



HAL
open science

Modèles et algorithmes pour la planification de production à moyen terme en environnement incertain

Arnaud Lenoir

► **To cite this version:**

Arnaud Lenoir. Modèles et algorithmes pour la planification de production à moyen terme en environnement incertain. Algorithmes et structure de données [cs.DS]. Université Blaise Pascal - Clermont-Ferrand II, 2008. Français. NNT : 2008CLF21875 . tel-00731056

HAL Id: tel-00731056

<https://theses.hal.science/tel-00731056>

Submitted on 11 Sep 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ BLAISE PASCAL - CLERMONT II
ÉCOLE DOCTORALE DES SCIENCES
POUR L'INGÉNIEUR DE CLERMONT-FERRAND

THÈSE

présentée pour
l'obtention du titre de

DOCTEUR D'UNIVERSITÉ

Spécialité : Informatique et Ingénierie

par **Arnaud LENOIR**

*Modèles et algorithmes pour la planification de
production à moyen terme en environnement incertain :
Application de méthodes de décomposition proximales*

Soutenue publiquement le 14 novembre 2008 devant le jury :

| | |
|----------------------|--------------------|
| M. Alain Quilliot | Président |
| M. Pierre Carpentier | Rapporteur |
| M. Patrick Combettes | Rapporteur |
| M. Claude Lemaréchal | Examineur |
| M. Adam Ouorou | Examineur |
| M. Kengy Barty | Examineur |
| M. Philippe Mahey | Directeur de thèse |

Table des matières

| | |
|--|-----------|
| Remerciements | v |
| Résumé | vii |
| Abstract | ix |
| Table des figures | xii |
| Liste des algorithmes | xiii |
| Notations | xv |
| Introduction | 1 |
| Avant-propos | 1 |
| Objectifs, cadre de l'étude | 2 |
| Contributions | 3 |
| Organisation du mémoire | 3 |
| 1 Gestion de production électrique : un grand système | 5 |
| 1.1 Introduction | 5 |
| 1.2 Modèle de gestion de la production électrique | 6 |
| 1.2.1 Centrales thermiques à flamme | 6 |
| 1.2.2 Centrales hydrauliques | 6 |
| 1.2.3 Centrales thermiques nucléaires | 7 |
| 1.2.4 Satisfaction de la demande | 8 |
| 1.3 Modélisation des incertitudes | 9 |
| 1.3.1 Caractère aléatoire des données | 9 |
| 1.3.2 Choix du critère | 10 |
| 1.3.3 Problèmes avec recours | 11 |
| 1.4 Discrétisation des problèmes multi-étapes | 12 |
| 1.4.1 Arbres de scénario | 12 |
| 1.4.2 Estimation non-paramétrique | 13 |
| 1.4.3 Pénalisation de l'anticipativité | 16 |
| 1.4.4 Partition de l'unité | 21 |
| 1.4.5 Couplages des aléas | 23 |
| 2 Régularisation d'inclusions monotones | 25 |
| 2.1 Introduction | 25 |
| 2.2 Algorithmes de points fixes pour la recherche de zéros | 25 |
| 2.2.1 Algorithme forward | 25 |

| | | |
|----------|---|------------|
| 2.2.2 | Algorithme du point proximal généralisé | 27 |
| 2.2.3 | Applications à l'optimisation | 29 |
| 2.3 | Quelques propriétés des réflexions généralisées | 32 |
| 2.3.1 | Propriétés de contraction | 33 |
| 2.3.2 | Vitesse de convergence de l'algorithme proximal | 37 |
| 2.3.3 | Conditions de différentiabilité | 38 |
| 2.3.4 | Lois internes \odot et \star | 42 |
| 3 | Éclatement d'opérateurs : décomposition et coordination | 45 |
| 3.1 | Introduction | 45 |
| 3.2 | Méthode Forward-Backward | 46 |
| 3.2.1 | Problème de point fixe équivalent | 46 |
| 3.2.2 | Algorithme | 46 |
| 3.2.3 | Application à la décomposition | 48 |
| 3.2.4 | Convergence | 49 |
| 3.3 | Méthode de Peaceman-Rachford | 50 |
| 3.3.1 | Problème de point fixe équivalent | 50 |
| 3.3.2 | Algorithme | 50 |
| 3.3.3 | Convergence | 52 |
| 3.4 | Méthode de Douglas-Rachford généralisée | 54 |
| 3.4.1 | Application à la décomposition | 55 |
| 3.4.2 | Méthodes de l'inverse partielle ou de décomposition proximale | 58 |
| 3.4.3 | Vitesse de convergence | 62 |
| 3.4.4 | Applications à la gestion de production | 62 |
| 4 | Accélération de la convergence | 67 |
| 4.1 | Introduction | 67 |
| 4.2 | Facteur d'échelle variable | 69 |
| 4.2.1 | Algorithme | 70 |
| 4.2.2 | Convergence globale | 70 |
| 4.2.3 | Vitesse de convergence | 74 |
| 4.3 | ∂F ou ∂C sont différentiables | 75 |
| 4.3.1 | Convergence super-linéaire | 75 |
| 4.3.2 | Mise à jour du facteur d'échelle | 76 |
| 4.3.3 | Expériences numériques | 78 |
| 4.4 | F et C sont polyédrales | 79 |
| 4.4.1 | Comportement local | 79 |
| 4.4.2 | Lien avec les opérateurs de pliage | 83 |
| 4.4.3 | Accélération de la convergence | 85 |
| 4.5 | Méthode itérative par bloc | 89 |
| 4.5.1 | Algorithme | 92 |
| 4.5.2 | Analyse de la convergence | 94 |
| 4.5.3 | Expériences numériques | 97 |
| | Conclusion et perspectives | 101 |

| | | |
|----------|--|------------|
| A | Optimisation convexe et inclusions monotones | 103 |
| A.1 | Minimisation d'une fonction convexe | 103 |
| A.1.1 | Opérateurs multivoques | 103 |
| A.1.2 | Opérateurs monotones | 104 |
| A.1.3 | Opérateur sous-différentiel | 105 |
| A.1.4 | Cône normal | 105 |
| A.1.5 | Caractérisation des solutions de (A.1) | 106 |
| A.1.6 | Distributivité de la sous-différentiation | 106 |
| A.2 | Problèmes de minimisation paramétrique | 107 |
| A.3 | Dualité | 109 |
| B | Algorithmes de points fixes | 111 |
| B.1 | Opérateurs contractants, faiblement contractants et moyennés | 111 |
| B.1.1 | Cas des opérateurs affines | 113 |
| B.2 | Opérateurs pseudo-contractants | 115 |
| B.3 | Théorèmes de convergence | 116 |
| C | Convergence graphique et proto-différentiabilité | 119 |
| C.1 | Différentiabilité | 119 |
| C.2 | Convergence graphique d'opérateurs | 119 |
| C.3 | Proto-différentiabilité | 120 |
| | Bibliographie | 120 |

Remerciements

Je tiens en premier lieu à remercier les personnes qui m'ont fait l'honneur de participer au jury de cette thèse. Je les remercie avec d'autant plus de respect que le travail effectué pendant ces trois années doit beaucoup à chacun d'entre eux. Les travaux de Patrick Combettes traitant de l'algorithmique des opérateurs monotones ont constitué une de mes références scientifiques principales. De même, les différentes discussions que j'ai pu avoir avec Pierre Carpentier, et le cours qu'il m'a permis de suivre en première année m'ont apporté une base solide en optimisation stochastique. Je me suis également fortement appuyé sur les excellents ouvrages d'optimisation convexe non-différentiable co-écrits par Claude Lemaréchal et sur les travaux d'Adam Ouorou sur la décomposition proximale. C'est donc une grande reconnaissance que je leur adresse pour avoir accepté d'être membre du jury de cette thèse. Je remercie Philippe Mahey, qui a accepté la responsabilité de diriger ce travail. Il a toujours su se rendre disponible et me faire bénéficier de ses conseils précieux et de son expérience tout en me laissant une grande liberté d'action. C'est à lui et à Alain Quilliot - que je remercie pour avoir présidé le jury de cette thèse - mais aussi à l'ensemble de mes professeurs à l'Institut Supérieur d'Informatique de Modélisation et de leurs Applications (ISIMA), que je dois cet intérêt pour l'optimisation. J'adresse enfin un grand merci à Kengy Barty à la fois pour mon intronisation dans le monde de l'optimisation en environnement incertain et aussi pour l'ambiance de bureau enrichissante qu'il a contribué à créer.

Ce doctorat a été effectué au Laboratoire d'Informatique de Modélisation et d'Optimisation des Systèmes (LIMOS) à Clermont-Ferrand et a fait l'objet d'un partenariat avec le département d'Optimisation SIMulation, RISque et Statistiques (OSIRIS) de la division Recherche & Développement d'EDF. Je tiens à remercier ici tous les membres de ces deux entités qui ont pu permettre la bonne conduite de ce projet et en particulier le groupe *gestion de portefeuille moyen-terme et optimisation stochastique* d'EDF R&D pour son accueil et sa disponibilité.

L'initiative du projet industriel est dû à la fois à Vincent Grellier - que je remercie pour toutes les discussions passionnantes que nous avons eues à propos de la décomposition proximale et du métier de la gestion de production électrique - et aussi à une personne pour qui j'ai une pensée émue : Jean-Sébastien Roy. Il fut notamment à l'origine de la plupart des idées exploitées dans la section 1.4 de ce manuscrit.

Cette section serait incomplète si j'omettais les chercheurs, doctorants et stagiaires qui m'ont accompagné durant cette aventure. Les activités extra-professionnelles que nous avons partagées ont contribué à l'ambiance de travail sereine et formatrice dans laquelle j'ai baigné.

Mes derniers remerciements vont vers mes amis, ma famille et vers Magalie pour le soutien qu'il m'ont toujours accordé, et pour leurs encouragements sans défaillance.

À JS...

Résumé

Nous nous intéressons dans cette thèse aux problèmes d'optimisation de systèmes de grande taille en environnement incertain et plus particulièrement à la résolution de leurs équivalents déterministes par des méthodes de décomposition de type proximal. L'application sous-jacente que nous avons à l'esprit est celle de la gestion optimale de la production électrique d'EDF soumise aux aléas climatique, de marché et de consommation.

Nous mettons à plat les couplages naturels espace-temps-aléas liés à cette application et proposons deux nouveaux schémas de discrétisation pour le couplage des aléas, basés sur l'estimation non-paramétrique de l'espérance conditionnelle, qui constituent des alternatives à la construction d'arbres de scénarios. Nous nous intéressons ensuite aux méthodes de décomposition en travaillant sur un modèle général, celui de la minimisation d'une somme de deux fonctions convexes, la première séparable et l'autre couplante. D'une part, ce modèle simplifié nous exonère de la technicité due à un choix particulier de couplage et de sous-système. D'autre part l'hypothèse de convexité permet de tirer parti de la théorie des opérateurs monotones et de l'identification des méthodes proximales comme des algorithmes de points fixes. Nous mettons l'accent sur les propriétés différentielles des opérateurs de *réflexion généralisée* dont on cherche un point fixe, qui permettent de borner la vitesse de convergence. Nous étudions ensuite deux familles d'algorithmes de décomposition-coordination issues des méthodes dites d'*éclatement d'opérateurs*, à savoir les méthodes Forward-Backward et de type Rachford.

Nous suggérons quelques techniques d'accélération de la convergence des méthodes de type Rachford. Pour cela, nous analysons dans un premier temps la méthode d'un point de vue théorique, fournissant ainsi des explications à certaines observations numériques, avant de proposer des améliorations en réponse. Parmi elles, une mise à jour automatique du facteur d'échelle permet de corriger son éventuel mauvais choix initial. La preuve de convergence de cette technique se voit facilitée grâce aux résultats de stabilité de certaines lois internes vis à vis de la convergence graphique établis en amont. Nous soumettons aussi l'idée d'introduire des "sauts" dans la méthode lorsqu'elle est appliquée à des problèmes polyédraux, en fondant nos arguments sur la géométrie formée par la suite des itérés. En dernier lieu, nous montrons qu'il est possible, en ajoutant un mécanisme de contrôle, de s'affranchir de la résolution de tous les sous-problèmes à chaque itération en préservant la convergence globale.

L'intérêt pratique de ces suggestions est confirmé par des tests numériques sur le problème de gestion de production électrique.

Abstract

We focus in this thesis, on the optimization process of large systems under uncertainty, and more specifically on solving the class of so-called deterministic equivalents with the help of splitting methods. The underlying application we have in mind is the electricity unit commitment problem under climate, market and energy consumption randomness, arising at EDF.

We set the natural time-space-randomness couplings related to this application and we propose two new discretization schemes to tackle the randomness one, each of them based on non-parametric estimation of conditional expectations. This constitute an alternative to the usual scenario tree construction. We use the mathematical model consisting of the sum of two convex functions, a separable one and a coupling one. On the one hand, this simplified model offers a general framework to study decomposition-coordination algorithms by elapsing technicality due to a particular choice of subsystems. On the other hand, the convexity assumption allows to take advantage of monotone operators theory and to identify proximal methods as fixed point algorithms. We underly the differential properties of the *generalized reflections* we are looking for a fixed point in order to derive bounds on the speed of convergence. Then we examine two families of decomposition-coordination algorithms resulting from *operator splitting methods*, namely Forward-Backward and Rachford methods.

We suggest some practical method of acceleration of the Rachford class methods. To this end, we analyze the method from a theoretical point of view, furnishing as a by-product explanations to some numerical observations. Then we propose as a response some improvements. Among them, an automatic updating strategy of scaling factors can correct a potential bad initial choice. The convergence proof is made easier thanks to stability results of some operator composition with respect to graphical convergence provided before. We also submit the idea of introducing “jumps” in the method when applied to polyedral problems based on the geometry shaped by the sequence of iterates. Finally, we prove it is possible to preserve global convergence while not taking into account all subproblems at every iteration but only a subset, via the addition of a control mechanism.

The practical interest of theses propositions is corroborated by numerical experiments performed on the electric production management problem.

Table des figures

| | | |
|-------|---|----|
| 1.4.1 | Aggrégation de scénarios sous la forme d'un arbre | 12 |
| 1.4.2 | Approximation de $\varphi_1(\cdot)$ pour $S = 100$ scénarios en fonction de la largeur de bande h_1 et de la pénalité c | 17 |
| 1.4.3 | Approximation de $\varphi_2(\cdot, \cdot)$ pour $S = 100$ scénarios en fonction de la largeur de bande h_1 et de la pénalité c | 18 |
| 1.4.4 | Approximations de $\varphi_1(\cdot)$ et $\varphi_2(\cdot, \cdot)$, pour $S = 100$ scénarios, avec une largeur de bande $\epsilon_1 = 0.1$ et sans pénalité <i>i.e.</i> $c = 0$ | 19 |
| 1.4.5 | Politique optimale $\varphi_2(\cdot, \cdot)$ | 19 |
| 1.4.6 | Meilleure approximation de $\varphi_1(\cdot)$ pour $S = 10, 27, 129$ et 999 | 20 |
| 1.4.7 | Meilleure approximation de $\varphi_2(\cdot, \cdot)$ pour $S = 10, 27, 129$ and 999 | 20 |
| 2.2.1 | Itérations de l'algorithme de gradient généralisé. | 26 |
| 2.2.2 | Graphe d'une fonction F et de sa régularisée de Moreau-Yosida \check{F} | 28 |
| 2.2.3 | Effet du paramètre λ sur la convergence de l'algorithme du point proximal. | 30 |
| 2.3.1 | Décomposition proximale de s sur le graphe de T | 34 |
| 2.3.2 | Décomposition de Moreau de s sur le graphe de ∇F où F est une forme quadratique. | 34 |
| 2.3.3 | Image $N(s)$ d'un point s par la réflexion généralisée N associée au sous-différentiel d'une forme quadratique | 35 |
| 2.3.4 | Opérateurs proto-dérivées associés à une fonction polyédrale | 42 |
| 3.2.1 | Suite des itérés de l'algorithme Forward-backward | 47 |
| 3.3.1 | Exemple où la méthode de Peaceman-Rachford ne converge pas. | 53 |
| 3.4.1 | Algorithme de Douglas-Rachford en Δ | 56 |
| 3.4.2 | Algorithme de Douglas-Rachford généralisé en Y | 57 |
| 3.4.3 | Schéma des itérés de la méthode de Spingarn pour deux opérateurs | 61 |
| 3.4.4 | Influence du paramètre λ sur la convergence de la méthode de Spingarn. | 62 |
| 3.4.5 | Nombre d'itérations avec un facteur d'échelle par sous-problème puis avec un facteur d'échelle unique. | 65 |
| 4.1.1 | Profil typique du nombre d'itérations en fonction du paramètre λ | 68 |
| 4.1.2 | Comportement de convergence oscillatoire typique | 68 |
| 4.3.1 | Nombre d'itérations de SALA pour différentes stratégies de mise à jour. | 80 |
| 4.3.2 | Trajectoires du paramètre de mise à l'échelle. | 81 |
| 4.3.3 | Trajectoires des paramètres de mise à l'échelle associés à chaque sous-problème pour différentes valeurs de départ $(\lambda_1^0, \lambda_2^0)$ | 81 |
| 4.4.1 | Replis U_{f_i} associés à la projection f_i d'une face d'une fonction polyédrale convexe de \mathbb{R}^2 dans \mathbb{R} | 84 |
| 4.4.2 | Amélioration du taux de convergence pour plusieurs choix de séquences. | 86 |

| | | |
|-------|--|-----|
| 4.4.3 | Domaine de $]0, \pi[\times]0, \pi[$, $\theta_1 \leq \theta_2$ où la convergence est accélérée pour $a = 1$, $L_1 = 2$ | 87 |
| 4.4.4 | Économie en terme du nombre de résolution des sous-problèmes et de temps CPU sur un problème à 50 périodes avec 6 réservoirs | 88 |
| 4.4.5 | Économie en terme du nombre de résolution des sous-problèmes et de temps CPU sur un problème à 350 périodes avec 2 réservoirs | 88 |
| 4.4.6 | Évolution de $\ s^{q\sigma} - s^{q\sigma-1}\ ^2$ pour différent choix de séquences. | 89 |
| 4.5.1 | Schéma algorithmique de la Méthode de Douglas-Rachford itérative par bloc | 93 |
| 4.5.2 | Temps d'exécution de l'algorithme 4.5.4 pour différentes stratégies de sélection des sous-problèmes. | 97 |
| 4.5.3 | Répartition du temps de calcul entre résolution des sous-problèmes et co- ordination pour différentes stratégies dans l'algorithme 4.5.4, $\lambda = 0.005$. . . | 98 |
| 4.5.4 | Profil de $\ \mathcal{Q}(s^k)\ $ en fonction du temps de calcul pour différentes stratégies et différentes valeurs de λ dans l'algorithme 4.5.4. | 99 |
| A.1.1 | Fonction convexe qui n'est pas semi-continue inférieurement (au bord gauche) et qui n'est pas sous-différentiable (en son bord droit) | 105 |
| A.1.2 | Exemple de non distributivité de la sous-différentiation. | 106 |
| B.1.1 | Propriété de contraction d'un opérateur moyenné | 112 |
| C.1.1 | Opérateur monotone maximal différentiable multivoque dans tout voisinage de \bar{y} | 120 |
| C.3.1 | Opérateur qui n'est pas proto-différentiable en \bar{y} | 121 |

Liste des algorithmes

| | | |
|--------|---|-----|
| 2.2.1 | Algorithme du gradient | 26 |
| 2.2.2 | Algorithme forward | 26 |
| 2.2.3 | Itérations de point fixe avec sous-relaxation | 27 |
| 2.2.4 | Algorithme du point proximal généralisé | 28 |
| 2.2.5 | Algorithme proximal appliqué à $T = \partial F$ avec mise à l'échelle | 30 |
| 2.2.6 | Algorithme de lagrangien augmenté | 31 |
| 2.2.7 | Algorithme de lagrangien augmenté proximal | 32 |
| 3.2.1 | Algorithme forward-backward | 47 |
| 3.2.2 | Algorithme forward-backward : version détaillée | 47 |
| 3.2.3 | Algorithme de décomposition par les ressources | 48 |
| 3.2.4 | Algorithme de décomposition par les prix | 49 |
| 3.3.1 | Algorithme de Peaceman-Rachford | 50 |
| 3.3.2 | Algorithme de Peaceman-Rachford (version détaillée) | 51 |
| 3.3.3 | Algorithme de Peaceman-Rachford avec facteur d'échelle | 51 |
| 3.3.4 | Algorithme de Peaceman-Rachford appliqué à la décomposition | 52 |
| 3.4.1 | Algorithme de Douglas-Rachford | 54 |
| 3.4.2 | Algorithme de Douglas-Rachford généralisé | 54 |
| 3.4.3 | Algorithme de Douglas-Rachford généralisé : version Δ | 56 |
| 3.4.4 | Algorithme de Douglas-Rachford généralisé : version Y | 56 |
| 3.4.5 | Algorithme de Douglas-Rachford généralisé (version Y) appliqué à la décomposition | 57 |
| 3.4.6 | Algorithme de lagrangien augmenté | 58 |
| 3.4.7 | Méthode des directions alternées | 58 |
| 3.4.8 | Méthode de l'inverse partielle | 59 |
| 3.4.9 | Méthode de l'inverse partielle (version détaillée) | 60 |
| 3.4.10 | Algorithme de couverture progressive | 63 |
| 3.4.11 | Algorithme de lagrangien augmenté séparable | 64 |
| 4.2.1 | Méthode de Douglas-Rachford généralisée avec facteur d'échelle variable | 70 |
| 4.4.1 | Algorithme de Douglas-Rachford avec sauts de moyenne | 86 |
| 4.5.1 | Algorithme de Douglas-Rachford (version Δ) appliqué à la décomposition | 89 |
| 4.5.2 | Algorithme de Douglas-Rachford | 90 |
| 4.5.3 | Algorithme de Douglas-Rachford par bloc | 91 |
| 4.5.4 | Algorithme de Douglas-Rachford par bloc (détaillé) | 91 |
| 4.5.5 | Algorithme de Douglas-Rachford par bloc avec contrôle de convergence | 93 |
| B.3.1 | Itérations de point fixe avec sous-relaxation | 116 |

Notations

Abbréviations

$\stackrel{def}{=}$ est défini par.
s.c.i semi-continue inférieurement.
 \xrightarrow{g} converge graphiquement vers.

\mathbb{R} Ensemble des réels.
 $B(x, r)$ Boule fermée de rayon r centrée en x .
 $\langle \cdot, \cdot \rangle$ Produit scalaire.
 $\|\cdot\|$ Norme euclidienne.

Opérateurs

$: \mathbb{R}^n \rightarrow \mathbb{R}^n$ Application de \mathbb{R}^n dans lui-même.
 $: \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ Opérateur multivoque de \mathbb{R}^n dans lui-même.
 I Opérateur ou matrice identité.
 I_n Opérateur ou matrice identité de \mathbb{R}^n .
 T, \mathcal{T} Opérateur monotone maximal.
 P, \mathcal{P} Opérateur pseudo-contractant. Résolvant de T, \mathcal{T} .
 Q, \mathcal{Q} Résolvant de T^{-1}, \mathcal{T}^{-1} .
 N, \mathcal{N} Opérateur faiblement contractant.
Réflexion généralisée de T, \mathcal{T} .
 J, \mathcal{J} Opérateur moyenné.

Pour un opérateur N

$dom(N)$ Domaine $\{s \in \mathbb{R}^n | N(s) \neq \emptyset\}$.
 $fix(N)$ Ensemble de points fixes $\{s \in \mathbb{R}^n | s = N(s)\}$.
 $DN(\bar{s}|\bar{d})$ Opérateur proto-différentiel en \bar{s} relativement à \bar{d} .

Pour un ensemble \mathcal{C}

$\delta_{\mathcal{C}}$ Fonction indicatrice ($\delta_{\mathcal{C}}(x) = 0$ si $x \in \mathcal{C}$ et $+\infty$ sinon).
 $\mathbf{N}_{\mathcal{C}}$ Opérateur cône normal.
 $\mathbb{I}_{\mathcal{C}}$ Fonction caractéristique $\mathbb{I}_{\mathcal{C}}(x) = 1$ si $x \in \mathcal{C}$ et 0 sinon.
 $int(\mathcal{C})$ Intérieur.
 $rint(\mathcal{C})$ Intérieur relatif.

Pour une fonction f convexe

| | |
|--------------|---|
| f^* | Conjuguée concave. |
| ∂f | Opérateur sous-différentiel. |
| $epi(f)$ | Épigraphe $\{(x, \nu) \in \mathbb{R}^n \times \mathbb{R} \mid f(x) < \nu\}$. |
| $dom(f)$ | Domaine $\{x \in \mathbb{R}^n \mid f(x) < +\infty\}$. |
| $supp(f)$ | Support. |

Pour une matrice A

| | |
|-------------|----------------------------------|
| A^\top | Matrice transposée. |
| A^\dagger | Pseudo-inverse de Penrose-Moore. |
| $im(A)$ | Espace image. |
| $ker(A)$ | Noyau. |

Introduction

Avant-propos

L'*optimisation* est le processus de recherche de la “meilleure décision” parmi plusieurs alternatives. Cette décision optimale est usuellement modélisée comme la solution d'un *programme mathématique* qui s'écrit de façon générique :

$$\underset{x \in X}{\text{Minimiser}} f(x)$$

La variable x identifie une décision ; X est l'ensemble des alternatives et $f : X \rightarrow \mathbb{R}$ fournit le critère sur lequel le décideur se base pour départager deux décisions. Par définition, une décision x_1 “est meilleure que” x_2 si $f(x_1) \leq f(x_2)$.

La nature des objets mathématiques X et f établit une classification des problèmes d'optimisation en diverses sous-catégories. On parlera d'optimisations continue, combinatoire, mixte, déterministe, stochastique, différentiable, linéaire, convexe, Cette taxinomie permet de mettre en place les approches algorithmiques et théoriques les plus adaptées aux propriétés du problème étudié et a donné lieu depuis de nombreuses années à des branches de recherche séparées.

Nous nous intéressons dans cette thèse à la programmation :

1. convexe : f est une fonction convexe et X un ensemble convexe
2. déterministe : x est un vecteur de \mathbb{R}^n
3. stochastique : x est un vecteur aléatoire de \mathbb{R}^n
4. de grande taille : n possède une “valeur élevée”

et plus précisément aux combinaisons 1-2-4 et 1-3-4. L'optimisation convexe déterministe de grande taille doit faire face à deux difficultés :

- la taille des problèmes à résoudre, qui ne peuvent parfois pas être représentés dans la mémoire d'un ordinateur
- la non différentiabilité des fonctions manipulées

Historiquement, le traitement de la première difficulté a débuté avec les travaux de Dantzig et Wolfe [25] sur la décomposition en programmation linéaire. Une riche exposition des méthodes de décomposition pour les problèmes de grande taille peut être trouvée dans l'ouvrage de référence [54]. Le deuxième obstacle, celui de la non différentiabilité a été abordé par deux grandes voies, celle des approches directes, à savoir les algorithmes de sous-gradient [47], [109] et de faisceaux [57], puis celle de la régularisation [74], [10], [19].

L'optimisation stochastique, quant à elle, a vu le jour avec les travaux de Dantzig [24] dans lesquels il souligne l'importance de prendre en compte le caractère aléatoire des paramètres dans les applications. Le lecteur pourra se référer à [51], [88], [104] pour un tour d'horizon plus complet du domaine.

L'interaction entre optimisations déterministe et stochastique est très forte et intervient ici à deux niveaux.

Du déterministe au stochastique

Les programmes stochastiques sont la plupart du temps issus d'un problème déterministe sous-jacent possédant des paramètres dont on ne connaît pas la valeur avec certitude. Ces paramètres sont alors remplacés naturellement par des variables aléatoires. Les décisions deviennent alors elle-même des variables aléatoires et un nouveau critère doit être défini. Ce peut être, par exemple, l'espérance du critère déterministe ou bien une combinaison avec un critère de risque.

Du stochastique au déterministe

Les variables aléatoires de décision manipulées dans un problème d'optimisation stochastique sont des vecteurs soit de dimension finie (lois sous-jacentes discrètes) soit de dimension infinie. Dans le premier cas, le problème se met naturellement sous la forme d'un programme déterministe en écrivant les espérances comme des sommes. Dans le second cas, il faut nécessairement effectuer une approximation. Une approche usuelle consiste à écrire un *équivalent déterministe* en utilisant une loi approchée discrète des aléas et en discrétisant les contraintes de mesurabilité, par quantification par exemple [2].

Objectifs, cadre de l'étude

Ce doctorat a été effectué au Laboratoire d'Informatique de Modélisation et d'Optimisation des Systèmes (LIMOS) de Clermont-Ferrand. Il prolonge une collaboration entamée en 2004 entre le LIMOS et le département Optimisation SIMulation, RISque et Statistiques (OSIRIS) de la division Recherche & Développement d'EDF sur le thème de l'optimisation des grands systèmes.

Le département OSIRIS intervient dans le domaine de la gestion de production, de la gestion des risques et de l'analyse de portefeuilles. Il participe en particulier au développement de méthodes et d'outils de statistiques, de modélisation et d'optimisation. Cette thèse s'inscrit plus précisément dans l'étude de la gestion de production à l'horizon moyen terme. Le but est de valoriser un certain nombre de stocks d'énergie (par exemple des réserves hydrauliques, du combustible nucléaire ou bien des produits financiers de type options) afin de permettre un arbitrage entre leurs utilisations. Il s'agit essentiellement d'un programme stochastique convexe. La démarche de résolution adoptée consiste à écrire un problème équivalent déterministe dont les solutions approchent celles du problème stochastique.

Une première difficulté provient de la taille du problème. En effet, le système qu'il modélise - le parc de production électrique - comprend plus d'une centaine de centrales aux comportements hétérogènes qu'il faut étudier à un horizon de plusieurs années. De plus, afin de prendre en compte des aléas qui impactent leur fonctionnement, il est nécessaire d'envisager un éventail de scénarios, amplifiant encore la taille du problème. En outre, les modèles résultants sont non différentiables voire non convexes.

Dans ce cadre, nous étudions les *méthodes d'éclatement d'opérateur*, qui permettent de décomposer le problème tout en supportant son caractère non différentiable. L'objectif est de comprendre le fonctionnement de cette famille de méthodes, en nous appuyant sur

les travaux connexes [52], [81], [36], pour l’appliquer de manière adaptée au problème de gestion de production.

Contributions

Nous avons suivi deux axes principaux de réflexion. Le premier concerne la représentation des aléas et notamment la discrétisation des contraintes de non-anticipativité en optimisation stochastique multi-étapes. Cette partie correspond à la section 1.4 et est le fruit d’un travail conjoint avec Jean-Sébastien Roy. Ce travail a été présenté en 2007 à la conférence Euro XXII [102] et a fait l’objet de l’article [103].

Le deuxième axe concerne globalement l’accélération numérique des méthodes d’éclatement d’opérateur. L’apport principal est l’analyse du comportement asymptotique et les modifications algorithmiques apportées en réponse. Nous introduisons notamment une mise à l’échelle “globale” du problème à la fin du chapitre 3 qui permet d’ajuster le conditionnement des sous-problèmes les uns par rapport aux autres. Dans le chapitre 4, la sous-section 4.4 porte sur le cas de l’optimisation polyédrale où un comportement bien spécifique est identifié. Cette partie a fait l’objet de la présentation [67] et de l’article [66]. Les sections 4.2 et 4.3 portent sur le réglage des paramètres proximaux en utilisant les propriétés d’ordre 2 des opérateurs. Ces sections peuvent être retrouvées partiellement dans le rapport interne du LIMOS [64] et l’article en ligne [65]. Elles ont fait l’objet d’une présentation en 2006 lors de la conférence CIMODE [63]. Nous donnons aussi des conditions générales de convergence lorsque les paramètres de mise à l’échelle proximaux varient au cours de l’algorithme. L’application à la gestion de production électrique a été présentée lors de la conférence ENGOPT [62].

Organisation du mémoire

Le premier chapitre décrit le problème de gestion de production électrique et analyse ses couplages naturels espace-temps-aléas. Nous y proposons deux nouveaux schémas de discrétisation pour le couplage des aléas, basés sur l’estimation non-paramétrique de l’espérance conditionnelle.

Le reste de l’étude est dédié aux méthodes d’éclatement d’opérateur et aux algorithmes de décomposition-coordination qui en résultent. Nous nous sommes efforcé de rester dans un contexte assez général en nous basant sur la théorie des opérateurs monotones qui est rappelée partiellement en annexe.

Le chapitre 2 met l’accent sur l’identification des méthodes proximales comme des algorithmes de points fixes. Nous nous intéressons aux propriétés différentielles des opérateurs de *réflexion généralisée* dont on cherche un point fixe, qui permettent de borner la vitesse de convergence. Nous fournissons aussi certains résultats préliminaires de stabilité de la composition vis à vis de la convergence graphique qui faciliteront l’analyse des méthodes d’éclatement dans les chapitres suivants.

Le chapitre 3 détaille deux familles d’algorithmes de décomposition-coordination issues des méthodes d’éclatement, à savoir les méthodes Forward-Backward et de type Rachford, ainsi que leur implémentation. Nous introduisons en fin de chapitre la mise à l’échelle “globale” du problème consistant à ajuster individuellement le conditionnement des sous-problèmes.

Le chapitre 4 final contient une analyse des méthodes de type Rachford d'un point de vue théorique. Il est le lieu de trois suggestions d'améliorations algorithmiques à savoir : la mise à jour automatique des facteurs d'échelle ; l'introduction de "sauts" dans le cadre de l'optimisation polyédrale ; la résolution d'un nombre partiel de sous-problèmes.

Les modifications algorithmiques que nous proposons ont été testées sur diverses instances du problème de gestion de production. Les résultats sont reportés dans les sous-sections 1.4.3, 3.4.4, 4.3.3, 4.4.3 et 4.5.3.

Chapitre 1

Gestion de la production électrique dans l'incertain : étude d'un grand système

1.1 Introduction

Nous nous intéressons dans ce chapitre à l'optimisation d'un *grand système*, c'est à dire d'un système dont le fonctionnement fait intervenir beaucoup de paramètres ([54], [17]). La taille à partir de laquelle un système devient "grand" est relativement subjective. Le consensus délivre ce qualificatif si les techniques frontales de résolution ne s'appliquent pas, soit parce que le temps de calcul est prohibitif, soit tout simplement parce que le modèle n'est pas implantable dans la mémoire d'un ordinateur. Il est alors nécessaire de mettre en place des méthodes de résolution alternatives en tirant profit d'éventuelles caractéristiques additionnelles du problème.

Nous nous focalisons ici sur les grands systèmes résultant de l'interconnexion de multiples sous-systèmes, numérotés par $i \in \{1, \dots, p\}$, un parc de production électrique se divisant en plusieurs centrales par exemple. Les sous-systèmes sont soumis à des décisions locale $y_i, i = 1, \dots, p$ dont l'agrégation forme la variable de décision globale du problème d'optimisation :

$$y = (y_1, \dots, y_p)$$

Le choix de y_i à un niveau local génère un coût $F_i(y_i)$ indépendant des décisions exogènes $y_{j \neq i}$. À ce stade, si les sous-systèmes étaient indépendants les uns des autres, il s'agirait donc de résoudre le problème :

$$\underset{y}{\text{Minimiser}} \quad F(y) = \sum_{i=1}^p F_i(y_i) \quad (1.1)$$

dont les solutions sont obtenues en agrégeant les arguments minimum y_i^* des F_i . L'écueil provient de l'existence d'un *couplage* entre les sous-systèmes, c'est à dire d'une incidence du choix de la décision y_i sur le choix optimal des autres décisions. Il provient en général de la présence dans l'objectif d'une *fonction couplante* $C(\cdot)$ dépendant simultanément de toutes les décisions y_1, \dots, y_p .

$$\underset{y}{\text{Minimiser}} \quad f(y) = \sum_{i=1}^p F_i(y_i) + C(y_1, \dots, y_p) \quad (1.2)$$

L'idée principale des méthodes de décomposition-coordination est de décentraliser la prise de décision vers les sous-systèmes en les rendant artificiellement indépendants comme dans le problème (1.1).

Nous analysons dans ce chapitre un problème de gestion de production électrique rencontré à EDF qui servira de support illustratif dans tout le manuscrit. Ce problème (décrit dans la section 1.2) présente deux couplages naturels, l'un spatial et l'autre temporel. La prise en compte du caractère stochastique de certaines données (section 1.3) et la modélisation de décisions de recours introduisent une troisième dimension de couplage, en aléas dont la formulation varie en fonction du schéma de discrétisation choisi. Le plus répandu, celui des *arbres de scénarios* se prête bien au découplage du problème mais fournit des solutions peu précises. Nous proposons deux nouveaux schémas dans les sous-sections 1.4.3 et 1.4.4 permettant de mieux approcher les solutions mais de structure plus difficiles à découpler.

1.2 Modèle de gestion de la production électrique

Nous considérons un problème de gestion de production électrique. Il consiste à trouver l'utilisation optimale d'un ensemble de centrales pendant une période donnée. Nous devons trouver un planning de production de coût minimal et qui permet de satisfaire la demande en électricité. Nous nous plaçons sur un horizon de T périodes indexées par $t = 1, \dots, T$ et de durées respectives $\delta_1, \dots, \delta_T$. Nous disposons d'usines de production d'électricité indexées par la lettre $u = 1, \dots, U$. Chacune d'entre elles fournit pendant la période t une puissance $p_{u,t} \in \mathbb{R}$. Le coût instantané de cette production est donné par une fonction $f_{u,t} : \mathbb{R} \rightarrow \mathbb{R}$. De plus il existe un coût C_u dépendant de la stratégie de production sur tout l'horizon $(p_{u,1}, p_{u,2}, \dots, p_{u,T}) \in \mathbb{R}^T$. A titre d'exemple, nous donnons les fonctions utilisées pour plusieurs types d'usines.

1.2.1 Centrales thermiques à flamme

Les centrales thermiques à flamme brûlent un combustible fossile (charbon, fuel ou gaz). Leurs caractéristiques physiques limitent la puissance productible :

$$\forall t = 1, \dots, T \quad 0 \leq p_{u,t} \leq \bar{P}_u$$

En première approximation, le coût de fonctionnement est linéaire en fonction de la production et indépendant du temps. La fonction coût instantané associée sera alors :

$$f_{u,t}(p_{u,t}) = \begin{cases} c_u p_{u,t} & \text{si } p_{u,t} \in [0, \bar{P}_u] \\ +\infty & \text{sinon.} \end{cases}$$

Les décisions sont indépendantes dans le temps d'où $C_u \equiv 0$.

1.2.2 Centrales hydrauliques

Une usine hydraulique utilise l'énergie cinétique de l'eau contenue dans un réservoir, un lac en général et la transforme en énergie électrique via un alternateur. Le niveau initial du réservoir (donné) est noté $l_{u,0}$, il varie au cours du temps et son niveau à la fin de la période $t = 1, \dots, T$ est noté $l_{u,t}$. Au début de chaque période t un apport naturel noté $a_{u,t}$ dû aux précipitations et à la fonte des neiges vient alimenter le réservoir. Il est alors décidé :

- soit de le vider partiellement pour entraîner l'alternateur
- soit de façon symétrique de pomper l'eau depuis l'aval pour reconstituer du stock en consommant cette fois-ci de l'électricité.

Si nous notons $v_{u,t}$ le volume d'eau (signé) transitant de l'amont vers l'aval, le débit traversant l'usine vaut $v_{u,t}/\delta_t$. Celui-ci donne alors la puissance fournie pendant la période (toujours notée $p_{u,t}$) grâce à une fonction de rendement $r_u : \mathbb{R} \rightarrow \mathbb{R}$:

$$p_{u,t} = r_u(v_{u,t}/\delta_t) \quad (1.3)$$

La puissance est positive en cas de turbinage et négative en cas de pompage ; il s'agira dans le deuxième cas d'une consommation d'énergie plutôt qu'une réelle production. Comme pour les usines thermiques à flamme, la puissance est limitée inférieurement et supérieurement dans un intervalle $[\underline{P}_u, \overline{P}_u]$. Enfin, le niveau du réservoir est astreint à des capacités maximum et minimum :

$$\underline{L}_u \leq l_{u,t} \leq \overline{L}_u \quad (1.4)$$

Afin de respecter cette contrainte, nous devons envisager la possibilité dans certains cas de déverser un éventuel surplus d'eau sans l'utiliser pour produire. Ainsi, l'équation de dynamique du réservoir s'écrit :

$$l_{u,t} = \min(l_{u,t-1} + a_{u,t} - v_{u,t}, \overline{L}_u) \quad (1.5)$$

Enfin, le coût de fonctionnement des usines hydrauliques dépend de la hauteur du réservoir final, valorisée à l'aide d'une fonction $V_u : \mathbb{R} \rightarrow \mathbb{R}$. Cette hauteur finale dépend de l'ensemble des productions sur l'horizon via la fonction :

$$C_u(p_{u,1}, \dots, p_{u,T}) = \begin{cases} -V_u(l_{u,T}) & \text{si (1.3), (1.4), (1.5);} \\ +\infty & \text{sinon.} \end{cases} \quad (1.6)$$

Le coût instantané est indépendant de la quantité d'eau utilisée et donc de la puissance produite. Il est donc choisi nul sous la contrainte de satisfaire les bornes de puissance :

$$f_{u,t}(p_{u,t}) = \begin{cases} 0 & \text{si } p_{u,t} \in [\underline{P}_u, \overline{P}_u] \\ +\infty & \text{sinon.} \end{cases}$$

1.2.3 Centrales thermiques nucléaires

Une centrale nucléaire de type REP (Réacteur à Eau Pressurisée) utilise la chaleur issue de la fission d'un combustible nucléaire pour mettre sous pression de l'eau qui va entraîner un alternateur et ainsi produire de l'électricité. La puissance $p_{u,t}$ produite pendant la période t doit appartenir à l'intervalle $[\underline{P}_{u,t}, \overline{P}_{u,t}]$.

Pour des raisons de sécurité, de sûreté et de performances, une centrale nucléaire subit régulièrement des opérations de maintenance qui la rend indisponible pendant plusieurs périodes consécutives. Une telle plage d'inactivité est appelée *arrêt*. Nous notons \mathfrak{A} l'ensemble des périodes d'arrêt et $\mathfrak{D} \subset \mathfrak{A}$ le sous-ensemble des périodes marquant leurs débuts. Si la centrale est au repos, nous aurons $\underline{P}_{u,t} = \overline{P}_{u,t} = 0$, impliquant ainsi $p_{u,t} = 0$. Le reste du temps, c'est à dire lorsque la centrale produit de l'électricité, le stock de combustible nucléaire noté $l_{u,t}$ diminue selon la dynamique :

$$l_{u,t} = l_{u,t-1} - \delta_t p_{u,t} \quad \forall t \notin \mathfrak{A} \quad (1.7)$$

avec $l_{u,0}$ donné. Les périodes d'arrêt sont aussi l'occasion de recharger le réacteur. Cette opération consiste à remplacer une fraction $0 < q_{u,t} \leq 1$ du combustible présent par une quantité $L_{u,t}$ de combustible neuf. Elle s'effectue lors de la première période de l'arrêt. Autrement dit, si $t \in \mathfrak{D}$ est une période de début d'arrêt, alors :

$$l_{u,t} = (1 - q_{u,t})l_{u,t-1} + L_{u,t} \quad (1.8)$$

et pour les autres périodes d'arrêt, le stock reste constant :

$$l_{u,t} = l_{u,t-1} \quad \forall t \in \mathfrak{A} \setminus \mathfrak{D} \quad (1.9)$$

Une contrainte physique vient limiter la puissance en fonction du niveau de combustible à l'aide d'une fonction $b_u : \mathbb{R} \rightarrow \mathbb{R}$:

$$p_{u,t} \leq b_u(l_{u,t-1}) \quad (1.10)$$

Comme pour les centrales hydrauliques, le coût de production est égal à la valorisation par une fonction $V_u : \mathbb{R} \rightarrow \mathbb{R}$ du stock de combustible final $l_{u,T}$. Ce coût de production dépend donc conjointement de l'ensemble des productions sur l'horizon :

$$C_u(p_{u,1}, \dots, p_{u,T}) = \begin{cases} \min_{(l_u) \in \mathbb{R}^T} & -V_u(l_{u,T}) \\ \text{s.c} & (1.7)(1.8)(1.9)(1.10) \end{cases} \quad (1.11)$$

Le coût instantané est là aussi choisi nul sous la contrainte de satisfaire les bornes de puissance :

$$f_{u,t}(p_{u,t}) = \begin{cases} 0 & \text{si } p_{u,t} \in [\underline{P}_{u,t}, \bar{P}_{u,t}] \\ +\infty & \text{sinon.} \end{cases}$$

1.2.4 Satisfaction de la demande

A chaque période, la puissance totale à fournir notée D_t doit être répartie entre toutes les usines. Les variables de production sont donc liées par la contrainte de satisfaction de la demande qui s'écrit :

$$\forall t = 1, \dots, T \quad \sum_u p_{u,t} = D_t$$

Le programme mathématique que l'on vient de décrire prend finalement la forme :

$$\underset{p_{u,t}}{\text{Minimiser}} \quad \sum_{u,t} f_{u,t}(p_{u,t}) + \sum_u C_u(p_{u,1}, \dots, p_{u,T}) \quad (1.12a)$$

$$\forall t = 1, \dots, T \quad \sum_u p_{u,t} = D_t \quad (1.12b)$$

On peut réécrire ce problème en définissant pour $t = 1, \dots, T$ la fonction indicatrice $C_t : \mathbb{R}^U \rightarrow \mathbb{R}$:

$$C_t(p_{1,t}, \dots, p_{U,t}) = \begin{cases} 0 & \text{si } \sum_u p_{u,t} = D_t \\ +\infty & \text{sinon.} \end{cases}$$

qui assure la satisfaction de la demande. Le problème (1.12) se réécrit :

$$\text{Minimiser}_{p_{u,t}} \quad \sum_{u,t} f_{u,t}(p_{u,t}) + \sum_u C_u(p_{u,1}, \dots, p_{u,T}) + \sum_t C_t(p_{1,t}, \dots, p_{U,t})$$

La décomposition la plus intuitive correspondant à la notion de système physique, consiste alors à considérer les usines comme sous-systèmes, couplés par la satisfaction de la demande :

$$\text{Minimiser}_{p_u} \quad \sum_u \underbrace{\left[\sum_t f_{u,t}(p_{u,t}) + C_u(p_{u,1}, \dots, p_{u,T}) \right]}_{f_u(p_u)} + \underbrace{\sum_t C_t(p_{1,t}, \dots, p_{U,t})}_{C(p_1, \dots, p_U)}$$

Une alternative est de considérer les périodes plutôt que les usines comme sous-systèmes :

$$\text{Minimiser}_{p_t} \quad \sum_t \underbrace{\left[\sum_u f_{u,t}(p_{u,t}) + C_t(p_{1,t}, \dots, p_{U,t}) \right]}_{f_t(p_t)} + \underbrace{\sum_u C_u(p_{u,1}, \dots, p_{u,T})}_{C(p_1, \dots, p_T)}$$

La troisième décomposition possible est le croisement des deux précédentes, c'est à dire une décomposition spatio-temporelle qui définit un sous-système comme une usine étudiée sur une période :

$$\min_{p_{u,t}} \quad \sum_{u,t} f_{u,t}(p_{u,t}) + \underbrace{\sum_u C_u(p_{u,1}, \dots, p_{u,T}) + \sum_t C_t(p_{1,t}, \dots, p_{U,t})}_{C(p_{1,1}, \dots, p_{U,T})}$$

Les problèmes pratiques à résoudre possèdent environ 80 centrales thermiques, 60 centrales nucléaire 20 centrales hydrauliques et 1 marché, soit $U \approx 160$. Le nombre de pas de temps peut varier de 365 (une année avec une période par jour) à 5475 (cinq années avec trois périodes par jour).

1.3 Modélisation des incertitudes

1.3.1 Caractère aléatoire des données

La modélisation précédente soulève la question de l'acquisition de certaines données telles que la demande, les apports naturels des centrales hydrauliques ou encore les prix de marché. Par nature, ces quantités ne sont connues que de façon *incertaine* au moment où nous écrivons notre problème d'optimisation et ne seront observées que dans le futur.

Afin de simplifier la discussion et d'étendre nos propos à d'autres modèles d'optimisation dans l'incertain, on reconsidère le problème d'optimisation (1.12) en mettant en relief les deux composantes d'intérêt, à savoir les décisions et les paramètres incertains. Nous notons $x_t \in \mathbb{R}^n$ l'ensemble des décisions à prendre à la date t *i.e.* l'ensemble des productions des centrales et $x = (x_1, x_2, \dots, x_T) \in \mathbb{R}^{nT}$ l'agrégation de ces décisions sur tout l'horizon. La lettre $\xi_t \in \mathbb{R}^m$ représentera l'ensemble des paramètres incertains observés au début de la période t (les apports en début de période t , la demande, ...) et $\xi = (\xi_1, \xi_2, \dots, \xi_T) \in \mathbb{R}^{mT}$ leur agrégat. On note aussi de façon condensée

$f : \mathbb{R}^{nT} \times \mathbb{R}^{mT} \rightarrow \mathbb{R}$ la fonction objectif. Le problème (1.12) se réécrit alors de manière abstraite :

$$\underset{(x_1, \dots, x_T) \in \mathbb{R}^{nT}}{\text{Minimiser}} \quad f(x_1, \dots, x_T; \xi_1, \dots, \xi_T) \quad (1.13)$$

L'incertitude sur la valeur de ξ est modélisée mathématiquement à l'aide d'un vecteur aléatoire défini sur un espace probabilisé $(\Omega, \mathcal{F}, \mu)$. Il s'agira en fait d'un processus aléatoire :

$$\begin{aligned} \xi : (\Omega, \mathcal{F}, \mu) &\rightarrow (\mathbb{R}^{mT}, \mathcal{B}) \\ \omega &\mapsto (\xi_1(\omega), \dots, \xi_T(\omega)) \end{aligned}$$

dont les réalisations sont communément appelés *scénarios*. \mathcal{B} est la tribu borélienne de \mathbb{R}^{mT} .

La question qui nous intéresse ici est la suivante : quel problème d'optimisation poser si l'on veut intégrer ce caractère aléatoire au problème (1.13)? Plusieurs réponses ont été apportées dans la littérature et de nombreuses méthodologies proposées. Le choix de la méthode est en grande partie dirigé par la nature des informations dont on dispose sur ξ mais aussi par la complexité du problème que l'on est prêt à accepter ainsi que de l'aversion au risque des décideurs.

1.3.2 Choix du critère

Optimisation robuste On ne dispose parfois que du support de ξ noté $\text{supp}(\xi)$, c'est à dire d'un ensemble d'incertitude pour la valeur du paramètre s . L'esprit des modèles d'optimisation robuste est d'envisager toutes les valeurs possibles des paramètres et d'optimiser la "pire" occurrence dans le sens suivant :

$$\underset{x \in \mathbb{R}^{nT}}{\text{Minimiser}} \quad \sup_{\xi \in \text{supp}(\xi)} f(x; \xi) \quad (1.14)$$

Ce type de technique est à privilégier lorsque l'on cherche une solution fiable (robuste) permettant de faire face à toutes les réalisations possibles de l'aléa et sera donc préférée par un décideur averse au risque.

Optimisation en espérance Pour les décideurs plus enclins au risque, optimiser le pire des cas n'est pas souhaitable. Ces pires cas engendrent un coût élevé pour se couvrir contre des scénarios ayant des probabilités faibles d'arriver. Ils préféreront alors optimiser un critère moyen, comme minimiser un coût de production ou maximiser un bénéfice moyen. Cette approche nécessite le calcul d'une espérance et donc la connaissance de la loi de ξ en plus de son support. Le problème à résoudre est alors :

$$\underset{x \in \mathbb{R}^{nT}}{\text{Minimiser}} \quad \mathbb{E}(f(x; \xi)) \quad (1.15)$$

Les deux critères, robuste et en espérance présentés sont du type "ici et maintenant" selon la terminologie de [124]. Ils visent à déterminer tout de suite l'ensemble des décisions sur l'horizon et ne prennent pas en compte la possibilité qu'a le décideur en pratique d'adapter ses décisions en temps réel. Nous adressons donc deux reproches. D'une part, on dispose d'une telle possibilité de recours dans la réalité et s'en priver, c'est à dire ne pas la modéliser, ne donnera qu'une solution sous-optimale. D'autre part, beaucoup de

problèmes ne peuvent tout simplement pas être formulés sans exprimer ce recours, car des décisions nécessitent pour être réalisables d'avoir observé certains aléas encore inconnus au début de l'horizon. Il est par exemple impossible de choisir dès maintenant une production satisfaisant toutes les demandes possibles ! Une solution de secours consiste à relâcher la contrainte en la pénalisant. La deuxième option présentée ci-après est d'introduire les recours dans le modèle.

1.3.3 Problèmes avec recours

La programmation multi-étapes avec recours consiste en l'optimisation d'un système où les décisions doivent être prises successivement, à certaines étapes d'indices $t = 1, \dots, T$ et être basées sur des informations acquises progressivement. En règle générale, pour modéliser la dépendance d'une décision à une certaine information, on procède comme suit :

1. On modélise l'information en question à l'aide d'une sous-tribu $\mathcal{G} \subset \mathcal{F}$ qui définit les événements de Ω distinguables par le décideur.
2. On exprime la décision comme une fonction appelée *fonction de recours* ou *politique* $\Omega \rightarrow \mathbb{R}^n$, c'est à dire un vecteur aléatoire.
3. On force cette fonction à être mesurable par rapport à \mathcal{G} .

Dans notre cas, l'information accessible au décideur à l'étape t est modélisée par la tribu \mathcal{F}_t engendrée (complétée avec les ensembles de mesure nulle) par le vecteur :

$$\boldsymbol{\pi}_t = (\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_t)$$

qui représente le *passé* du scénario à la période t , c'est à dire les aléas que l'on aura observé et sur lesquels on pourra se baser pour prendre la décision x_t . La séquence $\mathcal{F}_1 \subset \mathcal{F}_2 \subset \dots \subset \mathcal{F}_T \subset \mathcal{F}$ définit une *filtration* ; la propriété d'emboîtement signifiant que l'acquisition d'information est progressive et sans perte de mémoire.

