\right) & \text{si } -1 < t < 1 \\ 0 \text{ sinon.} \end{cases} \qquad \begin{array}{ccccc} & & 1 & & \\ & \nearrow & & \searrow & \\ 0 & & & & 0 \\ & \searrow & & \nearrow & \\ & & -1 \quad 1 & & \end{array} \tag{4.4}$$

Remplaçons les fonctions $\big(f_1(\mathbf{x}), f_2(\mathbf{x})\big)$ par les suivantes :

$$\begin{cases} \widetilde{f}_1(\mathbf{x}) = f_1(\mathbf{x}) - \phi\left(\dfrac{\|\mathbf{x} - \mathbf{x}_A\|}{R}\right) - 2\phi\left(\dfrac{\|\mathbf{x} - \mathbf{x}_B\|}{R}\right) \\ \widetilde{f}_2(\mathbf{x}) = \widetilde{f}_1(-\mathbf{x}) \end{cases} \tag{4.5}$$

($R = 0.3$). Ces perturbations n'altèrent pas la régularité C^∞ des fonctions ; de plus, elles sont inopérantes à l'extérieur des disques $\mathcal{D}_A$ et $\mathcal{D}_B$, en particulier sur l'ancien ensemble de Pareto. Ces fonctions sont représentées à la Figure 4.4.

Par contre, aux centres des disques, on a :

$$\text{en } \mathbf{x}_A : \ \big(\widetilde{f}_1, \widetilde{f}_2\big) = (0, -1), \ \text{en } \mathbf{x}_B : \ \big(\widetilde{f}_1, \widetilde{f}_2\big) = (-1, 0), \tag{4.6}$$

alors que les fonctions d'origine étaient uniformément positives. Ces centres ont donc pour images dans l'espace des fonctions des performances supérieures (en matière de minimisation) à celles de toutes les solutions précédemment Pareto-optimales ! Le front de Pareto est donc complètement bouleversé. Afin de le vérifier, comme précédemment, on évalue les fonctions aux points de discrétisation du domaine admissible, on reporte les images dans l'espace des fonctions, et on effectue le tri pour identifier les nouveaux points non dominés. Le résulat

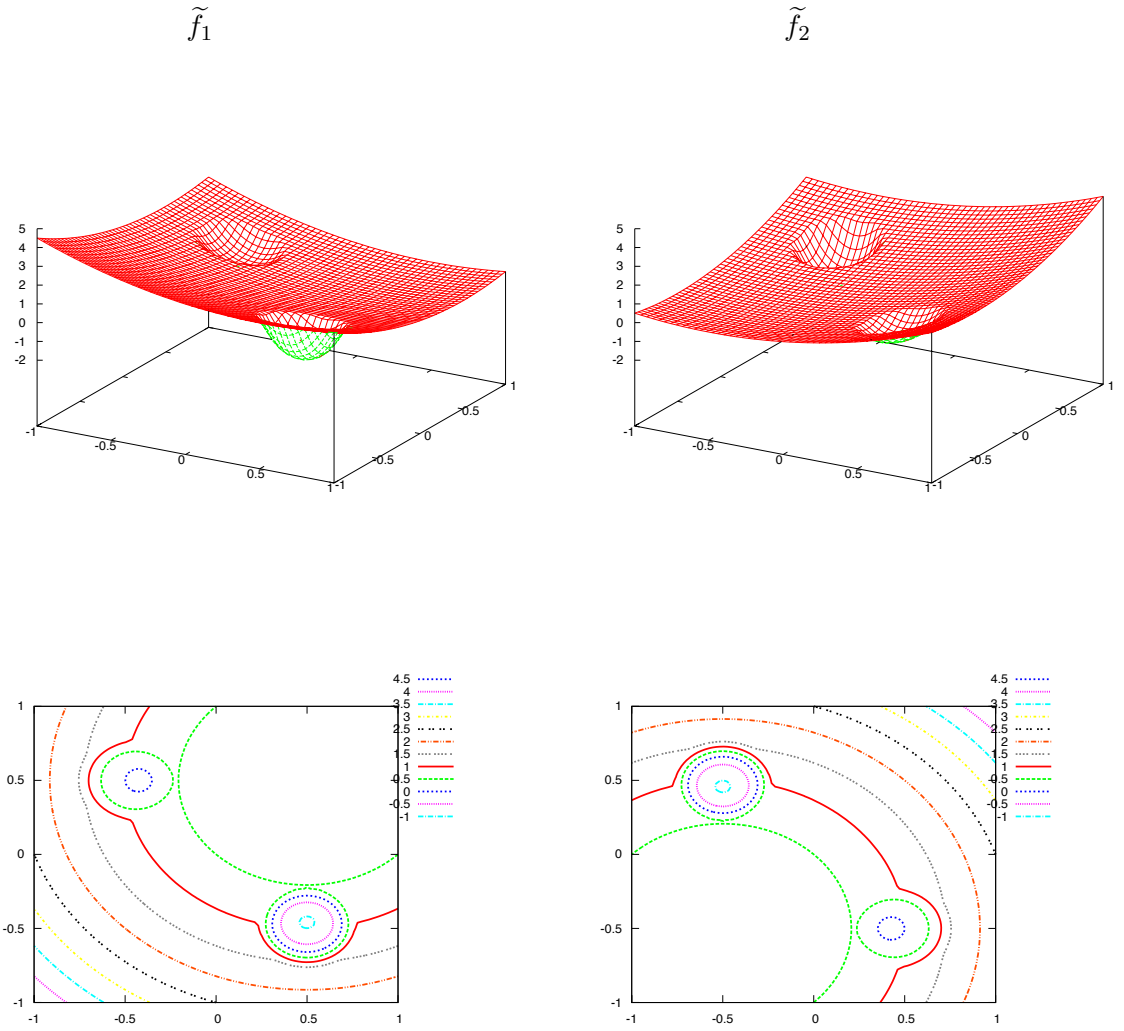


FIGURE 4.4 – Représentation des fonctions perturbées $\widetilde{f}_1$ et $\widetilde{f}_2$: en haut, les perturbations, négatives et locales, sont indiquées en vert ; en bas, la représentation est par lignes de niveau (iso-valeurs).

numérique est indiqué ci-après (Figure 4.5). L'image $Im\, F$ n'apparaît plus comme un domaine connexe, mais constitué de deux arcs disjoints symétriques situé dans le quart de plan $f_1 \leq 0$, $f_2 \leq 0$, bien qu'il soit difficile de l'affirmer à partir d'une discrétisation forcément limitée par la précision du calcul numérique.

En répétant l'expérience avec une discrétisation plus fine du carré ($h = \frac{1}{100}$), on obtient une définition plus précise du front de Pareto (Figure 4.6). Les solutions Pareto-optimales se situent à l'intérieur des disques et constituent deux parties disjointes symétriques. À l'échelle de la résolution numérique, le front a la même configuration.

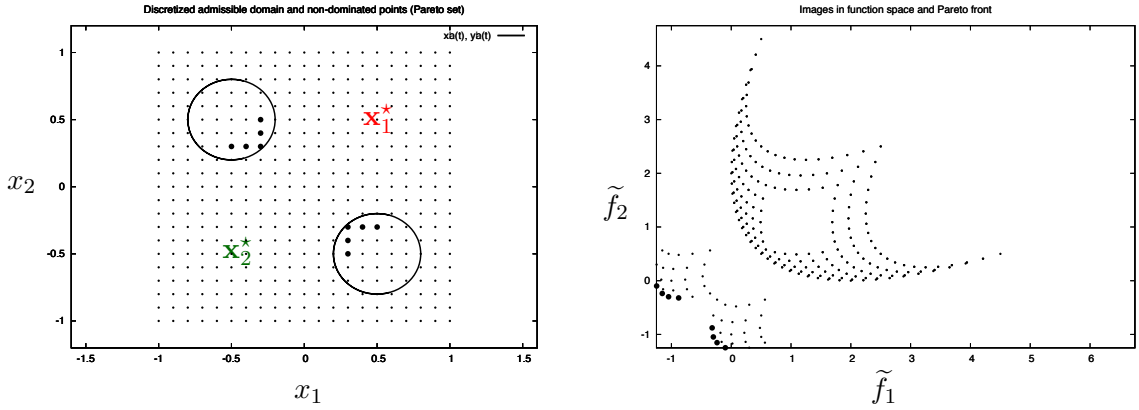


FIGURE 4.5 – Ensemble et front de Pareto associés aux fonctions perturbées $\widetilde{f}_1$ et $\widetilde{f}_2$

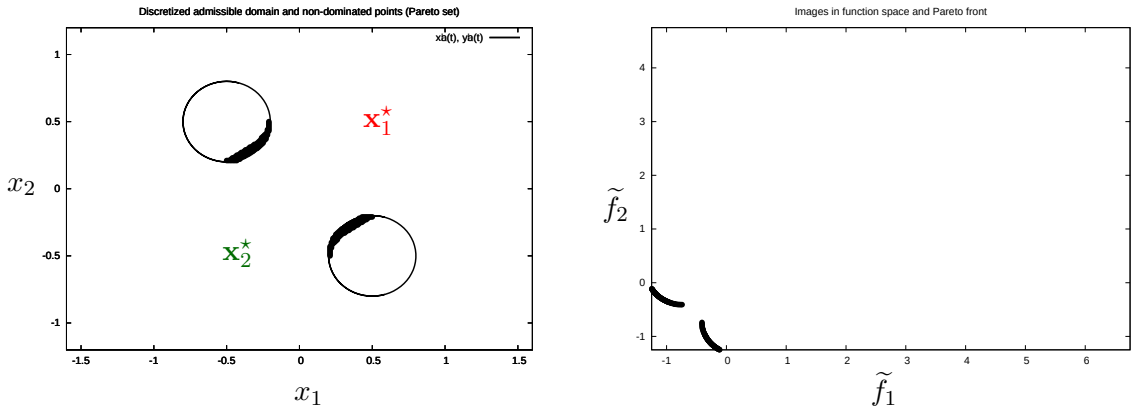


FIGURE 4.6 – Ensemble et front de Pareto associés aux fonctions perturbées $\widetilde{f}_1$ et $\widetilde{f}_2$ (discrétisation fine : $h = \frac{1}{100}$)

Conclusions :

- Notre expérience numérique suggère que le front de Pareto est discontinu bien que les fonctions soient C^∞. Il est difficile de l'affirmer sur la seule base d'une expérience conduite avec une certaine précision arithmétique. Néanmoins, il est vrai que la continuité des fonctions ne suffit pas à garantir la continuité du front comme le confirme le cas-test classique de Kursawe (voir plus loin, Figure 4.8).
- Régularité et convexité locale des fonctions sont des hypothèses insuffisantes pour décider si un point de conception correspond bien à une solution Pareto-optimale ; aucun critère strictement local ne le permet : les notions d'ensemble et de front de Pareto sont globales. En conséquence, la détermination du front par une méthode de gradient peut ou doit être complétée par un algorithme de tri global.

Cependant nous allons maintenant montrer que régularité et convexité globale conduisent à un front constitué de portions convexes.

Propriétés générales des ensembles et fronts de Pareto dans le contexte convexe

L'étudiant est invité à consulter un ouvrage de référence, ou Wikipedia (`https://en.wikipedia.org/wiki/Convex_function`), pour consolider ses connaissances en matière de convexité. On rappelle ici seulement quelques notions fondamentales.

Il est sous-entendu que les espaces métriques de travail sont ici $\mathbb{R}^N$ ou $\mathbb{R}^M$.

Définition 4 (Domaine convexe)
Une partie $D \subseteq \mathbb{R}^N$ est un domaine convexe ssi pour tout couple $(\mathbf{x}, \mathbf{y})$ de points de D, le segment $[\mathbf{x}, \mathbf{y}]$ est entièrement conenu dans D.

Définition 5 (Courbe, surface, variété convexe)
Toute partie connexe du bord d'un domaine convexe.

Définition 6 (Fonction convexe)
Toute fonction f définie dans un domaine convexe D telle que pour tout $\alpha \in [0, 1]$:

$$f\big((1-\alpha)\mathbf{x} + \alpha\mathbf{y}\big) \leq (1-\alpha)f(\mathbf{x}) + \alpha f(\mathbf{y}). \tag{4.7}$$

En particulier, si f est une fonction continue convexe de la variable réelle, son graphe est une "courbe convexe".

On suppose désormais que les fonctions-coûts sont continues et convexes, définies dans un domaine Ω_{ad} fermé et convexe. L'image du domaine, $Im\, F = F(\Omega_{ad})$ par la fonction

$$F = F(\mathbf{x}) = \{f_j(\mathbf{x})\} \quad (j = 1, \ldots, M) \tag{4.8}$$

est un fermé de $\mathbb{R}^M$. On admet alors que le front de Pareto est un bord de $Im\, F$ qui limite le domaine atteignable en efficacité.

Proposition 1
Sous les hypothèses faites de convexité du domaine admissible et des fonctions coûts, le front de Pareto est (une variété) convexe (de $\mathbb{R}^M$).

Preuve : Dans l'espace métrique des fonctions ($\mathbb{R}^M$), le domaine atteignable est l'image par la fonction continue $F(\mathbf{x})$ du domaine admissible Ω_{ad} supposé fermé. C'est donc un fermé. Le front de Pareto est une partie de son bord.

Par l'absurde, supposons que le front de Pareto contienne une partie concave contenant les points suivants (Pareto-optimaux par hypothèse) :

$$F_A = F(\mathbf{x}_A), \quad F_B = F(\mathbf{x}_B), \quad (\mathbf{x}_A, \mathbf{x}_B \in \Omega_{ad}). \tag{4.9}$$

Alors le segment $F_A F_B$, à l'exception des ses extrémités, est entièrement contenu dans la partie concave. Il ne contient donc aucun point atteignable. En particulier

$$F_\alpha = (1-\alpha)F_A + \alpha F_B \tag{4.10}$$

n'est pas atteignable, et ceci quel que soit $\alpha \in (0,1)$ $(0 < \alpha < 1)$. Soit alors le point suivant dans l'espace des variables :

$$\mathbf{x}_\alpha = (1-\alpha)\mathbf{x}_A + \alpha \mathbf{x}_B. \tag{4.11}$$

Le domaine admissible étant convexe, ce point est admissible. Par hypothèse de convexité des fonctions, on a (composante par composante) :

$$F(\mathbf{x}_\alpha) \leq (1-\alpha)F(\mathbf{x}_A) + \alpha F(\mathbf{x}_B) = F_\alpha, \tag{4.12}$$

ce qui est impossible car la performance F_α n'est pas atteignable. □

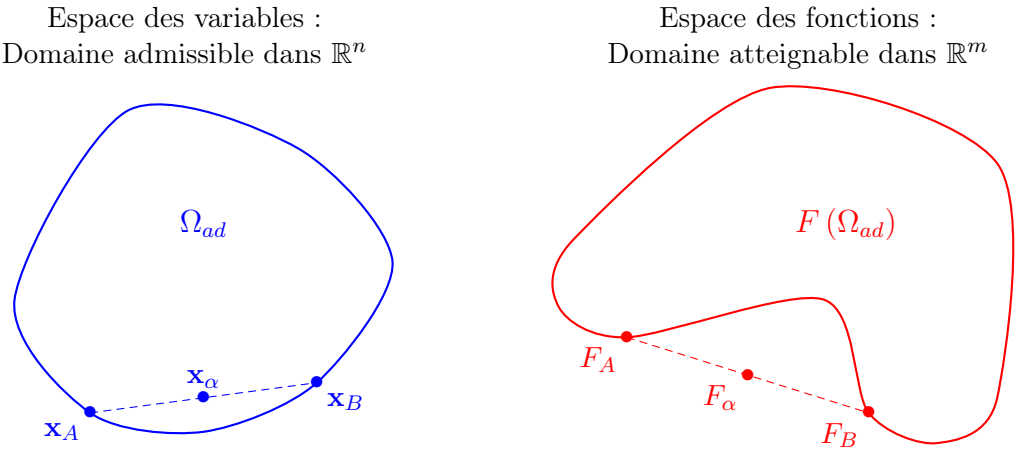


FIGURE 4.7 – Cas d'un front de Pareto contenant une partie concave

Remarque 1
Dans un problème de minimization différentiable sans contraintes dans un domaine ouvert, la stationnarité est une condition nécessaire d'extrêmalité. En un point stationnaire, la convexité est une condition suffisante de minimum local. Par contre, lorsque le problème est soumis à une contrainte, la condition de convexité ne s'applique pas à la fonction elle-même, mais au lagrangien. Par exemple, le problème

$$\min_{(x,y)\in\mathbb{R}^2} f(x,y) = 3 - \left(x + x^2 + y^2\right), \quad \textit{soumis à :}\ x^2 + y^2 = 1, \tag{4.13}$$

est bien posé. Il admet la solution unique $x^\star = 1$, $y^\star = 0$. Cependant la fonction $f(x,y)$ n'est pas convexe en $(x^\star, y^\star)$; elle est même concave dans tout le domaine ! Dans un problème de minimisation multiobjectif, même non contraint, chaque fonction-coût joue en quelque sorte le rôle de contrainte appliquée aux autres. Il ne faut donc pas s'étonner si en un point Pareto-optimal, l'une des fonctions-coûts n'est pas convexe. Au vu de la Proposition 1, une telle situation est manifeste lorsque le front de Pareto présente une partie connexe concave.

Une propriété d'inclusion. En ingénierie concourante, il est fréquent que le praticien après avoir réalisé l'optimisation multi-objectif des fonctions $\{f_1, f_2, \ldots, f_k\}$ souhaite faire intervenir une fonction coût additionnelle f_{k+1}. Ceci peut se justifier par l'intention d'introduire la difficulté numérique graduellement, pour introduire un critère lié à une nouvelle discipline physique, ou parce que l'optimisation initiale n'a pas conduit à un choix convaincant du concept. Très intuitivement, lorsqu'on augmente l'ensemble des fonctions coûts, on s'attend à ce que le critère d'optimalité de Pareto devienne de moins en moins discriminant, et qu'ensemble et front de Pareto en soient agrandis. Précisément :

Proposition 2
On se place toujours sous hypothèse de fonctions coûts continues et convexes, définies dans un domaine admissible fermé et convexe. Soit un entier $k \geq 2$. On note $\mathcal{S}_k$ et $\mathcal{S}_{k+1}$ les ensembles de Pareto associés aux fonctions $\{f_1, f_2, \ldots, f_k\}$ et $\{f_1, f_2, \ldots, f_{k+1}\}$ (dans $\mathbb{R}^N$), et $\mathcal{F}_k$ (dans $\mathbb{R}^k$) et $\mathcal{F}_{k+1}$ (dans $\mathbb{R}^{k+1}$) les fronts de Pareto associés. On note $\mathcal{F}'_k$ le relèvement de $\mathcal{F}_k$ à $\mathbb{R}^{k+1}$ parallèlement à l'axe f_{k+1}, c'est-à-dire l'ensemble des $(k+1)$-uplets $\{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_{k+1}(\mathbf{x})\}$ ($\mathbf{x} \in \mathcal{S}_k$). Alors :

1. *Tout élément du relèvement $\mathcal{F}'_k$ ou bien appartient au front $\mathcal{F}_{k+1}$, ou bien se projette (parallèlement à l'axe f_{k+1}) en un point de ce front.*
2. *Si de plus, l'existence d'un segment linéaire du domaine sur lequel les fonctions coûts $\{f_1, f_2, \ldots, f_k\}$ seraient toutes constantes est exclu, on a les propriétés d'unicité et d'inclusion suivantes : tout élément du front $\mathcal{F}_k$ est l'image d'un unique élément de $\mathcal{S}_k$; de plus : $\mathcal{S}_k \subseteq \mathcal{S}_{k+1}$, et $\mathcal{F}'_k \subseteq \mathcal{F}_{k+1}$.*

Voir Appendix B.

4.2 Techniques numériques d'identification du front de Pareto

4.2.1 Généralités, algorithmes génétiques

D'une manière générale, dans le cas de fonctions régulières et (au moins localement) convexes, les techniques d'optimisation différentiable peuvent être utilisées efficacement, au prix du calcul des gradients discrets ou continus. Dans ce cas, comme on l'a mentionné précédemment, il convient de les complémenter d'un algorithme de tri global pour une vision globale du front.

Par contre, dans le cas de fonctions non continues et/ou non convexes, l'optimisation différentiable est souvent mise en défaut par manque de robustesse. On peut alors appliquer une stratégie (semi-) stochastique : algorithme génétique (GA), par essaim de particules (PSO), etc.

L'algorithme génétique le plus couramment utilisé est connu sous le nom de NSGA-II : *Nondominated Sorting Genetic Algorithm, Srinivas & Deb, 1994* (algorithme génétique s'appuyant sur le tri des solutions non dominées). À la Figure 4.8, on en illustre le potentiel à identifier trois branches du front de Pareto du cas-test de Kursawe. (Tiré de : *A Fast and Elitist Multiobjective Genetic Algorithm : NSGA-II*, K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, IEEE Transactions in Evolutionary Computation, Vol. 6, No. 2, April 2002.) Le front est ici ni convexe, ni continu. Au regard de la Proposition 1, la non-convexité du front révèle le défaut de convexité d'au moins une fonction-coût dans le voisinage de l'ensemble de Pareto.

$$n = 3,\ m = 2$$

$$f_1(x) = \sum_{i=1}^{n-1} \left(-10 \exp \left(-\frac{\sqrt{x_i^2 + x_{i+1}^2}}{5} \right) \right), \quad f_2(x) = \sum_{i=1}^{n} \left(|x_i|^{\frac{4}{5}} + 5 \sin x_i^3 \right)$$

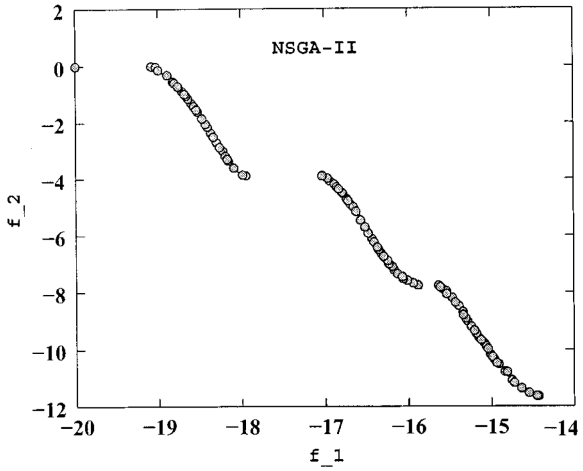


FIGURE 4.8 – Solutions non-dominées du cas-test de Kursawe identifiées par NSGA II

Le principe de fonctionnement de l'algorithme NSGA II consiste à une itération donnée à effectuer les opérations suivantes :

1. Tri des éléments de la population en différents fronts de solutions 2 à 2 non-dominées ; affectation à chaque front d'une valeur d'adaptation égale au numéro du front, éventuellement corrigée par une procédure de "niche" pour pénaliser les agglomérats ;
2. Évolution de la population à la génération suivante par un algorithme génétique standard (évaluation, croisement, mutation).

Le principe du tri par fronts est illustré à la Figure 4.9.

Pour mémoire, citons les stratégies évolutionnaires alternatives suivantes s'apparentant à NSGA II :

— NPGA : Niched Pareto Genetic Algorithm, Goldberg et al, 1991 [34] ;
— MOGA : Multiobjective Genetic Algorithms, Murata & Hishibuchi, 1995 [49] ;
— SPEA : Strength Pareto Evolutionary Algorithm, Zitzler & Thiele, 1999 [59] ;
— PAES : Pareto Archived Evolution Strategy, Knowles & Corne, 1999 [38] ;
— PSO : Particle-Swarm Optimization, Parsopoulos & Vrahatis, 2002 [50].

On illustre maintenant l'application de l'algorithme NSGA II à un problème d'optimisation de forme bicritère en aérodynamique. On considère un simple profil d'aile immergé dans un écoulement compressible stationnaire régi par les équations d'Euler en deux dimensions d'espace. La forme du profil est paramétrisée par deux courbes de Bézier dont on optimise les points de contrôle par leurs coordonnées. Chaque écoulement est calculé par une méthode de volumes-finis en maillage non-structuré.

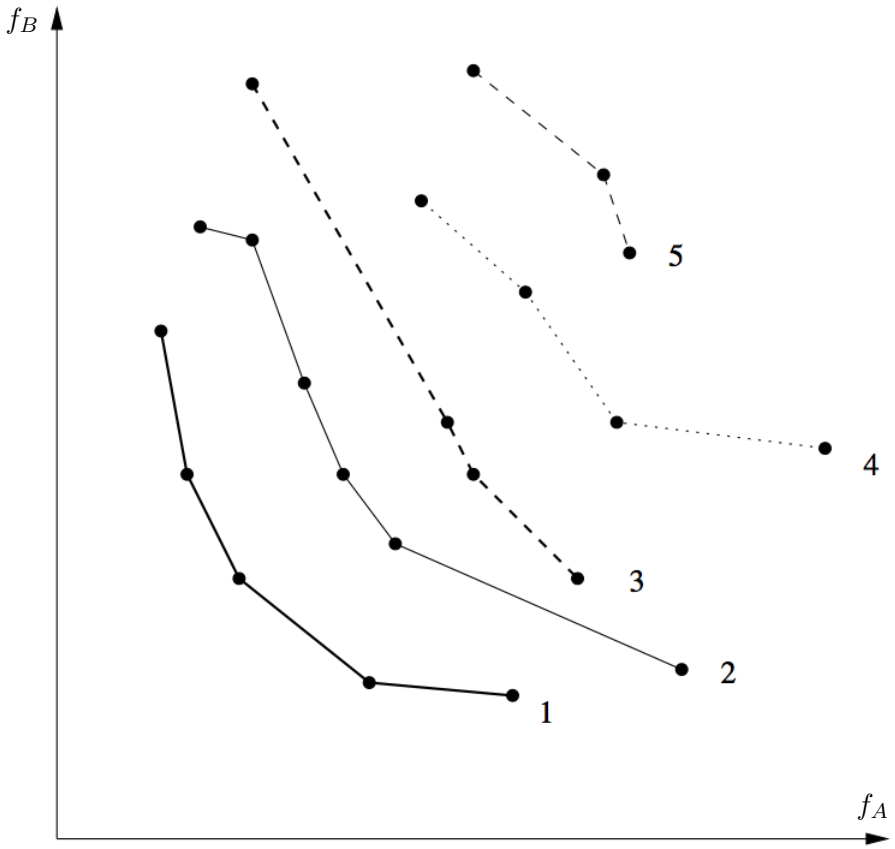


FIGURE 4.9 – Illustration du tri en différents fronts de l'image dans l'espace de deux fonctions (f_A, f_B) d'une population de solutions, et de l'affectation d'une valeur d'adaptation égale à l'indice de front

À chaque nouvelle définition des paramètres de forme correspond un nouveau profil, légèrement déformé par rapport au précédent. On recalcule le nouveau maillage extérieur au profil par déformation. La déformation est définie par analogie structurale en affectant à chaque arrête du maillage un ressort virtuel et en calculant les nouvelles positions des nœuds par résolution des équations d'équilibre de l'élasticité linéaire. L'écoulement est recalculé en s'appuyant sur ce nouveau maillage.

Une géométrie donnée est évaluée par deux fonctions correspondant à deux situations distinctes de l'enveloppe de vol :

— Situation représentative du décollage ou de l'atterrissage : écoulement subsonique à forte incidence (modélisation de l'actuation de volets hyper-sustentateurs), $M_\infty = 0.3$, $\alpha = AoA = 40^o$; critère lié au coefficient de portance C_L (fonction à minimiser $f_1 = -C_L$, à une certaine échelle) ;

— Situation représentative de la croisière : écoulement transsonique à faible incidence, $M_\infty = 0.8$, $\alpha = AoA = 1^o$; critère lié au coefficient de traînée C_D (fonction à minimiser $f_2 = C_D$, à une certaine échelle).

(Référence : Marco *et al*, Rapport de Recherche Inria, No. 3686 (1999), `https://hal.inria.fr/inria-00072983` .)

L'expérience numérique d'application de l'algorithme d'optimisation multicritère NSGA II a été réalisée deux fois avec des maillages de finesses différentes. Pour chaque maillage, on a calculé un grand nombre de générations et reporté les résultats à la Figure 4.10 où sont représentés à gauche les valeurs de fonctions, et à droite le front de Pareto pour les deux expériences.

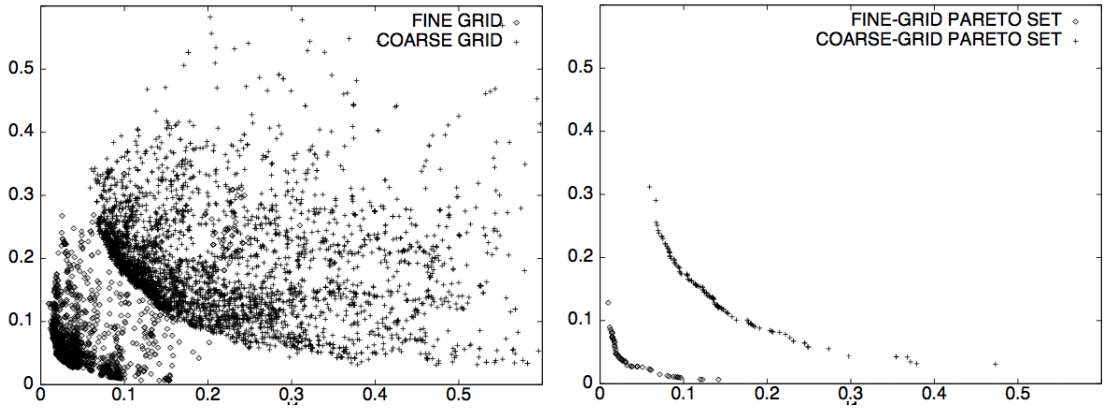


FIGURE 4.10 – Valeurs de fonctions au cours des générations de NSGA II (à gauche), fronts de Pareto (à droite)

L'expérience montre accessoirement l'influence de la finesse du maillage sur la qualité du résultat d'optimisation.

Enfin on illustre à la Figure 4.11 les profils corrrespondant aux solutions optimales du calcul en maillage fin. Ces solutions évoluent entre un profil épais à forte portance optimal en régime subsonique (en haut à gauche) et un profil mince à faible traînée optimal en régime transsonique (en bas à droite).

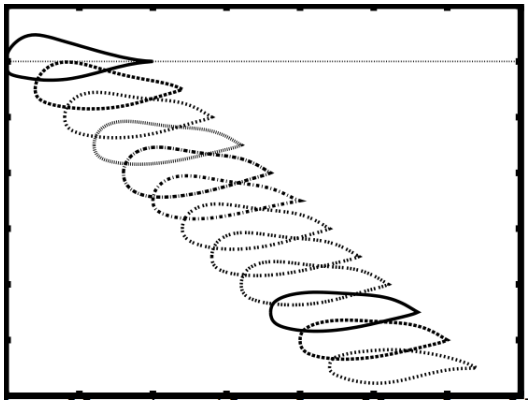


FIGURE 4.11 – Profils Pareto-optimaux vis-à-vis du problème d'optimisation concurrente de maximisation de portance en subsonique et de minimisation de traînée en transsonique

Au regard de cette expérience numérique, on peut identifier les principaux mérites et inconvénients de l'approche par algorithme génétique.

Principaux mérites :

— La stratége est très générale, très robuste : elle s'applique aux cas non convexes, non continus.
— L'algorithme est mis en œuvre facilement et rapidement dès lors que l'on a rendu disponible la procédure d'évaluation des fonctions en fonction des variables. La boucle d'optimisation est extérieure à l'application, simple et générique. Des résultats grossiers d'optimisation sont rapidement obtenus. Si le temps de calcul est élevé, le temps d'ingénierie par contre est faible : aucune procédure complexe, dépendant du problème n'est nécessaire.
— Une importante source d'information compilée par NSGA II est disponible au concepteur.

Principaux inconvénients :
— La technique est très coûteuse en nombre d'évaluations de fonctions, seule source d'information utilisée par l'algorithme.
— Les résultats d'optimisation sont grossiers. Le bruit joue un rôle important dans la robustesse de l'algorithme génétique, mais en réduit la précision, en particulier en codage binaire. En conséquence, un résultat fin d'optimisation requiert l'hybridation de l'algorithme génétique : la technique la plus simple consiste à appliquer une technique d'optimisation différentiable *a posteriori* pour affiner les résultats grossiers de l'AG.

4.2.2 Autres approches classiques de traitement de problèmes multi-objectifs

a) Technique du critère aggloméré
On choisit des poids $\{w_j\}$ et on traite le problème d'optimisation monoobjectif de minimisation du critère aggloméré suivant :

$$f(\mathbf{x}) = \sum_{j=1}^{m} w_j f_j(\mathbf{x})\,. \tag{4.14}$$

Cette approche est la plus couramment utilisée dans les applications. Elle souffre néanmoins d'un manque de généralité. Le choix des poids $\{w_j\}$ est très arbitraire. Si les fonctions $\{f_j(\mathbf{x})\}$ ont des origines physiques différentes, même sous forme adimensionnée, leurs échelles sont diffcilement comparables, donc "agglomérables". La calibration du problème par le praticien est donc un impératif.

Cependant cette approche est simple. Par ailleurs elle permet, à partir d'un point de départ admissible, de calculer une évolution dans le sens de l'amélioration des critères, au moins en moyenne. Dans les cas pratiques où le point de départ est par expérience, très proche de l'optimimalité, cette amélioration, si elle est sensible, peut constituer un progrès réel.

Par ailleurs, on peut améliorer la technique si les poids sont rationnellement choisis, notamment sur la base d'une analyse locale de sensibiltés [2].

b) Formulation apparentée :

1. Résoudre d'abord les m problèmes d'optimisation monoobjectif suivants :

$$f_j^\star = \min f_j(\mathbf{x}) \quad (\forall j = 1, \ldots, m) \tag{4.15}$$

(dont on suppose qu'ils sont individuellement bien posés).

2. *Aircraft Shape Optimization for Mission Performance*, F. Gallard, Thèse de doctorat, Université de Toulouse, INP de Toulouse (Spécialité : *AA : Aéronautique et Astronautique)*.

2. Choisir des poids $\{w_j\}$ de mêmes dimensions que $\{f_j\}$ (respectivement) et résoudre le problème d'optimisation monobjectif multicontraint suivant :

$$\min T \qquad \text{sous les contraintes : } f_j \leq f_j^\star + w_j T \quad (\forall j). \tag{4.16}$$

Dans cette formulation, on cherche à minimiser la dégradation des critères exprimée par les valeurs relatives $q_j = (f_j - f_j^\star)/w_j$.

Cette formulation, sans-doûte plus élégante, est néanmoins entâchée du même degré d'arbitraire que la précédente en raison du choix nécessaire des poids, et de la difficulté de comparer les quotients $\{q_j\}$ lorsque les fonctions $\{f_j\}$ sont de natures physiques différentes. Par ailleurs, elle soulève une difficulté nouvelle : celle de traiter des contraintes fonctionnelles lorsque les fonctions ne sont pas définies par des expressions formelles des variables, mais par le biais d'une chaîne fonctionnelle d'opérateurs, et en pratique, de procédures numériques.

c) Minimisation par traitement de $m-1$ critères en contraintes d'égalité. On obtient des points Pareto-optimaux en résolvant des problèmes du type suivant :

$$\min f_j(\mathbf{x}) \quad \text{sous les contraintes : } f_k(\mathbf{x}) = \alpha_k \quad (\forall k \neq j). \tag{4.17}$$

Cette méthode est illustrée à la Figure 4.12 dans le cas de deux critères.

