Algorithmes d'optimisation en grande dimension : applications à la résolution de problèmes inverses

Audrey Repetti

Université Paris-Est

Laboratoire d'Informatique Gaspard Monge

UMR CNRS 8049

Thèse de Doctorat

Présentée le 29 juin 2015

par

Audrey Repetti

Algorithmes d'optimisation en grande dimension : applications à la résolution de problèmes inverses

---

Rapporteurs :	Jérôme Bolte	*Université Toulouse 1 Capitole*
	Said Moussaoui	*École Centrale Nantes*
Examinateurs :	Patrick Louis Combettes	*Université Pierre et Marie Curie*
	Stéphane Mallat	*École Normale Supérieure*
	Gabriele Steidl	*Université Technologique de Kaiserslautern*
Directeur :	Jean-Christophe Pesquet	*Université Paris-Est Marne-la-Vallée*
Co-encadrante :	Émilie Chouzenoux	*Université Paris-Est Marne-la-Vallée*

Remerciements

Je souhaite en premier lieu remercier les personnes qui m'ont encadrée, M. Jean-Christophe Pesquet et Mme Emilie Chouzenoux, qui m'ont fait confiance et qui m'ont permis de mener à bien cette thèse. Je leur suis particulièrement reconnaissante pour le temps qu'ils m'ont accordé, pour leurs précieux conseils, ainsi que pour leur dynamisme.

Mes remerciements vont aussi aux membres du jury, qui ont accepté d'évaluer mon travail de thèse. Merci aux rapporteurs MM. Jérôme Bolte et Saïd Moussaoui pour leur relecture attentive de mon manuscrit, à M. Patrick Louis Combettes et Mme Gabriele Steidl d'avoir accepté de faire partie du jury de cette thèse, ainsi qu'à M. Stéphane Mallat d'en avoir accepté la présidence.

Je remercie également Mme Gabriele Steidl de m'avoir accueillie dans son équipe pendant un mois, et M. Laurent Duval de m'avoir permis de travailler avec lui.

Je tiens de plus à remercier Patrice pour m'avoir apporté son aide à plusieurs reprises, autant pour mes problèmes informatiques que lorsque ma voiture restait enfermée, Corinne, Laurence et Séverine pour leur sympathie et leur aide dans les démarches administratives, et Sylvie pour sa bonne humeur et sa patience.

Je souhaite enfin adresser mes remerciements aux doctorants arrivés avant/en même temps/ après moi, et qui ont aussi contribué, par leur bonne humeur et leur soutien, à cette thèse. Je remercie Aurélie, Gia-Thuy, Mohamed-Ali, Sonja, et plus particulièrement Ania, Feriel, Mai, Mireille et Yosra, sans qui l'ambiance de la thèse n'aurait pas été aussi bonne.

Finalement, mes remerciements vont à mes amis et ma famille, qui me soutiennent depuis toujours et qui sont certainement ma plus grande source de motivation.

Je remercie mon cousin Marc de m'avoir transmis son intérêt pour les maths, ainsi que Anne-Catherine, Roxanne, Eloise, et Victor.

Je remercie mon cousin Vincent, Valérie, Thomas et Lucie d'avoir été d'une grande patience lors des soirées raclette/travail, et qui ont toujours été à mes côtés pour m'apporter du bonheur. Merci aussi à mon parrain Patrick pour son soutien malgré la distance qui nous sépare.

Je remercie mon grand-père Nando, Michela, Antonella et Giorgio pour leur soutien lors de mes séjours en Italie.

Je remercie Angélique, Benjamin, Benoit, Bérénice, Dorian, Giovanni, Guillaume, Isa, JS, Margot, Sébastien, Walid, qui m'apportent une grande dose de joie de vivre et de folie.

Et enfin, je remercie du fond du coeur mes parents, qui m'ont épaulée tout au long de cette thèse, et pendant toutes les années qui l'ont précédée, ainsi que Clément, qui m'a permis de m'évader au quotidien par sa bonne humeur, son empathie et sa patience.

Table des matières

Table des figures

Liste des tableaux

Liste des algorithmes

Abréviations

BCD	:	Block Coordinate Descent
BC-FB	:	Block Coordinate Forward-Backward
BC-VMFB	:	Block Coordinate Variable Metric Forward-Backward
FB	:	Forward-Backward
KL	:	Kurdyka-Łojasiewicz
MAP	:	Maximum *a posteriori*
MM	:	Majoration-Minimisation
PALM	:	Proximal Alternating Linearized Minimization
SDP	:	Semi Définie Positive
VMFB	:	Variable Metric Forward-Backward

Résumé

Une approche efficace pour la résolution de problèmes inverses consiste à définir le signal (ou l'image) recherché(e) par minimisation d'un critère pénalisé. Ce dernier s'écrit souvent sous la forme d'une somme de fonctions composées avec des opérateurs linéaires. En pratique, ces fonctions peuvent n'être ni convexes ni différentiables. De plus, les problèmes auxquels on doit faire face sont souvent de grande dimension. L'objectif de cette thèse est de concevoir de nouvelles méthodes pour résoudre de tels problèmes de minimisation, tout en accordant une attention particulière aux coûts de calculs ainsi qu'aux résultats théoriques de convergence.

Une première idée pour construire des algorithmes rapides d'optimisation est d'employer une stratégie de préconditionnement, la métrique sous-jacente étant adaptée à chaque itération. Nous appliquons cette technique à l'algorithme explicite-implicite et proposons une méthode, fondée sur le principe de majoration-minimisation, afin de choisir automatiquement les matrices de préconditionnement. L'analyse de la convergence de cet algorithme repose sur l'inégalité de Kurdyka-Łojasiewicz.

Une seconde stratégie consiste à découper les données traitées en différents blocs de dimension réduite. Cette approche nous permet de contrôler à la fois le nombre d'opérations s'effectuant à chaque itération de l'algorithme, ainsi que les besoins en mémoire, lors de son implémentation. Nous proposons ainsi des méthodes alternées par bloc dans les contextes de l'optimisation non convexe et convexe. Dans le cadre non convexe, une version alternée par bloc de l'algorithme explicite-implicite préconditionné est proposée. Les blocs sont alors mis à jour suivant une règle déterministe acyclique. Lorsque des hypothèses supplémentaires de convexité peuvent être faites, nous obtenons divers algorithmes proximaux primaux-duaux alternés, permettant l'usage d'une règle aléatoire arbitraire de balayage des blocs. L'analyse théorique de ces algorithmes stochastiques d'optimisation convexe se base sur la théorie des opérateurs monotones.

Un élément clé permettant de résoudre des problèmes d'optimisation de grande dimension réside dans la possibilité de mettre en œuvre en parallèle certaines étapes de calculs. Cette parallélisation est possible pour les algorithmes proximaux primaux-duaux alternés par bloc que nous proposons : les variables primales, ainsi que celles duales, peuvent être mises à jour en parallèle, de manière tout à fait flexible. A partir de ces résultats, nous déduisons de nouvelles méthodes distribuées, où les calcul sont répartis sur différents agents communiquant entre eux suivant une topologie d'hypergraphe.

Finalement, nos contributions méthodologiques sont validées sur différentes applications en traitement du signal et des images. Nous nous intéressons dans un premier temps à divers problèmes d'optimisation faisant intervenir des critères non convexes, en particulier en restauration d'images lorsque l'image originale est dégradée par un bruit gaussien dépendant du signal, en démélange spectral, en reconstruction de phase en tomographie, et en déconvolution aveugle pour la reconstruction de signaux sismiques parcimonieux. Puis, dans un second temps, nous abordons des problèmes convexes intervenant dans la reconstruction de maillages 3D et dans l'optimisation de requêtes pour la gestion de bases de données.

Abstract

An efficient approach for solving an inverse problem is to define the recovered signal/image as a minimizer of a penalized criterion which is often split in a sum of simpler functions composed with linear operators. In the situations of practical interest, these functions may be neither convex nor smooth. In addition, large scale optimization problems often have to be faced. This thesis is devoted to the design of new methods to solve such difficult minimization problems, while paying attention to computational issues and theoretical convergence properties.

A first idea to build fast minimization algorithms is to make use of a preconditioning strategy by adapting, at each iteration, the underlying metric. We incorporate this technique in the forward-backward algorithm and provide an automatic method for choosing the preconditioning matrices, based on a majorization-minimization principle. The convergence proofs rely on the Kurdyka-Łojasiewicz inequality.

A second strategy consists of splitting the involved data in different blocks of reduced dimension. This approach allows us to control the number of operations performed at each iteration of the algorithms, as well as the required memory. For this purpose, block alternating methods are developed in the context of both non-convex and convex optimization problems. In the non-convex case, a block alternating version of the preconditioned forward-backward algorithm is proposed, where the blocks are updated according to an acyclic deterministic rule. When additional convexity assumptions can be made, various alternating proximal primal-dual algorithms are obtained by using an arbitrary random sweeping rule. The theoretical analysis of these stochastic convex optimization algorithms is grounded on the theory of monotone operators.

A key ingredient in the solution of high dimensional optimization problems lies in the possibility of performing some of the computation steps in a parallel manner. This parallelization is made possible in the proposed block alternating primal-dual methods where the primal variables, as well as the dual ones, can be updated in a quite flexible way. As an offspring of these results, new distributed algorithms are derived, where the computations are spread over a set of agents connected through a general hypergraph topology.

Finally, our methodological contributions are validated on a number of applications in signal and image processing. First, we focus on optimization problems involving non-convex criteria, in particular image restoration when the original image is corrupted with a signal dependent Gaussian noise, spectral unmixing, phase reconstruction in tomography, and blind deconvolution in seismic sparse signal reconstruction. Then, we address convex minimization problems arising in the context of 3D mesh denoising and in query optimization for database management.

Chapitre 1

Introduction générale

1.1 Contexte

Au cours de cette dernière décennie, de nombreuses applications, parmi lesquelles l'imagerie médicale et satellitaire, le traitement de maillages 3D de grande tailles, ou la gestion de bases de données, ont conduit à traiter des données dont la dimension est de plus en plus grande. Il devient alors nécessaire de développer des méthodes permettant un traitement rapide de ces grandes masses de données.

La plupart de ces applications font intervenir un problème inverse visant à estimer des données inconnues à partir d'une version dégradée de ces dernières. Une méthode efficace consiste à définir l'estimée comme étant une solution d'un problème d'optimisation, où l'objectif est de minimiser un critère composé de fonctions qui ne sont ni nécessairement différentiables, ni nécessairement convexes. Dans cette thèse, nous élaborerons plusieurs stratégies qui vont nous permettre d'obtenir des méthodes de minimisation rapides, tout en réduisant les coûts de calculs et les besoins en mémoire.

La première stratégie, permettant d'accélérer la convergence d'algorithmes d'optimisation, consiste à utiliser une technique de préconditionnement se basant sur une modification de la métrique sous-jacente au cours des itérations. Un aspect intéressant de notre étude concerne le choix des matrices de préconditionnement. Nous proposerons une méthode, basée sur une stratégie de majoration-minimisation (MM), permettant de les choisir efficacement. Cette méthode fera l'objet des chapitres 3 et 5. Nos résultats théoriques seront validés par des exemples numériques dans les chapitres 4 et 6.

La seconde stratégie, permettant d'optimiser les besoins en mémoire lors de l'implémentation d'algorithmes d'optimisation est de diviser les données traitées en différents blocs de dimension plus petite. Ainsi, chaque bloc peut être mis à jour séparément. Dans certains domaines du traitement de données de grande taille, par exemple en astronomie, l'image est partitionnée en plusieurs blocs, qui sont traités indépendamment les uns des autres. L'inconvénient de cette approche est qu'elle introduit des artefacts au niveau des frontières des blocs. Afin d'éviter ces effets indésirables, on peut introduire une étape empirique de lissage, et/ou considérer des blocs pouvant se chevaucher. Dans cette thèse, des méthodes rigoureuses opérant par blocs seront proposées dans le cadre de l'optimisation non convexe dans le chapitre 5, et convexe pour les méthodes primales-duales élaborées dans le chapitre 7. Nous montrerons, de plus, dans le chapitre 6 que

les stratégies par bloc peuvent également permettre, en plus d'une réduction de l'occupation mémoire, d'accélérer la vitesse de convergence des algorithmes.

Enfin, une dernière stratégie permettant de résoudre efficacement les problèmes d'optimisation de grande taille consiste à implémenter en parallèle certaines tâches des algorithmes. En raison des progrès technologiques actuels, la conception d'algorithmes parallèles permettant de bénéficier de systèmes multicœurs est souvent d'une importance primordiale pour obtenir des mises en œuvre rapides. Les algorithmes proximaux, basés sur des méthodes primales-duales développés ces dernières années en sont d'excellents exemples. Dans ces méthodes, les calculs des opérateurs proximaux et les étapes de gradient peuvent être effectués en parallèle. Cependant, lorsque le nombre de processeurs disponibles est inférieur au nombre d'opérations à effectuer en parallèle, la plupart des algorithmes proximaux parallèles existants deviennent alors d'un intérêt limité. Dans le chapitre 7 nous proposerons une méthode par bloc permettant de surmonter cette limitation en réduisant de manière flexible le nombre d'opérations effectuées à chaque itération. D'autre part, dans le chapitre 8 nous traiterons un exemple applicatif en bases de données à l'aide d'algorithmes parallèles.