Nous notons désormais $\mathbf{X}_t \in L^\infty(\Omega, \mathcal{F}, \mu, \mathbb{R}^n)$ la décision prise à l'étape t qui rappelle le doit être \mathcal{F}_t -mesurable. Ces contraintes de mesurabilité sont appelées dans la littérature contraintes de *non-anticipativité* : \mathbf{X}_t n'anticipe pas le futur, vu depuis la période t . Dans ce cas, il existe une fonction mesurable φ_t vérifiant :

$$\mathbf{X}_t = \varphi_t \circ \boldsymbol{\pi}_t \tag{1.16}$$

explicitant la dépendance de la décision à l'information. On trouve aussi parfois la terminologie de *variable en boucle fermée* pour les fonctions de recours par opposition à *variable en boucle ouverte* pour les variables non stochastiques.

Le problème de minimisation en espérance incluant la possibilité de recours s'écrit finalement :

$$\underset{(\mathbf{X}_1, \dots, \mathbf{X}_T)}{\text{Minimiser}} \quad \mathbb{E} (f(\mathbf{X}_1, \dots, \mathbf{X}_T; \boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_T)) \tag{1.17a}$$

$$s.c \quad \mathbf{X}_1, \dots, \mathbf{X}_T \in L^\infty(\Omega, \mathcal{F}, \mu, \mathbb{R}^n) \tag{1.17b}$$

$$\mathbf{X}_t \text{ est } \mathcal{F}_t\text{-mesurable} \tag{1.17c}$$

Ce problème n'est pas abordable numériquement puisqu'il manipule en général des objets de dimension infinie (des fonctions). La section suivante aborde la problématique de l'approximation de (1.17) en met en avant les couplages induit par sa discrétisation.

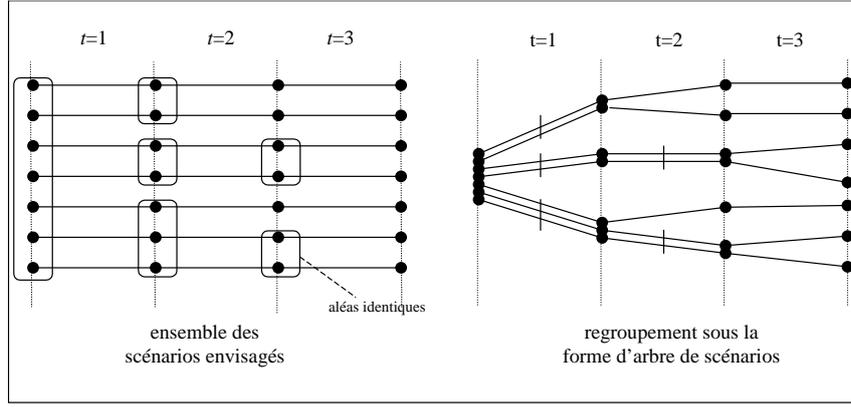


FIGURE 1.4.1 – Aggrégation de scénarios sous la forme d'un arbre

1.4 Discrétisation des problèmes multi-étapes

Le problème (1.17) ne peut être résolu qu'approximativement à cause de sa nature infini-dimensionnelle. En outre, dans beaucoup d'applications, on ne connaît pas la loi de ξ mais uniquement un ensemble de scénarios $\{\xi^{(1)}, \dots, \xi^{(S)}\}$ provenant de résultats d'expériences ou bien de relevés historiques. Dans ce cas, une façon naturelle d'approcher le problème est de remplacer ξ par une variable aléatoire discrète $\hat{\xi} = (\hat{\xi}_1, \dots, \hat{\xi}_T)$ prenant la valeur $\xi^{(s)}$ avec probabilité ρ_s ; $\sum_{s=1}^S \rho_s = 1$. Nous définissons alors l'analogue de π_t qui décrit le passé de $\hat{\xi}$:

$$\hat{\pi}_t = (\hat{\xi}_1, \dots, \hat{\xi}_t)$$

elle aussi discrète. Elle prend ses valeurs dans l'ensemble $\{\pi_t^{(1)}, \dots, \pi_t^{(S)}\}$ avec $\pi_t^{(s)} = (\xi_1^{(s)}, \dots, \xi_t^{(s)})$ par construction.

1.4.1 Arbres de scénario

Le processus $\hat{\xi} = (\hat{\xi}_1, \dots, \hat{\xi}_T)$ engendre lui aussi une filtration $\hat{\mathcal{F}}_1 \subset \dots \subset \hat{\mathcal{F}}_T$ que l'on peut employer pour exprimer la non-anticipativité. Le problème approché est alors le suivant :

$$\text{Minimiser}_{(\mathbf{X}_1, \dots, \mathbf{X}_T)} \mathbb{E} \left(f(\mathbf{X}_1, \dots, \mathbf{X}_T; \hat{\xi}_1, \dots, \hat{\xi}_T) \right) \quad (1.18a)$$

$$s.t \quad \mathbf{X}_1, \dots, \mathbf{X}_T \in L^\infty(\Omega, \mathcal{F}, \mu, \mathbb{R}^n) \quad (1.18b)$$

$$\mathbf{X}_t \text{ est } \hat{\mathcal{F}}_t\text{-mesurable} \quad (1.18c)$$

La nature discrète du processus $\hat{\xi}$ permet de représenter graphiquement l'évolution dans le temps de l'aléa sous la forme d'un *arbre de scénarios* (voir figure 1.4.1). La racine représente le début de la période 1 et chaque chemin reliant la racine à une feuille représente une suite possible de valeurs de $(\xi_1^{(s)}, \dots, \xi_T^{(s)}) = \xi^{(s)}$. Deux scénarios (s) et (s') sont donc identiques jusqu'à la période t , c'est à dire $\pi_t^{(s)} = \pi_t^{(s')}$ si les deux chemins les représentant ont leurs t premiers nœuds identiques. On dit alors qu'ils sont *indissociables* à la date t .

L'ensemble des vecteurs aléatoires $\hat{\mathcal{F}}_t$ -mesurables dans lequel nous cherchons \mathbf{X}_t est un espace vectoriel de dimension finie. En effet, \mathbf{X}_t est caractérisé par ses S images :

$$\begin{pmatrix} x_t^{(1)} \\ \vdots \\ x_t^{(S)} \end{pmatrix} = \begin{pmatrix} \varphi_t(\pi_t^{(1)}) \\ \vdots \\ \varphi_t(\pi_t^{(S)}) \end{pmatrix} \in \mathbb{R}^{nS} \quad (1.19)$$

où φ_t est la fonction de l'équation (1.16). La non-anticipativité s'exprime alors ainsi : si à la date t , le décideur ne peut pas distinguer deux scénarios (s) et (s') *i.e.* $\pi_t^{(s)} = \pi_t^{(s')}$, alors il doit prendre la même décision pour ces deux scénarios :

$$\pi_t^{(s)} = \pi_t^{(s')} \implies x_t^{(s)} = x_t^{(s')} \quad (1.20)$$

Nous arrivons alors au résultat attendu : le problème (1.18) est en fait un problème en dimension finie :

$$\min_{x_t^{(s)}} \sum_{s=1}^S \rho_s f(x^{(s)} ; \xi^{(s)}) \quad (1.21a)$$

$$\text{s.c. } x_t^{(s)} = x_t^{(s')} \quad \forall (s, s', t) / \pi_t^{(s)} = \pi_t^{(s')} \quad (1.21b)$$

Si $\hat{\xi}$ permet de bien approcher la loi de ξ , l'approximation de la structure d'information *i.e.* de la filtration engendrée est quant à elle catastrophique ([2]). En effet, dans les applications courantes la filtration $\{\hat{\mathcal{F}}_t\}_{t=1, \dots, T}$ décrit un *peigne* de scénarios qui ne reflète aucunement la structure diffuse de l'aléa, rendant le modèle anticipatif. Des techniques de *construction d'arbre* ([40] [45], [46, 99, 39, 117, 118]) consistantes au sens de l'épiconvergence ([86]) ont été développées afin de mieux approcher la structure d'information. Elles consistent essentiellement à perturber l'ensemble des scénarios $\{\xi^{(1)}, \dots, \xi^{(S)}\}$ afin d'améliorer la qualité de l'approximation de la filtration.

Voici deux freins à l'utilisation d'arbres. Comme nous venons de le mentionner, si nous disposons uniquement de scénarios *i.e.* nous n'avons pas de mécanisme de simulation des lois conditionnelles du processus ξ , alors un effort doit être produit pour transformer un paquet de scénarios en un arbre. Deuxièmement, lorsque l'on traite un programme stochastique multi-étapes, le décideur est souvent intéressé par la première décision mais aussi par les premières décisions de recours. Malheureusement, la topologie des arbres ne fournit généralement pas assez de nœuds à la première étape pour permettre de construire une approximation fonctionnelle (voir [23, 18] pour une discussion sur le sujet).

Nous proposons dans les sections suivantes deux méthodes qui ne requièrent pas de modification de l'échantillon de données comme dans les techniques basées sur les arbres de scénarios mais qui le gardent tel quel. Les contraintes de non-anticipativité seront prises en compte via un schéma de discrétisation basé sur des idées de l'estimateur de régression non-paramétrique de Nadaraya-Watson.

1.4.2 Estimation non-paramétrique

Nous disposons d'un échantillon $\{\xi^{(1)}, \dots, \xi^{(S)}\}$ et nous cherchons à déterminer une approximation ponctuelle de \mathbf{X}_t , $t = 1, \dots, T$, sous la forme (1.19). Le coût espéré peut alors être approché par :

$$\mathbb{E}(f(\mathbf{X} ; \xi)) \approx \mathbb{E}(f(\mathbf{X} ; \hat{\xi})) = \sum_{i=1}^S \rho_i f(x^{(i)} ; \xi^{(i)}) \quad (1.22)$$

Les contraintes informationnelles (1.17c) peuvent se réécrire de manière analytique à l'aide de l'opérateur d'espérance conditionnelle :

$$\underset{(\mathbf{X}_1, \dots, \mathbf{X}_T)}{\text{Minimiser}} \quad \mathbb{E}(f(\mathbf{X}_1, \dots, \mathbf{X}_T; \boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_T)) \quad (1.23a)$$

$$s.c \quad \mathbf{X}_1, \dots, \mathbf{X}_T \in L^\infty(\Omega, \mathcal{F}, \mu, \mathbb{R}^n) \quad (1.23b)$$

$$\mathbf{X}_t = \mathbb{E}(\mathbf{X}_t | \boldsymbol{\pi}_t) \quad (1.23c)$$

Pour un couple donné de vecteurs aléatoires (\mathbf{A}, \mathbf{B}) à valeurs dans $\mathbb{R}^n \times \mathbb{R}^m$, l'estimateur à noyau proposé par Watson ([123]) et Nadaraya ([79]) s'inscrit en estimation non-paramétrique, comme un outil rodé pour estimer la fonction de régression $g(b) = \mathbb{E}(\mathbf{A} | \mathbf{B} = b)$. Il se base sur la connaissance d'un échantillon de variables aléatoires :

$$(\mathbf{A}^{(1)}, \mathbf{B}^{(1)}), \dots, (\mathbf{A}^{(S)}, \mathbf{B}^{(S)})$$

indépendantes et identiquement distribuées de même loi que le couple (\mathbf{A}, \mathbf{B}) .

Si l'on choisit une fonction noyau $K : \mathbb{R}^m \rightarrow \mathbb{R}^+$ et $h_S > 0$ une *largeur de bande* dépendant de S , l'estimateur est défini comme la fonction aléatoire :

$$g_S((\mathbf{A}^{(1)}, \mathbf{B}^{(1)}), \dots, (\mathbf{A}^{(S)}, \mathbf{B}^{(S)}); b) = \frac{\sum_{i=1}^S \mathbf{A}^{(i)} K\left(\frac{\mathbf{B}^{(i)} - b}{h_S}\right)}{\sum_{i=1}^S K\left(\frac{\mathbf{B}^{(i)} - b}{h_S}\right)} \approx g(b)$$

où $0/0$ vaut 0. Les propriétés de convergence de cet estimateur ont été largement étudiées par Devroye et al. [29, 28]. Nous mentionnons en particulier la propriété de *consistance universelle* prouvée par Devroye et Wagner [29] et séparément par Spigleman et Sacks [112] :

Théorème 1.4.1. *Supposons qu'il existe des constantes $R_l, R_u, c_l, c_u > 0$ pour lesquelles le noyau K satisfait :*

$$c_l \mathbb{I}_{B(0, R_l)} \leq K \leq c_u \mathbb{I}_{B(0, R_u)}$$

et que $h \rightarrow 0, Sh^m \rightarrow 0$. Alors, pour tout $p \geq 1$ tel que $\|\mathbf{A}\|_{L^p} < +\infty$:

$$\mathbb{E}\left(\left\|g_S((\mathbf{A}^{(1)}, \mathbf{B}^{(1)}), \dots, (\mathbf{A}^{(S)}, \mathbf{B}^{(S)}); \mathbf{B}) - \mathbb{E}(\mathbf{A} | \mathbf{B})\right\|_{L^p}^p\right) \rightarrow 0$$

où l'espérance extérieure est prise par rapport au tirage de l'échantillon.

D'après la caractérisation de la mesurabilité grâce à l'espérance conditionnelle, nous obtenons le corollaire :

Corollaire 1.4.1. *\mathbf{A} est $\sigma(\mathbf{B})$ -mesurable si et seulement si :*

$$\mathbb{E}\left(\left\|g_S((\mathbf{A}^{(1)}, \mathbf{B}^{(1)}), \dots, (\mathbf{A}^{(S)}, \mathbf{B}^{(S)}); \mathbf{B}) - \mathbf{A}\right\|_{L^p}^p\right) \rightarrow 0$$

Démonstration. C'est une conséquence directe du fait que \mathbf{A} est $\sigma(\mathbf{B})$ -mesurable si et seulement si $\mathbf{A} = \mathbb{E}(\mathbf{A} | \mathbf{B})$ et de l'unicité de la limite dans le théorème 1.4.1. \square

Ce résultat est à l'origine des schémas de discrétisation que nous proposons par la suite. Etant donné un échantillon $(\xi^{(1)}, \dots, \xi^{(S)})$, et l'échantillon de passés correspondant $(\pi_t^{(1)}, \dots, \pi_t^{(S)})$ pour $t = 1, \dots, T$, nous approchons le membre de droite de la contrainte (1.23c) par :

$$\mathbb{E}(\mathbf{X}_t | \boldsymbol{\pi}_t = \pi_t) \approx g_S \left((x_t^{(1)}, \pi_t^{(1)}), \dots, (x_t^{(S)}, \pi_t^{(S)}); \pi_t \right) \quad (1.24)$$

où les $x_t^{(s)}$ sont les variables de décision du problème discrétisé qui constituent l'approximation ponctuelle de \mathbf{X}_t . Maintenant, afin d'approcher entièrement la contrainte (1.23c), une idée immédiate (mais naïve) serait d'ajouter les $S \times T$ égalités suivantes :

$$x_t^{(s)} = g_S \left((x_t^{(1)}, \pi_t^{(1)}), \dots, (x_t^{(S)}, \pi_t^{(S)}); \pi_t^{(s)} \right) \approx \mathbb{E} \left(\mathbf{X}_t | \boldsymbol{\pi}_t = \pi_t^{(s)} \right) \quad \forall s, \forall t \quad (1.25)$$

Cette discrétisation est en fait catastrophique car l'approximation résultante est inévitablement constante, ou dans le meilleur des cas, constante sur les parties connexes de $\text{supp}(\boldsymbol{\pi}_t)$. Pour mieux visualiser ce fait, notons :

$$\alpha_t^{(s,s')} = \frac{K \left(\frac{\pi_t^{(s)} - \pi_t^{(s')}}{h_S} \right)}{\sum_{s''=1}^S K \left(\frac{\pi_t^{(s)} - \pi_t^{(s'')}}{h_S} \right)} \quad \text{et} \quad A_t^{(s,s')} = \alpha_t^{(s,s')} I_n$$

avec I_n la matrice identité de rang n . L'égalité (1.25) se réécrit $x_t^{(s)} = \sum_{s'=1}^S \alpha_t^{(s,s')} x_t^{(s')}$ ou $x_t = A_t x_t$ avec $A_t = (A_t^{(s,s')})_{s,s'}$ définie par bloc. Deux scénarios s et s' sont dit *voisins* si $\alpha_t^{(s,s')} > 0$. En rassemblant les scénarios $1 \leq s, s' \leq N$ connectés par un chemin de voisins, nous obtenons la relation d'équivalence Δ_t :

$$s \Delta_t s' \iff \begin{cases} \exists k_1, k_2, \dots, k_p \in \{1, \dots, S\} \\ k_1 = s, k_p = s' \\ \forall \nu = 1, \dots, p-1 \quad \alpha_t^{(k_\nu, k_{\nu+1})} > 0 \end{cases}$$

Supposons pour le moment que $n = 1$; qu'il n'y ait qu'une seule classe *i.e.* la matrice $A = (\alpha_t^{(s,s')})_{s,s'}$ est irréductible. Cela arrive par exemple si le noyau $K(\cdot)$ est choisi de support \mathbb{R} tout entier, comme un noyau gaussien par exemple. Notons \mathcal{E} le sous-espace défini par le système de contraintes (1.25) et montrons qu'il est égal à l'espace \mathcal{E}' des vecteurs dont les composantes sont identiques les unes aux autres :

$$\mathcal{E}' = \left\{ x_t / x_t^{(1)} = \dots = x_t^{(S)} \right\}$$

Clairement, $\mathcal{E}' \subset \mathcal{E}$ car A calcule une moyenne pondérée. Maintenant, A est une matrice stochastique, donc son rayon spectral vaut 1. L'espace propre associé est par conséquent de dimension 1, par l'irréductibilité de A et par le théorème de Perron-Frobenius. Cet espace propre coïncide avec \mathcal{E} par définition et doit par conséquent se réduire à \mathcal{E}' puisqu'il est lui-même de dimension 1. Ce raisonnement s'étend à la dimension n en raisonnant composante par composante.

La convergence des estimateurs requiert généralement que chaque scénario ait beaucoup de voisins. Par conséquent, les classes d'équivalence correspondront aux parties connexes de $\text{supp}(\boldsymbol{\pi}_t)$.

1.4.3 Pénalisation de l'anticipativité

Nous venons de mettre en évidence qu'imposer l'égalité ponctuelle entre l'estimation de la fonction de régression et les décisions $(x_t^{(1)}, \dots, x_t^{(S)})$ amenait une très mauvaise approximation de l'espace des fonctions \mathcal{F}_t -mesurables. La première idée que nous développons consiste à relâcher légèrement ces T ensembles de N égalités en les pénalisant. Si $P(\cdot)$ est une fonction de pénalisation, à titre d'exemple :

$$P(x) = \frac{C}{2} \|x\|_2^2 \quad P(x) = C \|x\|_1 \quad P(x) = \begin{cases} 0 & \text{si } \|x\|_\infty \leq \epsilon \\ +\infty & \text{sinon.} \end{cases} \quad (1.26)$$

nous proposons de résoudre le programme suivant en dimension finie :

$$\text{Minimiser}_{x_t^{(s)}} \sum_s \rho_s f(x_1^{(s)}, \dots, x_T^{(s)}; \xi^{(1)}, \dots, \xi^{(S)}) \quad (1.27a)$$

$$+ \sum_{s,t} P\left(x_t^{(s)} - g_S((x_t^{(1)}, \pi_t^{(1)}), \dots, (x_t^{(S)}, \pi_t^{(S)}); \pi_t^{(s)})\right) \quad (1.27b)$$

Nous testons la validité numérique de cette approche en l'appliquant à un problème de gestion d'une usine hydraulique. Deux décisions de production successives $p_1 \geq 0$ et $p_2 \geq 0$ doivent être prises. Le réservoir contient initialement une quantité d'énergie l_0 , de sorte que nous imposons $p_1 + p_2 \leq l_0$. Ces décisions doivent être prises après avoir observé des prix de ventes aléatoires ξ_1 et ξ_2 . La décision p_1 est soumise à une contrainte de non-anticipativité, elle doit être prise avant de connaître ξ_2 . Le coût de production est le suivant :

$$f(p_1, p_2; \xi_1, \xi_2) = -[\xi_1 p_1 + \xi_2 p_2 + V(l_0 - p_1 - p_2)]$$

où $V(\cdot)$ est la fonction de valorisation du stock final à la fin des deux périodes. Nous la choisissons ici quadratique :

$$V(x) = ax^2 + bx + \eta$$

avec $a = -0.3683$, $b = 1.1009$ et $\eta = 0.3162$. ξ_1 et ξ_2 suivent des lois uniformes indépendantes sur l'intervalle $[0.4, 2]$ et $l_0 = 1$. Notre problème d'optimisation est par conséquent :

$$\begin{aligned} J &= \min_{p_1, p_2} \mathbb{E}(-p_1 \xi_1 - p_2 \xi_2 - V(l_0 - p_1 - p_2)) \\ \text{s.c} \quad p_1 &\in [0, l_0] && \text{presque sûrement} \\ p_2 &\in [0, l_0 - p_1] && \text{presque sûrement} \end{aligned}$$

Ce problème peut être résolu avec une grande précision par programmation dynamique. Nous comparons cette solution de référence avec l'approximation obtenue en utilisant une fonction de pénalisation quadratique. Soit $\left\{(\xi_1^{(1)}, \xi_2^{(1)}), \dots, (\xi_1^{(S)}, \xi_2^{(S)})\right\}$ un ensemble de réalisations des prix. Le problème approché est le suivant :

$$\tilde{J}_N = \min_{(p_1^{(s)}, p_2^{(s)})} \frac{1}{S} \sum_{s=1}^S f(p_1^{(s)}, p_2^{(s)}, \xi_1^{(s)}, \xi_2^{(s)}) + \frac{c}{S} \sum_{s=1}^S \left\| p_1^{(s)} - \frac{\sum_{s' \neq s} K \left(\frac{\xi_1^{(s)} - \xi_1^{(s')}}{h_1} \right) p_1^{(s')}}{\sum_{s' \neq s} K \left(\frac{\xi_1^{(s)} - \xi_1^{(s')}}{h_1} \right)} \right\|^2$$

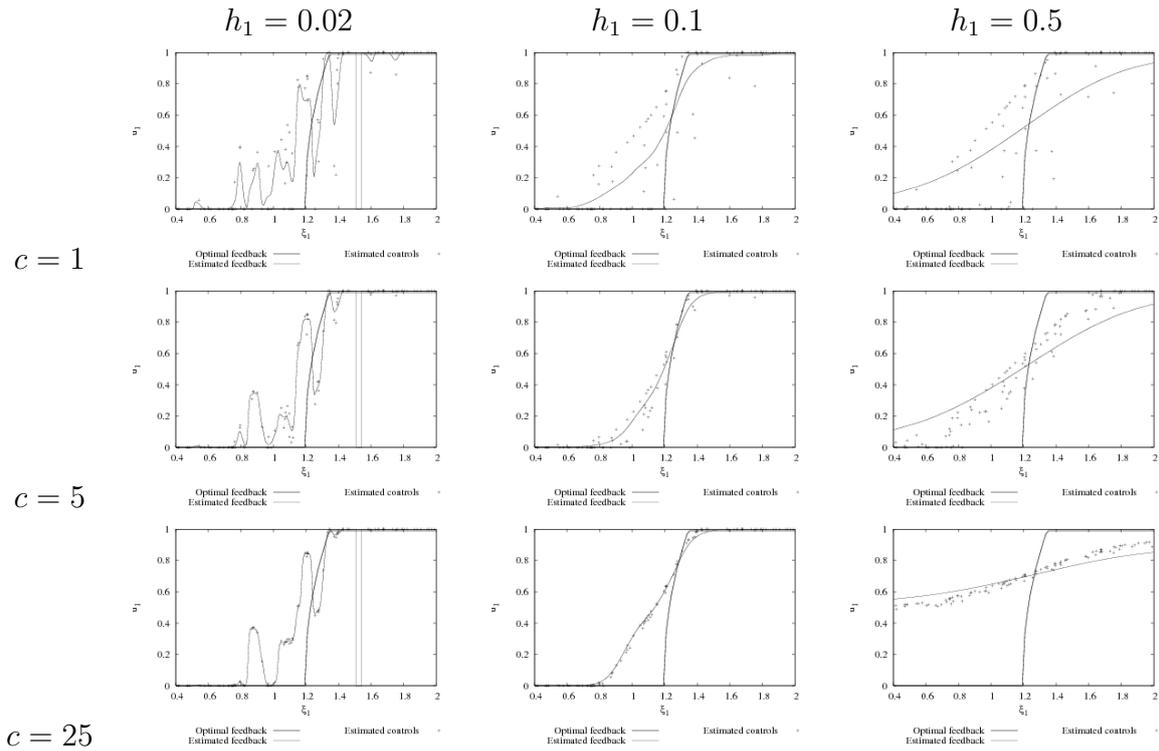


FIGURE 1.4.2 – Approximation de $\varphi_1(\cdot)$ pour $S = 100$ scénarios en fonction de la largeur de bande h_1 et de la pénalité c .

où nous avons choisi comme noyau la fonction $K(\cdot) = e^{-\|\cdot\|^2}$. Une fois ce problème résolu, nous synthétisons les stratégies en utilisant :

$$\varphi_1(\xi_1) = \frac{\sum_{s=1}^S K\left(\frac{\xi_1^{(s)} - \xi_1}{h_1}\right) p_1^{(s)}}{\sum_{s=1}^S K\left(\frac{\xi_1^{(s)} - \xi_1}{h_1}\right)}$$

$$\varphi_2(\xi_1, \xi_2) = \frac{\sum_{s=1}^S K\left(\frac{(\xi_1^{(s)}, \xi_2^{(s)}) - (\xi_1, \xi_2)}{h_2}\right) p_2^{(s)}}{\sum_{s=1}^S K\left(\frac{(\xi_1^{(s)}, \xi_2^{(s)}) - (\xi_1, \xi_2)}{h_2}\right)}$$

où nous avons choisi empiriquement $h_2 = \sqrt{\frac{h_1}{\pi}}$. Les figures 1.4.2 et 1.4.3 présentent les politiques obtenues pour différentes valeurs de c et de h_1 . La qualité de ces approximations est évaluée par une simulation de Quasi-Monte-Carlo. Les coûts obtenus pour $S = 100$ scénarios sont reportés dans le tableau 1.1. Ces résultats doivent être comparés

- à la valeur optimale $J = -1.7414$. La figure 1.4.5 présente la politique optimale $p_2(\cdot, \cdot)$ (la politique $p_1(\cdot)$ est tracée sur chaque graphe représentant une approximation de $p_1(\cdot)$).
- à la valeur -1.786 obtenue sans le terme de pénalisation *i.e.* sans modéliser la non-anticipativité
- à la valeur -1.69563 de la solution obtenue en synthétisant des feedbacks comme

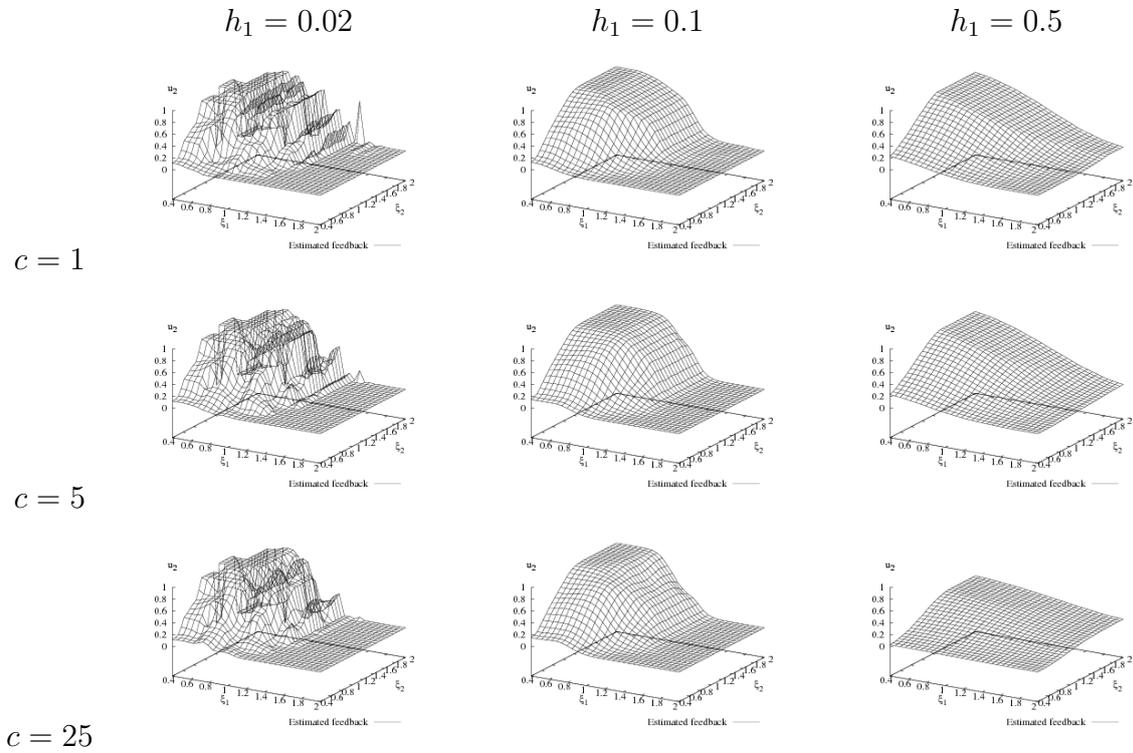


FIGURE 1.4.3 – Approximation de $\varphi_2(\cdot, \cdot)$ pour $S = 100$ scénarios en fonction de la largeur de bande h_1 et de la pénalité c .

| | $h_1 = 0.02$ | $h_1 = 0.1$ | $h_1 = 0.5$ |
|----------|--------------|-------------|-------------|
| $C = 1$ | -1.6788 | -1.72698 | -1.67559 |
| $C = 5$ | -1.67757 | -1.73394 | -1.66757 |
| $C = 25$ | -1.67638 | -1.72863 | -1.58707 |

TABLE 1.1 – Coût de production des stratégies approchées pour $S = 100$ scénarios en fonction de la largeur de bande h_1 et du paramètre de pénalité c .

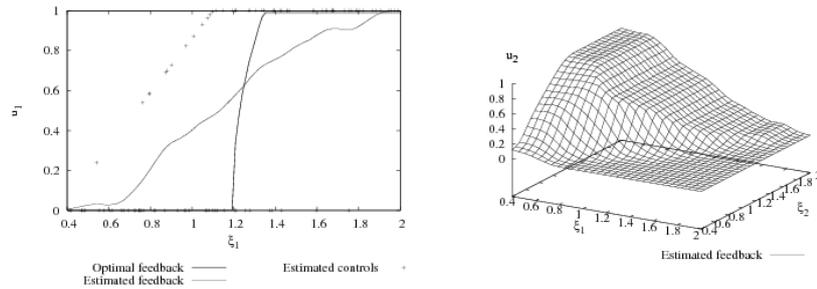


FIGURE 1.4.4 – Approximations de $\varphi_1(\cdot)$ et $\varphi_2(\cdot, \cdot)$, pour $S = 100$ scénarios, avec une largeur de bande $\epsilon_1 = 0.1$ et sans pénalité *i.e.* $c = 0$.

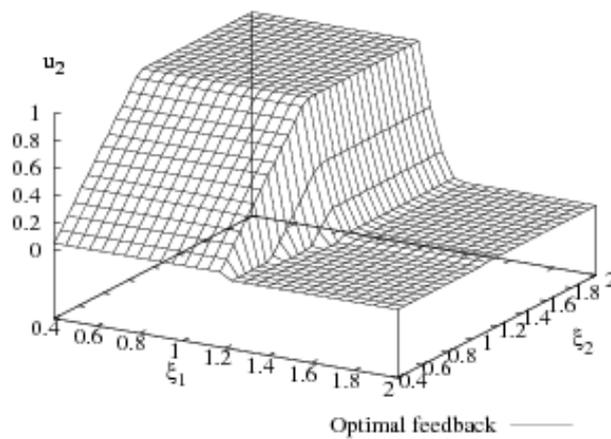


FIGURE 1.4.5 – Politique optimale $\varphi_2(\cdot, \cdot)$

décrit ci-dessus à partir de la solution anticipative avec $\epsilon_1 = 0.1$. Cette solution est représentée sur la figure 1.4.4.

Les paramètres h_1 et c peuvent être choisis de manière à optimiser le coût durant la simulation. Les figures 1.4.6 et 1.4.7 présentent les meilleures approximations obtenues pour $S = 10, 27, 129$ et 999 . Le tableau 1.2 résume la qualité de ces approximations, ainsi que les paramètres choisis. Il semble que le coût optimal ainsi que les politiques convergent vers les solutions exactes lorsque le nombre de scénarios disponible croît.

| S | h_1 | c | Coût |
|-----|-----------|---------|----------|
| 10 | 0.215443 | 5.99484 | -1.70561 |
| 27 | 0.129155 | 2.15443 | -1.72187 |
| 129 | 0.0774264 | 5.99484 | -1.73369 |
| 999 | 0.016681 | 1000 | -1.74018 |

TABLE 1.2 – Coût optimal et valeurs de h_1 et c fournissant les meilleures solutions pour $S = 10, 27, 129$ et 999 .

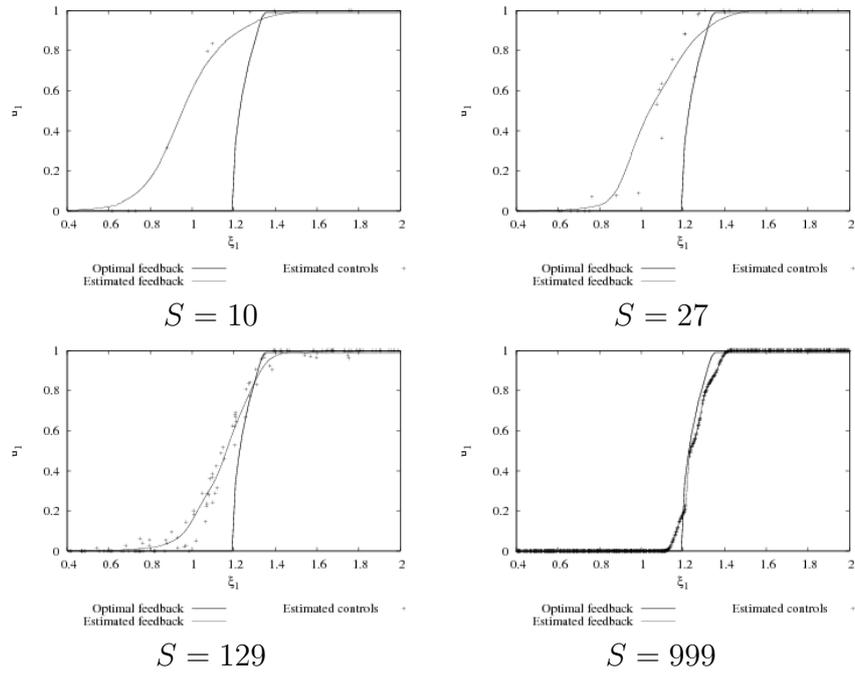


FIGURE 1.4.6 – Meilleure approximation de $\varphi_1(\cdot)$ pour $S = 10, 27, 129$ et 999 .

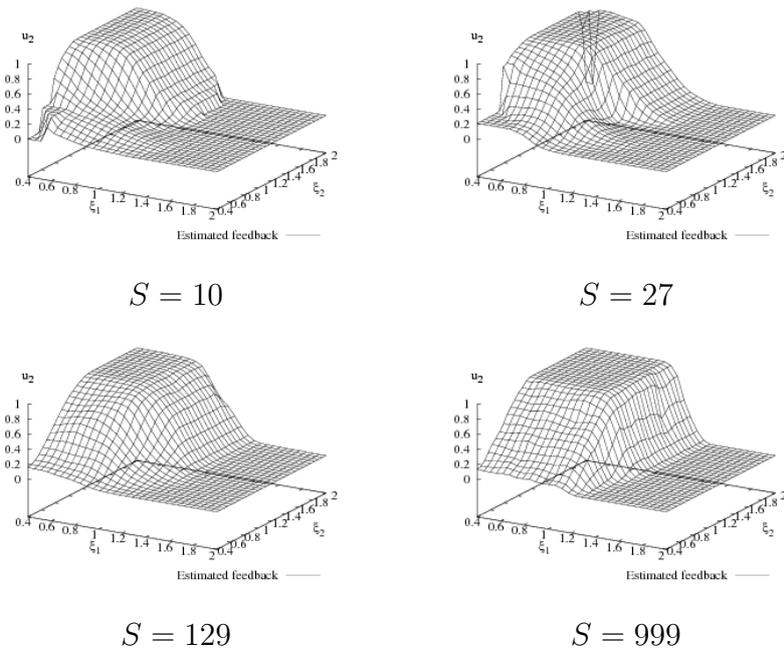


FIGURE 1.4.7 – Meilleure approximation de $\varphi_2(\cdot, \cdot)$ pour $S = 10, 27, 129$ and 999 .

1.4.4 Partition de l'unité

Nous adoptons maintenant un autre point de vue sur les estimateurs non-paramétriques en interprétant la fonction g_S dans (1.24) comme la somme de S fonctions de bases pondérées par des coefficients $x_t^{(s)}$. Cette base contient pour t donné, les fonctions $\psi_t^{(s)}$ définies par :

$$\psi_t^{(s)}(\cdot) = \frac{K\left(\frac{\pi_t^{(s)} - \cdot}{h_S}\right)}{\sum_{s'=1}^S K\left(\frac{\pi_t^{(s')} - \cdot}{h_S}\right)} \quad (1.28)$$

et nous nous intéressons aux fonctions $\psi_t : \mathbb{R}^{m_t} \rightarrow \mathbb{R}^n$ sous la forme $\psi_t(\cdot) = \sum_{s=1}^S \psi_t^{(s)}(\cdot) v_t^{(s)}$. Cette famille définit une partition de l'unité dans le sens où :

$$\sum_{s=1}^S \psi_t^{(s)} \equiv 1 \quad \text{sur} \quad \bigcup_{s=1}^S \left\{ \pi_t^{(s)} + h_S \text{supp}(K) \right\}$$

Nous noterons $\Psi_t \cdot v_t$ l'élément de cet espace de coefficients $(v_t^{(1)}, \dots, v_t^{(S)}) = v_t \in \mathbb{R}^{n_S}$. Alors :

$$(\Psi_t \cdot v_t)(\pi_t^{(s)}) = g_S\left((v_t^{(1)}, \pi_t^{(1)}), \dots, (v_t^{(S)}, \pi_t^{(S)}); \pi_t^{(s)}\right) \quad (1.29a)$$

$$= \sum_{s'=1}^S \left[\psi_t^{(s')}(\pi_t^{(s)}) \right] v_t^{(s')} \quad (1.29b)$$

Au lieu d'essayer de relier les valeurs ponctuelles de $\psi_t = \Psi_t \cdot v_t$ avec les valeurs de ses coefficients comme dans la section précédente, nous les différencions en introduisant de nouvelles variables. Pour $s = 1, \dots, S$ et $t = 1, \dots, T$, nous posons $x_t^{(s)} = (\Psi_t \cdot v_t)(\pi_t^{(s)})$ et en utilisant (1.29b), nous proposons de résoudre le problème suivant :

$$\text{Minimiser} \sum_{s=1}^S \rho_i f(x_1^{(s)}, \dots, x_T^{(s)}; \xi^{(s)}) \quad (1.30a)$$

$$x_t^{(s)} = \sum_{s'=1}^S \left[\psi_t^{(s')}(\pi_t^{(s)}) \right] v_t^{(s')} \quad (1.30b)$$

Il est à noter que cette approche n'est pas nouvelle. L'utilisation d'une partition de l'unité pour le calcul numérique d'intégrales a déjà été utilisé ([1, section 2]) pour résoudre des équations aux dérivées partielles et pour l'intégration en régression. Nous présentons maintenant certains faits sur la consistance de cette approche.

Nous omettons l'indice de l'étape t et posons $\Pi = \boldsymbol{\pi}(\Omega)$ en le munissant de la mesure transportée $\lambda = \mu \circ \boldsymbol{\pi}^{-1}$ et de sa tribu borellienne. Ensuite, nous considérons une suite $\{\lambda_S\}_S$ de mesures sur Π de supports atomiques $\{\pi^{(1)}, \dots, \pi^{(S)}\}$. La probabilité assignée à chaque atome importe peu ici puisqu'on ne s'intéresse pas à l'évaluation du coût mais uniquement à la structure des fonctions de base $\psi^{(s)}$ qui ne dépendent que de la localisation des atomes dans l'espace.

Pour tout S , une largeur de bande h_S est choisie et nous définissons l'ensemble :

$$E_S = \left\{ \psi = \sum_{s=1}^S v^{(s)} \psi^{(s)}, (v^{(1)}, \dots, v^{(S)}) \in \mathbb{R}^{n_S} \right\}$$

qui est l'espace de dimension finie des approximations. Nous nous intéressons en particulier aux fonctions qui peuvent être représentées par des suites de $\{E_S\}_S$. La convergence sera étudiée pour la norme L^p avec $1 \leq p < \infty$ sous les hypothèses suivantes :

- i) $\lim_{S \rightarrow \infty} h_S = 0$;
- ii) $\lim_{S \rightarrow \infty} \lambda \left(\bigcup_{s=1}^S \{ \pi^{(s)} + h_S \text{supp}(K) \} \right) = 1$;
- iii) il existe $c_l, R_l, c_u, R_u > 0$ tels que K satisfait les hypothèses du théorème 1.4.1.

Lemme 1.4.1. *Soit l définie sur Π , continue à support compact. Alors sous i, ii, iii, il existe une suite $\{l_S\}_S$; $l_S \in E_S$ convergeant vers l pour la norme L^p .*

Démonstration. Construisons la suite $\{l_S\}_S$ définie pour tout S :

$$l_S(\pi) = \sum_{s=1}^S l(\pi^{(s)}) \psi^s(\pi)$$

Choisissons $\epsilon > 0$. En notant $A_S = \bigcup_{s=1}^S \{ \pi^{(s)} + h_S \text{supp}(K) \}$ l'ensemble de l'hypothèse ii, nous pouvons alors choisir S assez grand pour avoir :

- a) h_S est si petit que $\sup_{\|\pi - \pi'\| < R_u h_S} \|l(\pi) - l(\pi')\| < \epsilon^{\frac{1}{p}}$.
- b) $\lambda(A_S) \geq 1 - \frac{\epsilon}{(\|l\|_\infty)^p}$

Pour chaque π , nous avons :

$$\|l_S(\pi) - l(\pi)\| = \left\| \frac{1}{\sum_{i=1}^S K \left(\frac{\pi^{(i)} - \pi}{h_S} \right)} \sum_{s=1}^S l(\pi^{(s)}) K \left(\frac{\pi^{(s)} - \pi}{h_S} \right) - l(\pi) \right\| \quad (1.31)$$

Les éléments de la somme sont non nuls pour les indices s tels que $\frac{\pi^{(s)} - \pi}{h_S} \in \text{supp}(K)$. S'il n'y a pas de tel s , cela signifie que $\pi \notin A_S$ et que la somme est nulle. Sinon $\pi \in A_S$ et pour chaque terme non nul, nous avons $\|\pi^{(s)} - \pi\| \leq R_u h_S$ d'après iii donc $\|l(\pi^{(s)}) - l(\pi)\| < \epsilon^{\frac{1}{p}}$ d'après notre choix de h_S . Maintenant, la somme dans (1.31) correspond à une combinaison convexe de points de $B(l(\pi), \epsilon^{\frac{1}{p}})$, la boule ouverte centrée en $l(\pi)$ et de rayon $\epsilon^{\frac{1}{p}}$ donc le résultat est aussi dans cette boule d'où :

$$\|l_S(\pi) - l(\pi)\|^p < \epsilon$$

Maintenant, nous pouvons intégrer cette inégalité :

$$\begin{aligned} \int \|l_S(\pi) - l(\pi)\|^p &= \int_{A_S} \|l_S(\pi) - l(\pi)\|^p + \int_{\Pi \setminus A_S} \|l_S(\pi) - l(\pi)\|^p \\ &\leq \lambda(A_S) \epsilon + \lambda(\Pi \setminus A_S) \|l\|_\infty^p \\ &\leq \epsilon + \epsilon \end{aligned}$$

La conclusion vient du fait que ceci est vrai pour tout ϵ . □

Nous inférons de ce résultat un corollaire similaire à la consistance forte universelle du théorème 1.4.1 :

Corollaire 1.4.2. *Pour tout $l \in L^p(\Omega, \mathcal{F}, \mu)$. Il existe une suite $\{l_S\}_S$; $l_S \in E_S$ convergeant vers l dans L^p .*

Démonstration. Ceci est direct d'après la densité de l'ensemble des fonctions continues à support compact dans L^p . □

1.4.5 Couplages des aléas

Les problèmes d'optimisation stochastique discrétisés (1.21), (1.27), (1.30), font apparaître les scénarios comme sous-systèmes naturels. Ils sont couplés par les contraintes issues de la discrétisation de la non-anticipativité. Ainsi ces trois problèmes peuvent être réécrits sous la forme générique :

$$\text{Minimiser}_{x^{(s)}} \sum_{s=1}^S f_{(s)}(x^{(s)}) + C(x^{(1)}, \dots, x^{(S)})$$

avec $f_{(s)}(x^{(s)}) = \rho_s f(x_1^{(s)}, \dots, x_T^{(s)}; \xi^{(s)})$ et $C(x^{(1)}, \dots, x^{(S)})$ égal, selon le schéma de discrétisation choisi, à :

pour (1.21) :

$$C(x^{(1)}, \dots, x^{(S)}) = \begin{cases} 0 & \text{si } x_t^{(s)} = x_t^{(s')} \quad \forall (s, s', t) / \pi_t^{(s)} = \pi_t^{(s')} \\ +\infty & \text{sinon.} \end{cases}$$

pour (1.27) :

$$C(x^{(1)}, \dots, x^{(S)}) = \sum_{s,t} P \left(x_t^{(s)} - g_s((x_t^{(1)}, \pi_t^{(1)}), \dots, (x_t^{(S)}, \pi_t^{(S)}); \pi_t^{(s)}) \right)$$

pour (1.30) :

$$C(x^{(1)}, \dots, x^{(S)}) = \begin{cases} 0 & \text{si } x_t^{(s)} = \sum_{s'=1}^S [\psi_t^{(s')}(\pi_t^{(s)})] v_t^{(s')} \quad \forall (s, s', t) \\ +\infty & \text{sinon.} \end{cases}$$

Conclusion

Nous avons mis en évidence les trois couplages naturels du problème de gestion de production en environnement incertain, déjà étudiés dans [71] :

- en espace : les décisions sur les différentes usines sont liées par le fait de devoir satisfaire la demande
- en temps : les décisions sur les différents pas de temps sont liées par des contraintes dynamiques de fonctionnement des usines
- en aléa : les décisions sur les différents scénarios sont liées par la nécessité de ne pas anticiper le futur

Le problème présente ainsi la structure :

$$\min_{p_{u,t}^{(s)}} \sum_{u,t,s} f_{u,t}^{(s)}(p_{u,t}^{(s)}) \tag{1.32a}$$

$$+ \sum_{u,t} C_{u,t}(p_{u,t}^{(1)}, \dots, p_{u,t}^{(S)}) \tag{1.32b}$$

$$+ \sum_{t,s} C_t^{(s)}(p_{1,t}^{(s)}, \dots, p_{U,t}^{(s)}) \tag{1.32c}$$

$$+ \sum_{u,s} C_u^{(s)}(p_{u,1}^{(s)}, \dots, p_{u,T}^{(s)}) \tag{1.32d}$$

où $f_{u,t}^{(s)}$ représente le coût de l'usine u au pas de temps t du scénario s ; $C_{u,t}(\cdot)$ le couplage en aléa, $C_t^{(s)}(\cdot)$ le couplage spatial et $C_u^{(s)}(\cdot)$ le couplage temporel. Nous disposons ainsi de

plusieurs visions du système global, chacune basée sur une définition différente de sous-système : usine, période, scénario ou bien des intersections usine-période, usine-scénario, période-scénario et enfin usine-période-scénario.

Cette énumération des couplages permet d'envisager de nouvelles façons de décomposer le problème. En effet, jusqu'ici la démarche communément adoptée a consisté à briser le couplage spatial [90, 85, 80, 27, 101] conjointement ou non au couplage temporel grâce à des approches de type programmation dynamique [119, 100] ou bien uniquement le couplage en aléas [97, 105, 106]. L'éventualité de sous-systèmes de type usine-scénario ou usine-période-scénario n'a jamais été testée à notre connaissance.

Nous avons aussi proposé deux nouvelles formulations du couplage en aléa. Si elles présentent l'avantage combiné d'éviter la construction d'un arbre de scénarios et de fournir une meilleure approximation des solutions relatives aux premières périodes, elles couplent en contrepartie plus fortement les scénarios, laissant penser que la décomposition en sera d'autant plus difficile.

Comme annoncé en introduction, le but sous-jacent à cette analyse des couplages est la mise en place de méthodes de décomposition traitées dans les chapitres 3 et 4 permettant de faire face à des problèmes de grande taille en tirant profit de leur structure. Le caractère non différentiable du modèle constitue une difficulté préliminaire, qu'il convient de contourner à l'aide de techniques de régularisation. Le chapitre 2 suivant est dédié à ces procédures.

Chapitre 2

Résolution d'inclusions monotones par régularisation

2.1 Introduction

Nous posons dans ce chapitre le cadre théorique de notre étude : celui de l'optimisation convexe non différentiable. Les fonctions F et C sont donc supposées convexes dans le problème (1.2) qui peut par conséquent être laissé de côté (voir annexe A) pour un problème équivalent de recherche de zéros d'un opérateur T monotone maximal :

$$\text{Trouver } y^* \in \mathbb{R}^n \text{ tel que } 0 \in T(y^*)$$

Les méthodes itératives permettant de chercher un tel y^* peuvent être classées en deux grandes familles. La première regroupe les généralisations de la méthode du gradient, consistant à l'itération k à :

- calculer en un point $y^k \in \mathbb{R}^n$ un vecteur $\tilde{u}^k \approx u^k \in T(y^k)$
- choisir un pas $\rho^k > 0$
- et effectuer l'équivalent d'un pas de descente $y^{k+1} = y^k - \rho^k \tilde{u}^k$

Le deuxième type de méthode effectue la même opération mais de manière implicite. Autrement dit, l'itérée y^{k+1} vérifie :

$$\begin{aligned} y^{k+1} &= y^k - \rho^k \tilde{u}^{k+1} \\ \tilde{u}^{k+1} &\approx u^{k+1} \in T(y^{k+1}) \end{aligned}$$

Les versions exactes de ces deux méthodes (obtenues en remplaçant l'approximation par une égalité) sont développées en détail dans la section 2.2. Elles peuvent être vues toutes les deux comme des cas particuliers d'un algorithme de point fixe 2.2.3 appliqué à un opérateur de *réflexion généralisée*. Afin d'étudier la convergence de cet algorithme, nous creusons les propriétés de cette classe d'opérateurs dans la section 2.3.

2.2 Algorithmes de points fixes pour la recherche de zéros

2.2.1 Algorithme forward

Pour une suite de pas $\rho_k > 0$, la méthode du gradient appliquée à la minimisation d'une fonction différentiable F (algorithme 2.2.1) se généralise naturellement à la recherche

Algorithme 2.2.1 Algorithme du gradient

1. $k = 0$; Choisir $y^0 \in \mathbb{R}^n$
2. $y^{k+1} = y^k - \rho_k \nabla F(y^k)$
3. $k \leftarrow k + 1$; Aller à l'étape 2

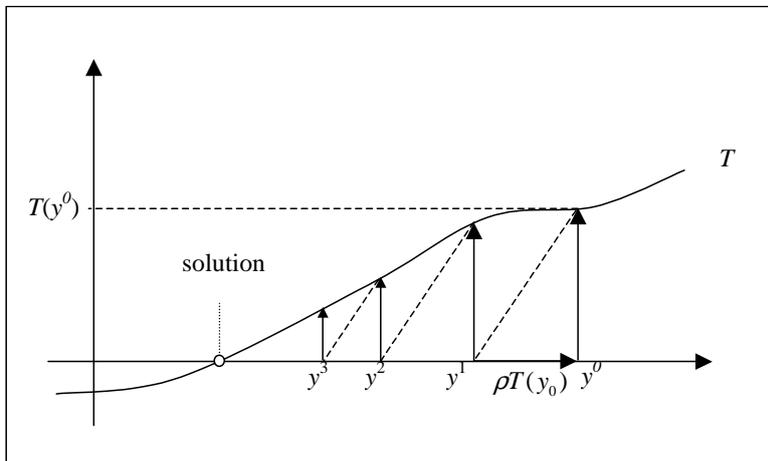


FIGURE 2.2.1 – Itérations de l'algorithme de gradient généralisé.

de zéros d'opérateurs monotones $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ à travers l'algorithme *forward* 2.2.2. Nous représentons graphiquement la suite des itérés sur la figure 2.2.1 pour un opérateur

Algorithme 2.2.2 Algorithme forward

1. $k = 0$; Choisir $y^0 \in \mathbb{R}^n$
2. $y^{k+1} = y^k - \rho_k T(y^k)$
3. $k \leftarrow k + 1$; Aller à l'étape 2

$T : \mathbb{R} \rightarrow \mathbb{R}$. Le passage de y^k à y^{k+1} s'effectue en appliquant l'opérateur $I - \rho_k T$ qui se réécrit aussi pour tout $\nu > 0$:

$$I - \rho_k T = (1 - \frac{\rho_k}{\nu})I + \frac{\rho_k}{\nu}(I - \nu T)$$

Les zéros de T sont les points fixes de l'opérateur $N \stackrel{def}{=} I - \nu T$. Ainsi, si nous posons $\alpha_k \stackrel{def}{=} 1 - \frac{\rho_k}{\nu}$, l'algorithme 2.2.2 peut être interprété comme des applications successives de l'opérateur N avec un facteur de sous-relaxation α_k (algorithme 2.2.3) dont le but est de trouver $\bar{s} \in \text{fix}(N)$. Les théorèmes généraux présentés en annexe B permettent d'établir des conditions simples sur T , ν et ρ^k pour obtenir la convergence.

Proposition 2.2.1. *Si $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est co-coercif de module b (voir définition A.1.3), alors l'algorithme forward 2.2.2 avec une suite de pas ρ_k satisfaisant :*

$$0 \leq \rho_k \leq 2b \tag{2.1}$$

$$\sum_{k=0}^{+\infty} \rho_k (2b - \rho_k) = +\infty \tag{2.2}$$

converge vers un zéro de T s'il en existe.