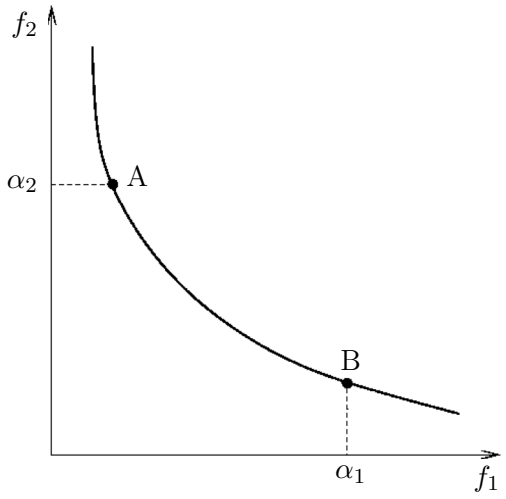


FIGURE 4.12 – Illustration de la méthode d'optimisation biobjectif par traitement d'un critère par une contrainte d'égalité. Le point A (resp. B) réalise le minimum de f_1 (resp. f_2) sous la contrainte $f_2 = \alpha_2$ (resp. $f_1 = \alpha_1$).

Cette méthode présente les inconvénients majeurs suivants :
— Nécessité de traiter des contraintes sur les fonctions.
— Complexité algorithmique lorsque le nombre de critères est grand.
— Méthode problématique lorsque le front n'est pas convexe, ou discontinu.

d) Méta-modèles, multi-fidélité, "Optimisation Multi-Disciplinaire" (MDO)

- Pour chaque discipline A, B, ... on construit une hiérarchie de modèles (notion de multi-fidélité). Le(s) modèle(s) les plus précis étant le(s) plus coûteux à évaluer.

  Les modèles économiques moins précis peuvent être le résultat
  - *d'une modélisation physique plus simple ; par exemple, en aérodynamique, les équations du potentiel complet représentent un modèle simplifié des équations d'Euler, qui, elles-mêmes constituent une simplification des équations de Navier-Stokes ;*
  - *d'un traitement statistique d'une base données de valeurs du modèle complexe précis ; par exemple par Décomposition Orthogonale Propre (POD), krigeage, réseau de neurones, surfaces de réponse, etc.*
- On conçoit une stratégie d'analyse/optimisation en traitant qu'un seul modèle complexe à la fois, en introduisant la complexité graduellement, de telle sorte qu'à convergence, la solution optimale associe les modèles complexes de toutes les disciplines, alors que les étapes intermédiaires ont été réalisées par des évaluations de modèles plus économiques.
  - BLISS
    "Bilevel integrated system synthesis for concurrent and distributed processing", J. Sobieszczanski-Sobieski, T. Altus, and R.R. Sandusky, *AIAA J. 41, 1996 ?2003* (2003)
  - EGO
    Efficient Global Optimization of Expensive Black-Box Functions, Donad R. Jones, Matthias Schonlau and William J. Welch, Journal of Global Optimization **13** : 455-492, 1998.

Ces techniques ont notamment été analysées et approfondies dans le cadre du projet de l'ANR (Agence Nationale pour la Recherche), "OMD" (Optimisation Multi-Disciplinaire) coordonné par l'École des Mines de St Étienne. Ces travaux ont donné lieu aux publications suivantes :

- *Optimisation multidisplinaire en mécanqiue 1 et 2*, Hermès Lavoisier, Paris (2009).
- *Multidiciplinary Design Optimization in Computational Mechanics*, Piotr Breitkopf & Rajan Filomeno Coelho Eds., ISTE Ltd and John Wiley & Sons, Inc., 2010. `http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1848211384.html`

Enfin, notons que les techniques d'optimisation du type d'EGO dans lesquelles une base de données est itérativement enrichie et sert à faire évoluer un modèle grossier économique de plus en plus précis à l'approche de la solution, se combinent et s'adaptent efficacement aux algorithmes efficaces de calcul parallèle. (Voir par exemple le site personnel du Pr. K. Giannakoglou : `http://velos0.ltt.mech.ntua.gr/research/`.)

e) Stratégies se référant à la théorie de la décision

Ces stratégies ont été notamment développées par l'école finlandaise [46] [32] [33].

f) Stratégies de jeux dynamiques

On associe à chaque fonction (discipline) un joueur virtuel dont l'objectif (ou stratégie) est la minimisation de cette fonction sous la contrainte des stratégies des autres joueurs. Chaque joueur contrôle tout ou partie des variables d'optimisation. On construit une dynamique virtuelle de mise en concurrence des stratégies, de telle sorte qu'un équilibre est atteint. Cet équilibre réalise une solution de compromis entre les minimisations monodisciplinaires concourantes.

Cette stratégie de jeux permet donc de découpler le problème d'optimisation multi-objectif en plusieurs sous-problèmes monodisciplinaires. Chaque sous-problème est généralement associé à une modélisation physique indépendante, donc un code de simulation indépendant, et un optimiseur indépendant. Le couplage se réalise seulement par échange de variables. Il en résulte une grande simplification algorithmique qui a notamment été abondamment exploitée dans la résolution de problèmes en aéronautique [51]. Cependant ce découplage a un coût : en général, la solution d'équilibre n'est pas Pareto-optimale.

On distingue principalement les jeux jeux symétriques de Nash et les jeux hiérarchisés de Stackelberg, dont on donne ici une définition rapide dans le cas de deux fonctions $f_1,\ f_2$, deux joueurs $J_1,\ J_2$ et une partition de la variable d'optimisation $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2)$ en deux sous-vecteurs $\mathbf{x}_1$, sous le contrôle de J_1, et $\mathbf{x}_2$, sous le contrôle de J_2.

Dans le problème à deux disciplines, on dit que $\bar{\mathbf{x}} = (\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2)$ réalise un équilibre de Nash si et seulement si $\bar{\mathbf{x}}_1$ réalise un minimum local de la fonction partielle $\phi_1(\mathbf{x}_1) = f_1(\mathbf{x}_1, \bar{\mathbf{x}}_2)$ et $\bar{\mathbf{x}}_2$ réalise un minimum local de la fonction partielle $\phi_2(\mathbf{x}_2) = f_2(\bar{\mathbf{x}}_1, \mathbf{x}_2)$:

$$\bar{\mathbf{x}}_1 = \arg\min f_1(\mathbf{x}_1, \bar{\mathbf{x}}_2) \text{ et } \bar{\mathbf{x}}_2 = \arg\min f_2(\bar{\mathbf{x}}_1, \mathbf{x}_2)\,. \tag{4.18}$$

Dans le jeu de Stackelberg, convenons que J_1 est le meneur (*leader*) et J_2 le suiveur (*follower*). Pour chaque valeur de $\mathbf{x}_1$ qui traduit la stratégie du meneur J_1, le joueur suiveur J_2 cale $\mathbf{x}_2$ de manière à minimiser la fonction partielle $\phi_{\mathbf{x}_1}(\mathbf{x}_2) = f_2(\mathbf{x}_1, \mathbf{x}_2)$. En conséquence, $\mathbf{x}_1$ est optimisé de manière à minimiser

$$\psi(\mathbf{x}_1) = f_1\big(\mathbf{x}_1, \arg\min \phi_{\mathbf{x}_1}(\mathbf{x}_2)\big)\,. \tag{4.19}$$

Les jeux de Nash font l'objet de la section ultérieure 4.4. Des applications en conception optimale de forme aérodynamique y sont présentées au 4.4.6.

4.3 Algorithme de descente à gradients multiples (MGDA)

4.3.1 Lemmes fondamentaux

Sont donnés :

- le domaine admissible, l'ouvert $\Omega_{ad} \subset \mathbb{R}^n$;
- les fonctions-objectifs, ou critères, $f_j(\mathbf{x})$ $(\mathbf{x} \in \Omega_{ad}\,;\ j = 1, \ldots, m)$ à minimiser, différentiables dans Ω_{ad} ; en pratique on fait généralement l'hypothèse plus forte $f_j \in C^2(\Omega_{ad})$;
- la matrice $\mathbf{A}_n$, réelle-symétrique définie-positive associée au produit scalaire dans $\mathbb{R}^n$:

$$\big(\mathbf{x}, \mathbf{y}\big) = \mathbf{x}^t \mathbf{A}_n \mathbf{y}\,. \tag{4.20}$$

Considérons une famille de vecteurs $\{\mathbf{u}_j\}$ $(\mathbf{u}_j \in \mathbb{R}^n\,;\ j = 1, \ldots, m)$. On rappelle la

Définition 7 (Enveloppe convexe)
On appelle enveloppe convexe de la famille de vecteurs $\{\mathbf{u}_j\}$ $(\mathbf{u}_j \in \mathbb{R}^n\,;\ j = 1, \ldots, m)$*, l'ensemble des vecteurs de* $\mathbb{R}^n$ *qui peuvent s'exprimer par au moins une combinaison convexe des éléments de la famille :*

$$\overline{\mathsf{U}} = \left\{ \mathbf{u} \in \mathbb{R}^n \ / \ \mathbf{u} = \sum_{j=1}^{m} \alpha_j \mathbf{u}_j;\ \alpha_j \geq 0\ (\forall j);\ \sum_{j=1}^{m} \alpha_j = 1 \right\}. \tag{4.21}$$

Noter que si la famille $\{\mathbf{u}_j\}$ est linéairement dépendante, il est possible que les coefficients $\{\alpha_j\}$ de la combinaison convexe ne soient pas uniques.

Dans une représentation affine de $\mathbb{R}^n$ où les vecteurs sont associés à des bipoints de même origine O, $\overline{\mathsf{U}}$ est associé à un polyhèdre ayant au plus m sommets. Topologiquement, $\overline{\mathsf{U}}$ est donc un fermé-borné. Bien évidemment, $\overline{\mathsf{U}}$ est également convexe, car si $\mathbf{u} \in \overline{\mathsf{U}}$ et $\mathbf{v} \in \overline{\mathsf{U}}$:

$$\begin{cases} \mathbf{u} = \sum_{j=1}^{m} \alpha_j \mathbf{u}_j ;\ \alpha_j \geq 0\ (\forall j);\ \sum_{j=1}^{m} \alpha_j = 1 \\ \mathbf{v} = \sum_{j=1}^{m} \beta_j \mathbf{u}_j ;\ \beta_j \geq 0\ (\forall j);\ \sum_{j=1}^{m} \beta_j = 1 \end{cases} \tag{4.22}$$

alors, $\forall \varepsilon \in [0,1]$:

$$\mathbf{w} := (1-\varepsilon)\mathbf{u} + \varepsilon \mathbf{v} = \sum_{j=1}^{m} \gamma_j \mathbf{u}_j \,, \quad \gamma_j = (1-\varepsilon)\alpha_j + \varepsilon \beta_j \,. \tag{4.23}$$

Par conséquent :

$$\gamma_j \geq 0\ (\forall j);\ \sum_{j=1}^{m} \gamma_j = 1 \,, \tag{4.24}$$

ce qui confirme que $\mathbf{w} \in \overline{\mathsf{U}}$. □

Du fait que $\overline{\mathsf{U}}$ soit un fermé, toute fonction continue minorée y admet un minimum (réalisé une ou plusieurs fois). C'est le cas en particulier de la norme associée au produit scalaire considéré :

$$\arg\min_{\mathbf{u} \in \overline{\mathsf{U}}} \|\mathbf{u}\| \neq \emptyset \,. \tag{4.25}$$

De plus, en raison de la convexité, le minimum est atteint en un unique (bi)point (vecteur) $\omega^\star$.

Preuve (de l'unicité) : supposons que le minimum soit atteint en deux points ω_1 et ω_2 et posons :

$$\overline{\overline{\omega}} = \|\omega_1\| = \|\omega_2\| = \min_{\mathbf{u} \in \overline{\mathsf{U}}} \|\mathbf{u}\| \ . \tag{4.26}$$

Posons :

$$\omega_s = \tfrac{1}{2}(\omega_1 + \omega_2) \,, \quad \omega_d = \tfrac{1}{2}(\omega_2 - \omega_1) \,, \tag{4.27}$$

et notons que $\omega_s \in \overline{\mathsf{U}}$ de sorte que $\|\omega_s\| \geq \overline{\overline{\omega}}$. Il vient :

$$\big(\omega_s, \omega_d\big) = \tfrac{1}{4}\left(\|\omega_2\|^2 - \|\omega_1\|^2\right) = 0 \tag{4.28}$$

de sorte que $\omega_s \perp \omega_d$. Or

$$\overline{\overline{\omega}}^2 = \|\omega_2\|^2 = \|\omega_s + \omega_d\|^2 = \|\omega_s\|^2 + \|\omega_d\|^2 \geq \overline{\overline{\omega}}^2 + \|\omega_d\|^2 \tag{4.29}$$

et ceci exige que $\omega_d = 0$, ce qui établit l'unicité. □

On rassemble ces observations dans le

Lemme 1 (Existence et unicité d'un élément de $\overline{\mathsf{U}}$ de plus petite norme)
L'enveloppe convexe $\overline{\mathsf{U}}$ admet un unique élément de plus petite norme :

$$\boxed{\exists!\ \omega^\star = \arg\min_{\mathbf{u}\in\overline{\mathsf{U}}} ||\mathbf{u}||} \tag{4.30}$$

L'élément $\omega^\star$ ainsi identifié admet la propriété suivante :

Lemme 2 (Propriété fondamentale de l'élément $\omega^\star$)

$$\boxed{\forall \mathbf{u} \in \overline{\mathsf{U}},\ \big(\mathbf{u}, \omega^\star\big) \geq ||\omega^\star||^2} \tag{4.31}$$

Preuve : Soit $\mathbf{u} \in \overline{\mathsf{U}}$ quelconque. Posons $\mathbf{v} = \mathbf{u} - \omega^\star$. Par convexité :

$$\forall \theta \in [0,1],\ (1-\theta)\omega^\star + \theta\mathbf{u} = \omega^\star + \theta\mathbf{v} \in \overline{\mathsf{U}}\,. \tag{4.32}$$

Par conséquent, puisque $\omega^\star$ est l'élément de plus petite norme :

$$||\omega^\star + \theta\mathbf{v}||^2 = \big(\omega^\star + \theta\mathbf{v}, \omega^\star + \theta\mathbf{v}\big) \quad \geq \quad ||\omega^\star||^2 = \big(\omega^\star, \omega^\star\big)\,. \tag{4.33}$$

On développe les produits scalaires et on ordonne suivant les puissances de θ :

$$2\theta\big(\omega^\star, \mathbf{v}\big) + \theta^2 \left\|\mathbf{v}\right\|^2 \geq 0\,, \tag{4.34}$$

et ceci exige que l'on ait :

$$\big(\omega^\star, \mathbf{v}\big) \geq 0\,, \tag{4.35}$$

ce qui fournit le résultat. □

4.3.2 Application

Supposons que les $\{\mathbf{u}_j\}$ soient les gradients des critères en un point de départ donné $\mathbf{x}_0$:

$$\mathbf{u}_j = \nabla f_j(\mathbf{x}_0) \qquad (j = 1, \ldots, m)\,. \tag{4.36}$$

Posons :

$$\boxed{\mathbf{d} = \mathbf{A}_n\omega^\star} \tag{4.37}$$

Il vient :

$$\big(\mathbf{u}_j, \omega^\star\big) = \mathbf{u}_j^t\mathbf{A}_n\omega^\star = \nabla f_j(\mathbf{x}_0)^t\mathbf{d} \geq ||\omega^\star||^2 \geq 0\,. \tag{4.38}$$

On reconnaît à gauche de l'inégalité la dérivée du critère $f_j(\mathbf{x})$ dans la direction du vecteur $\mathbf{d}$ (au facteur $\|\omega^\star\|$ près). Il en résulte le

Proposition 3
En $\mathbf{x}_0$, la direction $-\mathbf{d}$ est une direction de descente commune à tous les critères.

On peut donc généraliser la méthode du gradient en prenant $-\mathbf{d}$ comme direction de descente :

$$\boxed{\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \epsilon \mathbf{d}} \tag{4.39}$$

où $\epsilon > 0$ est le pas. Si $\omega^\star \neq 0$, et si le pas est suffisamment petit, tous les critères diminuent. Cette itération converge vers un point où $\omega^\star = 0$, dit point "Pareto-stationnaire". Si le domaine admissible est ouvert, et les critères localement convexes, la condition de Pareto-stationnarité est la condition nécessaire (mais pas suffisante) de Pareto-optimalité (voir sous-section suivante).

4.3.3 Notion de Pareto stationnarité

Notre développement nous conduit à poser la définition suivante :

Définition 8 (Pareto-stationnarité)
Soient m fonctions-objectifs, $f_j(\mathbf{x})$, de domaine admissible $\Omega_{ad} \subset \mathbb{R}^n$, différentiables dans la boule ouverte $\mathcal{B}$ de centre $\mathbf{x}_0$ contenue dans Ω_{ad}. On dit que $\mathbf{x}_0$ est un point Pareto-stationnaire ssi il existe une combinaison convexe des gradients, $\{\nabla f_j(\mathbf{x}_0)\}$, égale à 0 :

$$\exists \alpha = \{\alpha_j\} \in \mathbb{R}_+^m \text{ tel que : } \sum_{j=1}^{m} \alpha_j \nabla f_j(\mathbf{x}_0) = 0\,,\ \sum_{j=1}^{m} \alpha_j = 1\,. \tag{4.40}$$

On voit immédiatement que la condition de Pareto-stationnarité en $\mathbf{x}_0$ équivaut à la condition $\omega^\star = 0$ en ce point (propriété caractéristique).

Nous allons maintenant montrer que la relation entre Pareto-stationnarité (ainsi définie) et Pareto-optimalité généralise au cas multiobjectif celle classique en optimisation monoobjectif entre stationnarité et optimimalité en établissant le théorème suivant :

Proposition 4
(On fait les hypothèses de régularité de la definition 8.) Si le point $\mathbf{x}_0$ est Pareto-optimal, et si les fonctions-objectifs sont convexes dans $\mathcal{B}$, alors, $\mathbf{x}_0$ est Pareto-stationnaire.

Preuve : Pour alléger l'écriture on pose $\mathbf{u}_j = \nabla f_j(\mathbf{x}_0)$. Sans perte de généralité on suppose que $f_j(\mathbf{x}_0) = 0$ $(\forall j)$.

Puisque le point $\mathbf{x}_0$ est Pareto-optimal par hypothèse, $\mathbf{x}_0$ minimise l'un quelconque des critères (spécifiquement $f_m(\mathbf{x})$ dans ce qui suit), sous la contrainte de non-dépassement par les autres critères de leurs valeurs en $\mathbf{x}_0$ (c'est-à-dire 0) :

$$\mathbf{x}_0 \in \left\{ \arg\min_{\mathbf{x}} f_m(\mathbf{x}) \ / \text{ sous les contraintes : } f_j(\mathbf{x}) \leq 0 \ (\forall j \leq m-1) \right\} \tag{4.41}$$

où ici l'"arg min" représente l'ensemble des points réalisant le minimum contraint, qui ne se réduit pas forcément au seul point $\mathbf{x}_0$.

Soit $\overline{\mathsf{U}}_{m-1}$ l'enveloppe convexe des $m-1$ gradients $\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{m-1}\}$ et

$$\omega_{m-1}^\star = \arg \min_{\mathbf{u} \in \overline{\mathsf{U}}_{m-1}} \|\mathbf{u}\| \tag{4.42}$$

dont on a déjà établi l'existence, l'unicité et la propriété suivante :

$$\left(\mathbf{u}_j, \omega^\star_{m-1}\right) \geq \left\|\omega^\star_{m-1}\right\|^2 \ (\forall j \leq m-1)\,. \tag{4.43}$$

Deux situations sont alors possibles :

1. Ou bien $\omega^\star_{m-1} = 0$, et la condition de Pareto-stationnarité en $\mathbf{x}_0$ est satisfaite par les gradients des seules fonctions-objectifs $\{f_1, f_2, \ldots, f_{m-1}\}$, et *a fortiori* par l'ensemble d'entre elles.
2. Sinon $\omega^\star_{m-1} \neq 0$. On pose alors $\phi_j(\varepsilon) = f_j(\mathbf{x}^0 - \varepsilon\omega^\star_{m-1})$ $(j = 1, \ldots, m-1)$ de sorte que $\phi_j(0) = 0$ et $\phi'_j(0) = -\left(\mathbf{u}_j, \omega^\star_{m-1}\right) \leq -\left\|\omega^\star_{m-1}\right\|^2 < 0$, et pour ε suffisamment petit, on a :
$$\phi_j(\varepsilon) = f_j(\mathbf{x}^0 - \varepsilon\omega^\star_{m-1}) < 0 \qquad (\forall j \leq m-1)\,. \tag{4.44}$$
Ce résultat montre que pour le problème de minimisation sous contrainte (4.41), la *condition de qualification des contraintes de Slater*[3] est satisfaite, de sorte que l'optimalité exige la satisfaction des conditions de Karush-Kuhn-Tucker, c'est-à-dire notamment la stationnarité du lagrangien
$$\mathsf{L} = f_m(\mathbf{x}) + \sum_{j=1}^{m-1} \lambda_j f_j(\mathbf{x}) \tag{4.45}$$
et ceci donne
$$\mathbf{u}_m + \sum_{j=1}^{m-1} \lambda_j \mathbf{u}_j = 0 \tag{4.46}$$
où $\lambda_j > 0$ $(\forall j \leq m-1)$ en raison de la saturation des contraintes $(f_j(\mathbf{x}^0) = 0)$ et de la convention de signe adoptée pour les exprimer ; donc $\Lambda = 1 + \sum_{j=1}^{m-1} \lambda_j > 1$. En divisant l'équation précédente par $\Lambda \neq 0$, le résultat est acquis.

□

D'un point de vue algorithmique, ce résultat montre qu'un algorithme de descente multiobjectif, en optimisation différentiable, par nature local, lorsqu'il converge, converge naturellement vers un point Pareto-stationnaire. L'identification du front de Pareto à partir de l'ensemble des points Pareto-stationnaires exige un élément méthodologique supplémentaire : un algorithme global de tri.

Conclusion générale : Soit $\mathbf{x}_0$ un point du domaine admissible Ω_{ad}, centre d'une boule ouverte $\mathcal{B} \subset \Omega_{ad}$ dans laquelle les fonctions-objectifs sont différentiables et convexes. Alors :

— Si le point $\mathbf{x}_0$ est "Pareto-optimal" (dans $\mathcal{B}$), il est "Pareto-stationnaire" au sens de l'une des assertions suivantes qui sont équivalentes :

1. Il existe une combinaison convexe des gradients égale à 0 :
$$\sum_{j=1}^{n} \nabla f_j(\mathbf{x}_0) = 0\,;\ \alpha_j \geq 0\ (\forall j)\,;\ \sum_{j=1}^{n} \alpha_j = 1\,. \tag{4.47}$$

3. Voir : *Boyd, S. ; Vandenberghe, L. (2004) (pdf). Convex Optimization. Cambridge University Press. ISBN 978-0-521-83378-3. Retrieved October 3, 2011]*, ou l'annexe de ce document.

2. L'élément de plus petite norme $\omega^\star$ est nul ($\omega^\star = \mathbf{d} = 0$).
3. Le vecteur nul de $\mathbb{R}^n$ appartient à l'enveloppe convexe des gradients : $0 \in \overline{\mathsf{U}}$.

— Si, à l'inverse, $\omega^\star \neq 0$, $\mathbf{d} = \mathbf{A}_n\omega^\star \neq 0$, et $-\mathbf{d}$ est une direction de descente commune à tous les critères.

(S'en convaincre !)

4.3.4 Formulation de programmation quadratique

Plusieurs techniques peuvent être appliquées pour identifier l'élément $\omega^\star$ et font l'objet d'études actuelles. Une manière très naturelle, mais pas toujours la plus efficace, consiste à chercher le vecteur de coefficients $\alpha = \{\alpha_j\}$. Si $\mathbf{U}$ est la matrice $N \times n$ dont les colonnes sont les vecteurs $\mathbf{u}_j$, il vient $\omega^\star = \mathbf{U}\alpha^\star$:

$$\begin{cases} \alpha^\star = \arg\min\limits_{\alpha\in\mathbb{R}^m} \alpha^t \left(\mathbf{U}^t\mathbf{A}_n\mathbf{U}\right)\alpha \\ \alpha \geq 0 \\ \sum\limits_{j=1}^{m} \alpha_j = 1 \end{cases} \tag{4.48}$$

La matrice

$$\mathbf{Q} = \mathbf{U}^t\mathbf{A}_n\mathbf{U} \tag{4.49}$$

est réelle-symétrique semi-définie-positive. Il s'agit donc d'un cas particulier de formulation classique de **programmation quadratique** (QP) dans laquelle on minimise une forme quadratique semi-définie positive sous des contraintes de bornes sur les inconnues ($\alpha_j \geq 0\,,\ \forall j$), et une contrainte d'égalité linéaire ($\sum_j \alpha_j = 1$).

On peut notamment résoudre ce problème QP au moyen de la procédure quadprog de MATLAB, ou qpsolve de SCILAB. On sait que la solution existe (puisque $\omega^\star$ existe), mais le problème peut être numériquement mal conditionné si m ou n est grand, ce qui justifie le développement qui suit.

4.3.5 Algorithme par orthogonalisation des gradients

Cas d'une famille de gradients linéairement indépendante

On commence par examiner le cas encore plus spécifique où les gradients sont orthogonaux vis-à-vis du produit scalaire euclidien usuel. Cette situation est illustrée à la Figure 4.13 dans le cas $m = 3$. Dans cette représentation de l'espace affine (et non vectoriel) euclidien $\mathbb{R}^3$, l'enveloppe convexe $\overline{\mathsf{U}}$ est associée aux bipoints d'origine O dont les extrémités appartiennent au triangle ayant pour sommets les extrémités des bipoints représentatifs des vecteurs $\mathbf{u}_1$, $\mathbf{u}_2$ and $\mathbf{u}_3$. Soit $(\mathcal{P})$ le plan du triangle, et $O^\perp$ la projection orthogonale du point O dans le plan $(\mathcal{P})$. Par définition, le point $O^\perp$ est le point du plan $(\mathcal{P})$ le plus proche de O au sens de la métrique usuelle. Mais, en vertu de l'orthogonalité des gradients, les 3 secteurs angulaires plans $(\mathbf{u}_1, \mathbf{u}_2)$, $(\mathbf{u}_2, \mathbf{u}_3)$ et $(\mathbf{u}_3, \mathbf{u}_1)$ sont orthogonaux et délimitent le dièdre positif exactement et en totalité. Ces plans orthogonaux s'appuient sur les arêtes du triangle, de sorte que la projection orthogonale $O^\perp$ appartient obligatoirement au triangle. Par conséquent, le point $O^\perp$ est également de point du triangle le plus proche de O, et

$$\omega^\star = \overrightarrow{OO^\perp}\,. \tag{4.50}$$

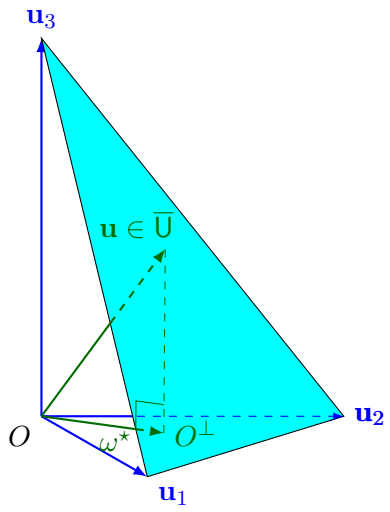


FIGURE 4.13 – Cas où les vecteurs gradients sont orthogonaux, et $m = 3$; l'orthogonalité est définie vis-à-vis du produit scalaire euclidien usuel ; l'enveloppe convexe $\overline{\mathsf{U}}$ est associée au triangle bleu, $O^\perp$ est la projection orthogonale de l'origine O dans le plan du triangle ; $\omega^\star = \overrightarrow{OO^\perp}$.

Donc, dans le cas où les vecteurs gradients sont orthogonaux, et quel que soit m, on identifie $\omega^\star$ par projection orthogonale de O, ce qui donne le résultat suivant

$$\omega^\star = \sum_{j=1}^{m} \alpha_j \mathbf{u}_j, \;\; \alpha_j = \frac{\frac{1}{\|\mathbf{u}_j\|^2}}{\sum_{k=1}^{m} \frac{1}{\|\mathbf{u}_k\|^2}} \tag{4.51}$$

qui peut également s'obtenir directement par la résolution algébrique du problème de minimisation sous contraintes en formulation QP [4]. Les dérivées directionnelles sont égales (voir figure, théorème de Pythagore) :

$$\left(\mathbf{u}_j, \omega^\star\right) = \mathbf{u}_j^t \omega^\star = \left(\sum_{k=1}^{m} \frac{1}{\|\mathbf{u}_k\|^2} \right)^{-1} = \sigma \quad (\forall j)\,. \tag{4.52}$$

Nous allons maintenant généraliser ce résultat au cas où les vecteurs gradients forment une famille libre, pas nécessairement orthogonale. L'expérience de l'exemple précédent nous a révélé le rôle clé de la notion d'orthogonalité. Par conséquent, nous sommes conduits à appliquer aux vecteurs gradients, $\{\mathbf{u}_j\}$, le processus d'orthogonalisation de Gram-Schmidt, ce qui fournit les nouveaux vecteurs $\{\mathbf{v}_j\}$, deux à deux orthogonaux vis-à-vis du produit scalaire euclidien usuel. Les familles $\{\mathbf{u}_j\}$ et $\{\mathbf{v}_j\}$ $(j = 1, \ldots, m)$ engendrent le même sous-espace $\mathcal{U}$ de dimension $\mathsf{r} = m$. Anticipant quelque peu sur le cas général où l'hypothèse d'indépendance linéaire sera abandonnée, le processus de Gram-Schmidt, sous la forme particulière présente, est effectué conformément à l'algorithme défini dans les Tables 4.1, 4.2 et 4.3. Plusieurs détails sont à noter :

4. Exercice : le vérifier.

— L'algorithme inclut un élément méthodologique supplémentaire de hiérarchisation qui permet à chaque étape du processus de choisir un vecteur particulier parmi ceux qui n'ont pas encore été utilisés pour construire la direction orthogonale suivante. Cet élément n'est pas utile pour le moment.
— La normalisation des vecteurs $\{\mathbf{v}_j\}$ est justifiée *a posteriori* par les conséquences sur les dérivées directionnelles.
— Au cours du processus, on permute les indices des vecteurs $\{\mathbf{u}_j\}$ de sorte qu'à la fin, ils sont placés dans l'ordre où ils ont servi à la construction des vecteurs orthogonaux, $\{\mathbf{v}_j\}$.

— Définition du premier élément

$$\mathsf{k} = \arg\max_i \left[\min_{\substack{j \\ (j \neq i)}} \frac{\big(\mathbf{u}_i, \mathbf{u}_j\big)}{\big(\mathbf{u}_i, \mathbf{u}_i\big)} \right] ,$$

permutation d'indices

$$\mathbf{u}_1 \rightleftharpoons \mathbf{u}_{\mathsf{k}} ,$$

et définition du premier élément de la famille orthogonale :

$$\mathbf{v}_1 = \mathbf{u}_1 .$$

— Définition de la borne supérieure sur le rang de la famille de vecteurs gradients

$$\mathsf{r}_{\max} = \min(m, n) .$$

TABLE 4.1 – Processus de Gram-Scmidt hiérarchique : initialisation

La matrice de coefficients $\mathbf{C} = \{c_{j,k}\}$ est triangulaire, et se calcule par colonnes, une nouvelle colonne à chaque étape de définition d'une nouvelle direction orthogonale. Ce calcul s'effectue soit complètement (si $\mathsf{r} = m$ en sortie), soit incomplètement ($\mathsf{r} < m$) lorsqu'une estimation intermédiaire du vecteur $\omega^\star$ se révèle satisfaisante pour l'ensemble de la famille.

Le principe hiérarchique apparaît à l'étape j destinée au calcul de $\mathbf{v}_j$, dans le choix qui est fait parmi tous les vecteurs gradients non-encore utilisés du vecteur $\mathbf{u}_\ell$ pour lequel $c_{\ell,\ell}$ est minimum. Si $c_{\ell,\ell} < 1$, le calcul de $\mathbf{v}_j$ s'effectue conformément à laTable 4.3, en commençant par une permutation des informations relatives aux vecteurs $\mathbf{u}_j$ et $\mathbf{u}_\ell$. Le test est réalisé numériquement à une précision donnée près, contrôlée par le paramètre de précision TOL, le plus souvent mis à 0. Sinon, $c_{\ell,\ell} \geq 1 - TOL$, le processus est interrompu car l'estimation provisoire $\omega_1^\star$ de $\omega^\star$, fournit une direction de descente commune à toute la famille, comme nous allons le montrer.

Supposons que le processus ait été poursuivi jusqu'à l'étape $\mathsf{r} \leq m$, avec $\mathsf{r} = m$ ou pas, et définissons la sous-famille suivante

$$\text{Groupe I} := \{\mathbf{u}_1, \ldots, \mathbf{u}_{\mathsf{r}}\} \tag{4.53}$$

après qu'un certain nombre de permutations aient modifié la définition de ces éléments. Les

Initialisation de paramètres : $\mathsf{r} := 1$; $\mu := m$.
Pour $j = 2, 3, \ldots, \mathsf{r}_{\max}$ (au plus), faire :

1. Calculer la colonne $j-1$ de coefficients :

$$c_{i,j-1} = \frac{\big(\mathbf{u}_i, \mathbf{v}_{j-1}\big)}{\big(\mathbf{v}_{j-1}, \mathbf{v}_{j-1}\big)} \qquad (\forall i = j, \ldots, \mu)$$

et stocker les sommes cumulées suivantes dans la diagonale :

$$c_{i,i} := c_{i,i} + c_{i,j-1} = \sum_{k<j} c_{i,k} \qquad (\forall i = j, \ldots, \mu)$$

2. Identifier l'indice ℓ correspondant à la plus petite somme cumulée, $\ell = \arg\min_i \ \{c_{i,i} \ / \ j \le i \le \mu\}$, et comparer $c_{\ell,\ell}$ à $1 - TOL$:
   — Si $c_{\ell,\ell} \ge 1 - TOL$: poser $\mathsf{a} := c_{\ell,\ell}$; aller en 3 (FIN).
   — Sinon $(1 - c_{\ell,\ell} > TOL)$: calculer le vecteur orthogonal suivant $\mathbf{v}_j$ conformément aux définitions de la Table 4.3.
3. FIN : interrompre le processus d'orthogonalisation et procéder à la phase suivante : calcul du vecteur provisoire $\omega_1^\star$ par (4.54).

TABLE 4.2 – Processus de Gram-Scmidt hiérarchique : boucle principale

vecteurs orthogonaux $\{\mathbf{v}_1, \ldots, \mathbf{v}_\mathsf{r}\}$ ont été calculés, et on pose

$$\omega_1^\star = \sum_{j=1}^{\mathsf{r}} \beta_j \mathbf{v}_j \,, \quad \beta_j = \frac{\frac{1}{\|\mathbf{v}_j\|^2}}{\sum_{k=1}^{\mathsf{r}} \frac{1}{\|\mathbf{v}_k\|^2}} \,, \tag{4.54}$$

transposant (4.51), de sorte qu'à l'instar du cas d'une famille de gradients orthogonaux :

$$\mathbf{v}_j^t \omega_1^\star = \left(\sum_{k=1}^{\mathsf{r}} \frac{1}{\|\mathbf{v}_k\|^2} \right)^{-1} = \sigma_1 \quad (\forall j \le \mathsf{r}) \,, \tag{4.55}$$

pour la même constante strictement-positive σ_1. En conséquence, les r premières dérivées directionnelles, celles associées au Groupe I, sont données par :

$$\mathbf{u}_j^t \omega_1^\star = \left(\sum_{k=1}^{j} c_{j,k} \mathbf{v}_k \right)^t \omega_1^\star = \left(\sum_{k=1}^{j} c_{j,k} \right) \sigma_1 = \sigma_1 \tag{4.56}$$

car

$$\sum_{k=1}^{j} c_{j,k} = 1 \quad (\forall j \le \mathsf{r}), \tag{4.57}$$

en vertu du choix fait dans l'algorithme de la normalisation des vecteurs orthogonaux. Noter que cette normalisation est rendue possible par l'inégalité $c_{\ell,\ell} < 1 - TOL$ à la Table 4.2. Finalement, le vecteur $\mathbf{v}_j$ est calculé par l'expression donnée à la Table 4.3.