1.2 Collaborations

Certains travaux présentés dans cette thèse ont été réalisés dans le cadre de collaborations :

- Une première collaboration avec Gabriele Steidl de l'Université de Kaiserslautern (Allemagne) dans le cadre du Labex Bezout a eu pour objectif de développer des méthodes basées sur les opérateurs proximaux et les projections épigraphiques pour traiter des fonctions quotient dans le cadre de l'optimisation de requêtes dans les systèmes de gestion de bases de données. Cette collaboration m'a amenée à passer un mois dans le laboratoire de mathématiques de l'Université de Kaiserslautern. Les résultats obtenus sont présentés dans le chapitre 8.
- Une seconde collaboration avec Laurent Duval de L'IFP Énergies Nouvelles a permis de développer une méthode pour traiter des problèmes de déconvolution aveugle de signaux sismiques parcimonieux. Cette collaboration m'a amené à travailler avec Mai Quyen Pham, doctorante à l'université de Paris-Est et à l'IFP Énergies Nouvelles. Les résultats que nous avons obtenus sont présentés dans le chapitre 6.
- Enfin une dernière collaboration s'est déroulée dans le cadre du projet ANR GRAPHSIP (Graph Signal Processing) démarré fin 2014, au cours de laquelle nous avons développé des algorithmes primaux-duaux permettant de résoudre des problèmes de débruitage de maillages 3D. Les résultats obtenus sont présentés dans le chapitre 7.

1.3 Organisation du document

Chapitre 2 : Généralités sur les problèmes inverses et l'optimisation

Le deuxième chapitre sera divisé en trois grandes parties. Tout d'abord, nous expliquerons brièvement comment, partant d'un problème inverse, on peut être conduit à un problème d'optimisation, en utilisant une interprétation de type Maximum a Posteriori (MAP). Dans la seconde partie, nous donnerons les principales notations et outils mathématiques qui seront utilisés tout au long de cette thèse. En particulier, nous nous intéresserons aux opérateurs proximaux de fonctions non nécessairement convexes et à l'inégalité de Kurdyka-Łojasiewicz. Pour finir, dans la dernière partie de ce chapitre, un état de l'art des algorithmes usuels existants permettant de résoudre des problèmes de minimisation sera présenté. En particulier, nous y évoquerons des méthodes telles que les algorithmes de gradient, les algorithmes proximaux, les algorithmes de Majoration-Minimisation, et les méthodes primales-duales.

Chapitre 3 : Algorithme explicite-implicite à métrique variable

Une méthode usuelle permettant de minimiser une somme de deux fonctions, l'une étant différentiable et l'autre convexe, est l'algorithme explicite-implicite, qui alterne, à chaque itération, une étape de gradient sur la fonction différentiable et une étape proximale sur la fonction convexe. Un inconvénient majeur de cet algorithme est sa vitesse de convergence, qui peut se révéler très lente en pratique. Dans ce chapitre nous proposerons donc d'en accélérer la convergence, en changeant, à chaque itération, la métrique sous-jacente, par l'introduction d'une métrique variable. Cette métrique sera choisie en se basant sur une stratégie de Majoration-Minimisation. Nous fournirons une analyse de la convergence de cet algorithme basée sur l'inégalité de Kurdyka-Łojasiewicz. De plus, une version inexacte de l'algorithme sera proposée, permettant de prendre en compte des erreurs de calcul pouvant apparaître au cours des itérations.

Chapitre 4 : Application de l'algorithme explicite-implicite à métrique variable à un bruit gaussien dépendant du signal

Dans ce chapitre nous allons nous intéresser au problème de restauration d'un signal dégradé par un bruit gaussien dépendant. Un tel bruit peut s'observer dans divers systèmes d'imagerie, dans lesquels les acquisitions sont dégradées à la fois par un bruit gaussien dépendant lié au comptage des photons, et par un bruit électronique indépendant du signal d'origine. Nous montrerons que l'interprétation MAP de ce modèle de bruit conduit à une fonction d'attache aux données non convexe. Nous proposerons donc de résoudre le problème de minimisation associé à ce modèle grâce à l'algorithme explicite-implicite à métrique variable, développé au chapitre précédent. L'intérêt d'utiliser une métrique variable sera illustrée sur un exemple pratique en restauration d'images.

Chapitre 5 : Algorithme explicite-implicite à métrique variable alterné par bloc

Dans de nombreuses applications, telles que la résolution de problèmes de déconvolution aveugle, ou lorsque les données considérées sont de très grande dimension, il est intéressant de considérer des méthodes de minimisation alternée. Dans ce chapitre, nous proposerons donc une version alternée par bloc de l'algorithme explicite-implicite à métrique variable développé dans le troisième chapitre. Nous étudierons la convergence et la vitesse de convergence de l'algorithme proposé, dans des versions exacte et inexacte, en considérant une règle de mise à jour acyclique des blocs.

Chapitre 6 : Quelques applications de l'algorithme explicite-implicite à métrique variable alterné par bloc

Afin d'illustrer les bonnes performances pratiques de l'algorithme explicite-implicite à métrique variable alterné par bloc, nous donnerons dans ce chapitre trois exemples de simulation. Dans le premier exemple, nous considérerons un problème de démélange spectral, une application en reconstruction de phase pour un problème de tomographie, et un exemple de déconvolution aveugle pour la restauration de signaux sismiques parcimonieux. Ces applications nous permettrons d'illustrer l'intérêt pratique de l'algorithme proposé au chapitre précédent. En particulier, nous montrerons que l'utilisation d'une métrique variable et d'une règle acyclique de mise à jour des blocs conduit à des méthodes de résolution rapides pour les trois problèmes considérés.

Chapitre 7 : Algorithmes primaux-duaux alternés et distribués pour les opérateurs monotones

Dans les chapitre précédents nous avons présenté des méthodes d'optimisation efficaces pour la résolution de problèmes non convexes. Cependant, lorsque le critère fait intervenir des fonctions convexes non lisses composées avec des opérateurs linéaires, ces méthodes requièrent des sous-itérations qui peuvent s'avérer coûteuses.

Dans ce chapitre, nous nous plaçons dans le cadre de l'optimisation convexe, et nous nous intéressons à la classe des algorithmes primaux-duaux. Ces méthodes permettent de traiter des opérateurs linéaires apparaissant dans le problème, sans passer par leur inversion ni par des sous-itérations. Nous nous intéresserons plus généralement aux algorithmes primaux-duaux permettant de résoudre des problèmes d'inclusion monotone. Nous proposerons deux classes de nouveaux algorithmes stochastiques primaux-duaux, permettant de ne mettre à jour, à chaque itération, qu'une sous-partie des données globales sélectionnées de façon aléatoire. Nous analyserons la convergence presque sûre des variables aléatoires générées par les algorithmes proposés. Un exemple applicatif de débruitage de maillage 3D sera considéré. De plus, dans une seconde partie du chapitre, nous déduirons des méthodes précédentes des algorithmes stochastiques distribués, dans lesquels les étapes de calculs sont réparties sur différents agents communiquant entre eux suivant un schéma d'hypergraphe.

Chapitre 8 : Opérateur proximal de fonctions quotient

Dans ce chapitre nous nous intéresserons à l'application des algorithmes proximaux primaux-duaux à l'estimation de requêtes dans les systèmes de gestion de bases de données. Les problèmes sous-jacents dans ce type d'application requièrent la manipulation de fonctions $\mathbf{q}_1$ et $\mathbf{q}_\infty$, respectivement définies comme étant les normes ℓ_1 et ℓ_∞ de fonctions quotients. Nous montrerons d'une part que l'opérateur proximal de la fonction $\mathbf{q}_1$ correspond à un opérateur de seuillage. D'autre part, nous formulerons les problèmes faisant intervenir la fonction $\mathbf{q}_\infty$ comme des problèmes avec contraintes épigraphiques. Pour cela, nous étudierons l'opérateur de projection sur l'épigraphe de $\mathbf{q}_\infty$. Pour finir, nous illustrerons les performances des méthodes proposées par un exemple numérique.

1.4 Publications

*Pour les articles mentionnés par *, les noms des auteurs suivent l'ordre alphabétique, suivant les usages dans les journaux de mathématiques appliquées.*

Articles parus ou acceptés pour publication dans des journaux internationaux :

- E. Chouzenoux, J.-C. Pesquet, et A. Repetti. *
  Variable metric forward-backward algorithm for minimizing the sum of a differentiable function and a convex function,
  Journal of Optimization Theory and Applications, vol. 162, no. 1, pp 107–132, Juillet 2014.
- A. Repetti, M. Q. Pham, L. Duval, E. Chouzenoux et J.-C. Pesquet.
  Euclid in a taxicab : sparse blind deconvolution with smoothed ℓ_1/ℓ_2 regularization,
  Signal Processing Letters, vol. 22, no. 5, pp. 539–543, Mai 2015.
- G. Moerkotte, M. Montag, A. Repetti et G. Steidl. *
  Proximal operator of quotient functions with application to a feasibility problem in query optimization,
  Journal of Computational and Applied Mathematics, vol. 285, pp 243–255, Sept. 2015.
- J.-C. Pesquet et A. Repetti. *
  A class of randomized primal-dual algorithms for distributed optimization,
  Accepté pour publication dans *Journal of Nonlinear and Convex Analysis*, 2014.

Article soumis dans un journal international :

- E. Chouzenoux, J.-C. Pesquet, et A. Repetti. *
  A block coordinate variable metric forward-backward algorithm,
  Soumis à *Global Optimization*, 2013.
  http://www.optimization-online.org/DB_HTML/2013/12/4178.html.

Articles de conférences acceptés :

- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  A penalized weighted least squares approach for restoring data corrupted with signal-dependent noise.
  Dans *Proceedings of the 20th European Signal Processing Conference (EUSIPCO 2012)*, pages 1553–1557, Bucarest, Roumanie, 27-31 août 2012.
- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  Reconstruction d'image en présence de bruit gaussien dépendant par un algorithme explicite-implicite à métrique variable.
  Dans *Actes du 24e colloque GRETSI*, Brest, France, 3-6 septembre 2013.
- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  A nonconvex regularized approach for phase retrieval.
  Dans *Proceedings of IEEE International Conference on Image Processing (ICIP 2014)*, pp. 1753–1757, Paris, France, 27-30 octobre 2014.
- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  A random block-coordinate primal-dual proximal algorithm with application to 3D mesh denoising.
  Accepté à *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2015)*, Brisbane, Australie, 21-25 avril 2015.

Articles de conférences invités :

- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  A preconditioned forward-backward approach with application to large-scale nonconvex spectral unmixing problems.
  Dans *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2014)*, pp. 1512–1516, Florence, Italie, 4-9 mai 2014.
- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  Proximal primal-dual optimization methods.
  Dans *Proceedings of international Biomedical and Astronomical Signal Processing (BASP) Frontiers workshop*, pp. 25, Villars-sur-Ollon, Switzerland, 25-30 janvier 2015.

Articles de conférences soumis :

- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  A parallel block-coordinate approach for primal-dual splitting with arbitrary random block selection.
  Soumis à *European Signal Processing Conference (EUSIPCO 2015)*.
- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  Un petit tutoriel sur les méthodes primales-duales pour l'optimisation convexe.
  Soumis à *GRETSI 2015*.

Autres présentations :

- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  Variable metric forward-backward algorithm for minimizing the sum of a differentiable function and a convex function.
  Advances in Mathematical Image Processing (AIP), Annweiler, Allemagne, 30 septembre - 2 octobre 2013.
- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  Une nouvelle approche régularisée pour la reconstruction de phase.
  Journées Imagerie Optique Non Conventionnelle (JIONC), Paris, France, 19-20 mai 2014.
- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  Proximal methods : tools for solving inverse problems on a large scale.
  BioImage Informatics and Modeling at the Cellular Scale, Toulouse, France, 30 juin - 2 juillet 2014.
- A. Repetti, P. L. Combettes et J.-C. Pesquet.
  Randomized primal-dual methods based on stochastic quasi-Fejér monotonicity property.
  European Conference on Stochastic Programming and Energy Applications (ECSP 2014), Paris, IHP, France, 24 - 26 septembre 2014.
- A. Repetti, E. Chouzenoux, M. Q. Pham, L. Duval et J.-C. Pesquet.
  Algorithme préconditionné explicite-implicite et application à la résolution de problèmes inverses en traitement du signal.
  GdR ISIS - Optimisation non-convexe, Paris, France, 16 octobre 2014.
- A. Repetti, E. Chouzenoux et J.-C. Pesquet.
  A random block-coordinate proximal primal-dual optimization method with application to 3D mesh denoising.
  EPFL - Laboratoire de systèmes d'information et d'inférence, Lausanne, Suisse, 27 mars 2015.

Chapitre 2

Généralités sur les problèmes inverses et l'optimisation

Sommaire

2.1 Introduction

Ce chapitre a pour but, dans un premier temps, d'introduire les problèmes inverses en traitement du signal et des images, puis, dans un second temps, de définir les outils fondamentaux d'analyse et d'optimisation qui seront utilisés tout au long de cette thèse pour résoudre ces problèmes.

Les problèmes inverses et leurs liens avec les problèmes variationnels seront présentés dans la section 2.2. Puis dans la section 2.3 nous introduirons les outils fondamentaux d'analyse et d'optimisation. Enfin, dans la section 2.4 nous donnerons des exemples d'algorithmes utilisant ces outils pour résoudre des problèmes variationnels.