Algorithme 2.2.3 Itérations de point fixe avec sous-relaxation

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
2. $s^{k+1} = (1 - \alpha_k)s^k + \alpha_k N(s^k)$
3. $k \leftarrow k + 1$; Aller à l'étape 2

Démonstration. Les zéros de T sont les points fixes de l'opérateur $N \stackrel{\text{def}}{=} I - 2bT$. Considérons l'opérateur $I - \rho_k T = (1 - \frac{\rho_k}{2b})I + \frac{\rho_k}{2b}(I - 2bT)$ pour k fixé. L'opérateur bT est pseudo-contractant d'après le lemme B.2.1, $I - 2bT$ est donc faiblement contractant. Si la suite des ρ_k satisfait (2.1), alors la suite $\alpha_k = \frac{\rho_k}{2b}$ satisfait l'hypothèse B.3.1 et la proposition B.3.1 nous fournit la convergence. \square

Dans les problèmes pratiques, la propriété de co-coercivité est assez rare. Elle nécessite en particulier que l'opérateur en question soit une application, excluant ainsi les sous-différentiels issus de l'optimisation non différentiable. Néanmoins, elle peut être fabriquée artificiellement par la procédure de régularisation de Moreau-Yosida que nous présentons maintenant.

2.2.2 Algorithme du point proximal généralisé

Etant donnée une fonction convexe F s.c.i propre et un point $s \in \mathbb{R}^n$, considérons le problème d'optimisation suivant :

$$\min_{y \in \mathbb{R}^n} F(y) + \frac{1}{2} \|y - s\|^2 \quad (2.3)$$

La fonction $F(\cdot) + \frac{1}{2} \|\cdot - s\|^2$ étant fortement convexe, elle admet un et un seul argument minimum noté $P(s)$ appelé le *point proximal* de s relativement à F :

$$P(s) = \arg \min_y F(y) + \frac{1}{2} \|y - s\|^2 \quad (2.4)$$

La fonction \check{F} qui à s associe le minimum dans (2.3) :

$$\check{F}(s) = \min_y F(y) + \frac{1}{2} \|y - s\|^2 = F(P(s)) + \frac{1}{2} \|P(s) - s\|^2 \quad (2.5)$$

est appelée *régularisée de Moreau-Yosida* de F ([77, 78, 50]). Graphiquement, \check{F} est obtenue en déplaçant une parabole concave sur le graphe de F (voir figure 2.2.2). Cette fonction régularisée possède exactement les mêmes minimums et arguments minimums que ceux de F . De plus, elle est différentiable et son gradient donné par la formule :

$$\nabla \check{F}(s) = s - P(s) \quad (2.6)$$

est pseudo-contractant. Ce dernier constitue donc un candidat idéal pour remplacer T dans l'algorithme forward 2.2.2 au vu de la proposition 2.2.1.

Ce procédé de régularisation s'étend aux opérateurs monotones généraux. En effet, l'opérateur P qui calcule les solutions du problème 2.3 et qui prend la forme :

$$P(s) = \{y \in \mathbb{R}^n / 0 \in \partial F(y) + y - s\}$$

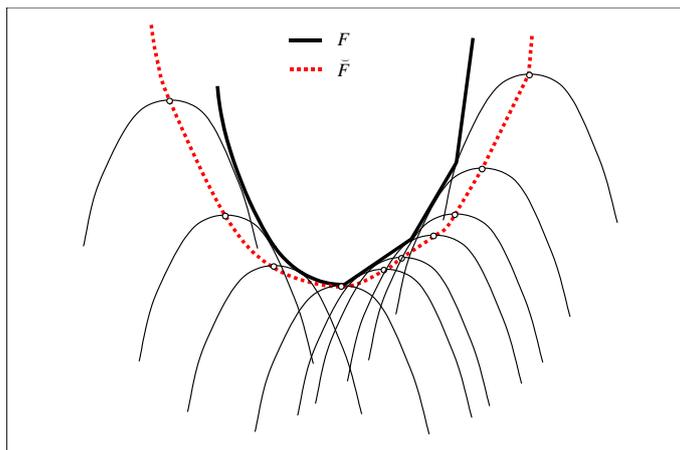


FIGURE 2.2.2 – Graphe d’une fonction F et de sa régularisée de Moreau-Yosida \bar{F} .

se transpose naturellement pour un opérateur T monotone maximal à :

$$P(s) = \{y \in \mathbb{R}^n / 0 \in T(y) + y - s\}$$

P est appelé opérateur *résolvant* ou *proximal* associé à T et s’écrit algébriquement $P = (I + T)^{-1}$. L’existence et l’unicité de la solution à l’inclusion 2.7 :

$$\text{Trouver } y \text{ tel que } 0 \in T(y) + y - s \quad (2.7)$$

vient comme corollaire de la proposition suivante, due à George Minty ([76]) :

Proposition 2.2.2 (Minty). *Soit T maximal monotone. Son résolvant $P = (I + T)^{-1}$ est pseudo-contractant et $\text{dom}(P) = \mathbb{R}^n$.*

Le gradient $\nabla \bar{F}$ au vu de la formule 2.6 possède son analogue :

$$Q = I - P = (I + T^{-1})^{-1} \quad (2.8)$$

appelé *opérateur régularisé de Moreau-Yosida* de T , qui est le résolvant de T^{-1} . $Q = \frac{1}{2}I + \frac{1}{2}(I - 2P)$ est pseudo-contractant lui aussi et partage les mêmes zéros que T , justifiant ainsi son appellation d’“opérateur régularisé”. L’application de l’algorithme 2.2.2 à l’opérateur Q constitue le célèbre *algorithme du point proximal* 2.2.4. On parle parfois de pas “backward”

Algorithme 2.2.4 Algorithme du point proximal généralisé

1. $k = 0$; Choisir $y^0 \in \mathbb{R}^n$
 2. $y^{k+1} = y^k - \rho_k Q(y^k) = (1 - \rho_k)y^k + \rho_k P(y^k)$
 3. $k \leftarrow k + 1$; Aller à l’étape 2
-

pour l’étape 2 de l’algorithme 2.2.4 par opposition au pas “forward” de l’algorithme 2.2.2 afin d’identifier le caractère implicite ou explicite de la résolution de l’inclusion (2.7). La convergence est donnée par la proposition 2.2.3 suivante.

Proposition 2.2.3. Soit $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal et Q son opérateur régularisé. Si

$$\begin{aligned} 0 &\leq \rho_k \leq 2 \\ \sum_{k=0}^{+\infty} \rho_k(2 - \rho_k) &= +\infty \end{aligned}$$

alors la suite $\{y^k\}_k$ générée par l'algorithme 2.2.4 converge vers un zéro de T .

Démonstration. Il s'agit en fait d'un corollaire de la proposition 2.2.1 en observant que Q est pseudo-contrayant, c'est à dire co-coercif de constante $b = 1$. \square

L'algorithme 2.2.4 fut initiée par Martinet dans [74] et [75] pour $T = \partial F$ et $T = \partial L$ sous le nom de “méthode de régularisations successives”. Rockafellar la généralisa ensuite ([93], [94]) au cas d'opérateurs monotones quelconques en la rebaptisant “méthode du point proximal”. L'introduction de coefficients de sous-relaxation est enfin due à Eckstein et Bertsekas [35].

2.2.3 Applications à l'optimisation

Changement de variable

En règle générale, avant d'appliquer un algorithme pour chercher un zéro d'un opérateur, il est souvent souhaitable d'effectuer un changement de variable afin d'obtenir un opérateur mieux “conditionné” sur lequel l'algorithme en question peut potentiellement converger plus vite. Soit M une matrice inversible. Si T est monotone maximal, alors $\mathcal{T} = M^{-\top} T M^{-1}$ l'est aussi ([98, Théorème 12.43]) et la correspondance entre les graphes de T et \mathcal{T} est donnée par :

$$u \in T(y) \iff M^{-\top} u \in \mathcal{T}(My)$$

en particulier, les zéros de T et de \mathcal{T} sont liés par la relation :

$$0 \in T(y) \iff \begin{cases} 0 \in \mathcal{T}(z) \\ z = My \end{cases}$$

Les algorithmes de recherche de zéro appliqués à T peuvent donc aussi l'être à \mathcal{T} et les solutions recouvrées par le changement de variable inverse $y = M^{-1}z$. L'inclusion du problème (2.7) se transforme alors en :

$$\begin{aligned} 0 &\in M^{-\top} T(M^{-1}z) + z - s \\ 0 &\in T(M^{-1}z) + M^{\top}(z - s) \end{aligned}$$

et en revenant aux variables originales du problèmes y :

$$0 \in T(y) + \Lambda(y - s) \tag{2.9}$$

avec $\Lambda = M^{\top} M$. Un changement de variable usuel consiste pour $\lambda > 0$ à poser $M = \sqrt{\lambda} I$, c'est à dire $\Lambda = \lambda I$. Nous pouvons imaginer sur la figure 2.3.1, l'impact du choix de la matrice Λ dans (2.9) qui remplace l'espace d'équation $y + u = s$ par celui d'équation

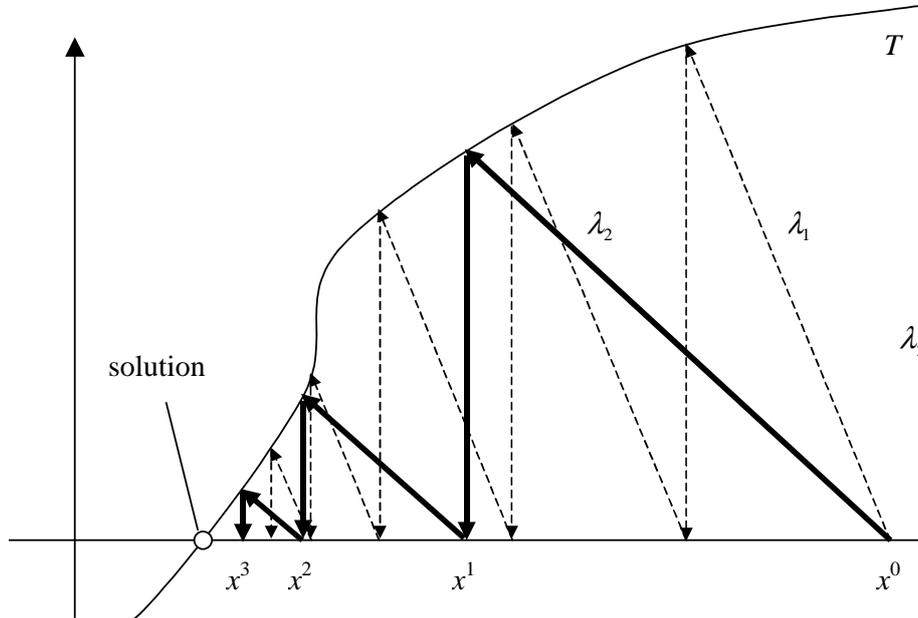


FIGURE 2.2.3 – Effet du paramètre λ sur la convergence de l'algorithme du point proximal.

Algorithme 2.2.5 Algorithme proximal appliqué à $T = \partial F$ avec mise à l'échelle

1. $k = 0$; Choisir $y^0 \in \mathbb{R}^n$
 2. $z^{k+1} = \arg \min_z F(z) + \frac{1}{2} \langle z - y^k, \Lambda(z - y^k) \rangle$
 3. $y^{k+1} = (1 - \rho_k)y^k + \rho_k z^{k+1}$
 4. $k \leftarrow k + 1$; Aller à l'étape 2
-

$My + M^\top u = s$. En dimension 1, plus λ est grand et plus la droite devient “horizontale”. La conséquence sur la convergence de l’algorithme est schématisée sur la figure 2.2.3 : plus λ est choisi grand, plus la convergence est rapide. A titre d’exemple, la méthode proximale 2.2.4 avec $T = \partial F$ devient dans ce cas l’algorithme 2.2.5.

De prime abord, l’algorithme du point proximal apparaît être inutilisable en pratique car les étapes 2 des algorithmes 2.2.4 ou 2.2.5 semblent présenter une complexité équivalente, voire supérieure à celle du problème de départ. Une façon de rendre l’approche moins rédhitoire, et très suivie dans la littérature [94, 110] est d’effectuer un calcul approché dans (2.7). Néanmoins, dans certains cas d’intérêt, le calcul de P est aisé, par exemple si $T = \partial H$ ou $T = \partial L$ avec L et H le lagrangien et la fonction duale du problème perturbé (A.8). Le calcul de P est alors de complexité équivalente à celui de T , tous deux nécessitant la résolution d’un problème d’optimisation. L’algorithme du point proximal coïncide alors avec les méthodes de Lagrangien Augmenté, proposées pour la première fois dans les années 60 par Hestenes ([48]) et de Lagrangien Augmenté proximal ([93]) présentées juste après. Enfin, lorsque l’on a affaire à un problème d’optimisation à structure décomposable comme le problème (1.2) qui nous intéresse, et lorsque T est un opérateur d’éclatement (voir le chapitre 3), le calcul de P deviendra plus facile que celui de T puisqu’il donnera lieu à une décomposition du problème.

Lagrangien augmenté

L’algorithme de lagrangien augmenté appliqué au problème d’optimisation ($P_{\bar{y}}$) peut se voir comme l’application de la méthode proximale pour résoudre le problème dual $D_{\bar{y}}$:

$$\underset{u \in \mathbb{R}^m}{\text{Minimiser}} H(u) - \langle \bar{y}, u \rangle \quad (2.10)$$

Le problème résolu à chaque itération s’écrit :

$$\begin{aligned} 0 &\in \partial H(u^{k+1}) + \lambda(u^{k+1} - u^k) - \bar{y} \\ \lambda(u^k - u^{k+1}) + \bar{y} &\in \partial H(u^{k+1}) \\ (x^{k+1}, \lambda(u^k - u^{k+1}) + \bar{y}) &\in \partial \varphi^*(0, u^{k+1}) \end{aligned}$$

En posant $y^{k+1} = \lambda(u^k - u^{k+1}) + \bar{y}$, cette inclusion se réécrit :

$$\begin{aligned} (0, 0) &\in \partial \varphi(x^{k+1}, y^{k+1}) + (0, \lambda^{-1}(y^{k+1} - \bar{y}) - u^k) \\ (x^{k+1}, y^{k+1}) &\in \arg \min_{x \in \mathbb{R}^n, y \in \mathbb{R}^m} \varphi(x, y) + \frac{1}{2\lambda} \|y - \bar{y}\|^2 - \langle u^k, y \rangle \end{aligned}$$

et nous retrouvons finalement la forme classique de l’algorithme de lagrangien augmenté :

Algorithme 2.2.6 Algorithme de lagrangien augmenté

1. $(y^0, u^0) \in \mathbb{R}^m \times \mathbb{R}^m$ donné
 2. $(x^{k+1}, y^{k+1}) \in \arg \min_{x \in \mathbb{R}^n, y \in \mathbb{R}^m} \varphi(x, y) + \frac{1}{2\lambda} \|y - \bar{y}\|^2 - \langle u^k, y \rangle$
 3. $u^{k+1} = u^k - \lambda^{-1}(y^{k+1} - \bar{y})$
 4. $k \leftarrow k + 1$; Aller à l’étape 2
-

Lagrangien augmenté proximal

En remplaçant maintenant T par $\partial L - (0, \bar{y})$ plutôt que $\partial H - \bar{y}$ dans l'algorithme 2.2.4, on obtient cette fois ci l'algorithme de lagrangien augmenté proximal ([93]). Le problème résolu à chaque itération s'écrit :

$$\begin{aligned} (0, \bar{y}) &\in \partial L(x^{k+1}, u^{k+1}) + \lambda \{(x^{k+1}, u^{k+1}) - (x^k, u^k)\} \\ (\lambda(x^k - x^{k+1}), \lambda(u^k - u^{k+1}) + \bar{y}) &\in \partial L(x^{k+1}, u^{k+1}) \\ (\lambda(x^k - x^{k+1}), u^{k+1}) &\in \partial \varphi(x^{k+1}, \lambda(u^k - u^{k+1}) + \bar{y}) \end{aligned}$$

en posant cette fois encore $y^{k+1} = \lambda(u^k - u^{k+1}) + \bar{y}$:

$$(0, 0) \in \partial \varphi(x^{k+1}, y^{k+1}) + (\lambda(x^{k+1} - x^k), \lambda^{-1}(y^{k+1} - \bar{y}) - u^k)$$

L'algorithme de lagrangien augmenté proximal prend la forme de l'algorithme 2.2.7.

Algorithme 2.2.7 Algorithme de lagrangien augmenté proximal

1. $(y^0, x^0, u^0) \in \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^m$ donné
 2. $(x^{k+1}, y^{k+1}) \in \arg \min_{x,y} \varphi(x, y) + \frac{\lambda}{2} \|x - x^k\|^2 + \frac{1}{2\lambda} \|y - \bar{y}\|^2 - \langle u^k, y \rangle$
 3. $u^{k+1} = u^k - \lambda^{-1}(y^{k+1} - \bar{y})$
 4. $k \leftarrow k + 1$; Aller à l'étape 2
-

Il s'agit de l'algorithme 2.2.6 auquel on a ajouté le terme proximal $\frac{\lambda}{2} \|x - x^k\|^2$.

Lors de l'étude de la convergence de l'algorithme du point proximal 2.2.4, nous nous sommes servis des propriétés de faible contraction de l'opérateur $N = I - 2Q$ intervenant dans le calcul $I - \rho_k T = (1 - \frac{\rho_k}{2})I + \frac{\rho_k}{2}(I - 2Q)$. Cet opérateur peut être interprété comme une généralisation des opérations de réflexion par rapport à des sous-espace affines de \mathbb{R}^n . Nous étudions dans la section 2.3 suivante ces *réflexions généralisées* qui sont à la base des méthodes d'éclatement présentées aux sections 3.3 et 3.4.

2.3 Quelques propriétés des réflexions généralisées

Nous étudions dans cette section les *réflexions généralisées* faisant l'objet de la définition 2.3.1.

Définition 2.3.1. Soit $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal, $P = (I + T)^{-1}$ et $Q = I - P$. L'opérateur :

$$N = P - Q = 2P - I = I - 2Q$$

est la réflexion généralisée associée à T .

Cette terminologie provient du cas où T est le cône normal $\mathbf{N}_{\mathcal{A}}$ à un espace vectoriel $\mathcal{A} \subset \mathbb{R}^n$ (voir la sous-section A.1.4 en annexe pour une définition). L'opérateur P est alors le projecteur sur l'espace \mathcal{A} ; Q le projecteur sur l'espace \mathcal{A}^\perp et N correspond à la réflexion usuelle par rapport à \mathcal{A} . La formule $N = 2P - I$ fournit la procédure pour calculer $d = N(s)$.

1 Trouver y tel que $0 \in T(y) + y - s$ i.e. $y = P(s)$

2 Poser $d = 2y - s$

Les réflexions généralisées correspondent exactement aux opérateurs faiblement contractants de domaine \mathbb{R}^n tout entier. Ces derniers peuvent donc être réinterprétés systématiquement comme la réflexion généralisée associée à l'opérateur maximal monotone sous-jacent $(\frac{1}{2}I + \frac{1}{2}N)^{-1} - I$. Cette relation donne lieu à la correspondance de la proposition 2.3.1 suivante entre les graphes de T et de N .

Proposition 2.3.1. *Soit $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal et $N : \mathbb{R}^n \rightarrow \mathbb{R}^n$ sa réflexion généralisée. $u \in T(y)$ si et seulement si $d = N(s)$ avec de façon équivalente :*

$$\begin{cases} s = y + u \\ d = y - u \end{cases} \text{ ou } \begin{cases} y = \frac{1}{2}(d + s) \\ u = \frac{1}{2}(d - s) \end{cases}$$

De la même manière, tout opérateur pseudo-contractant P de domaine \mathbb{R}^n peut s'interpréter comme l'opérateur résolvant de $P^{-1} - I$ ou bien comme le régularisé de $(P^{-1} - I)^{-1}$ qui sont tous deux monotones et maximaux. La proposition 2.3.1 possède alors l'analogue :

Proposition 2.3.2. *Soit $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ monotone maximal et $s \in \mathbb{R}^n$. Les trois propositions suivantes sont équivalentes.*

i) $u \in T(y)$

ii) $y = P(s)$ et $s = y + u$

iii) $u = Q(s)$ et $s = y + u$

Autrement dit, il existe un unique couple $(y, u) \in \text{gr}(T)$ tel que $y + u = s$. Ce couple étant donné par l'application $s \mapsto (P(s), Q(s))$ qui est donc une bijection de \mathbb{R}^n dans $\text{gr}(T)$.

Cette décomposition de s est appelée *décomposition proximale* sur le graphe de T ou encore *décomposition de Moreau* ou de *Minty*. Elle est représentée graphiquement pour $T : \mathbb{R} \rightarrow \mathbb{R}$ sur la figure 2.3.1. Il s'agit de l'unique point d'intersection entre le graphe de T et le sous-espace affine $\{(y, u)/y + u = s\}$. Une autre représentation est donnée sur la figure 2.3.2 pour $T = \nabla F : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ l'opérateur gradient d'une forme quadratique F . L'image de s par la réflexion généralisée N correspondante est dessinée sur la figure 2.3.3.

Si $T = \mathbf{N}_{\mathcal{A}}$ (le cône normal à l'espace vectoriel \mathcal{A}), alors il s'agit de la décomposition classique d'un vecteur s en la somme de deux éléments orthogonaux $y \in \mathcal{A}$ et $u \in \mathcal{A}^\perp$. Si $T = \mathbf{N}_C$ est le cône normal d'un cône convexe fermé C , alors la décomposition coïncide avec la décomposition en deux éléments mutuellement polaires $y \in C$ et $u \in C^\circ$.

Le tableau 2.1 récapitule la correspondance bijective entre les trois classes d'opérateurs : monotones maximaux, faiblement contractants de domaine \mathbb{R}^n tout entier et pseudo-contractants de domaine \mathbb{R}^n tout entier.

2.3.1 Propriétés de contraction

L'algorithme du point proximal 2.2.4 est l'algorithme 2.2.3 appliqué à N en posant $\alpha_k = \frac{\rho_k}{2}$ pour tout k . Sous certaines hypothèses sur T , sa réflexion généralisée N possède des propriétés de contraction supplémentaires qui permettent d'établir la convergence de 2.2.4 dans des cas limites.

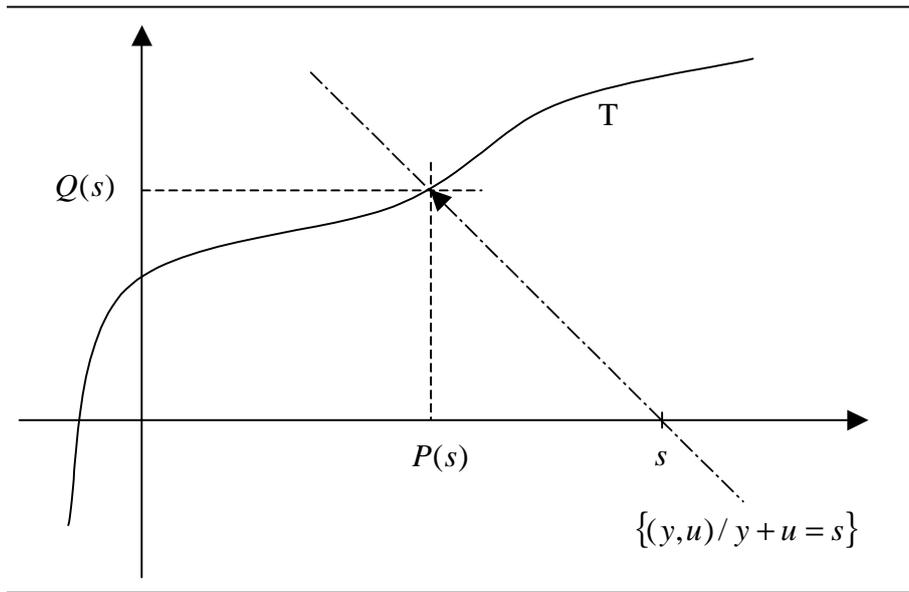


FIGURE 2.3.1 – Décomposition proximale de s sur le graphe de T .

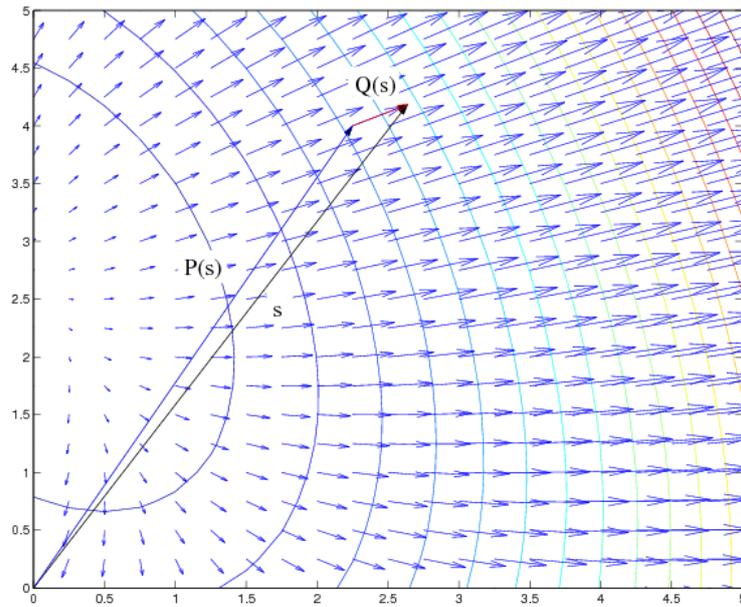


FIGURE 2.3.2 – Décomposition de Moreau de s sur le graphe de ∇F où F est une forme quadratique.

| Monotone | Faiblement contractant | Pseudo-contractant |
|-----------------------------|------------------------|----------------------|
| T | $2(I + T)^{-1} - I$ | $(I + T)^{-1}$ |
| $(N + I)^{-1} \circ 2I - I$ | N | $\frac{1}{2}(N + I)$ |
| $P^{-1} - I$ | $2P - I$ | P |

TABLE 2.1 – Bijection entre les trois types d'opérateurs : monotones, faiblement contractants et pseudo-contractants.

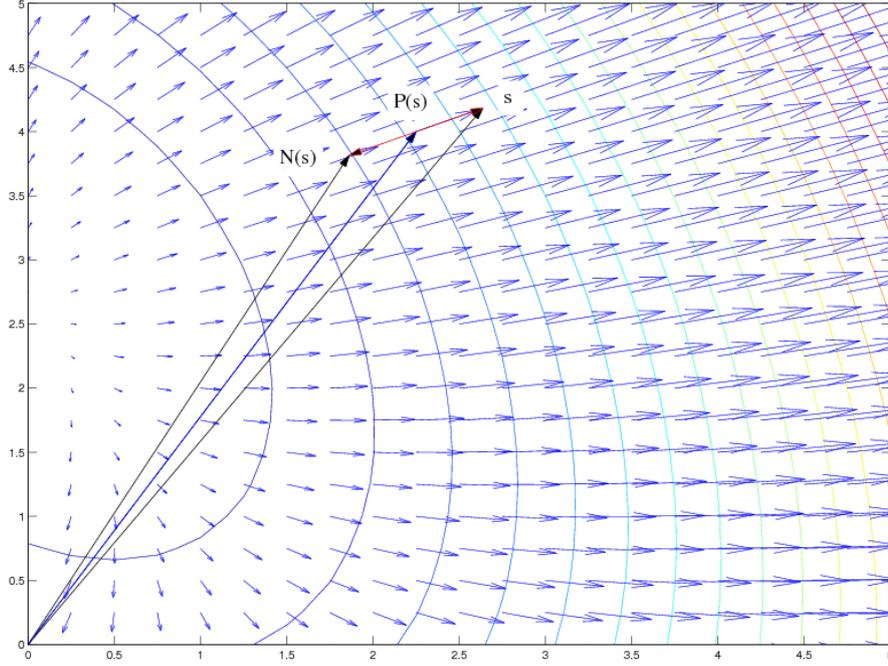


FIGURE 2.3.3 – Image $N(s)$ d'un point s par la réflexion généralisée N associée au sous-différentiel d'une forme quadratique

Lemme 2.3.1. Soit $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal et $N : \mathbb{R}^n \rightarrow \mathbb{R}^n$ sa réflexion généralisée.

$$\|N(s) - N(s')\|^2 \leq \|P(s) - P(s')\|^2 + \|Q(s) - Q(s')\|^2 - 2 \langle P(s) - P(s'), Q(s) - Q(s') \rangle \quad (2.11)$$

$$\|N(s) - N(s')\|^2 \leq \|s - s'\|^2 - 2 \langle P(s) - P(s'), Q(s) - Q(s') \rangle \quad (2.12)$$

Si T est fortement monotone ou co-coercif, alors N possède des propriétés plus fortes que la contraction faible, réunies dans la proposition suivante.

Proposition 2.3.3. Soit $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal et $N : \mathbb{R}^n \rightarrow \mathbb{R}^n$ sa réflexion généralisée.

i) Si T est co-coercif de constante b alors N est $\frac{1}{1+\frac{b}{2}}$ -moyenné.

ii) Si de plus, T est fortement monotone de constante a alors N est contractant de rapport $\sqrt{1 - \frac{2ab}{a+b}} < 1$.

iii) Si T est strictement monotone alors N est strictement contractant.

Démonstration. D'après la proposition 2.3.2, $Q(s) \in T(P(s))$ et $Q(s') \in T(P(s'))$. La co-coercivité se traduit par :

$$\langle P(s) - P(s'), Q(s) - Q(s') \rangle \geq b \|Q(s) - Q(s')\|^2 \quad (2.13)$$

impliquant ainsi :

$$\|N(s) - N(s')\|^2 \leq \|s - s'\|^2 - 2b \|Q(s) - Q(s')\|^2$$

et puisque $Q = \frac{1}{2}(I - N)$, nous obtenons :

$$\|N(s) - N(s')\|^2 \leq \|s - s'\|^2 - \frac{b}{2} \|(I - N)(s) - (I - N)(s')\|^2$$

qui caractérise d'après le lemme B.1.1 un opérateur α -moyenné avec $\frac{1-\alpha}{\alpha} = \frac{b}{2}$, soit $\alpha = \frac{1}{1+\frac{b}{2}} < 1$ prouvant ainsi *i*). Enfin, si T est aussi fortement monotone, ce qui se traduit symétriquement à (2.13) par :

$$\langle P(s) - P(s'), Q(s) - Q(s') \rangle \geq a \|P(s) - P(s')\|^2 \quad (2.14)$$

alors la multiplication de (2.14) par $\frac{b}{a+b}$ et de (2.13) par $\frac{a}{a+b}$ ajoutées l'une à l'autre donne :

$$-\langle P(s) - P(s'), Q(s) - Q(s') \rangle \leq -\frac{ab}{a+b} (\|P(s) - P(s')\|^2 + \|Q(s) - Q(s')\|^2)$$

qui insérée dans 2.11 permet d'obtenir *ii*) :

$$\|N(s) - N(s')\|^2 \leq \left(1 - \frac{2ab}{a+b}\right) \underbrace{(\|P(s) - P(s')\|^2 + \|Q(s) - Q(s')\|^2)}_{\leq \|s-s'\|^2}$$

□

Nous obtenons les corollaires suivant quant à la convergence de l'algorithme 2.2.4.

Corollaire 2.3.1. *Soit $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal et Q son opérateur régularisé. Si T est co-coercif de constante b et :*

$$\begin{aligned} 0 &\leq \rho_k \leq 2 + b \\ \sum_{k=0}^{+\infty} \rho_k (2 + b - \rho_k) &= +\infty \end{aligned}$$

alors la suite $\{y^k\}_k$ générée par l'algorithme 2.2.4 converge vers un zéro de T .

Démonstration. Soit N la réflexion généralisée de T et $J = (1 - \frac{\rho_k}{2})I + \frac{\rho_k}{2}N$ effectuant l'itération k . Puisque N est $\frac{1}{1+\frac{b}{2}}$ -moyenné,

$$J = \left(1 - \frac{\rho_k}{2}\right)I + \frac{\rho_k}{2} \left[\left(1 - \frac{1}{1+\frac{b}{2}}\right)I + \left(\frac{1}{1+\frac{b}{2}}\right)\tilde{N} \right]$$

pour un certain opérateur \tilde{N} faiblement contractant ayant les mêmes points fixes, soit en simplifiant.

$$J = \left(1 - \frac{\rho_k}{2+b}\right)I + \frac{\rho_k}{2+b}\tilde{N}$$

et la convergence vient de la proposition B.3.1 appliquée avec $\alpha_k = \frac{\rho_k}{2+b}$. □

Corollaire 2.3.2. *Soit $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal. Si T est co-coercif de constante b et fortement monotone de constante a alors la suite $\{y^k\}_k$ générée par l'algorithme 2.2.4 avec :*

$$0 \leq \rho_k \leq 2$$

$$\sum_{k=0}^{+\infty} \rho_k = +\infty$$

converge vers \bar{y} l'unique zéro de T . Si de plus, $\liminf \rho_k = \underline{\rho} > 0$, la vitesse est linéaire de taux :

$$\limsup_k \frac{\|y^{k+1} - \bar{y}\|}{\|y^k - \bar{y}\|} \leq 1 - \frac{\underline{\rho}}{2} \left(1 - \sqrt{1 - \frac{2ab}{a+b}} \right) < 1$$

Démonstration. Il s'agit de l'application directe de la proposition B.3.2. \square

Nous nous penchons maintenant sur les propriétés de différentiabilité et de proto-différentiabilité des réflexions généralisées, deux notions rappelées en annexe C.

2.3.2 Vitesse de convergence de l'algorithme proximal

Si $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal est proto-différentiable en \bar{y} pour \bar{u} , alors $DT(\bar{y}|\bar{u})$ est aussi un opérateur monotone maximal ([98, p. 569]) et positivement homogène ([96]). De plus P est semi-différentiable en $\bar{s} = \bar{y} + \bar{u}$ et sa semi-différentielle est le résolvant de $DT(\bar{y}|\bar{u})$ ([98, Proposition 9.50]). On montre aussi que $Q = I - P$ et $N = 2P - I$ sont semi-différentiables en $\bar{s} = \bar{y} + \bar{u}$ respectivement pour \bar{u} et $\bar{d} = \bar{y} - \bar{u}$ et que :

$$DP(\bar{s}) = (I_n + DT(\bar{y}|\bar{u}))^{-1} \quad (2.15a)$$

$$DQ(\bar{s}) = (I_n + DT(\bar{y}|\bar{u}))^{-1} \quad (2.15b)$$

$$DN(\bar{s}) = DP(\bar{s}) - DQ(\bar{s}) \quad (2.15c)$$

$$= 2DP(\bar{s}) - I_n \quad (2.15d)$$

$$= I_n - 2DQ(\bar{s}) \quad (2.15e)$$

Ces trois opérateurs sont positivement homogènes et possèdent un développement de type Taylor, par exemple pour N :

$$N(s) = N(\bar{s}) + DN(\bar{s})(s - \bar{s}) + o(\|s - \bar{s}\|) \quad (2.16)$$

Supposons que la suite des s^k générée par l'algorithme 2.2.4 converge vers un point \bar{s} où N possède un tel développement. Alors l'opérateur effectuant une itération noté :

$$J_{\alpha^k} \stackrel{def}{=} (1 - \alpha^k)I + \alpha^k N$$

est aussi semi-différentiable en \bar{s} . Nous avons alors le développement :

$$J_{\alpha^k}(s) = J_{\alpha^k}(\bar{s}) + DJ_{\alpha^k}(\bar{s})(s - \bar{s}) + o(\|s - \bar{s}\|)$$

En remplaçant s par s^k , l'erreur commise à chaque itération notée $e^k = s^k - \bar{s}$ est donnée récursivement par :

$$e^{k+1} = DJ_{\alpha^k}(\bar{s})e^k + o(\|e^k\|)$$

Nous introduisons pour un opérateur moyenné positivement homogène J la notation :

$$\|J\| = \sup_{e \in \mathbb{R}^n} \frac{\|Je\|}{\|e\|} \leq 1$$

qui généralise la norme classique des opérateurs linéaires. L'inégalité triangulaire fournit alors :

$$\|e^{k+1}\| \leq \|DJ_{\alpha^k}(\bar{s})\| \|e^k\| + o(\|e^k\|)$$

Si l'algorithme ne converge pas en un nombre fini d'étapes, alors

$$\frac{\|e^{k+1}\|}{\|e^k\|} \leq \|DJ_{\alpha^k}(\bar{s})\| + \frac{o(\|e^k\|)}{\|e^k\|}$$

et le taux de convergence est borné par :

$$\limsup_k \frac{\|e^{k+1}\|}{\|e^k\|} \leq \limsup_k \|DJ_{\alpha^k}(\bar{s})\| \quad (2.17)$$

2.3.3 Conditions de différentiabilité

Lorsque $T = \partial F$, alors $N = I - 2\nabla\check{F}$ où \check{F} est la régularisée de Moreau-Yosida de F . Les propriétés d'ordre 1 de N sont alors liées aux propriétés du second ordre de \check{F} . La question de la différentiabilité à l'ordre 2 de \check{F} a reçu un grand intérêt dans l'étude de la version exacte [12] ou de versions inexacts [61] de l'algorithme du point proximal afin de développer des méthodes de type quasi-Newton. L'existence d'une matrice hessienne $\nabla^2\check{F}$ a été étudié dans le cas convexe par Lucet [69], Lemaréchal et Sagastizabál ([58], [59]) et dans un cadre plus général par Poliquin et Rockafellar ([89]). Il est souligné dans [69] et [89] que la propriété clé pour obtenir la différentiabilité à l'ordre 2 de \check{F} est la linéarité au sens généralisé de $D[\partial F](\bar{y}|\bar{u})$, c'est à dire la linéarité de son graphe :

Définition 2.3.2. $D[\partial F](\bar{y}|\bar{u})$ est linéaire au sens généralisé si $D[\partial F](\bar{y}|\bar{u}) = S + \mathbf{N}_{\mathcal{H}}$ où \mathcal{H} est un sous-espace vectoriel de \mathbb{R}^n et S est une matrice symétrique semi-définie positive.

La linéarité générale pour les opérateurs multivoques est définie en toute généralité comme la linéarité du graphe. Cette caractérisation pour les opérateurs sous-différentiels monotones maximaux apparaît dans [95, Proposition 4.1]. La proposition suivante, adaptée de [69] caractérise la différentiabilité à l'ordre 2 de \check{F} et fournit la formule explicite de la matrice Hessienne par la même occasion.

Proposition 2.3.4. Les assertions suivantes sont équivalentes :

- i) ∂F est proto-différentiable en \bar{y} pour \bar{u} avec une proto-différentielle linéaire au sens généralisé $D[\partial F](\bar{y}|\bar{u}) = S + \mathbf{N}_{\mathcal{H}}$.
- ii) \check{F} est deux fois différentiable en $\bar{s} = \bar{y} + \bar{u}$ et

$$\nabla^2\check{F}(\bar{s}) = \nabla Q(\bar{s}) = [\Pi_{im(S)+\mathcal{H}^\perp} \circ [I_n + (\Pi_{\mathcal{H}}S\Pi_{\mathcal{H}})^\dagger] \circ \Pi_{im(S)+\mathcal{H}^\perp}]^\dagger$$

Si un opérateur sous-différentiel est linéaire au sens généralisé, il est le sous-différentiel d'une fonction *partiellement quadratique*. Comme il est noté dans [89], cette classe de fonctions peut être vue comme une extension des formes quadratiques classiques auxquelles on permet d'avoir des valeurs propres égales à $+\infty$. Afin de simplifier les notations, nous exploiterons une décomposition de l'espace en la somme directe de trois sous-espaces, correspondant respectivement à des valeurs propres dans $\{0\}$, $]0, +\infty[$ et $\{+\infty\}$.

Proposition 2.3.5. *Dans la définition 2.3.2, S peut être choisie de telle sorte que $\text{im}(S) \subset \mathcal{H}$.*

Démonstration. $D[\partial F](\bar{y}|\bar{u})$ est de la forme :

$$Sy + \mathbf{N}_{\mathcal{H}}(y) = \begin{cases} Sy + \mathcal{H}^{\perp} & \text{si } y \in \mathcal{H} \\ \emptyset & \text{sinon.} \end{cases}$$

Dans le premier cas, $y = \Pi_{\mathcal{H}}y$ et $Sy + \mathcal{H}^{\perp} = \Pi_{\mathcal{H}}(Sy) + \mathcal{H}^{\perp}$ donc S peut être remplacé par $\Pi_{\mathcal{H}}S\Pi_{\mathcal{H}}$, symétrique semi-définie positive et qui satisfait clairement $\text{im}(\Pi_{\mathcal{H}}S\Pi_{\mathcal{H}}) \subset \mathcal{H}$. \square

Dans ce qui suit, \mathbb{R}^n sera décomposé selon la somme directe :

$$\begin{aligned} \mathcal{H}_1 \oplus \mathcal{H}_2 \oplus \mathcal{H}_3 &= \mathbb{R}^n \\ \mathcal{H}_1 &= \mathcal{H} \cap \text{im}(S) = \text{im}(S) \\ \mathcal{H}_2 &= \mathcal{H} \cap \ker(S) \\ \mathcal{H}_3 &= \mathcal{H}^{\perp} \end{aligned}$$

Les dimensions respectives sont notées n_1, n_2, n_3 avec $n_1 + n_2 + n_3 = n$. Ces espaces sont mutuellement orthogonaux puisque $\ker(S) = \text{im}(S)^{\perp}$. Nous donnons la preuve que la somme est directe :

Démonstration. Soit $s \in \mathbb{R}^n$. s se décompose ainsi :

$$\begin{aligned} s &= \Pi_{\ker(S)}s + \Pi_{\text{im}(S)}s \\ &= \Pi_{\mathcal{H}}\Pi_{\ker(S)}s + \Pi_{\mathcal{H}^{\perp}}\Pi_{\ker(S)}s + \Pi_{\text{im}(S)}s \end{aligned}$$

Nous avons clairement :

$$\begin{aligned} \Pi_{\text{im}(S)}s &\in \text{im}(S) = \mathcal{H}_1 \\ \Pi_{\mathcal{H}^{\perp}}\Pi_{\ker(S)}s &\in \mathcal{H}^{\perp} = \mathcal{H}_3 \\ \Pi_{\mathcal{H}}\Pi_{\ker(S)}s &\in \mathcal{H} \end{aligned}$$

Il reste à montrer que $\Pi_{\mathcal{H}}\Pi_{\ker(S)}$ appartient aussi à $\ker(S)$. L'inclusion $\text{im}(S) \subset \mathcal{H}$ donne $\mathcal{H}^{\perp} \subset \ker(S)$ d'où :

$$\Pi_{\mathcal{H}}\Pi_{\ker(S)}s = \Pi_{\ker(S)}s - \underbrace{\Pi_{\mathcal{H}^{\perp}}\Pi_{\ker(S)}s}_{\in \ker(S)} \in \ker(S)$$

\square

Cette décomposition est très liée à la décomposition $\mathcal{U}\text{-}\mathcal{V}$ de [60]. Plus précisément, $\mathcal{U} = \mathcal{H}$ est l'espace suivant lequel F est différentiable et $\mathcal{V} = \mathcal{H}^\perp$ est son complémentaire, l'espace parallèle au sous-différentiel. Ici, nous faisons l'hypothèse supplémentaire que F possède une matrice hessienne dans l'espace \mathcal{U} . Nous considérons alors une sous-décomposition de cet espace en d'une part le sous-espace associé aux valeurs propres nulles, dans lequel F est localement linéaire et d'autre part le sous-espace associé aux valeurs propres strictement positives dans lequel F est fortement convexe.

Désormais, nous nous plaçons dans une base orthonormale B adaptée à $\mathcal{H}_1 \times \mathcal{H}_2 \times \mathcal{H}_3$. S s'écrit dans cette nouvelle base :

$$B^{-1}SB = \begin{pmatrix} L & & \\ & 0 & \\ & & 0 \end{pmatrix} \quad (2.18)$$

où L est une sous-matrice $n_1 \times n_1$ symétrique définie positive. Ces notations permettent de simplifier les calculs dans la proposition suivante :

Proposition 2.3.6. *Les assertions suivantes sont équivalentes :*

- i) ∂F est proto-différentiable en \bar{y} pour \bar{u} avec une proto-derivée linéaire au sens généralisé $D[\partial F](\bar{y}|\bar{u}) = S + \mathbf{N}_{\mathcal{H}_1 \oplus \mathcal{H}_2}$ et $\text{im}(S) = \mathcal{H}_1$.
- ii) La régularisée de Moreau-Yosida \check{F} est deux fois différentiable en $\bar{s} = \bar{y} + \bar{u}$ et

$$\nabla^2 \check{F}(\bar{s}) = \nabla Q(\bar{s}) = B \begin{pmatrix} (I_{n_1} + L^{-1})^{-1} & & \\ & 0 & \\ & & I_{n_3} \end{pmatrix} B^{-1}$$

- iii) N est différentiable en $\bar{s} = \bar{y} + \bar{u}$ et

$$\nabla N(\bar{s}) = B \begin{pmatrix} 2(I_{n_1} + L)^{-1} - I_{n_1} & & \\ & I_{n_2} & \\ & & -I_{n_3} \end{pmatrix} B^{-1} \quad (2.19)$$

Démonstration. *i* et *ii* sont de simples reformulations des points *i* et *ii* de la proposition 2.3.4 tandis que *iii* provient de (2.15e). Nous détaillons la formule de $\nabla Q(\bar{s})$:

$$\begin{aligned} \nabla Q(\bar{s}) &= \nabla^2 \check{F}(\bar{s}) \\ &= [\Pi_{\mathcal{H}_1 \oplus \mathcal{H}_3} \circ [I_n + (\Pi_{\mathcal{H}_1 \oplus \mathcal{H}_2} \circ S \circ \Pi_{\mathcal{H}_1 \oplus \mathcal{H}_2})^\dagger] \circ \Pi_{\mathcal{H}_1 \oplus \mathcal{H}_3}]^\dagger \\ &= [\Pi_{\mathcal{H}_1 \oplus \mathcal{H}_3} \circ [I_n + S^\dagger] \circ \Pi_{\mathcal{H}_1 \oplus \mathcal{H}_3}]^\dagger \\ &= [\Pi_{\mathcal{H}_1 \oplus \mathcal{H}_3} + S^\dagger]^\dagger \\ &= B \begin{pmatrix} (I_{n_1} + L^{-1})^{-1} & & \\ & 0 & \\ & & I_{n_3} \end{pmatrix} B^{-1} \end{aligned}$$

□

La matrice S^\dagger est la pseudo-inverse de Penrose-Moore de S . Pour une matrice symétrique définie positive, elle est obtenue en inversant les valeurs propres strictement positives et en conservant les valeurs propres nulles.

Pour finir, nous mettons en exergue deux situations extrêmes de la formule (2.19), correspondant respectivement aux cas $\mathcal{H}_3 = \{0\}$ et $\mathcal{H}_1 = \{0\}$.

Lorsque $\mathcal{H}_3 = \{0\}$

$D[\partial F](\bar{y}|\bar{u})$ est alors réduit à l'application linéaire S et ∂F est différentiable. La formule de $\nabla^2 \check{F}(\bar{s})$ donnée par la proposition 2.3.7 est alors un résultat classique figurant par exemple dans [59] et ses références.

Proposition 2.3.7. *Si ∂F est différentiable en \bar{y} avec $D[\partial F](\bar{y}) = S$ et $\partial F(\bar{y}) = \{\bar{u}\}$, alors \check{F} est deux fois différentiable en $\bar{s} = \bar{y} + \bar{u}$ et :*

$$\nabla Q(\bar{s}) = \nabla^2 \check{F}(\bar{s}) = I_n - (I_n + S)^{-1}$$

Corollaire 2.3.3. *Si ∂F est différentiable en \bar{y} avec $D[\partial F](\bar{y}) = S$ et $\partial F(\bar{y}) = \{\bar{u}\}$, alors N est différentiable en $\bar{s} = \bar{y} + \bar{u}$ et :*

$$\nabla N(\bar{s}) = 2(I_n + S)^{-1} - I_n$$

Notons que l'opérateur $\mathcal{T}(\cdot) = M^{-\top} \partial F(M^{-1} \cdot)$ obtenu grâce au changement de variable :

$$(z, w) \leftrightarrow (My, M^{-\top} u)$$

est alors proto-différentiable en $(\bar{z}, \bar{w}) = (M\bar{y}, M^{-\top} \bar{u})$ et sa réflexion généralisée notée \mathcal{N} est différentiable en $\bar{\sigma} = \bar{z} + \bar{w}$ avec les formules :

$$\begin{aligned} D[\mathcal{T}](\bar{z}|\bar{w}) &= M^{-\top} S M^{-1} \\ \nabla \mathcal{N}(\bar{\sigma}) &= 2(I_{n_1} + M^{-\top} S M^{-1})^{-1} - I_{n_1} \end{aligned}$$

Lorsque $\mathcal{H}_1 = \{0\}$

Dans cette deuxième situation, la proto-dérivée $D[\partial F](\bar{y}|\bar{u})$ se résume à $\mathbf{N}_{\mathcal{H}}$. Cela implique $\mathcal{H}_2 = \mathcal{H}$, $\mathcal{H}_3 = \mathcal{H}^\perp$ et d'après la proposition 2.3.6, N est différentiable avec :

$$\nabla N(\bar{s}) = B \begin{pmatrix} I_{n_2} & \\ & -I_{n_3} \end{pmatrix} B^{-1} = R_{\mathcal{H}} \quad (2.20)$$

Il s'agit de l'opérateur de réflexion par rapport à l'espace \mathcal{H} . Notamment, si F est une fonction polyédrale, alors ∂F possède partout une proto-dérivée qui est presque partout linéaire-généralisée avec $\mathcal{H}_1 = 0$. Nous avons considéré sur la figure 2.3.3 les opérateurs P , Q et N associés à une fonction polyédrale et représenté leur proto-dérivée en deux points distincts. Remarquons au passage que d'autres fonctions, qui ne sont pas polyédrales peuvent néanmoins présenter un sous-différentiel de proto-dérivée linéaire-généralisé ayant $\mathcal{H}_1 = \{0\}$. Par exemple :

$$F : y \mapsto \frac{1}{3} |y|^3 \qquad \nabla F : y \mapsto \begin{cases} y^2 & \text{if } y \geq 0 \\ -y^2 & \text{if } y < 0 \end{cases}$$

∇F est différentiable en zéro, de dérivée égale à $y \mapsto 0$. De même ∇F^* a une proto-dérivée égale au cône normal $\mathbf{N}_{\{0\}}$ en 0 pour 0.

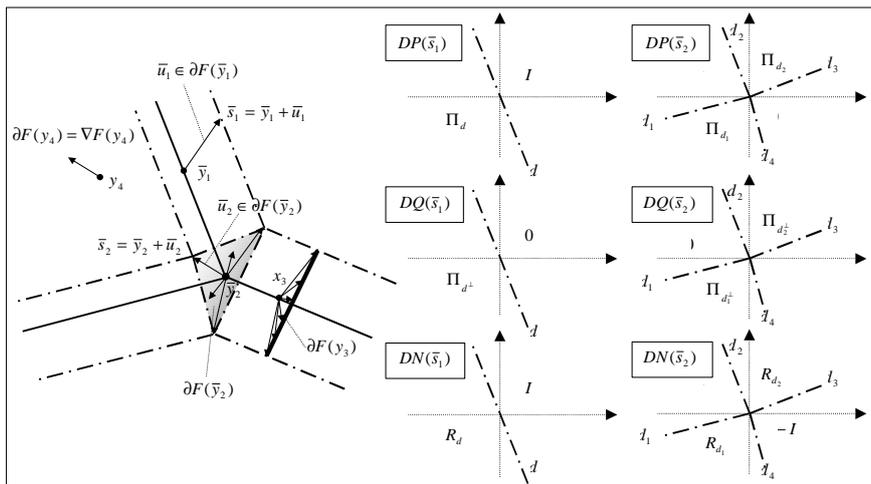


FIGURE 2.3.4 – Opérateurs proto-dérivées associés à une fonction polyédrale

2.3.4 Lois internes \odot et \star

L'ensemble des opérateurs faiblement contractants est stable par composition. Les correspondances du tableau 2.1 permettent alors d'induire une loi interne \odot sur l'ensemble des opérateurs pseudo-contractants et une autre \star sur l'ensemble des opérateurs monotones ([55]). La première est définie pour tout couple d'opérateurs pseudo-contractants (P_1, P_2) par :

$$P_2 \star P_1 = \frac{1}{2}I + \frac{1}{2} \left[\underbrace{(2P_2 - I)}_{N_2} \circ \underbrace{(2P_1 - I)}_{N_1} \right]$$

et la deuxième est définie pour tout couple d'opérateurs monotones (T_2, T_1) :

$$T_2 \odot T_1 = \underbrace{(I + T_2)^{-1}}_{P_2} \star \underbrace{(I + T_1)^{-1}}_{P_1} - I$$

Stabilité vis à vis de la convergence graphique

Nous montrons maintenant que ces lois sont stables par passage à la limite graphique (définition C.2.2). Dans ce but, nous nous attachons d'abord à la stabilité des correspondances du tableau 2.1.

Lemme 2.3.2. *Soit pour tout $k \in \mathbb{N}$, $\mathcal{T}^k : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ monotone maximal, $\mathcal{P}^k : \mathbb{R}^n \rightarrow \mathbb{R}^n$ son résolvant et $\mathcal{N}^k : \mathbb{R}^n \rightarrow \mathbb{R}^n$ la réflexion généralisée associée. Alors :*

$$\mathcal{T}^k \xrightarrow[g]{} \mathcal{T} \iff \mathcal{P}^k \xrightarrow[g]{} \mathcal{P} \iff \mathcal{N}^k \xrightarrow[g]{} \mathcal{N}$$

et alors \mathcal{T} est monotone maximal, \mathcal{P} est son résolvant et \mathcal{N} la réflexion généralisée associée.

Démonstration. D'après la proposition 2.3.1, le graphe de \mathcal{N}^k est obtenu à partir de celui de \mathcal{T}^k par la transformation linéaire :

$$(y, u) \mapsto (y + u, y - u)$$

Celui ce \mathcal{P}^k est obtenu à partir de celui de \mathcal{N}^k par l'application :

$$(s, d) \mapsto (s, \frac{1}{2}s + d)$$

Enfin, le graphe de \mathcal{T}^k est obtenu à partir de celui de \mathcal{P}^k par :

$$(s, p) \mapsto \frac{1}{2}(p, s - p)$$

Par continuité et coercivité de ces trois application, nous obtenons directement l'équivalence grâce à [98, Théorème 4.26]. Enfin si \mathcal{T}^k converge, sa limite est nécessairement monotone maximale [98, Théorème 12.32], \mathcal{N} et \mathcal{P} sont donc par continuité, la réflexion généralisée et le résolvant associés. \square

Afin d'établir la stabilité de \star et \odot vis à vis de la convergence graphique, il suffit donc de montrer celle de la composition pour les réflexions généralisées.