Ainsi, les dérivées directionnelles associées au Groupe I sont égales à la même constante strictement positives σ_1.

— Permutation de l'information relative aux indices j et ℓ :

$$\begin{gathered}\text{vecteurs : } \mathbf{u}_j \rightleftharpoons \mathbf{u}_\ell\,,\\ \text{lignes } j \text{ et } \ell \text{ de la matrice } \mathbf{C}\,,\\ \text{et sommes cumulées correspondantes } c_{j,j} \rightleftharpoons c_{\ell,\ell}\,.\end{gathered}$$

— Poser $A_j = 1 - c_{j,j}$; affecter cette valeur à $c_{j,j} := A_j$, et calculer :

$$\mathbf{v}_j = \frac{\mathbf{u}_j - \sum_{k<j} c_{j,k}\mathbf{v}_k}{c_{j,j}}$$

Si $\mathbf{v}_j = 0$:
— permuter l'information relative aux vecteurs $\mathbf{u}_j$ et $\mathbf{u}_\mu$ (et aux lignes correspondantes de la matrice $\mathbf{C}$) ;
— réactualiser $\mu := \mu - 1$;
— si $j \leq \mu$ aller en 2 (continuation du processus) ; sinon aller en 3 (FIN).

— Réactualiser $\mathsf{r} := \mathsf{r} + 1$, $j := j + 1$.
— Test : Si $j \leq \mu$, aller en 1 ; sinon, aller en 3 (FIN).

TABLE 4.3 – Processus de Gram-Scmidt hiérarchique : calcul de $\mathbf{v}_j$

Supposons maintenant que $\mathsf{r} < m$, et examinons la sous-famille complémentaire de gradients suivante :

$$\text{Groupe II} := \{\mathbf{u}_{\mathsf{r}+1}, \ldots, \mathbf{u}_\mu\} \tag{4.58}$$

où, dans le cas d'une famille linéairement indépendante, $\mu = m$; une définition plus générale de μ sera donnée ultérieurement pour prendre en compte également le cas inverse. Considérons le sous-espace $\mathcal{U}$ engendré par le Groupe I :

$$\mathcal{U} = \mathsf{Sp}\,(\mathbf{u}_1, \ldots, \mathbf{u}_\mathsf{r}) = \mathsf{Sp}\,(\mathbf{v}_1, \ldots, \mathbf{v}_\mathsf{r})\ . \tag{4.59}$$

Pour tout indice j tel que $\mathsf{r} + 1 \leq j \leq \mu$, on a :

$$\mathbf{u}_j = \sum_{k=1}^{\mathsf{r}} c_{j,k}\mathbf{v}_k + \mathbf{w}_j \tag{4.60}$$

où $\mathbf{w}_j \in \mathcal{U}^\perp$; donc $\mathbf{w}_j \perp \omega_1^\star$ et :

$$\mathbf{u}_j^t\omega_1^\star = \left(\sum_{k=1}^{\mathsf{r}} c_{j,k}\mathbf{v}_k\right)^t \omega_1^\star = \left(\sum_{k=1}^{\mathsf{r}} c_{j,k}\right)\sigma_1 = c_{j,j}\sigma_1 \geq \mathsf{a}\,\sigma_1 \tag{4.61}$$

où $\mathsf{a} = \min c_{j,j}$ $(\mathsf{r} + 1 \leq j \leq \mu)$. Quand le processus s'interrompt avant son exécution complète, on a $\mathsf{a} \geq 1 - TOL$. Cette constante est alors fournie par l'algorithme. Donc : **pour le Groupe II, les dérivées directionnelles sont supérieures ou égales à** $(1 - TOL)\sigma_1$ **(avec généralement** $TOL = 0$**).**

En résumé, dans le cas de vecteurs gradients linéairement indépendants, l'algorithme MGDA fournit un vecteur

$$\mathbf{d}_1 = \omega_1^\star\,, \tag{4.62}$$

un entier $\mathsf{r} \leq m$, une constante $\sigma_1 > 0$, et si $\mathsf{r} < m$, une constante $\mathsf{a} \geq 1 - TOL$. Alors, $(-\mathbf{d}_1)$ est une direction de descente commune à toutes les fonctions-objectifs ; compte tenu des permutations d'indices effectuées sur la famille de gradients $\{\mathbf{u}_j\}$, et implicitement sur les fonctions-objectifs, les dérivées directionnelles des r premières fonctions-objectifs sont égales à la constante σ_1, et si $\mathsf{r} < m$, les autres sont supérieures ou égales à $\mathsf{a}\,\sigma_1$. Nous désignerons désormais cet algorithme de détermination de la direction de descente commune par le terme de méthode primale.

On considère maintenant une voie alternative pour identifier la direction de descente. Construisons un nouveau produit scalaire euclidien par rapport auquel les vecteurs d'origine du Groupe I forment une famille orthogonale.

Le processus de Gram-Schmidt a été appliqué et a fourni la famille $\{\mathbf{v}_1, \ldots, \mathbf{v}_\mathsf{r}\}$, orthogonale vis-à-vis du produit scalaire usuel, ou disons, canonique. L'espace entier $\mathbb{R}^n$ est à nouveau décomposé en deux sous-espaces supplémentaires $\mathcal{U}$ et $\mathcal{U}^\perp$, où $\mathcal{U}$ est engendré par le Groupe I. Soit

$$\mathbf{x} = \mathbf{y} + \mathbf{z}\,,\ \mathbf{x}' = \mathbf{y}' + \mathbf{z}' \quad (\mathbf{y}, \mathbf{y}' \in \mathcal{U}) \quad (\mathbf{z}, \mathbf{z}' \in \mathcal{U}^\perp) \tag{4.63}$$

la décomposition d'un vecteur quelconque ; on a :

$$\mathbf{y} = \underline{\mathbf{U}}\eta\,, \quad \mathbf{y}' = \underline{\mathbf{U}}\eta'\,, \tag{4.64}$$

où $\underline{\mathbf{U}}$ est la matrice $n \times \mathsf{r}$ formée par les vecteurs-colonnes $\{\mathbf{u}_j\}$ $(j = 1, \ldots, \mathsf{r})$. Définissons alors le nouveau produit scalaire dans $\mathbb{R}^n$ comme suit :

$$\big(\mathbf{x}, \mathbf{x}'\big) := \eta^t\eta' + \mathbf{z}^t\mathbf{z}'\,. \tag{4.65}$$

Vis-à-vis de ce nouveau produit scalaire, les vecteurs d'origine du Groupe I sont deux à deux orthogonaux, et même orthonormés.

À l'inverse, une famille orthonormale vis-à-vis du produit scalaire canonique (ancien) s'obtient simplement en normalisant les vecteurs $\{\mathbf{v}_1, \ldots, \mathbf{v}_\mathsf{r}\}$. Soit la matrice

$$\tilde{\mathbf{V}} = \mathbf{V}\Delta^{-\frac{1}{2}} \tag{4.66}$$

où $\mathbf{V}$ est la matrice formée des vecteurs-colonnes $\{\mathbf{v}_j\}$, et $\Delta = \mathbf{Diag}\big(\mathbf{v}_j^t\mathbf{v}_j\big)$ la matrice diagonale contenant les carrés de leurs normes (dans la base canonique). En conséquence, la matrice de projection orthogonale dans le sous-espace $\mathcal{U}$ est la suivante :

$$\Pi = \sum_{j=1}^{\mathsf{r}} \tilde{\mathbf{v}}_j\tilde{\mathbf{v}}_j^t = \tilde{\mathbf{V}}\tilde{\mathbf{V}}^t = \mathbf{V}\Delta^{-1}\mathbf{V}^t \tag{4.67}$$

ce qui permet d'identifier le projection $\mathbf{y}$ à partir de $\mathbf{x}$:

$$\mathbf{y} = \Pi\mathbf{x} = \mathbf{V}\Delta^{-1}\mathbf{V}^t\mathbf{x}\,. \tag{4.68}$$

Il vient :

$$\underline{\mathbf{U}}^t\mathbf{y} = \underline{\mathbf{U}}^t\underline{\mathbf{U}}\eta\,, \tag{4.69}$$

où la matrice $\underline{\mathbf{U}}^t\underline{\mathbf{U}}$ est $\mathsf{r}\times\mathsf{r}$ et inversible car la relation $\mathbf{y}\longleftrightarrow\eta$ est bijective car elle exprime un simple changement de base dans le sous-espace $\mathcal{U}$ de dimension r. Ceci donne :

$$\eta=\left(\underline{\mathbf{U}}^t\underline{\mathbf{U}}\right)^{-1}\underline{\mathbf{U}}^t\mathbf{y}=\left(\underline{\mathbf{U}}^t\underline{\mathbf{U}}\right)^{-1}\underline{\mathbf{U}}^t\mathbf{V}\Delta^{-1}\mathbf{V}^t\mathbf{x}:=\mathbf{W}\mathbf{x}\,, \tag{4.70}$$

où :

$$\mathbf{W}=\left(\underline{\mathbf{U}}^t\underline{\mathbf{U}}\right)^{-1}\underline{\mathbf{U}}^t\mathbf{V}\Delta^{-1}\mathbf{V}^t\,, \tag{4.71}$$

et

$$\mathbf{z}=(\mathbf{I}-\Pi)\mathbf{x}\,. \tag{4.72}$$

Finalement :

$$\left(\mathbf{x},\mathbf{x}'\right)=\mathbf{x}^t\mathbf{A}_n\mathbf{x} \tag{4.73}$$

où :

$$\mathbf{A}_n=\mathbf{W}^t\mathbf{W}+(\mathbf{I}-\Pi)^2\,. \tag{4.74}$$

Cette nouvelle métrique nous ramène à la configuration de la Figure 4.13 avec ici, en plus, le fait que les gradients sont de même (nouvelle) norme 1. Le vecteur $\omega^\star$ correspondant est porté par la bissectrice et il est égal à la moyenne arithmétique des gradients :

$$\omega_2^\star=\frac{1}{\mathsf{r}}\sum_{j=1}^{\mathsf{r}}\mathbf{u}_j \tag{4.75}$$

et le vecteur de descente en résulte :

$$\mathbf{d}_2=\mathbf{A}_n\omega_2^\star\,. \tag{4.76}$$

Nous désignerons désormais cette voie alternative de détermination de la direction de descente commune par le terme de méthode duale.

Alors, quelle relation existe-t-il entre les vecteurs $\mathbf{d}_1$ et $\mathbf{d}_2$ fournis par les méthodes primale et duale ? Sans grande surprise, on a

Proposition 5
Les directions de descente définies par les méthodes primale et duale sont identiques ($\mathbf{d}_2/\sigma_2=\mathbf{d}_1/\sigma_1$*).*

Preuve : Posons

$$\mathbf{d}_2=\sum_{k=1}^{\mathsf{r}}\beta_k'\mathbf{v}_k \tag{4.77}$$

et déterminons les composantes inconnues $\{\beta_k'\}$. On a : $\mathbf{d}_2=\mathbf{A}_n\omega_2^\star=\frac{1}{\mathsf{r}}\sum_{k=1}^{\mathsf{r}}\mathbf{A}_n\mathbf{u}_k$, de sorte que :

$$\mathbf{u}_j^t\mathbf{d}_2=\frac{1}{\mathsf{r}}\sum_{k=1}^{\mathsf{r}}\left(\mathbf{u}_j^t\mathbf{A}_n\mathbf{u}_k\right)=\frac{1}{\mathsf{r}}=\sigma_2\,. \tag{4.78}$$

Les équations de définition du processus de Gram-Schmidt donnent $\mathbf{u}_j=\sum_{k\le j}c_{j,k}\mathbf{v}_k$. Il vient $\sum_{k\le j}c_{j,k}\mathbf{v}_k^t\mathbf{d}_2=\frac{1}{\mathsf{r}}$. Enfin, puisque les $\{\mathbf{v}_k\}$ sont deux à deux orthogonaux vis-à-vis du produit scalaire canonique, les composantes inconnues sont la solution du système linéaire suivant :

$$\sum_{k\le j}c_{j,k}\left\|\mathbf{v}_k\right\|^2\beta_k'=\sigma_2\quad(\forall j). \tag{4.79}$$

Or, en injectant

$$\beta'_k = \frac{\sigma_2}{\sigma_1}\beta_k \quad (\forall k), \tag{4.80}$$

où β_k est donné par (4.54), dans le membre de gauche du système, ce membre devient égal à $\sigma_2 \sum_{k\leq j} c_{j,k} = \sigma_2$, ce qui confirme qu'il s'agit bien de la solution, à savoir :

$$\mathbf{d}_2 = \frac{\sigma_2}{\sigma_1}\mathbf{d}_1\,. \tag{4.81}$$

□

Cas d'une famille de gradients linéairement dépendante

On examine maintenant le dernier cas où la famille des gradients est linéairement dépendante. Cette situation se produit en particulier lorsque $m > n$, ou même $m \gg n$; ce dernier sous-cas est important pour l'application au traitement de la problématique d'optimisation robuste par une formulation multipoint, comme l'illustrera la section 4.3.7, lorsque m est le nombre de points pris en considération dans l'intervalle de variation du paramètre par rapport auquel on vise une optimisation robuste.

Techniquement on commence par appliquer le processus d'orthogonalisation de Gram-Schmidt, avec les procédures particulières de hiérarchisation et de normalisation de l'Algorithme MGDA défini aux Tables 4.1, 4.2 and 4.3.

En général, au cours du processus, il survient parfois que le vecteur orthogonal nouvellement calculé, dont la définition est donnée à la Table 4.3, est égal à 0, par dépendance linéaire aux vecteurs précédemment calculés. Cette occurrence survient au moins une fois sauf dans le cas exceptionnel où le processus est interrompu avant complétion de la base parce qu'une direction de descente provisoire commune à tous les vecteurs est détectée. On ignore ce cas d'exception dans ce qui suit.

Lorsqu'il survient que $\mathbf{v}_j = 0$, le vecteur $\mathbf{u}_j$ est mis de côté, en fin de liste, après des permutations d'information, et définitivement inutilisé dans le processus, alimentant ainsi un troisième groupe de vecteurs, Groupe III, linéairement dépendants du Groupe I. On réduit alors l'entier μ d'une unité, pour réduire le Groupe II constitué des candidats à la constrruction du vecteur orthogonal suivant. À l'issue du processus :

- les vecteurs du Groupe I forment une base du sous-espace vectoriel $\mathcal{U}$, et ont servi à en construire une base orthogonale $\{\mathbf{v}_j\}$ $(j = 1, \ldots, \mathsf{r})$, et une direction de descente provisoire $-\mathbf{d}_1$ commune à ce groupe ; les dérivées dans la direction de $\mathbf{d}_1 = \omega_1^\star$ sont égales à la même constante positive σ_1 ;
- les vecteurs du Groupe II sont associés à des dérivées directionnelles du même signe au moins égales à $\mathsf{a}\,\sigma_1$ $(\mathsf{a} = 1 - TOL)$; ces vecteurs peuvent avoir des composantes dans $\mathcal{U}^\perp$ que l'on peut désormais ignorer en posant la contrainte additionnelle $\mathbf{d} \in \mathcal{U}$;
- les vecteurs du Groupe III sont linéairement dépendants des précédents, et à ce stade, aucune conclusion ne peut être tirée concernant le signe des dérivées directionnelles correspondantes :

$$\text{Groupe III} := \{\mathbf{u}_{\mu+1}, \ldots, \mathbf{u}_m\} \tag{4.82}$$

Ayant exclu le cas où le Groupe III est vide, l'ambiguïté persiste pour ce groupe.

Pour lever l'ambiguïté, on a recours à la formulation QP de **??** qui est complètement générale. Cependant, on a le choix de la base dans laquelle formuler le problème QP. On

choisit les éléments du Groupe I qui ont été identifiés, au delà de leur indépendance linéaire, par un critère hiérarchique visant à ce que le cône qu'ils forment contienne un grand nombre des directions associées à la famille donnée des gradients. Si ce critère était idéal pour définir le "cône englobant", aucune suite ne serait nécessaire, l'axe du cône serait une direction de descente commune. La suite tend donc à corriger numériquement l'imperfection du critère hiérarchique.

Une fois le changement de base effectué, trois situations sont possibles :

1. Ou bien, les vecteurs transformés n'ont aucune composante strictement négative. Aucun traitement supplémentaire n'est nécessaire [5] : la direction de $-\mathbf{d}_1$ est une direction commune de descente.
2. Ou bien, au moins un vecteur transformé n'a aucune composante strictement positive. Alors on peut conclure à une situation de Pareto-stationnarité [6], aucune direction de descente commune n'existe.
3. Ou bien, si aucun des deux cas précédents n'est réalisé, on ne peut conclure définitivement immédiatement. Le problème QP formulé dans la nouvelle base est résolu par appel à une procédure de bibliothèque, par exemple la procédure quadprog de MATLAB ou qpsolve de SCILAB. La procédure fournit un vecteur $\tilde{\omega}_2^\star$. Ce vecteur est ensuite exprimé dans la base canonique d'origine : $\omega_2^\star = \underline{\mathbf{U}}\,\tilde{\omega}_2^\star$, la matrice de métrique $\mathbf{A}_n$ est calculée conformément à (4.74), et la solution est donnée par le vecteur $\mathbf{d}_2 = \mathbf{A}_n\,\omega_2^\star$ conformément à (4.76).

En conclusion, on constate que si dans le cas d'une famille de vecteurs gradients linéairement indépendants, la méthode primale est directe, dans le cas inverse, la méthode duale est appropriée dans le cas général ci-dessus (sous-cas 3), car on veut privilégier une base particulière de représentation dont le cône associé est large pour englober les directions d'un grand nombre de vecteurs liés.

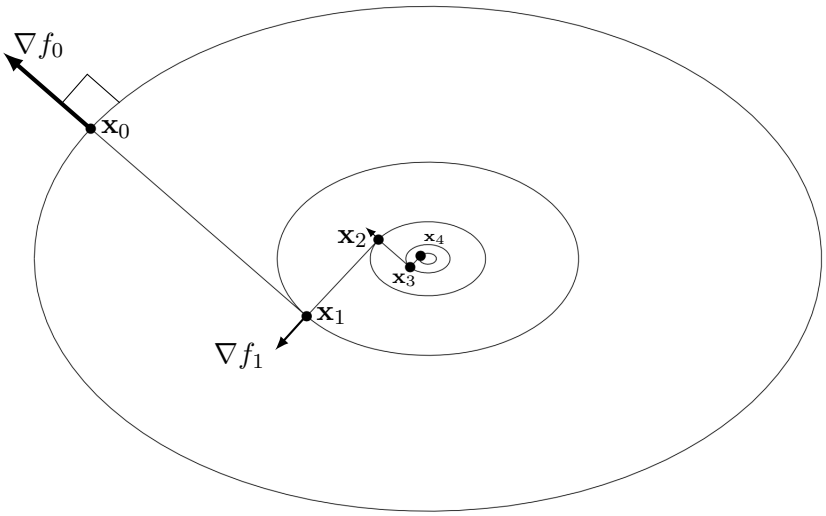
Enfin, on renvoie à l'Annexe C pour une description détaillée d'exemples simples de mise œuvre du processus d'orthogonalisation de Gram-Schmidt tel que nous venons de le définir dans l'algorithme MGDA.

Exercice 18 (Transformation exponentielle et convexité)
Le contexte de l'exercice est celui de la minimisation d'une, ou de m fonctions réguliÃ¨res de n variables par la méthode du gradient si $m = 1$ ("Steepest-Descent Method" : méthode de descente suivant la ligne de plus grande pente), ou plus généralement, par MGDA si $m > 1$. On examine précisément l'itération à partir d'un point $\mathbf{x}_0$ du domaine admissible $\Omega_{\text{ad}} \subseteq \mathbb{R}^n$.

Pour rappel, la méthode du gradient est illustrée par ses premiers itérés à la Figure 4.14 dans le cas de la minimisation de la fonction convexe de deux variables $f(x, y) = (x/1.5)^2 + y^2$ ($m = 1$, $n = 2$). Le minimum se situe à l'origine des coordonnées. En tout point du domaine (x, y), le gradient est orthogonal à la ligne de niveau de f (ou contour isovaleur) qui passe par ce point. Le pas optimal est réalisé par le point de l'axe de descente où cet axe est juste tangent à une autre ligne de niveau de f, ici une ellipse homothétique de plus petite taille. De ce fait, chaque nouveau déplacement est orthogonal au précédent. Le tableau indique la valeur du demi-axe vertical b_i de l'ellipse et son rapport à la valeur précédente, ce qui donne le rapport d'homothétie. On constate que ce rapport est à peu prÃ¨s constant au cours des itérations, ce qui indique une décroissance géométrique. La fonction f est proportionnelle au

5. Exercice : justifier.
6. Exercice : justifier.

carré de ce rapport. La convergence de la fonction, ici vers 0, est donc quadratique à l'instar des méthodes de type Newton.



i	b_i	b_i/b_{i-1}
0	1	-
1	0.3828	0.3828
2	0.1466	0.3830
3	0.0561	0.3829
4	0.0214	0.3829

FIGURE 4.14 – Illustration de la méthode du gradient appliquée à la minimisation de la fonction $f(x, y) = (x/1.5)^2 + y^2$; le tableau indique la valeur du demi-axe vertical b_i de l ?ellipse et son rapport à la valeur précédente.

Le but de l'exercice est de discuter de l'approximation du pas optimal lorsque la fonction, loin du minimum, n'est pas nécessairement convexe. Préalablement, on généralise un peu la méthode en considérant désormais que la descente est effectuée sur l'axe de vecteur directeur $(-\omega)$ ($\omega \in \mathbb{R}^n$) sous hypothÃ¨se de positivité du produit scalaire de ω avec le gradient de f en $\mathbf{x}_0$, ∇f_0 :

$$\big(\omega, \nabla f_0\big) > 0 \tag{4.83}$$

ce qui, bien évidemment, est vrai en particulier lorsque $\omega = \nabla f_0 \neq 0$.

Si $m = 1$, le vecteur ω peut être un gradient approché, incomplet, ou préconditionné. Si $m > 1$, il correspondra à la direction de descente commune à toutes les fonctions coûts définie par MGDA.

1. On considère une fonction régulièrre $f(\mathbf{x})$ de la variable vectorielle $\mathbf{x} \in \Omega_{\text{ad}}$:

$$f : \ \mathbf{x} \in \Omega_{\text{ad}} \subseteq \mathbb{R}^n \longrightarrow \mathbb{R} \tag{4.84}$$

(n quelconque ; $f \in C^2\big(\Omega_{\text{ad}}\big)$). On fixe $\mathbf{x}_0$ et ω et on définit la fonction restreinte

$$\phi(\epsilon) = f\big(\mathbf{x}_0 - \epsilon\omega\big). \tag{4.85}$$

Établir le développement limité de $\phi(\epsilon)$ au second-ordre. Identifier $\phi(0)$, $\phi'(0)$ et $\phi''(0)$ au moyen de $f_0 = f(\mathbf{x}_0)$, du gradient ∇f_0, de la matrice hessienne $H_0 = \nabla\nabla^t f_0$, et du vecteur ω. En déduire sous quelle(s) condition(s) l'itération est une descente effective, et la fonction restreinte convexe en $\epsilon = 0$. En cas de convexité, calculer la valeur du pas ϵ qui minimise le développement limité.

2. On se place maintenant dans le cas où $\phi(\epsilon)$ n'est pas convexe en $\epsilon = 0$ ($\phi''(0) < 0$). On

modifie alors la stratégie itérative pour faire porter l'itération de descente sur la fonction suivante en remplacement de f :

$$\tilde{f}(\mathbf{x}) = \frac{e^{\lambda\left(f(\mathbf{x})-f_0\right)}}{\lambda} \tag{4.86}$$

où λ est une constante positive à choisir ultérieurement. Comparer les fonctions f et $\tilde{f}$ en matière de : (i) régularité, (ii) sens de variation, (iii) valeur en $\mathbf{x}_0$, (iv) gradient, et (v) matrice hessienne en $\mathbf{x}_0$, et justifier la stratégie proposée. En déduire le développement limité au second-ordre de la fonction

$$\tilde{\phi}(\epsilon) = \tilde{f}\left(\mathbf{x}_0 - \epsilon\omega\right) \tag{4.87}$$

et montrer que cette fonction est localement convexe pourvu que λ soit suffisamment grand.
3. On considère maintenant une itération de descente par l'algorithme MGDA appliqué aux fonctions coûts $\{f_j(\mathbf{x})\}$ ($j = 1, \ldots, m;\ m > 1$), et on suppose que $\omega = \omega^\star$. Expliquer comment, sans changer la nature du problème, on peut se ramener au cas où toutes les fonctions coûts sont localement convexes dans la direction de $\omega^\star$.

On trouvera un corrigé complet de cet exercice en Annexe D.

4.3.6 Pareto stationnarité sous contrainte d'égalité, convergence de MGDA pénalisé, front de Pareto

Comme on l'a montré en Annexe A, les contraintes d'inégalité ou d'intervalle peuvent se formuler en contraintes d'égalité par l'introduction de nouvelles variables, dites d'ajustement (en anglais : "slack variables"). On considÃ¨re donc ce seul type de contrainte :

$$\mathbf{c}(\mathbf{x}) = 0 \tag{4.88}$$

où $\mathbf{c}(\mathbf{x}) = (c_1(\mathbf{x}), \ldots, c_K(\mathbf{x}))$ est à valeurs dans $\mathbb{R}^K$ et correspond à un ensemble de K contraintes qualifiées. Ici, spécifiquement, on suppose que les gradients de ces contraintes sont linéairement indépendants.

La condition de Pareto-stationnarité s'applique alors aux gradients projetés $\{\mathbf{P}\nabla f_j\}$:

$$\exists \alpha = \{\alpha_j\} \in \mathbb{R}_+^m \text{ tel que : } \sum_{j=1}^{m} \alpha_j \mathbf{P}\nabla f_j(\mathbf{x}_0) = 0\,,\ \sum_{j=1}^{m} \alpha_j = 1 \tag{4.89}$$

où la matrice de projection $\mathbf{P}$ s'exprime par le biais de projecteurs (matrices de rang 1) comme suit :

$$\mathbf{P} = \mathbf{I}_n - \sum_{k=1}^{K} [\gamma_k]\,[\gamma_k]^t \tag{4.90}$$

où les vecteurs $\{\gamma_k\}$ ($k = 1, \ldots, K$) forment une base orthonormale du sous-espace vectoriel localement orthogonal aux surfaces de contraintes. Ces vecteurs s'obtiennent par application du processus d'orthogonalisation de Gram-Schmidt aux gradients de contraintes. La notation $[\gamma_k]$ dénote le vecteur colonne des composantes du vecteur.

Le systÃ¨me d'équations 4.89 est seulement de rang $n-K$. Il est complété par le systÃ¨me généralement non linéaire des K équations de contraintes (4.88).

On peut alors généraliser l'algorithme MGDA pour tenir compte des contraintes en décomposant l'itération en deux étapes :

1. Étape 1 : descente par l'algorithme MGDA appliqué aux gradients projetés ;
2. Étape 2 : retour à la contrainte nonlinéaire par un algorithme de type méthode de Newton.

On obtient ainsi une méthode de type "gradients réduits" ([30]) ou méthode quasi-riemannienne [19].

4.3.7 Exemple d'application à un problème de contrôle d'écoulement

Il est connu que l'introduction d'un jet pulsé à l'extrados d'une voilure peut en améliorer la performance aérodynamique. Par exemple, à la Figure 4.15, on illustre le recollement d'un écoulement à grande incidence par le jet.

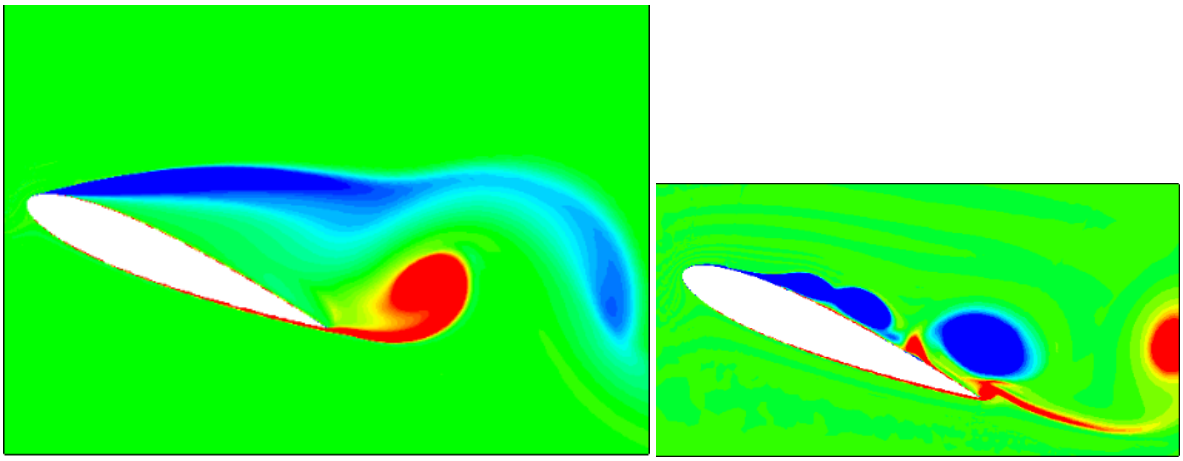


FIGURE 4.15 – Écoulement à l'extrados d'un profil d'aile (simulation par volumes-finis des équations de Navier-Stokes compressible, R. Duvigneau) ; à gauche : sans jet, écoulement décollé ; à droite, avec jet pulsé, écoulement recollé.

Il est donc un domaine de la conception en aérodynamique, dit "contrôle actif", dans lequel on cherche à améliorer la performance par l'introduction de dispositifs, tels que jets pulsés, qui modifient la structure même de l'écoulement. Ce domaine est hors de portée de ce cours, mais nous présentons ici un cas académique d'optimisation de jets pulsés à titre illustratif d'une problématique d'optimisation paramétrique multipoint traitée comme une optimisation multiobjectif dans laquelle les objectifs sont en surnombre par rapport à la dimension d'espace.

On considère un écoulement bidimensionnel, compressible, laminaire régi par les équations de Navier-Stokes sur une plaque plane sur laquelle se développe une couche limite. Ces équations s'écrivent symboliquement comme suit :

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{R}(\mathbf{W}) = 0 \tag{4.91}$$

où $\mathbf{W} = (\rho, \rho u, \rho v, E)$ est le vecteur des variables d'écoulement, et le vecteur $\mathbf{R}(\mathbf{W})$, le résidu, est la somme de flux convectifs et diffusifs.

Les équations de Navier-Stokes sont soumises à des conditions aux limites classiques. En particulier, sur la paroi solide, la condition d'adhérence ($V = (u, v) = 0$) est imposée.

On introduit trois jets pulsés ($j = 1, 2, 3$). Assez arbitrairement, on a fixé la position du jet central et les deux autres à équidistance de part et d'autre, ainsi que les fréquences de

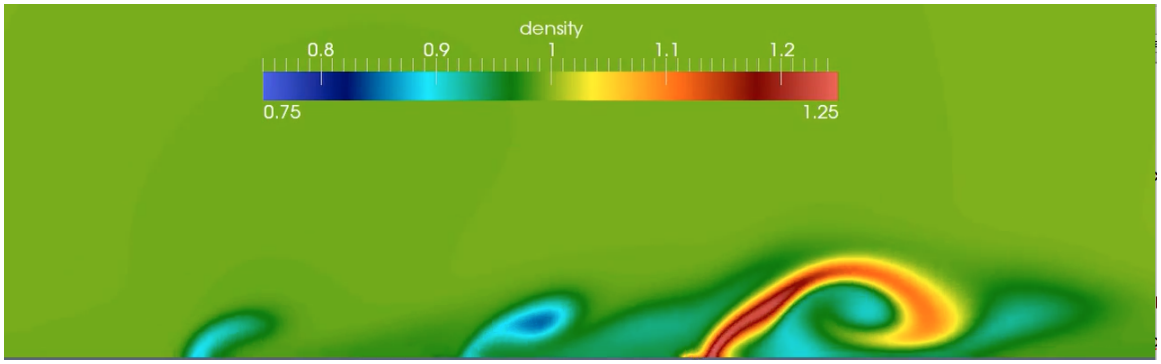


FIGURE 4.16 – Un instantanné d'écoulement instationnaire périodique sur une plaque plane avec trois jets pulsés

pulsation à des valeurs dans des rapports donnés, soit $f_1 = f$, $f_2 = 2f$ et $f_3 = f/2$. D'autres arrangements seraient certainement plus performants. Dans le voisinage du jet d'indice j, on impose la vitesse normale v sur un certain support X_j constitué par le bord de plusieurs cellules à la paroi, comme suit :

$$v(x) = A_j \phi_j(x) \sin(2\pi f_j t - \varphi_j) \qquad (\forall x \in X_j) \tag{4.92}$$

où $\phi_j(x)$ est une fonction donnée de support X_j, régulière et en forme de cloche, A_j une amplitude de flux, et φ_j une phase. Conformément à la théorie des caractéristiques, en phase de soufflage ($v > 0$), on impose également $u = 0$, $\rho = \rho_0$ et la pression p est libre, et en phase d'aspiration ($v \leq 0$), u, ρ et p sont libres [7].

Les trois valeurs d'amplitude $\{A_j\}$ et les trois valeurs de phase $\{\varphi_j\}$ constituent un vecteur $\mathbf{a} \in \mathbb{R}^6$ et font l'objet de l'optimisation, en prenant comme objectif de réduire la force de frottement exercée par le fluide sur la paroi.

Un régime complexe d'ondes en interférence non linéaire ("battement") s'établit et constitue un écoulement instationnaire périodique. La Figure 4.16 en illustre un instantanné.

Au plan numérique, les équations de Navier-Stokes, sont discrétisées en deux dimensions d'espace, par un schéma de volumes-finis s'appuyant sur une triangulation régulière du domaine de calcul, ici un rectangle dont la plaque plane est un côté :

$$\frac{\partial \mathbf{W}_h}{\partial t} + \mathbf{R}_h(\mathbf{W}_h) = 0 \tag{4.93}$$

où $\mathbf{W}_h = \mathbf{W}_h(t)$ contient les valeurs discrétisées en espace des variables d'écoulement, $\mathbf{W}$, aux nœuds d'un maillage triangulaire ("volumes-finis centrés sommets"), et de manière analogue, $\mathbf{R}_h(\mathbf{W}_h)$, celles du résidu intégré à la cellule de volume-fini. L'indice h réfère à cette discrétisation spatiale dont on omet les détails ici.