2.2 Problèmes inverses

2.2.1 Introduction

De nombreuses applications en traitement du signal et des images se ramènent à la résolution de problèmes inverses. Cette résolution consiste en l'estimation d'un signal original inconnu, à partir d'une version dégradée (ou incomplète) de celui-ci. Mathématiquement, le signal original peut être représenté par un vecteur $\overline{\mathbf{x}} \in \mathbb{R}^N$, et l'observation dégradée que l'on a de ce signal par un vecteur $\mathbf{z} \in \mathbb{R}^M$ lié à $\overline{\mathbf{x}}$ par la relation

$$\mathbf{z} = \mathbf{H}\overline{\mathbf{x}} + \mathbf{w}, \tag{2.1}$$

où $\mathbf{H} \in \mathbb{R}^{M \times N}$ est une matrice modélisant une dégradation (en traitement des images, cet opérateur peut, par exemple, représenter un flou apparaissant lors de la capture de l'image), et $\mathbf{w} \in \mathbb{R}^M$ est une réalisation d'une variable aléatoire $\boldsymbol{w}$, représentant un bruit d'acquisition et/ou une erreur de modèle. L'objectif est alors de produire une estimée $\widehat{\mathbf{x}} \in \mathbb{R}^N$ du signal original $\overline{\mathbf{x}}$ à partir de $\mathbf{z}$ et $\mathbf{H}$. La figure 2.1 donne deux exemples de problèmes inverses. L'exemple du haut correspond à un problème de restauration d'image, et celui du bas correspond à un problème de restauration de maillage 3D.

Une méthode usuelle pour trouver $\widehat{\mathbf{x}}$ est de le définir comme étant un minimiseur d'une somme de deux fonctions [Demoment, 1989] :

$$\widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x} \in \mathbb{R}^N} \ \mathsf{h}(\mathbf{x}) + \mathsf{g}(\mathbf{x}), \tag{2.2}$$

où $\mathsf{h} \colon \mathbb{R}^N \to]-\infty, +\infty]$ est le terme d'attache aux données qui contient les informations directement liées au problème traité (par exemple, un estimateur statistique lié à la loi de $\boldsymbol{w}$), et $\mathsf{g} \colon \mathbb{R}^N \to]-\infty, +\infty]$ est le terme de régularisation (ou de pénalisation) permettant d'introduire des informations que l'on connaît *a priori* sur le signal recherché (par exemple, parcimonie de la solution, contraintes variées).

Comme nous le verrons dans le chapitre 4, dans certains cas, la matrice $\mathbf{H}$ est elle aussi inconnue. Le problème inverse considéré est alors appelé *problème aveugle*, et l'objectif est de

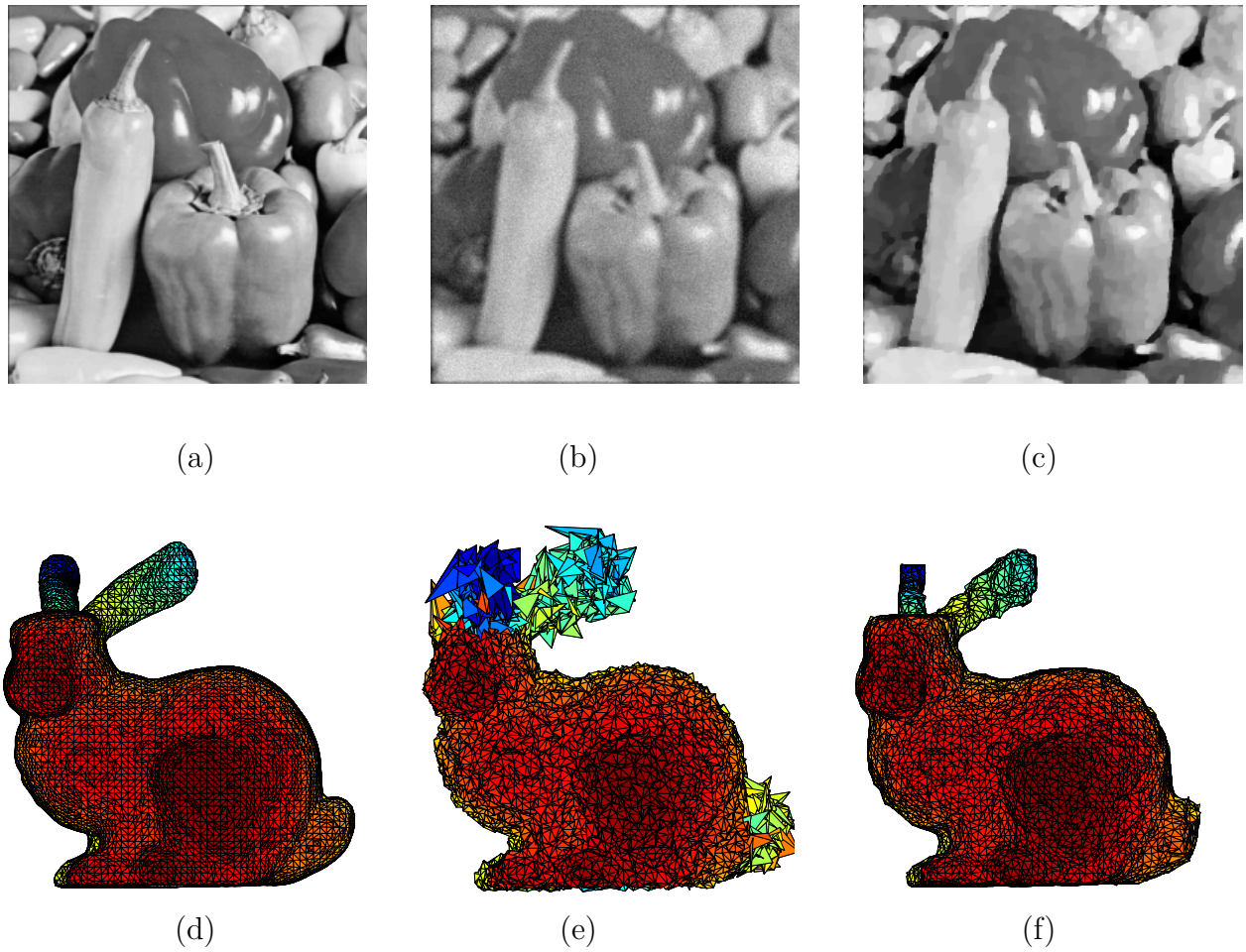


Figure 2.1 – (a) *Image originale* peppers $\overline{\mathbf{x}}$. (b) *Image dégradée* $\mathbf{z}$ *par un flou.* (c) *Image estimé* $\hat{\mathbf{x}}$. (d) *Maillage 3D original* bunny $\overline{\mathbf{x}}$. (e) *Maillage bruité* $\mathbf{z}$. (f) *Maillage estimé* $\hat{\mathbf{x}}$.

reconstruire à la fois $\overline{\mathbf{x}}$ et $\mathbf{H}$ à partir de $\mathbf{z}$. Pour cela, on définit les estimées $(\widehat{\mathbf{x}}, \widehat{\mathbf{H}}) \in \mathbb{R}^N \times \mathbb{R}^{M\times N}$ de $(\overline{\mathbf{x}}, \mathbf{H})$ comme des solutions du problème variationnel :

$$(\widehat{\mathbf{x}}, \widehat{\mathbf{H}}) \in \operatorname*{Argmin}_{(\mathbf{x},\mathbf{H})\in\mathbb{R}^N\times\mathbb{R}^{M\times N}} \mathsf{h}(\mathbf{x}, \mathbf{H}) + \mathsf{g}_1(\mathbf{x}) + \mathsf{g}_2(\mathbf{H}), \tag{2.3}$$

où $\mathsf{h}\colon \mathbb{R}^N \times \mathbb{R}^{M\times N} \to]-\infty, +\infty]$ est le terme de fidélité, et $\mathsf{g}_1\colon \mathbb{R}^N \to]-\infty, +\infty]$ (resp. $\mathsf{g}_2\colon \mathbb{R}^{M\times N} \to]-\infty, +\infty]$) est le terme de régularisation associé au signal $\mathbf{x}$ (resp. à la matrice $\mathbf{H}$). Notons cependant que dans la suite de cette section nous ne nous intéresserons qu'au cas où $\mathbf{H}$ est connue.

Nous allons tout d'abord donner dans la section 2.2.2 une interprétation bayésienne du problème d'estimation (2.1) conduisant à un choix du terme d'attache aux données dans (2.2), et dans la section 2.2.4 nous donnerons quelques exemples de fonctions de régularisation.

Remarque 2.1. *Dans ce manuscrit, nous utiliserons des caractères droits pour représenter les*

variables déterministes et des caractères italiques pour les variables aléatoires. De plus, les lettres en gras représenteront des vecteurs de variables tandis que les lettres fines représenteront généralement des variables scalaires.

Par exemple, le vecteur $\mathbf{x} = (\mathsf{x}^{(n)})_{1\leqslant n\leqslant N} \in \mathbb{R}^N$ *sera une réalisation du vecteur aléatoire* $\boldsymbol{x} = (x^{(n)})_{1\leqslant n\leqslant N}$.

2.2.2 Estimateur du *Maximum A Posteriori*

Dans cette section, nous allons nous intéresser au choix de la fonction d'attache aux données h pour résoudre le problème inverse modélisé par l'équation (2.1). Une approche pour choisir cette fonction est de se baser sur une formulation bayésienne du problème. Supposons que $\mathbf{z}$ et $\overline{\mathbf{x}}$ soient des réalisations de vecteurs aléatoires $\boldsymbol{z}$ et $\overline{\boldsymbol{x}}$ absolument continus. En utilisant une approche de type *Maximum A Posteriori* (MAP), nous pouvons définir une estimée $\widehat{\mathbf{x}}$ de $\overline{\mathbf{x}}$ par

$$\widehat{\mathbf{x}} \in \operatorname*{Argmax}_{\mathbf{x}\in\mathbb{R}^N} \ \mathsf{f}_{\overline{\boldsymbol{x}}|\boldsymbol{z}=\mathbf{z}}(\mathbf{x}), \tag{2.4}$$

où $\mathsf{f}_{\overline{\boldsymbol{x}}|\boldsymbol{z}=\mathbf{z}}$ est la densité de probabilité *a posteriori* de $\overline{\boldsymbol{x}}$ sachant que $\boldsymbol{z} = \mathbf{z}$.

Propriété 2.1. *Soit* $\mathbf{z} \in \mathbb{R}^M$ *défini par l'équation* (2.1). *Supposons que* $\overline{\boldsymbol{x}}$ *est indépendant de* $\boldsymbol{w}$. *Une solution* $\widehat{\mathbf{x}}$ *du problème* (2.4) *est donnée par*

$$\widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x}\in\mathbb{R}^N} \ \mathsf{h}(\mathbf{x}) + \mathsf{g}(\mathbf{x}), \tag{2.5}$$

où $\mathsf{h}\colon \mathbb{R}^N \to]-\infty,+\infty]$ *et* $\mathsf{g}\colon \mathbb{R}^N \to]-\infty,+\infty]$ *sont respectivement les termes d'attache aux données et de régularisation définis par*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \begin{cases} \mathsf{h}(\mathbf{x}) = -\log \mathsf{f}_{\boldsymbol{w}}(\mathbf{z} - \mathbf{H}\mathbf{x}) \\ \mathsf{g}(\mathbf{x}) = -\log \mathsf{f}_{\overline{\boldsymbol{x}}}(\mathbf{x}), \end{cases} \tag{2.6}$$

où $\mathsf{f}_{\overline{\boldsymbol{x}}}$ *est la densité de probabilité de* $\overline{\boldsymbol{x}}$.

Démonstration. Soit $\mathbf{z} \in \mathbb{R}^M$ tel que $\mathsf{f}_{\boldsymbol{z}}(\mathbf{z}) \neq 0$. D'après la règle de Bayes, on a

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{f}_{\overline{\boldsymbol{x}}|\boldsymbol{z}=\mathbf{z}}(\mathbf{x}) = \frac{\mathsf{f}_{\boldsymbol{z}|\overline{\boldsymbol{x}}=\mathbf{x}}(\mathbf{z})\mathsf{f}_{\overline{\boldsymbol{x}}}(\mathbf{x})}{\mathsf{f}_{\boldsymbol{z}}(\mathbf{z})}. \tag{2.7}$$

Ainsi, (2.4) se réécrit

$$\widehat{\mathbf{x}} \in \operatorname*{Argmax}_{\mathbf{x}\in\mathbb{R}^N} \ \mathsf{f}_{\boldsymbol{z}|\overline{\boldsymbol{x}}=\mathbf{x}}(\mathbf{z})\mathsf{f}_{\overline{\boldsymbol{x}}}(\mathbf{x}). \tag{2.8}$$

Par monotonicité de la fonction logarithme, on obtient alors

$$\widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x}\in\mathbb{R}^N} \ \left\{ -\log \mathsf{f}_{\boldsymbol{z}|\overline{\boldsymbol{x}}=\mathbf{x}}(\mathbf{z}) - \log \mathsf{f}_{\overline{\boldsymbol{x}}}(\mathbf{x}) \right\}. \tag{2.9}$$

En utilisant le modèle (2.1), on a $\boldsymbol{w} = \boldsymbol{z} - \mathbf{H}\overline{\boldsymbol{x}}$. Ainsi, pour tout $\mathbf{x} \in \mathbb{R}^N$,

$$(\forall \mathbf{z} \in \mathbb{R}^M) \quad \mathsf{f}_{\boldsymbol{z}|\overline{\boldsymbol{x}}=\mathbf{x}}(\mathbf{z}) = \mathsf{f}_{\boldsymbol{w}|\overline{\boldsymbol{x}}=\mathbf{x}}(\mathbf{w}), \tag{2.10}$$

où $\mathbf{w} = \mathbf{z} - \mathbf{H}\mathbf{x}$. Or, $\overline{\boldsymbol{x}}$ et $\boldsymbol{w}$ étant indépendants, on a

$$(\forall \mathbf{w} \in \mathbb{R}^M) \quad \mathsf{f}_{\boldsymbol{w}|\overline{\boldsymbol{x}}=\mathbf{x}}(\mathbf{w}) = \mathsf{f}_{\boldsymbol{w}}(\mathbf{w}). \tag{2.11}$$

Le résultat s'obtient en combinant les deux dernières inégalités et en les injectant dans (2.9). □

Remarque 2.2. *L'hypothèse d'indépendance entre $\overline{\boldsymbol{x}}$ et $\boldsymbol{w}$ revient à supposer que le signal et le bruit additif sont indépendants entre eux. Cependant, en pratique cette hypothèse n'est pas toujours vérifiée. En particulier, nous verrons au chapitre 4 une application en restauration d'image où le bruit de mesure dépend du signal.*

La propriété 2.1 nous indique qu'en utilisant une interprétation de type MAP, le choix de la fonction d'attache aux données h dépend du type de bruit considéré, tandis que le choix de la fonction de régularisation dépend des caractéristiques des données à estimer.