Lemme 2.3.3. *Soit pour tout $k \in \mathbb{N}$ \mathcal{N}_1^k et \mathcal{N}_2^k deux réflexions généralisées. Alors :*

$$\mathcal{N}_1^k \xrightarrow{g} \mathcal{N}_1, \mathcal{N}_2^k \xrightarrow{g} \mathcal{N}_2, \implies [\mathcal{N}_2^k \circ \mathcal{N}_1^k] \xrightarrow{g} [\mathcal{N}_2 \circ \mathcal{N}_1],$$

Démonstration. Nous utilisons le fait (voir [98, Corollaire 5.45 et Théorème 5.43]) que pour une suite d'opérateurs equi-continue, ce qui est le cas de la suite $\{\mathcal{N}_1^k\}_k$ dont tous les éléments sont lipschitziens de constante 1, la convergence graphique implique la convergence continue, c'est à dire que pour toute suite $\{s^k\}_k \rightarrow s$ nous avons :

$$\mathcal{N}_1^k(s^k) \rightarrow \mathcal{N}_1(s)$$

Soit $(s, t) \in gr(\mathcal{N}_2 \circ \mathcal{N}_1)$ et posons $d = \mathcal{N}_1(s)$ (nous avons donc $t = \mathcal{N}_2(d)$). Soit $\{s^k\}_k$ une suite convergente vers s avec pour tout k :

$$\begin{aligned} d^k &\stackrel{def}{=} \mathcal{N}_1^k(s^k) \\ t^k &\stackrel{def}{=} \mathcal{N}_2^k(d^k) \end{aligned}$$

Par construction, $(s^k, t^k) \in gr(\mathcal{N}_2^k \circ \mathcal{N}_1^k)$. La convergence continue de $\{\mathcal{N}_1^k\}_k$ implique $d^k \rightarrow d$, puis celle de $\{\mathcal{N}_2^k\}_k$ donne $t^k \rightarrow t$. La suite $\mathcal{N}_2^k \circ \mathcal{N}_1^k$ converge donc ponctuellement vers $\mathcal{N}_2 \circ \mathcal{N}_1$, ce qui est équivalent à la convergence graphique puisqu'il s'agit d'une suite equi-continue ([98, Théorème 5.40]). \square

Proposition 2.3.8. *Les lois \odot et \star sont stables par passage à la limite graphique.*

Démonstration. Il suffit de combiner le lemme 2.3.3 avec la proposition 2.3.2. \square

Conclusion

Ce chapitre a constitué l'occasion d'un bref rappel sur les méthodes de régularisation en optimisation convexe non différentiable. Nous avons choisi de mettre l'accent sur l'interprétation qu'elles ont en terme d'applications successives d'un opérateur de réflexion généralisée N (algorithme 2.2.3) pour l'aisance qu'elle offre dans l'analyse de convergence.

L'intérêt pour la différentiabilité de l'opérateur N dans l'étude de la convergence de l'algorithme du point proximal est à notre connaissance nouveau, l'opérateur regardé étant le plus souvent P ou Q . De même, la question de la stabilité des lois \odot et \star quant à la convergence graphique semble n'avoir jamais été abordée auparavant. Elle se révélera fort utile dans le chapitre suivant pour étudier la convergence des *méthodes d'éclatement* (*splitting methods* en anglais) développées spécifiquement pour tirer profit de la structure décomposable du problème (1.2), ce dont ne sont pas capables les applications directes de l'algorithme du point proximal (algorithmes 2.2.5, 2.2.6 et 2.2.7).

Chapitre 3

Éclatement d'opérateurs : décomposition et coordination

3.1 Introduction

Nous rappelons notre objectif développé au chapitre 1, d'exploiter la structure décomposable du problème générique (1.2) :

$$\underset{y \in \mathbb{R}^n}{\text{Minimiser}} \quad f(y) = \sum_{i=1}^p F_i(y_i) + C(y_1, \dots, y_p)$$

qui en plus de notre problème de gestion de production et de la programmation stochastique en général ([97, 91]), trouve des applications dans d'autres domaines [37, 52], notamment les télécommunications ([81, 9]), le contrôle optimal ([36]) et le traitement du signal [4].

Soit T_1 et T_2 deux opérateurs monotones maximaux (par exemple $T_1 = \partial F$ et $T_2 = \partial C$). Nous ferons dans la suite l'hypothèse 3.1.1, c'est à dire qu'il existe un couple de solutions primale-duale au problème de trouver un zéro de la somme $T \stackrel{def}{=} T_1 + T_2$.

Hypothèse 3.1.1. *Il existe $y^*, u^* \in \mathbb{R}^n$ tels que :*

$$u^* \in T_1(y^*) \tag{3.1a}$$

$$-u^* \in T_2(y^*) \tag{3.1b}$$

Dans le cas $(T_1, T_2) = (\partial F, \partial C)$, un couple satisfaisant (3.1) vérifie $0 \in \partial F(y^*) + \partial C(y^*) \subset \partial[F + C](y^*)$ et y^* minimise alors $F + C$. Cependant, les minimiseurs de $F + C$ ne sont pas en toute généralité les zéros de $\partial F + \partial C$. Les deux opérateurs $\partial F + \partial C$ et $\partial[F + C]$ coïncident en particulier sous les conditions suffisantes de la proposition A.1.2.

Le principe des techniques d'éclatement d'opérateur est de chercher un zéro de la somme $T = T_1 + T_2$ où T_1 et T_2 sont monotones maximaux (mais pas nécessairement T) en effectuant des calculs séparés sur T_1 et T_2 . La manipulation des opérateurs résolvants $P_1 = (I + T_1)^{-1}$ et $P_2 = (I + T_2)^{-1}$ est autorisée mais pas celle de $P = (I + T)^{-1}$, dont le calcul est jugé "trop difficile". L'utilisation de ce principe s'avère d'autant plus bénéfique que l'évaluation de P_1 et de P_2 est simple. C'est le cas par exemple :

- lorsque l'un des opérateurs, disons T_1 , est une somme séparable $T_1(y) = T_{1,1}(y_1) +$

$\dots + T_{1,p}(y_p)$. P_1 présente alors une structure bloc-angulaire :

$$P_1(y) = \begin{pmatrix} P_{1,1}(y_1) \\ \vdots \\ P_{1,p}(y_p) \end{pmatrix} \quad (3.2)$$

et son calcul peut être parallélisé. L'opérateur ∂F dans le problème (1.2) présente une telle structure.

- lorsque T_1 est l'opérateur cône normal d'un ensemble de structure simple comme une boule en normes $\|\cdot\|_1$, $\|\cdot\|_2$, $\|\cdot\|_\infty$ ou un espace vectoriel de structure simple. P_1 est alors un opérateur de projection qu'il est facile de calculer analytiquement.

La thèse de Jonathan Eckstein [33] contient une revue fournie de ces techniques. Nous nous contentons ici de présenter les méthodes qui ont suscité le plus d'attention dans la littérature, à savoir la méthode "Forward-Backward" (section 3.2) et les méthodes de type "Rachford" (sections 3.3 et 3.4) .

3.2 Méthode Forward-Backward

3.2.1 Problème de point fixe équivalent

Les opérateurs T_1 et T_2 peuvent être respectivement remplacés dans le système (3.1) par $\mathcal{T}_1 = \lambda^{-1}T_1$ et $\mathcal{T}_2 = \lambda^{-1}T_2$ pour $\lambda > 0$ fixé et donnent lieu, en effectuant le changement de variable $w \leftrightarrow \lambda^{-1}u$, au problème équivalent :

$$\text{Trouver } w^*, y^* \in \mathbb{R}^n \text{ tel que } w^* \in \mathcal{T}_1(y^*) \text{ et } -w^* \in \mathcal{T}_2(y^*)$$

La première inclusion se réécrit :

$$y^* - w^* \in (I - \mathcal{T}_1)(y^*) \quad (3.3)$$

tandis que (3.1b), à l'aide de la proposition 2.3.2 peut se réécrire :

$$y^* = (I + \mathcal{T}_2)^{-1}(y^* - w^*) \quad (3.4)$$

En composant avec (3.3), y^* est une solution si et seulement si elle vérifie l'équation de point fixe :

$$y^* \in (I + \mathcal{T}_2)^{-1}(I - \mathcal{T}_1)y^* \quad (3.5)$$

3.2.2 Algorithme

La méthode Forward-Backward (algorithme 3.2.1) fut proposée par Lions et Mercier ([68]) Passty ([83]) et par Han et Lou [42]. Elle exécute des itérations successives de l'opérateur $(I + \mathcal{T}_2)^{-1}(I - \mathcal{T}_1)$ pour résoudre (3.5). Le calcul de l'opérateur à l'étape 2 est détaillé dans l'algorithme 3.2.2. Si $T_1 = \nabla F$ et $T_2 = \mathbf{N}_{\mathcal{A}}$, alors il s'agit de l'algorithme de gradient projeté proposée par Goldstein et par Levitin et Polyak ([6]) pour le problème de minimisation de F sur \mathcal{A} . Pour deux opérateurs à valeurs dans \mathbb{R} , les itérations sont représentées schématiquement sur la figure 3.2.1.

La méthode Forward-Backward est très liée au principe du problème auxiliaire introduit par Guy Cohen [15], [16], [17] pour les problèmes d'optimisations puis généralisé

Algorithme 3.2.1 Algorithme forward-backward

1. Choisir $y^0 \in \mathbb{R}^n$
2. $y^{k+1} \in (I + \mathcal{T}_2)^{-1}(I - \mathcal{T}_1)y^k$
3. $k \leftarrow k + 1$; Aller à l'étape 2

Algorithme 3.2.2 Algorithme forward-backward : version détaillée

1. Choisir $y^0 \in \mathbb{R}^n$
2. $u^k = \mathcal{T}_1(y^k)$
3. Trouver y^{k+1} tel que $0 \in \mathcal{T}_2(y^{k+1}) + \lambda(y^{k+1} - y^k) + u^k$
4. $k \leftarrow k + 1$; Aller à l'étape 2

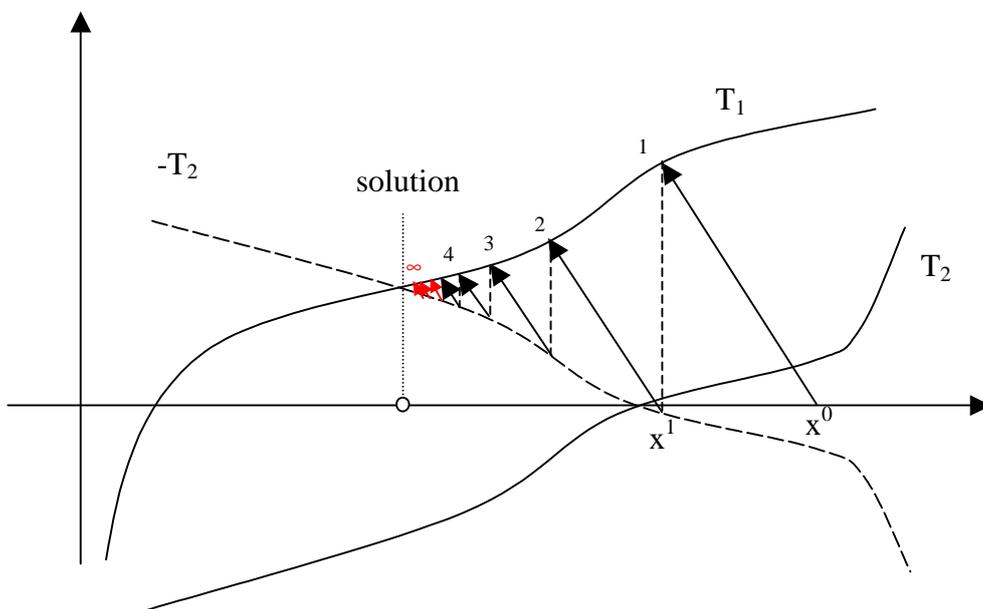


FIGURE 3.2.1 – Suite des itérés de l'algorithme Forward-backward

par la suite aux problèmes d'inéquations variationnelles. Ce principe algorithmique qui généralise et relie un grand nombre de méthodes variationnelles permet entre autre de remplacer l'opérateur identité dans le calcul du résolvant par un opérateur auxiliaire non linéaire.

3.2.3 Application à la décomposition

Appliquée à la décomposition ; la méthode forward-backward 3.2.2 permet d'obtenir les méthodes de décomposition par les prix et par les ressources (ou quantités) ([54],). En effet, soit le problème d'optimisation suivant.

$$\begin{aligned} \text{Minimiser}_{x_1, x_2, \dots, x_p} \quad & f_1(x_1) + \dots + f_p(x_p) \\ \text{s.c} \quad & G_1 x_1 + G_2 x_2 + \dots + G_p x_p = R \end{aligned}$$

qui modélise le problème du partage optimal d'une quantité de ressource R entre p unités.

Décomposition par les ressources

Supposons que les fonctions implicites $F_i(y_i) = \min_{G_i x_i = y_i} f_i(x_i)$ soient s.c.i. Le problème se met sous la forme équivalente :

$$\text{Minimiser}_{y_1, y_2, \dots, y_p} \quad F_1(y_1) + \dots + F_p(y_p) \quad (3.6a)$$

$$\text{s.c} \quad y_1 + y_2 + \dots + y_p = R \quad (3.6b)$$

Notons que cette formulation rentre dans le cadre du problème (1.2) en posant $C = \delta_{\mathcal{A}}$ la fonction indicatrice de l'espace affine :

$$\mathcal{A} = \{(y_1, y_2, \dots, y_p) / y_1 + y_2 + \dots + y_p = R\}$$

L'application de la méthode forward-backward donne alors l'algorithme de décomposition par les ressources 3.2.3.

Algorithme 3.2.3 Algorithme de décomposition par les ressources

1. Choisir $y^0 \in \mathbb{R}^n$
 2. $u^k = \nabla F(y^k)$
 3. $y^{k+1} = \Pi_{\mathcal{A}}(y^k - \lambda u^k)$
 4. $k \leftarrow k + 1$; Aller à l'étape 2
-

Décomposition par les prix

Soit pour $i = 1, \dots, p$ $H_i = F_i^*$ la conjuguée convexe de F_i . Le problème dual (voir annexe A.3) s'écrit :

$$\begin{aligned} \text{Minimiser}_{u_1, u_2, \dots, u_p} \quad & H_1(u_1) + \dots + H_p(u_p) \\ \text{s.c} \quad & u_1 = u_2 = \dots = u_p \end{aligned}$$

Ce problème rentre aussi dans le cadre (1.2) avec $F \leftarrow H$ et $C \leftarrow \delta_{\mathcal{A}^\perp}$ la fonction indicatrice de l'espace vectoriel :

$$\mathcal{A}^\perp = \{(u_1, u_2, \dots, u_p) / u_1 = u_2 = \dots = u_p\}$$

Algorithme 3.2.4 Algorithme de décomposition par les prix

1. Choisir $u^0 \in \mathbb{R}^n$
2. $y^k = \nabla H(u^k)$
3. $u^{k+1} = \Pi_{\mathcal{A}^\perp}(u^k - \lambda y^k)$
4. $k \leftarrow k + 1$; Aller à l'étape 2

3.2.4 Convergence

Passty montre en supposant la suite des itérés y^k bornée, que l'utilisation d'une σ -suite à la place du paramètre λ entraîne la convergence ergodique de y^k ([83]). Gabay [38] quant à lui donne des garanties de convergence reposant sur la co-coercivité de T_1 :

Proposition 3.2.1 (Gabay). *Si T_1 est co-coercif de module a et si $0 < \lambda^{-1} < 2a$ alors l'algorithme forward-backward 3.2.1 converge vers une solution du système (3.1).*

Démonstration. La preuve utilise celle de la proposition 2.2.1. En suivant les mêmes étapes, nous obtenons le caractère moyenné de $I - \mathcal{T}_1$. L'opérateur $(I + \mathcal{T}_2)^{-1}$ est de son côté pseudo-contractant en tant qu'opérateur résolvant de \mathcal{T}_2 d'après la proposition 2.2.2. L'opérateur $(I + \mathcal{T}_2)^{-1}(I - \mathcal{T}_1)$ est par conséquent moyenné grâce à la proposition B.1.3 et $\{y^k\}_k$ converge en se référant à la proposition B.3.1. \square

Tseng [120] a étendu ce résultat au cas d'un pas λ pouvant varier dans une certaine plage au cours des itérations. Dans [125], Zhu reprend la méthodologie de [70] pour étudier le comportement asymptotique en introduisant des propriétés quantitative de croissance "locales" de T_1 et T_2 autour de la solution par opposition aux propriétés globales plus contraignantes utilisées jusque là. Dans [13], Chen et Rockafellar passent en revue les résultats de convergence linéaire et s'intéressent au cas où T_1 est lipschitz continu de domaine tout \mathbb{R}^n et la somme $T_1 + T_2$ est fortement monotone avec λ dans une certaine plage.

Vitesse de convergence

Si l'algorithme converge vers (\bar{y}, \bar{w}) et que \mathcal{T}_1 est semi-différentiable en \bar{y} et \mathcal{T}_2 est proto-différentiable en \bar{y} relativement à \bar{w} , alors l'opérateur effectuant une itération :

$$J = (I + \mathcal{T}_2)^{-1}(I - \mathcal{T}_1)$$

est semi-différentiable en \bar{y} et :

$$DJ(\bar{y}) = (I + D\mathcal{T}_2(\bar{y}|\bar{w}))^{-1}(I - D\mathcal{T}_1(\bar{y}))$$

et la vitesse de convergence est bornée par :

$$\limsup_k \frac{\|e^{k+1}\|}{\|e^k\|} \leq \limsup_k \|DJ(\bar{y})\| \quad (3.7)$$

En particulier si $D\mathcal{T}_1 = I$, alors $DJ(\bar{y}) = 0$ et la convergence est super-linéaire.

Tous les résultats de convergence globale cités nécessitent des propriétés de croissance (co-coercivité ou forte monotonie) sur au moins l'un des deux opérateurs mis en jeu. Cette

régularité est absente de beaucoup d'applications notamment celles faisant intervenir des programmes linéaires. De plus, le choix de λ doit toujours s'effectuer, sous peine de diverger, dans un intervalle dépendant des constantes de lipschitz ou de forte monotonie, la plupart du temps inconnues.

Cette limitation, essentiellement due à l'utilisation directe de l'opérateur T_1 sans régularisation est palliée dans les méthodes de Peaceman-Rachford et Douglas-Rachford présentées respectivement dans les sections 3.3 et 3.4 qui viennent.

3.3 Méthode de Peaceman-Rachford

Les méthodes présentées ici et dans la section 3.4 furent proposées par Lions et Mercier dans [68]. Le nom de Rachford leur est associé car il s'agit d'extensions aux opérateurs monotones de techniques développées pour les opérateurs linéaires dans les travaux de Peaceman et Rachford [84] puis de Douglas et Rachford [30]. Une introduction éclairante peut être trouvée dans [36].

3.3.1 Problème de point fixe équivalent

Soit N_1 et N_2 les réflexions généralisées associées respectivement à T_1 et T_2 . Grâce à la proposition 2.3.1, les équations d'optimalité (3.1) peuvent se réécrire :

$$y^* - u^* = N_1(y^* + u^*) \quad (3.8a)$$

$$y^* + u^* = N_2(y^* - u^*) \quad (3.8b)$$

En combinant ces égalités, les solutions (y^*, u^*) de (3.1) sont caractérisées par l'équation de point fixe :

$$y^* + u^* = N_2 \circ N_1(y^* + u^*) \quad (3.9)$$

L'opérateur $N_2 \circ N_1$ est faiblement contractant de domaine \mathbb{R}^n comme composition d'opérateurs ayant cette propriété. L'algorithme de Peaceman-Rachford (3.3.1) consiste alors à l'évaluer itérativement.

3.3.2 Algorithme

Algorithme 3.3.1 Algorithme de Peaceman-Rachford

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
 2. $s^{k+1} = [N_2 \circ N_1] s^k$
 3. $k \leftarrow k + 1$; Aller à l'étape 2
-

Les étapes à suivre sont détaillées dans l'algorithme 3.3.2 en nous aidant de la procédure du calcul de N de la page 32 et en posant pour tout k :

$$\begin{aligned} s^k &= y^k + u^k \\ \tilde{y}^{k+1} &= P_1(y^k + u^k) \\ \tilde{u}^{k+1} &= Q_1(y^k + u^k) \\ y^{k+1} &= P_2(\tilde{y}^{k+1} - \tilde{u}^{k+1}) \\ u^{k+1} &= -Q_2(\tilde{y}^{k+1} - \tilde{u}^{k+1}) \end{aligned}$$

Algorithme 3.3.2 Algorithme de Peaceman-Rachford (version détaillée)

1. $k = 0$; Choisir $(y^0, u^0) \in \mathbb{R}^m \times \mathbb{R}^m$
2. Trouver \tilde{y}^{k+1} tel que $0 \in T_1(\tilde{y}^{k+1}) + (\tilde{y}^{k+1} - y^k) - u^k$
3. $\tilde{u}^{k+1} = u^k + (y^k - \tilde{y}^{k+1})$
4. Trouver y^{k+1} tel que $0 \in T_2(y^{k+1}) + (y^{k+1} - \tilde{y}^{k+1}) + \tilde{u}^{k+1}$
5. $u^{k+1} = \tilde{u}^{k+1} - (\tilde{y}^{k+1} - y^{k+1})$
6. $k \leftarrow k + 1$; Aller à l'étape 2

Facteur d'échelle

La méthode de Peaceman-Rachford est usuellement mise en pratique après avoir effectué un changement de variable préalable. Ce changement consiste essentiellement à remplacer T_1 et T_2 par les opérateurs monotones maximaux aT_1 et aT_2 ou bien encore $aT_1(b^{-1}\cdot)$ et $aT_2(b^{-1}\cdot)$ où $a > 0$ et $b > 0$. Ces coefficients ont pour rôle de dilater les espaces des variables primales et duales l'un par rapport à l'autre afin de modifier le conditionnement des opérateurs. Nous montrons maintenant que la méthode n'est sensible qu'au rapport des dilatations $\frac{b}{a}$ et que par conséquent, un seul paramètre suffit. Le problème résolu est le suivant :

Trouver $z^*, w^* \in \mathbb{R}^n$ tels que $w^* \in \mathcal{T}_1(z^*)$ et $-w^* \in \mathcal{T}_2(z^*)$

avec $\mathcal{T}_1(\cdot) = aT_1(b^{-1}\cdot)$ et $\mathcal{T}_2(\cdot) = aT_2(b^{-1}\cdot)$. Nous considérons donc les opérateurs non-expansifs \mathcal{N}_1 et \mathcal{N}_2 associés respectivement à \mathcal{T}_1 et à \mathcal{T}_2 ainsi que l'opérateur effectuant une itération $\mathcal{N}_2 \circ \mathcal{N}_1$.

En effectuant simultanément les changements $y^k = b^{-1}y^k$, $u^k = a^{-1}u^k$, $\tilde{y}^k = b^{-1}\tilde{y}^k$, $\tilde{u}^k = b^{-1}\tilde{u}^k$, $\lambda = \frac{b}{a}$, l'algorithme 3.3.2 appliqué au couple $(\mathcal{T}_1, \mathcal{T}_2)$ se spécialise en l'algorithme 3.3.3. Nous remarquons en premier lieu que les paramètres a et b ont disparu et

Algorithme 3.3.3 Algorithme de Peaceman-Rachford avec facteur d'échelle

1. $k = 0$; Choisir $(y^0, u^0) \in \mathbb{R}^m \times \mathbb{R}^m$
2. Trouver \tilde{y}^{k+1} tel que $0 \in T_1(\tilde{y}^{k+1}) + \lambda(\tilde{y}^{k+1} - y^k) - u^k$
3. $\tilde{u}^{k+1} = u^k + \lambda(y^k - \tilde{y}^{k+1})$
4. Trouver y^{k+1} tel que $0 \in T_2(y^{k+1}) + \lambda(y^{k+1} - \tilde{y}^{k+1}) + \tilde{u}^{k+1}$
5. $u^{k+1} = \tilde{u}^{k+1} - \lambda(\tilde{y}^{k+1} - y^{k+1})$
6. $k \leftarrow k + 1$; Aller à l'étape 2

que seul leur ratio λ intervient dans cette version finale. De plus, les suites y^k , u^k , \tilde{y}^k , \tilde{u}^k convergent respectivement à la même vitesse que les suites y^k , u^k , \tilde{y}^k , \tilde{u}^k , le second groupe étant obtenu à partir du premier par simple dilatation de facteur a ou b . En particulier, les trois choix suivants seront équivalents en terme de vitesse de convergence :

$$\mathcal{T}_1(\cdot) = T_1(\lambda^{-1}\cdot) \quad \text{et} \quad \mathcal{T}_2(\cdot) = T_2(\lambda^{-1}\cdot) \quad (3.10a)$$

$$\mathcal{T}_1(\cdot) = \lambda^{-1}T_1(\cdot) \quad \text{et} \quad \mathcal{T}_2(\cdot) = \lambda^{-1}T_2(\cdot) \quad (3.10b)$$

$$\mathcal{T}_1(\cdot) = \lambda^{-\frac{1}{2}}T_1(\lambda^{-\frac{1}{2}}\cdot) \quad \text{et} \quad \mathcal{T}_2(\cdot) = \lambda^{-\frac{1}{2}}T_2(\lambda^{-\frac{1}{2}}\cdot) \quad (3.10c)$$

Nous opterons pour la dernière écriture qui possède l'avantage de se généraliser au cas matriciel. En effet, si une matrice Λ est symétrique définie-positive, alors elle possède une racine carrée notée $\Lambda^{\frac{1}{2}}$ (telle que $(\Lambda^{\frac{1}{2}})^2 = \Lambda$) elle aussi symétrique et définie-positive et d'inverse notée $\Lambda^{-\frac{1}{2}}$. Si pour T monotone maximal, les opérateurs $T(\Lambda^{-1}\cdot)$ et $\Lambda^{-1}T(\cdot)$ ne sont pas nécessairement monotones, l'opérateur $\Lambda^{-\frac{1}{2}}T(\Lambda^{-\frac{1}{2}}\cdot)$ quant à lui l'est systématiquement. Dans ce cas le couple d'opérateurs suivant est un candidat valide pour l'application de l'algorithme.

$$\mathcal{T}_1(\cdot) = \Lambda^{-\frac{1}{2}}T_1(\Lambda^{-\frac{1}{2}}\cdot) \qquad \mathcal{T}_2(\cdot) = \Lambda^{-\frac{1}{2}}T_2(\Lambda^{-\frac{1}{2}}\cdot)$$

D'une manière plus générale le choix d'une matrice inversible M telle que $M^\top M = \Lambda$ et :

$$\mathcal{T}_1(\cdot) = M^{-\top}T_1(M^{-1}\cdot) \qquad \mathcal{T}_2(\cdot) = M^{-\top}T_2(M^{-1}\cdot) \qquad (3.11)$$

convient aussi on montre en suivant la même démarche que la vitesse de convergence ne dépend pas du choix de M , mais uniquement de celui de Λ . Les paramètres λ et Λ seront appelés par la suite *facteurs d'échelle* ou *paramètres de mise à l'échelle*.

Application à la décomposition

Nous portons maintenant sur le papier la spécialisation de l'algorithme 3.3.3 au problème qui nous motive, à savoir le problème d'optimisation décomposable 1.2 avec $T_1 = \partial F$ et $T_2 = \partial C$. La matrice de mise à l'échelle doit être choisie bloc-diagonale afin de permettre la séparabilité du calcul de \mathcal{N}_1 :

$$\Lambda = \begin{pmatrix} \Lambda_1 & & \\ & \ddots & \\ & & \Lambda_p \end{pmatrix}$$

Algorithme 3.3.4 Algorithme de Peaceman-Rachford appliqué à la décomposition

1. $k = 0$; Choisir $y^0, u^0 \in \mathbb{R}^m$
 2. Pour $i = 1, \dots, p$ faire
 3. $\tilde{y}_i^{k+1} \in \arg \min_{y_i} F_i(y_i) + \frac{1}{2} \langle y_i - y_i^k, \Lambda_i(y_i - y_i^k) \rangle - \langle u_i^k, y_i \rangle$
 4. $\tilde{u}_i^{k+1} = u_i^k + \Lambda_i(y_i^k - \tilde{y}_i^{k+1})$
 5. Fin pour
 6. $y^{k+1} = \arg \min_y C(y) + \frac{1}{2} \langle y - \tilde{y}^{k+1}, \Lambda(y - \tilde{y}^{k+1}) \rangle + \langle \tilde{u}^{k+1}, y \rangle$
 7. $u^{k+1} = \tilde{u}^{k+1} - \Lambda(\tilde{y}^{k+1} - y^{k+1})$
 8. $k \leftarrow k + 1$; Aller à l'étape 2
-

3.3.3 Convergence

Sans hypothèses supplémentaires, la convergence n'est pas garantie comme le montre le contre-exemple de la figure 3.3.1. Nous pouvons tout de même en nous inspirant des résultats de la section 2.3 déduire quelques résultats de convergence en effectuant des hypothèses supplémentaires sur T_1 et T_2 .

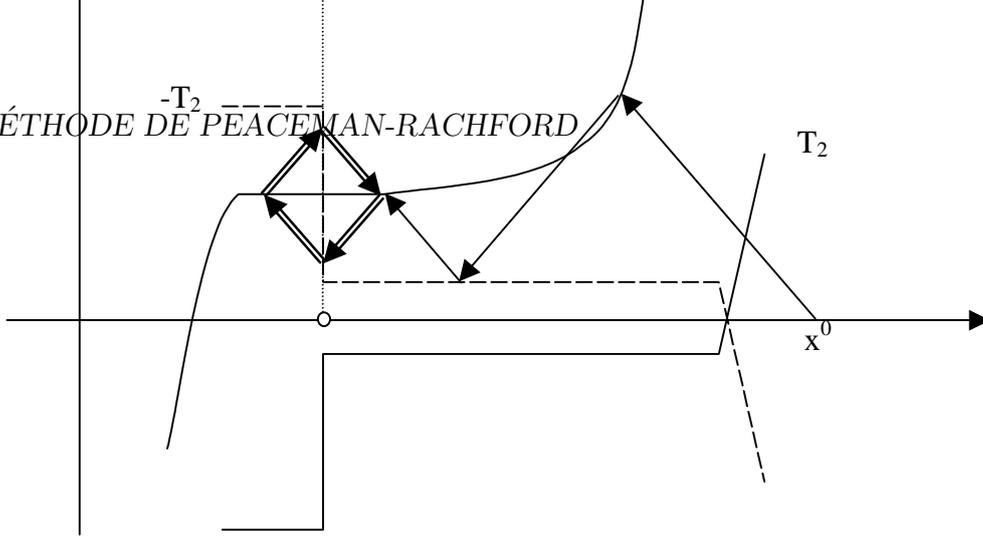


FIGURE 3.3.1 – Exemple où la méthode de Peaceman-Rachford ne converge pas.

Proposition 3.3.1.

- i) Si T_1 (respectivement T_2) est fortement monotone alors \tilde{y}^k (respectivement y^k) converge vers l'unique solution primale \bar{y} .
- ii) Si T_1 (respectivement T_2) est co-coercif alors \tilde{u}^k (respectivement u^k) converge vers l'unique solution duale \bar{u} .
- iii) Si T_1 et T_2 sont co-coercif alors les suites $\tilde{y}^k, y^k, \tilde{u}^k, u^k$ convergent.
- iv) Si T_1 (respectivement T_2) est fortement monotone de constante a et co-coercif de constante b alors les suites $\tilde{y}^k, y^k, \tilde{u}^k, u^k$ convergent linéairement vers l'unique solution primale-duale.

Démonstration. D'après l'équation 2.12 du lemme 2.3.1, nous avons pour tout $s, s', d = \mathcal{N}_1(s), d' = \mathcal{N}_1(s')$:

$$\begin{aligned} \|\mathcal{N}_2 \circ \mathcal{N}_1 s - \mathcal{N}_2 \circ \mathcal{N}_1 s'\|^2 &\leq \|s - s'\|^2 - \langle \mathcal{P}_1 s - \mathcal{P}_1 s', \mathcal{Q}_1 s - \mathcal{Q}_1 s' \rangle \\ &\quad - \langle \mathcal{P}_2 d - \mathcal{P}_2 d', \mathcal{Q}_2 d - \mathcal{Q}_2 d' \rangle \end{aligned}$$

Soit $\bar{s} \in \text{fix}(\mathcal{N}_2 \circ \mathcal{N}_1)$, $\bar{d} = \mathcal{N}_1 \bar{s}$, $\bar{y} = \mathcal{P}_1 \bar{s} = \mathcal{P}_2 \bar{d}$ et $\bar{u} = \mathcal{Q}_1 \bar{s} = -\mathcal{Q}_2 \bar{d}$. Les substitutions respectives de s^k et \bar{s} à s et s' donne :

$$\begin{aligned} \|s^{k+1} - \bar{s}\|^2 &\leq \|s^k - \bar{s}\|^2 - \langle \tilde{y}^{k+1} - \bar{y}, \tilde{u}^{k+1} - \bar{u} \rangle \\ &\quad - \langle y^{k+1} - \bar{y}, u^{k+1} - (-\bar{u}) \rangle \end{aligned}$$

Par monotonie de \mathcal{T}_1 et \mathcal{T}_2 , les deux produits scalaires du membre de droite sont positifs. En notant g^k leur somme, l'application du lemme B.3.1 avec $d_k = \|s^k - \bar{s}\|^2$, $(h_k^1, g_k^1) = (1, \langle \tilde{y}^{k+1} - \bar{y}, \tilde{u}^{k+1} - \bar{u} \rangle)$, $(h_k^2, g_k^2) = (1, \langle y^{k+1} - \bar{y}, u^{k+1} - (-\bar{u}) \rangle)$ donne :

$$\begin{aligned} \langle \tilde{y}^{k+1} - \bar{y}, \tilde{u}^{k+1} - \bar{u} \rangle &\rightarrow 0 \\ \langle y^{k+1} - \bar{y}, u^{k+1} - (-\bar{u}) \rangle &\rightarrow 0 \end{aligned}$$

Si de façon indépendante,

- I) T_1 est fortement monotone de constante a_1
- II) T_2 est fortement monotone de constante a_2

III) T_1 est co-coercif de constante b_1

IV) T_2 est co-coercif de constante b_2

alors :

$$\text{I) } a_1 \|\tilde{y}^{k+1} - \bar{y}\|^2 \leq \langle \tilde{y}^{k+1} - \bar{y}, \tilde{u}^{k+1} - \bar{u} \rangle \rightarrow 0$$

$$\text{II) } a_2 \|y^{k+1} - \bar{y}\|^2 \leq \langle y^{k+1} - \bar{y}, u^{k+1} - (-\bar{u}) \rangle \rightarrow 0$$

$$\text{III) } b_1 \|\tilde{u}^{k+1} - \bar{u}\|^2 \leq \langle \tilde{y}^{k+1} - \bar{y}, \tilde{u}^{k+1} - \bar{u} \rangle \rightarrow 0$$

$$\text{IV) } b_2 \|u^{k+1} - \bar{u}\|^2 \leq \langle y^{k+1} - \bar{y}, u^{k+1} - (-\bar{u}) \rangle \rightarrow 0$$

prouvant ainsi *i* et *ii*. Montrons maintenant *iii*. Si T_1 et T_2 sont co-coercifs, il en est de même pour \mathcal{T}_1 et \mathcal{T}_2 ce qui implique via la proposition 2.3.3 que \mathcal{N}_1 et \mathcal{N}_2 sont moyennés. Leur composition $\mathcal{N}_2 \circ \mathcal{N}_1$ l'est par conséquent aussi assurant ainsi la convergence de s^k d'après la proposition B.3.1 et de toutes les autres suites par continuité. Enfin, toujours d'après la proposition 2.3.3, si \mathcal{T}_1 est à la fois fortement monotone et co-coercif, alors \mathcal{N}_1 est contractant et $\mathcal{N}_2 \circ \mathcal{N}_1$ l'est aussi, assurant la convergence linéaire de toutes les suites et donc *iv*. Il en est de même pour \mathcal{T}_2 . \square

3.4 Méthode de Douglas-Rachford généralisée

La convergence de la méthode de Peaceman-Rachford nécessite des propriétés fortes sur les opérateurs T_1 et T_2 . Pour forcer la convergence à moindres frais et sans hypothèses autres que la monotonie et la maximalité, il suffit simplement de moyenner l'opérateur $\mathcal{N}_2 \circ \mathcal{N}_1$ comme l'affirme la proposition B.3.1. Le choix $\alpha_k \equiv \frac{1}{2}$ correspond historiquement à la *méthode de Douglas-Rachford* (algorithme 3.4.1) et le qualificatif *généralisé* correspond

Algorithme 3.4.1 Algorithme de Douglas-Rachford

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
 2. $s^{k+1} = [\frac{1}{2}I + \frac{1}{2}\mathcal{N}_2 \circ \mathcal{N}_1] s^k$
 3. $k \leftarrow k + 1$; Aller à l'étape 2
-

simplement aux autres choix de coefficients de relaxation α^k (algorithme 3.4.2).

L'opérateur intervenant dans l'algorithme 3.4.1 est en fait l'opérateur $\mathcal{P}_2 \star \mathcal{P}_1$, c'est à dire le résolvant de l'opérateur monotone maximal $\mathcal{T}_2 \odot \mathcal{T}_1$. Ce dernier est appelé *opérateur d'éclatement* dans [32] et [33]. Il permet d'interpréter l'algorithme de Douglas-Rachford comme un cas particulier de la méthode du point proximal.

Algorithme 3.4.2 Algorithme de Douglas-Rachford généralisé

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
 2. $s^{k+1} = [(1 - \alpha^k)I + \alpha^k(\mathcal{N}_2 \circ \mathcal{N}_1)] s^k$
 3. $k \leftarrow k + 1$; Aller à l'étape 2
-

Sa convergence est assurée par la proposition B.3.1 si la suite des α_k vérifie l'hypothèse B.3.1. Il existe une multitude de façons, toutes équivalentes entre elles, de mettre en oeuvre l'algorithme qui diffèrent uniquement par le choix des variables primales et duales

à moyenner. Nous en détaillons deux, les algorithmes 3.4.3 et 3.4.4. Introduisons d'abord dans l'algorithme 3.4.2 les variables intermédiaires :

$$\begin{aligned} d^{k+1} &= \mathcal{N}_1(s^k) \\ t^{k+1} &= \mathcal{N}_2(d^{k+1}) = [\mathcal{N}_2 \circ \mathcal{N}_1](s^k) \end{aligned}$$

Nous avons dans l'algorithme 3.4.3 :

$$\begin{aligned} M\tilde{y}^{k+1} &= \mathcal{P}_1(My^k + M^\top u^k) \\ M\tilde{u}^{k+1} &= \mathcal{Q}_1(My^k + M^\top u^k) \\ M\bar{y}^{k+1} &= \mathcal{P}_2(M\tilde{y}^{k+1} - M^\top \tilde{u}^{k+1}) \\ M\bar{u}^{k+1} &= -\mathcal{Q}_2(M\tilde{y}^{k+1} - M^\top \tilde{u}^{k+1}) \end{aligned}$$

et l'équivalence avec l'algorithme 3.4.2 est obtenue en posant pour tout k :

$$\begin{aligned} s^k &= My^k + M^\top u^k \\ d^k &= M\tilde{y}^{k+1} - M^\top \tilde{u}^{k+1} = \mathcal{N}_1(s^k) \\ t^k &= M\bar{y}^{k+1} + M^\top \bar{u}^{k+1} = \mathcal{N}_2(d^k) \end{aligned}$$

En ce qui concerne l'algorithme 3.4.4, nous avons :

$$\begin{aligned} M\bar{y}^{k+1} &= \mathcal{P}_2(My^k - M^\top u^k) \\ M\bar{u}^{k+1} &= -\mathcal{Q}_2(My^k - M^\top u^k) \\ My^{k+1} &= \mathcal{P}_1(M\hat{y}^{k+1} - M^\top \hat{u}^{k+1}) \\ Mu^{k+1} &= \mathcal{Q}_1(M\hat{y}^{k+1} - M^\top \hat{u}^{k+1}) \end{aligned}$$

et l'équivalence avec l'algorithme 3.4.2 est obtenue en posant :

$$\begin{aligned} s^k &= My^k + M^\top u^k \\ d^k &= My^k - M^\top u^k = \mathcal{N}_1(s^k) \\ t^k &= M\bar{y}^{k+1} + M^\top \bar{u}^{k+1} = \mathcal{N}_2(d^k) \end{aligned}$$

Nous différencions ces deux algorithmes en leur associant la lettre Δ et la lettre Y , repérant ainsi la géométrie formée par la suite des itérés en dimension 1, représentée sur les figures 3.4.1 et 3.4.2. Notons qu'en pratique, la requête $(y^0, u^0) \in gr(T_1)$ au début de l'algorithme 3.4.4 est superflue car elle est obtenue automatiquement dès $k \geq 1$.

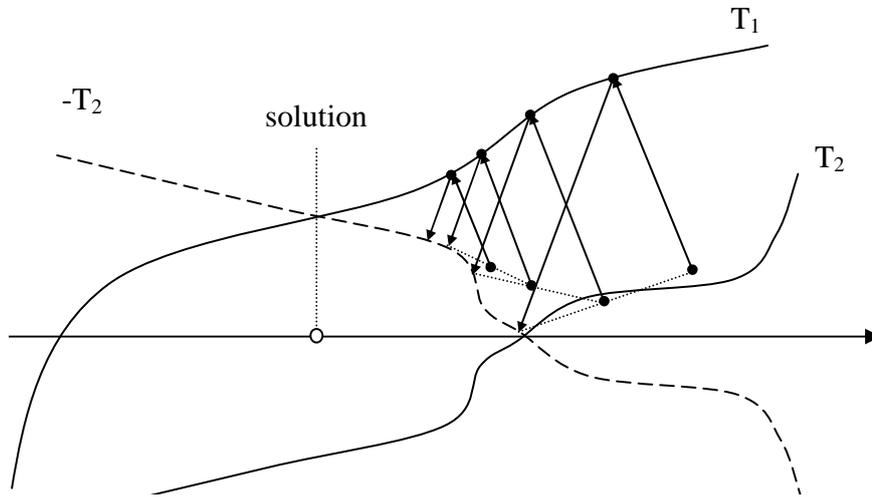
3.4.1 Application à la décomposition

Nous retranscrivons maintenant la spécialisation de l'algorithme 3.4.4 à la résolution du problème séparable (1.2). Comme pour l'algorithme 3.3.4, la matrice de mise à l'échelle doit être choisie bloc-diagonale afin de permettre la séparabilité du calcul de \mathcal{N}_1 .

Les étapes 2, 3 et 4 de cet algorithme tout comme les étapes 6 et 7 de l'algorithme 3.3.4 constituent une macro-étape appelée *coordination*. Elle s'interprète comme l'action d'un agent virtuel, le *coordonateur*, contrôlant la procédure d'optimisation. Les itérations successives de l'algorithme peuvent alors être vues comme une négociation entre le coordonateur et les sous-systèmes $i = 1, \dots, p$ dans lequel chaque partie va proposer successivement une décision y_i en s'adaptant aux propositions précédentes jusqu'à s'entendre.

Algorithme 3.4.3 Algorithme de Douglas-Rachford généralisé : version Δ

1. $k = 0$; Choisir $(y^0, u^0) \in \mathbb{R}^n \times \mathbb{R}^n$
2. Trouver \tilde{y}^{k+1} tel que $0 \in T_1(\tilde{y}^{k+1}) + \Lambda(\tilde{y}^{k+1} - y^k) - u^k$
3. $\tilde{u}^{k+1} = u^k + \Lambda(y^k - \tilde{y}^{k+1})$
4. Trouver \bar{y}^{k+1} tel que $0 \in T_2(\bar{y}^{k+1}) + \Lambda(\bar{y}^{k+1} - \tilde{y}^{k+1}) + \tilde{u}^{k+1}$
5. $\bar{u}^{k+1} = \tilde{u}^{k+1} - \Lambda(\tilde{y}^{k+1} - \bar{y}^{k+1})$
6. $(y^{k+1}, u^{k+1}) = (1 - \alpha^k)(\bar{y}^{k+1}, \bar{u}^{k+1}) + \alpha^k(y^k, u^k)$
7. $k \leftarrow k + 1$; Aller à l'étape 2

FIGURE 3.4.1 – Algorithme de Douglas-Rachford en Δ **Algorithme 3.4.4** Algorithme de Douglas-Rachford généralisé : version Y

1. $k = 0$; Choisir $(y^0, u^0) \in gr(T_1)$
2. Trouver \bar{y}^{k+1} tel que $0 \in T_2(\bar{y}^{k+1}) + \Lambda(\bar{y}^{k+1} - y^k) + u^k$
3. $\bar{u}^{k+1} = u^k - \Lambda(y^k - \bar{y}^{k+1})$
4. $(\hat{y}^{k+1}, \hat{u}^{k+1}) = (1 - \alpha^k)(y^k, u^k) + \alpha^k(\bar{y}^{k+1}, \bar{u}^{k+1})$
5. Trouver y^{k+1} tel que $0 \in T_1(y^{k+1}) + \Lambda(y^{k+1} - \hat{y}^{k+1}) - \hat{u}^{k+1}$
6. $u^{k+1} = \bar{u}^{k+1} - \Lambda(\bar{y}^{k+1} - y^{k+1})$
7. $k \leftarrow k + 1$; Aller à l'étape 2

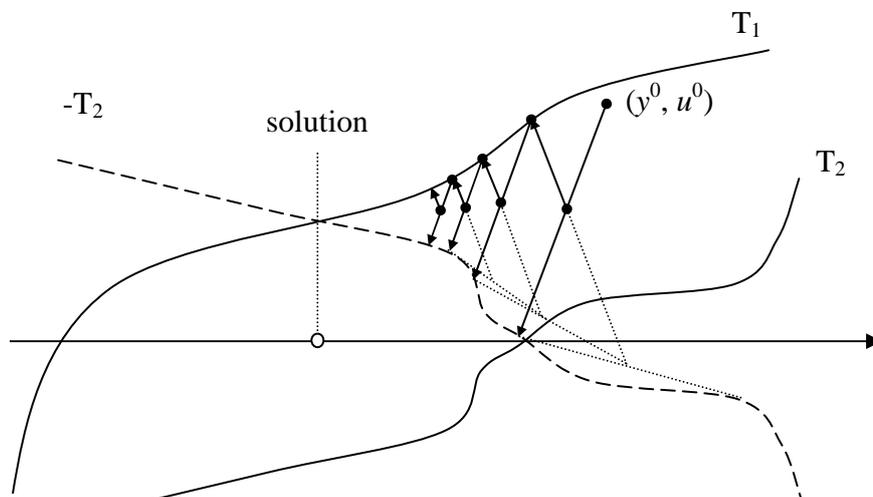


FIGURE 3.4.2 – Algorithme de Douglas-Rachford généralisé en Y

Algorithme 3.4.5 Algorithme de Douglas-Rachford généralisé (version Y) appliqué à la décomposition

1. $k = 0$; Choisir $(y^0, u^0) \in \mathbb{R}^n \times \mathbb{R}^n$
 2. $\bar{y}^{k+1} = \arg \min_y C(y) + \frac{1}{2} \langle y - y^k, \Lambda(y - y^k) \rangle + \langle u^k, y \rangle$
 3. $\bar{u}^{k+1} = u^k - \Lambda(y^k - \bar{y}^{k+1})$
 4. $(\hat{y}^{k+1}, \hat{u}^{k+1}) = (1 - \alpha^k)(y^k, u^k) + \alpha^k(\bar{y}^{k+1}, \bar{u}^{k+1})$
 5. Pour $i = 1, \dots, p$ faire
 6. $y_i^{k+1} \in \arg \min_{y_i} F_i(y_i) + \frac{1}{2} \langle y_i - \hat{y}_i^k, \Lambda_i(y_i - \hat{y}_i^k) \rangle - \langle \hat{u}_i^k, y_i \rangle$
 7. $u_i^{k+1} = \hat{u}_i^k + \Lambda_i(\hat{y}_i^{k+1} - y_i^{k+1})$
 8. Fin pour
 9. $k \leftarrow k + 1$; Aller à l'étape 2
-

Relaxation séquentielle et méthode des directions alternées

L'algorithme 3.4.5 peut s'interpréter comme une relaxation séquentielle de la méthode de lagrangien augmenté. Soit le problème d'optimisation :

$$\min_x f(x) + g(y) \quad (3.12)$$

$$\text{s.c. } Gx - y = 0 \quad (3.13)$$

Le lagrangien augmenté de paramètre λ associé à la dualisation de la contrainte s'écrit :

$$L(x, y; u) = f(x) + g(y) - \langle u, Gx - y \rangle + \frac{\lambda}{2} \|Gx - y\|^2$$

et l'algorithme de lagrangien augmenté 2.2.6 devient l'algorithme 3.4.6. Il n'est pas sépa-

Algorithme 3.4.6 Algorithme de lagrangien augmenté

1. $k = 0$; Choisir $u^0 \in \mathbb{R}^m$
 2. $(x^{k+1}, y^{k+1}) = \arg \min_{x,y} f(x) + g(y) - \langle u^k, Gx - y \rangle + \frac{\lambda}{2} \|Gx - y\|^2$
 3. $u^{k+1} = u^k - \lambda(Gx^{k+1} - y^{k+1})$
 4. $k \leftarrow k + 1$; Aller à l'étape 2
-

table; l'étape 2 est structurellement aussi difficile à résoudre que le problème de départ. Une manière de décomposer la résolution en faisant intervenir f et g indépendamment est d'effectuer une relaxation séquentielle. La méthode des directions alternées (algorithme 3.4.7) propose de résoudre le problème de l'étape 2 en deux sous-étapes, d'abord par rapport à x à (y^k, u^k) fixé, puis par rapport à y en fixant x^{k+1} et u^k . Différentes applications

Algorithme 3.4.7 Méthode des directions alternées

1. Choisir $y^0, u^0 \in \mathbb{R}^n$
 2. $x^{k+1} = \arg \min_x f(x) - \langle u^k, Gx \rangle + \frac{\lambda}{2} \|Gx - y^k\|^2$
 3. $y^{k+1} = \arg \min_y g(y) + \langle u^k, y \rangle + \frac{\lambda}{2} \|y - Gx^{k+1}\|^2$
 4. $u^{k+1} = u^k - \lambda(Gx^{k+1} - y^{k+1})$
 5. $k \leftarrow k + 1$; Aller à l'étape 2
-

sont étudiées dans [52]. La filiation entre cet algorithme et la méthode 3.4.2 d'éclatement de Douglas-Rachford est établie dans [32].

3.4.2 Méthodes de l'inverse partielle ou de décomposition proximale

En pratique, beaucoup de modèles d'optimisation de la forme (1.2) mènent vers un opérateur T_2 égal au cône normal d'un sous-espace vectoriel \mathcal{A} de \mathbb{R}^n . Les conditions d'optimalité 3.1 avec $\mathcal{T} = T_1$ se résument alors à :

$$\text{Trouver } y^* \in \mathcal{A}, u^* \in \mathcal{A}^\perp \text{ tel que } u^* \in \mathcal{T}(y^*) \quad (3.14)$$

Opérateur et méthode de l'inverse partielle

La *méthode de l'inverse partielle* développée par Spingarn dans [113] est spécialisée dans la résolution du problème d'inclusion (3.14). Elle est basée sur l'opérateur $\mathcal{T}_{\mathcal{A}}$ de la définition 3.4.1.

Définition 3.4.1. *L'inverse partielle d'un opérateur \mathcal{T} par rapport à l'espace vectoriel \mathcal{A} est l'opérateur dont le graphe est défini par :*

$$gr(\mathcal{T}_{\mathcal{A}}) = \{(y_{\mathcal{A}} + u_{\mathcal{A}^{\perp}}, y_{\mathcal{A}^{\perp}} + u_{\mathcal{A}}), u \in \mathcal{T}(y)\}$$

où les indices $_{\mathcal{A}}$ et $_{\mathcal{A}^{\perp}}$ indiquent les projections sur \mathcal{A} et \mathcal{A}^{\perp} .