L'intégration en temps s'effectue par un schéma de type Euler-implicite linéarisé :

$$\left[\mathbf{I} + \Delta t\, \mathbf{R}'_h(\mathbf{W}_h^n)\right]\left(\mathbf{W}_h^{n+1} - \mathbf{W}_h^n\right) = -\Delta t\, \mathbf{R}_h(\mathbf{W}_h^n) \tag{4.94}$$

où :

$$\mathbf{R}'_h(\mathbf{W}_h) = \frac{\partial \mathbf{R}_h(\mathbf{W}_h)}{\partial \mathbf{W}_h} \tag{4.95}$$

7. Optimization of active control devices for separated turbulent flows, Jérémie Labroquère, Thèse de doctorat, Université Nice Sophia Antipolis, 2014. `https://hal.inria.fr/tel-01127187v1`

est une matrice jacobienne $N \times N$ (N : nombre de degrés de liberté dépendant du maillage et du type d'approximation) ; cette matrice est de grande taille, mais très creuse en raison du caractère local des approximations de type volumes-finis. Il est important de noter que cette matrice, associée à ce schéma implicite, fait précisément intervenir le linéarisé du résidu $\mathbf{R}_h$ par rapport aux inconnues $\mathbf{W}_h$.

L'intégration en temps s'effectue sur plusieurs périodes $T = 2/f$ car les conditions d'écoulement périodique ne sont pas connues sur tous les bords du domaine d'espace et il convient de simuler jusqu'à ce que le régime périodique soit établi. Pour chaque période, on calcule 800 pas de temps. Pour un choix donné des paramètres, la simulation seule est donc réalisée par plusieurs milliers de pas de temps.

À chaque instant de la période, on obtient par intégration en espace, la force de frottement, $D(t)$, exercée par le fluide à la paroi Γ. Cette force est une fonction du temps :

$$D(t) = \int_\Gamma \mu \frac{\partial u}{\partial y}\, dx\,. \tag{4.96}$$

Cette fonction est représentée à la Figure 4.17 aux 800 pas de temps d'une période.

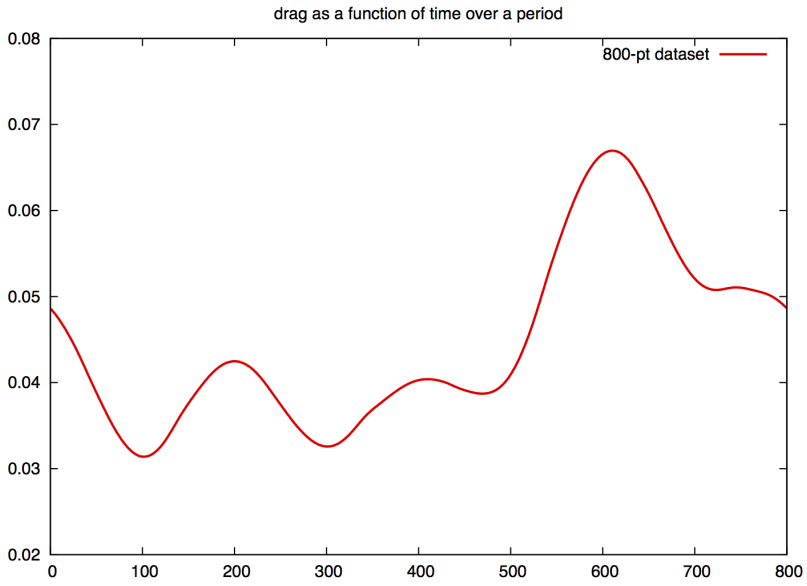


FIGURE 4.17 – Réglage initial des paramètres ; force de frottement à la paroi sur une période T ($T = 800\Delta t$).

Le calcul confirme que la valeur moyenne de la force de frottement sur une période,

$$\bar{D} = \frac{1}{T}\int_0^T D(t)\, dt \tag{4.97}$$

est effectivement inférieure à la valeur correspondant à l'écoulement permanent sans les jets. Les jets pulsés ont donc bien l'effet favorable attendu. Comment alors régler optimalement les 6 paramètres caractérisant les jets (amplitudes et phases) ?

Parallèlement à l'intégration temporelle de l'écoulement, on évalue une contribution au gradient de fonctionnelle qui, accumulée au temps final, donne le gradient au complet. On applique pour cela la technique d'évaluation des "sensibilités" que l'on présente dans un cadre plus simple dans l'Annexe E.

Ici précisément, on intègre pour chaque paramètre a auquel le systÃ¨me est sensible (composante du vecteur $\mathbf{a} \in \mathbb{R}^6$), une équation de "sensibilité de l'écoulement à ce paramètre" obtenue en dérivant (4.93) par rapport à a :

$$\frac{\partial \left(\mathbf{W}_h\right)'_a}{\partial t} + \mathbf{R}'_h(\mathbf{W}_h)\left(\mathbf{W}_h\right)'_a = 0 \tag{4.98}$$

où $\left(\mathbf{W}_h\right)'_a = \partial \mathbf{W}_h / \partial a$, et la matrice $\mathbf{R}'_h(\mathbf{W}_h)$ est précisément donnée par (4.95) [8]. On renvoie à l'Annexe **??** pour une présentation du calcul de sensibilités dans un cadre plus simple. Les termes sources de ce système proviennent des conditions à la paroi liées aux jets. L'intégration en temps par le schéma d'Euler implicite de ces équations linéaires fait donc intervenir la même matrice que pour l'intégrations de $\mathbf{W}_h$:

$$\left[\mathbf{I} + \Delta t\, \mathbf{R}'_h(\mathbf{W}_h^n)\right]\left(\left(\mathbf{W}_h\right)_a^{'(n+1)} - \left(\mathbf{W}_h\right)_a^{'n}\right) = -\Delta t\, \mathbf{R}'_h(\mathbf{W}_h^n)\ \left(\mathbf{W}_h\right)_a^{'n}\ . \tag{4.99}$$

La connaissance du champ de sensibilité $\left(\mathbf{W}_h\right)'_a$ donne accès à la dérivée de $D(t)$ par rapport à a :

$$D'_a(t) = \int_\Gamma \mu \frac{\partial u'_a}{\partial y}\, dx\,. \tag{4.100}$$

En effectuant cette intégration pour tous les paramètres d'optimisation a, on obtient le vecteur gradient par rapport au vecteur de conception $\mathbf{a}$ de la force $D(t)$ à tout instant t de la période :

$$\mathbf{g}(t) = \nabla_{\mathbf{a}} D(t)\,. \tag{4.101}$$

Dans notre application, on obtient ainsi 800 vecteurs de dimension 6. Les 6 composantes de ce gradient sont représentées à la Figure 4.18 qui reflète l'instationnarité du phénomène physique.

Afin de traiter un volume réduit de données, on décide de remplacer ces données brutes par des valeurs moyennées par tranche de $40\Delta t$, ce qui donne 20 valeurs moyennes de la force et des 6 composantes du gradient. Ceci conduit à remplacer les données brutes issues de la simulation, Figures 4.17 et 4.18, par les Figures 4.19 et 4.20 qui visiblement restent fidèles au phénomène instationnaire.

8. A Sensitivity Equation Method for Unsteady Compressible Flows : Implementation and Verification, Régis Duvigneau, Rapport de recherche Inria 8739, Inria, 2015. `https://hal.inria.fr/hal-01161957`

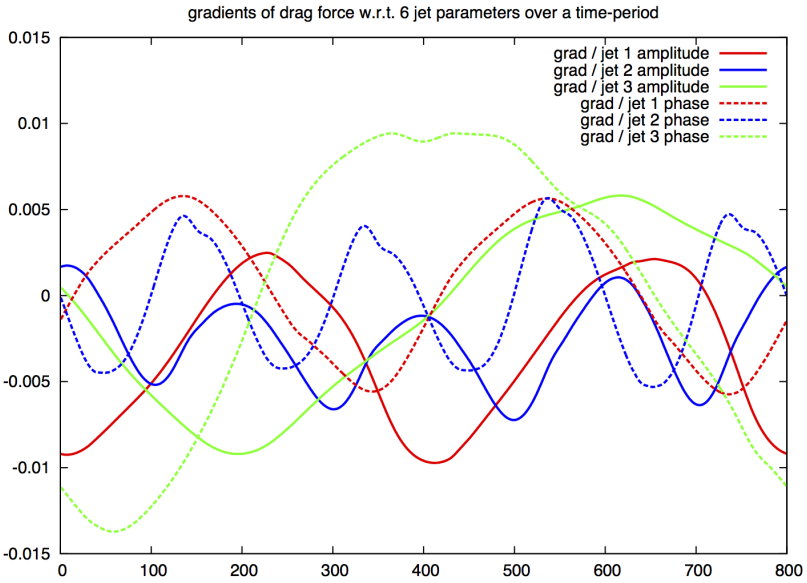


FIGURE 4.18 – Réglage initial des paramètres ; composantes du gradient $\mathbf{g}(t) = \nabla_{\mathbf{a}} D(t)$ sur une période T ($T = 800\Delta t$).

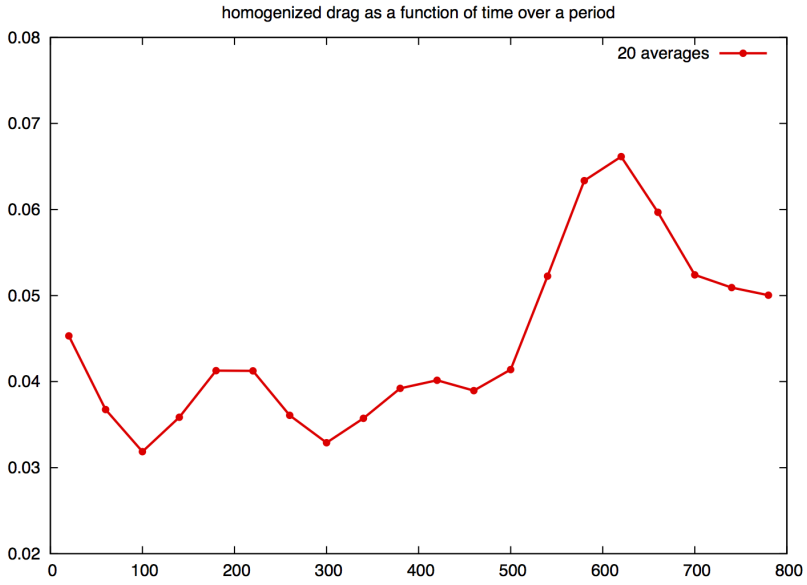


FIGURE 4.19 – Réglage initial des paramètres ; force de frottement à la paroi sur une période T moyennée par tranche de $40\Delta t$ ($T = 800\Delta t$).

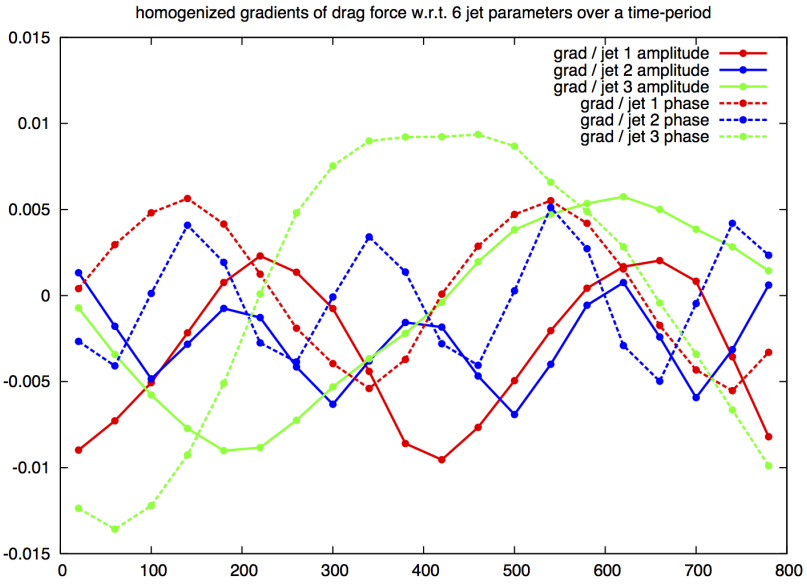


FIGURE 4.20 – Réglage initial des paramètres ; composantes du gradient $\mathbf{g}(t) = \nabla_a D(t)$ moyennées par tranche de $40\Delta t$ sur une période T ($T = 800\Delta t$).

Puis, on adimensionne chaque composante du gradient par le maximum de sa valeur absolue sur une période [9], et on applique l'algorithme MGDA du paragraphe 4.3.5. L'algorithme fournit une direction $\mathbf{d} \in \mathbb{R}^6$, dont les produits scalaires avec les 20 vecteurs gradients moyennés sont tous positifs. On effectue alors une variation bien calibrée du vecteur de paramètres $\mathbf{a}$ proportionnelle au vecteur $-\mathbf{d}$ et on exécute une nouvelle simulation de l'écoulement. Le post-traitement de l'écoulement permet de réactualiser la courbe de force de frottement en fonction du temps. On procède ainsi pour réaliser plusieurs itérations d'optimisation, dont les résultats sont consignés à la Figure 4.21.

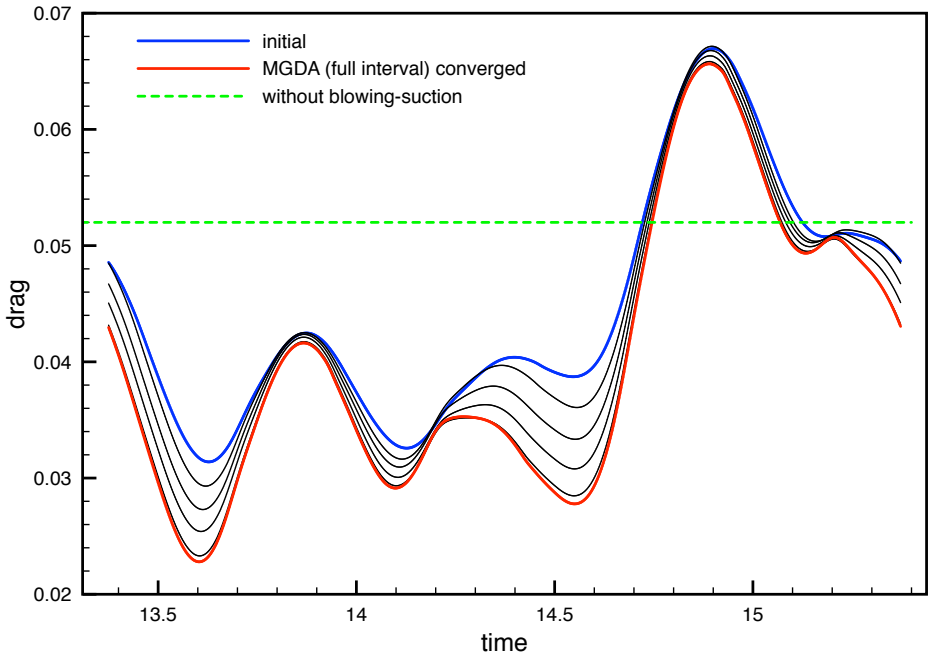


FIGURE 4.21 – Force de frottement à la paroi sur une période T ($T = 800\Delta t$) ; courbe initiale et courbes successives au cours d'une optimisation ciblant la période entière.

On constate que l'optimisation est efficace sur toute la période : à chaque itération, la courbe de force de frottement réactualisée ne dépasse nulle part la précédente, et indique presque partout une réduction significative, de sorte que la moyenne décroit. Cependant, on observe trois "zones de blocage", où aucune amélioration sensible n'est apparente. Mais la conclusion principale est que le gain n'est acquis au détriment d'aucune zone de la courbe.

Dans une deuxième expérience, on privilégie la réduction du maximum de la courbe situé aux environs des $\frac{3}{4}$ de la période. À cette fin, on focalise l'optimisation sur les derniers 40% de la période, quitte à dégrader le reste. On recalcule 20 moyennes de force et de gradients par tranche de $16\Delta t$ (voir Figures 4.22 et 4.23). Le résultat de plusieurs itérations d'optimisation successives est indiqué à la Figure 4.24. Visiblement on obtient une réduction de la zone du maximum bien supérieure et débordant de part et d'autre de la zone stricte des 40%. Cependant, comme prévu, ce gain est acquis au détriment de la majeure partie du reste de la courbe qui augmente.

9. Cette normalisation est rendue nécessaire pour donner aux différentes composantes du gradient des échelles comparables, mais aussi pour des raisons de compatibilité dimensionnelle lorsqu'on calcule des produits scalaires.

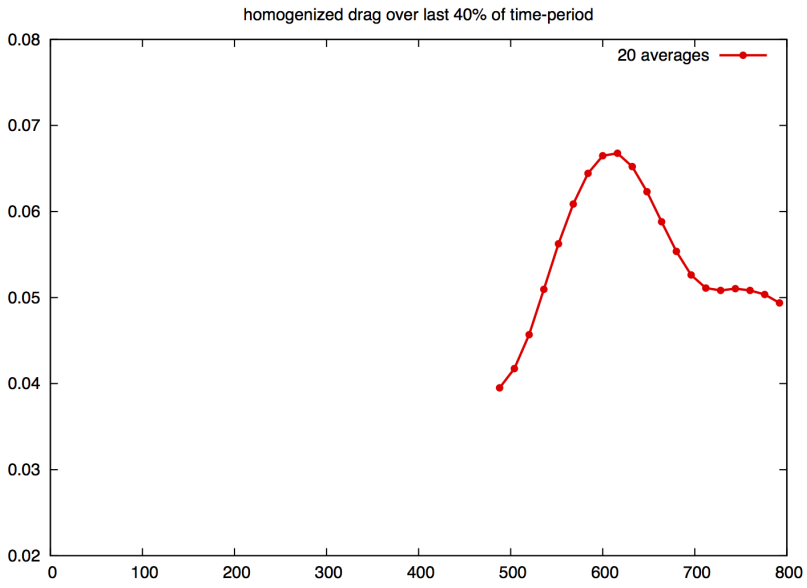


FIGURE 4.22 – Force de frottement à la paroi sur une période T ($T = 800\Delta t$) ; courbe initiale et courbes successives au cours de l'optimisation des paramètres de jets.

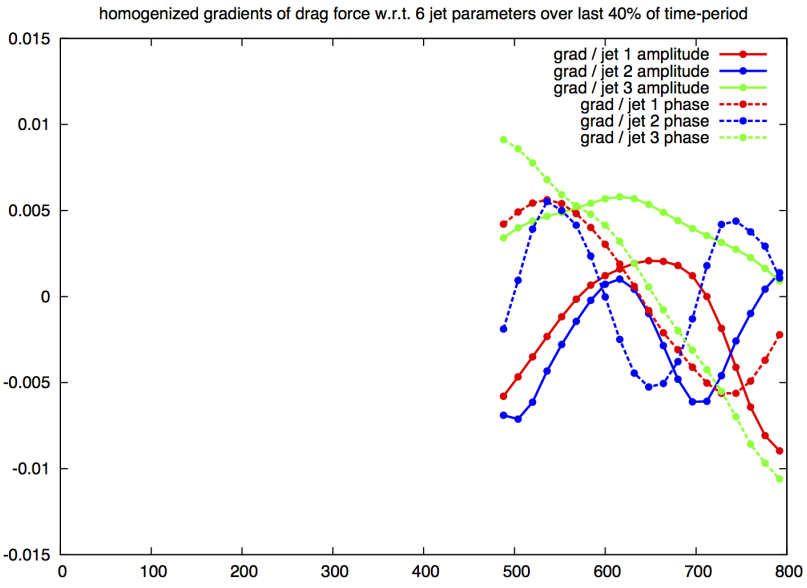


FIGURE 4.23 – Force de frottement à la paroi sur une période T ($T = 800\Delta t$) ; courbe initiale et courbes successives au cours de l'optimisation des paramètres de jets.

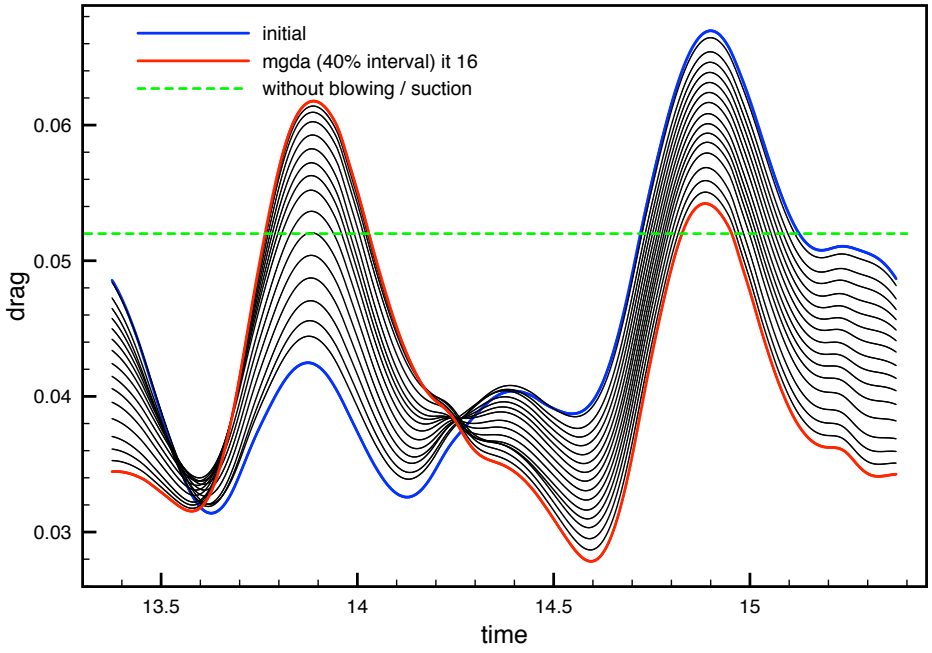


FIGURE 4.24 – Force de frottement à la paroi sur une période T ($T = 800\Delta t$) ; courbe initiale et courbes successives au cours de l'optimisation des paramètres de jets ciblant les derniers 40% de la période.

Cette expérience démontre que l'outil permet de cibler les zones d'action de l'optimisation sur la courbe de fonction d'intérêt en instationnaire.

Pour conclure, nous présentons deux itérations de réduction de la seule moyenne à la Figure 4.25. On y observe une réduction nette de la moyenne, mais au prix d'un raidissement et aggravation du maximum.

En conclusion, par le biais d'une formulation multipoint, traitée en problème d'optimisation multriobjectif par la méthode de descente MGDA s'appuyant sur des gradients calculés par résolution des équations de sensibilités, on a démontré la capacité de l'optimiseur à agir directement sur la courbe de fonction d'intérêt en instationnaire. Cet outil est donc plus puissant qu'un algorithme de réduction de grandeurs statistiques.

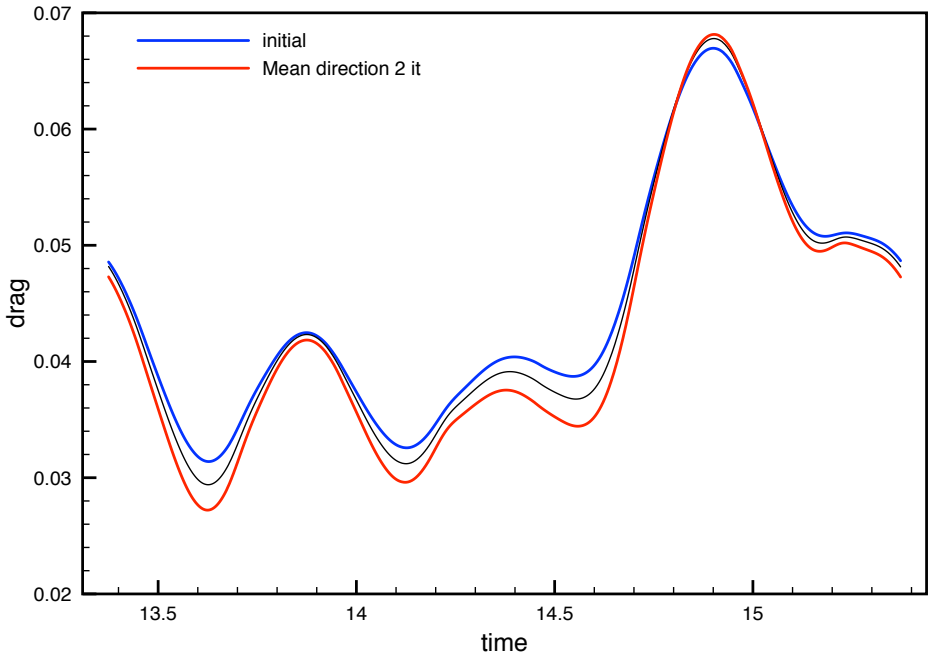


FIGURE 4.25 – Force de frottement à la paroi sur une période T ($T = 800\Delta t$) ; courbe initiale et courbes après une et deux itérations d'optimisation des paramètres de jets.

4.4 Jeux de Nash à deux joueurs pour l'optimisation bicritère

4.4.1 Formulation

On considère l'optimisation concourante de deux critères f_A et f_B fonctions-objectifs du même vecteur $\mathbf{x} \in \mathbb{R}^n$ de variables de conception. L'algorithme, sans réalité physique, est un artifice pour organiser cette compétition. On simule la dynamique entre deux joueurs virtuels A et B qui se partagent les variables selon deux sous-espaces supplémentaires E_A (de dimension $n-p$) et E_B (de dimension p). Si tout l'espace est admissible :

$$\mathbf{x} \in \mathbb{R}^n = E_A \oplus E_B \tag{4.102}$$

Par extension de la notation, on écrira :

$$\mathbf{x} = \{x_i\} = \mathbf{x}_A \oplus \mathbf{x}_B \tag{4.103}$$

pour dire que le sous-vecteur $\mathbf{x}_A \in \mathbb{R}^{n-p}$ engendre la projection de $\mathbf{x}$ dans E_A et $\mathbf{x}_B \in \mathbb{R}^p$ le sous-vecteur qui engendre la projection de $\mathbf{x}$ dans E_B.

La décomposition de ce type la simple consiste à partager les composantes de $\mathbf{x}$ dans la base canonique en deux sous-ensembles non vides complémentaires. Dans ce cas on dira qu'on partage les "variables primitives".

On écrira :

$$f_A = f_A(\mathbf{x}) = f_A(\mathbf{x}_A, \mathbf{x}_B) \qquad f_B = f_B(\mathbf{x}) = f_B(\mathbf{x}_A, \mathbf{x}_B) \tag{4.104}$$

La stratégie du joueur A (resp. B) consiste à optimiser le sous-vecteur $\mathbf{x}_A$ (resp. $\mathbf{x}_B$) dans le but de minimiser le critère f_A (resp. f_B) sous la contrainte de la stratégie $\mathbf{x}_B$ du joueur B (resp. A).

4.4.2 Équilibre de Nash

On dit que

$$\bar{\mathbf{x}} = \bar{\mathbf{x}}_A \oplus \bar{\mathbf{x}}_B \tag{4.105}$$

réalise un équilibre de Nash de la formulation précédente, ssi :

$$\bar{\mathbf{x}}_A = \arg\min_{\mathbf{x}_A} f_A(\mathbf{x}_A, \bar{\mathbf{x}}_B) \quad \text{et} \quad \bar{\mathbf{x}}_B = \arg\min_{\mathbf{x}_B} f_B(\bar{\mathbf{x}}_A, \mathbf{x}_B) \tag{4.106}$$

Autrement dit, $\bar{\mathbf{x}}_A$ est optimal vis-à-vis de f_A si $\mathbf{x}_B$ est fixé à la valeur $\bar{\mathbf{x}}_B$, et symétriquement $\bar{\mathbf{x}}_B$ est optimal vis-à-vis de f_B si $\mathbf{x}_A$ est fixé à la valeur $\bar{\mathbf{x}}_A$. Si un tel point d'equilibre est atteint, et bien que $\bar{\mathbf{x}}$ ne soit globalement optimal ni pour A (vis-à-vis de f_A), ni pour B (vis-à-vis de f_B), il n'est alors dans l'intérêt d'aucun des joueurs de modifier le sous-vecteur qu'il contrôle.

4.4.3 Algorithme et convergence

Supposons que l'on dispose de deux codes itératifs, l'un capable d'optimiser f_A par rapport à l'ensemble des variables, et l'autre f_B.

Alors la formulation précédente se prête à construire un "algorithme additif" adapté au calcul parallèle :

1. Initialisation de $\mathbf{x}$
2. Effectuer en parallèle k_A (resp. k_B) itérations de minimisation de f_A (resp. f_B) par modification de $\mathbf{x}_A$ (resp. $\mathbf{x}_B$) en maintenant $\mathbf{x}_B$ (resp. $\mathbf{x}_A$) fixé.
3. Échange : reconstituer $\mathbf{x}$ en assemblant les valeurs réactualisées des sous-vecteurs $\mathbf{x}_A$ et $\mathbf{x}_B$; retour à l'étape 2 sauf si la convergence du jeu, ou à défaut, une borne spéficiée sur le nombre d'échanges est atteinte.

Il est évident que :si cet algorithme itératif converge, alors le point limite réalise un équilibre de Nash.

En pratique, cet algorithme converge difficilement sauf si le partage des variables fait sens physiquement. On peut rajouter des mécanismes de "ressort" pour faciliter cette convergence (cf. travaux de Attouch sur les "algorithmes proximaux").

4.4.4 Partage de territoire adapté à une concurrence hiérarchisée

On considère une situation où f_A représente une "discipline principale", prépondérante ou fragile, et f_B une discipline secondaire. On propose de procéder en trois étapes, dont la troisième est une optimisation concourante par jeu de Nash.

Étape 1 - On optimise le seul critère principal f_A par rapport à l'ensemble des variables et sous contrainte d'égalité :

$$\mathbf{g}(\mathbf{x}) = \big(\mathbf{g}_1(\mathbf{x}),\ \mathbf{g}_2(\mathbf{x}),\ \ldots, \mathbf{g}_\kappa(\mathbf{x})\big) = 0 \tag{4.107}$$

($\kappa < n$). On note $\mathbf{x}_A^\star$ le point qui réalise cet optimum (global ou local). En ce point, les conditions de stationnarité du lagrangien (conditions Karush-Kuhn-Tucker) sont satisfaites :

$$\nabla f_A^\star + \sum_{k=1}^{\kappa} \mu_k \nabla \mathbf{g}_k^\star = 0 \tag{4.108}$$

où l'indice supérieur * indique l'évaluation en $\mathbf{x} = \mathbf{x}_A^\star$ et les coefficients $\{\mu_k\}$ sont des multiplicateurs de Lagrange. On modifie la fonction f_A comme suit pour garantir la convexité :

$$f_A(\mathbf{x}) \longrightarrow f_A(\mathbf{x}) + \frac{c}{2} \left\|\mathbf{x} - \mathbf{x}_A^\star\right\|^2 \tag{4.109}$$

où c est une constante choisie suffisamment grande. Cette modification ne change pas le point d'optimalité.

Étape 2- On effectue un post-traitement des résultats de l'étape précédente. On calcule, ou on estime par méta-modèle local [10], les informations de "sensibilités locales" suivantes :

— la matrice hessiennne de $f_A(\mathbf{x})$ en $\mathbf{x} = \mathbf{x}_A^\star$:

$$\mathbf{H}_A^\star = \left\{ \frac{\partial^2 f_A}{\partial x_i \partial x_j}(\mathbf{x}_A^\star) \right\} \tag{4.110}$$

(matrice symétrique $n \times n$ des dérivées secondes) ; on a la garantie que cette matrice est réelle-symétrique est définie-positive grâce au rajout du terme $\frac{c}{2}\left\|\mathbf{x} - \mathbf{x}_A^\star\right\|^2$;

10. Si l'estimation rend nécessaire la construction d'un méta-modèle, d'autres évaluations du critère $f_A(\mathbf{x})$ dans le voisinage de $\mathbf{x}_A^\star$ sont nécessaires.

— les gradients des contraintes d'égalité,

$$\{\nabla \mathbf{g}_k^\star\}_{(k=1,\ldots,\kappa)} \tag{4.111}$$

que l'on suppose linéairement indépendants, une hypothèse standard de "qualification des contraintes".

Les vecteurs $\{\nabla \mathbf{g}_k^\star\}$ $(k = 1, \ldots, \kappa)$ engendrent un sous-espace $\mathcal{N}_C$ de $\mathbb{R}^n$ de dimension κ. En leur appliquant le processus d'orthogonalisation de Gram-Schmidt, on définit κ nouveaux vecteurs orthonormés $\{\omega^1,\ \omega^2,\ \ldots,\ \omega^\kappa\}$ tels que $\left(\omega^j, \omega^k\right) = \delta_{j,k}$ $(\forall j, k)$, qui engendrent pareilllement $\mathcal{N}_C$. On note $\left[\omega^j\right]$ le vecteur-colonne des composantes de ω^j, vu comme une matrice $n \times 1$. On forme la matrice $\mathbf{P}$ de projection sur le sous-espace tangent aux contraintes $\mathcal{T}_C$, sous-espace supplémentaire de $\mathcal{N}_C$:

$$\mathbf{P} = \mathbf{I} - \sum_{k=1}^{\kappa} \left[\omega^k\right] \left[\omega^k\right]^t \tag{4.112}$$

dont $\mathcal{N}_C$ est le noyau :

$$\forall k = 1, \ldots, \kappa \ :\ \mathbf{P}\left[\omega^k\right] = 0. \tag{4.113}$$

Une illustration de ces sous-espaces est donnée par la Figure 4.26 dans le cas d'une seule contrainte scalaire $\mathbf{g}_1(\mathbf{x}) = 0$.

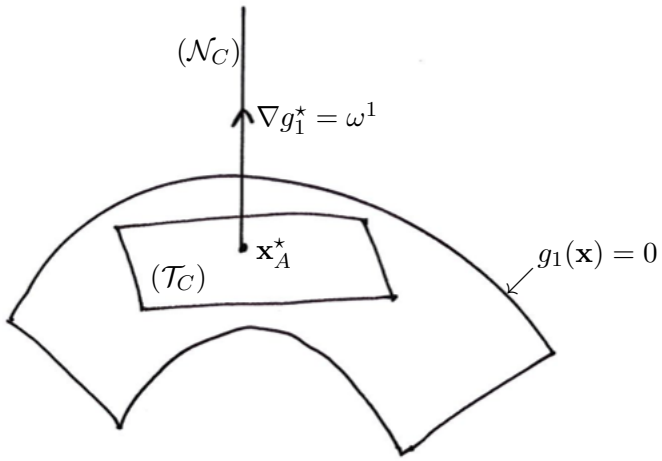


FIGURE 4.26 – Illustration des sous-espaces tangent, $\mathcal{T}_C$, et normal, $\mathcal{N}_C$, à la surface de contrainte lorsque celle-ci est scalaire.

On calcule et on diagonalise la matrice Hessienne réduite :

$$\mathbf{H}'_A = \mathbf{P}\mathbf{H}_A\mathbf{P} = \mathbf{\Omega}\mathcal{H}\mathbf{\Omega}^t \tag{4.114}$$

Les vecteurs-colonnes de la matrice $\mathbf{\Omega}$ forment la base orthogonale des vecteurs-propres de la matrice $\mathbf{H}'_A$. Parmi ceux-ci, on retrouve obligatoirement les vecteurs $\{\omega^1,\ \omega^2,\ \ldots,\ \omega^\kappa\}$ qui engendrent le noyau commun aux matrices $\mathbf{P}$ et $\mathbf{H}'_A$, associés aux valeurs propres $h_1 = h_2 =$

$\cdots = h_\kappa = 0$, ainsi que d'autres $\{\omega^{\kappa+1},\ \omega^{\kappa+2},\ \ldots,\ \omega^n\}$ orthogonaux aux précédents, associés aux valeurs propres strictement positives $h_{\kappa+1},\ h_{\kappa+2},\ \ldots,\ h_n$, de sorte que

$$\mathbf{\Omega} = \begin{pmatrix} \vdots & & \vdots & \vdots & & \vdots \\ \left[\omega^1\right] & \ldots & \left[\omega^\kappa\right] & \left[\omega^{\kappa+1}\right] & \ldots & \left[\omega^n\right] \\ \vdots & & \vdots & \vdots & & \vdots \\ & & & & & \end{pmatrix}, \quad \mathcal{H} = \begin{pmatrix} 0 & & & & & \\ & \ddots & & & & \\ & & 0 & & & \\ & & & h_{\kappa+1} & & \\ & & & & \ddots & \\ & & & & & h_n \end{pmatrix}. \tag{4.115}$$

On fait l'hypothèse que l'on a ordonné les modes propres de telle sorte que :

$$h_{\kappa+1} \geq h_{\kappa+2} \geq \cdots \geq h_n > 0\,. \tag{4.116}$$

En procédant de la sorte, les κ premiers vecteurs propres engendrent la "direction de plus grande pente" portée par le vecteur $\nabla f_A^\star$, en raison de la condition d'optimalité (4.108). À l'inverse, les derniers vecteurs propres correspondent aux directions de moindre sensibilité de $f_A(\mathbf{x})$, que l'on va allouer à la stratégie d'un joueur virtuel B en charge de la minimisation du critère secondaire $f_B(\mathbf{x})$ présumé antagoniste.