2.2.3 Fonction d'attache aux données

Bruit gaussien corrélé : Soit $\boldsymbol{w}$ la variable aléatoire modélisant le bruit additif dans (2.1). Plaçons-nous dans le cas où $\boldsymbol{w}$ suit une loi normale d'espérance nulle et de matrice de covariance définie positive $\mathbf{\Gamma} \in \mathbb{R}^{M\times M}$:

$$\boldsymbol{w} \sim \mathcal{N}(\mathbf{0}_M, \mathbf{\Gamma}). \tag{2.12}$$

Sa densité de probabilité est alors donnée par

$$(\forall \mathbf{w} \in \mathbb{R}^M) \quad \mathsf{f}_{\boldsymbol{w}}(\mathbf{w}) = \left(2^M \pi^M \det(\mathbf{\Gamma})\right)^{-1/2} \exp\left(-\frac{1}{2}\mathbf{w}^\top \mathbf{\Gamma}^{-1}\mathbf{w}\right), \tag{2.13}$$

où $\det(\mathbf{\Gamma})$ désigne le déterminant de la matrice $\mathbf{\Gamma}$, et $(\cdot)^\top$ l'opérateur de transposition. D'après la propriété 2.1, la fonction d'attache aux données h correspond à

$$\begin{aligned}(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}(\mathbf{x}) &= -\log \mathsf{f}_{\boldsymbol{w}}(\mathbf{z} - \mathbf{H}\mathbf{x}) \\ &= \frac{1}{2}(\mathbf{y} - \mathbf{H}\mathbf{x})^\top \mathbf{\Gamma}^{-1}(\mathbf{y} - \mathbf{H}\mathbf{x}) + \kappa,\end{aligned} \tag{2.14}$$

où $\kappa = \dfrac{1}{2}\log\left(2^M \pi^M \det(\mathbf{\Gamma})\right) \in \mathbb{R}$ est une constante ne dépendant pas de $\mathbf{x}$. On peut donc choisir comme terme d'attache aux données dans (2.5) la fonction

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}(\mathbf{x}) = \frac{1}{2}(\mathbf{y} - \mathbf{H}\mathbf{x})^\top \mathbf{\Gamma}^{-1}(\mathbf{y} - \mathbf{H}\mathbf{x}). \tag{2.15}$$

Bruit blanc gaussien : On dit que le vecteur aléatoire $\boldsymbol{w}$ modélise un *bruit blanc gaussien* s'il suit une loi normale d'espérance nulle, et ses composantes sont indépendantes, identiquement distribuées, de variance égale à σ^2. On se place donc dans un cas particulier de (2.12) où $\mathbf{\Gamma} = \sigma^2 \mathbf{I}_M$, $\mathbf{I}_M$ étant la matrice identité de $\mathbb{R}^{M\times M}$. Ainsi, la fonction d'attache aux données associée se réduit à

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}(\mathbf{x}) = \frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2. \tag{2.16}$$

2.2.4 Fonction de régularisation

Comme nous l'avons mentionné dans la section 2.2.2, le terme de régularisation g permet de prendre en compte des caractéristiques connues des données à estimer. De manière plus générale, la fonction g peut s'écrire comme une somme de plusieurs fonctions de régularisation :

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}(\mathbf{x}) = \sum_{j=1}^{J} \lambda_j \mathsf{g}_j(\mathbf{x}), \tag{2.17}$$

où, pour tout $j \in \{1, \ldots, J\}$, $\lambda_j \in [0, +\infty]$ est appelé *paramètre de régularisation* et $\mathsf{g}_j \colon \mathbb{R}^N \to]-\infty, +\infty]$. L'utilisation de plusieurs fonctions de régularisation permet d'espérer de meilleurs résultats de reconstruction, puisque chacune d'elles permet de contraindre l'estimée à vérifier une caractéristique particulière.

Les paramètres de régularisation $(\lambda_j)_{1 \leqslant j \leqslant J}$ sont à fixer de façon à obtenir la meilleure qualité de reconstruction possible. Notons que fixer $\lambda_j = 0$, pour tout $j \in \{1, \ldots, J\}$, revient à trouver une solution $\widehat{\mathbf{x}}$ non régularisée :

$$\text{trouver } \widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x} \in \mathbb{R}^N} \mathsf{h}(\mathbf{x}). \tag{2.18}$$

Au contraire, faire tendre λ_j vers $+\infty$, pour tout $j \in \{1, \ldots, J\}$, revient généralement à trouver une solution $\widehat{\mathbf{x}}$ ne prenant en compte que le terme de régularisation :

$$\text{trouver } \widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x} \in \mathbb{R}^N} \mathsf{g}(\mathbf{x}). \tag{2.19}$$

Il faut donc régler les paramètres de régularisation de manière à obtenir un compromis entre le terme d'attache aux données et le terme de régularisation. En pratique, ces paramètres peuvent être réglés de manière à minimiser l'erreur entre les données originales $\overline{\mathbf{x}}$ et leurs estimations $\widehat{\mathbf{x}}$. Par exemple, on peut utiliser l'*erreur quadratique moyenne* (EQM) ou le *rapport signal sur bruit* (RSB) définis par

$$\mathrm{EQM}(\widehat{\mathbf{x}}, \overline{\mathbf{x}}) = \frac{1}{N} \|\widehat{\mathbf{x}} - \overline{\mathbf{x}}\|^2 \quad \text{et} \quad \mathrm{RSB}(\widehat{\mathbf{x}}, \overline{\mathbf{x}}) = 20 \log_{10} \left(\frac{\|\overline{\mathbf{x}}\|}{\|\widehat{\mathbf{x}} - \overline{\mathbf{x}}\|} \right), \tag{2.20}$$

où $\|\cdot\|$ désigne la norme euclidienne usuelle. Dans le cas où les données originales ne sont pas disponibles, on peut avoir recours à des estimateurs d'erreur [Chaux *et al.*, 2008; Deledalle *et al.*, 2013, 2014].

Dans cette section, nous allons donner quelques exemples de fonctions de régularisation souvent rencontrés dans le domaine des problèmes inverses.

2.2.4.1 Contraintes

Le problème de minimisation variationnel (2.2) permet de prendre en compte des contraintes lorsque la fonction de régularisation g correspond à la fonction indicatrice (donnée dans la définition 2.3(viii)) d'un sous-ensemble C de $\mathbb{R}^N$:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}(\mathbf{x}) = \lambda\, \iota_C(\mathbf{x}) = \iota_C(\mathbf{x}), \tag{2.21}$$

(le paramètre de régularisation $\lambda > 0$ est "absorbé" par la fonction indicatrice). Ici, C représente un ensemble de contraintes, qui peut s'écrire comme une intersection de contraintes distinctes. Par exemple, on peut considérer des contraintes de type

$$C = \{\mathbf{x} \in \mathbb{R}^N \mid \mathbf{a} \leqslant \mathsf{L}(\mathbf{x}) \leqslant \mathbf{b}\}, \tag{2.22}$$

où $\mathsf{L}\colon \mathbb{R}^N \to \mathbb{R}^M$ et $(\mathbf{a}, \mathbf{b}) \in [-\infty, +\infty[^M \times]-\infty, +\infty]^M$ [1]. Remarquons qu'en choisissant $\mathbf{a} = \mathbf{b}$, on peut considérer des contraintes d'égalité.

Exemple 2.1. *Pour une image en niveaux de gris codée sur* 8 *bits, l'intensité de chaque pixel est comprise entre* 0 *et* 255. *Ainsi, si la variable* $\mathbf{x} \in \mathbb{R}^N$ *représente une image réorganisée sous forme de vecteur, on peut utiliser la fonction de régularisation* $\mathsf{g} = \iota_{[0,255]^N}$. *Cela correspond à la contrainte présentée ci-dessus où* L *est la fonction identité et, pour tout* $n \in \{1, \ldots, N\}$, $(\mathsf{a}^{(n)}, \mathsf{b}^{(n)}) = (0, 255)$.

En utilisant des contraintes pour régulariser l'estimée, nous n'avons plus besoin d'optimiser le paramètre de régularisation, ce qui peut être utile dans les cas où nous n'avons pas accès aux estimateurs de type (2.20). Cependant, il faut connaître une estimation des paramètres $(\mathbf{a}, \mathbf{b})$.

2.2.4.2 Parcimonie

Un vecteur $\mathbf{x} \in \mathbb{R}^N$ est dit *parcimonieux* si la plupart de ses composantes sont (approximativement) nulles.

Certains signaux sont parcimonieux dans leur domaine d'observation, par exemple les signaux sismiques sont parcimonieux dans le domaine temporel (c.f. figure 2.2). D'autres signaux

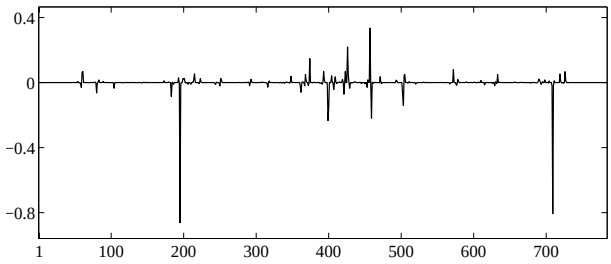


Figure 2.2 – *Signal sismique parcimonieux.*

sont parcimonieux après une représentation linéaire appropriée, c'est-à-dire après avoir subi une transformation par un opérateur. Dans ce cas, on considèrera une fonction de régularisation de la forme

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}(\mathbf{x}) = \sum_{s=1}^{S} \mathsf{g}_s(\mathbf{F}_s \mathbf{x}), \tag{2.23}$$

où, pour tout $s \in \{1, \ldots, S\}$, $\mathbf{F}_s \in \mathbb{R}^{M_s \times N}$ et $\mathsf{g}_s\colon \mathbb{R}^{M_s} \to]-\infty, +\infty]$. Ici, l'objet $\mathbf{x} \in \mathbb{R}^N$ est supposé parcimonieux après application des matrices $(\mathbf{F}_s)_{1 \leqslant s \leqslant S}$, c'est-à-dire que, pour tout $s \in \{1, \ldots, S\}$, $\mathbf{F}_s\mathbf{x}$ a un nombre important de coefficients égaux à zéros.

1. La notation $\mathbf{a} \leqslant \mathsf{L}(\mathbf{x}) \leqslant \mathbf{b}$ signifie que $\mathsf{L}(\mathbf{x}) - \mathbf{a} \in [0, +\infty[^M$ et $\mathbf{b} - \mathsf{L}(\mathbf{x}) \in [0, +\infty[^M$.

Exemples de pénalisations renforçant la parcimonie : Il existe plusieurs choix de fonctions g_s permettant de promouvoir la parcimonie d'un objet estimé. Nous en donnerons ici quelques exemples. Intuitivement, la fonction permettant de favoriser au mieux la parcimonie d'un vecteur $\mathbf{x} \in \mathbb{R}^N$ est la "pseudo-norme" ℓ_0 [Donoho *et al.*, 1995] qui compte le nombre de coefficients non nuls de $\mathbf{x}$:

$$(\forall \mathbf{x} = (\mathsf{x}^{(n)})_{1 \leqslant n \leqslant N} \in \mathbb{R}^N) \quad \ell_0(\mathbf{x}) = \|\mathbf{x}\|_0 = \sum_{n=1}^{N} \chi(\mathsf{x}^{(n)}), \tag{2.24}$$

où la fonction $\chi \colon \mathbb{R} \to \{0, 1\}$ est définie par

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \chi(\mathsf{x}) = \begin{cases} 0 & \text{si } \mathsf{x} = 0, \\ 1 & \text{sinon.} \end{cases} \tag{2.25}$$

De façon plus générale, les fonctions ℓ_α [Bouman et Sauer, 1996], pour $0 \leqslant \alpha < 1$, sont des fonctions de régularisation permettant de promouvoir la parcimonie d'un objet :

$$(\forall \mathbf{x} = (\mathsf{x}^{(n)})_{1 \leqslant n \leqslant N} \in \mathbb{R}^N) \quad \ell_\alpha(\mathbf{x}) = \|\mathbf{x}\|_\alpha = \left(\sum_{n=1}^{N} |\mathsf{x}^{(n)}|^\alpha \right)^{1/\alpha}, \tag{2.26}$$

quand $\alpha \neq 0$. Cependant, ces fonctions ne sont ni différentiables, ni convexes. Elles peuvent donc se révéler difficiles à optimiser en pratique. Une façon usuelle de réduire la difficulté du problème est d'utiliser la norme ℓ_1 [Bect *et al.*, 2004; Donoho, 2006; Figueiredo *et al.*, 2007] comme approximation convexe des pénalisations ci-dessus :

$$(\forall \mathbf{x} = (\mathsf{x}^{(n)})_{1 \leqslant n \leqslant N} \in \mathbb{R}^N) \quad \ell_1(\mathbf{x}) = \|\mathbf{x}\|_1 = \sum_{n=1}^{N} |\mathsf{x}^{(n)}|. \tag{2.27}$$

La figure 2.3 représente différentes fonctions ℓ_α, pour $0 \leqslant \alpha \leqslant 1$. Nous pouvons voir que plus α tend vers 0, moins les coefficient proches de 0 seront pénalisés.