Il s'agit d'inverser uniquement les composantes suivant le sous-espace vectoriel \mathcal{A}^{\perp} des éléments de départ et de leurs images. Cette transformation généralise l'effet de la conjugaison partielle (A.9) obtenu en prenant $\mathcal{A} = \mathbb{R}^n \times \{0_m\}$. $\mathcal{T}_{\mathcal{A}}$ est maximal monotone si et seulement si \mathcal{T} l'est lui-même et par construction, nous avons :

$$0 \in \mathcal{T}_{\mathcal{A}}(s) \iff s_{\mathcal{A}^{\perp}} \in \mathcal{T}(s_{\mathcal{A}})$$

Autrement dit, les projections sur \mathcal{A} et sur \mathcal{A}^{\perp} d'un zéro de $\mathcal{T}_{\mathcal{A}}$ sont respectivement des solutions primale et duale du problème (3.14). La méthode de l'inverse partielle (algorithme 3.4.8) consiste alors à appliquer l'algorithme du point proximal à $\mathcal{T}_{\mathcal{A}}$. Le problème

Algorithme 3.4.8 Méthode de l'inverse partielle

1. Choisir $s^0 \in \mathbb{R}^n$
 2. $s^{k+1} = (I + \mathcal{T}_{\mathcal{A}})^{-1}(s^k)$
 3. $k \leftarrow k + 1$; Aller à l'étape 2
-

d'inclusion de l'étape 2 se réécrit :

$$s^k - s^{k+1} \in \mathcal{T}_{\mathcal{A}}(s^{k+1})$$

soit, par construction de $\mathcal{T}_{\mathcal{A}}$:

$$\begin{aligned} (s^k - s^{k+1})_{\mathcal{A}} + s_{\mathcal{A}^{\perp}}^{k+1} &\in \mathcal{T}(s_{\mathcal{A}}^{k+1} + (s^k - s^{k+1})_{\mathcal{A}^{\perp}}) \\ 0 &\in \mathcal{T}(s_{\mathcal{A}}^{k+1} - s_{\mathcal{A}^{\perp}}^{k+1} + s_{\mathcal{A}^{\perp}}^k) + s_{\mathcal{A}}^{k+1} - s_{\mathcal{A}^{\perp}}^{k+1} - s_{\mathcal{A}}^k \end{aligned}$$

en posant $\tilde{y}^{k+1} = s_{\mathcal{A}}^{k+1} - s_{\mathcal{A}^{\perp}}^{k+1} + s_{\mathcal{A}^{\perp}}^k$, ce problème peut encore se réécrire :

$$\begin{aligned} 0 &\in \mathcal{T}(\tilde{y}^{k+1}) + \tilde{y}^{k+1} - s^k \\ s^{k+1} &= \Pi_{\mathcal{A}}\tilde{y}^{k+1} + \Pi_{\mathcal{A}^{\perp}}(s^k - \tilde{y}^{k+1}) \end{aligned}$$

En décomposant s^k comme la somme de deux éléments primal $y^k = s_{\mathcal{A}}^k$ et dual $u^k = s_{\mathcal{A}^{\perp}}^k$, la méthode prend enfin la forme : Plusieurs applications sont présentées dans [113], [114], [115], [116] et [49]. Étant donné que l'algorithme du point proximal converge en un nombre fini d'itérations lorsque $T = \partial F$ où F est une fonction polyédrale, certains auteurs ([56], [55] et [22]) se sont posés la question de la convergence finie de la méthode de l'inverse partielle dans le même contexte. La réponse est un peu décevante; il ne suffit pas que F soit polyédrale pour que l'algorithme appliqué à $(\partial F)_{\mathcal{A}}$ converge en un nombre fini

Algorithme 3.4.9 Méthode de l'inverse partielle (version détaillée)

1. Choisir $y^0, u^0 \in \mathbb{R}^n$
2. Trouver \tilde{y}^{k+1} tel que $0 \in \mathcal{T}(\tilde{y}^{k+1}) + \tilde{y}^{k+1} - y^k - u^k$
3. $y^{k+1} = \Pi_{\mathcal{A}} \tilde{y}^{k+1}$
4. $u^{k+1} = u^k - \Pi_{\mathcal{A}^\perp} \tilde{y}^{k+1}$
5. $k \leftarrow k + 1$; Aller à l'étape 2

d'étapes, une condition nécessaire consiste en la non vacuité de l'intérieur relatif à \mathcal{A} (ou à \mathcal{A}^\perp) des solutions primales (ou duales). Nous mentionnons enfin le développement de calculs approchés des itérations grâce à des méthodes de plans coupants dans [82] puis dans [11].

L'opérateur d'inverse partielle est aussi donné par la formule :

$$\mathcal{T}_{\mathcal{A}} = \mathbf{N}_{\mathcal{A}} \odot \mathcal{T}$$

ce qui classe cette méthode comme un cas particulier de la méthode de Douglas-Rachford appliquée au couple $(\mathbf{N}_{\mathcal{A}}, \mathcal{T})$. Notons d'ailleurs que l'opération \odot s'exprime tout comme l'inverse partielle en termes de graphe :

$$gr(\mathcal{T}_2 \odot \mathcal{T}_1) = \{(y - Q_2(y - u), u + Q_2(y - u)) / u \in T_1(y)\}$$

Méthode de décomposition proximale

Notons $\mathcal{P} = (I + \mathcal{T})^{-1}$ et $\mathcal{Q} = I - \mathcal{P}$ les opérateurs résolvant et régularisé de \mathcal{T} . Le problème (3.14) peut se réécrire :

$$\text{Trouver } s \in \mathbb{R}^n \text{ tel que : } \begin{cases} \mathcal{P}(s) \in \mathcal{A} \\ \mathcal{Q}(s) \in \mathcal{A}^\perp \end{cases}$$

Si nous notons $y = \mathcal{P}(s)$ et $u = \mathcal{Q}(s)$, ceci est équivalent d'après la proposition 2.3.2 à :

$$\text{Trouver } y, u \in \mathbb{R}^n \text{ tel que : } \begin{cases} \mathcal{P}(y + u) = y \in \mathcal{A} \\ \mathcal{Q}(y + u) = u \in \mathcal{A}^\perp \end{cases}$$

En utilisant le fait qu'un vecteur appartient à un sous-espace si et seulement s'il est égal à sa projection sur ce sous-espace, on peut encore réécrire (3.14) sous la forme :

$$\text{Trouver } y, u \in \mathbb{R}^n \text{ tel que : } \begin{cases} y = \Pi_{\mathcal{A}} \mathcal{P}(y + u) \\ u = \Pi_{\mathcal{A}^\perp} \mathcal{Q}(y + u) \end{cases} \quad (3.15)$$

L'opérateur :

$$\mathcal{D} : (y, u) \mapsto (\Pi_{\mathcal{A}} \mathcal{P}(y + u), \Pi_{\mathcal{A}^\perp} \mathcal{Q}(y + u))$$

dont les points fixes sont les solutions de (3.15), est prouvé dans [73] être pseudo-contractant sur l'espace $\mathcal{A} \times \mathcal{A}^\perp$. La méthode de décomposition proximale consiste à effectuer des applications successives de \mathcal{D} . Étant donné un couple (y^k, u^k) , une itération calcule donc la décomposition de Moreau $(\tilde{y}^{k+1}, \tilde{u}^{k+1})$ de $s^k = y^k + u^k$ avant de la projeter sur $\mathcal{A} \times \mathcal{A}^\perp$. Cet

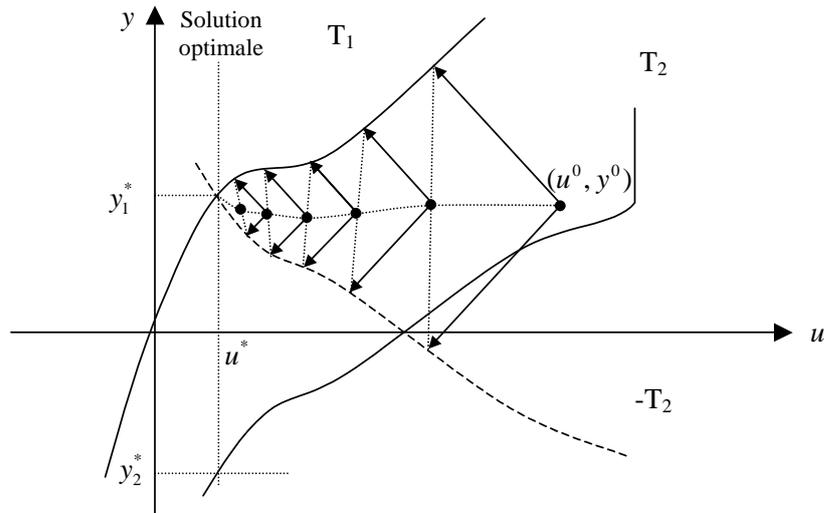


FIGURE 3.4.3 – Schéma des itérés de la méthode de Spingarn pour deux opérateurs

algorithmique donne un autre point de vue sur la méthode de l'inverse partielle à laquelle il est équivalent. Appliquée à des problèmes d'optimisation à structure décomposable ([41], [72], [31]), la méthode s'interprète comme une méthode de décomposition par les prix augmentée par des termes de "freinage" qui interdisent de trop s'éloigner de la solution courante.

Éclatement d'une somme quelconque d'opérateurs

Le problème d'inclusion (3.14) permet en particulier de traiter le problème suivant qui fait intervenir une somme de p opérateurs monotones maximaux :

$$\text{Trouver } y^* \in \mathbb{R}^n \text{ tel que } 0 \in T_1(y^*) + T_2(y^*) + \dots + T_p(y^*) \quad (3.16)$$

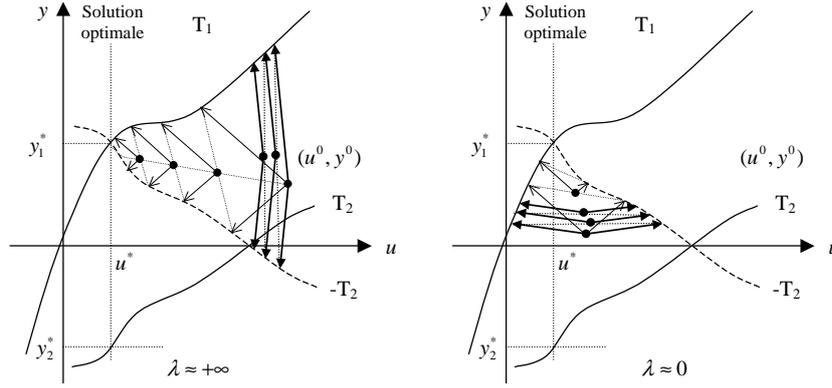
en nous plaçant dans l'espace plus grand \mathbb{R}^{np} , et en considérant l'opérateur produit $\mathcal{T} = T_1 \times T_2 \times \dots \times T_p$ qui est monotone maximal :

$$\begin{aligned} \mathcal{T} : \mathbb{R}^{np} &\rightarrow \mathbb{R}^{np} \\ z &\mapsto (T_1(z_1), T_2(z_2), \dots, T_p(z_p)) \end{aligned}$$

ainsi que l'espace \mathcal{A} :

$$\mathcal{A} = \{z \in \mathbb{R}^{np} / z_1 = z_2 = \dots = z_p\}$$

nous pouvons alors réécrire le problème sous la forme (3.14). Pour le cas de deux opérateurs ($p = 2$), la suite des itérés est représentée graphiquement sur la figure 3.4.3. Comme pour les méthodes précédentes, il est possible d'introduire un facteur d'échelle λ . Contrairement à l'algorithme du point proximal, le choix du facteur d'échelle est ici délicat. L'exemple schématisé sur la figure 3.4.4 montre qu'un choix trop grand ou bien trop petit conduira à une convergence ralentie par rapport à un choix "compromis".


 FIGURE 3.4.4 – Influence du paramètre λ sur la convergence de la méthode de Spingarn.

3.4.3 Vitesse de convergence

Supposons que la suite (z^k, w^k) converge vers (\bar{z}, \bar{w}) ; s^k converge vers $\bar{s} = \bar{z} + \bar{w}$ et $d^k = \mathcal{N}_1(s^k)$ converge vers $\bar{d} = \bar{z} - \bar{w} = \mathcal{N}_2(\bar{s})$. L'opérateur effectuant une itération est noté :

$$\mathcal{J}_{\alpha^k} = (1 - \alpha^k)I + \alpha^k \mathcal{N}_2 \circ \mathcal{N}_1 \quad (3.17)$$

Si \mathcal{T}_1 et \mathcal{T}_2 sont proto-différentiables en $(\bar{z}|\bar{w})$ et $(\bar{z}|\bar{w})$, alors \mathcal{N}_1 et \mathcal{N}_2 sont semi-différentiables en \bar{s} et \bar{d} et \mathcal{J}_{α^k} l'est aussi en \bar{s} avec :

$$D\mathcal{J}_{\alpha^k}(\bar{s}) = (1 - \alpha^k)I + \alpha^k D\mathcal{N}_2(\bar{d}) \circ D\mathcal{N}_1(\bar{s})$$

et le taux de convergence de l'algorithme est borné d'après (2.17) par :

$$\limsup_k \frac{\|e^{k+1}\|}{\|e^k\|} \leq \limsup_k \|D\mathcal{J}_{\alpha^k}(\bar{s})\| \quad (3.18)$$

3.4.4 Applications à la gestion de production

Nous passons en revue les diverses applications possibles de l'algorithme de Douglas-Rachford au problème (1.32).

Décomposition des aléas : méthode de couverture progressive (progressive hedging)

La méthode de Douglas-Rachford a déjà été appliquée pour résoudre des problèmes d'optimisation stochastique. Dans [97], le modèle (1.21) est décomposé grâce à la méthode de décomposition proximale où l'espace couplant est celui des décisions non-anticipatives défini par les contraintes (1.21b). En notant $\mathfrak{S}_t(s) = \{s' / \pi_t^{(s)} = \pi_t^{(s')}\}$, cet espace s'écrit :

$$\mathcal{A} = \left\{ y = (y_1^{(1)}, \dots, y_t^{(s)}, \dots, y_T^{(S)}) / \forall t \quad s' \in \mathfrak{S}_t(s) \implies y_t^{(s)} = y_t^{(s')} \right\}$$

dont l'orthogonal est :

$$\mathcal{A}^\perp = \left\{ u = (u_1^{(1)}, \dots, u_t^{(s)}, \dots, u_T^{(S)}) / \forall t, s \quad \sum_{s' \in \mathfrak{S}_t(s)} u_t^{(s')} = 0 \right\}$$

Algorithme 3.4.10 Algorithme de couverture progressive

-
1. Choisir $(y, u) \in \mathcal{A} \times \mathcal{A}^\perp$
 2. Pour $s = 1, \dots, S$ faire
 3. $x^{(s)} \leftarrow \arg \min_{x^{(s)}} \rho_s f(x^{(s)}; \xi^{(s)}) + \frac{\rho_s}{2} \|x^{(s)} - y^{(s)}\|^2 - \langle u^{(s)}, x^{(s)} \rangle$
 4. fin pour
 5. Pour $t = 1, \dots, T$ faire
 6. Pour $s = 1, \dots, S$ faire
 7.
$$y_t^{(s)} = \left(\sum_{s' \in \mathfrak{G}_t(s)} \rho^{(s')} \right)^{-1} \sum_{s' \in \mathfrak{G}_t(s)} \rho^{(s')} x_t^{(s')}$$
 8. fin pour
 9. fin pour
 10. $u \leftarrow u - \Lambda(x - y)$
-

donnant ainsi lieu à l'algorithme 3.4.10.

L'étape 3 consiste à minimiser le coût du scénario s en tenant compte de la solution non-anticipative de référence $y^{(s)}$ et du prix $u^{(s)}$. La non-anticipativité est ensuite obtenue progressivement au cours des itérations, d'où le nom de *couverture progressive*. L'étape 5 de projection consiste simplement, pour chaque nœud de l'arbre, à calculer la moyenne des variables de décision, c'est à dire à calculer l'espérance conditionnelle sachant le passé des observations.

Cette approche a ensuite été étendue dans [91] pour gérer des contraintes couplantes supplémentaires sur les scénarios. L'exemple développé dans cette référence possède par exemple des contraintes faisant intervenir l'espérance de certaines variables. Dans [107], un problème de gestion de réserve hydraulique sur un arbre de scénarios est étudié. Il s'agit d'un problème de point-selle séparé en deux grâce à la méthode de l'inverse partielle via la reformulation préalable (3.16). Le choix spécial de l'éclatement donne lieu d'une part à un problème dynamique linéaire quadratique sans contrainte pouvant être résolu par programmation dynamique et d'autre part à des problèmes mono-dimensionnels avec contraintes de bornes résolus analytiquement. Ces deux derniers modèles sont repris et expérimentés dans [87].

Décomposition en espace : lagrangien augmenté séparable

L'algorithme de *lagrangien augmenté séparable* ([41, 31]) peut être vu comme une généralisation (au cas non convexe) de la méthode de décomposition proximale appliquée au problème de partage de ressources (3.6), avec le même choix d'éclatement qu'à la page 48. Il s'agit en quelque sorte d'un intermédiaire entre les algorithmes 3.2.4 de décomposition par les prix et 3.2.3 de décomposition par les ressources. Il peut être appliquée pour découpler le problème 1.32 en espace, c'est à dire centrale par centrale $i = 1, \dots, p$. L'étape 3 consiste alors à optimiser individuellement la gestion des centrales. Le vecteur y_i joue le rôle d'un planning de production (sur l'ensemble des scénarios) dont on ne veut pas trop s'écarter et u est homogène à un prix de l'électricité. Lors de l'étape de coordination, la quantité r (étape 5) représente le défaut de production pour satisfaire la demande. Cette charge de production manquante est alors partagée entre les centrales à

Algorithme 3.4.11 Algorithme de lagrangien augmenté séparable

1. Choisir $(y, u) \in \mathcal{A} \times \mathcal{A}^\perp$
2. Pour $i = 1, \dots, p$ faire
3. $x_i \leftarrow \arg \min_{x_i} f_i(x_i) + \frac{1}{2} \langle G_i x_i - y_i, \Lambda_i (G_i x_i - y_i) \rangle - \langle u, G_i x_i \rangle$
4. fin pour
5. $r \leftarrow R - \sum_{i=1}^p G_i x_i$
6. Pour $i = 1, \dots, p$ faire
7. $y_i \leftarrow G_i x_i + \Lambda_i^{-1} (\sum_{j=1}^p \Lambda_j^{-1})^{-1} r$
8. fin pour
9. $v \leftarrow v + (\sum_{j=1}^p \Lambda_j^{-1})^{-1} r$

l'étape 7. Enfin, le prix u est mis à jour dans le but d'inciter ou de décourager les centrales à produire suivant le signe de r .

Lors de l'étape 7, les centrales possédant des poids Λ_i faibles recevront une charge de production plus élevée que les autres. Cette fonctionnalité est absente de la version classique de l'algorithme ([41],[31]); nous la baptisons *mise à l'échelle globale*. Elle permet de pondérer l'importance de chaque sous-système vis à vis de la redistribution des quantités violées dans le couplage. Cette possibilité est étudiée sur des exemples dans [64] et d'un point de vue théorique dans [65]. Nous mentionnons ici l'intérêt pratique d'une telle mise à l'échelle "distribuée" sur un exemple en dimension 4. Il s'agit de minimiser la somme de deux formes quadratiques $x_1 \mapsto \frac{1}{2} x_1^\top Q_1 x_1 + c_1^\top x_1$ et $x_2 \mapsto \frac{1}{2} x_2^\top Q_2 x_2 + c_2^\top x_2$ sous la contrainte $G_1 x_1 + G_2 x_2 = b_1 + b_2$ avec les données numériques :

$$\begin{aligned}
 Q_1 &= \begin{pmatrix} 100 & 50 \\ 50 & 100 \end{pmatrix}, Q_2 = \begin{pmatrix} 1 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{pmatrix} \\
 c_1 &= \begin{pmatrix} 1 \\ 2 \end{pmatrix}, c_2 = \begin{pmatrix} 3 \\ 4 \end{pmatrix} \\
 G_1 &= G_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\
 b_1 &= b_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}
 \end{aligned}$$

La figure 3.4.5 reporte à gauche le nombre d'itérations obtenu avec un facteur d'échelle par sous-problème, c'est à dire $\Lambda_i = \lambda_i I_2$, pour divers couples (λ_1, λ_2) ; et à droite le cas d'un unique facteur d'échelle, que l'on peut voir comme la restriction aux couples vérifiant $\lambda_1 = \lambda_2$ (*i.e.* au plan vert). Il est clair sur cette instance que les performances de l'algorithme peuvent être améliorées en autorisant $\lambda_1 \neq \lambda_2$. Une analyse plus approfondie sur le rôle des Λ_i est effectuée dans la section 4.3.

Conclusion

Ce chapitre a joué un rôle dual. Le premier est celui de support de présentation des méthodes d'éclatement et de leurs applications à l'optimisation. Son deuxième rôle a été d'éclaircir certains points techniques souvent laissés au lecteur dans la littérature tels que

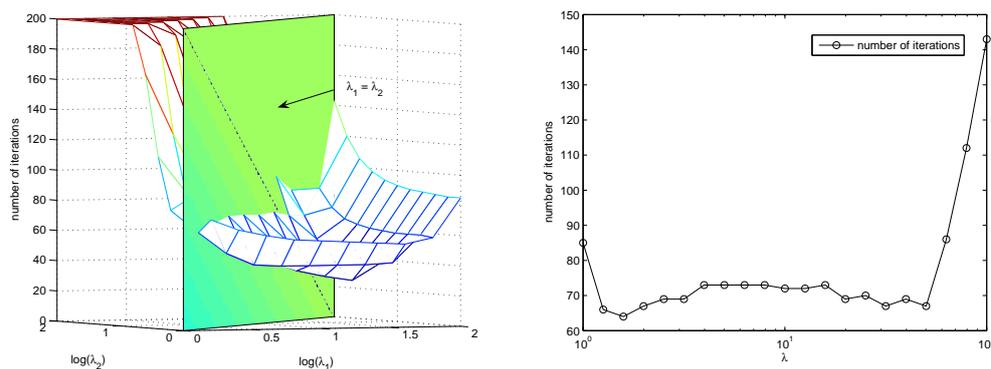


FIGURE 3.4.5 – Nombre d’itérations avec un facteur d’échelle par sous-problème puis avec un facteur d’échelle unique.

le fait qu’on peut se contenter d’un seul facteur d’échelle λ plutôt que deux paramètres multiplicatifs a et b ou encore la mise en avant de différentes façons d’implanter la méthode de Douglas-Rachford (dont les versions Δ et Y). Les formulations (3.7) et (3.18) des taux de convergence grâce à la semi-différentielle est aussi à notre connaissance nouvelle et sera réutilisée intensivement dans les sections 4.3 et 4.4 du chapitre 4. C’est justement dans ce chapitre que nous allons développer l’essentiel de nos contributions en tentant de répondre à la question : “Comment accélérer la convergence de ce type de méthodes?”.

Chapitre 4

Accélération de la convergence

4.1 Introduction

L’algorithme de Douglas-Rachford généralisé 3.4.2 est usuellement appliqué au problème :

$$\begin{aligned} & \text{Trouver } z^* \text{ tel que } 0 \in \mathcal{T}_1(z^*) + \mathcal{T}_2(z^*) \\ & \mathcal{T}_1(\cdot) = M^{-\top} T_1(M^{-1}\cdot) \\ & \mathcal{T}_2(\cdot) = M^{-\top} T_2(M^{-1}\cdot) \end{aligned}$$

(voir (3.11)) avec le choix $M^\top M = \Lambda = \lambda I$. Il a été reporté dans de nombreuses études numériques ([33, 73, 41, 72, 26, 111, 107]) que la vitesse de convergence dépend très fortement de λ . Nous avons tracé sur la figure 4.1.1 un exemple typique de la variation du nombre d’itérations en fonction de ce choix. Pour des valeurs trop élevées ou trop faibles, il croît fortement et il existe une plage de valeurs compromis pour lesquelles la méthode atteint ses meilleures performances. Cette observation nous conduit dans la section 4.2 à étudier la convergence de la méthode lorsque l’on désire modifier la matrice $\Lambda = M^\top M$ au cours de l’algorithme pour corriger un éventuel mauvais choix initial.

Le comportement vis à vis de λ est analysé d’un point de vue théorique dans [68] puis dans [73] et expliqué partiellement lorsque T_1 est à la fois lipschitz de constante L et fortement monotone de module a . La moyenne géométrique \sqrt{aL} des deux constantes est alors une bonne valeur pour λ . Il semble donc que le facteur d’échelle Λ devrait refléter une certaine “croissance moyenne” de T_1 ou encore une croissance moyenne à l’ordre 2 de F si $T_1 = \partial F$. Nous confirmons cette intuition dans la section 4.3 et proposons une procédure de mise à jour automatique de Λ à partir d’informations d’ordre 2 obtenues au cours des itérations.

D’autre part, dans la plupart des applications, comprenant la programmation linéaire ou quadratique de grande taille une convergence (linéaire) s’effectuant en spirale est observée ([55], [33], [34]); les composantes des itérés oscillent autour de leur valeur limite. La figure 4.1 représente ce phénomène pour un exemple en dimension 4. Cette géométrie particulière expliquée dans [55] pour des problèmes d’optimisation polyédrale est étudiée dans la section 4.4. Nous proposons notamment une adaptation de l’algorithme permettant d’accélérer la convergence dans certaines situations.

Les analyses des sections 4.3 et 4.4 sont menées dans un cadre unifié et reposent sur les résultats généraux de convergence présent en annexe B et dans la section 3.4.3. Ces résultats se basent essentiellement sur la notion de proto-différentiabilité présentée à la section 2.3 appliquée aux opérateurs de réflexions généralisées \mathcal{N}_2 et \mathcal{N}_1 intervenant dans l’algorithme.

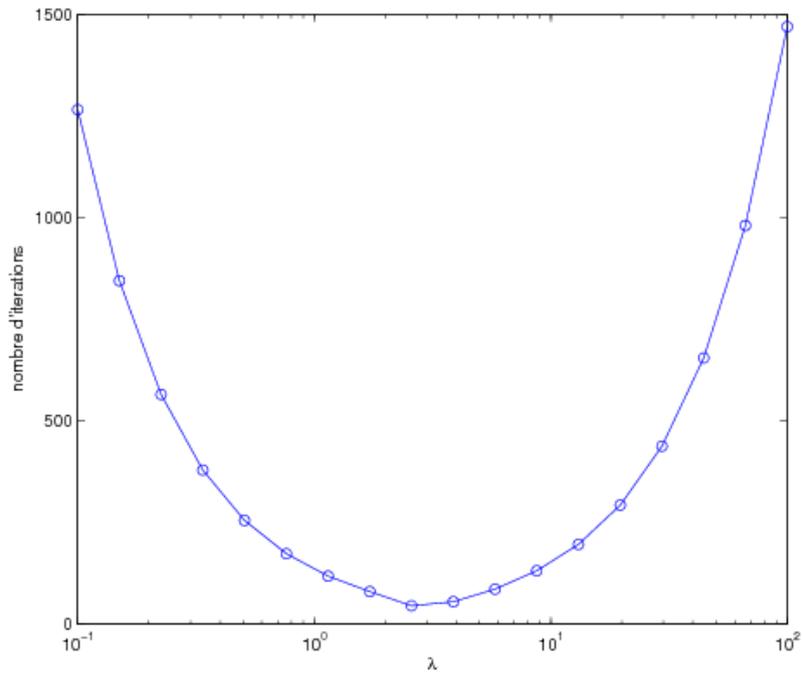
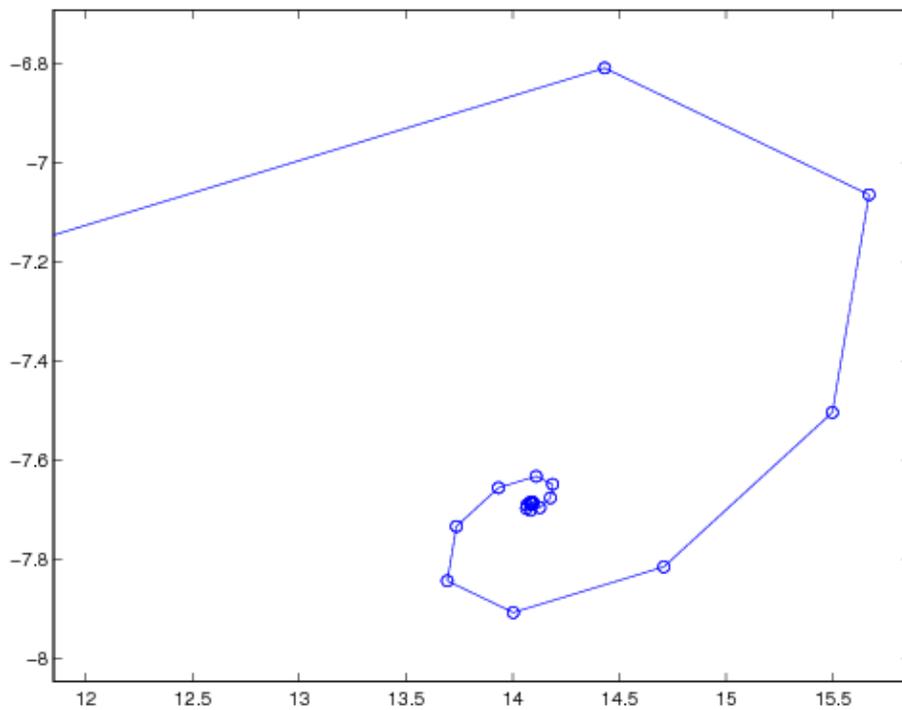
FIGURE 4.1.1 – Profil typique du nombre d'itérations en fonction du paramètre λ .

FIGURE 4.1.2 – Comportement de convergence oscillatoire typique

Enfin dans la section 4.5, nous nous intéressons à la possibilité de ne résoudre qu'une partie des sous-problèmes. Nous analysons la convergence dans ce cas et discutons des résultats obtenus numériquement.

4.2 Facteur d'échelle variable

Il est admis que le choix du facteur d'échelle se révèle dans la pratique crucial pour obtenir des taux de convergence raisonnables. Il serait donc intéressant de pouvoir changer sa valeur à chaque itération, d'une part pour corriger un choix initial difficile à effectuer et d'autre part car la valeur "idéale" de λ est sans doute inhomogène dans l'espace. Un prérequis est donc l'étude de la convergence lorsque \mathcal{T}_1 et \mathcal{T}_2 sont remplacés à l'itération k par :

$$\begin{aligned}\mathcal{T}_1^k &= M_k^{-\top} \mathcal{T}_1(M_k^{-1} \cdot) \\ \mathcal{T}_2^k &= M_k^{-\top} \mathcal{T}_2(M_k^{-1} \cdot)\end{aligned}$$

La principale difficulté réside dans l'observation suivante. Dans l'algorithme classique, c'est à dire lorsque $M^k \equiv M$, la solution est cherchée sous la forme d'une somme d'éléments primal et dual $s = My + M^\top u$ et génère le couple (y, u) comme produit dérivé. La convergence globale est alors assurée par le fait que les opérateurs $\mathcal{J}_{\alpha^k} = \alpha^k I + (1 - \alpha^k)\mathcal{N}_1 \circ \mathcal{N}_2$ définis en (3.17) partagent le même ensemble de points fixes qui dépend de M :

$$\text{fix}(\mathcal{J}_{\alpha^0}) = \dots = \text{fix}(\mathcal{J}_{\alpha^k}) = \dots = \left\{ My^* + M^{-\top} u^* / \begin{cases} u^* & \in T_1(y^*) \\ -u^* & \in T_2(y^*) \end{cases} \right\}$$

Le fait de changer le facteur d'échelle M^k à chaque itération modifie par la même occasion l'ensemble des points fixes traqué par l'algorithme. Autrement dit, en notant \mathcal{N}_1^k et \mathcal{N}_2^k les opérateurs faiblement contractants associés à \mathcal{T}_1^k et à \mathcal{T}_2^k et

$$\mathcal{J}_{\alpha^k}^k = \alpha^k I + (1 - \alpha^k)\mathcal{N}_1^k \circ \mathcal{N}_2^k \quad (4.1)$$

l'opérateur effectuant l'itération k , alors $\text{fix}(\mathcal{J}_{\alpha^k}^k)$ varie aussi à chaque itération :

$$\text{fix}(\mathcal{J}_{\alpha^k}^k) = \left\{ M_k y^* + M_k^{-\top} u^* / \begin{cases} u^* & \in T_1(y^*) \\ -u^* & \in T_2(y^*) \end{cases} \right\}$$

Par conséquent, si nous notons s^k et \bar{s}^{k+1} les sommes primal-dual en entrée et en sortie de l'itération k :

$$\bar{s}^{k+1} = \mathcal{J}_{\alpha^k}^k(s^k)$$

\bar{s}^{k+1} sera plus proche de l'ensemble $\text{fix}(\mathcal{J}_{\alpha^k}^k)$ que ne l'est s^k (voir formule (4.2) plus bas), mais rien ne garantit qu'il se rapproche aussi de $\text{fix}(\mathcal{J}_{\alpha^{k'}}^{k'})$ pour $k' \geq k$ ce qui peut compromettre la convergence. Pour effectuer la transition entre deux itérations successives, nous aurons recours au couple (y^{k+1}, u^{k+1}) servant dans les calculs intermédiaires et vérifiant :

$$\bar{s}^{k+1} = M_k y^{k+1} + M_k^{-\top} u^{k+1}$$

pour former l'itérée suivante :

$$s^{k+1} = M_{k+1} y^{k+1} + M_{k+1}^{-\top} u^{k+1}$$

en changeant simplement la pondération de la somme.

4.2.1 Algorithme

Le choix entre les formulations Δ et Y n'est ici plus anodin puisque nous travaillons maintenant aussi sur le couple (y, u) . Nous étudions une adaptation de la formulation Y 3.4.4 qui présente l'avantage de manipuler une suite (y^k, u^k) incluse dans $gr(T_1)$ ce qui facilite l'analyse.

Algorithme 4.2.1 Méthode de Douglas-Rachford généralisée avec facteur d'échelle variable

1. $k = 0$; Choisir $(y^0, u^0) \in \mathbb{R}^n$
 2. Trouver \bar{y}^{k+1} tel que $0 \in T_2(\bar{y}^{k+1}) + \Lambda_k(\bar{y}^{k+1} - y^k) + u^k$
 3. $\bar{u}^{k+1} = u^k - \Lambda_k(y^k - \bar{y}^{k+1})$
 4. $(\hat{y}^{k+1}, \hat{u}^{k+1}) := \alpha^k(y^k, u^k) + (1 - \alpha^k)(\bar{y}^{k+1}, \bar{u}^{k+1})$
 5. Trouver y^{k+1} tel que $0 \in T_1(y^{k+1}) + \Lambda_k(y^{k+1} - \hat{y}^{k+1}) - \hat{u}^{k+1}$
 6. $u^{k+1} = \hat{u}^{k+1} + \Lambda_k(\hat{y}^{k+1} - y^{k+1})$
 7. $k \leftarrow k + 1$; Aller à l'étape 2
-

De telle méthodes à métriques variables ont déjà été introduites dans la littérature ([72, 53, 43]). Dans [53], une matrice variable est employée dans la méthode des directions alternées. La convergence est prouvée sous l'hypothèse que la différence entre deux matrices successives est semi-définie positive à partir d'un certain rang. Dans [44, 43] la cas d'un facteur d'échelle scalaire est traité et les conditions sous lesquelles la convergence est démontrée implique notamment :

$$\sum_{k=0}^{+\infty} |\lambda^{k+1} - \lambda^k| < +\infty$$

Nous prouvons la convergence de l'algorithme 4.2.1 sous une hypothèse similaire (hypothèse 4.2.3).

4.2.2 Convergence globale

Soit (y^*, u^*) une solution du système (3.1). Nous introduisons la suite $\{\sigma_k\}_k, \sigma_k \in \text{fix}(\mathcal{J}_{\alpha^k}^k)$ de points fixes qui lui est associée définie pour tout k par :

$$\sigma^k = M_k y^* + M_k^{-\top} u^*$$

Comme nous l'avons déjà mentionné, l'itération k rapproche de l'ensemble des points fixes de $\mathcal{J}_{\alpha^k}^k$. En effet, par définition, nous avons pour tout k :

$$\begin{aligned} \bar{s}^{k+1} &= \mathcal{J}_{\alpha^k}^k(s^k) \\ \sigma^k &= \mathcal{J}_{\alpha^k}^k(\sigma^k) \end{aligned}$$

et d'après la définition 4.1, il vient par le lemme B.1.1 l'inégalité :

$$\|\bar{s}^{k+1} - \sigma^k\|^2 \leq \|s^k - \sigma^k\|^2 - \alpha^k(1 - \alpha^k) \|\mathcal{N}_2^k \circ \mathcal{N}_1^k(s^k) - s^k\|^2 \quad (4.2)$$

Il reste maintenant à quantifier l'effet du changement de métrique pour passer de \bar{s}^{k+1} à s^{k+1} . Nous effectuons l'hypothèse suivante :

Hypothèse 4.2.1.

- a) M_k est inversible pour tout k
- b) $M_k \rightarrow M_\infty$ inversible

qui s'exprime de manière équivalente sur la suite de matrice $\Lambda_k = M_k^\top M_k$ par :

Hypothèse 4.2.2.

- a) Λ_k est symétrique définie-positif pour tout k
- b) $\Lambda_k \rightarrow \Lambda_\infty$ définie positive

Lemme 4.2.1. Soit (y^*, u^*) une solution de (3.1) et $\sigma^k = M_k y^* + M_k^{-\top} u^*$. Sous l'hypothèse 4.2.1, il existe une constante $c > 0$ et une suite de réels positifs $\{\mu_k\}$ telle que l'erreur commise lors du changement de facteur d'échelle est donnée par :

$$\|s^{k+1} - \sigma^{k+1}\|^2 \leq (1 + \mu_k) \|\bar{s}^{k+1} - \sigma^k\|^2 \quad (4.3)$$

avec :

$$\mu_k \leq c \|\Lambda_{k+1} - \Lambda_k\|$$

Démonstration. Nous développons d'une part :

$$\begin{aligned} \|s^{k+1} - \sigma^{k+1}\|^2 &= \|M_{k+1}(y^{k+1} - y^*) + M_{k+1}^{-\top}(u^{k+1} - u^*)\|^2 \\ &= \langle y^{k+1} - y^*, \Lambda_{k+1}(y^{k+1} - y^*) \rangle + \langle u^{k+1} - u^*, \Lambda_{k+1}^{-1}(u^{k+1} - u^*) \rangle \\ &\quad + 2 \langle y^{k+1} - y^*, u^{k+1} - u^* \rangle \end{aligned}$$

et d'autre part :

$$\begin{aligned} \|\bar{s}^{k+1} - \sigma^k\|^2 &= \|M_k(y^{k+1} - y^*) + M_k^{-\top}(u^{k+1} - u^*)\|^2 \\ &= \langle y^{k+1} - y^*, \Lambda_k(y^{k+1} - y^*) \rangle + \langle u^{k+1} - u^*, \Lambda_k^{-1}(u^{k+1} - u^*) \rangle \\ &\quad + 2 \langle y^{k+1} - y^*, u^{k+1} - u^* \rangle \end{aligned}$$

La différence des deux termes donne :

$$\begin{aligned} &\|s^{k+1} - \sigma^{k+1}\|^2 - \|\bar{s}^{k+1} - \sigma^k\|^2 \\ &= \langle y^{k+1} - y^*, (\Lambda_{k+1} - \Lambda_k)(y^{k+1} - y^*) \rangle + \langle u^{k+1} - u^*, (\Lambda_{k+1}^{-1} - \Lambda_k^{-1})(u^{k+1} - u^*) \rangle \\ &\leq \|\Lambda_{k+1} - \Lambda_k\| \|y^{k+1} - y^*\|^2 + \|\Lambda_{k+1}^{-1} - \Lambda_k^{-1}\| \|u^{k+1} - u^*\|^2 \\ &\leq \|\Lambda_k^{-1}\| \|(\Lambda_{k+1} - \Lambda_k)\| \langle y^{k+1} - y^*, \Lambda_k(y^{k+1} - y^*) \rangle \\ &\quad + \|\Lambda_k\| \|\Lambda_{k+1}^{-1} - \Lambda_k^{-1}\| \langle u^{k+1} - u^*, \Lambda_k^{-1}(u^{k+1} - u^*) \rangle \\ &\leq \mu_k [\langle y^{k+1} - y^*, \Lambda_k(y^{k+1} - y^*) \rangle + \langle u^{k+1} - u^*, \Lambda_k^{-1}(u^{k+1} - u^*) \rangle] \end{aligned}$$

avec $\mu_k \stackrel{def}{=} \max(\|\Lambda_k^{-1}\| \|\Lambda_{k+1} - \Lambda_k\|, \|\Lambda_k\| \|\Lambda_{k+1}^{-1} - \Lambda_k^{-1}\|)$. Par construction $u^{k+1} \in T_1(y^{k+1})$ et puisque la solution satisfait aussi $u^* \in T_1(y^*)$, la monotonie de T_1 donne :

$$\langle y^{k+1} - y^*, u^{k+1} - u^* \rangle \geq 0$$

d'où :

$$\|s^{k+1} - \sigma^{k+1}\|^2 - \|\bar{s}^{k+1} - \sigma^k\|^2 \leq \mu_k \|\bar{s}^{k+1} - \sigma^k\|^2$$

Nous avons par ailleurs :

$$\begin{aligned} \|\Lambda_{k+1}^{-1} - \Lambda_k^{-1}\| &= \|\Lambda_k^{-1}(\Lambda_k \Lambda_{k+1}^{-1} - I)\| \\ &= \|\Lambda_k^{-1}(\Lambda_k - \Lambda_{k+1})\Lambda_{k+1}^{-1}\| \\ &\leq \|\Lambda_k^{-1}\| \|\Lambda_{k+1}^{-1}\| \|\Lambda_k - \Lambda_{k+1}\| \end{aligned}$$

En nous rappelant que la suite $\{\Lambda_k\}_k$ converge vers une limite définie positive et que par conséquent $\{\|\Lambda_k\|\}_k$ et $\{\|\Lambda_k^{-1}\|\}_k$ sont bornées, nous concluons en posant :

$$c = \sup_k (\|\Lambda_k^{-1}\|, \|\Lambda_k\| \|\Lambda_k^{-1}\| \|\Lambda_{k+1}^{-1}\|) < +\infty$$

□

Afin de prouver la convergence de l'algorithme 4.2.1, nous allons profiter de la convergence graphique des opérateurs mis en jeu, que nous nous attachons à démontrer maintenant.

Lemme 4.2.2. *Sous l'hypothèse 4.2.1 les suites $\{\mathcal{T}_1^k\}_k$ et $\{\mathcal{T}_2^k\}_k$ convergent graphiquement vers \mathcal{T}_1 et \mathcal{T}_2 définis par :*

$$\begin{aligned} \mathcal{T}_1 &= M^{-\top} T_1(M^{-1}\cdot) \\ \mathcal{T}_2 &= M^{-\top} T_2(M^{-1}\cdot) \end{aligned}$$

Démonstration. Ecrivons la preuve pour \mathcal{T}_1 . Soit $w \in \mathcal{T}_1(z)$ et considérons la suite définie pour tout k par :

$$(z_k, w_k) = (M_k M^{-1} z, M_k^{-\top} M^{\top} w) \in gr(\mathcal{T}_1^k)$$

Elle converge vers (z, w) par continuité de l'inversion d'où :

$$gr(\mathcal{T}_1) \subset \liminf \{gr(\mathcal{T}_1^k)\}_k \quad (4.4)$$

Soit maintenant $(z, w) \in \limsup \{gr(\mathcal{T}_1^k)\}_k$. Il existe par définition une sous-suite $\{(z_{\nu(k)}, w_{\nu(k)})\}_k$ convergente vers (z, w) et telle que $(z_{\nu(k)}, w_{\nu(k)}) \in gr(\mathcal{T}_1^{\nu(k)})$, c'est à dire :

$$\begin{aligned} w_{\nu(k)} &\in M_{\nu(k)}^{-\top} T_1(M_{\nu(k)}^{-1} z_{\nu(k)}) \\ M_{\nu(k)}^{\top} w_{\nu(k)} &\in T_1(M_{\nu(k)}^{-1} z_{\nu(k)}) \end{aligned}$$

La suite $\{(M_{\nu(k)}^{-1} z_{\nu(k)}, M_{\nu(k)}^{\top} w_{\nu(k)})\}_k$ est donc incluse dans le graphe de T_1 qui est fermé puisque T_1 est monotone maximal. La limite $(M^{-1} z, M^{\top} w)$ est donc aussi dans $gr(T_1)$:

$$\begin{aligned} M^{\top} w &\in T_1(M^{-1} z) \\ w &\in \mathcal{T}_1(z) \end{aligned}$$

ce qui permet d'affirmer que :

$$\limsup \{gr(\mathcal{T}_1^k)\}_k \subset gr(\mathcal{T}_1)$$

Combinée à (4.4), cette inclusion donne la convergence graphique de \mathcal{T}_1^k vers \mathcal{T}_1 . □

Nous prouvons maintenant la convergence de l'algorithme 4.2.1 sous des hypothèses similaires à celles de [43], c'est à dire :

Hypothèse 4.2.3. $\sum_{k=0}^{+\infty} \|\Lambda_{k+1} - \Lambda_k\| < +\infty$

Lemme 4.2.3. *Sous les hypothèses 4.2.2 et 4.2.3, la suite $\{s^k\}_k$ générée par l'algorithme 4.2.1 est bornée.*

Démonstration. En combinant (4.2) et (4.3), nous obtenons pour tout k :

$$\|s^{k+1} - \sigma^{k+1}\|^2 \leq (1 + \mu_k) \|s^k - \sigma^k\|^2 \quad (4.5)$$

L'hypothèse 4.2.3 donne $\sum_{k=0}^{+\infty} \mu_k < +\infty$ ou de manière équivalente :

$$\prod_{k=0}^{+\infty} (1 + \mu_k) \leq \Gamma < +\infty \quad (4.6)$$

pour un certain $\Gamma \geq 0$. L'utilisation récursive de (4.5) pour k allant de 0 à $l-1$ fournit :

$$\begin{aligned} \|s^l - \sigma^l\|^2 &\leq \left(\prod_{k=0}^{l-1} (1 + \mu_k) \right) \|s^0 - \sigma^0\|^2 \\ \|s^l - \sigma^l\|^2 &\leq \Gamma \|s^0 - \sigma^0\|^2 \end{aligned}$$

Par conséquent, la suite $\left\{ \|s^k - \sigma^k\|^2 \right\}_k$ est bornée. Par ailleurs, puisque M_k converge, il en est de même pour $\{\sigma^k\}_k$ ce qui permet de conclure. \square

Proposition 4.2.1. *Soit \mathcal{N}_1 et \mathcal{N}_2 les réflexions généralisées associés aux opérateurs limites \mathcal{T}_1 et \mathcal{T}_2 . Sous les hypothèses B.3.1, 4.2.2 et 4.2.3, la suite $\{s^k\}_k$ générée par l'algorithme 4.2.1 converge vers un point fixe de $\mathcal{N}_2 \circ \mathcal{N}_1$.*

Démonstration. Notons d'abord que le lemme 4.2.3 fournit l'existence de points d'accumulation de la suite $\{s^k\}_k$. La combinaison de (4.2) et (4.3) donne pour tout k :

$$\|s^{k+1} - \sigma^{k+1}\|^2 \leq (1 + \mu_k) \left[\|s^k - \sigma^k\|^2 - \alpha^k (1 - \alpha^k) \|t^{k+1} - s^k\|^2 \right]$$

avec $t^{k+1} \stackrel{\text{def}}{=} \mathcal{N}_2^k \circ \mathcal{N}_1^k(s^k)$. En sommant cette relation de 0 jusqu'à $l-1$, nous obtenons :

$$\begin{aligned} &\sum_{k=0}^{l-1} (1 + \mu_k) \alpha^k (1 - \alpha^k) \|t^{k+1} - s^k\|^2 \\ &\leq \|s^l - \sigma^l\|^2 - \|s^0 - \sigma^0\|^2 + \sum_{k=0}^{l-1} \mu_k \|s^{k+1} - \sigma^{k+1}\|^2 \end{aligned}$$

Chaque terme du membre de droite est borné, la série du membre de gauche est donc convergente ce qui donne $\liminf \|t^{k+1} - s^k\| = 0$ grâce à l'hypothèse B.3.1. En utilisant le fait que $\mathcal{N}_2^k \circ \mathcal{N}_1^k$ est faiblement contractant pour tout k , la suite $\{(s^k, t^{k+1})\}_k$ est bornée d'après le lemme 4.2.3 et possède donc un point d'accumulation de la forme (\bar{s}, \bar{s}) .

D'autre part $\mathcal{N}_2^k \circ \mathcal{N}_1^k$ converge graphiquement vers $\mathcal{N}_2 \circ \mathcal{N}_1$ d'après les lemmes 4.2.2 et 2.3.3. Nous en concluons que $(\bar{s}, \bar{s}) \in \text{gr}(\mathcal{N}_2 \circ \mathcal{N}_1)$ et que tous les points d'accumulation de $\{s^k\}_k$ sont des points fixes de $\mathcal{N}_2 \circ \mathcal{N}_1$. Montrons maintenant que ce point d'accumulation est unique. Soit (\bar{y}, \bar{u}) la solution associée au point d'accumulation s . La suite $\sigma^k = M_k y + M_k^{-\top} u$ converge vers s . Soit $\epsilon > 0$ et K un rang tel que :

a) $\{\sigma^k\}_{k \geq K} \subset B(\bar{s}, \epsilon)$

b) $s_K \in B(\bar{s}, \epsilon)$

Nous avons d'après a) et b) :

$$\|s^K - \sigma^K\| \leq 2\epsilon$$

L'utilisation récursive de (4.5) pour k allant de K jusqu'à $l-1$ donne :

$$\begin{aligned} \|s^l - \sigma^l\|^2 &\leq \left[\prod_{k=K}^{l-1} (1 + \mu_k) \right] \|s^K - \sigma^K\|^2 \\ &\leq \left[\prod_{k=0}^{K-1} (1 + \mu_k) \right] \left[\prod_{k=K}^{l-1} (1 + \mu_k) \right] \|s^K - \sigma^K\|^2 \\ &\leq 4\Gamma\epsilon^2 \end{aligned}$$

et :

$$\begin{aligned} \|s^l - \bar{s}\| &\leq \|s^l - \sigma^l\| + \|\sigma^l - \bar{s}\| \\ &\leq \sqrt{4\Gamma\epsilon^2} + \epsilon \\ &\leq (1 + 2\sqrt{\Gamma})\epsilon \end{aligned}$$

Ceci est vrai pour tout $\epsilon > 0$ donc $s^k \rightarrow s$. \square

Nous étudions dans la section suivante le comportement asymptotique des algorithmes 3.4.3 et 3.4.4 en utilisant les propriétés locales des opérateurs mis en jeu. Les résultats obtenus sont ensuite étendus à généralisés au cas d'un facteur d'échelle variable (algorithme 4.2.1).

4.2.3 Vitesse de convergence

Nous dérivons maintenant un résultat analogue à (3.18) lorsque le facteur d'échelle varie en suivant la même démarche. Si T_1 et T_2 sont proto-différentiables en $(\bar{y}|\bar{u})$ et $(\bar{y}|\bar{u})$. Alors, pour tout k , \mathcal{T}_1^k et \mathcal{T}_2^k sont proto-différentiables en $(\bar{z}|\bar{w}) = (M_k\bar{y}|M_k^{-\top}\bar{u})$ et $(\bar{z}|\bar{w})$ et puis \mathcal{N}_1^k et \mathcal{N}_2^k sont semi-différentiables en $\bar{s} = \bar{z} + \bar{w}$ et $\bar{d} = \bar{z} - \bar{w}$. L'opérateur $\mathcal{J}_{\alpha^k}^k$ l'est finalement aussi en \bar{s} avec :

$$D\mathcal{J}_{\alpha^k}^k(\bar{s}) = (1 - \alpha^k)I + \alpha^k D\mathcal{N}_2^k(\bar{d}) \circ D\mathcal{N}_1^k(\bar{s}) \quad (4.7)$$

Le développement de $\mathcal{J}_{\alpha^k}^k$ en son point fixe $\sigma^k = M_k\bar{y} + M_k^{-\top}\bar{u}$ et évalué en s^k s'écrit :

$$\begin{aligned} \mathcal{J}_{\alpha^k}^k(s^k) &= \mathcal{J}_{\alpha^k}^k(\sigma^k) + D\mathcal{J}_{\alpha^k}^k(\sigma^k)(s^k - \sigma^k) + o(\|s^k - \sigma^k\|) \\ \bar{s}^{k+1} &= \sigma^k + D\mathcal{J}_{\alpha^k}^k(\sigma^k)(s^k - \sigma^k) + o(\|s^k - \sigma^k\|) \end{aligned}$$

En posant $e^k = s^{k+1} - \sigma^{k+1}$, nous obtenons :

$$\begin{aligned} e^{k+1} &= D\mathcal{J}_{\alpha^k}^k(\sigma^k)(e^k) + \sigma^k - \sigma^{k+1} + s^{k+1} - \bar{s}^{k+1} + o(\|e^k\|) \\ e^{k+1} &= D\mathcal{J}_{\alpha^k}^k(\sigma^k)(e^k) + (M_{k+1} - M_k)(y^{k+1} - \bar{y}) \\ &\quad + (M_{k+1} - M_k)^\top(u^{k+1} - \bar{u}) + o(\|e^k\|) \\ \|e^{k+1}\| &\leq \|D\mathcal{J}_{\alpha^k}^k(\sigma^k)\| \|e^k\| + \|M_{k+1}M_k^{-1} - I\| \|M_k(y^{k+1} - \bar{y})\| \\ &\quad + \|M_{k+1}^\top M_k^{-\top} - I\| \|M_k^\top(u^{k+1} - \bar{u})\| + o(\|e^k\|) \end{aligned}$$

Par monotonie de \mathcal{T}_1^k , nous avons $\|M_k(y^{k+1} - \bar{y})\| \leq \|e^{k+1}\|$, et $\|M_k^\top(u^{k+1} - \bar{u})\| \leq \|e^{k+1}\|$. La suite définie pour tout k par $\mu_k \stackrel{\text{def}}{=} \|M_{k+1}M_k^{-1} - I\| + \|M_{k+1}^\top M_k^{-\top} - I\|$ tend vers 0 puisque M_k converge et nous avons :

$$\begin{aligned} \|e^{k+1}\| &\leq (\|D\mathcal{J}_{\alpha^k}^k(\sigma^k)\| + \mu_k) \|e^k\| + o(\|e^k\|) \\ \frac{\|e^{k+1}\|}{\|e^k\|} &\leq \|D\mathcal{J}_{\alpha^k}^k(\sigma^k)\| + \mu_k + \frac{o(\|e^k\|)}{\|e^k\|} \end{aligned}$$

et nous obtenons ainsi par passage à la limite supérieure la borne suivante sur le taux de convergence :

$$\limsup_k \frac{\|e^{k+1}\|}{\|e^k\|} \leq \limsup_k \|D\mathcal{J}_{\alpha^k}^k(\sigma^k)\| \quad (4.8)$$

Il est difficile dans le cas général d'obtenir des informations sur $D\mathcal{J}_{\alpha^k}^k$. Plus de choses peuvent être dites lorsque \mathcal{N}_1^k et/ou \mathcal{N}_2^k sont différentiables, c'est à dire lorsque $D\mathcal{N}_1^k$ et $D\mathcal{N}_2^k$ sont les opérateurs linéaires $\nabla\mathcal{N}_1^k$ et $\nabla\mathcal{N}_2^k$. $D\mathcal{J}_{\alpha^k}^k = \nabla\mathcal{J}_{\alpha^k}^k$ est alors lui aussi linéaire et la formule (4.7) se spécialise en :

$$D\mathcal{J}_{\alpha^k}^k(\bar{s}) = (1 - \alpha^k)I + \alpha^k \nabla\mathcal{N}_2^k(\bar{d}) \nabla\mathcal{N}_1^k(\bar{s})$$

Nous distinguerons dans les sections 4.3 et 4.4 deux cas dans lesquels il est possible de déterminer l'influence des paramètres de l'algorithme sur la vitesse de convergence. Il s'agit d'une part du cas où l'un des deux opérateurs T_1 ou T_2 est différentiable et d'autre part du cas où T_1 et T_2 sont les sous-différentiels de fonctions polyédrales.