Les vecteurs propres ainsi ordonnés constituent une base de l'espace admissible, hiérarchisé selon la sensibilité du critère principal $f_A(\mathbf{x})$ aux variations du vecteur x dans ses directions caractéristiques, quelquefois appelées "fonctions principales".

À l'étape suivante, on utilise cette décomposition en modes propres pour partager les variables en "stratégies" associées à deux joueurs concurrents engagés dans un jeu de Nash virtuel.

Étape 3 - Les considérations précédentes nous amènent à séparer le territoire admissible en deux sous-espaces, selon le changement de variables suivant :

$$\mathbf{x} = \mathbf{x}_A^\star + \mathbf{\Omega} \begin{pmatrix} \mathbf{u} \\ \mathbf{v} \end{pmatrix} = \mathbf{x}(\mathbf{u}, \mathbf{v}) \qquad (\mathbf{u} \in \mathbb{R}^{n-p},\ \mathbf{v} \in \mathbb{R}^p) \tag{4.117}$$

de telle sorte que le sous-vecteur $\mathbf{u}$ engendre la stratégie du joueur A, et le sous-vecteur $\mathbf{v}$ celle d'un joueur concurrent B. L'entier $p \geq 1$ est à choisir tel que le joueur A dispose d'au moins $\kappa + 1$ degrés de liberté, de sorte que

$$1 \leq p \leq n - \kappa - 1\,. \tag{4.118}$$

On définit le critère suivant, concurrent de f_A :

$$f_{AB} = \frac{f_A}{f_A^\star} + \varepsilon \left(\theta \frac{f_B}{f_B^\star} - \frac{f_A}{f_A^\star} \right) \tag{4.119}$$

Ainsi, le critère f_{AB} est une combinaison convexe de f_A et f_B réglée par le paramètre de continuation $\varepsilon \in [0, 1]$. Le critère f_{AB} est proportionnel à f_A pour $\varepsilon = 0$, et à f_B pour $\varepsilon = 1$: le paramètre ε mesure donc le degré d'antagonisme que l'on souhaite introduire entre les critères dans le jeu. Le paramètre $\theta \in]0, 1]$ est un facteur de sous-relaxation dont l'intérêt est avant-tout théorique. En pratique, on utilise souvent $\theta = 1$.

Précisément, on organise la compétition par un jeu de Nash entre les critères suivants

$$\begin{cases} \tilde{f}_A(\mathbf{u},\mathbf{v}) = f_A\big(\mathbf{x}(\mathbf{u},\mathbf{v})\big) \quad \text{(sous contrainte } \mathbf{g}(\mathbf{u},\mathbf{v}) = \mathbf{g}\big(\mathbf{x}(\mathbf{u},\mathbf{v})\big) = 0\text{)}, \\ \tilde{f}_{AB}(\mathbf{u},\mathbf{v}) = f_{AB}\big(\mathbf{x}(\mathbf{u},\mathbf{v})\big) \quad \text{(sans constrainte)}. \end{cases} \tag{4.120}$$

respectivement associés aux joueurs A et B. Dans ce jeu, le joueur A (resp. B) tente de minimiser le critère $\tilde{f}_A$ (resp. $\tilde{f}_{AB}$) par son choix du sous-vecteur $\mathbf{u}$ (resp. $\mathbf{v}$) sous la contrainte du choix fait par le joueur B (resp. A) du sous-vecteur $\mathbf{v}$ (resp. $\mathbf{u}$). On cherche un équilibre de Nash entre $\mathbf{u}$ et $\mathbf{v}$.

Dans ce qui suit, pour alléger l'écriture, on omet les ˜ sur les symboles f_A, f_{AB} et $\mathbf{g}$ étant implicite que désormais, on considère ces critères comme des fonctions de $\mathbf{u}$ et $\mathbf{v}$.

Le résultat majeur qui découle de cette formulation est le suivant :

Proposition 6 (Consistance de la formulation)
(Sous les hypothèses faites dans cette sous-section, ainsi que : $\kappa \leq n-1$ et $1 \leq p \leq n-\kappa$.) Pour $\varepsilon = 0$, l'équilibre de Nash existe, et il correspond à la solution optimale du seul critère principal sous contrainte :

$$\bar{\mathbf{x}}_0 = \mathbf{x}_A^\star \,. \tag{4.121}$$

Preuve : en raison du changement de variable (4.117), il s'agit de démontrer que pour $\varepsilon = 0$,

$$\bar{\mathbf{u}} = \bar{\mathbf{v}} = 0 \tag{4.122}$$

correspond bien à un équilibre de Nash de la formulation. Cette question se décompose en deux, chacune symétrique de l'autre :

1. Question Q1 : Le sous-vecteur $\mathbf{v}$ étant fixé à $\bar{\mathbf{v}} = 0$, le sous-vecteur $\bar{\mathbf{u}} = 0$ est-il la solution du problème

$$\min_{\mathbf{u}} f_A(\mathbf{u},0) \text{ sous contrainte : } \mathbf{g}(\mathbf{u},0) = 0 \ ? \tag{4.123}$$

2. Question Q2 : Le sous-vecteur $\mathbf{u}$ étant fixé à $\bar{\mathbf{u}} = 0$, le sous-vecteur $\bar{\mathbf{v}} = 0$ est-il la solution du problème non-contraint

$$\min f_B(0,\mathbf{v}) \ ? \tag{4.124}$$

La question Q1 admet immédiatement la réponse affirmative : en effet, le problème partiel ainsi formulé correspond à la minimisation du même critère, sous la même contrainte que problème principal mono-discipline sauf que le domaine admissible y est restreint par la paramétrisation lorsque $\mathbf{v}$ est fixé à 0. Mais cet espace restreint contient le point d'optimalité associé au domaine admissible complet, atteint précisément pour $\mathbf{u} = 0$, c'est-à-dire $\mathbf{x} = \mathbf{x}_A^\star$ qui en est donc la solution.

Pour ce qui est du problème non contraint soulevé par la question Q2, examinons d'abord la condition de stationnarité. Pour $\varepsilon = 0$, f_{AB} se comporte comme f_A par proportionnalité. Sa diiférentielle s'exprime par le produit scalaire

$$df_{AB} = \frac{1}{f_A^\star}\big(\nabla f_A^\star, d\mathbf{x}\big) \tag{4.125}$$

et on doit restreindre $d\mathbf{x}$ à l'image de la paramétrisation lorsque $\mathbf{u} = 0$, et seul $\mathbf{v}$ varie :

$$d\mathbf{x} = \mathbf{\Omega} \begin{pmatrix} 0 \\ d\mathbf{v} \end{pmatrix} \tag{4.126}$$

Or, en raison des conditions nécessaires d'optimalité, (4.108), le vecteur $\nabla f_A^\star$ est engendré par les premiers vecteurs de la base orthogonale, $\{\omega^1,\ \omega^2,\ \ldots, \omega^\kappa\}$. À l'inverse, en vertu de la paramétrisation et du choix fait de la stratégie du joueur B, le vecteur $d\mathbf{x}$ est engendré par les derniers vecteurs de la base orthogonale, $\{\omega^n,\ \omega^{n-1},\ \ldots,\ \omega^{n-p+1}\}$. En conséquence, le produit scalaire (4.125) est nul pour autant que ces deux sous-familles soient disjointes, c'est-à-dire $n - p + 1 > \kappa$. De plus, la matrice hessienne de f_{AB} est proportionnelle à celle de f_A qui, par hypothèse, est définie-positive de sorte que la stationnarité est une condition suffisante d'optimalité.

Les deux questions posées admettent la réponse affirmative, ce qui établit le résultat. □

Ce résultat établit que bien que les problèmes d'optimisation partiels qui définissent l'équilibre de Nash soient posés dans des domaines admissibles différents entre eux, et distincts du domaine admissible du problème principal initial, la solution de celui-ci reste le point d'équilibre de la formulation de jeu concourant, au moins pour $\varepsilon = 0$. En conséquence, par continuité, on admet le

Corollaire 1
Sous le mêmes hypothèses que le Théorème 6, pour ε suffisamment petit, la formulation proposée est associée à un continuum de points d'équilbres de Nash paramétré par ε, $\bar{\mathbf{x}}_\varepsilon$, de telle sorte que $\bar{\mathbf{x}}_0 = \mathbf{x}_A^\star$.

Ce résultat nous amène à suivre les variations de critères le long du continuum en fonction du paramètre de continuation ε :

$$\phi_A(\varepsilon) = f_A\big(\bar{\mathbf{x}}_\varepsilon\big)\,, \quad \phi_B(\varepsilon) = f_B\big(\bar{\mathbf{x}}_\varepsilon\big)\,, \quad \phi_{AB}(\varepsilon) = f_{AB}\big(\bar{\mathbf{x}}_\varepsilon\big)\,. \tag{4.127}$$

Quand ε varie, la contrainte nonlinéaire est satisfaite en tout point :

$$\mathbf{g}\big(\bar{\mathbf{x}}_\varepsilon\big) = 0\,. \tag{4.128}$$

En dérivant cette relation par rapport à ε, il vient :

$$\big(\nabla \mathbf{g}_k\big(\bar{\mathbf{x}}_\varepsilon\big), \bar{\mathbf{x}}'_\varepsilon\big) = 0 \qquad (\forall k = 1, \ldots, \kappa)\,, \tag{4.129}$$

où $\bar{\mathbf{x}}'_\varepsilon$ est la dérivée du point d'équilbre $\bar{\mathbf{x}}_\varepsilon$ par rapport à ε. En faisant $\varepsilon = 0$, il vient :

$$\big(\nabla \mathbf{g}_k^\star, \bar{\mathbf{x}}'_0\big) = 0 \qquad (\forall k = 1, \ldots, \kappa)\,. \tag{4.130}$$

Ce résultat, injecté dans la condition d'optimalité, (4.108), donne

$$\phi'_A(0) = \big(\nabla f_A^\star, \bar{\mathbf{x}}'_0\big) = 0\,. \tag{4.131}$$

ce qui implique que, dans un voisinage de $\varepsilon = 0$, on a :

$$\phi_A(\varepsilon) = f_A^\star + O(\varepsilon^2)\,. \tag{4.132}$$

Au départ du continuum, la dégradation du critère principal f_A est du second-ordre en ε (sous-optimalité).

Examinons maintenant comment varie $\phi_{AB}(\varepsilon)$. L'expression de la dérivée

$$\phi'_{AB}(\varepsilon) = \frac{1}{f_A^\star}\phi'_A(\varepsilon) + \left(\theta\frac{\phi_B(\varepsilon)}{f_B^\star} - \frac{\phi_A(\varepsilon)}{f_A^\star}\right) + \varepsilon\left(\theta\frac{\phi'_B(\varepsilon)}{f_B^\star} - \frac{\phi'_A(\varepsilon)}{f_A^\star}\right) \tag{4.133}$$

pour $\varepsilon = 0$, se réduit à :

$$\phi'_{AB}(0) = \theta - 1\,. \tag{4.134}$$

Par conséquent, à condition que $\theta < 1$ (sous-relaxation), on a la garantie pour ε suffisamment petit, que le critère $\phi_{AB}(\varepsilon)$ décroit. L'intérêt du paramètre θ réside essentiellement dans ce résultat théorique ; en pratique, on le fixe presque toujours à 1.

Pour garantir que le critère f_B décroit, au prix d'une dégradation (augmentation) du critère f_A, des hypothèses supplémentaires doivent être faites en relation avec l'antagonisme entre ces critères.

Conclusion - Notre formulation d'un continuum de jeux de Nash paramétré par ε, est consistante avec l'optimisation du critère principal f_A sous contrainte d'égalité $\mathbf{g} = 0$ au sens suivant : il lui est associée un continuum de points d'équilibres de Nash paramétré par ε, $\bar{\mathbf{x}}_\varepsilon$, et $\bar{\mathbf{x}}_0 = \mathbf{x}_A^\star$, point d'optimalité de la discipline A seule. Pour ε suffisamment petit, la dégradation du critère principal f_A est du second-ordre en ε (sous-optimalité) et le critère auxiliaire f_{AB} décroit.

Notons enfin que si on incrémente le paramètre de continuation très progressivement, la convergence itérative de l'algorithme parallèle de coordination est facilement atteinte, car chaque nouveau point d'équilibre peut être rendu aussi proche que souhaité du précédent.

4.4.5 Exemples

Cas-test académique - On considère la minimisation de deux formes quadratiques de 4 variables ($N = 4$) :

$$f_A(\mathbf{x}) = \sum_{i=1}^{4} \frac{x_i^2}{3^i}\,, \qquad f_B(\mathbf{x}) = \sum_{i=1}^{4} x_i^2\,, \tag{4.135}$$

soumis à une contrainte d'égalité ($\kappa = 1$) :

$$\mathbf{g} = g_1 = x_1^4 x_2^3 x_3^2 x_4\,. \tag{4.136}$$

On attribue deux variables à chaque critère ($p = 2$), et $\theta = 1$. Pour les détails de l'expérience numérique, voir : *Split of territories in concurrent optimization*, J.A.D., Inria Research Report 6108, October 2007 ; `https://hal.inria.fr/inria-00127194`

La Figure 4.27 illustre la variation des critères évalués à l'équilibre de Nash pour toute valeur du paramètre $\varepsilon \in [0, 1]$. Dans ce cas de contrainte non-linéaire, on observe que $\phi_A(\varepsilon)$ est monotone-croissante (dégration de performance), $\phi_{AB}(\varepsilon)$ monotone décroissante (amélioration de performance), mais que $\phi_B(\varepsilon)$ est d'abord décroissante, puis croissante. Le changement de monotonie s'opère en $\varepsilon \doteq 0.487$. Il n'y a aucun avantage à poursuivre au delà de cette valeur puisqu'alors les deux critères se dégradent.

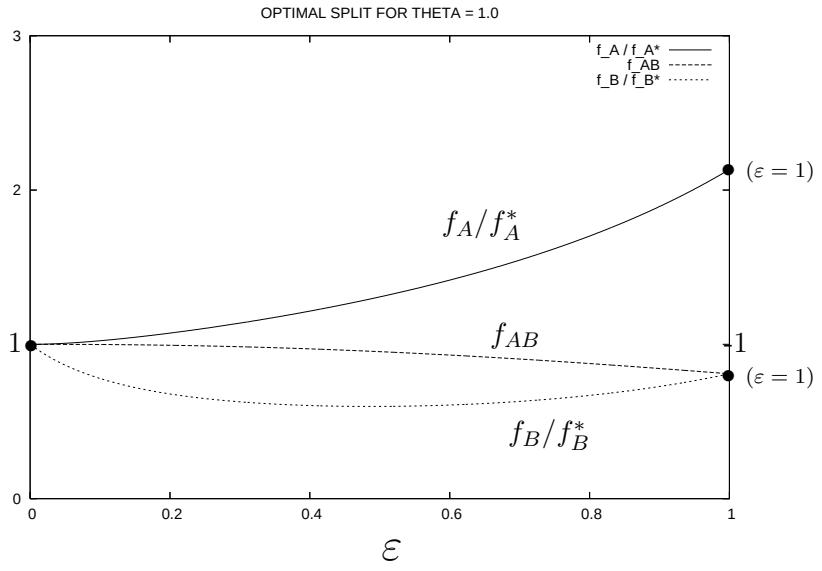


FIGURE 4.27 – Le continuum des équilibres de Nash quand ε varie

4.4.6 Application à la conception optimale de forme en aérodynamique

Les stratégies de jeux de Nash ont été appliquées en optimisation multidisciplinaire avec grand succès à l'Inria et en collaboration avec l'ONERA. En particulier, les trois applications suivantes ont été particulièrement marquantes. Dans ces applications, les variables ont été partagées suivant une stratégie de décomposition propre en fonctions principales (voir Section 4.4) :

- optimisation de forme d'une voilure d'avion transsonique pour en réduire la traînée (sous contrainte de portance) concouramment à un critère d'analyse structurale lié au champ de contraintes sur la coque (thèse de B. Abou El Majd, 2007, `http://www.theses.fr/2007NICE4073`) ;
- optimisation de forme d'une configuration d'avion supersonique pour en réduire la traînée (sous contrainte de portance) concouramment à un critère de *bang sonique* (thèse de A. Minelli, 2013, `http://www.theses.fr/2013NICE4107`) ;
- optimisation de forme d'une pale d'hélicoptère pour en maximiser la "figure de mérite" (finesse) en configuration stationnaire (*hover*) en concurrence avec la minimisation de la puissance à développer pour réaliser un avancement à vitesse donnée (thèse de E. Roca León, 2014) ; `http://www.theses.fr/2014NICE4076`).

Noter que dans les deux premières applications, la performance aérodynamique est évaluée par la simulation numérique d'un écoulement stationnaire (équations d'Euler ou RANS, en stationnaire). Ceci reste vrai dans la dernière application seulement pour l'évaluation de la figure de mérite, le calcul de puissance étant soumis à l'analyse d'un écoulement périodique en temps.

Une discussion approfondie de ces techniques est fournie dans la référence suivante :
Multiobjective Design Optimization Using Nash Games, J.-A. Désidéri, R. Duvigneau and A. Habbal, in : Computational Intelligence in Aerospace Sciences, M. Vasile and V. M. Becerra Eds., Progress in Astronautics and Aeronautics, volume 244, T. C. Lieuwen, Ed. in Chief,

AIAA Publish. Reston VA (2014).

4.4.7 Développements méthodologiques et applicatifs récents

La stratégie de jeu de Nash à deux disciplines a été étendue au cas d'un nombre quelconque de disciplines [20]-[22]-[24] On considère désormais un partage des fonctions-coûts en deux sous-familles :

1. les fonctions-coûts prioritaires $\{f_j\}$ $(j = 1, \ldots, m)$;
2. les fonctions-coûts secondaires $\{f_j\}$ $(j = m+1, \ldots, M)$.

Le problème est soumis à un ensemble de contraintes d'égalité $\mathbf{c} = 0$.

À nouveau l'optimisation est conduite en plusieurs étapes :

1. Optimisation multi-critère des seules fonctions-coûts prioritaires par tout algorithme efficace à cette fin ; par exemple par l'algorithme MGDA dans sa version avec contraintes (MGDA Quasi-riemannien : [19]) jusqu'à obtention d'un point Pareto-optimal $\mathbf{x}_A^\star$. Cette optimisation conduit à l'dentification d'une combinaison convexe de ces fonctions
$$f_A(\mathbf{x}) = \sum_{j=1}^{m} \alpha_j \frac{f_j(\mathbf{x})}{f_j(\mathbf{x}_A^\star)} \tag{4.137}$$
atteignant un minimum sous contrainte en $\mathbf{x}_A^\star$.
2. Calcul de la matrice hessienne $\mathbf{H}_A^\star = \nabla^2 f_A(\mathbf{x}_A^\star)$ et de la matrice de projection $\mathbf{P}$ liée aux contraintes ; convexification. Construction du partage de territoire s'appuyant, comme précédemment, sur la diagonalisation de la matrice hessienne réduite $\mathbf{H}'_A = \mathbf{P}\mathbf{H}_A^\star\mathbf{P}$.
3. Définition d'une nouvelle variable $\mathbf{w}$ liée à $\mathbf{v}$, et calcul des gradients des fonctions-coûts secondaires par rapport à $\mathbf{w}$ (à $\mathbf{u}$ fixé). Test de Pareto-stationnarité lié à ces gradients réduits. Le test s'effectue par appel au logiciel de la plateforme MGDA `http://mgda.inria.fr`. Si le test est négatif, on est dans le cas général favorable d'une situation de non Pareto-stationnarité ; on a alors la garantie de l'efficacité de l'approche par jeu de Nash. La plateforme fournit les coefficients $\{\alpha_j\}$ $(j = m+1, \ldots, M)$ d'une combinaison convexe particulière des fonctions-coûts secondaires
$$f_B(\mathbf{x}) = \sum_{j=m+1}^{M} \alpha_j \frac{f_j(\mathbf{x})}{f_j(\mathbf{x}_A^\star)} \tag{4.138}$$
4. Application de la technique de jeu de Nash à deux disciplines au couple (f_A, f_B) sous contrainte $\mathbf{c} = 0$.

La construction permet de garantir l'existence d'un continuum d'équilibres de Nash paramétré par ε et ayant pour origine le point de Pareto-optimalité $\mathbf{x}_A^\star$. Quand ε croît, la condition de Pareto-stationnarité de f_A est maintenue au second-ordre près en ε, alors que les fonctions-coûts secondaires décroissent initialement au moins aussi vite que $-\sigma\varepsilon$, où σ est une constante strictement positive fournie par la plateforme au cours du test de Pareto-stationarité.

Dans cette extension, les fonctions-coûts f_A et f_B ne sont pas définies dans la formulation initiale du problème, mais construites par le calcul dans un contexte plus vaste d'optimisation priorisée.

Cette extension est plus puissante que la théorie précédente à deux disciplines

- par le cadre étendu à deux familles de fonctions-coûts (les prioritaires et les secondaires),
- et par le résulat de convergence plus fort qui permet de savoir, avant le calcul du continuum des équilibres de Nash, si celui-ci permet la réduction effective des critères secondaires tout en préservant la quasi Pareto-optimalité des critères principaux.

Application à l'optimisation structurale d'un élément sandwich en aluminium - Les épaisseurs des couches constitutives de l'élément sandwich ont été optimisées [24] (4 variables) afin de prendre en compte concouramment jusqu'à 4 fonctions coûts liées à :

— la minimisation de la masse de l'élément,
— la maximisation de la résistance mécanique de l'élément soumis à un chargement en flexion par l'optimisation de deux fonctions coûts liées aux forces critiques de défaillance de l'élément suivant les deux premiers modes,
— la maximisation de la résistance de l'élément à une explosion par la maximisation de l'énergie absorbée par la couche centrale (cœur) et la minimisation de la déflection au centre.

Les expériences numériques ont été réalisées au moyen de la plateforme logicielle

`http://mgda.inria.fr`

Optimisation du dimensionnement d'un avion d'affaires supersonique (SSBJ) - Une expérience numérique de démonstration a été réalisée [20] en utilisant la plateforme logicielle de l'Inria. L'application concerne le dimensionnement d'un avion (15 variables "métier") pour optimiser sa performance de vol (masse au décollage, rayon d'action, distance de décollage, vitesse d'approche). Pour réaliser cette expérience, on a utilisé des logiciels de mécanique du vol (Lois de Bréguet) développés chez Dassault Aviation et mis à disposition dans le cadre du Projet ANR OMD ("Optimisation Multi-Disciplinaire") [10] [11] [54].

Chapitre 5

Bibliographie

Bibliographie

[1] B. Abou El Majd, *Hierarchical algorithms and game strategies for multidisciplinary optimization : application to the wing shape optimization of a business jet*, PhD thesis, Université Nice Sophia-Antipolis, 2007. `https://www.theses.fr/2007NICE4073`.

[2] G. Allaire, *Analyse numérique et optimisation*, Les Editions de l'Ecole Polytechnique, Palaiseau, 2005. Ellipses, ISBN 2-7302-1255-8.

[3] ——, *Conception optimale de structures*, Springer, 2007.

[4] R. Allmendinger, A. Jaszkiewicz, A. Liefooghe, and C. Tammer, *What if we increase the number of objectives ? theoretical and empirical implications for many-objectve combinatorial optimization*, Computers and Operations Research, 145 (2022).

[5] J.-P. Aubin, *Mathematical methods of game and economic theory*, Courier Corporation, 2007.

[6] M. Bartholomew-Biggs, *Nonlinear Optimization with Engineering Applications*, Springer, 2008.

[7] S. P. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004. ISBN 978-0-521-83378-3. Retrieved October 3, 2011.

[8] A. E. Bryson, Jr. and Y.-C. Ho, *Applied Optimal Control, Optimization, Estimation, and Control*, Blaisdell Publishing Company, Waltham, Massachussets, Toronto, London, 1969.

[9] L. Cambier and J.-P. Veuillot, *Status of the elsA CFD software for flow simulation and multidisciplinary applications*, AIAA Paper 2008-664, 2008.

[10] R. F. Coelho and P. Breitkopf, eds., *Optimisation multidisciplinaire en mécanique 1 - démarche de conception, stratégies collaboratives et concourantes, multiniveaux de modèles et de paramètres*, Mécanique et Ingénierie des Matériaux, Hermes Lavoisier, Paris, 2009. `www.hermes-science.com` `www.lavoisier.fr`.

[11] ——, eds., *Optimisation multidisciplinaire en mécanique 2 - réduction de modèles, robustesse, fiabilité, réalisations logicielles*, Mécanique et Ingénierie des Matériaux, Hermes Lavoisier, Paris, 2009. `www.hermes-science.com` `www.lavoisier.fr`.

[12] I. Das and J. Dennis, *Normal-boundary intersection : An alternate method for generating pareto optimal points in multicriteria optimization problems*, ICASE Report 96-62, ICASE, 1996.

[13] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, *A fast and elitist multiobjective genetic algorithm : NSGA-II*, IEEE Transactions on Evolutionary Computation, 6 (2002), pp. 182–197.

[14] M. C. Delfour and J.-P. Zolésio, *Shapes and Geometries - Metrics, Analysis, Differential Calculus, and Optimization*, Advances in Design and Control, SIAM, second ed., 2010.

[15] J.-A. Désidéri, *Modèles discrets et schémas itératifs - application aux algorithmes multigrilles et multidomaines*, Editions Hermès, Paris, 1998.

[16] ——, *Multiple-gradient descent algorithm (MGDA) for multiobjective optimization*, Comptes Rendus de l'Académie des Sciences Paris, 350 (2012), pp. 313–318. http://dx.doi.org/10.1016/j.crma.2012.03.014.

[17] ——, *Numerical Methods for Differential Equations, Optimization, and Technological Problems*, vol. 34 of Modeling, Simulation and Optimization for Science and Technology, Fitzgibbon, W. ; Kuznetsov, Y.A. ; Neittaanmäki, P. ; Pironneau, O. Eds., Springer-Verlag, 2014, ch. Multiple-Gradient Descent Algorithm (MGDA) for Pareto-Front Identification. J. Périaux and R. Glowinski Jubilees.

[18] ——, *Révision de l'algorithme de descente à gradients multiples (MGDA) par orthogonalisation hiérarchique*, Research Report 8710, INRIA, April 2015. https://hal.inria.fr/hal-01139994.

[19] ——, *Quasi-riemannian multiple gradient descent algorithm for constrained multiobjective differential optimization*, Research Report 9159, INRIA, 21 March 2018. https://hal.inria.fr/hal-01740075v1.

[20] ——, *Platform for prioritized multi-objective optimization by metamodel-assisted Nash games*, Research Report 9290, INRIA, September 2019.

[21] J.-A. Désidéri and R. Duvigneau, *Parametric optimization of pulsating jets in unsteady flow by Multiple-Gradient Descent Algorithm (MGDA)*, in Numerical Methods for Differential Equations, Optimization, and Technological Problems, Modeling, Simulation and Optimization for Science and Technology, J. Périaux, W. Fitzgibbon, B. Chetverushkin, and O. Pironneau, eds., Jan 2017. https://hal.inria.fr/hal-01414741v1.

[22] ——, *Direct and adaptive approaches to multi-objective optimization*, Research Report 9291, Inria, September 2019. https://hal.inria.fr/hal-02285899.

[23] J.-A. Désidéri, R. Duvigneau, and A. Habbal, *Computational Intelligence in Aerospace Sciences, V. M. Becerra and M. Vassile Eds.*, vol. 244 of Progress in Astronautics and Aeronautics, T. C. Lieuwen Ed.-in-Chief, American Institute for Aeronautics and Astronautics Inc., Reston, Virginia, 2014, ch. Multi-Objective Design Optimization Using Nash Games.

[24] J.-A. Désidéri, P. Leite, and Q. Mercier, *Prioritized multi-objective optimization of a sandwich panel*, Research Report 9362, Inria, October 2020. https://hal.inria.fr/hal-02931770.

[25] J.-A. Désidéri et al, *MGDA Platform, Multiple Gradient Descent Algorithm for Multi-Objective Differentiable Optimization*. http://mgda.inria.fr.

[26] L. G. Drummond and B. Svaiter, *A steepest descent method for vector optimization*, Journal of Computational and Applied Mathematics, (2005), pp. 395–414.

[27] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons, Chichester, New York, 1987.

[28] J. Fliege and B. F. Svaiter, *Steepest descent methods for multicriteria optimization*, Mathematical Methods of Operations Research, (2000), pp. 479–494.

[29] M. B. Giles and N. A. Pierce, *An introduction to the adjoint approach to design*, Flow, Turbulence and Combustion, (2000), pp. 393–415.

[30] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, New York London, twelfth printing 2000 ed., 1986.

[31] N. Hansen and A. Ostermeier, *Completely derandomized self-adaptation in evolution strategies*, Evolutionary Computation, 9 (2001), pp. 159–195.

[32] M. Hartikainen and K. Miettinen, *Constructing a pareto front approximation for decision making*, Mathematical Methods of Operations Research, (2011), pp. 209–234.

[33] M. Hartikainen, K. Miettinen, and M. M. Wiecek, *PAINT : Pareto front interpolation for nonlinear multiobjective optimization*, Computational Optimization and Applications, (2012), pp. 845–867.

[34] J. Horn, N. Nafpliotis, and D. E. Goldberg, *A niched pareto genetic algorithm for multiobjective optimization*, in Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence, 1991.

[35] Inria Project Team Ecuador, *Tapenade, version 3.* `https://www-sop.inria.fr/tropics/`.

[36] D. R. Jones, M. Schonlau, and W. J. Welch, *Efficient global optimization of expensive black-box functions*, Journal of Global Optimization, 13 (1998), pp. 455–492.

[37] A. J. Keane and P. B. Nair, *Computational Approaches for Aerospace Design, The Pursuit of Excellence*, John Wiley & Sons Ltd, 2005.

[38] J. Knowles and D. Corne, *The Pareto Archived Evolution Strategy : A New Baseline Algorithm for Pareto Multiobjective Optimisation*, in Proceedings of the 1999 Congress on Evolutionary Computation, CEC 1999, January 1999. doi = 10.1109/CEC.1999.781913.

[39] E. R. León, *Aeromechanical simulations for the optimization of helicopter rotors in forward flight*, PhD thesis, Université Nice Sophia-Antipolis, 2014. `https://www.theses.fr/2014NICE4076`.

[40] E. R. León, A. L. Pape, J.-A. Désidéri, D. Alfano, and M. Costes, *Concurrent aerodynamic optimization of rotor blades using a nash game method*, Journal of the American Helicopter Society, 61 (2016), pp. 1–13. `http://ingentaconnect.com/contentone/ahs/jahs/2016/00000061/00000002/art00009`.

[41] M. Martinelli and R. Duvigneau, *On the use of second-order derivatives and metamodel-based Monte-Carlo for uncertainty estimation in aerodynamics*, Computer & Fluids, 37 (2010).

[42] M. Martinelli and L. Hascoët, *Tangent-on-tangent vs. tangent-on-reverse for second differentiation of constrained functionals*, in Advances in Automatic Differentiation, vol. 64 of Lecture Notes in Computational Science and Engineering, Springer Verlag, 2008.

[43] J. R. R. A. Martins and A. B. Lambe, *Multidisciplinary design optimization : A survey of architectures*, AIAA Journal, 51 (2013).

[44] Q. Mercier, F. Poirion, and J.-A. Désidéri, *A stochastic multiple gradient descent algorithm*, European Journal of Operational Research, ELSEVIER Publish., 271 (2018), pp. 808–817.

[45] A. Messac, A. Isamail-Yahaya, and C. Mattson, *The normalized normal constraint method for generating the pareto frontier*, Structural and Multidisciplinary Optimization, (2003), pp. 86–98.

[46] K. Miettinen, *Nonlinear Multiobjective Optimization*, Springer, 1998.

[47] A. Minelli, *Aero-acoustic shape optimization of a supersonic business jet*, PhD thesis, Université Nice Sophia-Antipolis, 2013. https://www.theses.fr/2013NICE4107.

[48] B. Mohammadi and O. Pironneau, *Applied Shape Optimization for Fluids*, Oxford University Press, second ed., 2009. http://ukcatalogue.oup.com/product/9780199546909.do.

[49] T. Murata and H. Hishibuchi, *Moga : Multi-objective genetic algorithms*, in Proc. of 1995 IEEE International Conference on Evolutionary Computation, Perth, Australia, 1995, pp. 289–294.

[50] K. Parsopoulos and M. Vrahatis, *Recent approaches to global optimization problems through particle swarm optimization*, Natural Computing, (2002), pp. 235–306.

[51] J. Périaux, F. Gonzalez, and D. S. C. Lee, *Evolutionary Optimization and Game Strategies for Advanced Multi-Disciplinary Design - Applications to Aeronautics and UAV Design*, vol. 75 of Intelligent Systems, Control and Automation : Science and Engineering, Springer, 2015.

[52] F. Poirion, Q. Mercier, and J.-A. Désidéri, *Descent algorithm for nonsmooth stochastic multiobjective optimization*, Computational Optimization and Applications, 68 (2017), pp. 317–331.

[53] Q.Mercier, F.Poirion, and J.-A. Désidéri, *Non-convex multiobjective optimization under uncertainty : a descent algorithm. application to sandwich plate design and reliability*, Engineering Optimization, 0 (2018), pp. 1–20.

[54] M. Ravachol, *Multidisciplinary Design Optimization in Computational Mechanics*, Wiley-ISTE, May 2010, ch. Multilevel Multidisciplinary Optimization in Airplane Design. https://www.wiley.com/en-fr/Multidisciplinary+Design+Optimization+in+Computational+Mechanics-p-9781848211384.

[55] M. Samuelides, *Optimisation multidisciplinaire en mécanique 2 - réduction de modèles, robustesse, fiabilité, réalisations logicielles, R. Filomeno Coelho & P. Breitkopf eds.*, Mécanique et Ingénierie des Matériaux, Hermes Science Lavoisier, Paris, 2009, ch. Surfaces de réponse et réduction de modèles, pp. 21–72.

[56] Z. Tang, J.-A. Désidéri, and J. Périaux, *Multi-criterion aerodynamic shape-design optimization and inverse problems using control theory and Nash games*, Journal of Optimization Theory and Applications, 135 (2007).

[57] J. H. Wilkinson, *The Algebraic Eigenvalue Problem (Numerical Mathematics and Scientific Computation)*, Oxford Science Publications, 1965 (first edition).

[58] A. Zerbinati, A. Minelli, I. Ghazlane, and J.-A. Désidéri, *Meta-model-assisted MGDA for multi-objective functional optimization*, Computers and Fluids, 102 (2014), pp. 116–130. http://www.sciencedirect.com/science/article/pii/S0045793014002576.