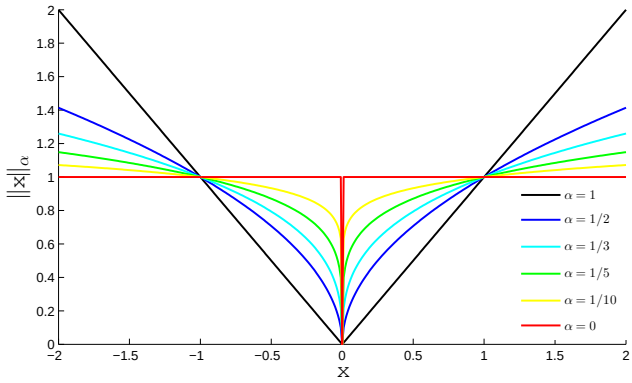


Figure 2.3 – *Fonctions ℓ_α pour $\alpha \in \{0, 1/10, 1/5, 1/3, 1/2, 1\}$.*

Bien que l'utilisation de la norme ℓ_1 permette de s'affranchir de la non-convexité de la régularisation, cette fonction est non différentiable. Dans certains cas pratiques, il peut être plus

avantageux d'utiliser des approximations lisses de la fonction ℓ_1, comme par exemple la fonction de Huber [Huber, 1981] définie par $\mathbf{x} = (\mathsf{x}^{(n)})_{1 \leqslant n \leqslant N} \in \mathbb{R}^N \mapsto \mathsf{g}(\mathbf{x}) = \sum_{n=1}^{N} \varphi_\delta(\mathsf{x}^{(n)})$, où $\delta > 0$ et

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \varphi_\delta(\mathsf{x}) = \begin{cases} \frac{\mathsf{x}^2}{2\delta} & \text{si } |\mathsf{x}| \leqslant \delta, \\ |\mathsf{x}| - \frac{\delta}{2} & \text{sinon.} \end{cases} \tag{2.28}$$

Enfin, des fonctions de type quotient ou différence de normes peuvent aussi être utilisées pour favoriser la parcimonie de données. Ces fonction ne sont, généralement, ni convexes ni différentiables. Dans le chapitre 6, nous définirons une approximation lisse mais non convexe de la pénalisation ℓ_1/ℓ_2, et nous l'utiliserons dans une application de déconvolution aveugle de signaux sismiques.

Variation totale : Une des fonctions de régularisation les plus utilisées en traitement des images est la *Variation Totale* introduite dans [Rudin *et al.*, 1992] pour des données continues. La version discrète proposée notamment dans [Chambolle, 2004] est définie par :

Définition 2.1. *Soit $\mathbf{x} \in \mathbb{R}^{N_1 \times N_2}$ une matrice modélisant une image de dimension $N = N_1 \times N_2$. La variation totale (TV) de $\mathbf{x}$ est donnée par*

$$\mathsf{g}(\mathbf{x}) = \mathrm{tv}(\mathbf{x}) = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \sqrt{\left([\nabla_v \mathbf{x}]^{(i,j)}\right)^2 + \left([\nabla_h \mathbf{x}]^{(i,j)}\right)^2}, \tag{2.29}$$

où $\nabla_v \mathbf{x} \in \mathbb{R}^{N_1 \times N_2}$ et $\nabla_h \mathbf{x} \in \mathbb{R}^{N_1 \times N_2}$ sont les gradients verticaux et horizontaux de $\mathbf{x} = (\mathsf{x}^{(i,j)})_{1 \leqslant i \leqslant N_1,\, 1 \leqslant j \leqslant N_2}$ définis par

$$(\forall (i,j) \in \{1, \ldots, N_1\} \times \{1, \ldots, N_2\}) \quad [\nabla_v \mathbf{x}]^{(i,j)} = \begin{cases} \mathsf{x}^{(i+1,j)} - \mathsf{x}^{(i,j)} & \textit{si } i < N_1, \\ 0 & \textit{sinon,} \end{cases}$$

$$[\nabla_h \mathbf{x}]^{(i,j)} = \begin{cases} \mathsf{x}^{(i,j+1)} - \mathsf{x}^{(i,j)} & \textit{si } j < N_2, \\ 0 & \textit{sinon.} \end{cases}$$

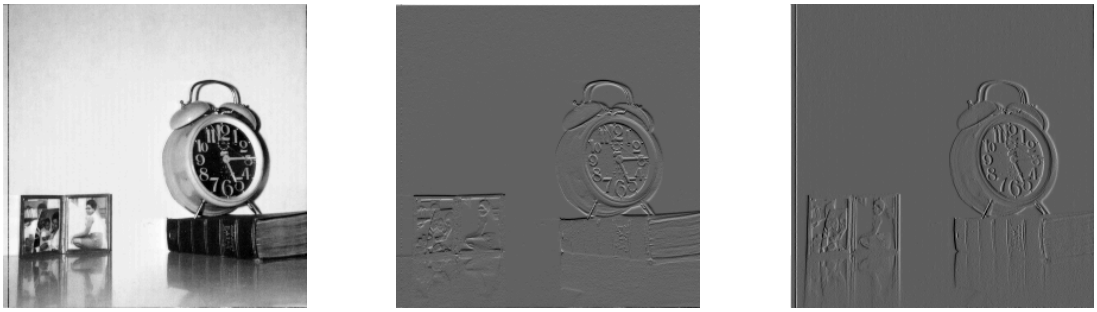
Remarque 2.3. *Dans la définition 2.1 nous avons fait l'hypothèse que les coefficients se trouvant sur les bords droit et inférieur de l'image sont égaux à 0. Cependant, d'autres choix sont possibles pour la valeur de ces coefficients (par exemple, l'hypothèse de périodicité ou de symétrie miroir).*

Un exemple de gradients verticaux et horizontaux d'une image en niveaux de gris est donné dans la figure 2.4.

Notons que l'équation (2.29) peut se réécrire de la façon suivante :

$$(\forall \mathbf{x} \in \mathbb{R}^{N_1 \times N_2}) \quad \mathsf{g}(\mathbf{x}) = \sum_{n=1}^{N} \mathsf{g}_n(\mathbf{F}_n \mathbf{x}), \tag{2.30}$$

où $N = N_1 \times N_2$, et, pour tout $n \in \{1, \ldots, N\}$, $\mathbf{F}_n = [\nabla_v, \nabla_h]^\top$. Comme il l'est souligné dans [Chouzenoux *et al.*, 2013], diverses fonctions $(\mathsf{g}_n)_{1 \leqslant n \leqslant N}$ peuvent être considérées pour pénaliser plus ou moins la parcimonie des gradients de l'image.



(a) (b) (c)

Figure 2.4 – (a) *Image originale* clock *de dimension* $N_1 = N_2 = 256$, (b) *gradients verticaux, et* (c) *gradients horizontaux.*

La variation totale a été généralisée à l'emploi de *gradients non locaux*. Cette méthode a été introduite dans [Gilboa et Osher, 2008], et dans ce cas, on parle de *Variation Totale Non Locale* (NLTV). La principale différence entre la TV et la NLTV réside dans la définition des voisins d'un pixel. Pour la TV, les voisins d'un pixel d'indice $(i,j) \in \{2,\dots,N_1-1\} \times \{2,\dots,N_2-1\}$ sont les pixels d'indices

$$\begin{array}{ll} (i-1,j) & (i+1,j) \\ (i,j-1) & (i,j+1). \end{array}$$

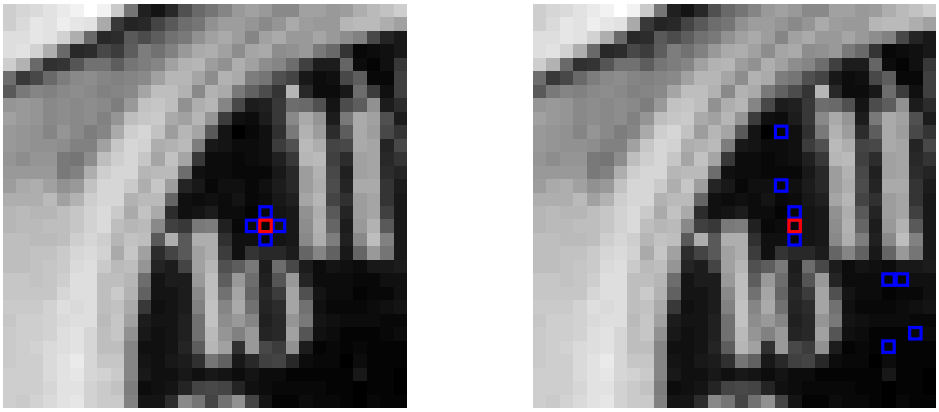
Tandis que pour la NLTV, les voisins d'un pixel sont définis selon leur degré de similarité. Une illustration est donnée dans la figure 2.5. Ainsi, on peut utiliser la fonction de régularisation (2.30), avec, pour tout $n \in \{1,\dots,N\}$, $\mathbf{F}_n = [\omega_n^1 \mathbf{F}_n^1, \dots, \omega_n^{M_n} \mathbf{F}_n^{M_n}]^\top$ où M_n détermine le nombre d'orientations considérées au n-ème pixel et les $(\omega_n^m)_{1 \leqslant m \leqslant M_n}$ sont des poids à fixer dans une phase de prétraitement. Il a été montré expérimentalement que cette régularisation permettait d'améliorer les résultats de reconstruction dans certains cas de figure [Chierchia *et al.*, 2013a; Peyré, 2011; Werlberger *et al.*, 2010].

Opérateurs de trames : On considère le cas où la fonction de régularisation s'écrit sous la forme (2.23), où pour tout $s \in \{1,\dots,S\}$, $\mathbf{F}_s = \mathbf{F}$ est un opérateur de trame défini par

$$\mathbf{F} : \mathbb{R}^N \to \mathbb{R}^K : \mathbf{x} \mapsto \mathbf{F}\mathbf{x} = (\langle \mathbf{x}, \mathbf{e}_k \rangle)_{1 \leqslant k \leqslant K}, \tag{2.31}$$

où $(\mathbf{e}_k)_{1 \leqslant k \leqslant K}$ est un dictionnaire de signaux de $\mathbb{R}^N$ constituant une trame [Mallat, 2009, Chap. 5], i.e., vérifiant la condition suivante :

$$(\exists (\underline{\mu}, \overline{\mu}) \in]0,+\infty[^2) \text{ tel que } (\forall \mathbf{x} \in \mathbb{R}^N) \quad \underline{\mu}\|\mathbf{x}\|^2 \leqslant \sum_{k=1}^{K} |\langle \mathbf{x}, \mathbf{e}_k \rangle|^2 \leqslant \overline{\mu}\|\mathbf{x}\|^2. \tag{2.32}$$



(a) (b)

Figure 2.5 – *Détail de l'image* clock. (a) *Voisins locaux (carrés bleus) verticaux et horizontaux d'un pixel donné (carré rouge).* (b) *Voisins non locaux (carrés bleus) du même pixel (carré rouge). Dans le deuxième cas, les voisins sont sélectionnés de façon à ce que l'intensité des pixels bleus soit similaire à l'intensité du pixel rouge.*

Notons que dans $\mathbb{R}^N$, l'existence de la borne supérieure est toujours garantie, ce qui n'est pas toujours le cas en dimension infinie. Lorsque $\underline{\mu} = \overline{\mu}$, on dit que la trame est *ajustée*. L'adjoint de l'opérateur de trame est donné par

$$\mathsf{F}^* \colon \mathbb{R}^K \to \mathbb{R}^N \colon \mathsf{y} = (\mathsf{y}^{(k)})_{1 \leqslant k \leqslant K} \mapsto \mathsf{F}^*\mathsf{y} = \sum_{k=1}^{K} \mathsf{y}^{(k)} \mathsf{e}_k. \tag{2.33}$$

Soit $\mathsf{x} \in \mathbb{R}^N$ un signal donné, on notera alors $\mathsf{y} = \mathsf{F}\mathsf{x}$ ses coefficient de trames.