4.3 ∂F ou ∂C sont différentiables

4.3.1 Convergence super-linéaire

Nous étudions l'algorithme 4.2.1. en supposant qu'il converge vers (\bar{y}, \bar{u}) , $\bar{s} = M\bar{y} + M^{-\top}\bar{u}$. Nous étudions le cas où ∂F est différentiable en \bar{y} avec $\nabla[\partial F]$ inversible. En reprenant les notations de la section 2.3.3, cela correspond à $\mathcal{H}_3 = \{0\}$ pour la différentiabilité et $\mathcal{H}_2 = \{0\}$ pour l'inversibilité, ainsi nous avons tout simplement :

$$\nabla[\partial F](\bar{y}) = S$$

avec S une matrice symétrique définie positive. A l'itération k , l'opérateur $\mathcal{T}^k = M_k^\top \partial F M_k$ est différentiable et

$$\begin{aligned} \nabla\mathcal{T}_1^k(M_k^{-1}\bar{y}) &= M_k^\top S M_k \\ \nabla\mathcal{N}_1^k(\sigma_k) &= 2(I_n + M_k^\top S M_k)^{-1} - I_n \end{aligned}$$

Nous avons :

$$\begin{aligned} \|D\mathcal{J}_{\alpha^k}^k(\sigma^k)\| &= \|(1 - \alpha^k)I + \alpha^k D\mathcal{N}_2^k(\bar{d}) \nabla\mathcal{N}_1^k(\bar{s})\| \\ &= \|(1 - \alpha^k)I + \alpha^k D\mathcal{N}_2^k(\bar{d}) [2(I_n + M_k^\top S M_k)^{-1} - I_n](\bar{s})\| \end{aligned}$$

En particulier, si $M_k \rightarrow S^{\frac{1}{2}}$, la formule (4.8) devient alors :

$$\limsup_k \frac{\|e^{k+1}\|}{\|e^k\|} \leq \limsup_k (1 - \alpha^k)$$

L'hypothèse B.3.1 autorise le choix d'une suite $\{\alpha^k\}_k$ de limite égale à 1 avec $\sum_{k=0}^{+\infty} (1 - \alpha^k) = +\infty$, procurant alors une vitesse de convergence super-linéaire. Notons ici que la même démarche peut être appliquée à ∂C plutôt qu'à ∂F .

4.3.2 Mise à jour du facteur d'échelle

Nous proposons ici une famille de stratégies pour accélérer la convergence de l'algorithme. Elles sont plus particulièrement adaptées au cas différentiable mais s'appliquent dans le cas général. Lorsque ∂F est différentiable en \bar{y} , il serait souhaitable d'obtenir des informations sur $\nabla^2 F(y^*)$ au fur et à mesure que l'algorithme progresse en utilisant les itérés intermédiaires y^k et u^k qui satisfont :

$$u^k \in \partial F(y^k)$$

La différence entre deux couples d'itérés vérifie donc :

$$u^{k+1} - u^k \in \nabla^2 F(y^k)(y^{k+1} - y^k) + o(\|y^{k+1} - y^k\|)B(0, 1)$$

L'idée consiste à effectuer une mise à jour matricielle afin d'obtenir une suite approchant $\nabla^2 F(\bar{y})$ lorsque $y^k \rightarrow \bar{y}$, tout en respectant l'hypothèse 4.2.3. Pour cela, nous mettons à jour la matrice de mise à l'échelle à chaque itération en prenant une combinaison convexe (de coefficients $0 < \beta_k \leq 1$) de la matrice courante Λ_k et d'une matrice D_k que l'on espère être proche de $\nabla T(\tilde{y}^k)$.

$$\Lambda_{k+1} = (1 - \beta_k)\Lambda_k + \beta_k D_k \quad (4.9)$$

Nous imposerons pour assurer l'hypothèse 4.2.3 à la suite des β_k et des D_k de se conformer aux hypothèses du lemme suivant :

Lemme 4.3.1. *Soit $\{\Lambda_k\}_k$ une suite de matrices satisfaisant (4.9). Notons $0 < d_1^k \leq \dots \leq d_m^k$ les m valeurs propres de D_k . S'il existe $0 < \underline{d} \leq \bar{d} < +\infty$ tel que :*

$$\underline{d} \leq d_1^k \leq \dots \leq d_m^k \leq \bar{d}$$

et la série de terme $\beta_k \geq 0$ est convergente :

$$\sum_{k=0}^{+\infty} \beta_k = S < +\infty$$

alors la suite $\{\Lambda_k\}_k$ satisfait les hypothèses 4.2.2 et 4.2.3.

Démonstration. Choisissons $[\underline{d}, \bar{d}]$ contenant les valeurs propres de Λ_0 . Alors, l'ensemble \mathcal{S} des matrices symétriques dont les valeurs propres sont dans $[\underline{d}, \bar{d}]$ est convexe et compact. Il contient la suite $\{D_k\}_k$ ainsi que Λ_0 , et par conséquent toute la suite des $\{\Lambda_k\}_k$ d'après (4.9) ainsi que ses points d'accumulation. De plus, pour tout k :

$$\begin{aligned} \Lambda_{k+1} - \Lambda_k &= -\beta_k \Lambda_k + \beta_k D_k \\ \|\Lambda_{k+1} - \Lambda_k\| &\leq \beta_k (\|\Lambda_k\| + \|D_k\|) \\ &\leq 2\bar{d}\beta_k \end{aligned}$$

donc $\sum_{k=0}^{+\infty} \|\Lambda_{k+1} - \Lambda_k\| = S' \in \mathbb{R}$. □

Nous pouvons aussi définir la récurrence :

$$\Lambda_{k+1} = \Lambda_k^{(1-\beta_k)} D_k^{\beta_k} \quad (4.10)$$

au lieu de (4.9). En utilisant les mêmes arguments, nous obtenons que la suite $\{\ln(\Lambda_k)\}_k$ est incluse dans l'ensemble des matrices dont les valeurs propres sont dans $[\ln(\underline{d}), \ln(\bar{d})]$ obtenant ainsi la définie positivité stricte de $\{\Lambda_k\}_k$ et de sa limite. En utilisant enfin le fait que le logarithme est lipschitzien sur tout compact, nous obtenons aussi la convergence absolue.

La mise en œuvre pratique d'une métrique variable nécessite un compromis entre :

- choisir des matrices de structures simples (diagonales) faciles à implanter, mais qui ne fournissent qu'une approximation frustrée de la matrice à approcher.
- choisir des matrices permettant d'approcher finement les informations de second ordre, mais dont le calcul peut ralentir la progression globale de l'algorithme.

Nous avons implémenté en pratique les techniques suivantes.

Un paramètre unique En nous restreignant à un paramètre réel *i.e.* $\Lambda_k = \lambda^k I$, une façon d'utiliser des informations du second ordre est de tirer directement profit du ratio :

$$\gamma^k = \frac{\|\tilde{u}^{k+1} - \tilde{u}^k\|}{\|\tilde{y}^{k+1} - \tilde{y}^k\|}$$

Dans le cas général, nous n'avons aucune idée sur le caractère borné de $\{\gamma_k\}_k$. Nous considérerons donc un intervalle $[\underline{\gamma}, \bar{\gamma}]$ et mettrons en œuvre :

$$D_k = (\Pi_{[\underline{\gamma}, \bar{\gamma}]} \gamma^k) I \quad (4.11)$$

où $\Pi_{[\underline{\gamma}, \bar{\gamma}]}(\cdot) = \max(\underline{\gamma}, \min(\cdot, \bar{\gamma}))$ représente la projection sur l'intervalle $[\underline{\gamma}, \bar{\gamma}]$.

Mise à jour sous-problème par sous-problème Lorsque T_1 a une structure bloc-angulaire :

$$T_1(y) = \begin{pmatrix} T_{1,1}(y_1) \\ \vdots \\ T_{1,p}(y_p) \end{pmatrix} \quad (4.12)$$

et que son calcul se décompose (voir étape 7 de l'algorithme), il est avantageux d'appliquer la même politique, mais séparément dans chacun des sous problèmes *i.e.* pour $i = 1, \dots, p$, nous posons :

$$\gamma_i^{k+1} = \frac{\|\tilde{u}_i^{k+1} - \tilde{u}_i^k\|}{\|\tilde{y}_i^{k+1} - \tilde{y}_i^k\|} \quad (4.13)$$

et D_k est la matrice diagonale :

$$D_k = \begin{pmatrix} (\Pi_{[\underline{\gamma}, \bar{\gamma}]} \gamma_1^k) I_m & & \\ & \ddots & \\ & & (\Pi_{[\underline{\gamma}, \bar{\gamma}]} \gamma_p^k) I_m \end{pmatrix}$$

| | | Règle de mise à jour | | | |
|----|----|----------------------|--------------------|----------------|-------------------|
| p | m | aucune | paramètre scalaire | sous-problèmes | matrice diagonale |
| 2 | 5 | 39 | 41 | 48 | 37 |
| | 10 | 78 | 74 | 74 | 89 |
| | 20 | 93 | 81 | 81 | 86 |
| 5 | 5 | 61 | 81 | 53 | 59 |
| | 10 | 77 | 83 | 80 | 88 |
| | 20 | 104 | 101 | 99 | 110 |
| 10 | 5 | 85 | 102 | 104 | 127 |
| | 10 | 126 | 141 | 135 | 152 |
| | 20 | 126 | 161 | 149 | 178 |
| 20 | 5 | 150 | 178 | 143 | 160 |
| | 10 | 187 | 256 | 191 | 200 |
| | 20 | 181 | 268 | 236 | 244 |

TABLE 4.1 – Nombre d’itérations minimal pour $\lambda^0 \in [10^{-3}, 100]$ pour différentes stratégies de mise à jour.

obtenu. Nous pouvons voir que le meilleur des cas est comparable que l’on utilise ou non une mise à jour. Cependant, le besoin de choisir une bonne valeur initiale disparaît. Le tableau 4.2 contient l’écart type du nombre d’itérations en fonction de λ^0 . Le symbole ‘ \geq ’ dans la première colonne signifie que la valeur est sous-estimée car plusieurs lancements ont dépassé le seuil des 5000 iterations. Les algorithmes implémentés avec une stratégie de mise à jour sont moins sensibles à la valeur initiale que la méthode classique. Comme illustration de ce fait, nous avons tracé pour chaque stratégie le nombre d’itérations en fonction de $\lambda^{k=0}$ dans le cas $p = 20$ et $m = 10$ sur la figure 4.3.1.

Il est à souligner que la mise à jour de type (4.9) a un comportement asymétrique selon que l’ordre de grandeur du paramètre de départ. Plus précisément, les petites valeurs seront bien mieux corrigées que les valeurs élevées à cause de la difficulté à faire décroître l’ordre de grandeur avec l’expression (4.9). La mise à jour la plus adaptée semble donc être (4.10). Nous traçons à titre d’exemple pour l’instance ($p = 2, m = 20$) les trajectoires suivies par la suite λ^k dans le cas d’un seul paramètre (figure 4.3.2) et par les suites $(\lambda_1^k, \lambda_2^k)$ pour le cas d’un paramètre par sous-problème (figure 4.3.3).

4.4 F et C sont polyédrales

Nous étudions dans cette section le cas particulier où $T_1 = \partial F$ et $T_2 = \partial C$ avec F et C deux fonctions polyédrales. Nous nous restreignons au cas où le facteur d’échelle est fixe et nous supposons toujours que l’algorithme 3.4.2 converge vers \bar{s} avec $\bar{y} = P_1(\bar{s})$ et $\bar{u} = Q_1(\bar{s})$, $\bar{d} = \bar{y} - \bar{u}$.

4.4.1 Comportement local

Supposons que T_1 et T_2 soient proto-différentiables en \bar{y} relativement à \bar{u} et à $-\bar{u}$ respectivement et de proto-dérivées linéaires au sens général. En reprenant les notations de la section 3.4.4, nous avons $\mathcal{H}_1 = \{0\}$, la proto-dérivée $DT_1(\bar{y}|\bar{u})$ se résume à $\mathbf{N}_{\mathcal{H}}$ et d’après la proposition 2.3.6, N_1 et N_2 sont différentiables respectivement en \bar{s} et en \bar{d} . De

| | | Règle de mise à jour | | | |
|----|----|----------------------|--------------------|----------------|-------------------|
| p | m | aucune | paramètre scalaire | sous-problèmes | matrice diagonale |
| 2 | 5 | ≥ 415 | 17 | 9 | 21 |
| | 10 | ≥ 2076 | 56 | 62 | 93 |
| | 20 | ≥ 1776 | 60 | 56 | 58 |
| 5 | 5 | ≥ 2038 | 38 | 39 | 54 |
| | 10 | ≥ 2129 | 37 | 31 | 58 |
| | 20 | ≥ 2158 | 59 | 51 | 55 |
| 10 | 5 | ≥ 2104 | 72 | 39 | 54 |
| | 10 | ≥ 2112 | 79 | 55 | 112 |
| | 20 | ≥ 2129 | 180 | 123 | 354 |
| 20 | 5 | ≥ 2108 | 119 | 67 | 430 |
| | 10 | ≥ 2081 | 251 | 133 | 320 |
| | 20 | ≥ 2092 | 220 | 131 | 321 |

TABLE 4.2 – Écart-type du nombre d'itérations en fonction de λ^0 pour différentes stratégies de mise à jour.

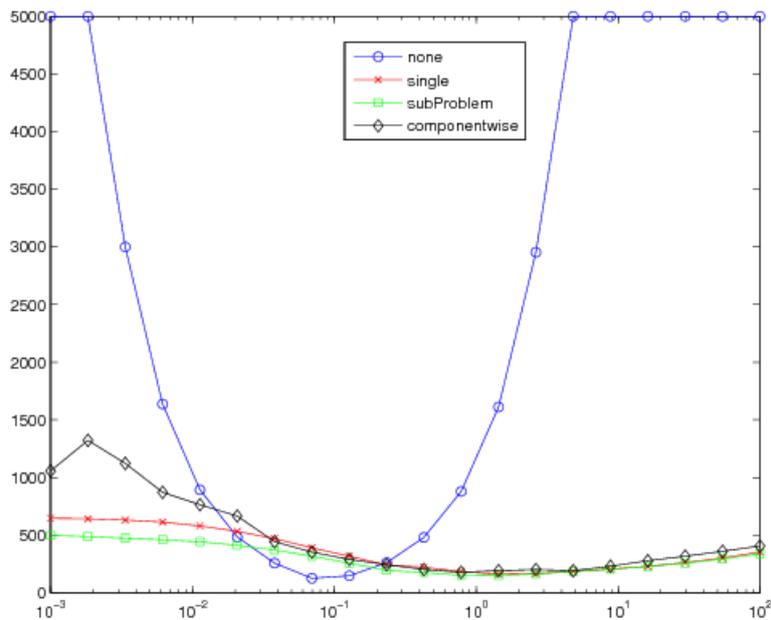


FIGURE 4.3.1 – Nombre d'itérations de SALA pour différentes stratégies de mise à jour.

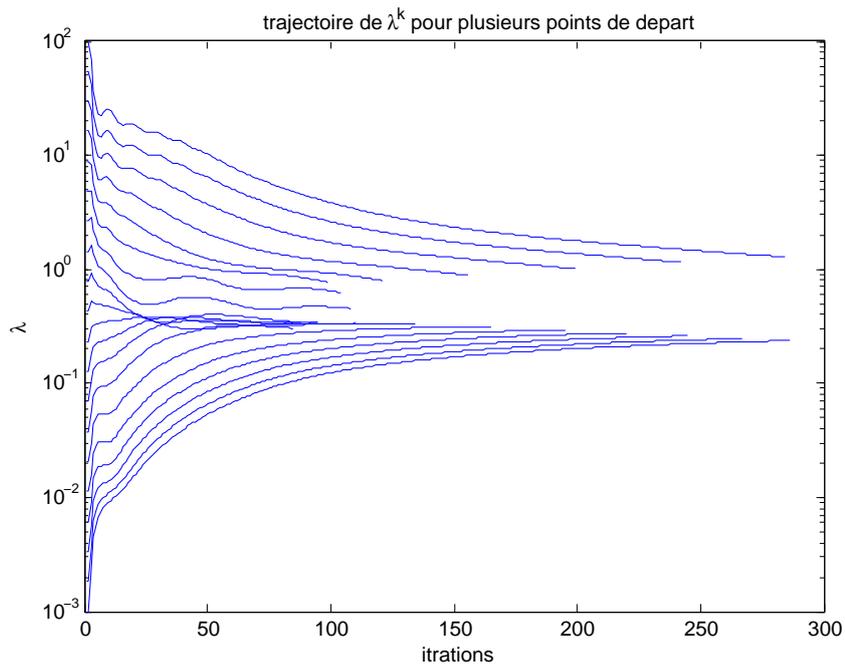
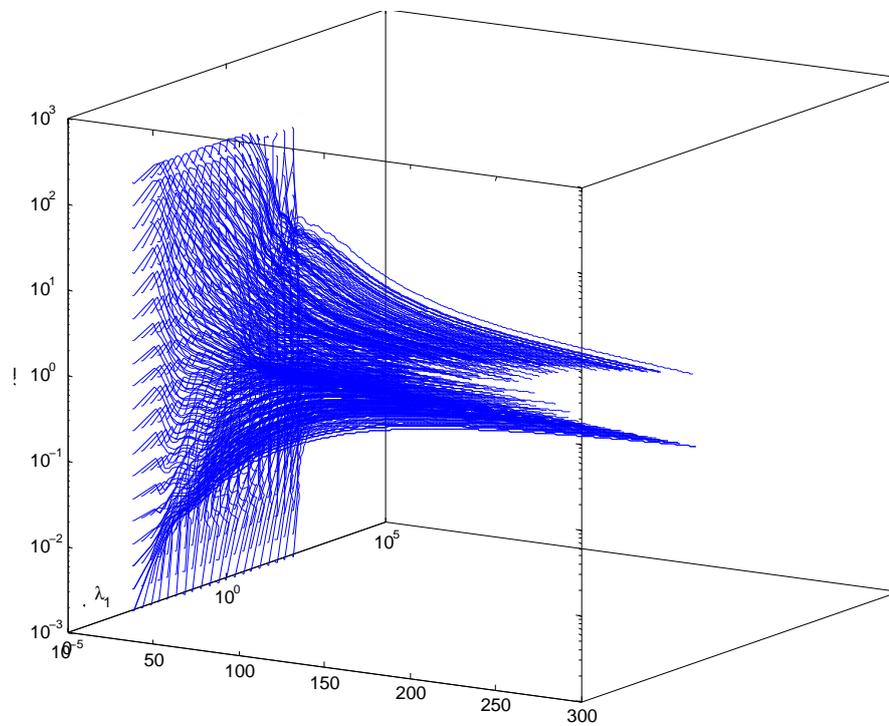


FIGURE 4.3.2 – Trajectoires du paramètre de mise à l'échelle.

FIGURE 4.3.3 – Trajectoires des paramètres de mise à l'échelle associés à chaque sous-problème pour différentes valeurs de départ $(\lambda_1^0, \lambda_2^0)$.

Lemme 4.4.1. *La rotation moyennée $(1 - \alpha)I_2 + \alpha E_i$ a pour norme :*

$$\|(1 - \alpha)I_2 + \alpha E_i\|_2 = \left[(1 - 2\alpha)^2 \sin^2 \left(\frac{\theta_i}{2} \right) + \cos^2 \left(\frac{\theta_i}{2} \right) \right]^{\frac{1}{2}}$$

Démonstration. Choisissons $v \in \mathbb{R}^2$ et calculons la norme de son image $v' = (1 - \alpha)v + \alpha E_i v$. En utilisant le fait que $\|E_i v\| = \|v\|$ et $\langle v, E_i v \rangle = \cos(\theta_i)$, il vient :

$$\begin{aligned} \|v'\|^2 &= \|(1 - \alpha)v + \alpha E_i v\|^2 \\ &= \left\| (1 - 2\alpha) \frac{1}{2} (v - E_i v) + \frac{1}{2} (v + E_i v) \right\|^2 \\ &= (1 - 2\alpha)^2 \frac{1}{4} \|v - E_i v\|^2 + \frac{1}{4} \|v + E_i v\|^2 + \frac{1}{4} (1 - 2\alpha) \underbrace{\left(\|v\|^2 - \|E_i v\|^2 \right)}_{=0} \\ &= (1 - 2\alpha)^2 \frac{1}{4} (\|v\|^2 + \|E_i v\|^2 - 2 \langle v, E_i v \rangle) + \frac{1}{4} (\|v\|^2 + \|E_i v\|^2 + 2 \langle v, E_i v \rangle) \\ &= (1 - 2\alpha)^2 \frac{1 - \cos(\theta_i)}{2} \|v\|^2 + \frac{1 + \cos(\theta_i)}{2} \|v\|^2 \\ &= (1 - 2\alpha)^2 \sin^2 \left(\frac{\theta_i}{2} \right) \|v\|^2 + \cos^2 \left(\frac{\theta_i}{2} \right) \|v\|^2 \\ \|v'\| &= \left[(1 - 2\alpha)^2 \sin^2 \left(\frac{\theta_i}{2} \right) + \cos^2 \left(\frac{\theta_i}{2} \right) \right]^{\frac{1}{2}} \|v\| \end{aligned}$$

□

Nous déduisons des observations que nous venons de faire la borne suivante sur le taux de convergence :

$$\limsup \frac{\|e^{k+1}\|}{\|e^k\|} \leq \max_{i=1}^r \left[(1 - 2\alpha)^2 \sin^2 \left(\frac{\theta_i}{2} \right) + \cos^2 \left(\frac{\theta_i}{2} \right) \right]^{\frac{1}{2}}$$

qui est minimisée pour le choix $\alpha = \frac{1}{2}$ et se réduit alors à :

$$\limsup \frac{\|e^{k+1}\|}{\|e^k\|} \leq \max_{i=1}^r \left[\cos \left(\frac{\theta_i}{2} \right) \right] = \cos \left(\frac{\theta_1}{2} \right)$$

Pour un facteur d'échelle fixe, la méthode de Douglas-Rachford classique $\alpha_k = \frac{1}{2}$ est donc supérieure en terme de taux de convergence asymptotique que n'importe quel autre choix de suite α_k .

4.4.2 Lien avec les opérateurs de pliage

Cet algorithme a déjà été étudié dans le cas où F et C sont polyédrales et de plus avec ∂F et ∂C ayant des proto-dérivées non nécessairement linéaires au sens généralisé. N_1 et N_2 , ainsi que leur composition $N_2 \circ N_1$ appartiennent à une classe spéciale d'opérateurs appelés *pliage* introduite dans [55]. Un pliage est un opérateur faiblement contractant isométrique par morceau $\mathcal{F} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ avec un nombre de morceaux fini localement. Ces morceaux appelés *replis* sont des polyèdres convexes fermés d'intérieurs disjoints deux à deux et non vides. En fait, lorsque F (ou de façon équivalente C) est polyédrale, les

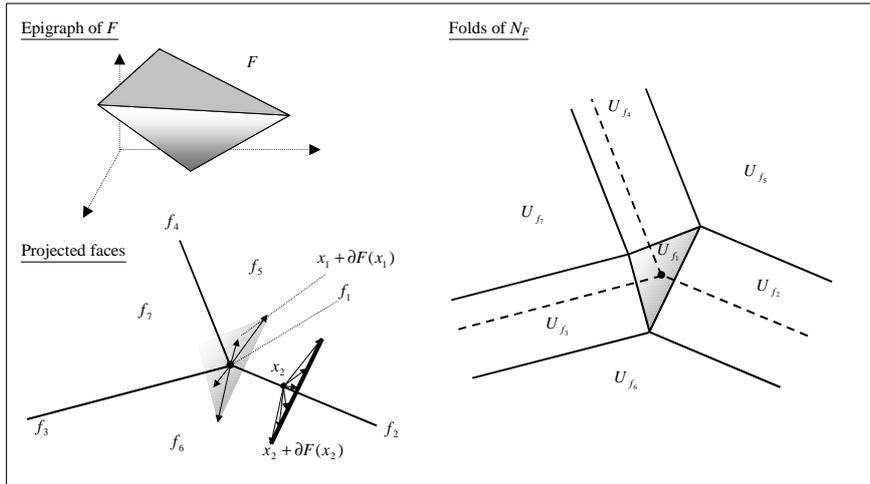


FIGURE 4.4.1 – Replis U_{f_i} associés à la projection f_i d'une face d'une fonction polyédrale convexe de \mathbb{R}^2 dans \mathbb{R} .

replis de N_1 (respectivement de N_2) sont en bijection avec les projections sur l'espace $\mathbb{R}^n \times \{0\}$ des faces de $\text{epi}(F)$, l'épigraph de F . Plus précisément, si f est la projection d'une telle face, alors le polyèdre $U_f = f + \partial F(f)$ est un repli (voir la figure 4.4.1 pour un exemple d'une telle construction). Si l'on note d_f la projection de $\partial F(f)$ sur $\text{lin}(f)$ le sous-espace vectoriel engendré par f (c'est un singleton puisque $\partial F(f)$ est orthogonal à $\text{lin}(f)$), alors la restriction de N_1 à U_f est $R_{\text{aff}(f)} - 2d_f$, *i.e.* la composition de la réflexion à travers l'espace affine engendré par f et de la translation de $-2d_f$.

Dans [55], Lawrence et Spingarn ont étudié le comportement asymptotique d'applications successives de pliages moyennés dans le cas $\alpha = \frac{1}{2}$:

$$s^{k+1} = \frac{1}{2}s^k + \frac{1}{2}\mathcal{F}(s^k)$$

en se restreignant à la classe des pliages positivement homogènes, en centrant \mathcal{F} autour du point de convergence et en considérant uniquement un voisinage de 0. Cette manipulation consiste en fait à considérer la semi-différentielle $D\mathcal{F}(\bar{s})$ de \mathcal{F} en \bar{s} et à étudier la suite des vecteurs d'erreur $\{e^k\}_k$ donné par la récurrence :

$$e^{k+1} = \frac{1}{2}e^k + \frac{1}{2}D\mathcal{F}(\bar{s})(e^k)$$

Leur résultat principal librement adapté dans le théorème 4.4.1 établit qu'à partir d'un certain rang, la suite $\{e^k\}_k$ stagne dans un certain sous-espace $V(e^0)$ (qui dépend de l'erreur initiale e^0) et que la restriction de $D\mathcal{F}(\bar{s})$ à $V(e^0)$ est une isométrie. De manière plus précise, les conséquences *i)* et *ii)* affirment que la suite $\{e^k\}_k$ converge en spirale dans $V(e^0)$.

Théorème 4.4.1 (Lawrence-Spingarn [55]). *Soit \mathcal{F} un pliage et $e^{k+1} = \frac{1}{2}(I + D\mathcal{F})e^k$ convergeant vers 0. Alors :*

i) Il existe un sous-espace $V(e^0) \subset \mathbb{R}^n$ et un rang K tel que :

$$k \geq K \Rightarrow \text{vect}(e^k, e^{k+1}, \dots) = V(e^0) = \text{cone}(e^k, e^{k+1}, \dots)$$

ii) La restriction de $D\mathcal{F}$ à $V(e^0)$ est une isométrie.

$\text{vect}(e^k, e^{k+1}, \dots)$ et $\text{cone}(e^k, e^{k+1}, \dots)$ représentent l'espace vectoriel et le cône engendrés par la suite e^k, e^{k+1}, \dots . Ce résultat est plus général que notre développement puisqu'il ne suppose pas la différentiabilité de \mathcal{F} mais simplement sa semi-différentiabilité, propriété vraie partout pour des pliages associés à des fonctions polyédrales. Il est important de noter que même dans le cas différentiable, la conclusion i) ne s'étend pas en général à des choix $\alpha < \frac{1}{2}$ à cause du second bloc dans (4.16) qui préserve la composante associée dans une demi-droite. Remarquons aussi que la conséquence ii) certifie que la restriction de $\frac{1}{2}(I + D\mathcal{F}(\bar{s}))$ à $V(e^0)$ a la même structure que $J_{\alpha=\frac{1}{2}}$ dans (4.16). Par conséquent, le taux de convergence satisfait aussi pour $0 < \theta_1 \leq \theta_2 \leq \dots \leq \theta_{r'} < \pi$:

$$\limsup \frac{\|e^{k+1}\|}{\|e^k\|} \leq \max_{i=1}^{r'} \cos\left(\frac{\theta_i}{2}\right) = \cos\left(\frac{\theta_1}{2}\right)$$

Cette interprétation géométrique de convergence en spirale sera utilisée dans la section suivante afin de proposer une possible accélération de la méthode.

4.4.3 Accélération de la convergence

Introduction de sauts dans la méthode

Le théorème 4.4.1 affirme que la suite des erreurs décrit asymptotiquement une spirale autour de 0. Nous proposons ici une légère modification de la récurrence et de l'algorithme de coût de calcul additionnel négligeable. L'idée consiste à suivre les "cordes" de la spirale plutôt que la spirale elle-même.

Nous avons montré que le taux de convergence dépend du plus grand angle θ_1 . Plus cet angle sera proche de π , plus rapide sera la convergence. Maintenant, si nous appliquons plusieurs fois \mathcal{F} avant de moyenner, disons L fois, alors l'ensemble des angles $(\theta_1, \dots, \theta_r)$ sera remplacé par $(L\theta_1, \dots, L\theta_r)$. Par conséquent, si certains angles sont petits, il peut être profitable d'alterner le nombre d'applications de \mathcal{F} avant d'effectuer la relaxation. Formellement, on choisit une séquence de $a + 1$ entiers L_0, \dots, L_a avec $L_0 = 1$ correspondant au nombre de ces applications, et en posant $\sigma = \sum_{l=0}^a L_l$, on applique la récurrence :

$$s^{(q+1)\sigma} = \frac{1}{2} (I + \mathcal{F}^{L_0}) \circ \dots \circ \frac{1}{2} (I + \mathcal{F}^{L_a}) s^{k\sigma} \quad (4.17)$$

L'exposant de s fournit le nombre de calculs de l'opérateur \mathcal{F} requis. La figure 4.4.2 montre graphiquement l'accélération obtenue pour plusieurs choix de séquence, à nombre d'applications de \mathcal{F} constant. L'opérateur sous-jacent :

$$\mathcal{R} = \frac{1}{2} (I + \mathcal{F}) \circ \frac{1}{2} (I + \mathcal{F}^{L_1}) \circ \dots \circ \frac{1}{2} (I + \mathcal{F}^{L_a})$$

est moyenné en tant que composition d'opérateurs moyennés donc la suite converge vers un élément de $\text{fix}(\mathcal{R}) = \bigcap_{l=0}^a \text{fix}(\mathcal{F}^{L_l}) = \text{fix}(\mathcal{F})$. Les itérations modifiées (4.17) donnent lieu à l'algorithme 4.4.1.

Comportement asymptotique

Nous avons :

$$\limsup \frac{\|e^{(q+1)\sigma}\|}{\|e^{q\sigma}\|} = \max_{i=1, \dots, r} \left\{ \left| \prod_{l=0}^a \cos\left(\frac{L_l \theta_i}{2}\right) \right| \right\}$$

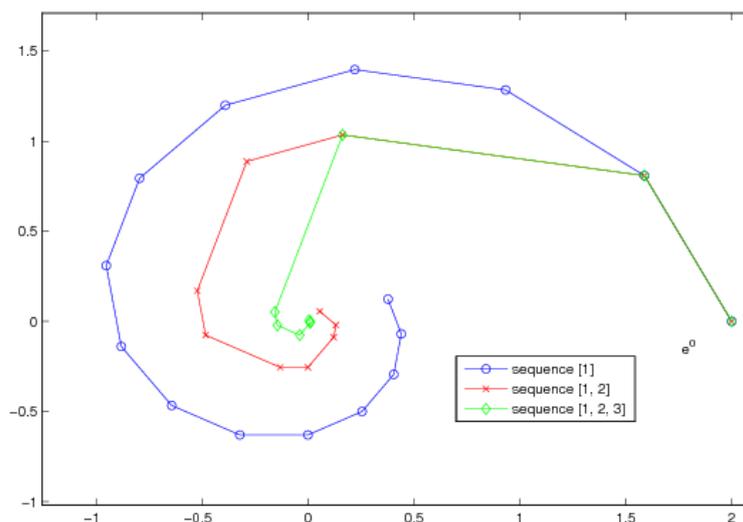


FIGURE 4.4.2 – Amélioration du taux de convergence pour plusieurs choix de séquences.

Algorithme 4.4.1 Algorithme de Douglas-Rachford avec sauts de moyenne

1. $k = 0; q = 0$ Choisir $y^0, u^0 \in \mathbb{R}^n$
 2. Pour $l = 0, \dots, a$ faire //calcul de \mathcal{R}
 3. $(\bar{y}^k, \bar{u}^k) = (y^k, u^k)$
 4. Pour $j = 1, \dots, L_l$ faire //calcul de \mathcal{F}^{L_l}
 5. $\tilde{y}^{k+1} = \arg \min_y F(y) + \frac{1}{2} \langle y - \bar{y}^k, \Lambda(y - \bar{y}^k) \rangle - \langle \bar{u}^k, y \rangle$
 6. $\tilde{u}^{k+1} = \bar{u}^k + \Lambda(\bar{y}^k - \tilde{y}^{k+1})$
 7. $\bar{y}^{k+1} = \arg \min_y C(y) + \frac{1}{2} \langle y - \tilde{y}^{k+1}, \Lambda(y - \tilde{y}^{k+1}) \rangle + \langle \tilde{u}^{k+1}, y \rangle$
 8. $\bar{u}^{k+1} = \tilde{u}^{k+1} - \Lambda(\tilde{y}^{k+1} - \bar{y}^{k+1})$
 9. $k \leftarrow k + 1$
 10. Fin Pour
 11. $(y^k, u^k) \leftarrow \alpha(\bar{y}^k, \bar{u}^k) + (1 - \alpha)(y^{k-L_l}, u^{k-L_l})$
 12. Fin Pour
-

Afin d'exprimer la vitesse de convergence en termes de nombre d'appels à l'opérateur \mathcal{F} , nous nous ramenons à un taux moyenné consistant en la racine σ -ième :

$$\tau(\theta) = \max_{i=1, \dots, r} \left\{ \left| \prod_{l=0}^a \cos \left(\frac{L_l \theta_i}{2} \right) \right| \right\}^{\frac{1}{\sigma}}$$

La question de l'accélération ou non de la convergence en comparaison de l'algorithme classique ($a = 0$) dépend de la répartition des θ_i dans l'intervalle $]0, \pi[$. Si θ_1 est proche de 0, alors la convergence sera accélérée. Au contraire, lorsque tous les θ_i sont proches de π , la méthode classique est déjà très rapide et effectuer plusieurs applications du pliage avant de relaxer ralentira la convergence. Certains calculs de \mathcal{F} seront utilisés pour faire de petits pas tandis qu'ils auraient été utilisés pour faire de grands pas dans la méthode originelle.

Nous considérons sur la figure 4.4.3 le cas où $\dim(V(e^0)) = 4$. L'aire bleutée est une sous-partie de $]0, \pi[\times]0, \pi[$, $\theta_1 \leq \theta_2$ où le taux de convergence décroît pour $a = 1$ et $L_1 = 2$ *i.e.* l'aire où $\tau(\theta) \leq \rho(\theta) = |\cos(\theta_1)|$. Il semble difficile de déduire *a priori* les valeurs

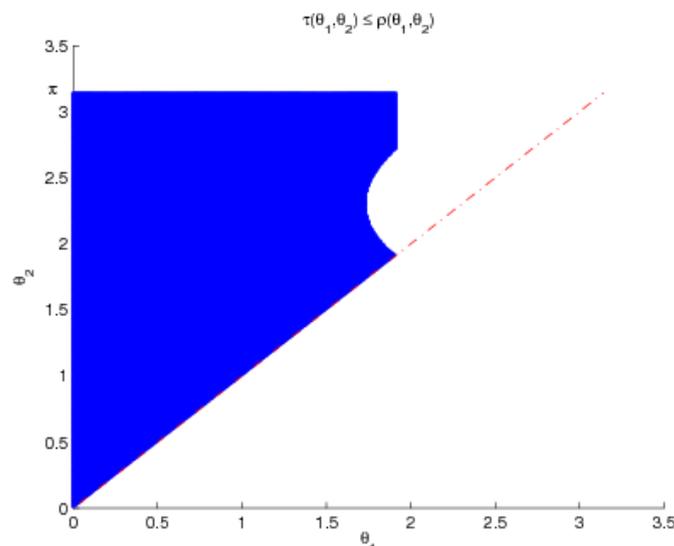


FIGURE 4.4.3 – Domaine de $]0, \pi[\times]0, \pi[$, $\theta_1 \leq \theta_2$ où la convergence est accélérée pour $a = 1$, $L_1 = 2$

de ces angles d'après les données du problème et à plus forte raison de déterminer une séquence de relaxation adéquate. Nous effectuons maintenant quelques essais de séquences en observant les résultats sur un problème de gestion de production.

Expériences numériques

Nous comparons ici l'algorithme classique ($a = 0$) et sa version modifiée ($a = 1$ et $L_1 = 2$) sur un problème de gestion de centrales électriques. Nous appliquons les deux algorithmes pour plusieurs valeurs d'un paramètre de mise à l'échelle scalaire λ . L'économie en terme du nombre de résolutions de sous-problèmes et de temps CPU est reporté sur les figures 4.4.4 et 4.4.5. Le chiffre indiqué représente le temps de calcul de l'algorithme modifiée rapporté à celui de l'algorithme classique.

Dans la plupart des cas, on observe entre 20% et 40% d'économie de temps CPU. De façon intéressante, l'économie maximum est obtenue pour les valeurs de λ où la méthode

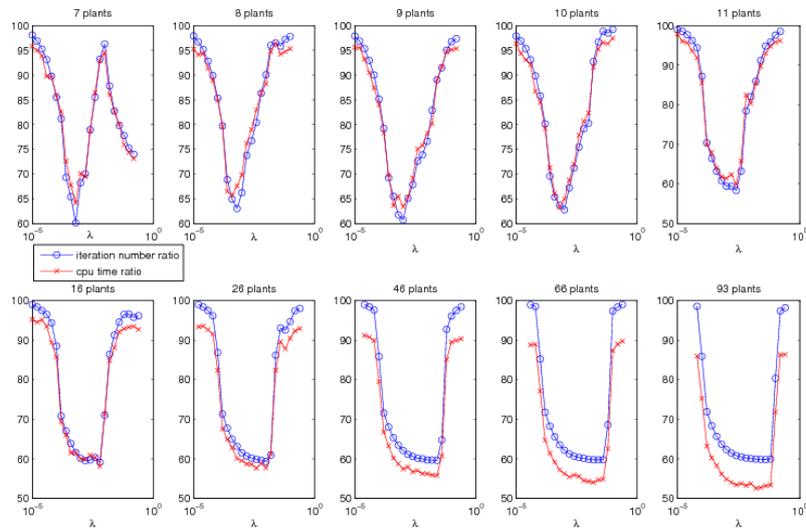


FIGURE 4.4.4 – Économie en terme du nombre de résolution des sous-problèmes et de temps CPU sur un problème à 50 périodes avec 6 réservoirs

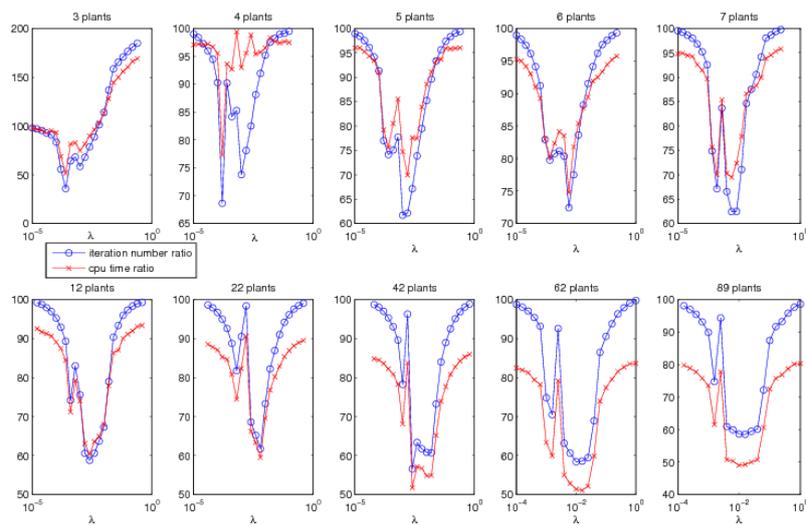


FIGURE 4.4.5 – Économie en terme du nombre de résolution des sous-problèmes et de temps CPU sur un problème à 350 périodes avec 2 réservoirs

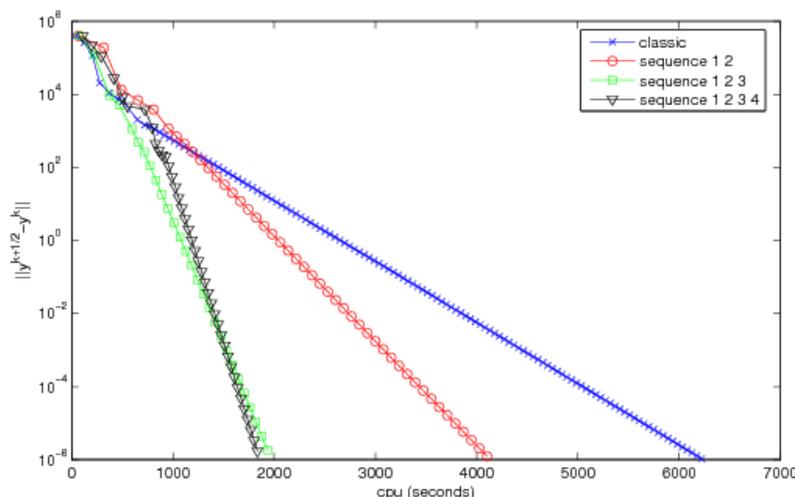


FIGURE 4.4.6 – Évolution de $\|s^{q\sigma} - s^{q\sigma-1}\|^2$ pour différents choix de séquences.

classique est la plus efficace. Nous avons aussi comparé plusieurs choix de séquences, $[1]$; $[1, 2]$; $[1, 2, 3]$ et $[1, 2, 3, 4]$ sur un problème plus grand. L'algorithme est stoppé lorsque $\|s^{q\sigma} - s^{q\sigma-1}\|^2$ est plus petit que 10^{-6} . Cette quantité est tracée en fonction du temps CPU sur la figure 4.4.6. Sur cet exemple, nous avons réussi à réduire le temps de calcul d'un facteur 3 à l'aide des deux dernières séquences.

4.5 Méthode itérative par bloc

La version Δ de l'algorithme 3.4.2 avec $\alpha_k \equiv \frac{1}{2}$, appliquée au problème décomposable (1.2) :

$$\min_y f(y) = \sum_{i=1}^p F_i(y_i) + C(y_1, \dots, y_p)$$

donne l'algorithme 4.5.1. Nous nous intéressons dans cette section à la possibilité de ne

Algorithme 4.5.1 Algorithme de Douglas-Rachford (version Δ) appliqué à la décomposition

1. $k = 0$; Choisir $y^0, u^0 \in \mathbb{R}^n$
 2. Pour $i = 1, \dots, p$ faire
 3. $\tilde{y}_i^{k+1} \in \arg \min_{y_i} F_i(y_i) + \langle y_i - y_i^k, \Lambda_i(y_i - y_i^k) \rangle - u_i^k$
 4. $\tilde{u}_i^{k+1} = u_i^k + \Lambda_i(y_i^k - \tilde{y}_i^{k+1})$
 5. Fin pour
 6. $\bar{y}^{k+1} = \arg \min_y C(y) + \langle y - \tilde{y}^{k+1}, \Lambda(y - \tilde{y}^{k+1}) \rangle + \tilde{u}^{k+1}$
 7. $\bar{u}^{k+1} = \tilde{u}^{k+1} - \Lambda(\tilde{y}^{k+1} - \bar{y}^{k+1})$
 8. $(y^{k+1}, u^{k+1}) = \frac{1}{2}(y^k, u^k) + \frac{1}{2}(\bar{y}^{k+1}, \bar{u}^{k+1})$
 9. $k \leftarrow k + 1$; Aller à l'étape 2
-

résoudre qu'une partie des sous-problèmes durant les étapes 2 à 5, le but étant de réaliser des économies en termes de temps de calcul. Ceci peut être utile dans deux cas :

- lorsque le temps de résolution de certains sous-problèmes est plus long que les autres. Il est alors possible de résoudre les sous-problèmes les plus rapides plusieurs fois pendant que les plus longs ne le sont qu'une seule et ainsi progresser vers l'optimum global plus rapidement.
- lorsque l'architecture matérielle ne permet pas de résoudre tous les sous-problèmes simultanément en parallèle. Nous pouvons alors choisir de ne résoudre qu'un nombre de sous-problèmes égal au nombre de processeurs disponibles

L'opérateur effectuant une itération de l'algorithme 4.5.1 est noté :

$$\mathcal{P} = \frac{1}{2}I + \frac{1}{2}\mathcal{N}_2 \circ \mathcal{N}_1$$

\mathcal{N}_1 possède une structure décomposable par bloc :

$$\mathcal{N}_1 = \begin{pmatrix} \mathcal{N}_{1,1} & & \\ & \ddots & \\ & & \mathcal{N}_{1,p} \end{pmatrix}$$

dont le calcul de chaque bloc $\mathcal{N}_{1,i} : \mathbb{R}^{n_i} \rightarrow \mathbb{R}^{n_i}$ correspond à la résolution du sous-problème i . Le calcul de \mathcal{N}_2 correspond quant à lui à l'étape de coordination. Nous notons aussi :

$$\mathcal{Q} = I - \mathcal{P}$$

l'opérateur régularisé de $\mathcal{T}_2 \odot \mathcal{T}_1$ qui permet de réécrire l'algorithme 3.4.2 sous la forme 4.5.2.

Algorithme 4.5.2 Algorithme de Douglas-Rachford

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
 2. Pour $i = 1, \dots, p$ faire
 3. $s^{k+1} = s^k - \mathcal{Q}(s^k)$
 4. Fin pour
 5. $k \leftarrow k + 1$; Aller à l'étape 2
-

La méthode que nous proposons est basée sur le principe suivant. Lors d'une itération de l'algorithme 4.5.2 donnée, si nous projetons le vecteur $\mathcal{Q}(s^k)$ sur l'espace \mathcal{G}_i :

$$\mathcal{G}_i = \{0\} \times \dots \times \{0\} \times \mathbb{R}^{n_i} \times \{0\} \times \dots \times \{0\}$$

avant d'effectuer l'étape 3, alors l'itérée suivante s^{k+1} est obtenue à partir de l'itérée courante s^k en remplaçant uniquement le sous-vecteur i par le bloc correspondant dans la solution issue de la coordination. Par conséquent, durant l'itération suivante, seul le sous-problème i devra être résolu de nouveau, les autres l'ayant déjà été.

La projection permet ainsi de simuler le fait que certains sous-problèmes ne sont pas résolus et gardent la solution qu'ils avaient à l'itération précédente. Afin d'explicitier l'algorithme que nous venons de décrire, introduisons quelques notations. $\mathcal{I}_k \subset \{1, \dots, p\}$

est l'ensemble des indices des sous-problèmes qui ne sont pas résolus à l'itération $k+1 \geq 1$ et pour tout $\mathcal{I} \subset \{1, \dots, p\}$, nous notons la somme directe :

$$\mathcal{G}_{\mathcal{I}} = \bigoplus_{i \in \mathcal{I}} \mathcal{G}_i$$

Π_i et $\Pi_{\mathcal{I}}$ les projecteurs sur \mathcal{G}_i et $\mathcal{G}_{\mathcal{I}}$ ainsi que les opérateurs projetés :

- $\mathcal{Q}_i = \Pi_i \circ \mathcal{Q}$ et $\mathcal{Q}_{\mathcal{I}} = \Pi_{\mathcal{I}} \circ \mathcal{Q}$
- $\mathcal{P}_i = I - \mathcal{Q}_i$ et $\mathcal{P}_{\mathcal{I}} = I - \mathcal{Q}_{\mathcal{I}}$

La méthode proposée prend alors la forme de l'algorithme 4.5.3. où d'une manière faisant

Algorithme 4.5.3 Algorithme de Douglas-Rachford par bloc

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
 2. Pour $i \in \mathcal{I}_k$ faire
 3. $s_i^{k+1} = \mathcal{P}_i(s^k) = s_i^k - \mathcal{Q}_i(s^k)$
 4. Fin pour
 5. Pour $i \notin \mathcal{I}_k$ faire
 6. $s_i^{k+1} = s_i^k$
 7. Fin pour
 8. $k \leftarrow k + 1$; Aller à l'étape 2
-

intervenir explicitement la résolution partielle des sous-problèmes, l'algorithme 4.5.4.

Algorithme 4.5.4 Algorithme de Douglas-Rachford par bloc (détaillé)

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
 2. Pour $i = 1, \dots, p$ faire
 3. $d_i^{k+1} = \mathcal{N}_{1,i}(s_i^k)$
 4. Fin pour
 5. $s^{k+1} = s^k - \frac{1}{2} \Pi_{\mathcal{I}_k}(s^k - \mathcal{N}_2 d^k)$
 6. $k = 1$
 7. Pour $i = 1, \dots, p$ faire
 8. Si $i \in \mathcal{I}_{k-1}$ alors
 9. $d_i^{k+1} = d_i^k$
 10. Sinon
 11. $d_i^{k+1} = \mathcal{N}_{1,i}(s_i^k)$
 12. Fin si
 13. Fin pour
 14. $s^{k+1} = s^k - \frac{1}{2} \Pi_{\mathcal{I}_k}(s^k - \mathcal{N}_2 d^k)$
 15. $k \leftarrow k + 1$; Aller à l'étape 7
-

Les implémentations itératives par bloc, et plus généralement asynchrones de l'algorithme 3.4.2 ont beaucoup été étudiées par Bertsekas, Tsitsiklis et al ([7], [121], [122], [8])

pour des classes générales d'opérateurs comme les gradients dans un contexte déterministe ou bien stochastique. Plus récemment, dans [5] Benhamed fournit des conditions suffisantes pour la convergence de la méthode du point proximal asynchrone. Ces conditions sont quelque peu sévères; elles imposent notamment que \mathcal{P} soit une contraction pour la norme $\|s\|_\infty = \max_{i=1,\dots,p} \|s_i\|$ ou plus généralement une norme infinie pondérée. Cette condition n'est généralement pas satisfaite pour l'opérateur de Douglas-Rachford en pratique. Les expérimentations numériques ([111]) montrent que les implantations asynchrones divergent si l'on s'éloigne trop de la version synchrone.

Nous proposons ici de forcer la suite des itérés à satisfaire une certaine propriété de monotonie (définition 4.5.1) qui assure la convergence. Nous ajoutons pour cela un mécanisme de contrôle qui teste à la fin de chaque itération si l'ensemble \mathcal{I}_k qui a été choisi a permis de satisfaire cette propriété; en l'enrichissant et en recommençant l'itération dans le cas contraire.

4.5.1 Algorithme

Une propriété importante dans l'analyse de la convergence des algorithmes de type proximaux est la forte Fejér-monotonie ([108], [3], [20]) :

Définition 4.5.1. *Une suite $\{x^k\}_k$, $x^k \in \mathbb{R}^n$ est fortement Fejér-monotone de rapport a relativement à l'ensemble $\bar{X} \subset \mathbb{R}^n$ si pour tout $\bar{x} \in \bar{X}$:*

$$\|x^{k+1} - \bar{x}\|^2 \leq \|x^k - \bar{x}\|^2 - a \|x^{k+1} - x^k\|^2$$

Notons que cette propriété implique en particulier le caractère borné et la régularité asymptotique de $\{x^k\}_k$ (c'est à dire $\|x^{k+1} - x^k\| \rightarrow 0$), mais pas la convergence en général. La suite générée par l'algorithme du point proximal (algorithme 2.2.4 avec $\alpha_k \equiv \alpha$) et en particulier par l'algorithme de Douglas-Rachford 4.5.2 sont par exemple Fejér-monotone par rapport à $\text{fix}(\mathcal{P})$. En effet, le lemme B.1.1 donne pour tout $s^* \in \text{fix}(\mathcal{P})$:

$$\|s^{k+1} - s^*\|^2 \leq \|s^k - s^*\|^2 - \|s^{k+1} - s^k\|^2$$

De même, toujours d'après le lemme B.1.1, la suite duale :

$$q^k \stackrel{\text{def}}{=} Q(s^k) = s^{k+1} - s^k \tag{4.18}$$

est elle aussi Fejér-monotone relativement au singleton $\{0\}$:

$$\|q^{k+1}\|^2 \leq \|q^k\|^2 - \|q^{k+1} - q^k\|^2$$

Nous proposons de forcer pour un certain $a \leq 1$ donné, la forte Fejér-monotonie de rapport a de la suite des $\{q^k\}_k$ générée par l'algorithme 4.5.3. Nous procédons pour cela par essais. Nous calculons d'abord $\mathcal{Q}_{\mathcal{I}_k}$ pour un sous-ensemble prédéterminé d'indices \mathcal{I}_k . Si la a -Fejér-monotonie forte n'est pas atteinte, alors nous enrichissons \mathcal{I}_k jusqu'à obtenir l'inégalité désirée. Notons que nous sommes certains de l'obtenir, quitte à inclure tous les indices dans \mathcal{I}_k d'après le lemme B.1.1. L'algorithme 4.5.5 correspondant est décrit par le schéma de la figure 4.5.1.