[59] E. Zitzler and L. Thiele, *Multiobjective evolutionary algorithms : A comparative case study and the strength Pareto approach*, IEEE Transactions on Evolutionary Computation, 3 (1999), pp. 257–271.

Annexe A

Notions sur le traitement des contraintes

A.1 Différents types de contraintes

On considère ici seulement les contraintes qui s'expriment au moyen de fonctions continues, et même généralement de classe $C^2(\Omega_{\text{ad}})$. Les principaux types de contraintes sont les suivants :

— contrainte d'égalité :

$$c(\mathbf{x}) = 0 \tag{A.1}$$

— contrainte d'inégalité (ou de borne) :

$$E(\mathbf{x}) \leq B \tag{A.2}$$

— contrainte d'intervalle :

$$a \leq F(\mathbf{x}) \leq b \tag{A.3}$$

dans lesquelles $\mathbf{x} \in \mathbf{\Omega}_{\text{ad}} \subseteq \mathbb{R}^n$, et les fonctions $c(\mathbf{x})$, $E(\mathbf{x})$ et $F(\mathbf{x})$ sont à valeurs dans $\mathbb{R}$.

Nous allons d'abord montrer qu'avec la technique de la variable d'ajustement ou d'écart [1], et un peu d'astuce, on peut, dans tous ces cas, se ramener à une contrainte d'égalité nonlinéaire quitte à augmenter le nombre de variables d'optimisation indépendantes.

A.1.1 Contrainte d'égalité linéaire

Si la fonction $c(\mathbf{x})$ de (A.1) est linéaire par rapport à au moins une variable, disons x_n, on peut simplement résoudre l'équation formellement par rapport à cette variable, et par substitution, éliminer la variable de la formulation, ce qui réduit n d'une unité.

A.1.2 Contrainte d'inégalité

De nombreuses possibilités existent pour transformer (A.2). On peut en particulier poser

$$c(\mathbf{x}) = E(\mathbf{x}) - B + x_{n+1}^2 = 0 \tag{A.4}$$

où x_{n+1} est une variable d'optimisation additionnelle, dite d'ajustement, ce qui augmente n d'une unité.

1. en anglais : *"slack variable"*

A.1.3 Contrainte d'intervalle

De nombreuses possibilités existent pour transformer (A.3) par lesquelles une seule variable d'ajustement n'est introduite. Par exemple, on pose :

$$c(\mathbf{x}) = F(\mathbf{x}) - \left(\frac{a+b}{2} + \frac{b-a}{2} \sin x_{n+1} \right) = 0. \tag{A.5}$$

Exercice 19
Proposer des solutions alternatives à (A.4)-(A.5).

A.1.4 Conclusion

L'introduction de variables d'ajustement permet de ramener les contraintes usuelles à des contraintes d'égalités nonlinéaires pour lesquelles les conditions d'optimalité sont rappelées par la description intuitive de la Section A.2. Néanmoins, ces conditions peuvent aussi être exprimées directement dans le cas formellement plus général où les contraintes d'égalités et d'inégalités sont distinguées (Section A.3).

A.2 Introduction intuitive aux conditions nécessaires d'optimalité de Karush-Kuhn-Tucker (KKT)

A.2.1 Optimisation dans $\mathbb{R}^2$

Soit f et c deux fonctions régulières d'un domaine admissible $\mathbf{\Omega}_{ad} \subseteq \mathbb{R}^2$ dans $\mathbb{R}$. On considère le problème :

$$\begin{cases} \min\limits_{\mathbf{x} \in \mathbf{\Omega}_{ad}} f(\mathbf{x}) \\ \text{sous la contrainte : } c(\mathbf{x}) = 0 \end{cases} \tag{A.6}$$

où $\mathbf{x} = (x, y)$, dont on suppose qu'il admet une solution. En général, localement la contrainte $c = 0$ intersecte une ligne de niveau de f, ou courbe iso-valeur, en 0 ou 2 points. Quand les deux points se confondent, les deux courbes sont tangentes, et l'optimum sous contrainte est atteint (voir Fig. A.1). Les normales à ces courbes sont données par les gradients. Ceux-ci sont parallèles ssi il existe une constante $\lambda \in \mathbb{R}$ telle que :

$$\nabla f = -\lambda \nabla c \tag{A.7}$$

Il vient donc :

$$\nabla(f + \lambda c) = 0. \tag{A.8}$$

La règle est donc la suivante : on adjoint la contrainte c à la fonction f à minimiser au moyen d'un multiplicateur de Lagrange λ pour former le lagrangien

$$\mathcal{L} = \mathcal{L}(\mathbf{x}; \lambda) = f + \lambda c \tag{A.9}$$

dont on exprime la stationnarité par rapport à la variable d'optimisation, ici le couple $\mathbf{x} = (x, y)$,

$$\mathcal{L}_{\mathbf{x}} = \nabla \mathcal{L} = \nabla(f + \lambda c) = 0 \tag{A.10}$$

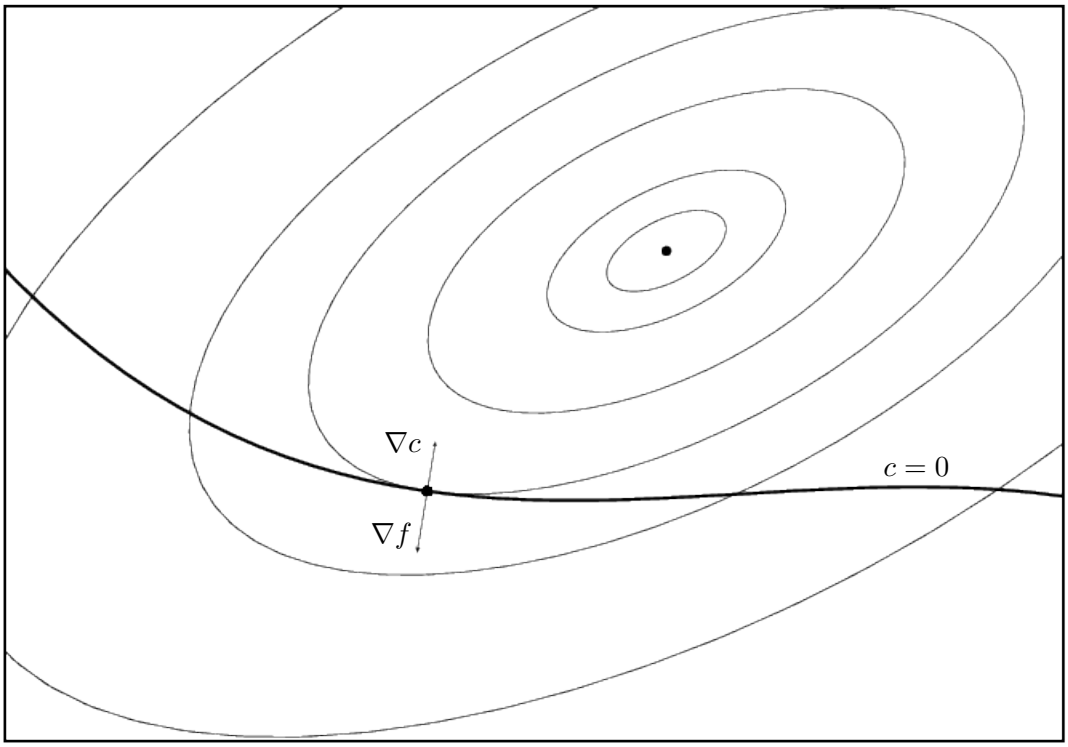


FIGURE A.1 – Réseau des courbes de niveau de f (ici des ellipses), avec le minimum absolu au centre ; courbe associée à la contrainte $g = 0$; gradients au point du minimum sous contrainte

ainsi que par rapport au multiplicateur de Lagrange λ (pour exprimer la contrainte) :

$$\mathcal{L}_\lambda = c(\mathbf{x}) = 0 \tag{A.11}$$

Ces 2+1 équations scalaires permettent en général d'identifier les 3 inconnues du problème : les deux composantes du point optimal $\mathbf{x}^\star = (x^\star, y^\star)$, et la valeur du multiplicateur de Lagrange $\lambda^\star$.

A.2.2 Généralisation au cas de p contraintes scalaires d'égalité, et $n > p \geq 2$

On considère à nouveau le problème d'optimisation sous contrainte (A.6), mais maintenant c est à valeurs dans $\mathbb{R}^p$ ($p \geq 2$),

$$c(\mathbf{x}) = \Big(c_1(\mathbf{x}), c_2(\mathbf{x}), \ldots, c_p(\mathbf{x})\Big)^t \tag{A.12}$$

et

$$\mathbf{x} = (x_1, x_2, ..., x_n)^t \in \mathbf{\Omega}_{ad} \subseteq \mathbb{R}^n \tag{A.13}$$

avec $n > p \geq 2$. Précédemment, dans le cas de 2 variables ($n = 2$), il était naturel d'adopter pour le vecteur $\mathbf{x}$ la notation d'un vecteur ligne. Pour $n > 2$, on choisit à l'inverse, de considerer les vecteurs $\mathbf{x}$ et c comme des vecteurs colonnes de dimension n et p respectivement, soumis à la condition $n > p$. On adopte désormais les notations de l'algèbre linéaire (matrice, vecteurs colonnes, etc) ; l'indice supérieur ^{t} correspond à la transposition et sert notamment à former les produits scalaires.

Au point $\mathbf{x}$, la différentielle de f s'exprime au moyen du gradient :

$$Df(\mathbf{x}; d\mathbf{x}) = [\nabla f]^t d\mathbf{x} = \sum_{i=0}^{n} \frac{\partial f}{\partial x_i} dx_i \tag{A.14}$$

où $[\nabla f]^t$ est le vecteur-ligne des composantes du gradient. On doit exprimer que :

$$Df(\mathbf{x}; d\mathbf{x}) = 0 \qquad (\forall\, d\mathbf{x} \in \mathcal{P}) \tag{A.15}$$

où $\mathcal{P}$ est le sous-espace vectoriel tangent à la variété de $\mathbb{R}^n$ d'équation $c = 0$ au point $\mathbf{x}$. Or ces vecteurs $d\mathbf{x}$ sont précisément caractérisés au moyen de la différentielle de c par la condition :

$$Dc(\mathbf{x}; d\mathbf{x}) = [\nabla c]^t d\mathbf{x} = 0 \tag{A.16}$$

où $[\nabla c]^t$ est la matrice à p lignes et n colonnes suivante :

$$[\nabla c]^t = \begin{pmatrix} [\nabla c_1]^t \\ [\nabla c_2]^t \\ \vdots \\ [\nabla c_p]^t \end{pmatrix} \tag{A.17}$$

La condition équivaut donc à l'implication

$$\Big(d\mathbf{x} \text{ tel que } Dc(\mathbf{x}; d\mathbf{x}) = 0\Big) \Longrightarrow \Big(d\mathbf{x} \text{ tel que } Df(\mathbf{x}; d\mathbf{x}) = 0\Big) \tag{A.18}$$

Cette implication est satisfaite lorsqu'il existe un multiplicateur $\lambda \in \mathbb{R}^p$ tel que :

$$[\nabla f]^t = -\,[\nabla c]^t\, \lambda. \tag{A.19}$$

On définit donc encore le lagrangien en adjoignant la contrainte c à la fonction f à minimiser au moyen d'un multiplicateur de Lagrange λ, ici vectoriel ($\lambda \in \mathbb{R}^p$) :

$$\mathcal{L} = \mathcal{L}(x; \lambda) = f(\mathbf{x}) + \lambda^t c(\mathbf{x}) = f(\mathbf{x}) + \sum_{k=1}^{p} \lambda_k c_k(\mathbf{x}) \tag{A.20}$$

On exprime la stationnarité du lagrangien par rapport à la variable indépendante $\mathbf{x}$:

$$\mathcal{L}_x^t = [\nabla f] + \lambda^t\, [\nabla c] = [\nabla f] + \sum_{k=1}^{p} \lambda_k\, [\nabla c_k] = 0 \tag{A.21}$$

pour traduire l'extrémalité, et par rapport au multiplicateur de Lagrange λ pour traduire les contraintes :

$$\mathcal{L}_\lambda^t = c(\mathbf{x}) = 0 \tag{A.22}$$

(Les transpositions proviennent de la convention adoptée selon laquelle $\mathbf{x}$ et λ sont des vecteurs colonnes.) Les équations (A.21) et (A.22) fournissent $n + p$ équations scalaires pour les $n + p$ inconnues (n composantes de $\mathbf{x}$, p multiplicateurs de Lagrange λ_k). Si le problème d'optimisation est bien posé, on trouve sa solution parmi la, ou les solution(s) de ces équations.

A.2.3 Cas des fonctionnelles types du calcul des variations

On considère le problème de la minimisation de la fonctionnelle $J(y)$ de (1.1) sous les p contraintes suivantes :

$$G_k(y) = \int_a^b N_k\Big(s; y(s), y'(s), ..., y^{(n)}(s)\Big)\, ds - C_k = 0 \tag{A.23}$$

où les p constantes C_k $(k = 1, ..., p)$ sont données.

A nouveau, on forme un lagrangien en adjoignant les p contraintes à la fonctionnelle $J(y)$ à minimiser au moyen de p multiplicateurs de Lagrange $\{\lambda_k\}_{(k=1,...,p)}$:

$$\mathcal{L} = \mathcal{L}(y; \lambda) = J(y) + \sum_{k=1}^{p} \lambda_k G_k(y) \tag{A.24}$$

Il vient :

$$\mathcal{L} = \int_a^b \Phi\Big(s; y(s), y'(s), ..., y^{(n)}(s)\Big)\, ds - C \tag{A.25}$$

où :

$$\begin{cases} \Phi = F + G^t\lambda = F + \lambda^t G = F + \displaystyle\sum_{k=1}^{p} \lambda_k N_k \\ C = \displaystyle\sum_{k=1}^{p} \lambda_k C_k \end{cases} \tag{A.26}$$

Alors, la stationnarité du lagrangien par rapport à y s'exprime par l'équation d'Euler-Lagrange, (1.18), dans laquelle on remplace la fonction F par Φ. La stationnarité de $\mathcal{L}$ par rapport à λ redonne les contraintes.

A.3 Qualification des contraintes

Plusieurs jeux de conditions suffisantes (voir extraits de Wikipedia attachés) ; notamment :
— Indépendance linéaire des gradients de contraintes
— Condition de Slater

Karush–Kuhn–Tucker conditions

From Wikipedia, the free encyclopedia
(Redirected from Karush–Kuhn–Tucker)

In mathematical optimization, the **Karush–Kuhn–Tucker (KKT) conditions** (also known as the **Kuhn–Tucker conditions**) are first order necessary conditions for a solution in nonlinear programming to be optimal, provided that some regularity conditions are satisfied. Allowing inequality constraints, the KKT approach to nonlinear programming generalizes the method of Lagrange multipliers, which allows only equality constraints. The system of equations corresponding to the KKT conditions is usually not solved directly, except in the few special cases where a closed-form solution can be derived analytically. In general, many optimization algorithms can be interpreted as methods for numerically solving the KKT system of equations.[1]

The KKT conditions were originally named after Harold W. Kuhn, and Albert W. Tucker, who first published the conditions in 1951.[2] Later scholars discovered that the necessary conditions for this problem had been stated by William Karush in his master's thesis in 1939.[3][4]

Nonlinear optimization problem [edit]

Consider the following nonlinear optimization problem:

Maximize $f(x)$

subject to

$$g_i(x) \leq 0, h_j(x) = 0$$

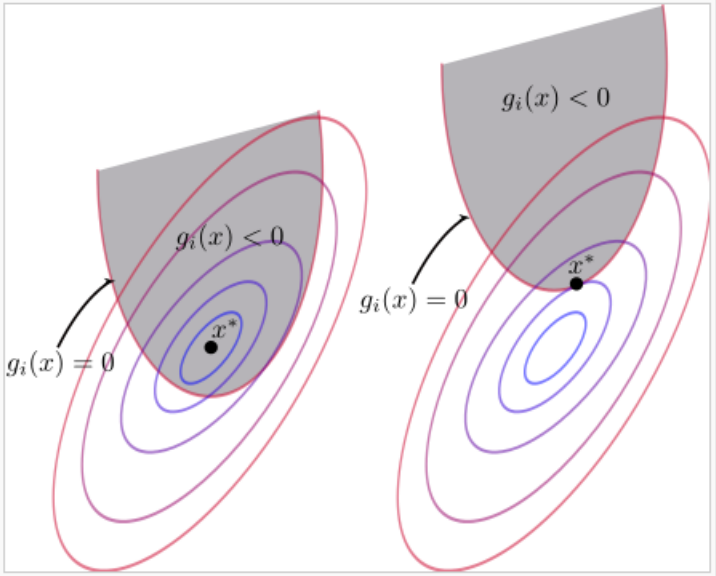
where *x* is the optimization variable, f is the *objective* or *cost* function, $g_i \ (i = 1, \ldots, m)$ are the inequality constraint functions, and $h_j \ (j = 1, \ldots, l)$ are the equality constraint functions. The numbers of inequality and equality constraints are denoted *m* and *l*, respectively.

Necessary conditions [edit]

Suppose that the objective function $f : \mathbb{R}^n \to \mathbb{R}$ and the constraint functions $g_i : \mathbb{R}^n \to \mathbb{R}$ and $h_j : \mathbb{R}^n \to \mathbb{R}$ are continuously differentiable at a point x^*. If x^* is a local minimum that satisfies some regularity conditions (see below), then there exist constants $\mu_i \ (i = 1, \ldots, m)$ and $\lambda_j \ (j = 1, \ldots, l)$, called KKT multipliers, such that

Stationarity

For maximizing f(x):



Inequality constraint diagram for optimization problems

$$\nabla f(x^*) = \sum_{i=1}^{m} \mu_i \nabla g_i(x^*) + \sum_{j=1}^{l} \lambda_j \nabla h_j(x^*),$$

For minimizing f(x): $-\nabla f(x^*) = \sum_{i=1}^{m} \mu_i \nabla g_i(x^*) + \sum_{j=1}^{l} \lambda_j \nabla h_j(x^*),$

Primal feasibility

$$g_i(x^*) \leq 0, \text{ for all } i = 1, \ldots, m$$
$$h_j(x^*) = 0, \text{ for all } j = 1, \ldots, l$$

Dual feasibility

$$\mu_i \geq 0, \text{ for all } i = 1, \ldots, m$$

Complementary slackness

$$\mu_i g_i(x^*) = 0, \text{for all } i = 1, \ldots, m.$$

In the particular case $m = 0$, i.e., when there are no inequality constraints, the KKT conditions turn into the Lagrange conditions, and the KKT multipliers are called Lagrange multipliers.

If some of the functions are non-differentiable, subdifferential versions of Karush–Kuhn–Tucker (KKT) conditions are available.[5]

Regularity conditions (or constraint qualifications) [edit]

In order for a minimum point x^* to satisfy the above KKT conditions, the problem should satisfy some regularity conditions; the most used ones are listed below:

- Linearity constraint qualification: If g_i and h_j are affine functions, then no other condition is needed.
- Linear independence constraint qualification (LICQ): the gradients of the active inequality constraints and the gradients of the equality constraints are linearly independent at x^*.
- Mangasarian–Fromovitz constraint qualification (MFCQ): the gradients of the active inequality constraints and the gradients of the equality constraints are positive-linearly independent at x^*.
- Constant rank constraint qualification (CRCQ): for each subset of the gradients of the active inequality constraints and the gradients of the equality constraints the rank at a vicinity of x^* is constant.
- Constant positive linear dependence constraint qualification (CPLD): for each subset of the gradients of the active inequality constraints and the gradients of the equality constraints, if it is positive-linear dependent at x^* then it is positive-linear dependent at a vicinity of x^*.
- Quasi-normality constraint qualification (QNCQ): if the gradients of the active inequality constraints and the gradients of the equality constraints are positive-linearly dependent at x^* with associated multipliers λ_i for equalities and μ_j for inequalities, then there is no sequence $x_k \to x^*$ such that $\lambda_i \neq 0 \Rightarrow \lambda_i h_i(x_k) > 0$ and $\mu_j \neq 0 \Rightarrow \mu_j g_j(x_k) > 0$.
- Slater condition: for a convex problem, there exists a point x such that $h(x) = 0$ and $g_i(x) < 0$.

$(v_1, \ldots, v_n)$ is positive-linear dependent if there exists $a_1 \geq 0, \ldots, a_n \geq 0$ not all zero such that $a_1 v_1 + \cdots + a_n v_n = 0$.

It can be shown that LICQ⇒MFCQ⇒CPLD⇒QNCQ, LICQ⇒CRCQ⇒CPLD⇒QNCQ (and the converses are not true), although MFCQ is not equivalent to CRCQ[6] . In practice weaker constraint qualifications are preferred since they provide stronger optimality conditions.

Sufficient conditions [edit]

In some cases, the necessary conditions are also sufficient for optimality. In general, the necessary conditions are not sufficient for optimality and additional information is necessary, such as the Second Order Sufficient Conditions (SOSC). For smooth functions, SOSC involve the second derivatives, which explains its name.

The necessary conditions are sufficient for optimality if the objective function f is a concave function, the inequality constraints g_j are continuously differentiable convex functions and the equality constraints h_i are affine functions.

It was shown by Martin in 1985 that the broader class of functions in which KKT conditions guarantees global optimality are the so-called Type 1 **invex functions**.[7][8]

A.4 Pénalisation

Cf. : *Introduction à l'analyse numérique matricielle et à l'optimisation*, P. G. Ciarlet, Masson, Paris, 1990 (p. 205).

Théorème 8.6-3. *Soit* $J : \mathbf{R}^n \to \mathbf{R}$ *une fonction continue coercive strictement convexe,* U *une partie non vide convexe fermée de* $\mathbf{R}^n$, *et* $\psi : \mathbf{R}^n \to \mathbf{R}$ *une fonction continue convexe vérifiant*

$$\psi(v) \geqslant 0 \quad \text{pour tout} \quad v \in \mathbf{R}^n \quad \text{et} \quad \psi(v) = 0 \Leftrightarrow v \in U.$$

Alors, pour tout $\varepsilon > 0$, *il existe un et un seul élément* u_ε *vérifiant*

$$(P_\varepsilon) \qquad u_\varepsilon \in \mathbf{R}^n \quad \text{et} \quad J_\varepsilon(u_\varepsilon) = \inf_{v \in \mathbf{R}^n} J_\varepsilon(v) \quad \text{où} \quad J_\varepsilon(v) \overset{\text{déf}}{=} J(v) + \frac{1}{\varepsilon}\psi(v),$$

et $\lim_{\varepsilon \to 0} u_\varepsilon = u$, *où* u *est la solution unique du problème : trouver* u *tel que*

$$(P) \qquad u \in U \quad \text{et} \quad J(u) = \inf_{v \in U} J(v).$$

Annexe B

Nested Pareto sets and fronts

Very intuitively, as the number of cost functions is increased, one expects the Pareto-optimality criterion to discriminate less, and the Pareto set and front to be enlarged without loss. This question is of some relevance since it is natural in case of concurrent engineering to introduce additional criteria gradually. In [4], this issue has been considered theoretically and empirically in the context of combinatorial optimization. For sake of completeness, a justification is provided here for the basic result according which, in a continuous and convex setting, Pareto sets and fronts are generally nested as the number of cost functions is increased.

Consider a set of $k+1$ ($k \geq 2$) continuous and convex cost functions $\{f_1(\mathbf{x}), \ldots, f_{k+1}(\mathbf{x})\}$ defined over the closed and convex domain $\Omega_{ad} \subseteq \mathbb{R}^n$. Let $\{\mathcal{S}_k, \mathcal{F}_k\}$ be the Pareto set and front associated with the first k cost functions, and $\{\mathcal{S}_{k+1}, \mathcal{F}_{k+1}\}$ the Pareto set and front associated with the whole set of cost functions.

Let $\bar{F}_k$ be an arbitrary element of $\mathcal{F}_k$, image of at least one point $\overline{\mathbf{x}}$ in the kth-dimensional function space, that is

$$\bar{F}_k = (\bar{f}_1, \bar{f}_2, \ldots, \bar{f}_k) = (f_1(\overline{\mathbf{x}}), f_2(\overline{\mathbf{x}}), \ldots, f_k(\overline{\mathbf{x}})). \tag{B.1}$$

Let $\bar{F}_{k+1} = (\bar{F}_k, f_{k+1}(\overline{\mathbf{x}}))$ be the image of $\overline{\mathbf{x}}$ in the $k+1$st-dimensional function space. Then, either $\bar{F}_{k+1} \in \mathcal{F}_{k+1}$, or there exists $\overline{\mathbf{x}}' \in \Omega_{ad}$ that dominates $\overline{\mathbf{x}}$ in efficiency w.r.t. $\{f_1, f_2, \ldots, f_{k+1}\}$, i.e. :

$$f_1(\overline{\mathbf{x}}') \leq f_1(\overline{\mathbf{x}}) = \bar{f}_1, \quad \ldots f_k(\overline{\mathbf{x}}') \leq f_k(\overline{\mathbf{x}}) = \bar{f}_k, \quad f_{k+1}(\overline{\mathbf{x}}') \leq f_{k+1}(\overline{\mathbf{x}}) = \bar{f}_{k+1}, \tag{B.2}$$

and at least one of the above inequalities is strict. The latter cannot be one of the first k since otherwise $\overline{\mathbf{x}}'$ would also dominate $\overline{\mathbf{x}}$ in efficiency in $\mathcal{S}_k$. Therefore, it must be the last one :

$$f_1(\overline{\mathbf{x}}') = f_1(\overline{\mathbf{x}}) = \bar{f}_1, \quad \ldots f_k(\overline{\mathbf{x}}') = f_k(\overline{\mathbf{x}}) = \bar{f}_k, \quad f_{k+1}(\overline{\mathbf{x}}') < f_{k+1}(\overline{\mathbf{x}}) = \bar{f}_{k+1}. \tag{B.3}$$

Since all the functions are continuous and the domain is closed, $f_{k+1}(\mathbf{x})$ subject to the constraints defined by the above equalities admits a minimum. Without loss of generality, we may assume that $\overline{\mathbf{x}}'$ realizes this minimum. Then we claim that $\overline{\mathbf{x}}' \in \mathcal{S}_{k+1}$. Indeed otherwise there would exist an element $\overline{\mathbf{x}}'' \in \Omega_{ad}$ that would dominate $\overline{\mathbf{x}}'$ in efficiency in $\mathcal{S}_{k+1}$, i.e. :

$$\begin{aligned} f_1(\overline{\mathbf{x}}'') \leq f_1(\overline{\mathbf{x}}') = f_1(\overline{\mathbf{x}}) = \bar{f}_1, \quad \ldots f_k(\overline{\mathbf{x}}'') \leq f_k(\overline{\mathbf{x}}') = f_k(\overline{\mathbf{x}}) = \bar{f}_k, \\ \text{and } f_{k+1}(\overline{\mathbf{x}}'') \leq f_{k+1}(\overline{\mathbf{x}}') < f_{k+1}(\overline{\mathbf{x}}) = \bar{f}_{k+1}, \end{aligned} \tag{B.4}$$

and at least one of the above large inequalities should be strict. The latter could not be one of the first k since otherwise $\overline{\mathbf{x}}''$ would dominate $\overline{\mathbf{x}}$ in efficiency in $\mathcal{S}_k$. Hence, the following should hold

$$f_1(\overline{\mathbf{x}}'') = f_1(\overline{\mathbf{x}}') = f_1(\overline{\mathbf{x}}) = \bar{f}_1,\ \ldots f_k(\overline{\mathbf{x}}'') = f_k(\overline{\mathbf{x}}') = f_k(\overline{\mathbf{x}}) = \bar{f}_k,\ f_{k+1}(\overline{\mathbf{x}}'') < f_{k+1}(\overline{\mathbf{x}}'). \quad \text{(B.5)}$$

But this is in contradiction with the fact that $\overline{\mathbf{x}}'$, among all the elements for which (B.3) holds, minimizes $f_{k+1}(\overline{\mathbf{x}}')$. To reject this contradiction, we conclude that indeed $\overline{\mathbf{x}}'$ belongs to $\mathcal{S}_{k+1}$, and its image in the $k+1$st-dimensional function space

$$\bar{F}'_{k+1} = (f_1(\overline{\mathbf{x}}'), \ldots, f_{k+1}(\overline{\mathbf{x}}')) \quad \text{(B.6)}$$

belongs to $\mathcal{F}_{k+1}$. But $\bar{F}'_{k+1} = (\bar{F}_k, f_{k+1}(\overline{\mathbf{x}}))$. Hence in the $k+1$st-dimensional function space, $\bar{F}_{k+1}$ admits the projection $\bar{F}'_{k+1}$ in the direction of the f_{k+1} axis that belongs to the front $\mathcal{F}_{k+1}$.

In conclusion, given the arbitrary element $\bar{F}_k \in \mathcal{F}_k$, and its extension $\bar{F}_{k+1} = (\bar{F}_k, f_{k+1}(\overline{\mathbf{x}}))$ to the $k+1$st-dimensional function space, either $\bar{F}_{k+1}$ belongs to the front $\mathcal{F}_{k+1}$, or it admits a projection in the direction the f_{k+1} axis onto the front $\mathcal{F}_{k+1}$.

We presently examine the inclusion of Pareto sets. For this, we further make the hypothesis that the existence of a line segment of the domain over which $\{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})\}$ are all constant is excluded. Then :

1. Uniqueness : $\bar{F}_k \in \mathcal{F}_k$ admits no antecedent in $\mathcal{S}_k$ other than $\overline{\mathbf{x}}$.
2. Inclusions : $\mathcal{S}_k \subseteq \mathcal{S}_{k+1}$, and the image of $\mathcal{S}_k$ in the $k+1$st-dimensional function space is included in $\mathcal{F}_{k+1}$.

Proof : Suppose that another point $\overline{\mathbf{y}}$ of the Pareto set $\mathcal{S}_k$ realizes the same first k cost function values as $\overline{\mathbf{x}}$, and let $\mathbf{x}_\alpha = (1-\alpha)\overline{\mathbf{x}} + \alpha\overline{\mathbf{y}}$ $(0 \leq \alpha \leq 1)$ be an element of the line segment $[\overline{\mathbf{x}}, \overline{\mathbf{y}}]$ which lies entirely in the assumed convex admissible domain Ω_{ad}. Then, by convexity of the cost functions :

$$\forall j \leq k, \quad f_j(\mathbf{x}_\alpha) \leq (1-\alpha) f_j(\overline{\mathbf{x}}) + \alpha f_j(\overline{\mathbf{y}}) = \bar{f}_j. \quad \text{(B.7)}$$

However, none of these inequalities can be strict, since otherwise $\mathbf{x}_\alpha$ would dominate $\overline{\mathbf{x}}$ in efficiency in $\mathcal{S}_k$. Hence these inequalities can be replaced by equalities, and we conclude that the first k cost functions are constant over the line segment $[\overline{\mathbf{x}}, \overline{\mathbf{y}}]$. But this eventuality has been excluded by assumption. Therefore the existence of $\overline{\mathbf{y}} \neq \overline{\mathbf{x}}$ is rejected : $\overline{\mathbf{x}}$ is unique.

Now, let $\mathbf{x}$ be an arbitrary element of $\mathcal{S}_k$ and make the tentative hypothesis that $\mathbf{x} \notin \mathcal{S}_{k+1}$. Then, there exists an admissible element $\mathbf{y}$ that dominates $\mathbf{x}$ in efficiency w.r.t. $\{f_1, f_2, \ldots, f_{k+1}\}$, that is :

$$f_1(\mathbf{y}) \leq f_1(\mathbf{x}), \quad f_2(\mathbf{y}) \leq f_2(\mathbf{x}), \ldots, \quad f_{k+1}(\mathbf{y}) \leq f_{k+1}(\mathbf{x}), \quad \text{(B.8)}$$

and at least one of these inequalities is strict. That inequality cannot be one of the first k since otherwise $\mathbf{y}$ would dominate $\mathbf{x}$ in efficiency also w.r.t. $\{f_1, f_2, \ldots, f_k\}$ which is excluded since $\mathbf{x} \in \mathcal{S}_k$. Hence in (B.8) :

$$f_1(\mathbf{y}) = f_1(\mathbf{x}), \quad f_2(\mathbf{y}) = f_2(\mathbf{x}), \quad \ldots,\ f_k(\mathbf{y}) = f_k(\mathbf{x}), \quad \text{(B.9)}$$

and $f_{k+1}(\mathbf{y}) < f_{k+1}(\mathbf{x})$. But, we have shown that a point of the Pareto front of the functions $\{f_1, f_2, \ldots, f_k\}$ is the image of a unique element of $\mathcal{S}_k$, and since $\mathbf{y} \neq \mathbf{x}$, we arrive at a contradiction. The contradiction is waived by rejecting the tentative hypothesis ($\mathbf{x} \notin \mathcal{S}_{k+1}$), and we conclude that instead $\mathbf{x} \in \mathcal{S}_{k+1}$. Since $\mathbf{x}$ is arbitrary in $\mathcal{S}_k$, this conclusion implies that $\mathcal{S}_k \subseteq \mathcal{S}_{k+1}$ and the remaining follows. □

Remarque 2
Excluding situations in which the first k cost functions would be constant over some line segment is not very restrictive. Nevertheless, the following example was fabricated to exhibit a situation in which the exclusion does not hold :

$$\mathbf{x} = (x, y, z) \in \Omega_{ad} = \mathbb{R}^3,\ f_1(\mathbf{x}) = \phi(x, y),\ f_2(\mathbf{x}) = \psi(x, y),\ f_3(\mathbf{x}) = z^2,\ \{\phi, \psi\} \text{ convex.} \tag{B.10}$$

Then in $\mathbb{R}^3$, $\mathcal{S}_2$ is an infinite cylinder invariant by translation parallel to the z axis, while $\mathcal{S}_3$ is limited to its trace in the plane $z = 0$. Of course, $\mathcal{S}_2$ is larger than $\mathcal{S}_3$ only if we consider Ω_{ad} to be $\mathbb{R}^3$, which we would not do in a study limited to the pair $\{f_1, f_2\}$.

Annexe C

Exemples d'orthogonalisation de gradients

C.1 Exemple 1

Dans cet exemple, on a

$$m = 5\,, \quad n = 2\,,$$

et les 5 vecteurs gradients $\{\mathbf{u}_j^0\}$ sont spécifiés comme suit :

```
Specified vectors (U0 matrix) :
   1.0000   1.0000   0.3000  -0.6000  -1.0000
   0.0000   1.4000   1.5000   2.4000   1.0000
```

Ces vecteurs sont représentés à la Figure C.1.