Des cas particuliers des opérateurs de trames sont les *opérateurs d'ondelettes*. Un exemple de transformée en ondelettes, utilisant une ondelette de Haar est donné dans la figure 2.6. Plus précisément, la figure 2.6(b) représente une décomposition sur une base d'ondelettes de l'image *clock* (figure 2.6(a)) sur un niveau de résolution. Les coefficients de cette décomposition se trouvant dans la partie supérieure gauche sont appelés *coefficients d'approximations* de la figure 2.6(a), tandis que le reste de l'image est composée de *coefficients de détails*. Ce sont les coefficients de détails qui sont les plus parcimonieux. La figure 2.6(c) représente aussi une décomposition en ondelettes de l'image mais sur deux niveaux de résolution, i.e. la décomposition est itérée une seconde fois sur les coefficients d'approximations.

Notons que l'approche que nous avons décrite ici s'appelle *approche à l'analyse*, c'est à dire que l'on cherche à estimer directement l'image via le problème variationnel :

$$\text{trouver } \widehat{\mathsf{x}} \in \operatorname*{Argmin}_{\mathsf{x} \in \mathbb{R}^N} \ \mathsf{h}(\mathsf{x}) + \mathsf{g}(\mathsf{F}\mathsf{x}). \tag{2.34}$$

(a) (b) (c)

Figure 2.6 – (a) *Image originale* clock. *Décomposition sur une base d'ondelettes sur 1 niveau de résolution* (b) *et sur 2 niveaux de résolution* (c) *avec une ondelette de Haar.*

Nous verrons un exemple de cette approche dans le chapitre 4. Une autre façon d'utiliser des trames est de considérer une *approche à la synthèse*. Dans ce cas, on cherche à estimer les coefficients de trame en résolvant le problème variationnel :

$$\text{trouver } \widehat{\mathbf{y}} \in \underset{\mathbf{y}\in\mathbb{R}^N}{\text{Argmin}} \ \mathsf{h}(\mathbf{F}^*\mathbf{y}) + \mathsf{g}(\mathbf{y}), \tag{2.35}$$

où $\widehat{\mathbf{y}} = \mathsf{F}\widehat{\mathbf{x}}$ sont les coefficients de trame de l'image estimée. L'image est ensuite déduite en utilisant l'adjoint de l'opérateur de trame $\widehat{\mathbf{x}} = \mathbf{F}^*\widehat{\mathbf{y}}$. Un exemple de cette approche sera présenté dans le chapitre 6.

2.3 Outils d'analyse variationnelle

2.3.1 Notations et définitions

Nous allons, dans un premier temps, donner nos notations et rappeler des définitions d'analyse qui nous seront utiles dans la suite de ce manuscrit.

Soit $\mathcal{H}$ un espace de Hilbert réel, muni d'un produit scalaire noté $\langle\cdot,\cdot\rangle$ et d'une norme notée $\|\cdot\| = \sqrt{\langle\cdot,\cdot\rangle}$. Nous noterons $\mathcal{B}(\mathcal{H},\mathcal{G})$ l'ensemble des opérateurs linéaires bornés de $\mathcal{H}$ dans $\mathcal{G}$, où $\mathcal{G}$ est un espace de Hilbert réel, $\mathcal{S}(\mathcal{H})$ l'ensemble des opérateur linéaires auto-adjoints [2] bornés de $\mathcal{H}$ dans $\mathcal{H}$, et $\mathcal{S}^+(\mathcal{H})$ l'ensemble des opérateurs de $\mathcal{S}(\mathcal{H})$ fortement positifs [3]. Remarquons que si $\mathcal{H} = \mathbb{R}^N$, alors $\mathcal{S}_N := \mathcal{S}(\mathbb{R}^N)$ désigne l'ensemble des matrices symétriques de $\mathbb{R}^{N\times N}$, et

2. $\mathbf{U} \in \mathcal{B}(\mathcal{H}) = \mathcal{B}(\mathcal{H},\mathcal{H})$ est auto-adjoint si $\mathbf{U}^* = \mathbf{U}$.
3. $\mathbf{U} \in \mathcal{S}(\mathcal{H})$ est fortement positif s'il existe $\alpha \in]0,+\infty[$ tel que $(\forall \mathbf{x} \in \mathcal{H})\ \langle \mathbf{x} \mid \mathbf{U}\mathbf{x}\rangle \geqslant \alpha\|\mathbf{x}\|^2$.

$\mathcal{S}_N^+ := \mathcal{S}^+(\mathbb{R}^N)$ correspond à l'ensemble des matrices symétriques définies positives (notées SDP) de $\mathbb{R}^{N\times N}$. De plus, notons $\mathbf{Id}$ l'opérateur identité de $\mathcal{H}$, et $\mathbf{I}_N$ la matrice identité de $\mathbb{R}^{N\times N}$. On définira l'ordre partiel de Loewner sur $\mathcal{H}$ comme suit :

Définition 2.2. *Soient* $\mathbf{U}_1$ *et* $\mathbf{U}_2$ *deux opérateurs de* $\mathcal{S}(\mathcal{H})$*. L'ordre partiel de Loewner sur* $\mathcal{H}$ *vérifie*

$$\mathbf{U}_1 \succcurlyeq \mathbf{U}_2 \quad \Leftrightarrow \quad (\forall \mathbf{x} \in \mathcal{H}) \quad \langle \mathbf{x} \mid \mathbf{U}_1\mathbf{x}\rangle \geqslant \langle \mathbf{x} \mid \mathbf{U}_2\mathbf{x}\rangle.$$

Nous noterons le produit scalaire et la norme pondérés, relativement à $\mathbf{U} \in \mathcal{S}^+(\mathcal{H})$, de la façon suivante :

$$(\forall \mathbf{x} \in \mathcal{H})\,(\forall \mathbf{y} \in \mathcal{H}) \quad \langle \mathbf{x} \mid \mathbf{y}\rangle_{\mathbf{U}} := \langle \mathbf{x} \mid \mathbf{U}\mathbf{y}\rangle \quad \text{et} \quad \|\mathbf{x}\|_{\mathbf{U}} := \langle \mathbf{x} \mid \mathbf{U}\mathbf{x}\rangle^{1/2}. \tag{2.36}$$

Remarquons que lorsque $\mathbf{U}$ est égal à $\mathbf{Id}$, alors $\|\cdot\|_{\mathbf{Id}} = \|\cdot\|$.

Définition 2.3. *Soit* $\mathsf{h}\colon \mathcal{H} \to]-\infty, +\infty]$.

(i) *Le* domaine *de la fonction* h *est défini par*

$$\operatorname{dom} \mathsf{h} := \{\mathbf{x} \in \mathcal{H} \mid \mathsf{h}(\mathbf{x}) < +\infty\}.$$

(ii) *La fonction* h *est dite* propre *si* $\operatorname{dom}\mathsf{h}$ *est non vide.*

(iii) *Soit* $\delta \in \mathbb{R}$*. L'ensemble des* lignes de niveau δ *de* h *est donné par*

$$\operatorname{lev}_{\leqslant\delta} \mathsf{h} := \{\mathbf{x} \in \mathcal{H} \mid \mathsf{h}(\mathbf{x}) \leqslant \delta\}.$$

(iv) *La fonction* h *est dite* coercive *si*

$$\lim_{\|\mathbf{x}\|\to+\infty} \mathsf{h}(\mathbf{x}) = +\infty.$$

(v) *Le* graphe *de* h *est défini par*

$$\operatorname{Graph} \mathsf{h} := \{(\mathbf{x}, \xi) \in \mathcal{H} \times \mathbb{R} \mid \xi = \mathsf{h}(\mathbf{x})\}.$$

(vi) *L'*épigraphe *de* h *est défini par*

$$\operatorname{epi} \mathsf{h} := \{(\mathbf{x}, \xi) \in \mathcal{H} \times \mathbb{R} \mid \mathsf{h}(\mathbf{x}) \leqslant \xi\}.$$

(vii) *La fonction* h *est* semi-continue inférieurement *(s.c.i) en* $\mathbf{x}_0 \in \mathcal{H}$ *si*

$$\liminf_{\mathbf{x}\to\mathbf{x}_0} \mathsf{h}(\mathbf{x}) \geqslant \mathsf{h}(\mathbf{x}_0).$$

La fonction h *est* semi-continue inférieurement *si elle est s.c.i en tout point* $\mathbf{x}_0 \in \mathcal{H}$*.*

(viii) *Soit E un sous-ensemble de $\mathcal{H}$. La fonctions indicatrice $\iota_E \colon \mathcal{H} \to]-\infty, +\infty]$ de cet ensemble est définie par*

$$(\forall \mathbf{x} \in \mathcal{H}) \quad \iota_E(\mathbf{x}) = \begin{cases} 0 & \text{si } \mathbf{x} \in E, \\ +\infty & \text{sinon.} \end{cases}$$

(ix) *Soit $\mathsf{g} \colon \mathcal{H} \to]-\infty, +\infty]$. L'*infimale convolution *de h et g est définie par*

$$\mathsf{h} \,\square\, \mathsf{g} \colon \mathcal{H} \to [-\infty, +\infty] \colon \mathbf{x} \mapsto \inf_{\mathbf{y} \in \mathcal{H}} \left(\mathsf{h}(\mathbf{y}) + \mathsf{g}(\mathbf{x} - \mathbf{y})\right).$$

L'élément neutre de l'infimale convolution est la fonction indicatrice de l'ensemble $\{\mathbf{0}\}$, où $\mathbf{0}$ représente le vecteur nul de $\mathcal{H}$:

$$(\forall \mathbf{x} \in \mathcal{H}) \quad \mathsf{h} \,\square\, \iota_{\{\mathbf{0}\}}(\mathbf{x}) = \mathsf{h}(\mathbf{x}).$$

Une représentation graphique des définitions 2.3(i) et (iii) est donnée dans la figure 2.7. De même, une représentation graphique de la définition 2.3(vi) est donnée dans la figure 2.8.

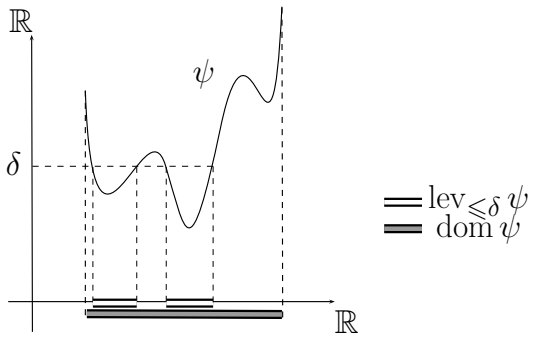


Figure 2.7 – *Illustration du domaine et de l'ensemble des lignes de niveau $\delta > 0$ d'une fonction $\psi \colon \mathbb{R} \to \mathbb{R}$.*

Lorsqu'une fonction associe à un point un ensemble, elle est dite *multivaluée*. La définition suivante généralise la notion de graphe pour ce type de fonctions.

Définition 2.4. *Soit $\mathsf{L} \colon \mathcal{H} \to 2^{\mathcal{G}}$ une fonction multivaluée où $\mathcal{G}$ est un ensemble de Hilbert séparable réel. Le* graphe *de L est défini par*

$$\operatorname{Graph} \mathsf{L} := \{(\mathbf{x}, \mathbf{y}) \in \mathcal{H} \times \mathcal{G} \mid \mathbf{y} \in \mathsf{L}(\mathbf{x})\}.$$

Les deux théorèmes suivants mettent en avant des relations existantes entre certaines des notions données dans la définition 2.3.

Théorème 2.1. [Rockafellar, 1970, Thm. 1.6][Bauschke et Combettes, 2011, Lem. 1.24] *Soit $\mathsf{h} \colon \mathcal{H} \to]-\infty, +\infty]$. La fonction h est semi-continue inférieurement si et seulement si, pour tout $\delta \in \mathbb{R}$, les ensembles $\operatorname{lev}_{\leqslant \delta} \mathsf{h}$ sont fermés dans $\mathcal{H}$. De plus, si h est coercive, alors $\operatorname{lev}_{\leqslant \delta} \mathsf{h}$ est un ensemble borné.*

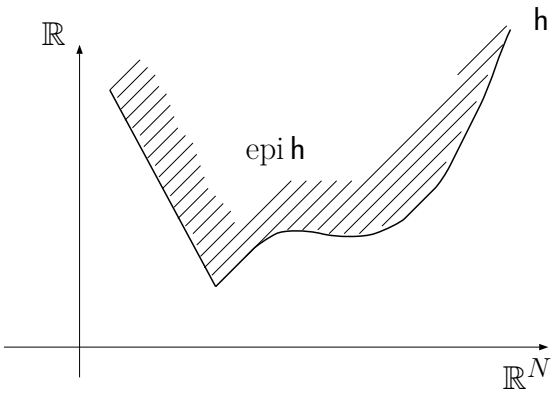


Figure 2.8 – *Illustration de l'épigraphe d'une fonction* $\mathsf{h}\colon \mathbb{R}^N \to \mathbb{R}$.

Théorème 2.2. [Rockafellar, 1970, Thm. 1.9] *Soit* $\mathsf{h}\colon \mathbb{R}^N \to]-\infty,+\infty]$ *une fonction propre, semi-continue inférieurement et coercive. L'ensemble des minimiseurs de* h *est non vide et compact.*

La définition suivante introduit la notion de conjuguée d'une fonction. En particulier, cette notion peut se révéler utile lors de la résolution de certains problèmes inverses.