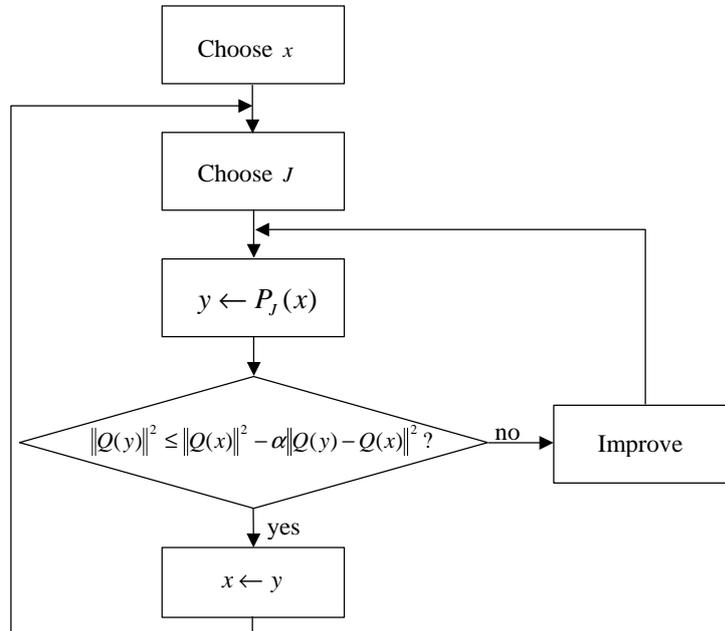


FIGURE 4.5.1 – Schéma algorithmique de la Méthode de Douglas-Rachford itérative par bloc

Algorithme 4.5.5 Algorithme de Douglas-Rachford par bloc avec contrôle de convergence

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
 2. Choisir $\mathcal{I}_k \subset \{1, \dots, p\}$
 3. Faire
 4. $\tilde{s}^{k+1} = \mathcal{P}_{\mathcal{I}_k} s^k$
 5. $\mathcal{I}_k \leftarrow \mathcal{I}_k \cup \{i\}$
 6. Tant que $\|Q(\tilde{s}^{k+1})\|^2 > \|Q(s^k)\|^2 - a \|Q(\tilde{s}^{k+1}) - Q(s^k)\|^2$
 7. $s^{k+1} \leftarrow \tilde{s}^{k+1}$
 8. $k \leftarrow k + 1$; Aller à l'étape 2
-

4.5.2 Analyse de la convergence

Nous aurons besoin, pour établir la convergence de l'algorithme 4.5.5 de l'hypothèse suivante.

Hypothèse 4.5.1. *Il existe une constante L telle que pour tout $i = 1, \dots, p$:*

$$\forall l \geq 0 \quad i \in \bigcap_{k=l}^{l+L} \mathcal{I}_k$$

Pour tout k , nous notons $q^k = \mathcal{Q}(s^k)$. Nous commençons par montrer que si la suite $\{(s^k, q^k)\}_k$ générée par l'algorithme 4.5.5 admet un point d'accumulation alors il est de la forme $(\bar{s}, \bar{q}) = (\bar{s}, 0)$. Introduisons pour cela l'ensemble des indices des composantes nulles de \bar{q} :

$$\mathcal{I}_{zer} = \{i \in \{1, \dots, p\} / \Pi_i \bar{q} = 0\} \quad (4.19)$$

ainsi que sont complémentaire :

$$\mathcal{I}_+ = \{i \in \{1, \dots, p\} / \Pi_i \bar{q} \neq 0\} = \{1, \dots, p\} \setminus \mathcal{I}_{zer} \quad (4.20)$$

Le lemme technique suivant montre que la sous-suite $\{(s^{\phi(\nu)}, q^{\phi(\nu)})\}_\nu$ convergente vers (\bar{s}, \bar{q}) peut être choisie de telle sorte que $\mathcal{I}^{\phi(\nu)}$ contient infiniment souvent le même indice de \mathcal{I}_+ .

Lemme 4.5.1. *Soit (\bar{s}, \bar{q}) un point d'accumulation de la suite $\{(s^k, Q(s^k))\}_k$ générée par l'algorithme 4.5.5, \mathcal{I}_{zer} défini par (4.19) et \mathcal{I}_+ défini par (4.20). Si $\mathcal{I}_+ \neq \emptyset$, alors il existe $i \in \mathcal{I}_+$ et une sous-suite $\{(s^{\phi(\nu)}, q^{\phi(\nu)})\}_\nu \rightarrow (\bar{s}, \bar{q})$ tels que :*

$$i \in \mathcal{I}_{\phi(\nu)} \quad \forall \nu \geq 0$$

Démonstration. Soit $\{(s^{\varphi(\nu)}, q^{\varphi(\nu)})\}_\nu$ une sous-suite convergente vers (\bar{s}, \bar{q}) et considérons les L suites $\{s^{\varphi(\nu)+l}\}_\nu$ obtenues par translation d'indices pour $1 \leq l \leq L$ où L est la constante de l'hypothèse 4.5.1. Pour tout ν , au moins l'un des $L+1$ ensembles $\mathcal{I}_{\varphi(\nu)}, \mathcal{I}_{\varphi(\nu)+1}, \dots, \mathcal{I}_{\varphi(\nu)+L}$ intersecte \mathcal{I}_+ . Il existe donc au moins un $l \in \{0, \dots, L\}$ tel que :

$$\mathcal{I}_+ \cap \limsup \{\mathcal{I}_{\varphi(\nu)+l}\}_\nu \neq \emptyset$$

Notons l^* le plus petit de ces indices :

$$l^* = \min \{l \in \{0, \dots, L\} / \mathcal{I}_+ \cap \limsup \{\mathcal{I}_{\varphi(\nu)+l}\}_\nu \neq \emptyset\}$$

et montrons d'abord que $\{(s^{\varphi(\nu)+l^*}, q^{\varphi(\nu)+l^*})\}_\nu$ converge aussi vers (\bar{s}, \bar{q}) . Par construction de l^* , il existe un rang $K \geq 0$ tel que pour tout $0 \leq l \leq l^* - 1$ et pour $\nu \geq K$, $\mathcal{I}_{\varphi(\nu)+l} \subset \mathcal{I}_{zer}$. Or, nous avons par ailleurs :

$$s^{\varphi(\nu)+l^*} - \bar{s} = \left(s^{\varphi(\nu)} - \sum_{l=0}^{l^*-1} \gamma^{\varphi(\nu)+l} \right) - \bar{s} \quad (4.21)$$

$$\|s^{\varphi(\nu)+l^*} - \bar{s}\| \leq \|s^{\varphi(\nu)} - \bar{s}\| + \sum_{l=0}^{l^*-1} \|\gamma^{\varphi(\nu)+l}\| \quad (4.22)$$

et pour $0 \leq l \leq l^*$:

$$\begin{aligned}
\|\gamma^{\varphi(\nu)+l}\| &= \left\| \mathcal{Q}_{\mathcal{I}_{\varphi(\nu)+l}}(s^{\varphi(\nu)+m}) \right\| \\
&\leq \left\| \mathcal{Q}_{\mathcal{I}_{zer}}(s^{\varphi(\nu)+l}) \right\| \\
&\leq \left\| \mathcal{Q}_{\mathcal{I}_{zer}}(s^{\varphi(\nu)}) \right\| + \sum_{l'=0}^{l-1} \left\| \mathcal{Q}_{\mathcal{I}_{zer}}(s^{\varphi(\nu)+l'+1}) - \mathcal{Q}_{\mathcal{I}_{zer}}(s^{\varphi(\nu)+l'}) \right\| \\
&\leq \left\| \mathcal{Q}_{\mathcal{I}_{zer}}(s^{\varphi(\nu)}) \right\| + e_\nu
\end{aligned} \tag{4.23}$$

avec :

$$e_\nu = \sum_{l'=0}^{l^*-1} \left\| q^{\varphi(\nu)+l'+1} - q^{\varphi(\nu)+l'} \right\|$$

En injectant (4.23) dans (4.22), nous obtenons :

$$\|s^{\varphi(\nu)+l^*} - \bar{s}\| \leq \|s^{\varphi(\nu)} - \bar{s}\| + l^* \left\| \mathcal{Q}_{\mathcal{I}_{zer}}(s^{\varphi(\nu)}) \right\| + l^* e_\nu$$

D'une part, $\left\| \mathcal{Q}_{\mathcal{I}_{zer}}(s^{\varphi(\nu)}) \right\| \rightarrow \left\| \mathcal{Q}_{\mathcal{I}_{zer}}(\bar{s}) \right\| = 0$ par continuité de $\mathcal{Q}_{\mathcal{I}_{zer}}$, d'autre part $e_\nu \rightarrow 0$ par la régularité asymptotique de $\{q^k\}_k$. Ainsi, $\{s^{\varphi(\nu)+l^*}\}_\nu$ converge elle aussi vers \bar{s} . Le fait que $\{q^{\varphi(\nu)+l^*}\}_\nu$ converge vers \bar{q} vient de l'inégalité triangulaire :

$$\|q^{\varphi(\nu)+l^*} - \bar{q}\| \leq \|q^{\varphi(\nu)} - \bar{q}\| + l^* e_\nu$$

Rappelons enfin que par construction :

$$\mathcal{I}_+ \cap \limsup \{\mathcal{I}_{\varphi(\nu)+l^*}\}_\nu \neq \emptyset$$

Il existe donc $i \in \mathcal{I}_+$ tel que $i \in \limsup \{\mathcal{I}_{\varphi(\nu)+l^*}\}_\nu$, c'est à dire tel qu'il existe $\psi(\cdot)$ et $i \in \mathcal{I}_{\varphi(\psi(\nu))+l^*}$ pour tout ν . Nous pouvons alors conclure en posant $\phi(\cdot) = \varphi(\psi(\cdot)) + l^*$. \square

Lemme 4.5.2. *Si la suite $\{(s^k, \mathcal{Q}(s^k))\}_k$ générée par l'algorithme 4.5.5 admet un point d'accumulation (\bar{s}, \bar{q}) , alors $\bar{q} = 0$.*

Démonstration. Nous allons montrer que pour tout point d'accumulation \bar{q} l'ensemble \mathcal{I}_{zer} défini par (4.19) est nécessairement vide en suivant un raisonnement par l'absurde qui suppose sa non vacuité. Soit $(\bar{s}^{(0)}, \bar{q})$ un point d'accumulation de la suite $\{(s^k, q^k)\}_k$. D'après le lemme 4.5.1, nous pouvons choisir $i \in \mathcal{I}_+$ ainsi qu'une sous-suite $\{s^{\phi(\nu)}, q^{\phi(\nu)}\}_\nu \rightarrow (\bar{s}^{(0)}, \bar{q})$ tels que $i \in \mathcal{I}_{\phi(\nu)}$ pour tout ν . Il existe alors un sous ensemble d'indices $\mathcal{I}^{(0)}$ et une suite extraite $\mathcal{I}_{\phi(\psi(\nu))} \ni i$ telle que $\mathcal{I}_{\psi(\phi(\nu))} = \mathcal{I}^{(0)}$ pour tout ν . La sous-suite $\{s^{\psi(\phi(\nu))}, q^{\psi(\phi(\nu))}\}_\nu \rightarrow (\bar{s}^{(0)}, \bar{q})$. Posons enfin $\bar{s}^{(1)} = \mathcal{P}_{\mathcal{I}^{(0)}}(\bar{s}^{(0)})$. Nous avons pour tout k , $\|\mathcal{Q}(s^{k+1})\|^2 \leq \|\mathcal{Q}(s^k)\|^2 - \alpha \|\mathcal{Q}(s^{k+1}) - \mathcal{Q}(s^k)\|^2$. En particulier, pour tout ν ,

$$\|\mathcal{Q}(s^{\psi(\phi(\nu))+1})\|^2 \leq \|\mathcal{Q}(s^{\psi(\phi(\nu))})\|^2 - \alpha \|\mathcal{Q}(s^{\psi(\phi(\nu))+1}) - \mathcal{Q}(s^{\psi(\phi(\nu))})\|^2$$

Le passage à la limite donne :

$$\|\mathcal{Q}(\bar{s}^{(1)})\|^2 \leq \|\mathcal{Q}(\bar{s}^{(0)})\|^2 - \alpha \|\mathcal{Q}(\bar{s}^{(1)}) - \mathcal{Q}(\bar{s}^{(0)})\|^2$$

mais $\{\|\mathcal{Q}(s^k)\|\}_k$ converge puisque $\{\mathcal{Q}(s^k)\}_k$ est Fejer-monotone donc

$$\|\mathcal{Q}(\bar{s}^{(1)}) - \mathcal{Q}(\bar{s}^{(0)})\| \leq 0$$

et $\mathcal{Q}(\bar{s}^{(1)}) = \mathcal{Q}(\bar{s}^{(0)})$. En répétant ce processus en partant de $\bar{s}^{(1)}$, nous obtenons par construction une suite de points d'accumulations $(\bar{s}^{(0)}, \bar{s}^{(1)}, \dots, \bar{s}^{(K)}, \dots)$ satisfaisant :

$$\bar{q} = \mathcal{Q}(\bar{s}^{(0)}) = \mathcal{Q}(\bar{s}^{(1)}) = \dots = \mathcal{Q}(\bar{s}^{(K)}) = \dots$$

avec $\mathcal{I}^{(k)} \cap \mathcal{I}_+ \neq \emptyset$ pour tout k . Le point d'accumulation $\bar{s}^{(K)}$ s'écrit :

$$\bar{s}^{(K)} = \bar{s}^{(0)} - \sum_{l=0}^{K-1} \Pi_{\mathcal{I}^{(l)}}(\bar{q}) \quad (4.24)$$

Soit s^* un zéro de \mathcal{Q} . Nous avons, puisque \mathcal{Q} est pseudo-contractant :

$$\begin{aligned} \|\mathcal{Q}(\bar{s}^{(K)})\|^2 &\leq \langle \bar{s}^{(K)} - s^*, \mathcal{Q}(\bar{s}^{(K)}) \rangle \\ \|\bar{q}\|^2 &\leq \langle \bar{s}^{(K)} - s^*, \bar{q} \rangle \end{aligned}$$

qui, combiné à (4.24) donne :

$$\sum_{l=0}^{K-1} \langle \Pi_{\mathcal{I}^{(l)}}(\bar{q}), \bar{q} \rangle \leq \langle \bar{s}^{(0)} - s^*, \bar{q} \rangle - \|\bar{q}\|^2 \quad (4.25)$$

Par construction, $\mathcal{I}_+ \cap \limsup \{\mathcal{I}^{(l)}\}_l \neq \emptyset$. En choisissant $i \in \mathcal{I}_+ \cap \limsup \{\mathcal{I}^{(l)}\}_l$, et en extrayant une sous-suite $\mathcal{I}^{(\psi(\nu))} \ni i$, l'inégalité (4.25) s'écrit :

$$\sum_{\nu=0}^N \|\Pi_i(\bar{q})\|^2 \leq \sum_{\nu=0}^N \langle \Pi_{\mathcal{I}^{(\psi(\nu))}}(\bar{q}), \bar{q} \rangle \leq \langle \bar{s}^{(0)} - s^*, \bar{q} \rangle - \|\bar{q}\|^2$$

d'où $\|\Pi_i(\bar{q})\| = 0$ contradisant ainsi le choix de $i \in \mathcal{I}_+$. Nous en déduisons donc que $\mathcal{I}_+ = \emptyset$. \square

Proposition 4.5.1. *Soit $\{(s^k, \mathcal{Q}(s^k))\}_k$ la suite générée par l'algorithme 4.5.5. Si l'opérateur \mathcal{Q} est coercif, c'est à dire :*

$$\lim_{\|s\| \rightarrow +\infty} \|\mathcal{Q}(s)\| = +\infty$$

alors $\mathcal{Q}(s^k) \rightarrow 0$.

Démonstration. Le test 6 de l'algorithme 4.5.5 implique que $\{q^k\}_k$ est a -fortement Fejer-monotone et donc bornée. La coercivité de \mathcal{Q} implique alors que $\{s^k\}_k$ est bornée elle aussi. La suite $\{(s^k, q^k)\}_k$ admet donc un point d'accumulation $(\bar{s}, 0)$ d'après le lemme 4.5.2 et $\liminf \|q^k\| = 0$. Or, la suite $\{\|q^k\|\}_k$ est décroissante car Fejér-monotone d'où $\lim \|q^k\| = 0$. \square

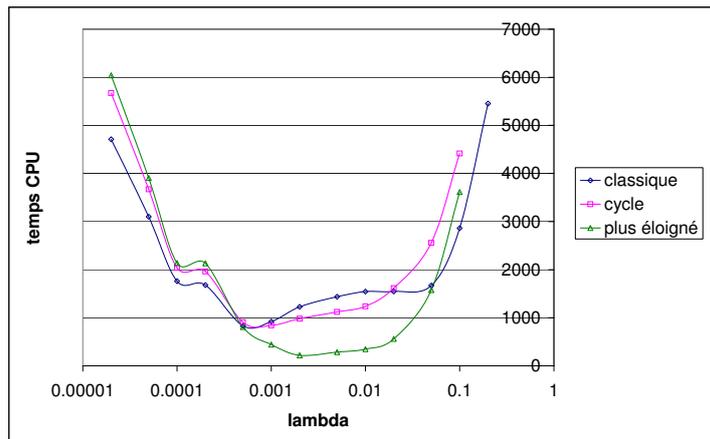


FIGURE 4.5.2 – Temps d’exécution de l’algorithme 4.5.4 pour différentes stratégies de sélection des sous-problèmes.

4.5.3 Expériences numériques

Nous avons appliqué l’algorithme bloc-itératif 4.5.4 à un problème de gestion de 24 réserves hydrauliques. Celles-ci sont gérées sur un arbre de scénario comportant 1750 noeuds. Deux stratégies pour choisir les sous-problèmes à résoudre ont été envisagées.

- La première consiste à résoudre les problèmes cycliquement, les uns après les autres.
- La deuxième choisit en priorité le sous-problème i de composante $\|Q_i(s^k)\|$ la plus élevée.

La figure 4.5.4 montre la façon dont décroît $\|Q(s^k)\|$ en fonction du temps CPU pour ces deux stratégies et pour l’implémentation classique de la méthode. Les sous-problèmes sont incorporés un à un dans l’ensemble \mathcal{I}_k et le test 6 est répété à chaque inclusion d’un nouveau sous-problème. L’algorithme s’arrête lorsque $\|Q(s^k)\| \leq 10^{-3}$ ou bien lorsque le nombre d’itérations dépasse 3000. La stratégie de la composante “la plus éloignée” entraîne la décroissance de $\|Q(s^k)\|$ la plus rapide au début de l’algorithme pour la plupart des valeurs de λ . Le temps d’exécution des trois versions en fonction du paramètre λ est reporté sur la figure 4.5.2. La méthode itérative par bloc semble montrer son intérêt uniquement dans une certaine plage de valeurs, là où la méthode classique est déjà la plus efficace.

Nous représentons sur la figure 4.5.3 la manière dont se répartit le temps de calcul entre coordination et résolution des sous-problèmes pour $\lambda = 0.005$. Nous pouvons observer qu’à partir d’un certain rang, le test de l’étape 6 échoue systématiquement. Tous les sous-problèmes sont alors inclus dans $\mathcal{I}_k = \{1, \dots, p\}$ avant de passer à l’itération suivante et l’algorithme 4.5.4 agit comme l’algorithme classique. Ceci explique la contre-performance des méthodes par bloc dont une telle itération incluant tous les sous-problèmes est à peu près 30% plus coûteuse en temps de calcul que dans la version classique; cet écart provenant des évaluations de \mathcal{N}_2 (étape 14 dans l’algorithme 4.5.4) inutiles. Pour résumer, cette méthode est donc à proscrire si elle est associée à un mauvais facteur d’échelle. Elle devrait dans ce cas être employée uniquement pour les premières itérations puis remplacée par l’algorithme classique ensuite.

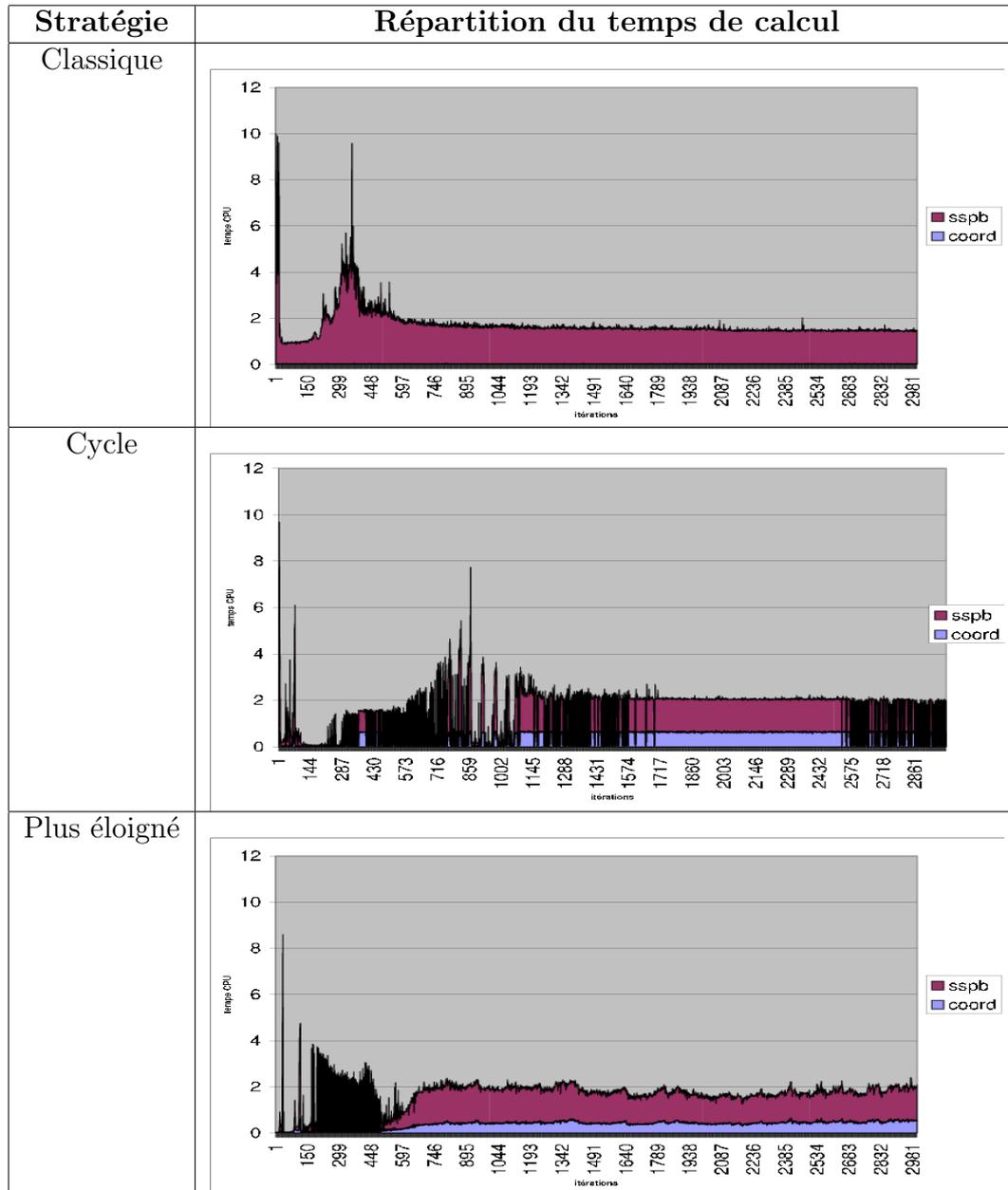


FIGURE 4.5.3 – Répartition du temps de calcul entre résolution des sous-problèmes et coordination pour différentes stratégies dans l’algorithme 4.5.4, $\lambda = 0.005$.

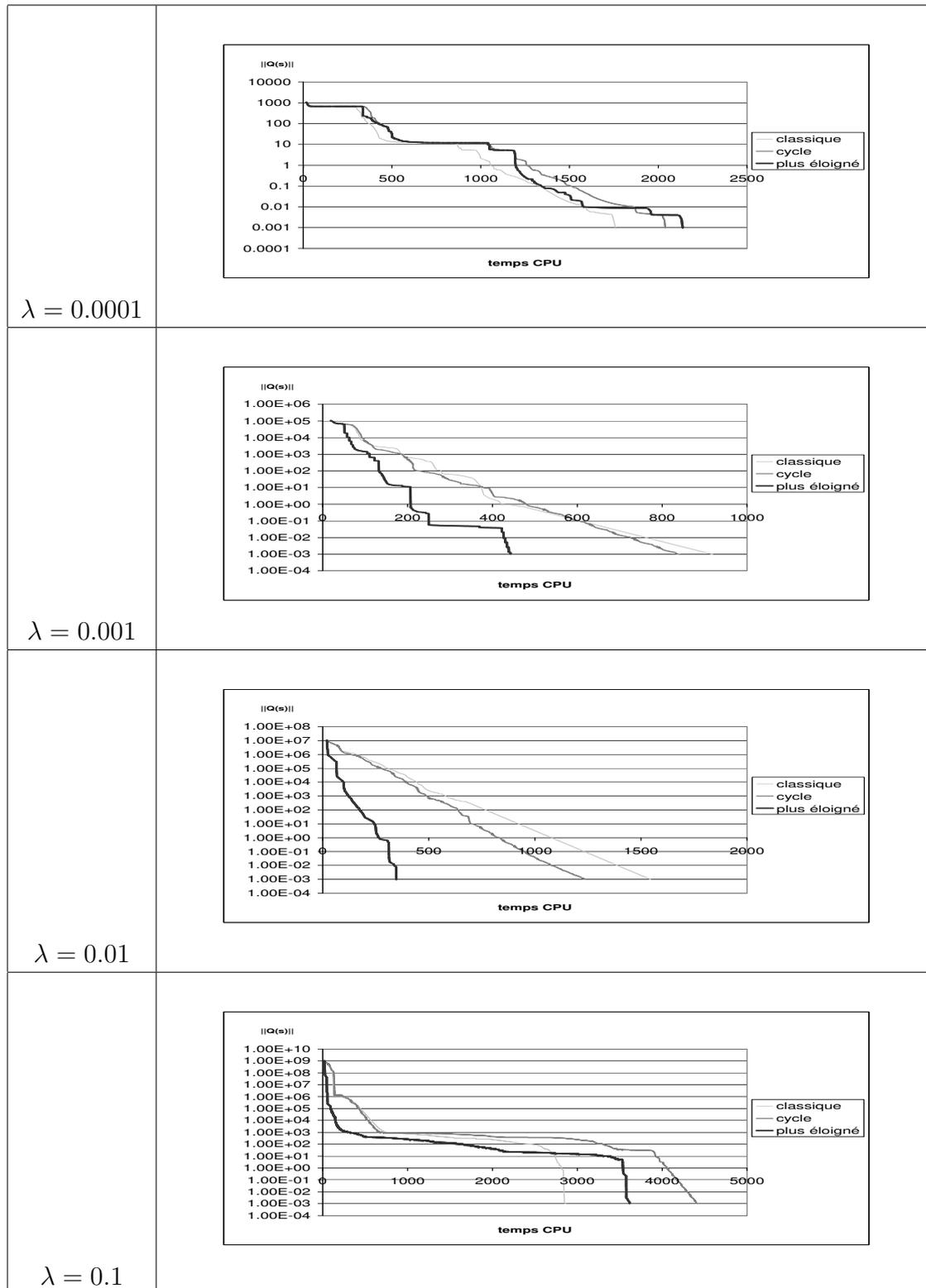


FIGURE 4.5.4 – Profil de $\|Q(s^k)\|$ en fonction du temps de calcul pour différentes stratégies et différentes valeurs de λ dans l'algorithme 4.5.4.

Conclusion

Nous nous sommes intéressés dans ce chapitre à l'amélioration des performances de l'algorithme 3.4.2 et au bon usage des paramètres α_k et Λ_k . Des explications partielles ont été apportées aux observations numériques relevées dans l'introduction et nous ont permis de suivre plusieurs axes d'amélioration. Le premier était de montrer qu'il était possible d'une manière générale de faire varier la matrice de mise à l'échelle Λ_k , pierre d'angle des performances de la méthode, si en contrepartie elle converge assez rapidement. Nous avons ensuite pu déduire d'après les résultats de la section 2.3 des comportements différents suivant la structure des fonctions F et C . Si l'une des deux possède une matrice hessienne généralisée S au point de convergence, alors une valeur idéale du facteur d'échelle est S . Il est alors possible en théorie d'obtenir une convergence superlinéaire en faisant tendre Λ^k vers S et α_k vers 0.

Dans un second temps, l'analyse du cas de l'optimisation polyédrale a montré que la vitesse de convergence s'effectuait en spirale, rejoignant ainsi des résultats connus de [55]. L'algorithme peut alors être accéléré en effectuant des sauts de moyenne et ainsi "prendre la corde" de la spirale.

La dernière section s'inscrit quant à elle à part puisqu'il n'y est pas fait mention de vitesse de convergence. Nous avons montré qu'il était possible, modulo l'ajout d'un mécanisme de contrôle, de ne pas résoudre tous les sous-problèmes à chaque itération en préservant tout de même la convergence globale.

Conclusion et perspectives

Les contributions de cette thèse sont présentes dans les quatre chapitres du mémoire mais en proportion plus importante dans les chapitres 1 et 4, les deux autres servant principalement à présenter les méthodes de régularisation et de décomposition étudiées.

Nous avons mis en évidence dans le chapitre 1 les trois couplages naturels du problème de gestion de production en environnement incertain, offrant plusieurs visions du système global, chacune basée sur une définition différente de sous-système. Cette énumération systématique des couplages permet d'envisager de nouvelles façons de décomposer le problème, résumées jusqu'alors aux découplages spatial, spatio-temporel ou aléatoire. Il serait intéressant dans des travaux futurs, d'effectuer des comparaisons de ces différentes options de décomposition d'un point de vue numérique, travail esquissé lors du stage [14], en y incorporant notamment les décompositions aléa-espace et aléa-espace-temps.

Nous avons aussi proposé deux nouvelles formulations du couplage en aléa présentant l'avantage combiné d'éviter la construction d'un arbre de scénarios et de fournir une meilleure approximation des solutions relatives aux premières périodes. L'intérêt de ces approches demande maintenant à être confirmé sur des problèmes multi-étapes avec un grand nombre de variables en les comparant à l'approche classique par arbre.

Le chapitre 2 sert d'appui à l'étude des méthodes d'éclatement menée dans les chapitres lui succédant. Sa tâche principale est de présenter l'importance de la différentiabilité de l'opérateur de réflexion généralisée dans l'étude de la convergence de l'algorithme du point proximal. Un résultat nouveau de stabilité des lois internes \odot et \star vis à vis de la convergence graphique y est aussi donné, facilitant les analyses des méthodes d'éclatement développées après.

La contribution du chapitre 3 consiste essentiellement en une présentation relativement complète des méthodes d'éclatement et de leurs applications à l'optimisation en éclaircissant plusieurs points techniques souvent obscurs dans la littérature.

Les développements du dernier chapitre sont à notre connaissance originaux. Les formulations des taux de convergence grâce à la semi-différentielle du chapitre 2 y est utilisée intensivement, premièrement dans la section 4.3 où nous nous concentrons sur le cas où un des deux opérateurs est différentiable puis dans 4.4 où nous regardons le cas de l'optimisation polyédrale. Ces deux chapitres nous semblent importants à double titre. Ils offrent une justification aux observations numériques : dépendance de la vitesse de convergence au facteur d'échelle et convergence en spirale, et permettent par la même occasion de mettre en œuvre assez naturellement des techniques d'accélération.

La poursuite de ces travaux nous semble passer par plusieurs étapes. Premièrement, un recouvrement complet, puis une extension des travaux de Spingarn à l'aide du formalisme des semi-dérivées, ensuite une unification des sections 4.3 et 4.4 pour établir un résultat de convergence plus global qui permettrait par exemple de fournir les réglages optimaux des algorithmes en programmation quadratique. Il serait d'autre part intéressant d'étudier l'effet d'un paramètre multi-dimensionnel dans la section 4.4.

L'originalité de la dernière section tient dans l'approche adoptée pour interpréter l'algorithme par blocs, puis analyser sa convergence. Deux progrès théoriques conséquents seraient le relâchement de l'hypothèse de coercivité de l'opérateur \mathcal{Q} et aussi la généralisation à un opérateur seulement moyenné, mais pas nécessairement pseudo-contractant.

D'une manière plus pragmatique, du temps devrait être consacré à l'implantation des idées proposées dans le chapitre 4 sur des problèmes de tailles réelles, afin de confirmer ou bien invalider les résultats encourageants obtenus ici.

Annexe A

Optimisation convexe et inclusions monotones

A.1 Minimisation d'une fonction convexe

Considérons le problème d'optimisation suivant :

$$\underset{x \in \mathbb{R}^n}{\text{Minimiser}} f(x) \tag{A.1}$$

où f est convexe à valeurs dans $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$. Nous noterons :

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

le *domaine* de f et $\text{epi}(f) = \{(x, \nu) \in \mathbb{R}^n \times \mathbb{R} / f(x) \leq \nu\}$ son épigraphe. Nous supposons de plus que f est *propre*, c'est à dire qu'elle ne prend ni la valeur $-\infty$, ni son domaine n'est vide. De façon équivalente, son épigraphe n'est pas vide et ne contient pas non plus de droite verticale $\{(\bar{x}, \nu), \nu \in \mathbb{R}\}$ pour $\bar{x} \in \mathbb{R}^n$.

La classe des fonctions pseudo-polyédrales caractérisée par la définition A.1.1 inclut la plupart des fonctions rencontrées en pratiques, par exemple celles issues de la programmation linéaire et quadratique.

Définition A.1.1 (Fonctions pseudo-polyédrales). $f : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ est pseudo-polyédrale si elle s'écrit comme la somme

$$f = g + \delta_C$$

d'une fonction g de domaine $\text{dom}(g) = \mathbb{R}^n$ et de la fonction indicatrice δ_C d'un polyèdre convexe fermé $C \subset \mathbb{R}^n$.

En particulier, une fonction polyédrale (dont l'épigraphe est un polyèdre) est pseudo-polyédrale.

Afin de faciliter l'analyse des problèmes d'optimisation convexes et de leur algorithmes de résolution, il est commode d'introduire les *opérateurs monotones multivoques*.

A.1.1 Opérateurs multivoques

Un opérateur est *multivoque* s'il peut "prendre plusieurs valeurs". Il s'agit rigoureusement d'une application T de \mathbb{R}^n à valeur dans les parties de \mathbb{R}^n , notée usuellement

$T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ à l'aide d'une double flèche \rightrightarrows . Si $T(x)$ est réduit à un singleton pour tout x , alors nous confondrons T avec l'application correspondante, que nous noterons aussi alors de façon indifférente à l'aide d'une simple flèche $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$. Un opérateur multivoque est entièrement caractérisé par son *graphe* :

$$gr(T) \stackrel{def}{=} \{(y, u) \in \mathbb{R}^n \times \mathbb{R}^n / u \in T(y)\}$$

consistant un sous-ensemble de $\mathbb{R}^n \times \mathbb{R}^n$. L'opérateur inverse T^{-1} existe toujours ; il est défini pour tout $v \in \mathbb{R}^n$ par :

$$T^{-1}(v) \stackrel{def}{=} \{x \in \mathbb{R}^n / v \in T(x)\}$$

ou bien en terme de graphe par $gr(T^{-1}) = \{(u, y) \in \mathbb{R}^n \times \mathbb{R}^n / u \in T(y)\}$. Enfin, le domaine de T est l'ensemble des éléments de \mathbb{R}^n ayant une image non vide :

$$dom(T) \stackrel{def}{=} \{x \in \mathbb{R}^n / T(x) \neq \emptyset\}$$

A.1.2 Opérateurs monotones

Un opérateur $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ est *monotone* si :

$$\forall y, y' \in \mathbb{R}^n, u \in T(y), u' \in T(y') \quad \langle y - y', u - u' \rangle \geq 0$$

Il est *maximal* si en plus son graphe n'est pas strictement contenu dans celui d'un autre opérateur monotone. Deux sous-classes sont à distinguer ; les opérateurs *fortement monotones* et les opérateurs *co-coercifs*.

Définition A.1.2 (Opérateurs fortement monotones).

L'opérateur $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ est *fortement monotone de constante a* si

$$\forall y, y' \in \mathbb{R}^n, u \in T(y), u' \in T(y') \quad \langle y - y', u - u' \rangle \geq a \|y - y'\|^2$$

Notons que T est alors injectif. Si la monotonie généralise la définie-positivité pour les opérateurs linéaires, la forte monotonie quant à elle correspond à la stricte définie-positivité, le rôle de a étant joué alors par la plus petite valeur propre. La *co-coercivité* constitue la propriété duale.

Définition A.1.3 (Opérateur co-coercifs).

Un opérateur $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est *co-coercif de module a* (ou a la propriété de Dunn avec la constante a^{-1}) si :

$$\forall y, y' \in \mathbb{R}^n \quad \langle y - y', T(y) - T(y') \rangle \geq a \|T(y) - T(y')\|^2$$

Un opérateur est co-coercif de module a si et seulement si son inverse est fortement monotone de module a . Une application linéaire symétrique semi-définie positive est toujours co-coercive.

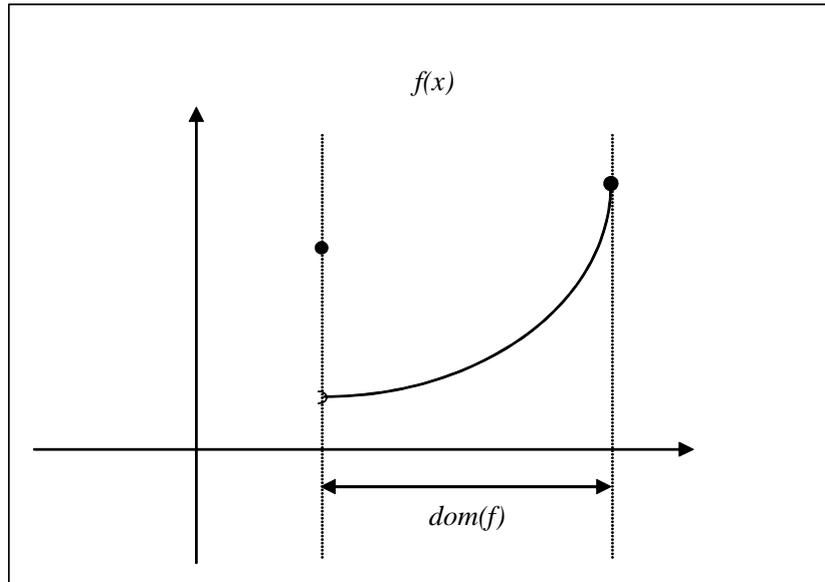


FIGURE A.1.1 – Fonction convexe qui n’est pas semi-continue inférieurement (au bord gauche) et qui n’est pas sous-différentiable (en son bord droit)

A.1.3 Opérateur sous-différentiel

Le premier opérateur auquel nous nous intéressons est l’opérateur *sous-différentiel* associé à une fonction $f : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$, noté ∂f . Il est défini pour tout $\bar{x} \in \mathbb{R}^n$ par :

$$\partial f(\bar{x}) = \left\{ v \in \mathbb{R}^n, \bar{x} \in \arg \max_x \langle v, x \rangle - f(x) \right\} \quad (\text{A.2})$$

Si f est une fonction convexe, alors ∂f est un opérateur monotone. Il est maximal si de plus f est s.c.i ([92, Corollary 31.5.2]). $\partial f(\bar{x})$ peut être vide *i.e.* $\bar{x} \notin \text{dom}(\partial f)$ même si $\bar{x} \in \text{dom}(f)$. Dans le cas contraire, c’est à dire si $\partial f(\bar{x}) \neq \emptyset$ alors f est *sous-différentiable* en \bar{x} et les éléments $\bar{v} \in \partial f(\bar{x})$ sont appelés les *sous-gradients* de f en \bar{x} . La fonction de la figure A.1.1 n’est par exemple pas sous-différentiable au bord droit de son domaine où “la pente devient infinie”. Ce cas de figure n’intervient pas pour les fonctions pseudo-polyédrales qui vérifient $\text{dom}(\partial f) = \text{dom}(f)$. Nous dirons par ailleurs que f est *différentiable* en \bar{x} si $\partial f(\bar{x})$ est réduit à un singleton.

A.1.4 Cône normal

Les opérateurs *cônes normaux* constituent une sous-famille remarquable d’opérateurs sous-différentiels. Il s’agit de ceux associés à une fonction indicatrice $\delta_{\mathcal{C}}$ d’un ensemble \mathcal{C} . Le sous-différentiel $\partial[\delta_{\mathcal{C}}]$ noté alors $\mathbf{N}_{\mathcal{C}}$ prend la forme :

$$\mathbf{N}_{\mathcal{C}}(\bar{x}) = \{ v \in \mathbb{R}^n / \forall x \in \mathcal{C} \langle x - \bar{x}, v \rangle \leq 0 \}$$

Il est monotone si \mathcal{C} est convexe et maximal si \mathcal{C} est de plus fermé. En un mot, $\mathbf{N}_{\mathcal{C}}(\bar{x})$ désigne l’ensemble des directions qui forment un angle négatif avec toutes les directions permettant de se déplacer dans \mathcal{C} depuis \bar{x} . $\mathbf{N}_{\mathcal{C}}(\bar{x})$ est toujours un cône, vide si \bar{x} n’appartient pas à \mathcal{C} et réduit au vecteur nul si \bar{x} est dans l’intérieur de \mathcal{C} . Lorsque \mathcal{C} est un

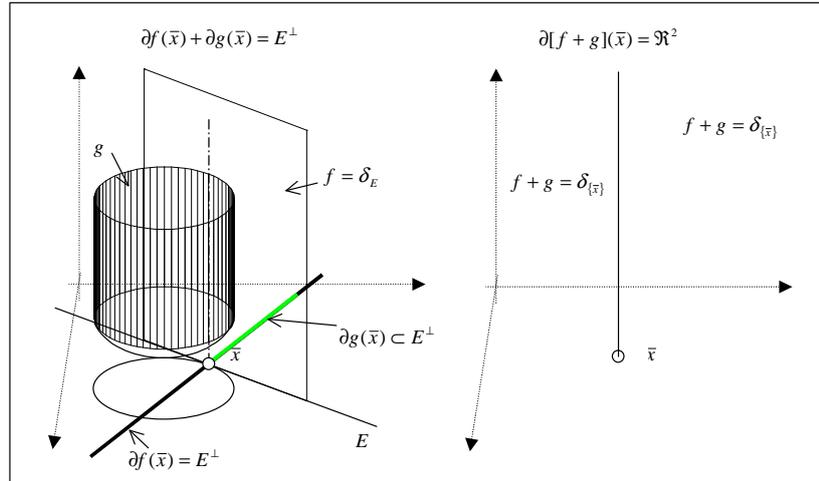


FIGURE A.1.2 – Exemple de non distributivité de la sous-différentiation.

sous-espace vectoriel $\mathcal{C} = \mathcal{A}$, alors il prend la forme spéciale :

$$N_{\mathcal{A}}(y) = \begin{cases} \mathcal{A}^{\perp} & \text{si } y \in \mathcal{A}; \\ \emptyset & \text{sinon.} \end{cases}$$

A.1.5 Caractérisation des solutions de (A.1)

L'opérateur sous-différentiel permet entre autre de caractériser les solutions du problème (A.1) via la proposition A.1.1 suivante.

Proposition A.1.1. x^* est une solution optimale du problème (A.1) si et seulement si :

$$0 \in \partial f(x^*) \quad (\text{A.3})$$

Démonstration. La définition A.2 du sous-différentiel donne :

$$x^* \in \arg \max_{x \in \mathbb{R}^n} \langle 0, x \rangle - f(x) = \arg \min_{x \in \mathbb{R}^n} f(x)$$

□

Ainsi, le problème d'optimisation (A.1) est équivalent au problème d'inclusion :

$$\text{Trouver } x^* \text{ tel que } 0 \in \partial f(x^*) \quad (\text{A.4})$$

A.1.6 Distributivité de la sous-différentiation

La fonction objectif intervenant dans le problème décomposable 1.2 est la somme de deux fonctions F et C . Pouvons nous ajouter ∂F et ∂C pour obtenir $\partial[F + C]$? Comme le montre le contre-exemple de la figure A.1.2, la réponse est non en général. L'opération de sous-différentiation n'est pas distributive; seule l'inclusion $\partial F + \partial C \subset \partial(F + C)$ est toujours vraie. L'inclusion réciproque peut tout de même parfois être assurée; par exemple sous les conditions de la proposition suivante ([33, proposition 3.23]).

Proposition A.1.2. Soit $f, g : \mathbb{R}^n \rightarrow \mathbb{R}$ convexes, s.c.i. Si l'une des conditions suivantes est vérifiée

- a) $\text{rint}(\text{dom}(f)) \cap \text{rint}(\text{dom}(g)) \neq \emptyset$
 - b) $\text{dom}(f) \cap \text{rint}(\text{dom}(g)) \neq \emptyset$ et f est pseudo-polyédrale
 - c) $\text{dom}(f) \cap \text{dom}(g) \neq \emptyset$ et f et g sont pseudo-polyédrales
- alors $\partial f + \partial g = \partial(f + g)$.

A.2 Problèmes de minimisation paramétrique

Très souvent, le problème (A.1) prend la forme :

$$\underset{x \in \mathbb{R}^n}{\text{Minimiser}} f(x) = \varphi(x, \bar{y}) \quad (P_{\bar{y}})$$

où la fonction

$$\begin{aligned} \varphi : \mathbb{R}^n \times \mathbb{R}^m &\rightarrow \overline{\mathbb{R}} \\ (x, y) &\mapsto \varphi(x, y) \end{aligned}$$

est convexe s.c.i (sous-entendu conjointement en x et en y). Le calcul du coût de fonctionnement d'une centrale en fonction d'un planning de production \bar{y} fixé (1.6) nécessite par exemple la résolution d'un tel problème d'optimisation paramétrique. Les techniques se basant sur la dualité utilisent la structure de φ plutôt que celle de f pour résoudre ce problème. Cela permet ainsi :

- d'une part d'évaluer la sensibilité du modèle (en terme de coût optimal) relativement à des variations du paramètre y ,
- d'autre part de mettre en œuvre des algorithmes pour des problèmes d'optimisation sous contraintes du type :

$$\underset{x}{\text{Minimiser}} f_0(x) \quad (\text{A.5a})$$

$$\text{s.c. } g_i(x) \leq 0 \quad i = 1, \dots, p \quad (\text{A.5b})$$

$$A_j(x) = 0 \quad j = 1, \dots, q \quad (\text{A.5c})$$

où les fonctions f_0 et g_i sont convexes s.c.i et les A_j sont affines. La sensibilité du coût optimal pour y variant autour de \bar{y} est établie grâce à la fonction valeur $F : \mathbb{R}^m \rightarrow \overline{\mathbb{R}}$ définie implicitement pour tout $y \in \mathbb{R}^m$ par :

$$F(y) = \inf_{x \in \mathbb{R}^n} \varphi(x, y)$$

donnant le coût optimal à y fixé. Si F est sous-différentiable en \bar{y} , l'ensemble $\partial F(\bar{y})$ fournit alors une sous-estimation de l'augmentation du coût optimal pour des variations marginales de y autour de \bar{y} :

$$\forall u \in \partial F(\bar{y}), \forall \epsilon \in \mathbb{R}^m \quad F(\bar{y} + \epsilon) \geq F(\bar{y}) + \langle u, \epsilon \rangle$$

F est toujours convexe mais pas nécessairement propre ni s.c.i même si φ l'est (voir [92, page 72] pour un contre-exemple). L'hypothèse A.2.1 suivante constitue une condition suffisante à ces deux caractéristiques.

Hypothèse A.2.1. Pour tout $y \in \mathbb{R}^m$, la fonction $x \mapsto \varphi(x, y)$ est coercive i.e. $\lim_{\|x\| \rightarrow \infty} \varphi(x, y) = +\infty$.

Proposition A.2.1. Soit $\varphi : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \overline{\mathbb{R}}$ convexe propre s.c.i et satisfaisant l'hypothèse A.2.1, alors :

- i) F est convexe propre et s.c.i
- ii) La relation suivante entre les graphes de ∂F et de $\partial\varphi$ existe.

$$\forall y, u \in \mathbb{R}^m \quad u \in \partial F(y) \iff \exists x \in \mathbb{R}^n / (0, u) \in \partial\varphi(x, y) \quad (\text{A.6})$$

Démonstration. *i* viens de [92, Theorem 9.2] et *ii* est obtenu en écrivant les définitions des sous-différentiels et en utilisant la définition de F . \square

Les conditions d'optimalité du problème (A.1) mis sous la forme ($P_{\bar{y}}$) sont données par la proposition suivante :

Proposition A.2.2. Si l'une des deux conditions suivantes est satisfaite

- a) φ est pseudo-polyédrale.
- b) Il existe $x \in \mathbb{R}^n$ tel que $(x, \bar{y}) \in \text{rint}(\text{dom}(\varphi))$

alors,

- i) x^* est une solution optimale du problème (A.1) si et seulement s'il existe $u^* \in \mathbb{R}^m$ tel que $(0, u^*) \in \partial\varphi(x^*, \bar{y})$.

Si un tel couple (x^*, u^*) existe, alors :

- ii) $u^* \in \partial F(\bar{y})$

Démonstration. Considérons le sous-espace affine de $\mathbb{R}^n \times \mathbb{R}^m$:

$$E = \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^m / y = \bar{y}\}$$

Minimiser f équivaut à minimiser $\varphi + \delta_E$. Sous a) ou b) nous avons $\partial[\varphi + \delta_E] = \partial\varphi + \partial\delta_E$ d'après la proposition A.1.2. Ainsi, $(x^*, y^*) \in \mathbb{R}^n \times \mathbb{R}^m$ est solution d'après la proposition A.1.1 si et seulement si :

$$\begin{aligned} (0, 0) &\in \partial\varphi(x^*, y^*) + \partial\delta_E(x^*, y^*) \\ (0, 0) &\in \partial\varphi(x^*, y^*) + E^\perp \text{ et } (x^*, y^*) \in E \end{aligned}$$

c'est à dire s'il existe $(v^*, u^*) \in \partial\varphi(x^*, y^*)$ tel que $-(v^*, u^*) \in E^\perp$ et $(x^*, y^*) \in E$, ou plus simplement s'il existe $u^* \in \mathbb{R}^m$ tel que $(0, u^*) \in \partial\varphi(x^*, \bar{y})$. Enfin, si x^* est une solution optimale et $(0, u^*) \in \partial\varphi(x^*, \bar{y})$, alors pour tout x, y :

$$\varphi(x, y) \geq \varphi(x^*, \bar{y}) + \langle u^*, y \rangle$$

$\varphi(x^*, \bar{y})$ est par définition égal à $F(\bar{y})$. La conclusion vient alors en remarquant que $x \mapsto \varphi(x, y)$ est bornée inférieurement et en faisant tendre $\varphi(x, y)$ vers $F(y)$ qui existe. \square

A.3 Dualité

Les techniques de dualité lagrangienne reposent sur l'opération de conjugaison. Étant donnée une fonction convexe $f : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$, sa conjuguée notée $f^* : \mathbb{R}^n \rightarrow \overline{\mathbb{R}}$ est définie pour tout $v \in \mathbb{R}^n$ comme le maximum (éventuellement $+\infty$) dans (A.2) :

$$f^*(v) = \max_{x \in \mathbb{R}^n} \langle v, x \rangle - f(x) \quad (\text{A.7})$$

Pour les fonctions s.c.i, l'opération de conjugaison est involutive, c'est à dire que $(f^*)^* = f$ et inverse les opérateurs sous-différentiels, c'est à dire $\partial f^* = [\partial f]^{-1}$. Par analogie au sous-différentiel d'une fonction convexe, une fonction concave h possède un opérateur sur-différentiel $\partial h : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ défini comme le sous-différentiel de son opposé $\partial h = \partial[-h] = \{x \in \mathbb{R}^n / \bar{v} \in \arg \max_v -h(v) - \langle x, v \rangle\}$.

Le *lagrangien* associé au problème $(P_{\bar{y}})$ est la fonction définie par *conjugaison partielle*, c'est à dire uniquement par rapport au paramètre y :

$$(x, u) \mapsto L(x, u) = \inf_{y \in \mathbb{R}^m} \varphi(x, y) - \langle u, y \rangle$$

qui est convexe en x et concave en u . A titre d'exemple, la fonction :

$$\varphi(x, y) = \begin{cases} f_0(x) & \text{si } f_1(x) \leq y_1 \text{ et } f_2(x) = y_2 \\ +\infty & \text{sinon.} \end{cases} \quad (\text{A.8})$$

correspondant à une perturbation du second membre dans le problème (A.5), donne lieu au lagrangien classique :

$$L(x, u) = \begin{cases} f_0(x) - \langle u_1, f_1(x) \rangle - \langle u_2, f_2(x) \rangle & \text{si } u_1 \leq 0 \\ -\infty & \text{sinon.} \end{cases}$$

Sous-différentiel du lagrangien

Pour tout x et tout u les fonctions $L_x : u \mapsto L(x, u)$ et $L_u : x \mapsto L(x, u)$ sont respectivement convexe et concave. Le sous-différentiel de L est alors défini comme l'opérateur multivoque :

$$\begin{aligned} \partial L : \mathbb{R}^n \times \mathbb{R}^m &\rightrightarrows \mathbb{R}^n \times \mathbb{R}^m \\ (x, u) &\mapsto \partial L(x, u) = \partial L_u(x) \times \partial L_x(u) \end{aligned}$$

La conjugaison partielle a l'effet d'une inversion partielle sur le sous-différentiel $\partial\varphi(\cdot, \cdot)$, c'est à dire une inversion uniquement par rapport à la deuxième composante ([92, Theorem 37.5]) :

$$(v, u) \in \partial\varphi(x, y) \iff (v, y) \in \partial L(x, u) \quad (\text{A.9})$$

ce qui permet d'obtenir la caractérisation suivante des solutions de $(P_{\bar{y}})$.

Proposition A.3.1. *Si l'une des deux conditions suivantes est satisfaite*

- a) φ est pseudo-polyédrale.
- b) Il existe $x \in \mathbb{R}^n$ tel que $(x, \bar{y}) \in \text{rint}(\text{dom}(\varphi))$

alors,

i) x^* est une solution optimale du problème $(P_{\bar{y}})$ si et seulement s'il existe $u^* \in \mathbb{R}^m$ tel que :

$$(0, 0) \in \partial L(x^*, u^*) - (0, \bar{y}) \quad (\text{A.10})$$

Démonstration. Il s'agit de la proposition A.2.2 réécrite à l'aide de l'inversion partielle (A.9). \square

Le problème primal $(P_{\bar{y}})$ est alors équivalent au problème d'inclusion :

$$\text{Trouver } (x^*, u^*) \text{ tel que } (0, 0) \in \partial L(x^*, u^*) - (0, \bar{y}) \quad (\text{A.11})$$

Si φ est convexe s.c.i alors ∂L est un opérateur monotone maximal [92, Corollary 37.5.2] et $\partial L - (0, \bar{y})$ l'est aussi par translation.