Au retour de la procédure d'orthogonalisation de Gram-Schmidt, on obtient les informations suivantes :

```
Returning from Gram-Schmidt :

    Dimension of orthogonal basis, r =    2
    Number of vectors admitting a known common descent direction, mu =    4
    Defect, m-mu =    1

Permutation of u-vectors =
   2   5   3   4   1

Reordered U matrix :
   1.0000  -1.0000   0.3000  -0.6000   1.0000
   1.4000   1.0000   1.5000   2.4000   0.0000

Orthogonal basis (V matrix) :
   1.0000  -1.3125
   1.4000   0.9375

Coefficient-vector beta =
```

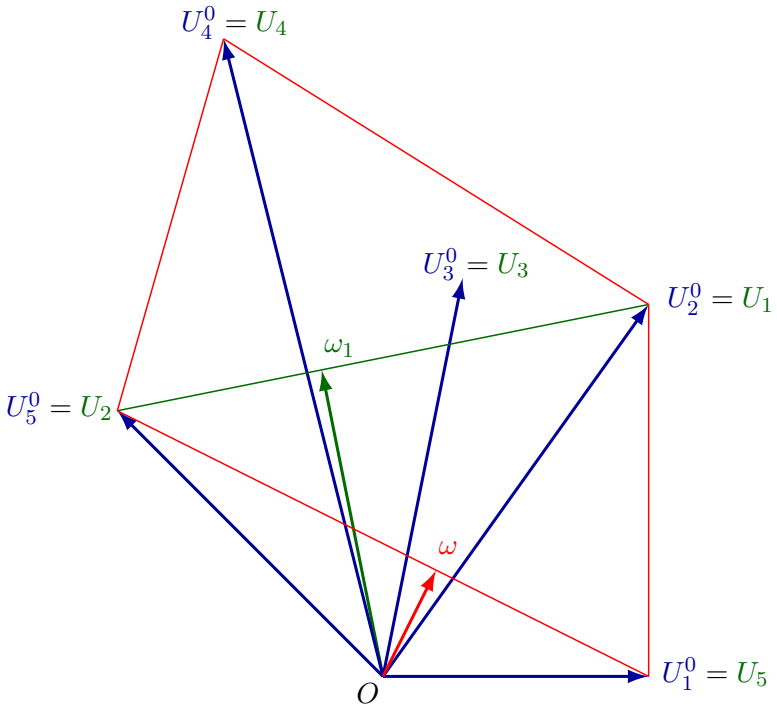


FIGURE C.1 – Illustration de l'exemple 1

```
   0.4678    0.5322

Coefficient-vector alfa (w.r.t. reordered U matrix) =
   0.3846    0.6154

Coefficient-vector alpha (w.r.t. original U matrix) =
   0.0000    0.3846    0.0000    0.0000    0.6154

Provisional direction of search
 omega_1 = V * beta = G * alfa = U * alpha =
  -0.2308
   1.1538

Directional derivatives = dot products of omega_1
 with column-vectors of reordered U matrix :
   1.3846    1.3846    1.6615    2.9077   -0.2308
Constant a =    1.2000
```

La famille est de rang maximal (r = 2). La permutation effectuée sur les vecteurs $\{\mathbf{u}_j^0\}$ est précisée et conduit à la définition de la matrice $\mathbf{U}$ dont les r = 2 premiers vecteurs-colonnes servent à définir la base orthogonale, $\{\mathbf{v}_1, \mathbf{v}_2\}$ (matrice $\mathbf{V}$). La procédure fournit une direction de recherche provisoire, notée 'omega_1' (ω_1) telle que $-\omega_1$ est une direction

de descente pour les $\mu = 4$ premiers vecteurs-colonnes de $\mathbf{U}$. Les coefficients 'beta', 'alfa' et 'alpha' correspondent aux composantes de ω_1 dans les bases formées par les vecteurs-colonnes des matrices $\mathbf{V}$, $\mathbf{U}$ (à l'issue de la procédure) et $\mathbf{U}^0$ (spécifiée initialement et invariante) respectivement. Les dérivées dans la direction de ω_1 sont calculées pour les 5 gradients. Les $\mathsf{r} = 2$ premières sont associées à la base orthogonale et ont la même valeur σ ($\sigma = 1.3846$). Les $\mu - \mathsf{r} = 2$ dérivées suivantes sont supérieures, au moins égales à $\mathsf{a}\sigma$ avec ici $\mathsf{a} = 1.2$; noter que 1.6615 = 1.2 * 1.3846. Le déficit $m - \mu$ correspond au nombre de vecteurs liés à la base orthogonale pour lesquels $c_{\ell,\ell} < 1$. Ici $m - \mu = 1$; en conséquence, la seule dernière dérivée est négative. Le vecteur ω_1 n'est donc pas une solution globale, et pour poursuivre la recherche de ω, il convient de poursuivre l'analyse.

On calcule la matrice $\underline{\mathbf{U}}^t\underline{\mathbf{U}}$, que l'on factorise par la procédure DPOTRF de la bibliothèque LAPACK,

```
 UtU matrix :
   2.9600   0.4000
   0.4000   2.0000

Factorization : upon exit from DPOTRF, info =
   0
```

en préparation de l'inversion du sytème

$$\left(\underline{\mathbf{U}}^t\underline{\mathbf{U}}\right)\mathbf{W} = \underline{\mathbf{U}}^t\mathbf{V}\Delta^{-1}\mathbf{V}^t$$

réalisée par la procédure DPOTRS qui fournit la matrice $\mathbf{W}$:

```
Inversion : upon exit from DPOTRS, info =
   0

Matrix W :
   0.4167   0.4167
  -0.5833   0.4167
```

Enfin on calcule les vecteurs-colonnes $\{\eta_j\}$:

```
Eta-vectors :
   1.0000   0.0000   0.7500   0.7500   0.4167
   0.0000   1.0000   0.4500   1.3500  -0.5833
```

Le premier bloc 2×2 correspond à la matrice identité $\mathbf{I}_2$. Sans surprise, les composantes des vecteurs-colonnes η_3 et η_4 sont toutes positives, de sommes au moins égales à la constante $\mathsf{a} = 1.2$, et aucun traitement n'est rendu nécessaire par ces vecteurs qui peuvent désormais être ignorés.

Examinons maintenant le dernier vecteur, η_5. Sa deuxième composante est négative, ainsi que la somme des deux composantes. Ceci traduit le fait, déjà établi, que $-\omega_1$ n'est pas une direction de descente pour le critère correspondant. Par contre, la première composante est positive, et on en conclut que la famille de départ ne forme pas une configuration de Pareto stationnarité. Il existe donc une solution $\omega \neq 0$.

Ces conclusions apparaissent clairement à la Figure C.1. Les vecteurs spécifiés $\mathbf{u}_j^0$ ($j = 1, \ldots, 5$) sont représentés par les bipoints $\overrightarrow{OU_j^0}$. Par application de (**??**), la procédure élit

$\mathbf{u}_1 = \mathbf{u}_2^0$, représenté par $\overrightarrow{OU_1} = \overrightarrow{OU_2^0}$. Conformément à la permutation, les points U_j suivants sont dans l'ordre U_5^0, U_3^0, U_4^0, et U_1^0. Le vecteur ω_1 est le vecteur de plus petite norme de l'enveloppe convexe de $\{\mathbf{u}_1^0, \mathbf{u}_2^0\} = \{\mathbf{u}_2^0, \mathbf{u}_5^0\}$. Sa représentation est associée à la projection orthogonale de O sur le segment U_1U_2. Clairement, les produits scalaires $(\mathbf{u}_j^0, \omega_1)$ sont tous positifs à l'exception de $(\mathbf{u}_1^0, \omega_1)$. Le vecteur ω_1 n'est donc pas une solution globale, mais celle-ci existe puisque le plus petit polygone convexe englobant les 5 points U_i^0 ne contient pas l'origine O. Le vecteur ω est l'élément de plus petite norme de l'enveloppe convexe de $\{\mathbf{u}_1^0.\mathbf{u}_2^0, \mathbf{u}_5^0\}$. Il s'obtient par projection orthogonale du point O sur le segment $U_2U_5 = U_5^0U_1^0$.

Remarque 3
*On voit que la procédure donnerait directement ω si on choisissait $(\mathbf{u}_1, \mathbf{u}_2) = (\mathbf{u}_1^0, \mathbf{u}_5^0)$, ce qui serait le cas si on normalisait préalablement les vecteurs $\{\mathbf{u}_j^0\}$. Dans ce cas, (**??**) équivaudrait à choisir pour deux premiers éléments, le couple faisant l'angle maximum. On ne recommande pas de faire cela en général car la normalisation des gradients, équivalente à la mise des critères à une certaine échelle, nous parait relever d'un choix certes respectable, mais subjectif du praticien.*

Noter que la droite de support ω a pour pente 2. Retrouvons ce résultat par le calcul. À l'issue de la procédure d'orthogonalisation, le problème est reformulé dans la base $\{\mathbf{u}_1^0, \mathbf{u}_2^0\}$. Dans cette base, la solution est associée à l'élément de plus petite norme des seuls 3 vecteurs :

$$\eta_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \eta_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad \eta_5 = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$$

où $c_1 = 0.4167$, et $c_2 = -0.5833$. On cherche donc la combinaison convexe de coefficients (x_1, x_2, x_5) tels que

$$\tilde{\omega} = x_1\eta_1 + x_2\eta_2 + x_5\eta_5 = \begin{pmatrix} x_1 + c_1x_5 \\ x_2 + c_2x_5 \end{pmatrix}$$

soit de norme euclidienne minimale. On minimise donc

$$q(\mathbf{x}) = \tfrac{1}{2}(x_1 + c_1x_5)^2 + \tfrac{1}{2}(x_2 + c_2x_5)^2 = \tfrac{1}{2}\mathbf{x}^t\mathbf{H}\mathbf{x}$$

où

$$\mathbf{H} = \nabla^2 q(\mathbf{x}) = \begin{pmatrix} 1 & 0 & c_1 \\ 0 & 1 & c_2 \\ c_1 & c_2 & c_1^2 + c_2^2 \end{pmatrix}$$

sous les contraintes habituelles ($0 \leq x_i \leq 1, \forall i;\ \sum_i x_i = 1$). La minimisation peut s'effectuer par la procédure QUADPROG de MATLAB suivant la séquence suivante :

```
c1 = 0.4167;
c2 = -0.5833;
H = [1 0 c1; 0 1 c2; c1 c2 (c1^2+c2^2)];
f = [0 0 0]';
Aeq = [1 1 1];
beq = 1;
A = [];
b = [];
lb = [0 0 0]';
```

```
ub = [1 1 1]';
x0 = [0.2 0.3 0.5];
x = quadprog(H,f,A,b,Aeq,beq,lb,ub,x0)
```

On obtient le résultat suivant :

$$x_1 = 0 \qquad x_2 = 0.4093 \qquad x_5 = 0.5907$$

c'est-à-dire le vecteur

$$\tilde{\omega} = x_2\eta_2 + x_5\eta_5 \doteq \begin{pmatrix} 0.2461 \\ 0.0647 \end{pmatrix},$$

représenté dans la base canonique de départ par le vecteur-colonne suivant :

$$\omega^\star = x_2\mathbf{u}_2 + x_5\mathbf{u}_5\,.$$

Attention ! $\omega^\star \neq \omega$, car nous avons changé de métrique. La direction de descente, prédemment notée ω est la suivante :

$$\mathbf{d} = \mathbf{W}^t\mathbf{W}\omega^* = \mathbf{W}^t\tilde{\omega}$$

Il vient :

$$\mathbf{d} \doteq \begin{pmatrix} 0.4167 & -0.5833 \\ 0.4167 & 0.4167 \end{pmatrix} \begin{pmatrix} 0.2461 \\ 0.0647 \end{pmatrix} \doteq \begin{pmatrix} 0.0648 \\ 0.1295 \end{pmatrix}.$$

Aux erreurs d'arrondis près, la pente de ce vecteur est bien égale à 2 comme on l'avait précédemment obtenu pour le vecteur ω de la Figure C.1.

C.2 Exemple 2

On considère à nouveau le cas de $m = 5$ vecteurs de dimension $n = 2$. Ici la spécification est la suivante :

```
Specified vectors (U0 matrix) :
   1.0000   1.0000   0.2400  -0.4500  -1.0000
   0.0000   1.4000   1.2000   1.8000  -0.4000
```

Ces vecteurs sont repésentés à la Figure C.2. On voit immédiatement que le polygone convexe englobant les points $U_1^0, \ldots, U_5^0$ contient l'origine O. On est donc dans une configuration de Pareto-stationnarité pour laquelle il n'existe aucune solution globale. Néanmoins il est intéressant d'analyser les sorties de la procédure numérique :

```
Returning from Gram-Schmidt :

    Dimension of orthogonal basis, r =    2
    Number of vectors admitting a known common descent direction, mu =    4
    Defect, m-mu =    1

Permutation of u-vectors =
   4   1   3   2   5

Reordered U matrix :
```

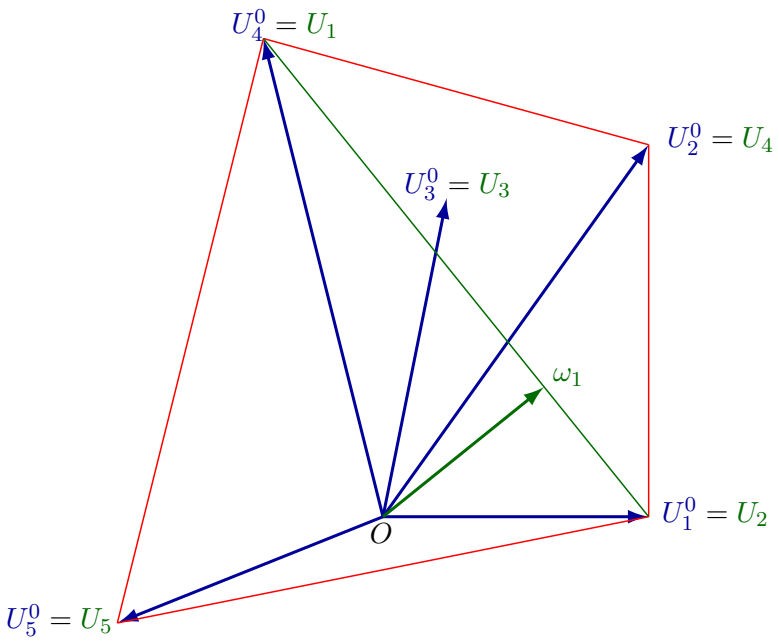


FIGURE C.2 – Illustration de l'exemple 2

```
  -0.4500    1.0000    0.2400    1.0000   -1.0000
   1.8000    0.0000    1.2000    1.4000   -0.4000

Orthogonal basis (V matrix) :
  -0.4500    0.8324
   1.8000    0.2081

Coefficient-vector beta =
   0.1762    0.8238

Coefficient-vector alfa (w.r.t. reordered U matrix) =
   0.2714    0.7286

Coefficient-vector alpha (w.r.t. original U matrix) =
   0.7286    0.0000    0.0000    0.2714    0.0000

Provisional direction of search
 omega = V * beta = G * alfa = U * alpha =
   0.6065
   0.4885

Directional derivatives = dot products of omega
```

```
 with column-vectors of reordered U matrix :
   0.6065   0.6065   0.7318   1.2904  -0.8019
Constant a =   1.2067
```

À nouveau, r = 2, $\mu = 4$. La base orthogonale est ici construite à partir des vecteurs $\{\mathbf{u}_1, \mathbf{u}_2\} = \{\mathbf{u}_4^0, \mathbf{u}_1^0\}$ conformément à la permutation indiquée. Le vecteur ω_1 est donc obtenu par projection orthogonale de O sur le segment $U_4^0 U_1^0$. Les dérivées directionnelles sont calculées : 4 sont positives (dont les 2 premières égales à $\sigma = 0.6065$, la 3e égale à $\mathsf{a}\sigma$, où $\mathsf{a} = 1.2067$, et la 4e supérieure encore), et la 5e est négative, -0.8019, associée au vecteur $\mathbf{u}_5 = \mathbf{u}_5^0$.

On passe ensuite au calcul des matrices $\underline{\mathbf{U}}^t\underline{\mathbf{U}}$ et $\mathbf{W}$ et des vecteurs $\eta_j = \mathbf{W}\mathbf{u}_j$.

```
 UtU matrix :
   3.4425  -0.4500
  -0.4500   1.0000

Factorization : upon exit from DPOTRF, info =
   0

Inversion : upon exit from DPOTRS, info =
   0

Matrix W :
  -0.0000   0.5556
   1.0000   0.2500

Eta-vectors :
   1.0000  -0.0000   0.6667   0.7778  -0.2222
  -0.0000   1.0000   0.5400   1.3500  -1.1000
```

La nouveauté ici est que les deux composantes du vecteur η_5 sont négatives, ce qui confirme que la configuration des gradients correspond à une situation de Pareto-stationnarité. Il n'y a donc pas de solution globale pour ω.

C.3 Exemple 3

Dans cet exemple, on a

$$m = 5\,, \quad n = 8\,,$$

et les 5 vecteurs gradients $\{\mathbf{u}_j^0\}$ sont spécifiés comme suit :

```
Specified vectors (U0 matrix) :
  -1.0000   0.3586   0.3423   0.6923  -0.4751
  -0.7369   0.8694  -0.9846   0.0539  -0.9051
   0.5112  -0.2330  -0.2332  -0.8161   0.4722
  -0.0827   0.0388  -0.8663   0.3078  -0.3435
   0.0655   0.6619  -0.1650  -0.1680   0.2653
  -0.5621  -0.9309   0.3735   0.4024   0.5128
  -0.9059  -0.8931   0.1780   0.8206   0.9821
   0.3577   0.0594   0.8609   0.5244  -0.2693
```

Ces données ont été obtenues par tirages aléatoires dans l'intervalle [-1,1].

La procédure numérique fournit les informations suivantes :

```
Returning from Gram-Schmidt :

    Dimension of orthogonal basis, r =    5
    Number of vectors admitting a known common descent direction, mu =    5
    Defect, m-mu =    0

Permutation of u-vectors =
   3   2   5   4   1

Reordered U matrix :
   0.3423   0.3586  -0.4751   0.6923  -1.0000
  -0.9846   0.8694  -0.9051   0.0539  -0.7369
  -0.2332  -0.2330   0.4722  -0.8161   0.5112
  -0.8663   0.0388  -0.3435   0.3078  -0.0827
  -0.1650   0.6619   0.2653  -0.1680   0.0655
   0.3735  -0.9309   0.5128   0.4024  -0.5621
   0.1780  -0.8931   0.9821   0.8206  -0.9059
   0.8609   0.0594  -0.2693   0.5244   0.3577

Orthogonal basis (V matrix) :
   0.3423   0.3535  -0.1274   0.2952  -0.2560
  -0.9846   0.2930  -0.1408   0.1568  -0.1773
  -0.2332  -0.2330   0.1783  -0.3454  -0.0775
  -0.8663  -0.2426  -0.1721   0.2106   0.2774
  -0.1650   0.4048   0.4416   0.2030   0.0556
   0.3735  -0.5253  -0.1062   0.0543  -0.1618
   0.1780  -0.5601   0.1863   0.4709  -0.0172
   0.8609   0.3086  -0.1430   0.0984   0.2416

Coefficient-vector beta =
   0.0370   0.0894   0.2920   0.1903   0.3913

Coefficient-vector alfa (w.r.t. reordered U matrix) =
  -0.0556   0.2679   0.2729   0.3134   0.2013

Coefficient-vector alpha (w.r.t. original U matrix) =
   0.2013   0.2679  -0.0556   0.3134   0.2729

Provisional direction of search
 omega_1 = V * beta = G * alfa = U * alpha =
  -0.0370
  -0.0909
  -0.0735
   0.0446
```

```
   0.2194
  -0.1172
   0.0938
   0.1310

Directional derivatives = dot products of omega_1
 with column-vectors of reordered U matrix :
   0.1048   0.1048   0.1048   0.1048   0.1048
```

La procédure établit que ces 5 vecteurs sont linéairement indépendants (famille de rang maximal r = 5 ; $\mu = m = 5$; défaut $m - \mu = 0$). La direction de recherche ω_1 est donc une solution globale, ce qui est confirmé par les 5 dérivées directionnelles égales à $\sigma = +0.1048$.

On peut retrouver la même direction de recherche en effectuant le changement de base et de métrique. la procédure fournit les informations suivantes :

```
 UtU matrix :
   2.8310  -1.2773   1.0067   0.8831   0.2617
  -1.2773   3.0460  -2.2754  -0.6903   0.2753
   1.0067  -2.2754   2.7562  -0.0423   0.1550
   0.8831  -0.6903  -0.0423   2.3815  -1.9677
   0.2617   0.2753   0.1550  -1.9677   3.0801

Factorization : upon exit from DPOTRF, info =
   0

Inversion : upon exit from DPOTRS, info =
   0

Matrix W :
   0.3522  -0.1598   0.0090  -0.6538  -0.0697   0.1170  -0.2077   0.1621
   0.1476   0.0425  -0.1237  -0.1704   0.8872  -0.4469   0.1373   0.1206
  -0.1547  -0.2187   0.0665  -0.0448   0.8530  -0.2122   0.5542  -0.0746
  -0.2061  -0.1910  -0.5037   0.7617   0.3158  -0.2645   0.4439   0.5771
  -0.4916  -0.3405  -0.1488   0.5328   0.1067  -0.3108  -0.0330   0.4640

Eta-vectors :
   1.0000   0.0000   0.0000   0.0000  -0.0000
   0.0000   1.0000   0.0000   0.0000  -0.0000
   0.0000   0.0000   1.0000  -0.0000  -0.0000
   0.0000   0.0000   0.0000   1.0000   0.0000
   0.0000  -0.0000   0.0000  -0.0000   1.0000

Bissector omega_0 of g-vectors =
  -0.0164
  -0.3407
  -0.0598
  -0.1892
   0.1319
```

```
  -0.0408
   0.0363
   0.3066

Corresponding direction of search, d_0 :
  -0.0705
  -0.1735
  -0.1401
   0.0851
   0.4186
  -0.2235
   0.1789
   0.2498

Directional derivatives = dot products of d_0
 with column-vectors of reordered U matrix :
   0.2000   0.2000   0.2000   0.2000   0.2000
```

Comme dans l'exemple 1, on a calculé les matrices $\underline{\mathbf{U}}^t\underline{\mathbf{U}}$ et $\mathbf{W}$. Ici, la direction de descente $\mathbf{d}_0$ est associée à la moyenne des vecteurs gradients

$$\omega_0 = \frac{1}{m}\sum_{j=1}^{m}\mathbf{u}_j$$

et à une nouvelle métrique :

$$\mathbf{d}_0 = \mathbf{W}^t\mathbf{W}\omega_0$$

Le calcul du vecteur $\mathbf{d}_0$ confirme que ce vecteur est proportionnel à ω_1 (voir ci-dessus). Les dérivées directionnelles toutes égales à $\frac{1}{m} = 0.2$.

C.4 Exemple 4

On a défini $m = 8$ vecteurs $\{\mathbf{u}_j^0\}$ de dimension $n = 5$ en assignant des composantes aléatoires dans l'intervalle [-1,1], correspondant à la spécification suivante :

```
Specified vectors (U0 matrix)  :
  -1.0000  -0.5621  -0.2330   0.0594  -0.1650   0.0539   0.8206  -0.3435
  -0.7369  -0.9059   0.0388   0.3423   0.3735  -0.8161   0.5244   0.2653
   0.5112   0.3577   0.6619  -0.9846   0.1780   0.3078  -0.4751   0.5128
  -0.0827   0.3586  -0.9309  -0.2332   0.8609  -0.1680  -0.9051   0.9821
   0.0655   0.8694  -0.8931  -0.8663   0.6923   0.4024   0.4722  -0.2693
```

La procédure numérique fournit les informations suivantes :

```
Returning from Gram-Schmidt :

    Dimension of orthogonal basis, r =    5
    Number of vectors admitting a known common descent direction, mu =    7
```

```
    Defect, m-mu =    1

Permutation of u-vectors =
   3   5   4   6   7   1   2   8

Reordered U matrix :
  -0.2330  -0.1650   0.0594   0.0539   0.8206  -1.0000  -0.5621  -0.3435
   0.0388   0.3735   0.3423  -0.8161   0.5244  -0.7369  -0.9059   0.2653
   0.6619   0.1780  -0.9846   0.3078  -0.4751   0.5112   0.3577   0.5128
  -0.9309   0.8609  -0.2332  -0.1680  -0.9051  -0.0827   0.3586   0.9821
  -0.8931   0.6923  -0.8663   0.4024   0.4722   0.0655   0.8694  -0.2693

Orthogonal basis (V matrix) :
  -0.2330  -0.1899  -0.0807  -0.0525  -0.0159
   0.0388   0.2508   0.3040  -0.0382   0.0076
   0.6619   0.3554  -0.2368  -0.0080  -0.0088
  -0.9309   0.2041   0.0935   0.0203  -0.0180
  -0.8931   0.1112  -0.2386  -0.0151   0.0167

Coefficient-vector beta =
   0.0004   0.0029   0.0037   0.1665   0.8265

Coefficient-vector alfa (w.r.t. reordered U matrix) =
   0.1500   0.3089   0.2287   0.2697   0.0426

Coefficient-vector alpha (w.r.t. original U matrix) =
   0.0000   0.0000   0.1500   0.2287   0.3089   0.2697   0.0426   0.0000

Provisional direction of search
 omega_1 = V * beta = G * alfa = U * alpha =
  -0.0229
   0.0018
  -0.0082
  -0.0109
   0.0104

Directional derivatives = dot products of omega_1
 with column-vectors of reordered U matrix :
   0.0008   0.0008   0.0008   0.0008   0.0008   0.0189   0.0135  -0.0094
Constant a =  16.4348
```

La famille est de rang maximal $\mathtt{r} = 5$. La permutation (3,5,...,8) indique l'ordre dans lequel les vecteurs spécifiés (matrice $\mathbf{U}^0$) ont été replacés pour former la matrice $\mathbf{U}$. La base orthogonale permet de calculer une direction de recherche provisoire ω_1. Pour ces 5 gradients, la dérivée dans la direction de ω_1 a la même valeur $\sigma = 0.0008$, très faible, révélatrice d'une configuration de ces vecteurs proche d'une situation de Pareto-stationnarité. Les dérivées des $\mu = 7$ premiers vecteurs-colonnes de la matrice $\mathbf{U}$ dans la direction de ω_1 sont positives. La

7e valeur de dérivée, 0.0135, est dans le rapport $\mathsf{a} = 16.4348$ avec σ. La 6e est supérieure. La seule 8e et dernière valeur est négative, ce qui correspond à un défaut de $m - \mu = 1$ vecteur.

On calcule ensuite les matrices $\underline{\mathbf{U}}^t\underline{\mathbf{U}}$ et $\mathbf{W}$, et les vecteurs $\eta_j = \mathbf{W}\mathbf{u}_j$:

```
 UtU matrix :
   2.1580  -1.2489   0.3384  -0.0434  -0.0645
  -1.2489   1.4189  -0.8577  -0.1250  -0.4763
   0.3384  -0.8577   1.8950  -0.8887   0.4980
  -0.0434  -0.1250  -0.8887   0.9538  -0.1880
  -0.0645  -0.4763   0.4980  -0.1880   2.2163

Factorization : upon exit from DPOTRF, info =
   0

Inversion : upon exit from DPOTRS, info =
   0

Matrix W :
  -4.3588   0.6167  -0.9699  -2.2268   1.6465
  -8.7919   1.1151  -2.8564  -3.6980   4.0795
  -6.4555   0.3346  -2.8526  -2.8364   2.5408
  -7.4724  -0.2919  -2.8436  -3.5892   3.5702
  -0.8293   0.3943  -0.4567  -0.9350   0.8696

Eta-vectors :
   1.0000  -0.0000  -0.0000   0.0000   0.0000   3.7005   2.1773  -1.4667
  -0.0000   1.0000  -0.0000   0.0000   0.0000   7.0829   5.1304  -2.8792
  -0.0000  -0.0000   1.0000   0.0000   0.0000   5.1515   3.4967  -2.6264
  -0.0000  -0.0000  -0.0000   1.0000   0.0000   6.7645   5.2642  -3.4552
  -0.0000  -0.0000  -0.0000   0.0000   1.0000   0.4396   0.3664  -0.9972
```

Les $\mathsf{r} = 5$ premiers vecteurs-colonnes η_j forment, aux erreurs d'arrondis près, la matrice identité $\mathbf{I}_5$. Les composantes du vecteur η_7 sont toutes positives et de somme égale à $\mathsf{a} \doteq 16.435$. Les composantes du vecteur η_6 sont toutes positives et de somme supérieure.

Cependant, la recherche d'un $\omega \neq 0$ échoue car non seulement certaines composantes du vecteur η_8 sont négatives, mais toutes le sont, ce qui correspond à une configuration de gradients associée à une situation de Pareto-stationnarité ($\omega = 0$).

Cette dernière conclusion, négative, n'est pas surprenante dans le cas considéré où les composantes des vecteurs gradients ont été définies par tirage aléatoire. Lorsque les vecteurs gradients sont en surnombre par rapport à la dimension d'espace, $\omega \neq 0$ n'est possible que si les gradients diffèrent peu les uns des autres parce qu'ils sont les réalisations d'une même fonction vectorielle régulière, par exemple un gradient de forme, lorsqu'un paramètre, le temps ou un paramètre global du problème, évolue continûment et progressivement.

C.5 Exemple 5

Cet exemple est similaire au précédent, sauf que l'on considère un plus grand nombre de vecteurs :

$$m = 15\,, \quad n = 5\,.$$

À nouveau, on utilise le tirage aléatoire pour la spécification des vecteurs $\{\mathbf{u}_j^0\}$:

```
Specified vectors (U0 matrix) :
  -1.0000  -0.5621  -0.2330   0.0594  -0.1650   0.0539   0.8206  -0.3435
  -0.7369  -0.9059   0.0388   0.3423   0.3735  -0.8161   0.5244   0.2653
   0.5112   0.3577   0.6619  -0.9846   0.1780   0.3078  -0.4751   0.5128
  -0.0827   0.3586  -0.9309  -0.2332   0.8609  -0.1680  -0.9051   0.9821
   0.0655   0.8694  -0.8931  -0.8663   0.6923   0.4024   0.4722  -0.2693

  -0.5059  -0.8546   0.5330  -0.6670   0.8093  -0.0120   0.0014
   0.9651   0.2633  -0.0445  -0.0270   0.0090  -0.4677  -0.2317
   0.4453   0.7694  -0.5245   0.7953   0.0326  -0.8185  -0.4458
   0.5067  -0.4546  -0.4502   0.8184  -0.3619   0.8955   0.8276
   0.3030  -0.1272  -0.2815  -0.8789   0.9733  -0.8525   0.0595
```

Les informations à l'issue de la procédure sont les suivantes :

```
Returning from Gram-Schmidt :

    Dimension of orthogonal basis, r =    5
    Number of vectors admitting a known common descent direction, mu =   10
    Defect, m-mu =    5

Permutation of u-vectors =
  14  13  10   4   9   2  15   1
   5   6   7  11   8  12   3

Reordered U matrix :
  -0.0120   0.8093  -0.8546   0.0594  -0.5059  -0.5621   0.0014  -1.0000
  -0.4677   0.0090   0.2633   0.3423   0.9651  -0.9059  -0.2317  -0.7369
  -0.8185   0.0326   0.7694  -0.9846   0.4453   0.3577  -0.4458   0.5112
   0.8955  -0.3619  -0.4546  -0.2332   0.5067   0.3586   0.8276  -0.0827
  -0.8525   0.9733  -0.1272  -0.8663   0.3030   0.8694   0.0595   0.0655

  -0.1650   0.0539   0.8206   0.5330  -0.3435  -0.6670  -0.2330
   0.3735  -0.8161   0.5244  -0.0445   0.2653  -0.0270   0.0388
   0.1780   0.3078  -0.4751  -0.5245   0.5128   0.7953   0.6619
   0.8609  -0.1680  -0.9051  -0.4502   0.9821   0.8184  -0.9309
   0.6923   0.4024   0.4722  -0.2815  -0.2693  -0.8789  -0.8931

Orthogonal basis (V matrix) :
  -0.0120   0.5377  -0.0209  -0.0070  -0.0147
  -0.4677  -0.1486  -0.0545  -0.0128   0.0074
```

```
  -0.8185  -0.2489   0.0161  -0.0308  -0.0128
   0.8955   0.0539   0.0038  -0.0438   0.0061
  -0.8525   0.3695   0.0188  -0.0093   0.0148

Coefficient-vector beta =
   0.0002   0.0010   0.1230   0.1565   0.7194

Coefficient-vector alfa (w.r.t. reordered U matrix) =
   0.2839   0.3311   0.3107   0.0441   0.0303

Coefficient-vector alpha (w.r.t. original U matrix) =
   0.0000   0.0000   0.0000   0.0441   0.0000   0.0000   0.0000   0.0000
   0.0303   0.3107   0.0000   0.0000   0.3311   0.2839   0.0000

Provisional direction of search
 omega_1 = V * beta = G * alfa = U * alpha =
  -0.0137
  -0.0036
  -0.0125
  -0.0018
   0.0117

Directional derivatives = dot products of omega_1
 with column-vectors of reordered U matrix :
   0.0005   0.0005   0.0005   0.0005   0.0005   0.0161   0.0056   0.0109
   0.0052   0.0034  -0.0001  -0.0031  -0.0076  -0.0124  -0.0140
Constant a =   6.8714

 UtU matrix :
   2.4176  -1.1945  -1.0413   1.1748  -0.6144
  -1.1945   1.7344  -0.6235  -0.7397  -0.2747
  -1.0413  -0.6235   1.6145  -0.5020   0.7602
   1.1748  -0.7397  -0.5020   1.8950  -0.5188
  -0.6144  -0.2747   0.7602  -0.5188   1.7343

Factorization : upon exit from DPOTRF, info =
   0

Inversion : upon exit from DPOTRS, info =
   0

Matrix W :
  -7.8733  -2.6467  -7.0742  -0.5518   6.6030
  -8.7901  -2.6384  -8.2994  -1.3284   8.1449
  -8.9311  -2.7133  -7.4047  -1.5104   7.1378
  -1.1300   0.1957  -1.5490  -0.5985   0.7671
  -0.8961   0.4490  -0.7812   0.3694   0.9045
```

```
Eta-vectors :
   1.0000   0.0000  -0.0000   0.0000  -0.0000   9.8352   3.6923   6.6855
   0.0000   1.0000  -0.0000   0.0000  -0.0000  10.9667   3.6842   7.1351
   0.0000   0.0000   1.0000   0.0000  -0.0000  10.4931   3.0919   7.7378
   0.0000   0.0000  -0.0000   1.0000  -0.0000   0.3560   0.1939   0.2937
   0.0000   0.0000  -0.0000   0.0000   1.0000   0.7362   0.6025   0.1946

   3.1483   2.3078  -0.8710  -1.9786  -3.9454  -6.5583  -8.3344
   3.4836   2.6254   0.3940  -1.9094  -5.4345  -8.9122  -9.5855
   2.7841   2.5797  -0.4969  -2.0850  -4.8547  -7.3684  -7.8945
  -0.0002  -0.2882   0.8151   0.2549  -1.1486  -1.6475  -0.8824
   1.1207  -0.3533  -0.0360  -0.5087   0.1455  -0.5284  -1.4425
```

La famille de vecteurs-gradients est donc de rang maximal ($\mathsf{r} = n = 5$). On dispose d'une direction de recherche provisoire ω_1 convenable pour les $\mu = 10$ premiers vecteurs de la liste réordonnée.

La nouveauté apparaît à l'examen des vecteurs $\{\eta_j\}$, et plus précisément, des 5 vecteurs d'indice j intermédiaire : $\mathsf{r} + 1 = 6 \leq j \leq \mu = 10 < m = 15$. Les composantes des 3 premiers d'entre eux sont toutes positives, ce qui prouve que ces vecteurs appartiennent bien au secteur conique s'appuyant sur l'enveloppe convexe de la base $\{\mathbf{u}_1, \ldots, \mathbf{u}_5\}$. Ces vecteurs pourraient être définitivement ignorés dans la suite de l'analyse puisqu'il seraient automatiquement pris en compte par la base. Par contre les 2 vecteurs suivants (indices 8 et 9) ont chacun au moins une composante négative, et devraient être retenus. En fait, l'analyse ne sera pas poursuivie car les 2 derniers vecteurs de la liste complète (indices 14 et 15) ont seulement des composantes négatives. On est donc dans une configuration de gradients de Pareto-stationnarité ($\omega = 0$).