Définition 2.5. [Bauschke et Combettes, 2011; Rockafellar, 1974] *Soit* $\mathsf{h}\colon \mathcal{H} \to]-\infty,+\infty]$ *une fonction propre. La* conjuguée *(de Legendre-Fenchel)* $\mathsf{h}^*\colon \mathcal{H} \to]-\infty,+\infty]$ *de la fonction* h *est définie par*

$$(\forall \mathbf{x} \in \mathcal{H}) \quad \mathsf{h}^*(\mathbf{x}) := \sup_{\mathbf{y}\in\mathcal{H}} \langle \mathbf{y} \mid \mathbf{x} \rangle - \mathsf{h}(\mathbf{y}).$$

Remarque 2.4. [Bauschke et Combettes, 2011; Komodakis et Pesquet, 2014] *Soient* $\mathsf{h}\colon \mathcal{H} \to]-\infty,+\infty]$, $\mathsf{g}\colon \mathcal{G} \to]-\infty,+\infty]$, *où* $\mathcal{G}$ *est un ensemble de Hilbert réel, et* $\mathbf{L} \in \mathcal{B}(\mathcal{H},\mathcal{G})$. *On appellera* problème primal *le problème de minimisation*

$$\text{trouver } \widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x}\in\mathcal{H}} \ \mathsf{h}(\mathbf{x}) + \mathsf{g}(\mathbf{L}\mathbf{x}), \tag{2.37}$$

et problème dual

$$\text{trouver } \widehat{\mathbf{v}} \in \operatorname*{Argmin}_{\mathbf{v}\in\mathcal{G}} \ \mathsf{h}^*(-\mathbf{L}^*\mathbf{v}) + \mathsf{g}^*(\mathbf{v}). \tag{2.38}$$

Les solutions du problème (2.37) *sont alors appelées solutions primales, et les solutions du problème* (2.4) *sont les solutions duales.*

2.3.2 Analyse convexe

Dans cette section, nous nous intéresserons plus particulièrement à une sous-discipline de l'analyse variationnelle, fondamentale en optimisation, qui est l'analyse convexe.

Définition 2.6. [Bauschke et Combettes, 2011, Déf. 3.1] *Soit C un sous-ensemble de $\mathcal{H}$. L'ensemble C est* convexe *si*

$$(\forall \lambda \in]0,1[)(\forall (\mathbf{x},\mathbf{y}) \in C^2) \quad \lambda\mathbf{x} + (1-\lambda)\mathbf{y} \in C.$$

Définition 2.7. *Une fonction $\mathsf{f} \colon \mathcal{H} \to]-\infty,+\infty]$ est dite*

(i) [Bauschke et Combettes, 2011, Prop. 8.4] convexe *si elle vérifie*

$$(\forall \lambda \in]0,1[)\,(\forall (\mathbf{x},\mathbf{y}) \in (\operatorname{dom}\mathsf{f})^2) \quad \mathsf{f}(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leqslant \lambda\mathsf{f}(\mathbf{x}) + (1-\lambda)\mathsf{f}(\mathbf{y}).$$

(ii) [Bauschke et Combettes, 2011, Déf. 8.6] strictement convexe *si elle vérifie*

$$(\forall \lambda \in]0,1[)\,(\forall (\mathbf{x},\mathbf{y}) \in (\operatorname{dom}\mathsf{f})^2) \quad \mathbf{x} \neq \mathbf{y} \quad \Rightarrow \quad \mathsf{f}(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) < \lambda\mathsf{f}(\mathbf{x}) + (1-\lambda)\mathsf{f}(\mathbf{y}).$$

(iii) [Bauschke et Combettes, 2011, Déf. 10.5] fortement convexe *de constante β si $\mathsf{f} - (\beta/2)\|\cdot\|^2$ est convexe.*

(iv) [Bauschke et Combettes, 2011, Déf. 10.18] quasi-convexe *si ses lignes de niveaux $\left(\operatorname{lev}_{\leqslant\xi}\mathsf{f}\right)_{\xi\in\mathbb{R}}$ sont convexes.*

Remarque 2.5. *Si $\mathsf{f} \colon \mathcal{H} \to]-\infty,+\infty]$ est une fonction convexe, alors son domaine est convexe.*

Dans la suite nous noterons $\Gamma_0(\mathcal{H})$ l'ensemble des fonctions de $\mathcal{H}$ dans $]-\infty,+\infty]$ convexes, propres et semi-continues inférieurement.

Proposition 2.1. [Bauschke et Combettes, 2011] *Si $\varphi \in \Gamma_0(\mathcal{H})$, alors les assertions suivantes sont vérifiées :*

(i) $\operatorname{epi}\varphi$ *est un ensemble non vide, convexe et fermé.*

(ii) $\varphi^* \in \Gamma_0(\mathcal{H})$ *et* $\varphi^{**} = (\varphi^*)^* = \varphi$.

La proposition suivante nous permet d'assurer l'existence de minimiseurs d'une somme de fonctions de $\Gamma_0(\mathcal{H})$.

Proposition 2.2. [Bauschke et Combettes, 2011, Cor. 11.15] *Soient f et g des fonctions de $\Gamma_0(\mathcal{H})$. Supposons que $\operatorname{dom}\mathsf{f} \cap \operatorname{dom}\mathsf{g} \neq \varnothing$, que f est coercive et que g est bornée inférieurement. La somme $\mathsf{f}+\mathsf{g}$ est alors coercive et admet un minimiseur sur $\mathcal{H}$.*

Notons qu'il existe d'autres conditions d'existence de minimiseurs de fonctions convexe, données par exemple dans [Bauschke et Combettes, 2011, Sec. 11.4] et dans [Combettes, 2013, Prop. 5.3].

Définition 2.8. [Bauschke et Combettes, 2011, Déf. 16.1] [Rockafellar, 1970, Sec. 23] *Soit $\mathsf{f} \in \Gamma_0(\mathcal{H})$. Le* sous-différentiel de Moreau *de f en $\mathbf{x} \in \operatorname{dom}\mathsf{f}$ est défini par*

$$\partial\mathsf{f}(\mathbf{x}) := \left\{ \mathbf{t} \in \mathcal{H} \;\middle|\; (\forall \mathbf{y} \in \mathcal{H}) \quad \mathsf{f}(\mathbf{y}) - \langle \mathbf{y}-\mathbf{x} \mid \mathbf{t} \rangle \geqslant \mathsf{f}(\mathbf{x}) \right\}$$

Ainsi, les minimiseurs globaux d'une fonction de $\Gamma_0(\mathcal{H})$ peuvent être caractérisés par la *règle de Fermat* :

Théorème 2.3. [Bauschke et Combettes, 2011, Thm. 16.2] *Soit* $\mathsf{f} \in \Gamma_0(\mathcal{H})$. *L'ensemble des minimiseurs globaux de* f *vérifie alors*

$$\operatorname{Argmin}\ \mathsf{f} = \{\mathbf{x} \in \mathcal{H} \mid \mathbf{0} \in \partial \mathsf{f}(\mathbf{x})\}.$$

Comme nous l'avons vu dans la section 2.2, la résolution de problèmes inverses passe souvent par la minimisation d'une fonction f. Le théorème ci-dessus nous permet donc d'établir un lien entre la solution de ce problème et le sous-différentiel de f. Nous allons cependant voir que d'autres définitions du sous-différentiel sont possibles.

2.3.3 Calcul sous-différentiel

Dans la section 2.2 nous avons vu que les fonctions apparaissant dans les problèmes variationnels associés aux problèmes inverses peuvent être non différentiables et/ou non convexes. Dans cette section, nous donnerons donc les notions de base de sous-différentiabilité pour les fonctions non nécessairement convexes. Ainsi nous pourrons faire le lien entre les points critiques d'une fonction et son sous-différentiel.

2.3.3.1 Sous-différentiel

Nous considérerons dans cette section que $\mathcal{H} = \mathbb{R}^N$.

Définition 2.9. [Rockafellar et Wets, 1997, Def. 8.3],[Mordukhovich, 2006, Sec.1.3] *Soit* $\mathsf{f} \colon \mathbb{R}^N \to]-\infty, +\infty]$ *une fonction propre et soit* $\mathbf{x} \in \operatorname{dom} \mathsf{f}$.

- *Le* sous-différentiel de Fréchet *(ou sous-différentiel régulier) de* f *en* $\mathbf{x}$ *correspond à l'ensemble suivant*

$$\widehat{\partial}\mathsf{f}(\mathbf{x}) := \left\{ \hat{\mathbf{t}} \in \mathbb{R}^N \;\middle|\; \liminf_{\substack{\mathbf{y}\to\mathbf{x} \\ \mathbf{y}\neq\mathbf{x}}} \frac{1}{\|\mathbf{x}-\mathbf{y}\|} \left(\mathsf{f}(\mathbf{y}) - \mathsf{f}(\mathbf{x}) - \langle \mathbf{y}-\mathbf{x}, \hat{\mathbf{t}} \rangle\right) \geqslant 0 \right\}.$$

  Si $\mathbf{x} \notin \operatorname{dom} \mathsf{f}$, *alors* $\widehat{\partial}\mathsf{f}(\mathbf{x}) = \varnothing$.
- *Le* sous-différentiel *(limite) de* f *en* $\mathbf{x}$ *correspond à l'ensemble suivant*

$$\partial \mathsf{f}(\mathbf{x}) := \left\{ \mathbf{t} \in \mathbb{R}^N \;\middle|\; \exists \mathbf{y}_k \to \mathbf{x},\, \mathsf{f}(\mathbf{y}_k) \to \mathsf{f}(\mathbf{x}),\, \hat{\mathbf{t}}_k \in \widehat{\partial}\mathsf{f}(\mathbf{y}_k) \to \mathbf{t} \right\}.$$

Dans le cas où la fonction $\mathsf{f} \colon \mathbb{R}^N \to]-\infty, +\infty]$ est convexe et propre, alors pour tout $\mathbf{x} \in \mathbb{R}^N$ les sous-différentiels ci-dessus coïncident avec le sous-différentiel utilisé en analyse convexe [Rockafellar et Wets, 1997, Prop. 8.12].

Définition 2.10. *Soit* $\mathsf{f} \colon \mathbb{R}^N \to]-\infty, +\infty]$ *une fonction propre. L'ensemble des points critiques de* f *correspond aux zéros de* $\partial \mathsf{f}$.

Remarque 2.6. *Lorsque* $\mathsf{f} \in \Gamma_0(\mathbb{R}^N)$*, les points critiques de* f *correspondent à ses minimiseurs globaux.*

Lorsque $\mathsf{f}\colon \mathbb{R}^N \to]-\infty, +\infty]$ n'est pas nécessairement convexe, la règle de Fermat donnée par le théorème 2.3 n'est pas valable. Cependant, une des implications reste vraie :

Proposition 2.3. *Si* $\mathsf{f}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *est une fonction propre, alors on a*

$$\mathbf{x} \in \operatorname{Argmin}\ \mathsf{f} \quad \Rightarrow \quad \mathbf{0} \in \partial \mathsf{f}(\mathbf{x}).$$

Le théorème suivant permet de faire le lien entre la notion de sous-différentiel d'une fonction propre et la notion de cône normal :

Théorème 2.4. [Rockafellar et Wets, 1997, Thm. 8.9] *Soit* $\mathsf{f}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *une fonction propre. Pour tout* $\mathbf{x} \in \operatorname{dom} \mathsf{f}$*, on a*

$$\partial \mathsf{f}(\mathbf{x}) = \{\mathbf{t} \in \mathbb{R}^N \mid (\mathbf{t}, -1) \in \mathrm{N}_{\operatorname{epi} \mathsf{f}}(\mathbf{x}, \mathsf{f}(\mathbf{x}))\},$$

où $\mathrm{N}_{\operatorname{epi} \mathsf{f}}$ *est le cône normal de l'ensemble* $\operatorname{epi} \mathsf{f}$*.*

Remarque 2.7. *Ce théorème permet de se représenter graphiquement le sous différentiel d'une fonction propre non nécessairement convexe. Un exemple est donné dans la figure 2.9 dans le cas d'une fonction* $\mathsf{f}\colon \mathbb{R} \to]-\infty, +\infty]$*.*

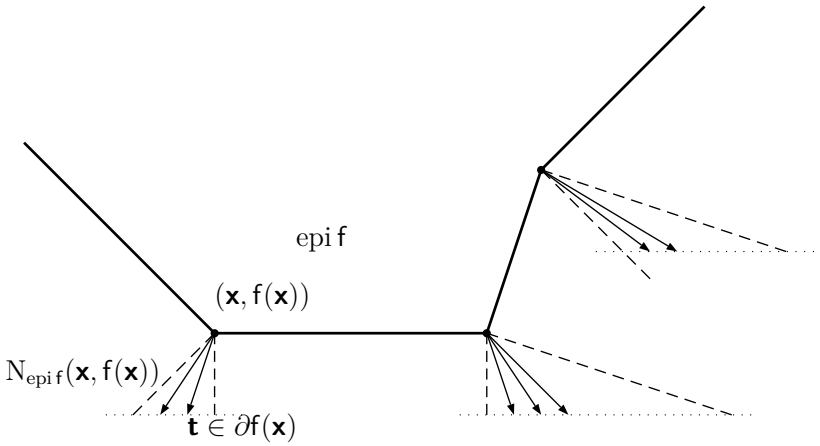


Figure 2.9 – *Cônes normaux de l'épigraphe d'une fonction* $\mathsf{f}\colon \mathbb{R} \to]-\infty, +\infty]$*.*

La propriété suivante, de fermeture du graphe de $\partial \mathsf{f}$, est une conséquence directe de la proposition 6.6 de [Rockafellar et Wets, 1997] et du théorème 2.4 précédent :

Proposition 2.4. *Soit* $(\mathbf{x}_k, \mathbf{t}_k)_{k\in\mathbb{N}}$ *une suite de l'ensemble* $\operatorname{Graph} \partial \mathsf{f}$*. Si* $(\mathbf{x}_k, \mathbf{t}_k)$ *converge vers* $(\mathbf{x}, \mathbf{t})$ *et* $\mathsf{f}(\mathbf{x}_k)$ *converge vers* $\mathsf{f}(\mathbf{x})$*, alors* $(\mathbf{x}, \mathbf{t}) \in \operatorname{Graph} \partial \mathsf{f}$*.*

Notons que cette propriété est utile pour démontrer la convergence de suites générées par des algorithmes minimisant des critères non convexes [Attouch *et al.*, 2010a, 2011; Bolte *et al.*, 2013]. Cependant, comme il est indiqué dans [Rockafellar et Wets, 1997, Chap. 6], elle n'est vérifiée que pour le sous-différentiel limite, mais pas pour le sous-différentiel de Fréchet.