Problème dual

La condition d'optimalité (A.10) fait apparaître une symétrie dans les rôles joués par x et u . Un *problème dual* est habituellement ainsi défini de manière artificielle en intervertissant les rôles de x et u . Associons pour ce faire au problème $(P_{\bar{y}})$ la fonction duale :

$$u \mapsto H(u) = \varphi^*(0, u) \quad (\text{A.12})$$

La première variable de φ^* tient lieu de paramètre susceptible d'être perturbé autour de sa valeur 0, u devient la variable de décision et la variable d'origine x joue maintenant le rôle de variable duale associée à la perturbation. H est convexe s.c.i et ∂H est relié à ∂L par la relation :

$$\forall y, u \in \mathbb{R}^m \quad y \in \partial H(u) \iff \exists x \in \mathbb{R}^n / (0, y) \in \partial L(x, u)$$

Le problème dual consiste alors à trouver u^* tel que $\bar{y} \in \partial H(u^*)$, c'est à dire :

$$\text{Trouver } u^* \text{ tel que } 0 \in \partial H(u^*) - \bar{y} \quad (\text{A.13})$$

ou encore :

$$\underset{u \in \mathbb{R}^m}{\text{Minimiser}} H(u) - \langle \bar{y}, u \rangle \quad (D_{\bar{y}})$$

Proposition A.3.2. *Si l'une des deux conditions suivantes est satisfaite*

a) φ est pseudo-polyédrale.

b) Il existe $x \in \mathbb{R}^n$ tel que $(x, \bar{y}) \in \text{rint}(\text{dom}(\varphi))$

alors, u^* est une solution optimale du problème dual $(D_{\bar{y}})$ si et seulement s'il existe $x^* \in \mathbb{R}^n$ tel que :

$$(0, 0) \in \partial L(x^*, u^*) - (0, \bar{y})$$

auquel cas x^* est solution du problème primal $(P_{\bar{y}})$.

Les problèmes d'inclusion (A.4), (A.11) et (A.13) peuvent être étudiés sous une même forme abstraite :

$$\text{Trouver } y^* \text{ tel que } 0 \in T(y^*) \quad (\text{A.14})$$

où $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ est monotone maximal et joue le rôle au choix de ∂f , ∂L ou ∂H . Un tel élément y^* est appelé un *zéro* de l'opérateur T .

Annexe B

Algorithmes de points fixes

Cette annexe est consacrée aux résultats de convergence d'algorithmes de point fixe. Plus précisément, si $N : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est un opérateur et $fix(N)$ l'ensemble de ses points fixes (potentiellement vide) :

$$fix(N) \stackrel{def}{=} \{s \in \mathbb{R}^n / N(s) = s\}$$

nous nous intéressons à l'algorithme générique [B.3.1](#) et à sa convergence éventuelle vers un élément de $fix(N)$.

B.1 Opérateurs contractants, faiblement contractants et moyennés

Définition B.1.1 (Opérateurs contractants et faiblement contractants).

Un opérateur $N : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est contractant s'il existe une constante $a < 1$ telle que pour tout $s, s' \in \mathbb{R}^n$:

$$\|N(s) - N(s')\| \leq a \|s - s'\|$$

et faiblement contractant (ou non-expansif) si $a = 1$.

En particulier via l'inégalité de Cauchy-Schwarz, les opérateurs co-coercifs de module a sont contractants de rapport a^{-1} si $a > 1$ et faiblement contractants si $a = 1$. Une combinaison convexe $\alpha N_1 + (1 - \alpha)N_2$ d'opérateurs faiblement contractants est faiblement contractante et si la combinaison est stricte (*i.e.* $0 < \alpha < 1$) alors les points fixes de la combinaison sont donnés par la proposition [B.1.1](#) suivante.

Proposition B.1.1. Soit $N_1, N_2 : \mathbb{R}^n \rightarrow \mathbb{R}^n$ faiblement contractants et $0 < \alpha < 1$, alors :

$$fix(\alpha N_1 + (1 - \alpha)N_2) = fix(N_1) \cap fix(N_2)$$

Dans ce cas, si l'un des deux opérateurs est contractant, alors la proposition [B.1.2](#) établit que la combinaison l'est aussi.

Proposition B.1.2. Si $J : \mathbb{R}^n \rightarrow \mathbb{R}^n$ s'écrit $J = \alpha N_1 + (1 - \alpha)N_2$ pour $0 < \alpha \leq 1$ avec N_1 contractant de rapport a et N_2 faiblement contractant, alors J est lui-même contractant de rapport $1 - \alpha(1 - a) < 1$.

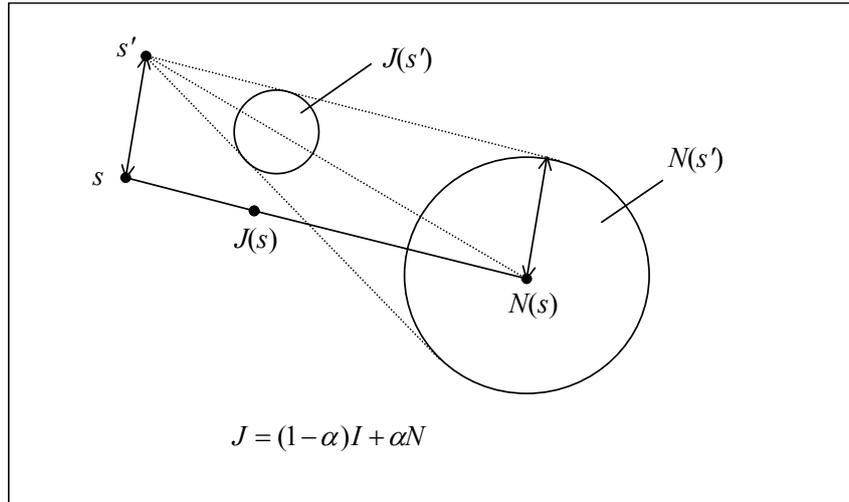


FIGURE B.1.1 – Propriété de contraction d'un opérateur moyenné

Démonstration. L'inégalité triangulaire donne :

$$\begin{aligned} \|J(s) - J(s')\| &\leq \alpha \|N_1(s) - N_1(s')\| + (1 - \alpha) \|N_2(s) - N_2(s')\| \\ &\leq (\alpha a + 1 - \alpha) \|s - s'\| \end{aligned}$$

□

On parle d'*opérateur moyenné* si l'un des deux opérateurs est l'identité et que la combinaison est stricte.

Définition B.1.2 (Opérateurs moyennés).

Un opérateur $J : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est moyenné ou α -moyenné s'il s'écrit sous la forme :

$$J = (1 - \alpha)I_n + \alpha N$$

avec N un opérateur faiblement contractant et $0 < \alpha < 1$.

Puisque $\text{fix}(I_n) = \mathbb{R}^n$, l'opération de "moyennisation" conserve l'ensemble des points fixes d'après la proposition B.1.1. Les opérateurs moyennés possèdent la propriété de contraction exprimée par le lemme B.1.1 ([21, Lemma 2.1 -ii]). Elle est traduite graphiquement sur la figure B.1.1. Etant donné un point s et son image $J(s)$ par un opérateur moyenné. La boule de droite sur la figure B.1.1 représente la zone où se situe $N(s')$, et celle de gauche celle où se situe $J(s')$. Cette propriété peut aussi s'exprimer à l'aide de l'opérateur N via le corollaire B.1.1 qui suit.

Lemme B.1.1. $J : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est α -moyenné si et seulement si :

$$\|Js - Js'\|^2 \leq \|s - s'\|^2 - \frac{1 - \alpha}{\alpha} \|(I - J)s - (I - J)s'\|^2 \quad (\text{B.1})$$

Démonstration. Par hypothèse, $J = (1 - \alpha)I + \alpha N$ avec $N = (1 - \alpha^{-1})I + \alpha^{-1}J$. Le

développement de $\|Ns - Ns'\|^2$ donne :

$$\begin{aligned}
 \|Ns - Ns'\|^2 &= (1 - 2\alpha^{-1} + \alpha^{-2}) \|s - s'\|^2 + \alpha^{-2} \|Js - Js'\|^2 \\
 &\quad + \alpha^{-1}(1 - \alpha^{-1}) \langle (I - J)s - (I - J)s', s - s' \rangle \\
 &= (1 - \alpha^{-1}) \|s - s'\|^2 + \alpha^{-1} \|Js - Js'\|^2 \\
 &\quad + (-\alpha^{-1} + \alpha^{-2}) \|s - s'\|^2 + (\alpha^{-2} - \alpha^{-1}) \|Js - Js'\|^2 \\
 &\quad + \alpha^{-1}(1 - \alpha^{-1}) \langle (I - J)s - (I - J)s', s - s' \rangle \\
 \|Ns - Ns'\|^2 - \|s - s'\|^2 &= \alpha^{-1} \|s - s'\|^2 + \alpha^{-1} \|Js - Js'\|^2 \\
 &\quad + \alpha^{-1}(\alpha^{-1} - 1) \|(I - J)s - (I - J)s'\|^2
 \end{aligned} \tag{B.2}$$

qui, conjointement à la non-expansivité de N donne le résultat. \square

Corollaire B.1.1. *Si $J : \mathbb{R}^n \rightarrow \mathbb{R}^n$ s'écrit $J = (1 - \alpha)I + \alpha N$ avec N faiblement contractant avec $0 \leq \alpha \leq 1$ (nous ne parlons pas d'opérateur moyenné puisque $\alpha = 1$ est exclu de la définition), alors :*

$$\|Js - Js'\|^2 \leq \|s - s'\|^2 - \alpha(1 - \alpha) \|(I - N)s - (I - N)s'\|^2 \tag{B.3}$$

Démonstration. Le résultat est trivialement vrai pour $\alpha = 0$ et $\alpha = 1$ puisqu'il s'agit de la faible contraction des deux cas extrêmes $J = I$ et $J = N$. Dans les autres cas, puisque $\alpha > 0$, nous pouvons utiliser la formule du lemme B.1.1 dans lequel l'égalité $I - J = I - (1 - \alpha)I - \alpha N = \alpha(I - N)$ donne le résultat. \square

En plus d'être convexe, l'ensemble des opérateurs moyennés est stable par composition :

Proposition B.1.3. *Si $J_1, J_2 : \mathbb{R}^n \rightarrow \mathbb{R}^n$ sont respectivement moyennés avec les constantes α_1 et α_2 alors $J_1 \circ J_2$ est aussi moyenné (avec la constante $1 - (1 - \alpha_1)(1 - \alpha_2)$).*

Nous traitons maintenant à l'écart les opérateurs moyennés affines, qui de par leur structure, peuvent être assimilés à des opérateurs contractants (voir la proposition B.1.4 qui suit).

B.1.1 Cas des opérateurs affines

Lemme B.1.2. *Un opérateur moyenné $J : \mathbb{R}^n \rightarrow \mathbb{R}^n$ affine s'écrit dans une certaine base :*

$$J : s \mapsto J(s) = \begin{pmatrix} I_{n_1} & 0 \\ 0 & A \end{pmatrix} s + j$$

avec $n_1 \leq n$, A une matrice carrée de dimension $n - n_1$ de norme $\|A\| < 1$. De plus, si $\text{fix}(J) \neq \emptyset$ alors $j = (0, a^\top)^\top$ où a est de taille $n - n_1$.

Autrement dit, l'espace se décompose en un produit cartésien ; dans le premier espace, J effectue une simple translation tandis qu'il se comporte comme une contraction dans le deuxième.

Démonstration. $J = (1 - \alpha)I + \alpha N$ avec N affine et faiblement contractant :

$$N : s \mapsto N(s) = Gs + n$$

Sa faible contraction donne $\|G\| \leq 1$. Notons $H = (1 - \alpha)I + \alpha G$ la partie linéaire de J . Montrons tout d'abord l'équivalence suivante.

$$\|Hs\| = \|s\| \iff Hs = s \quad (\text{B.4})$$

Le sens \Leftarrow est évident. Soit maintenant $s \in \mathbb{R}^n$.

$$\begin{aligned} \|Hs\|^2 &= \|(1 - \alpha)s + \alpha Gs\|^2 \\ &= (1 - \alpha)^2 \|s\|^2 + \alpha^2 \|Gs\|^2 + (1 - \alpha)\alpha \langle Gs, s \rangle \end{aligned}$$

Or, $\|Gs\|^2 \leq \|s\|^2$ et $\langle Gs, s \rangle \leq \|s\|^2$ donc $\|Hs\|^2 = \|s\|^2$ implique les deux égalités $\|Gs\|^2 = \|s\|^2$ et $\langle Gs, s \rangle = \|s\|^2$. Cette dernière égalité, avec le fait que :

$$\begin{aligned} \|Gs - s\|^2 &= \|s\|^2 + \|Gs\|^2 - 2 \langle Gs, s \rangle \\ &\leq 2(\|s\|^2 - \langle Gs, s \rangle) \end{aligned}$$

donne $Gs = s$ et par conséquent $Hs = s$ prouvant l'équivalence. En notant $E \subset \mathbb{R}^n$ le sous espace $E = \text{fix}(G) = \text{fix}(H)$, H prend donc la forme dans une base adaptée au produit $E \times E^\perp$:

$$H = \begin{pmatrix} I_{n_1} & B \\ 0 & A \end{pmatrix}$$

Montrons maintenant que la sous-matrice B est nulle. Pour tout $s = (s_1^\top, s_2^\top)^\top$, $s_1 \in \mathbb{R}^{n_1}$, $s_2 \in \mathbb{R}^{n-n_1}$, nous avons puisque $\|H\| \leq 1$:

$$\begin{aligned} \|Hs\|^2 &\leq \|s\|^2 \\ \|s_1 + Bs_2\|^2 + \|As_2\|^2 &\leq \|s_1\|^2 + \|s_2\|^2 \\ \|Bs_2\|^2 + 2 \langle s_1, Bs_2 \rangle + \|As_2\|^2 &\leq \|s_2\|^2 \end{aligned}$$

Procédons par l'absurde en supposant que B n'est pas nulle. Nous pouvons alors choisir s_2 tel que $b = Bs_2 \neq 0$. En choisissant alors $s_1 = tb$ avec $t > 0$, le produit scalaire du membre de gauche vérifie :

$$\langle s_1, Bs_2 \rangle = \langle tb, b \rangle = t \|b\|^2$$

et en faisant croître t arbitrairement, nous obtenons la contradiction voulue, d'où $B = 0$. Le fait que $\|A\| < 1$ vient enfin de l'équivalence B.4. Pour tout $s_2 \neq 0$, $(0, s_2^\top)^\top \notin E$ donc $\|H(0, s_2^\top)^\top\| \neq \|(0, s_2^\top)^\top\|$. Mais $\|H\| \leq 1$ donc $\|H(0, s_2^\top)^\top\| < \|(0, s_2^\top)^\top\|$ c'est à dire $\|As_2\| < \|s_2\|$. En ce qui concerne la dernière partie du lemme, si $\bar{s} = (\bar{s}_1^\top, \bar{s}_2^\top)^\top \in \text{fix}(J)$ et $j = (b^\top, a^\top)^\top$ alors $J(s) = ((\bar{s}_1 + b)^\top, (A\bar{s}_2 + a)^\top)^\top = (\bar{s}_1^\top, \bar{s}_2^\top)^\top$ implique $b = 0$. \square

Proposition B.1.4. *Soit $J : \mathbb{R}^n \rightarrow \mathbb{R}^n$ un opérateur moyenné affine tel que $\text{fix}(J) \neq \emptyset$. Alors, il existe κ , $0 \leq \kappa < 1$ tel que pour tout $s \in \mathbb{R}^n$:*

$$\text{dist}(J(s), \text{fix}(J)) \leq \kappa \text{dist}(s, \text{fix}(J))$$

Démonstration. Soit $s = (s_1^\top, s_2^\top)^\top \in \mathbb{R}^n$ et $\bar{s} = (\bar{s}_1^\top, \bar{s}_2^\top)^\top \in \text{fix}(J)$ la projection de s sur $\text{fix}(J)$. Alors le vecteur $\tilde{s} = (s_1^\top, \bar{s}_2^\top)^\top$ est aussi un point fixe de J et $\|s - \tilde{s}\| \leq \|s - \bar{s}\|$ donc $\tilde{s} = \bar{s}$ et $s_1 = \bar{s}_1$. Nous avons alors $\text{dist}(s, \text{fix}(J)) = \|s_2 - \bar{s}_2\|$.

$$\begin{aligned} \|J(s) - J(\bar{s})\| &= \|As_2 + a - A\bar{s}_2 - a\| \\ \|J(s) - \bar{s}\| &\leq \|A\| \|s_2 - \bar{s}_2\| \\ &\leq \|A\| \text{dist}(s, \text{fix}(J)) \end{aligned}$$

Enfin, puisque $\text{dist}(J(s), \text{fix}(J)) \leq \|J(s) - \bar{s}\|$, nous obtenons notre inégalité avec $\kappa = \|A\|$. \square

Revenons maintenant dans le cadre non linéaire. Les opérateurs moyennés de constante $\alpha = \frac{1}{2}$ forment une classe à part du fait de ses liens étroits avec les opérateurs monotones. Il s'agit des opérateurs pseudo-contractants.

B.2 Opérateurs pseudo-contractants

Définition B.2.1. *Un opérateur $P : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est pseudo-contractant (ou fermement non-expansif) s'il est $\frac{1}{2}$ -moyenné.*

Il est habituel de rencontrer la caractérisation suivante en guise de définition.

Proposition B.2.1. *$P : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est pseudo-contractant si et seulement s'il est co-coercif de constante 1 :*

$$\forall y, y' \in \mathbb{R}^n \quad \|P(y) - P(y')\|^2 \leq \langle y - y', P(y) - P(y') \rangle$$

Démonstration. Soit $P : \mathbb{R}^n \rightarrow \mathbb{R}^n$ un opérateur quelconque et $N = 2P - I$. Soit $s, s' \in \mathbb{R}^n$, $d = N(s)$ et $d' = N(s')$.

$$\begin{aligned} \|N(s) - N(s')\|^2 &= \|2P(s) - 2P(s') + s' - s\|^2 \\ &= \|2P(s) - 2P(s')\|^2 + \|s' - s\|^2 - 2 \langle 2P(s) - 2P(s'), s - s' \rangle \\ &= \|s - s'\|^2 + 4 \left(\|P(s) - P(s')\|^2 - \langle P(s) - P(s'), s - s' \rangle \right) \end{aligned}$$

Nous avons bel et bien $\|N(s) - N(s')\| \leq \|s - s'\|$ pour tout $s, s' \in \mathbb{R}^n$ si et seulement si $\|P(s) - P(s')\|^2 \leq \langle P(s) - P(s'), s - s' \rangle$ pour tout $s, s' \in \mathbb{R}^n$. \square

Un opérateur co-coercif devient pseudo-contractant par simple multiplication scalaire :

Lemme B.2.1. *Si $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est co-coercif de module a , alors aT est pseudo-contractant.*

Démonstration. Multiplions par a l'inégalité de la définition A.1.3.

$$\begin{aligned} a^2 \|T(y) - T(y')\|^2 &\leq a \langle T(y) - T(y'), y - y' \rangle \\ \|aT(y) - aT(y')\|^2 &\leq \langle aT(y) - aT(y'), y - y' \rangle \end{aligned}$$

\square

Nous passons maintenant en revue plusieurs conditions sous lesquelles l'algorithme B.3.1 converge.

Algorithme B.3.1 Itérations de point fixe avec sous-relaxation

1. $k = 0$; Choisir $s^0 \in \mathbb{R}^n$
 2. $s^{k+1} = (1 - \alpha^k)s^k + \alpha^k N(s^k)$
 3. $k \leftarrow k + 1$; Aller à l'étape 2
-

B.3 Théorèmes de convergence

Nous nous intéressons à la convergence de l'algorithme générique B.3.1.

La grande majorité des preuves de convergence présentes dans ce manuscrit reposent sur le lemme technique suivant.

Lemme B.3.1. Soit $\{d_k\}_k$, une suite de réels positifs et $\{(h_k^1, g_k^1)\}_k, \dots, \{(h_k^q, g_k^q)\}_k$ q suites de couples de réels positifs. Considérons les hypothèses suivantes.

- a) $d_{k+1} \leq d_k - h_k^1 g_k^1 - h_k^2 g_k^2 \dots - h_k^q g_k^q$ pour tout $k \geq 0$
- b) $\sum_k h_k^i = +\infty$
- c) $\liminf_k h_k^i > 0$

Nous avons les conséquences suivantes.

- i) $a \implies \{d_k\}_k$ converge.
- ii) a et $b \implies \liminf_{k \rightarrow \infty} g_k^i = 0$
- iii) a et $c \implies \lim_{k \rightarrow \infty} g_k^i = 0$

Démonstration. *i* vient du fait que la suite des d_k est décroissante et positive. Pour *ii*, nous procédons par l'absurde. Supposons que $\liminf_{k \rightarrow \infty} g_k^i \neq 0$, alors il existe $\bar{g} > 0$ et un rang $K \geq 0$ tel que $\bar{g} \leq g_k^i$ pour tout $k \geq K$. Par conséquent, d'après *a*, pour tout $l \geq K$, nous avons

$$\bar{g} \sum_{k=K}^{l-1} h_k^i \leq \sum_{k=K}^{l-1} h_k^i g_k^i \leq d^K$$

La série de gauche est donc bornée contraignant ainsi *b*. Finissons maintenant par *iii*. L'hypothèse *c* implique l'existence d'un $\underline{h} > 0$ et d'un rang K' à partir duquel $h_k^i \geq \underline{h}$. En sommant encore une fois l'inégalité de *a* pour k variant de K' à $l - 1$, nous obtenons :

$$\underline{h} \sum_{k=K'}^{l-1} g_k^i \leq \sum_{k=K'}^{l-1} h_k^i g_k^i \leq d^K$$

et cette fois, le caractère borné de la série de gauche implique *iii*. □

Lorsque N est faiblement contractant, l'opérateur effectuant une itération dans l'algorithme B.3.1 est moyenné et l'inégalité clé (B.1) assure alors essentiellement la convergence vers un point fixe de N . L'hypothèse principale sur les α_k sera la suivante.

Hypothèse B.3.1. $\{\alpha^k\}_k$ est une suite de réels telle que :

$$0 \leq \alpha^k \leq 1 \text{ pour tout } k$$

$$\sum_{k=0}^{+\infty} \alpha^k (1 - \alpha^k) = +\infty$$

La deuxième requête est notamment vérifiée si α^k ou $1 - \alpha^k$ converge en série divergente vers 0 ; si $\alpha^k \equiv \alpha \in]0, 1[$ est constant ou bien encore si les limites inférieures et supérieures de α^k sont dans $]0, 1[$.

Proposition B.3.1. *Soit $N : \mathbb{R}^n \rightarrow \mathbb{R}^n$ un opérateur faiblement contractant tel que $\text{dom}(N) = \mathbb{R}^n$ et possédant au moins un point fixe et $\{\alpha^k\}_k$ une suite vérifiant l'hypothèse B.3.1. Alors l'algorithme B.3.1 converge vers un point fixe de N .*

Démonstration. Soit \bar{s} un point fixe de N . Le corollaire B.1.1 appliqué à l'opérateur $J = (1 - \alpha_k)I + \alpha_k N$ avec $s = s^k$ et $s' = \bar{s}$ donne :

$$\|s^{k+1} - \bar{s}\|^2 \leq \|s^k - \bar{s}\|^2 - \alpha^k(1 - \alpha^k) \|N(s^k) - s^k\|^2 \quad (\text{B.5})$$

Les suites définies pour tout k par $d_k = \|s^k - \bar{s}\|^2$ et

$$(h_k, g_k) = (\alpha^k(1 - \alpha^k), \|N(s^k) - s^k\|^2)$$

vérifient les hypothèses *a* et *b* du lemme B.3.1 qui fournit alors :

$$\liminf_{k \rightarrow \infty} \|N(s^k) - s^k\| = 0$$

ainsi que la convergence de la suite $\{\|s^k - \bar{s}\|^2\}_k$ et donc le caractère borné de $\{s^k\}_k$. Il existe par conséquent une sous-suite extraite $\{s^{\varphi(k)}\}_k$ telle que :

$$\|N(s^{\varphi(k)}) - s^{\varphi(k)}\| \rightarrow 0$$

, bornée elle aussi et donc possédant un point d'accumulation s^∞ qui est un point fixe de N . Par définition ce point d'accumulation vérifie $\liminf_{k \rightarrow \infty} \|s^k - s^\infty\| = 0$. Or, d'après B.5 en remplaçant \bar{s} par s^∞ , la suite $\{\|s^k - s^\infty\|\}_k$ est décroissante entraînant aussi $\lim_{k \rightarrow \infty} \|s^k - s^\infty\| = 0$. \square

Si N est contractant, alors l'hypothèse B.3.1 peut être relâchée quelque peu en autorisant α^k à prendre la valeur 1 infiniment souvent.

Hypothèse B.3.2. $\{\alpha^k\}_k$ est une suite de réels telle que :

$$0 \leq \alpha^k \leq 1 \text{ pour tout } k$$

$$\sum_{k=0}^{+\infty} \alpha^k = +\infty$$

Proposition B.3.2. *Soit $N : \mathbb{R}^n \rightarrow \mathbb{R}^n$ un opérateur contractant de rapport a tel que $\text{dom}(N) = \mathbb{R}^n$ et \bar{s} son unique point fixe, $\{\alpha^k\}_k$ une suite vérifiant l'hypothèse B.3.2. Alors la suite $\{s^k\}_k$ générée par l'algorithme B.3.1 converge vers \bar{s} . Si de plus, $\liminf_{k \rightarrow \infty} \alpha_k = \underline{\alpha} > 0$ alors la convergence est linéaire de taux :*

$$\limsup_k \frac{\|s^{k+1} - \bar{s}\|}{\|s^k - \bar{s}\|} \leq 1 - \underline{\alpha}(1 - a) < 1$$

Démonstration. En suivant la proposition B.1.2, nous avons pour tout k :

$$\begin{aligned}\|s^{k+1} - \bar{s}\| &\leq (1 - \alpha^k(1 - a)) \|s^k - \bar{s}\| \\ \|s^{k+1} - \bar{s}\| &\leq \|s^k - \bar{s}\| - \alpha^k(1 - a) \|s^k - \bar{s}\|\end{aligned}$$

En posant alors $d_k = \|s^k - \bar{s}\|$, $(h_k, g_k) = (\alpha_k(1 - a), \|s^k - \bar{s}\|^2)$, les hypothèses a et b du lemme B.3.1 sont vérifiées donc $\liminf \|s^k - \bar{s}\| = 0$ et comme il s'agit d'une suite décroissante s^k converge en fait vers \bar{s} . Maintenant, si s^k ne converge pas en un nombre fini d'étapes, alors :

$$\frac{\|s^{k+1} - \bar{s}\|}{\|s^k - \bar{s}\|} \leq 1 - \alpha^k(1 - a)$$

et le passage à la limite supérieure donne le résultat. □

Annexe C

Convergence graphique et proto-différentiabilité

C.1 Différentiabilité

La différentiabilité pour les opérateurs multivoques est définie comme suit.

Définition C.1.1. $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ est différentiable en \bar{y} si $T(\bar{y})$ est réduit à un singleton $T(\bar{y}) = \{\bar{u}\}$ et s'il existe une application linéaire S telle que :

$$T(y) \subset \bar{u} + S(y - \bar{y}) + o(\|y - \bar{y}\|)B(0, 1)$$

Pour $T = \partial F$, cela correspond à l'existence d'un développement à l'ordre 2 pour F :

$$F(y) = F(\bar{y}) + \langle \bar{u}, y \rangle + \frac{1}{2} \langle y - \bar{y}, S(y - \bar{y}) \rangle + o(\|y - \bar{y}\|^2)$$

Il est à noter que F peut posséder un tel développement et ∂F être différentiable sans que F ne possède de matrice hessienne $\nabla^2 F$ au sens classique. C'est le cas par exemple si ∂F n'est pas continu sur un voisinage de \bar{y} (voir figure C.1.1). F possède alors des points de non différentiabilité dans tout voisinage de \bar{y} .

Dans [96], Rockafellar propose la notion de proto-différentiabilité afin de généraliser ce type de développements à l'ordre 2 pour les fonctions convexes. Sa définition fait intervenir la notion de convergence graphique d'opérateurs comme la convergence des graphes.

C.2 Convergence graphique d'opérateurs

Définition C.2.1. Soit $\{E^k\}_k$ une suite de sous-ensembles de \mathbb{R}^n .

- La limite inférieure de $\{E^k\}_k$ est l'ensemble noté :

$$\liminf E^k \stackrel{def}{=} \{e \in \mathbb{R}^n / \forall k \in \mathbb{N}, \exists e^k \in E^k, e^k \rightarrow_k e\}$$

- La limite supérieure de $\{E^k\}_k$ est l'ensemble noté :

$$\limsup E^k \stackrel{def}{=} \{e \in \mathbb{R}^n / \exists \nu : \mathbb{N} \rightarrow \mathbb{N} \text{ croissante}, \forall k \in \mathbb{N}, \exists e^k \in E^{\nu(k)}, e^k \rightarrow_k e\}$$

Ces deux ensembles existent toujours. S'ils sont égaux i.e. $\liminf E^k = \limsup E^k = E$, alors $\{E^k\}_k$ converge vers E ce que nous notons :

$$E^k \rightarrow E$$

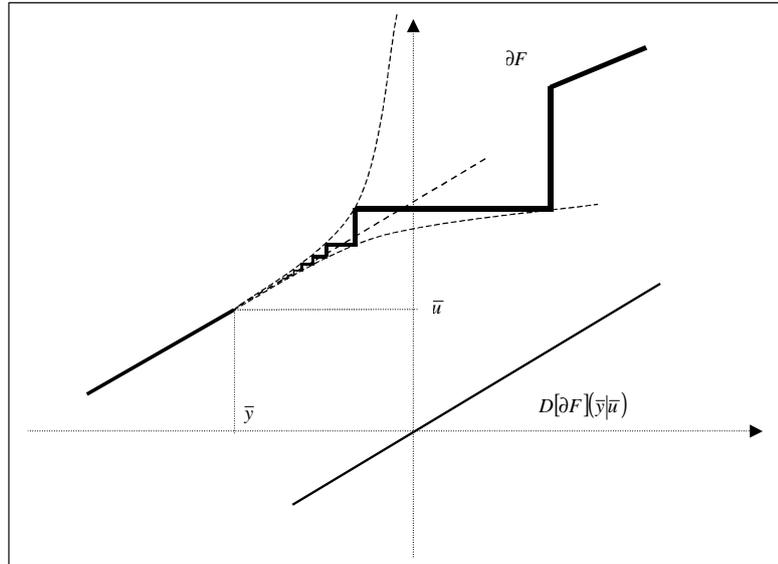


FIGURE C.1.1 – Opérateur monotone maximal différentiable multivoque dans tout voisinage de \bar{y} .

Cette notion permet de définir la convergence graphique d’une suite d’opérateurs multivoques.

Définition C.2.2. Soit $\{T^k\}_k$ une suite d’opérateurs, $T^k : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$. $\{T^k\}_k$ converge graphiquement vers T , noté :

$$T^k \xrightarrow{g} T$$

si la suite d’ensemble $\{gr(T^k)\}_k$ converge vers $gr(T)$.

C.3 Proto-différentiabilité

La proto-différentiabilité d’un opérateur T est définie comme la convergence graphique de la famille d’opérateurs indexée par $\tau > 0$:

$$\Delta_\tau(\bar{y}|\bar{u}) : h \mapsto \frac{T(\bar{y} + \tau h) - \bar{u}}{\tau} \quad (\text{C.1})$$

Le graphe de cet opérateur est obtenu en “zoomant” d’un facteur τ^{-1} sur le graphe de T . Intuitivement, si un “zoom” infini donne une image stable, alors cette image stabilisée est l’opérateur proto-différentiel. Plus rigoureusement :

Définition C.3.1 (Proto-différentiabilité). T est proto-différentiable en \bar{y} pour $\bar{u} \in T(\bar{y})$ si $\{\Delta_{\tau^k}(\bar{y}|\bar{u})\}_k$ converge graphiquement lorsque τ^k tend vers 0 par valeurs positives. L’opérateur limite noté $DT(\bar{y}|\bar{u})$ est l’opérateur proto-différentielle.

La figure C.3.1 présente un opérateur qui n’est pas proto-différentiable en \bar{y} . La limite supérieure représentée en haut à droite est différente de la limite inférieure dessinée juste en dessous. Pour fixer les idées, nous mentionnons le fait que T est différentiable s’il est proto-différentiable et que la proto-différentielle est une application linéaire (qui correspond alors à la différentielle).

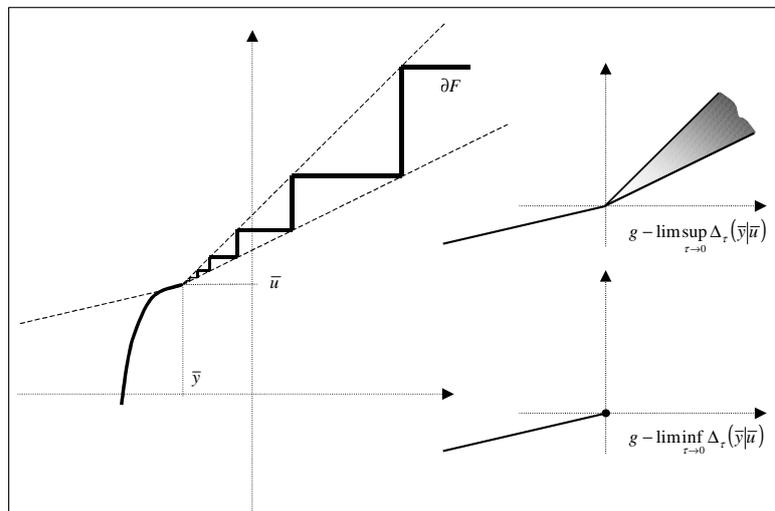


FIGURE C.3.1 – Opérateur qui n'est pas proto-différentiable en \bar{y}

Bibliographie

- [1] I. Babuska and J.M. Melenk, *The partition of unity method*, International Journal for Numerical Methods in Engineering **40** (1998), 727–758.
- [2] Kengy Barty, *Contributions à la discrétisation des contraintes de mesurabilité pour les problèmes d'optimisation stochastique.*, Ph.D. thesis, ENPC, 2004.
- [3] Heinz H. Bauschke and Jonathan M. Borwein, *On projection algorithms for solving convex feasibility problems*, SIAM Review **38** (1996), no. 3, 367–426.
- [4] Heinz H. Bauschke, Patrick Louis Combettes, and D. Russel Luke, *Phase retrieval, error reduction algorithm, and fienu variants : a view from convex optimization*, J. Opt. Soc. Am. A **19** (2002), no. 7.
- [5] A. Benahmed, *A convergence result for asynchronous algorithms and applications*, ArXiv Mathematics e-prints (2007).
- [6] Dimitri P. Bertsekas, *Constrained optimization and lagrange multiplier methods*, Academic Press, New York, 1982.
- [7] ———, *Distributed asynchronous computation of fixed points*, Tech. report, Laboratory for information and decision systems, MIT, Cambridge, 1982.
- [8] Dimitri P. Bertsekas and John N. Tsitsiklis, *Some aspects of parallel and distributed iterative algorithms - a survey*, Automatica **27** (1991), no. 1, 3–21.
- [9] J. Frédéric Bonnans, Mounir Haddou, Abdel Lissier, and Raja Rébaï, *Proximal decomposition method for solving global survivability in telecommunication networks*, Tech. report, INRIA, Novembre 2000.
- [10] H. Brezis, *Opérateurs maximaux montones et semi-groupes de contractions dans les espaces de Hilbert*, North Holland, 1973.
- [11] R. S. Burachik, C. A. Sagastizabal, and S. Scheimberg, *An inexact method of partial inverses : Application to splitting methods*, Optimization Methods and Software **3** (2006), no. 21, 385–400.
- [12] J. V. Burke and Maijian Qian, *A variable metric proximal point algorithm for monotone operators*, SIAM Journal on Control and Optimization **37** (1999), no. 2, 353–375.
- [13] George H.G. Chen and R.T. Rockafellar, *Convergence rates in forward-backward splitting*, SIAM, Journal on Optimization **7** (1997), no. 2, 421–444.
- [14] Alice Chiche, *Comparaison d'algorithmes de décomposition*, Tech. report, EDF, 2008.
- [15] Guy Cohen, *Optimization by decomposition and coordination : a unified approach*, IEEE Transactions on Automatic Control (1978), no. AC-23, 222–232.
- [16] ———, *Auxiliary problem principle and decomposition of optimization problems*, J. Opt. Theory Appl. (1980), no. 32, 277–305.

- [17] ———, *Décomposition et coordination en optimisation déterministe, différentiable et non-différentiable*, Ph.D. thesis, Université Paris Dauphine, 1984.
- [18] ———, *Optimal scenario tree topology and corresponding rate of convergence*, 11th Conference on Stochastic Programming (SPXI), 2007.
- [19] Guy Cohen and Dao Li Zhu, *Decomposition coordination methods in large scale optimization problems : the non-differentiable case and the use of augmented lagrangians*, Advances in large scale systems **1** (1984), 203–266.
- [20] Patrick Louis Combettes, *Fejér-monotonicity in convex optimization*, Encyclopedia of Optimization (2000).
- [21] ———, *Solving monotone inclusions via compositions of nonexpansive averaged operators*, Optimization **53** (2004), no. 5, 475–504.
- [22] M. Daldoul, *Finite convergence of the partial inverse algorithm*, Journal of Optimization Theory and Applications **95** (1997), no. 3, 693–699.
- [23] Anes Dallagi, *Méthodes particulières en commande optimale stochastique*, Ph.D. thesis, Université Paris I, Janvier 2007.
- [24] G.B Dantzig, *Linear programming under uncertainty*, Management Science (1955), no. 1, 197–206.
- [25] G.B Dantzig and P. Wolfe, *The decomposition algorithm for linear programming*, Operations Research (1960), no. 8.
- [26] Amal de Silva and David Abramson, *Computational experience with the parallel progressive hedging algorithm for stochastic linear programs*, Proceedings of 1993 Parallel Computing and Transputers Conference Brisbane (1993), 164–174.
- [27] Darinka Dentcheva and Werner Romisch, *Optimal power generation under uncertainty via stochastic programming*, in : Stochastic Programming Methods and Technical Applications (K. Marti and P. Kall Eds.), Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, 1998, pp. 22–56.
- [28] Luc P Devroye, *On the almost everywhere convergence of nonparametric regression function estimates*, The Annals of Statistics **9** (1981), no. 6, 1310–1319.
- [29] Luc P. Devroye and T.J. Wagner, *Distribution-free consistency results in nonparametric discrimination and regression function estimation*, The Annals of Statistics **8** (1980), no. 2, 231–239.
- [30] J Douglas and H. H. Rachford, *On the numerical solution of the heat conduction problem in 2 and 3 space variables*, Trans. Amer. Math. Soc. (1956), no. 82, 421–439.
- [31] Jean-Pierre Dussault, Oumar Mandione Guèye, and Philippe Mahey, *Separable augmented lagrangian algorithm with multidimensional scaling for monotropic programming*, Journal of Optimization Theory and Application **127** (2005), 1–20.
- [32] Jonathan Eckstein, *The Lions-Mercier splitting algorithm and the alternating direction method are instances of the proximal point algorithm*, Tech. Report LIDS-P-1769, 1988.
- [33] ———, *Splitting methods for monotone operators with applications to parallel optimization*, Ph.D. thesis, Massachusetts institute of technology, cambridge, June 1989.
- [34] Jonathan Eckstein and Dimitri P. Bertsekas, *An alternating direction method for linear programming*, Tech. report, Laboratory for Information and Decision Sciences, MIT, April 1990.

- [35] ———, *On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotones operators*, *Mathematical Programming* **55** (1992), 293–318.
- [36] Jonathan Eckstein and Michael C. Ferris, *Operator splitting methods for monotone affine variational inequalities, with a parallel application to optimal control*, *INFORMS Journal on Computing* **10** (1998), no. 2, 218–235.
- [37] Jonathan Eckstein and Masao Fukushima, *Some reformulations and applications of the alternating direction method of multipliers*, *Large Scale Optimization : State of the Art* (1993), 119–138.
- [38] M. Gabay, *Applications of the method of multipliers to variational inequalities*, *Augmented Lagrangian Methods : Applications to the Solution of Boundary Value Problems* (1983).
- [39] Vincent Grellier, Pascal Martinetto, Jean Maeght, and Olivier Teytaud, *Choisir les bons arbres*, *Rapport EDF* (2004).
- [40] Nicole Gröwe-Kuska, Holger Heitsch, and Werner Römisch, *Scenario reduction and scenario tree construction for power management problem*, *Power Tech Conference Proceedings, 2003 IEEE Bologna* (2003).
- [41] A. Hamdi, P. Mahey, and J.P Dussault, *A new decomposition method in nonconvex programming via a separable augmented lagrangian*, *Lecture Notes in Economics and Mathematical Systems* **452** (1997), 90–104.
- [42] S.P. Han and G. Lou, *A parallel algorithm for a class of convex programs*, *SIAM J. Control Optim.* (1988), no. 29, 345–355.
- [43] B.S He, L-Z Liao, and S.L Wang, *Self-adaptive operator splitting methods for monotone variational inequalities*, *Numerische Mathematik* **94** (2003), 715–737.
- [44] B.S He, H. Yang, and S.L Wang, *Alternating directions method with self-adaptive penalty parameters for monotone variational inequalities*, *Journal of Optimization Theory and applications* **106** (2000), 349–368.
- [45] Holger Heitsch and Werner Romisch, *Generation of multivariate scenario trees to model stochasticity in power management*, *IEEE St. Petersburg Power Tech 2005* (2005).
- [46] ———, *Scenario tree modeling for multistage stochastic programs*, *Preprint, Humboldt-University Berlin, Institute of Mathematics* (2005).
- [47] Michael Held and Richard M. Karp, *The traveling salesman problem and minimum spanning trees*, *Operations Research* (1970), 1138–1162.
- [48] M.R Hestenes, *Multiplier and gradient methods*, *Journal of Optimization Theory and Applications* (1969), no. 4, 303–320.
- [49] H. Idrissi, Odile Lefebvre, and Christian Michelot, *Applications and numerical convergence of the partial inverse method*, *Lecture notes in mathematics* **1405** (1989), 39–54.
- [50] Yosida K., *Functional analysis*, Springer Verlag, Berlin, 1964.
- [51] Peter Kall and Stein W. Wallace, *Stochastic programming*, John Wiley & Sons, 1994.
- [52] Spyridon A. Kontogiorgis, *Alternating directions methods for the parallel solution of large-scale block-structured optimization problems*, Ph.D. thesis, University of Wisconsin - Madison, 1994.

- [53] Spyridon A. Kontogiorgis and Robert R. Meyer, *A variable-penalty alternating directions method for convex optimization*, Mathematical Programming (1998).
- [54] Leon S. Lasdon, *Optimization theory for large systems*, Macmillan, 1970.
- [55] Jim Lawrence and Jonathan. E. Spingarn, *On fixed points of nonexpansive piecewise isometric mappings*, Proceedings of the London Mathematical Society **55** (1987), 605–624.
- [56] Odile Lefebvre and Christian Michelot, *Calcul d'un point fixe d'une application prox par la méthode des approximations successives ; conditions de convergence finie*, Comptes rendus de l'Académie des Sciences, Paris **303** (1986), no. 17, 905–908.
- [57] Claude Lemaréchal, *Optimisation non-différentiable : Méthodes des faisceaux*, Lecture notes in Mathematics **704** (1977), no. 1979, 53–61.
- [58] Claude Lemaréchal and Claudia Sagastizábal, *More than first-order developments of convex functions : primal-dual relations*, Journal of convex analysis **3** (1996), no. 2, 255–268.
- [59] ———, *Practical aspects of the moreau-yosida regularization : theoretical preliminaries*, SIAM J. Optim **7** (1997), no. 2, 367–385.
- [60] Claude Lemaréchal, François Oustry, and Claudia Sagastizábal, *The u-lagrangian of a convex function*, Transactions of the American Mathematical Society **352** (2000), no. 2, 711–729.
- [61] Claude Lemaréchal and Claudia Sagastizábal, *Variable metric bundle methods : From conceptual to implementable forms*, Mathematical Programming **76** (1997), no. 3, 393–410.
- [62] Arnaud Lenoir and Kengy Barty, *Mid-term electricity production scheduling by decomposition-coordination*, EngOpt 2008 - International Conference on Engineering Optimization Rio de Janeiro, Brazil, June 2008.
- [63] Arnaud Lenoir and Philippe Mahey, *Accelerating convergence of separable augmented lagrangian algorithms*, International Conference on Mathematics of Optimization and Decision Making, Université Antilles-Guyane Guadeloupe, Point-à-Pitre, avril 2006.
- [64] ———, *Accelerating the convergence of a separable augmented lagrangian algorithm*, Research Report LIMOS, Université Blaise Pascal (2007), no. RR-07-14.
- [65] ———, *Global and adaptive scaling in a separable augmented lagrangian algorithm*, Optimization Online (2007).
- [66] ———, *Accelerating convergence of a class of splitting algorithms with iterative foldings*, Acta Mathematica Vietnamica (2008).
- [67] ———, *On the application of the proximal decomposition method on linear programs*, Journées MODE 2008, ISIMA-LIMOS Clermont-Ferrand, février 2008.
- [68] P.L. Lions and B. Mercier, *Splitting algorithms for the sum of two nonlinear operators*, SIAM Journal on Numerical Analysis **16** (1979), 964–979.
- [69] Yves Lucet, *Formule explicite du hessien de la régularisée de moreau-yosida d'une fonction convexe f en fonction de l'épi-différentielle seconde de f.*, Tech. report, Laboratoire Approximation et Optimisation, Université Paul Sabatier, Toulouse, France, 1995.

- [70] Fernando Ravier Luque, *Asymptotic convergence analysis of the proximal point algorithm*, SIAM J. Control and optimization **22** (1984), 277–293.
- [71] Philippe Mahey, *Modèle géode de planification moyen terme, étude des décompositions potentielles, découplage et décentralisation*, Document de travail EDF (2004).
- [72] Philippe Mahey, Jean-Pierre Dussault, Abdelhamid Benchakroun, and Abdelouahed Hamdi, *Adaptive scaling and convergence rates of a separable augmented lagrangian algorithm*, Lecture Notes in Economics and Mathematical Systems **481** (2000), 278–287.
- [73] Philippe Mahey, Said Oualibouch, and Pham Din Tao, *Proximal decomposition on the graph of a maximal monotone operator*, SIAM J. Optimization **5** (1995), 454–466.
- [74] B. Martinet, *Régularisation d'inéquations variationnelles par approximations successives*, Revue Française d'Informatique et de Recherche Opérationnelle (1970), 154–159.
- [75] ———, *Détermination approchée d'un point fixe d'une application pseudo-contractante. cas de l'application prox*, Comptes rendus de l'Académie des Sciences, Paris **A274** (1972), 163–165.
- [76] G.J. Minty, *Monotone (nonlinear) operators in Hilbert space*, Duke Math. J. (1962), no. 29, 341–346.
- [77] Jean-Jacques Moreau, *Fonctions convexes duales et points proximaux dans un espace hilbertien*, Comptes rendus de l'Académie des Sciences, Paris **93** (1962), 273–299.
- [78] ———, *Proximité et dualité dans un espace hilbertien*, Bulletin de la Société Mathématique Française **93** (1965), 273–299.
- [79] E. A. Nadaraya, *On estimating regression*, Theory Probab. Applic. **9** (1964), 141–142.
- [80] M. P. Nowak and W. Romisch, *Optimal power dispatch via multistage stochastic programming*, in : Progress in Industrial Mathematics at ECMI 96, eds. M. Brns, M.P. Bendse and M.P. Srensen, Teubner, 1997, pp. 324–331.
- [81] Adam Ouorou, *Décomposition proximale des problèmes de multiflot à critère convexe. application aux problèmes de routage dans les réseaux de télécommunication*, Thèse de doctorat, Université Blaise Pascal, 1995.
- [82] ———, *Epsilon-proximal decomposition method*, Mathematical programming (2004), no. 99, 89–108.
- [83] G.B Passty, *Ergodic convergence to a zero of the sum of monotone operators in Hilbert space*, J. Math. Anal. Appl. (1979), no. 72, 383–390.
- [84] D.H. Peaceman and H. H. Rachford, *The numerical solution of parabolic elliptic differential equations*, J. Soc. Indust. Appl. Math. (1955), no. 3, 28–41.
- [85] François Pelligrino, *Optimiser les plannings de production sur la journée : le modèle apogée*, Tech. report, Électricité de France, Direction des Études et Recherches, Service Études de Réseaux, Département MOS, 1991.
- [86] Teemu Pennanen, *Epi-convergent discretizations of multistage stochastic programs*, Mathematics of Operations Research **30** (2005), 245–256.

- [87] Teemu Pennanen and Markku Kallio, *A splitting method for stochastic programs*, Annals of Operations Research **142** (2006), no. 1, 259–268.
- [88] G. Ch. Pflug, *Optimization of stochastic models. the interface between simulation and optimization*, Kluwer, Dordrecht, 1996.
- [89] R. A. Poliquin and R. T. Rockafellar, *Generalized hessian properties of regularized nonsmooth functions*, SIOPT **6** (1996), no. 4, 1121–1137.
- [90] Arnaud Renaud, *Dédoublément des variables et lagrangien augmenté : proposition de deux algorithmes pour la gestion journalière de l'hydraulique*, Tech. report, Électricité de France, Direction des Études et Recherches, Service Études de Réseaux, Département MOS, 1991.
- [91] Stephen. M Robinson, *Extended scenario analysis*, Annals of Operations Research **31** (1991), 385–398.
- [92] R. Tyrell Rockafellar, *Convex analysis*, Princeton University Press, 1970.
- [93] ———, *Augmented lagrangians and applications of the proximal point algorithm in convex programming*, Mathematics of operations research **1** (1976), 97–116.
- [94] ———, *Monotone operators and the proximal point algorithm*, SIAM J. Control and optimization **14** (1976), 877–898.
- [95] ———, *Maximal monotone relations and the second derivatives of nonsmooth functions*, Annales de l'I.H.P. **tome 2** (1985), no. 3, 167–184.
- [96] ———, *Proto-differentiability of set-valued mappings and its applications in optimization*, Annales de l'I.H.P. **tome S6** (1989), 449–482.
- [97] R. Tyrell Rockafellar and Roger. J-B Wets, *Scenarios and policy aggregation in optimization under uncertainty*, Mathematics of Operations Research **16** (1991), 119–147.
- [98] ———, *Variational analysis*, Springer, 1998.
- [99] François Romanet, *Soprano : Construction des arbres d'aléa*, Tech. report, EDF, avril 2003.
- [100] ———, *Soprano : Modélisation et résolution des sous-problèmes thermiques nucléaires*, Tech. report, EDF, avril 2003.
- [101] ———, *Soprano : Méthodologie générale de résolution*, Tech. report, EDF, avril 2003.
- [102] Jean-Sébastien Roy and Arnaud Lenoir, *Non-parametric approximation of non-anticipativity*, 22nd European Conference on Operational Research, July 2007.
- [103] ———, *Non-parametric approximation of non-anticipativity constraints in scenario-based multistage stochastic programming*, Kybernetika **44** (2008), no. 2.
- [104] A. Ruszczyński and A Shapiro (eds.), *Stochastic programming*, Handbooks in Operations Research and Management Science, vol. 10, Elsevier, Amsterdam, 2003.
- [105] Andrzej Ruszczyński, *Regularized decomposition of stochastic programs : algorithmic techniques and numerical results*, Working Paper (1994).
- [106] ———, *Decomposition methods in stochastic programming*, Mathematical Programming **79** (1997), 333–353.
- [107] David. H Salinger and R. Tyrrel Rockafellar, *Dynamic splitting :an algorithm for deterministic and stochastic multiperiod optimization*, Working Paper (2003).

- [108] Dieter Schott, *Basic properties of fejer monotone sequences*, Rostock. Math. Kolloq. (1995), no. 49, 57–74.
- [109] N. Z. Shor, *Application of the gradient descent method to solution of the network transport problem*, Cyber. Oper. Res. **1** (1962), 9–17.
- [110] M. V. Solodov and B. F. Svaiter, *A hybrid projection-proximal point algorithm*, Journal of convex analysis **6** (1999), 59–70.
- [111] Michael Somervell, *Progressive hedging in parallel*, Proceedings of the 33rd ORSNZ annual conference (1998).
- [112] C. Spiegelman and J. Sacks, *Consistent window estimation in nonparametric regression*, Ann. Statist. **8** (1980), 240–246.
- [113] Jonathan E. Spingarn, *Partial inverse of a monotone operator*, Applied mathematics and optimization **10** (1983), 247–265.
- [114] ———, *Applications of the method of partial inverses to convex programming :decomposition*, Mathematical Programming **32** (1985), 199–223.
- [115] ———, *A primal-dual projection method for solving systems of linear inequalities*, Linear algebra and its applications **65** (1985), 45–62.
- [116] ———, *A projection method for least-squares solutions to overdetermined systems of linear inequalities*, Linear algebra and its applications **86** (1987), 211–236. MR 88k :65058
- [117] Cyrille Strugarek, *Quelle approche pour discrétiser un problème d’optimisation stochastique ?*, Rapport EDF (2004).
- [118] Cyrille Strugarek and Hubert Fedry, *Un autre regard sur la construction d’arbres d’aléas pour soprano*, Rapport EDF (2004).
- [119] Philippe Torrion and Jean Leveugle, *Comparaison de différentes méthodes d’optimisation appliquées à la gestion annuelle du système offre-demande d’électricité français*, Tech. report, Électricité de France, Direction générale, Département Méthodes, 1991.
- [120] Paul Tseng, *Applications of a splitting algorithm to decomposition in convex programming and variational inequalities*, SIAM Journal on Control and Optimization **29** (1991), no. 1, 119–138.
- [121] John N. Tsitsiklis, *On the stability of asynchronous iterative processes*, Tech. report, Laboratory for information and decision systems, MIT, Cambridge, 1985.
- [122] John N. Tsitsiklis, Dimitri P. Bertsekas, and Michael Athans, *Distributed asynchronous deterministic and stochastic gradient optimization algorithms*, IEEE transactions on automatic control **AC-31** (1986), no. 9, 803–812.
- [123] G.S. Watson, *Smooth regression analysis*, Sankhya Ser. A **26** (1964), 359–372.
- [124] Roger. J-B Wets, *Stochastic programming models : wait-and-see versus here-and-now*, Decision Making Under Uncertainty : Energy and Power (2002), 1–16.
- [125] Ciyou Zhu, *Asymptotic convergence analysis of the forward-backward splitting algorithm*, Mathematics of Operations Research **20** (1995), 449–464.