C.6 Exemple 6

Cet exemple vise à simuler un cas d'optimisation multipoint dans lequel l'ensemble des vecteurs gradients provient de la discrétisation en temps d'une fonction vectorielle périodique. À cette fin, on considère la fonction suivante :

$$\phi(t) = a_\phi + b_\phi \cos(2\pi t)$$

La constante a_ϕ représente la moyenne de la fonction sur une période, et b_ϕ l'amplitude de la variation autour de cette moyenne. On définit m vecteurs de $\mathbb{R}^n$ comme suit :

$$\mathbf{u}_j^0 = \begin{pmatrix} \phi(t_j) \\ \phi(t_j - \frac{1}{n}) \\ \vdots \\ \phi(t_j - \frac{n-1}{n}) \end{pmatrix}$$

où $t_j = \frac{j-1}{m}$ $(j = 1, \ldots, m)$.

Considérons tout d'abord deux cas extrêmes :

— Si $a_\phi = 0$, et si la discrétisation est assez fine, tout vecteur de cette famille peut être associé à un autre égal, ou quasiment égal à son opposé, et ceci correspond certainement à une configuration de gradients de type Pareto-stationnarité.

— À l'inverse si $b_\phi = 0$, tous les vecteurs sont égaux et la solution est triviale.
On exclut désormais ces deux cas, et on pose :

$$a_\phi = 1\,.$$

On teste d'abord un exemple simple correspondant à

$$m = 15\,, \quad n = 5\,, \quad b_\phi = 0.1\,,$$

pour lequel on sait que la solution existe puisque toutes les composantes de tous les vecteurs de la famille sont positives (par exemple $\omega = (1, 1, \ldots, 1)^t$). Paradoxalement, à l'issue de la procédure numérique, on obtient notamment les résultats surprenants suivants :

```
Returning from Gram-Schmidt :

    Parameter r (lower bound on rank) =    6
    Number of vectors admitting a known common descent direction, mu =   15
    Defect, m-mu =    0

Permutation of u-vectors =
   4  11  15   1   8   7   6   5
   9  10   2  12  13  14   3

Reordered U matrix :
   1.0309   0.9500   1.0914   1.1000   0.9022   0.9191   0.9500   0.9895
   1.1000   0.9022   0.9895   1.0309   0.9895   1.0309   1.0669   1.0914
   1.0309   0.9895   0.9022   0.9191   1.0914   1.1000   1.0914   1.0669
   0.9191   1.0914   0.9500   0.9191   1.0669   1.0309   0.9895   0.9500
   0.9191   1.0669   1.0669   1.0309   0.9500   0.9191   0.9022   0.9022

   0.9022   0.9191   1.0914   0.9895   1.0309   1.0669   1.0669
   0.9500   0.9191   1.0669   0.9022   0.9191   0.9500   1.0914
   1.0669   1.0309   0.9500   0.9500   0.9191   0.9022   0.9895
   1.0914   1.1000   0.9022   1.0669   1.0309   0.9895   0.9022
   0.9895   1.0309   0.9895   1.0914   1.1000   1.0914   0.9500


Orthogonal basis (V matrix) :
   1.0309  -7.1895 175.7936  28.2500  28.2500  28.2500
   1.1000 -19.0000 -19.0000  -1.7188  -1.7188  -1.7188
   1.0309  -3.1711-186.1543 -26.1250 -26.1250 -26.1250
   0.9191  18.4221 -94.6677 -12.8750 -12.8750 -12.8750
   0.9191  15.9386 129.0284  19.3750  19.3750  19.3750
```

Bien évidemment, ces résulats sont faux car le rang est au plus égal à 5. En outre, on constate que les 3 premiers vecteurs-colonnes de la matrice $\mathbf{V}$ sont bien orthogonaux entre eux, mais le 4e est loin de l'être. De plus il est anormalement construit 3 fois de suite.

Ces anomalies sont dues au fait qu'aucune tolérance n'a été autorisée dans le test "$c_{\ell,\ell} \geq 1$?". Dans ce cas précis, dans la construction du 4e vecteur orthogonal, la valeur de $c_{\ell,\ell}$ est

inférieure à 1, et très proche de la limite. Raisonnablement, on doit alors considérer que la solution provisoire est acceptable pour tous les vecteurs, et c'est bien la conclusion que tire la procédure, malgré les erreurs visibles ; mais au lieu d'interrompre le processus, on le prolonge en accumulant les erreurs d'arrondis. En conséquence, la normalisation de $\mathbf{v}_4$ s'effectue avec une énorme erreur d'arrondi qui se propage ensuite et invalide rapidement tout ce qui suit.

On remédie à ces problèmes, en relaxant le test comme suit :

$$"c_{\ell,\ell} \geq 1 - TOL?"$$

où "TOL" est une tolérance. Avec $TOL = 10^{-2}$, l'orthogonalisation est interrompue immédiatement après la définition du premier vecteur. On obtient les informations suivantes :

```
Returning from Gram-Schmidt :

    Parameter r (lower bound on rank) =    1
    Number of vectors admitting a known common descent direction, mu =   15
    Defect, m-mu =    0

Permutation of u-vectors =
   4   2   3   1   5   6   7   8
   9  10  11  12  13  14  15

Reordered U matrix :
   1.0309   1.0914   1.0669   1.1000   0.9895   0.9500   0.9191   0.9022
   1.1000   1.0669   1.0914   1.0309   1.0914   1.0669   1.0309   0.9895
   1.0309   0.9500   0.9895   0.9191   1.0669   1.0914   1.1000   1.0914
   0.9191   0.9022   0.9022   0.9191   0.9500   0.9895   1.0309   1.0669
   0.9191   0.9895   0.9500   1.0309   0.9022   0.9022   0.9191   0.9500

   0.9022   0.9191   0.9500   0.9895   1.0309   1.0669   1.0914
   0.9500   0.9191   0.9022   0.9022   0.9191   0.9500   0.9895
   1.0669   1.0309   0.9895   0.9500   0.9191   0.9022   0.9022
   1.0914   1.1000   1.0914   1.0669   1.0309   0.9895   0.9500
   0.9895   1.0309   1.0669   1.0914   1.1000   1.0914   1.0669


Orthogonal basis (V matrix) :
   1.0309
   1.1000
   1.0309
   0.9191
   0.9191


Coefficient-vector beta =
   1.0000

Coefficient-vector alfa (w.r.t. reordered U matrix) =
```

```
   1.0000

Coefficient-vector alpha (w.r.t. original U matrix) =
   0.0000   0.0000   0.0000   1.0000   0.0000   0.0000   0.0000   0.0000
   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000

Provisional direction of search
 omega_1 = V * beta = G * alfa = U * alpha =
   1.0309
   1.1000
   1.0309
   0.9191
   0.9191

Directional derivatives = dot products of omega_1
 with column-vectors of reordered U matrix :
   5.0250   5.0167   5.0228   5.0077   5.0228   5.0167   5.0077   4.9974
   4.9875   4.9798   4.9755   4.9755   4.9798   4.9875   4.9974
Constant a =   0.9902

 UtU matrix :
   5.0250

Factorization : upon exit from DPOTRF, info =
   0

Inversion : upon exit from DPOTRS, info =
   0

Matrix W :
   0.2052   0.2189   0.2052   0.1829   0.1829

Eta-vectors :
   1.0000   0.9984   0.9996   0.9966   0.9996   0.9984   0.9966   0.9945

   0.9925   0.9910   0.9902   0.9902   0.9910   0.9925   0.9945
```

Le paramètre r = 1, de sorte que le rang, compris entre 1 et 5, n'est pas identifié précisément ; $\mu = m = 15$, ce qui signifie que le premier vecteur choisi ($\mathbf{u}_1 = \mathbf{u}_4^0$, d'après la permutation) est une direction satisfaisante pour la totalité des gradients. Plusieurs dimensions matricielles se réduisent à l'unité, ainsi que la dimension des vecteurs η_j, scalaires tous positifs. Les dérivées directionnelles correspondantes sont toutes positives et proches de 5.

Recommençons l'expérience avec cette fois-ci $b_\phi = 2$ afin que certaines composantes des gradients soient négatives :

```
Specified vectors (U0 matrix) :
   3.0000   2.8271   2.3383   1.6180   0.7909   0.0000  -0.6180  -0.9563
   1.6180   2.3383   2.8271   3.0000   2.8271   2.3383   1.6180   0.7909
```

```
  -0.6180  -0.0000   0.7909   1.6180   2.3383   2.8271   3.0000   2.8271
  -0.6180  -0.9563  -0.9563  -0.6180  -0.0000   0.7909   1.6180   2.3383
   1.6180   0.7909   0.0000  -0.6180  -0.9563  -0.9563  -0.6180  -0.0000

  -0.9563  -0.6180  -0.0000   0.7909   1.6180   2.3383   2.8271
   0.0000  -0.6180  -0.9563  -0.9563  -0.6180   0.0000   0.7909
   2.3383   1.6180   0.7909   0.0000  -0.6180  -0.9563  -0.9563
   2.8271   3.0000   2.8271   2.3383   1.6180   0.7909   0.0000
   0.7909   1.6180   2.3383   2.8271   3.0000   2.8271   2.3383

 Parameter r_max (upper bound on rank) =            5
```

Avec $TOL = 0$, comme précédemment, on obtient des résultats, omis ici, partiellement incohérents en raison de l'accumulation des erreurs d'arrondis. Fixant à nouveau le paramètre TOL à 10^{-2}, il vient :

```
 Tolerance, TOL, in test << C_{L,L} >= 1 ? >> :  1.0000000000000002E-002

Returning from Gram-Schmidt :

    Parameter r (lower bound on rank) =    3
    Number of vectors admitting a known common descent direction, mu =   15
    Defect, m-mu =    0

Permutation of u-vectors =
  14   6  10   4   5   2   7   8
   9   3  11  12  13   1  15

Reordered U matrix :
   2.3383   0.0000  -0.6180   1.6180   0.7909   2.8271  -0.6180  -0.9563
   0.0000   2.3383  -0.6180   3.0000   2.8271   2.3383   1.6180   0.7909
  -0.9563   2.8271   1.6180   1.6180   2.3383  -0.0000   3.0000   2.8271
   0.7909   0.7909   3.0000  -0.6180  -0.0000  -0.9563   1.6180   2.3383
   2.8271  -0.9563   1.6180  -0.6180  -0.9563   0.7909  -0.6180  -0.0000

  -0.9563   2.3383  -0.0000   0.7909   1.6180   3.0000   2.8271
   0.0000   2.8271  -0.9563  -0.9563  -0.6180   1.6180   0.7909
   2.3383   0.7909   0.7909   0.0000  -0.6180  -0.6180  -0.9563
   2.8271  -0.9563   2.8271   2.3383   1.6180  -0.6180   0.0000
   0.7909   0.0000   2.3383   2.8271   3.0000   1.6180   2.3383


Orthogonal basis (V matrix) :
   2.3383   0.5652  -6.7397
   0.0000   1.7731  -6.7397
  -0.9563   1.9126   3.9563
   0.7909   0.7909  10.5668
```

```
   2.8271  -0.0418   3.9563


Coefficient-vector beta =
   0.3333   0.6453   0.0214

Coefficient-vector alfa (w.r.t. reordered U matrix) =
   0.4527   0.4527   0.0946

Coefficient-vector alpha (w.r.t. original U matrix) =
   0.0000   0.0000   0.0000   0.0000   0.0000   0.4527   0.0000   0.0000
   0.0000   0.0946   0.0000   0.0000   0.0000   0.4527   0.0000

Provisional direction of search
 omega_1 = V * beta = G * alfa = U * alpha =
   1.0000
   1.0000
   1.0000
   1.0000
   1.0000

Directional derivatives = dot products of omega_1
 with column-vectors of reordered U matrix :
   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000
   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000
Constant a =   1.0000

 UtU matrix :
  15.0000  -4.7815   3.9547
  -4.7815  15.0000   3.9547
   3.9547   3.9547  15.0000

Factorization : upon exit from DPOTRF, info =
   0

Inversion : upon exit from DPOTRS, info =
   0

Matrix W :
   0.2229   0.1047  -0.0331   0.0000   0.1582
   0.1047   0.2229   0.1582  -0.0000  -0.0331
  -0.1276  -0.1276   0.0749   0.2000   0.0749

Eta-vectors :
   1.0000   0.0000   0.0000   0.5234   0.2436   1.0000  -0.1654  -0.2238
   0.0000   1.0000   0.0000   1.1144   1.1144   0.7909   0.7909   0.5234
   0.0000  -0.0000   1.0000  -0.6378  -0.3580  -0.7909   0.3744   0.7004
```

```
  -0.1654   0.7909   0.2436   0.5234   0.7909   1.1144   1.1144
   0.2436   1.0000  -0.1654  -0.2238  -0.1654   0.5234   0.2436
   0.9217  -0.7909   0.9217   0.7004   0.3744  -0.6378  -0.3580
```

Une base orthogonale de dimension 3 est formée, et la procédure identifie que bissectrice

$$\omega_1 = (1, 1, 1, 1, 1)^t$$

est une direction pour laquelle toutes dérivées directionnelles sont positives et même égales (à 5). Néanmoins, le calcul est poursuivi. En définitive, les 12 vecteurs η_j d'indices $j > 3$ ont la même structure : 2 composantes strictement positives, 1 strictement négative, la somme des 3 composantes égale à 1. Ceci confirme que les 15 dérivées dans la direction de ω_1 sont égales. Cette conclusion très favorable est évidemment liée aux nombreuses symétries du cas-test.

On réalise enfin un cas-test plus difficile en fixant $b_\phi = 10$ et $TOL = 10^{-2}$:

```
Specified vectors (U0 matrix) :
  11.0000  10.1355   7.6913   4.0902  -0.0453  -4.0000  -7.0902  -8.7815
   4.0902   7.6913  10.1355  11.0000  10.1355   7.6913   4.0902  -0.0453
  -7.0902  -4.0000  -0.0453   4.0902   7.6913  10.1355  11.0000  10.1355
  -7.0902  -8.7815  -8.7815  -7.0902  -4.0000  -0.0453   4.0902   7.6913
   4.0902  -0.0453  -4.0000  -7.0902  -8.7815  -8.7815  -7.0902  -4.0000

  -8.7815  -7.0902  -4.0000  -0.0453   4.0902   7.6913  10.1355
  -4.0000  -7.0902  -8.7815  -8.7815  -7.0902  -4.0000  -0.0453
   7.6913   4.0902  -0.0453  -4.0000  -7.0902  -8.7815  -8.7815
  10.1355  11.0000  10.1355   7.6913   4.0902  -0.0453  -4.0000
  -0.0453   4.0902   7.6913  10.1355  11.0000  10.1355   7.6913

 Parameter r_max (upper bound on rank) =            5
```

Malgré les variations plus grandes d'un vecteur au suivant, notamment dans le signe des composantes, les observations sont identiques. Ceci est sans doûte le résultat des symétries du cas-test et de la permutation qui tend à prendre en compte au mieux ces variations :

```
 Tolerance, TOL, in test << C_{L,L} >= 1 ? >> :  1.0000000000000002E-002

Returning from Gram-Schmidt :

    Parameter r (lower bound on rank) =    3
    Number of vectors admitting a known common descent direction, mu =   15
    Defect, m-mu =    0

Permutation of u-vectors =
  14   6  10   4   5   2   7   8
   9   3  11  12  13   1  15

Reordered U matrix :
   7.6913  -4.0000  -7.0902   4.0902  -0.0453  10.1355  -7.0902  -8.7815
```

```
  -4.0000   7.6913  -7.0902  11.0000  10.1355   7.6913   4.0902  -0.0453
  -8.7815  10.1355   4.0902   4.0902   7.6913  -4.0000  11.0000  10.1355
  -0.0453  -0.0453  11.0000  -7.0902  -4.0000  -8.7815   4.0902   7.6913
  10.1355  -8.7815   4.0902  -7.0902  -8.7815  -0.0453  -7.0902  -4.0000

  -8.7815   7.6913  -4.0000  -0.0453   4.0902  11.0000  10.1355
  -4.0000  10.1355  -8.7815  -8.7815  -7.0902   4.0902  -0.0453
   7.6913  -0.0453  -0.0453  -4.0000  -7.0902  -7.0902  -8.7815
  10.1355  -8.7815  10.1355   7.6913   4.0902  -7.0902  -4.0000
  -0.0453  -4.0000   7.6913  10.1355  11.0000   4.0902   7.6913


Orthogonal basis (V matrix) :
   7.6913   1.6629  -0.5479
  -4.0000   2.0284  -0.5479
  -8.7815   0.9727   1.5913
  -0.0453  -0.0453   2.9134
  10.1355   0.3812   1.5913


Coefficient-vector beta =
   0.0196   0.6271   0.3533

Coefficient-vector alfa (w.r.t. reordered U matrix) =
   0.4527   0.4527   0.0946

Coefficient-vector alpha (w.r.t. original U matrix) =
   0.0000   0.0000   0.0000   0.0000   0.0000   0.4527   0.0000   0.0000
   0.0000   0.0946   0.0000   0.0000   0.0000   0.4527   0.0000

Provisional direction of search
 omega_1 = V * beta = G * alfa = U * alpha =
   1.0000
   1.0000
   1.0000
   1.0000
   1.0000

Directional derivatives = dot products of omega_1
 with column-vectors of reordered U matrix :
   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000
   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000   5.0000
Constant a =   1.0000

 UtU matrix :
 255.0000-239.5369 -21.1321
-239.5369 255.0000 -21.1321
```

```
 -21.1321 -21.1321 255.0000

Factorization : upon exit from DPOTRF, info =
   0

Inversion : upon exit from DPOTRS, info =
   0

Matrix W :
   0.1170   0.0934   0.0658   0.0724   0.1041
   0.0934   0.1170   0.1041   0.0724   0.0658
  -0.0104  -0.0104   0.0301   0.0551   0.0301

Eta-vectors :
   1.0000   0.0000  -0.0000   0.5234   0.2436   1.0000  -0.1654  -0.2238
  -0.0000   1.0000  -0.0000   1.1144   1.1144   0.7909   0.7909   0.5234
  -0.0000   0.0000   1.0000  -0.6378  -0.3580  -0.7909   0.3744   0.7004

  -0.1654   0.7909   0.2436   0.5234   0.7909   1.1144   1.1144
   0.2436   1.0000  -0.1654  -0.2238  -0.1654   0.5234   0.2436
   0.9217  -0.7909   0.9217   0.7004   0.3744  -0.6378  -0.3580
```

C.7 Conclusion

Ces exemples ont mis en évidence que le processus d'orthogonalisation de Gram-Schmidt, organisé selon l'Algorithme MGDA, défini précisément par les Tables 4.1-4.2-4.3, fournit une information riche relative aux gradients, leur rang, la configuration éventuelle de Pareto-stationnarité, et le cas échéant, simplifie la formulation du problème de programmation quadratique qu'il reste à résoudre pour déterminer le vecteur de plus petite norme ω et la direction de descente $\mathbf{d}$ qui en résulte.

Annexe D

Transformation exponentielle et convexité

Correction de l'Exercice 18

1. Un développement limité général au second-ordre de la fonction $f(\mathbf{x})$ autour du point $\mathbf{x}_0$ s'écrit :

$$f(\mathbf{x}_0 + \delta x) = f_0 + \big(\nabla f_0, \delta x\big) + \tfrac{1}{2}\big(\delta x, H_0\,\delta x\big) + O\left(\|\delta x\|^3\right) \tag{D.1}$$

où le produit scalaire est noté $\big(.,.\big)$, l'indice $_0$ réfère à une évaluation en $\mathbf{x} = \mathbf{x}_0$, ∇f_0 est le vecteur gradient et $H_0 = \nabla\nabla^t f_0$ la matrice hessienne. Avec $\delta x = -\epsilon\omega$, il vient :

$$\phi(\epsilon) = f(\mathbf{x}_0 - \epsilon\omega) = f_0 \underbrace{-\epsilon\big(\nabla f_0, \omega\big)}_{\epsilon\phi'(0)} + \tfrac{1}{2}\epsilon^2 \underbrace{\big(\omega, H_0\,\omega\big)}_{\phi''(0)} + O\left(\epsilon^3\right) \tag{D.2}$$

ce qui donne $\phi(0) = f_0$, $\phi'(0) = -\big(\omega, \nabla f_0\big)$, et $\phi''(0) = \big(\omega, H_0\,\omega\big)$.

Par conséquent, dès lors que le produit scalaire $\big(\omega, \nabla f_0\big)$ est positif, ce qui est vrai en particulier si $\omega = \nabla f_0 \neq 0$, $\phi'(0) < 0$ et

$$f(\mathbf{x}_0 - \epsilon\omega) < f_0 \tag{D.3}$$

pourvu que ϵ soit strictement positif et suffisamment petit ("itération de descente").

Ce résultat classique étant établi, la question du choix optimal du pas ϵ se pose. Dans le cas convexe ($\phi''(0) > 0$), il est naturel de fixer ϵ au minimum local du développement limité :

$$\epsilon^\star \approx \frac{-\phi'(0)}{\phi''(0)} = \frac{\big(\omega, \nabla f_0\big)}{\big(\omega, H_0\,\omega\big)}. \tag{D.4}$$

Dans le cas inverse (concave), le développement limité au lieu de présenter un minimum pour une valeur positive de ϵ, présente un maximum pour une valeur négative. On ne peut donc plus ajuster le pas sur la seule base de la parabole osculatrice. Il est alors courant d'appliquer un algorithme de minimisation 1D de type essai-erreur dichotomique. La construction suivante est une alternative à un tel algorithme d'essai-erreur.

2. La fonction $\tilde{f}(\mathbf{x})$ a les propriétés suivantes :
(i) même régularité que $f(\mathbf{x})$ car l'exponentielle est une fonction entière (analytique sur $\mathbb{R}$, et de rayon de convergence ∞),

(ii) même sens de variation que $f(\mathbf{x})$ car l'exponentielle est une fonction uniformément strictement monotone croissante,
(iii) valeurs : $\tilde{f}(\mathbf{x}) > 0$, $\forall \mathbf{x}$ (utile dans le cas d'une minimisation en ordre de grandeur, plutôt qu'en valeur absolue) ; $\tilde{f}_0 = 1/\lambda$,
(iv) gradient : par différentiation :

$$\frac{\partial \tilde{f}(\mathbf{x})}{\partial x_i} = e^{\lambda\left(f(\mathbf{x})-f_0\right)}\, \frac{\partial f(\mathbf{x})}{\partial x_i} \tag{D.5}$$

et, en particulier, par assemblage en $\mathbf{x} = \mathbf{x}_0$:

$$\nabla \tilde{f}_0 = \nabla f_0 \tag{D.6}$$

(v) matrice hessienne : retour à (D.5), par différentiation :

$$\frac{\partial^2 \tilde{f}(\mathbf{x})}{\partial x_i \partial x_j} = e^{\lambda\left(f(\mathbf{x})-f_0\right)} \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} + \lambda e^{\lambda\left(f(\mathbf{x})-f_0\right)}\, \frac{\partial f(\mathbf{x})}{\partial x_j}\, \frac{\partial f(\mathbf{x})}{\partial x_i} \tag{D.7}$$

et assemblage :

$$\tilde{H}(\mathbf{x}) = e^{\lambda\left(f(\mathbf{x})-f_0\right)}\, \left(H(\mathbf{x}) + \lambda\, [\nabla f(\mathbf{x})]\, [\nabla f(\mathbf{x})]^t\right) \tag{D.8}$$

où, conformément à la notation matricielle, $[\nabla f(\mathbf{x})]$ désigne le vecteur colonne des composantes du gradient, et l'indice supérieur ^{t} indique la transposition. En particulier :

$$\tilde{H}_0 = H_0 + \lambda\, [\nabla f_0]\, [\nabla f_0]^t\,. \tag{D.9}$$

La matrice $[\nabla f_0]\, [\nabla f_0]^t$ est réelle-symétrique semi-définie positive. Elle n'est que de rang 1, mais effective dans toute direction non orthogonale au gradient ∇f_0, en particulier :

$$\tilde{\phi}''(0) = \left(\omega, \tilde{H}_0\, \omega\right) = \phi''(0) + \lambda\, \phi'(0)^2. \tag{D.10}$$

Par hypothèse $\phi'(0) < 0$ (donc non nul), et $\phi''(0) < 0$. Par conséquent $\tilde{\phi}''(0) > 0$ pourvu que λ soit suffisamment grand, à savoir

$$\lambda > -\frac{\phi''(0)}{\phi'(0)^2} = \frac{|\phi''(0)|}{\phi'(0)^2} \tag{D.11}$$

ce qui permet d'appliquer la technique de la parabole osculatrice à la descente de $\tilde{f}$ en lieu et place de f.

En conclusion la transformation permet d'agrandir le domaine à partir duquel l'itération du gradient peut plus facilement être initialisée sans que la question de la convexité locale se pose.

3. On détermine d'abord la direction de descente $\omega = \omega^\star$ associée aux gradients locaux des fonctions coûts $\{f_j(\mathbf{x})\}$. Si ces fonctions sont toutes convexes dans la direction de descente on applique aucune transformation. Sinon, on applique une transformation exponentielle particulière à chacune de celles dont les restrictions ne sont pas convexes. Ces transformations préservent les gradients locaux, et par conséquent aussi la direction commune de descente $\omega^\star$. Ainsi les fonctions transformées $\{\tilde{f}_j(\mathbf{x})\}$ sont convexes dans la direction de descente. Pour chaque j, on détermine le pas optimal $\epsilon_j = -\phi_j'(0)/\phi_j''(0)$ ou $-\tilde{\phi}_j'(0)/\tilde{\phi}_j''(0)$. Un choix stable du pas consiste à poser $\epsilon = \min(\epsilon_j)$.

Remarque 4
Dans l'optimisation structurale concourante d'un élément sandwich[1] *vis-à-vis de sa masse, des forces critiques de défaillance en flexion (deux premiers modes), et de la résistance à une explosion (critères d'énergie absorbée et de déflection), on a observé une très nette amélioration de la stabilité de l'approche numérique en remplaçant le critère d'énergie absorbée par son exponentielle.*

Remarque 5
Dans le problème de dimensionnement d'un avion d'affaires supersonique (SSBJ)[2] *on avait utilisé une transformation exponentielle sur chaque fonction pour normaliser les échelles. On peut penser que cette transformation avait aussi amélioré la convexité des fonctions, et rendu le processus numérique plus robuste.*

1. *Jean-Antoine Désidéri, Pierre Leite, Quentin Mercier. Prioritized multi-objective optimization of a sandwich panel.* [Research Report] RR-9362, INRIA Sophia Antipolis - Méditerranée (France). 2020. `https://hal.inria.fr/hal-02931770v1`

2. *Jean-Antoine Désidéri. Platform for prioritized multi-objective optimization by metamodel-assisted Nash games.* [Research Report] RR-9290, Inria Sophia Antipolis. 2019. `https://hal.inria.fr/hal-02285197v1`

Annexe E

Équation de sensibilité en instationnaire

À titre d'exemple, on considère d'abord le problème d'évolution parabolique suivant :

$$\begin{cases} \dfrac{\partial u}{\partial t} = \nu(a)\dfrac{\partial^2 u}{\partial x^2} & (0 < x < 1;\ 0 < t < T) \\ u(0,t) = u(1,t) - \phi(a,t) = 0 & (\forall t \in [0,T]) \\ u(x,0) = u_0(x) & (\forall x \in [0,1]) \end{cases} \tag{E.1}$$

dans lequel le paramètre a conditionne à la fois l'EDP par le biais du coefficient de viscosité variable $\nu(a)$, et la condition au bord droit du domaine spatial $\Omega = [0,1]$ par la fonction $\phi(a,t)$. En conséquence, la solution de cette EDP dépend paramètriquement de a :

$$u = u_a(x,t) \tag{E.2}$$

et on suppose (ou on démontre) que cette dépendance est régulière, au sens de l'existence de dérivées par rapport à a. En particulier, on adopte la notation suivante :

$$u'_a = \frac{\partial u_a(x,t)}{\partial a} = u'_a(x,t)\,. \tag{E.3}$$

On s'intéresse à une fonctionnelle de la solution, exprimée sous la forme d'une intégrale de u et de ses dérivées partielles en espace,

$$J = \int_0^1 \psi\big[u(x,t), u_x(x,t), \ldots\big]\, dx \tag{E.4}$$

qui, de fait, dépend du temps t, mais aussi du paramètre a :

$$J = J(a,t) \tag{E.5}$$

Comment alors exprimer le gradient paramétrique de fonctionnelle (par rapport à a) en fonction du temps ?

On obtient l'équation de sensibilité par rapport à a en dérivant l'EDP et les conditions auxquelles elle est soumise par rapport à a :

$$\begin{cases} \dfrac{\partial u'_a}{\partial t} = \nu(a)\dfrac{\partial^2 u'_a}{\partial x^2} + \nu'(a)\dfrac{\partial^2 u_a}{\partial x^2} & (0 < x < 1;\ 0 < t < T) \\ u'_a(0,t) = u'_a(1,t) - \phi'(a,t) = 0 & (\forall t \in [0,T]) \\ u'_a(x,0) = 0 & (\forall x \in [0,1]) \end{cases} \tag{E.6}$$

où :

$$\nu'(a) = \frac{\partial \nu(a)}{\partial a}\,, \quad \phi'(a,t) = \frac{\partial \phi(a,t)}{\partial a}\,. \tag{E.7}$$

En conséquence le gradient paramétrique de la fonctionnelle s'exprime comme suit :

$$g = \frac{\partial J}{\partial a} = \int_0^1 \left[\psi_u u'_a + \psi_{u_x} \frac{\partial u'_a}{\partial x} + \dots \right] dx = g(a,t) \tag{E.8}$$

où l'on a permuté les opérateurs de dérivation :

$$\frac{\partial u_x}{\partial a} = \frac{\partial u'_a}{\partial x}\,. \tag{E.9}$$

On obtient donc le gradient paramétrique de fonctionnelle par :

- intégration directe en temps de l'équation de sensibilité, (E.6), simultanément à l'intégration du problème d'évolution de départ, (E.1), soumise à des conditions initiales et aux limites obtenues par différentiation (ou "linéarisation") par rapport au paramètre a ;
- substitution dans l'expression du gradient paramétrique à l'instant t, (E.8).

Cette approche diffère notoirement de celle par "équation adjointe en instationnaire". L'équation de sensibilité s'intègre en "mode direct", c'est-à-dire dans le sens de t croissant, simultanémemnt et à l'instar du problème d'évolution de départ. La connaissance sur l'intervalle en temps complet de $u_a(x,t)$ (en tout x) n'est donc pas requise. Numériquement, on procède par pas de temps successifs, en calculant au temps $t^{n+1} = (n+1)\Delta t$, les valeurs de $u(x_j, t^{n+1})$ et $u'_a(x_j, t^{n+1})$ par résolution d'un système élargi, discrétisant (E.1) et (E.6) globalement.

Précisons maintenant quelques aspects numériques de la résolution pour un problème d'évolution plus géneral dont la version discrétisée en espace seulement s'écrit :

$$\frac{d}{dt}\mathbf{u}_h + \mathbf{R}_h(\mathbf{u}_h) = 0 \tag{E.10}$$

où $\mathbf{u}_h$ est la solution discrète, et $\mathbf{R}_h(\mathbf{u}_h)$ le résidu discret. L'intégration en temps de ce problème peut s'effectuer par exemple par la méthode implicite d'Euler linéarisée :

$$\frac{\mathbf{u}_h^{n+1} - \mathbf{u}_h^n}{\Delta t} + \underbrace{\mathbf{R}_h(\mathbf{u}_h^{n+1})}_{\mathbf{R}_h(\mathbf{u}_h^n) + \mathbf{R}'_h(\mathbf{u}_h^n)\left(\mathbf{u}_h^{n+1} - \mathbf{u}_h^n\right) + \dots} = 0 \tag{E.11}$$

où :

$$\mathbf{R}'_h(\mathbf{u}_h^n) = \frac{\partial \mathbf{R}_h(\mathbf{u}_h^n)}{\partial \mathbf{u}_h^n}\,. \tag{E.12}$$

On aboutit ainsi à la résolution au pas de temps $(n+1)$ du système linéaire suivant mis "sous forme-Δ" :

$$\left[\mathbf{I} + \Delta t\, \mathbf{R}'_h(\mathbf{u}_h^n)\right]\left(\mathbf{u}_h^{n+1} - \mathbf{u}_h^n\right) = -\Delta t\, \mathbf{R}_h(\mathbf{u}_h^n) \quad (+\text{ CL's}) \tag{E.13}$$

où les conditions aux limites discrètes (CL) ne sont pas précisées ici. Ce système est souvent résolu par relaxation partielle ou complète.

Dans ce contexte, l'équation de sensibilité s'écrit :

$$\frac{d}{dt}\mathbf{u}'_h + \mathbf{R}'_h(\mathbf{u}_h)\mathbf{u}'_h + \frac{\partial}{\partial a}\mathbf{R}_h(\mathbf{u}_h) = 0 \tag{E.14}$$

où :

$$\mathbf{u}'_h = \frac{\partial \mathbf{u}_h}{\partial a} \tag{E.15}$$

est la sensibilité de $\mathbf{u}_h$ au paramètre a. Cette équation qui résulte d'une différentiation, ou linéarisation, est toujours linéaire, quel que soit le problème d'évolution considéré, mais possiblement non homogène par le terme source $\frac{\partial}{\partial a}\mathbf{R}_h(\mathbf{u}_h)$, et la linéarisation des conditions aux limites, omise ici. En intégrant cette équation en temps par la méthode d'Euler, il vient :

$$\frac{\mathbf{u}_h'^{n+1} - \mathbf{u}_h'^{n}}{\Delta t} + \mathbf{R}'_h(u_n^n)\mathbf{u}_h'^{n+1} + \frac{\partial}{\partial a}\mathbf{R}_h(\mathbf{u}_h^n) = 0\,. \tag{E.16}$$

Cette équation peut s'écrire sous forme-Δ en remarquant que $\mathbf{u}_h'^{n+1} = \left(\mathbf{u}_h'^{n+1} - \mathbf{u}_h'^{n}\right) + \mathbf{u}_h'^{n}$:

$$\left[\mathbf{I} + \Delta t\, \mathbf{R}'_h(\mathbf{u}_h^n)\right]\left(\mathbf{u}_h'^{\,n+1} - \mathbf{u}_h'^{\,n}\right) = -\Delta t\left[\mathbf{R}'_h(\mathbf{u}_h^n)\mathbf{u}_h'^{n} + \frac{\partial}{\partial a}\mathbf{R}_h(\mathbf{u}_h^n)\right] \quad (+\text{ CL's linéarisées})\,. \tag{E.17}$$

En comparant les systèmes linéaires (E.14) et (E.17), on observe qu'ils font apparaître la même matrice,

$$\mathbf{A} = \left[\mathbf{I} + \Delta t\, \mathbf{R}'_h(\mathbf{u}_h^n)\right] \tag{E.18}$$

construite à partir du linéarisé du résidu par rapport à la solution discrète $\mathbf{u}_h$. Techniquement, la prise en compte de l'équation de sensibilité se fait donc en chargeant des membres de droite supplémentaires, sans changer de matrice, et en réalisant la linéarisation des conditions aux limites.

Ici, on a choisi la méthode d'Euler implicite pour intégrer en temps. Cette méthode fait intervenir les solutions discrètes aux deux niveaux en temps n et $n+1$. Deux niveaux de stockage ont donc été nécessaires. Avec une méthode d'intégration en temps plus précise, le nombre de niveaux de stockage serait supérieur [1], mais fixé. En aucune manière, il serait requis de stocker toute la solution instationnaire, sauf à d'autre fin.

1. Typiquement 3 pour une formule du second ordre rétrograde en temps.

Table des matières