2.3.3.2 Sous-différentiel partiel

Dans le chapitre 5 nous nous intéresserons à la minimisation de fonctions séparables par blocs, définies en dimension finie. Nous allons donc donner ici les règles de sous-différentiation de ces fonctions.

Soit $\mathsf{f}\colon \mathbb{R}^N \to]-\infty,+\infty]$ une fonction séparable par bloc. Plus précisément, soit $J \in \{1,\dots,N\}$, on peut alors décomposer tout vecteur $\mathbf{x} \in \mathbb{R}^N$ de la façon suivante

$$\mathbf{x} = (\mathbf{x}^{(j)})_{1\leqslant j\leqslant J} \in \mathbb{R}^{N_1} \times \ldots \times \mathbb{R}^{N_J}, \tag{2.39}$$

où $\sum_{j=1}^{J} N_j = N$, et on suppose que la fonction f peut s'écrire

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{f}(\mathbf{x}) = \sum_{j=1}^{J} \mathsf{f}_j(\mathbf{x}^{(j)}), \tag{2.40}$$

avec, pour tout $j \in \{1,\dots,J\}$, $\mathsf{f}_j\colon \mathbb{R}^{N_j} \to]-\infty,+\infty]$.

La proposition suivante nous donne les règles de base de sous-différentiation d'une fonction séparable.

Proposition 2.5. [Rockafellar et Wets, 1997, Prop. 10.5] *Soit* $\mathsf{f}\colon \mathbb{R}^N \to]-\infty,+\infty]$ *une fonction propre séparable vérifiant* (2.40)*. On a alors*

$$(\forall \mathbf{x} = (\mathbf{x}^{(j)})_{1\leqslant j\leqslant J} \in \operatorname{dom} \mathsf{f}) \quad \partial \mathsf{f}(\mathbf{x}) = \partial \mathsf{f}_1(\mathbf{x}^{(1)}) \times \ldots \times \partial \mathsf{f}_J(\mathbf{x}^{(J)}).$$

2.3.3.3 Gradient

Lorsque la fonction $\mathsf{f}\colon \mathcal{H} \to]-\infty,+\infty]$ (avec $\mathcal{H} = \mathbb{R}^N$ si f est non convexe) est différentiable au point $\mathbf{x} \in \mathcal{H}$, on a

$$\partial \mathsf{f}(\mathbf{x}) = \{\nabla \mathsf{f}(\mathbf{x})\}.$$

La notion suivante sera utile pour analyser la convergence des algorithmes d'optimisation que nous étudierons par la suite :

Définition 2.11. *Une fonction différentiable* $\mathsf{f}\colon \mathcal{H} \to]-\infty,+\infty]$ *est dite de gradient* β*-Lipschitz sur un sous-ensemble* E *de* $\mathcal{H}$*, pour* $\beta > 0$*, si*

$$\big(\forall (\mathbf{x},\mathbf{y}) \in E^2\big) \quad \|\nabla \mathsf{f}(\mathbf{x}) - \nabla \mathsf{f}(\mathbf{y})\| \leqslant \beta \|\mathbf{x}-\mathbf{y}\|.$$

Lorsqu'une fonction $\mathsf{f}\colon \mathbb{R}^N \to]-\infty,+\infty]$ est la somme d'une fonction différentiable et d'une fonction non différentiable, la règle de différentiation est alors donnée par la proposition suivante :

Proposition 2.6. [Rockafellar et Wets, 1997, Ex. 8.8] *Soit* $\mathsf{f}\colon \mathbb{R}^N \to]-\infty,+\infty]$ *une fonction propre telle que, pour tout* $\mathbf{x} \in \mathbb{R}^N$, $\mathsf{f}(\mathbf{x}) = \mathsf{f}_1(\mathbf{x}) + \mathsf{f}_2(\mathbf{x})$, *où* f_1 *est une fonction propre différentiable sur* $\operatorname{dom}\mathsf{f}$ *et* f_2 *est une fonction propre. On a alors*

$$(\forall \mathbf{x} \in \operatorname{dom}\mathsf{f}) \quad \partial \mathsf{f}(\mathbf{x}) = \nabla \mathsf{f}_1(\mathbf{x}) + \partial \mathsf{f}_2(\mathbf{x}).$$

2.3.4 Opérateurs proximaux

2.3.4.1 Définition

L'opérateur proximal en un point $\mathbf{x} \in \mathcal{H}$ a été défini dans [Moreau, 1965] pour une fonction $\mathsf{f} \in \Gamma_0(\mathcal{H})$ comme étant l'unique minimiseur de la fonction $\mathsf{f} + \frac{1}{2}\|\cdot - \mathbf{x}\|^2$. On a ainsi, pour $\gamma > 0$,

$$\operatorname{prox}_{\gamma \mathsf{f}}\colon \mathcal{H} \to \mathcal{H}\colon \mathbf{x} \mapsto \operatorname{prox}_{\gamma \mathsf{f}}(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{y} \in \mathcal{H}} \ \mathsf{f}(\mathbf{y}) + \frac{1}{2\gamma}\|\mathbf{y} - \mathbf{x}\|^2. \tag{2.41}$$

Cet opérateur a été largement étudié pour des fonctions convexes usuelles du traitement du signal, par exemple dans [Chaux *et al.*, 2007; Combettes et Pesquet, 2007b, 2010; Combettes et Wajs, 2005; Parikh et Boyd, 2013]. Cette définition peut être généralisée pour des fonctions $\mathsf{f}\colon \mathcal{H} \to]-\infty,+\infty]$ non nécessairement convexes dans le cas de normes pondérées (c.f. [Hiriart-Urruty et Lemaréchal, 1993, Sec. XV.4] et [Attouch *et al.*, 2011; Chouzenoux *et al.*, 2014b; Combettes et Vũ, 2013]) :

Définition 2.12. *Soient* $\mathsf{f}\colon \mathcal{H} \to]-\infty,+\infty]$ *une fonction propre et semi-continue inférieu-rement,* $\mathbf{U} \in \mathcal{S}^+(\mathcal{H})$, *et* $\mathbf{x} \in \mathcal{H}$. *L'opérateur proximal de* f *en* $\mathbf{x}$ *relativement à la métrique induite par* $\mathbf{U}$ *est l'ensemble défini par*

$$\operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) = \operatorname*{Argmin}_{\mathbf{y} \in \mathcal{H}} \ \mathsf{f}(\mathbf{y}) + \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|_{\mathbf{U}}^2. \tag{2.42}$$

Par abus de notation, lorsque cet ensemble est égal à un singleton, on note $\operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x})$ *son unique élément.*

Remarque 2.8.

(i) *Si* $\mathbf{U} = \gamma^{-1}\mathbf{Id}$, *pour* $\gamma > 0$, *et* $\mathsf{f} \in \Gamma_0(\mathcal{H})$, *alors* $\operatorname{prox}_{\gamma^{-1}\mathbf{Id},\mathsf{f}} \equiv \operatorname{prox}_{\gamma\mathsf{f}}$ *correspond à l'opérateur proximal* (2.41) *de* [Moreau, 1965]. *De plus, pour tout* $\mathbf{x} \in \mathcal{H}$, $\operatorname{prox}_{\gamma\mathsf{f}}(\mathbf{x})$ *est unique.*

(ii) *Si* $\mathcal{H} = \mathbb{R}^N$ *et* $\mathbf{U} = \gamma^{-1}\,\mathbf{I}$, *pour* $\gamma > 0$, *alors* $\operatorname{prox}_{\gamma\mathsf{f}}$ *est l'opérateur proximal défini dans* [Rockafellar et Wets, 1997, Déf. 1.22] *et utilisé dans* [Attouch *et al.*, 2011; Bolte *et al.*, 2013].

(iii) *Notons que, pour* $\mathcal{H} = \mathbb{R}^N$ *et* $\mathbf{U} \in \mathcal{S}_N^+$,

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) = \mathbf{U}^{-1/2} \operatorname{prox}_{\mathsf{f}\circ\mathbf{U}^{-1/2}}(\mathbf{U}^{1/2}\mathbf{x}). \tag{2.43}$$

En dimension finie, lorsque $\mathcal{H} = \mathbb{R}^N$ et $\mathsf{f}\colon \mathcal{H} \to]-\infty, +\infty]$ est une fonction non nécessairement convexe, la question d'existence de l'opérateur proximal de f se pose. Nous allons donc énoncer un ensemble de conditions permettant d'assurer que, pour tout $\mathbf{x} \in \mathbb{R}^N$, l'ensemble $\operatorname{prox}_{\gamma\mathsf{f}}(\mathbf{x})$ soit non vide.

Définition 2.13. [Rockafellar et Wets, 1997, Déf. 1.23] *Soit* $\mathsf{f}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *une fonction propre et semi-continue inférieurement. On dit que* f *est* prox-bornée *s'il existe* $\gamma > 0$ *tel que*

$$(\exists \mathbf{x} \in \mathbb{R}^N) \quad \inf_{\mathbf{y}\in\mathbb{R}^N} \mathsf{f}(\mathbf{y}) + \frac{1}{2\gamma}\|\mathbf{y} - \mathbf{x}\|^2 > -\infty. \tag{2.44}$$

De plus, on note $\gamma_{\mathsf{f}} = \sup\{\gamma > 0 \mid (2.44) \text{ est satisfaite}\}$ *et* f *est dite* γ_{f}-prox-bornée.

Théorème 2.5. [Rockafellar et Wets, 1997, Thm. 1.25] *Si* $\mathsf{f}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *est une fonction propre, semi-continue inférieurement et* γ_{f}-prox-*bornée, de seuil* $\gamma_{\mathsf{f}} > 0$, *alors, pour tout* $\gamma \in]0, \gamma_{\mathsf{f}}[$ *et tout* $\mathbf{x} \in \mathbb{R}^N$, *l'ensemble* $\operatorname{prox}_{\gamma\mathsf{f}}(\mathbf{x})$ *est non vide et compact.*
En particulier, si $\gamma_{\mathsf{f}} = +\infty$, *alors pour tout* $\gamma > 0$ *l'ensemble* $\operatorname{prox}_{\gamma\mathsf{f}}(\mathbf{x})$ *est non vide et compact.*

Proposition 2.7. [Rockafellar et Wets, 1997, Ex. 1.24] *Soit* $\mathsf{f}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *une fonction propre et semi-continue inférieurement. Les assertions suivantes sont équivalentes :*

(i) f *est* prox-*bornée.*

(ii) *Il existe une fonction polynomiale* q *de degré inférieur ou égal à* 2 *tel que* $\mathsf{f} \geqslant \mathsf{q}$.

(iii) *Il existe* $\lambda \in \mathbb{R}$ *tel que* $\inf \mathsf{f} + \dfrac{\lambda}{2}\|\cdot\|^2 > -\infty$.

(iv) $\liminf_{\|\mathbf{x}\|\to+\infty} \dfrac{\mathsf{f}(\mathbf{x})}{\|\mathbf{x}\|^2} > -\infty$.

Remarque 2.9. *Remarquons que si* λ_{f} *est l'infimum des* λ *vérifiant (iii) alors le seuil de* f *est* $\gamma_{\mathsf{f}} = 1/\max\{0, \lambda_{\mathsf{f}}\}$ *(avec* $1/0 = \infty$*). En particulier, si* f *est bornée inférieurement, i.e.* $\inf \mathsf{f} > -\infty$ *et donc* $\lambda_{\mathsf{f}} = 0$, *alors* f *est* γ_{f}-prox-*bornée, avec* $\gamma_{\mathsf{f}} = +\infty$, *et donc, pour tout* $\gamma > 0$ *et* $\mathbf{x} \in \mathbb{R}^N$, $\operatorname{prox}_{\gamma\mathsf{f}}(\mathbf{x}) \neq \varnothing$.

2.3.4.2 Propriétés

L'opérateur proximal étant très utilisé dans la résolution de problèmes variationnels par le biais d'algorithmes proximaux, il a largement été étudié ces dernières années, en particulier pour des fonctions $\mathsf{f} \in \Gamma_0(\mathcal{H})$ dans le cadre de la métrique usuelle euclidienne [Chaux *et al.*, 2007; Combettes et Pesquet, 2007b, 2010; Combettes et Wajs, 2005] et de métriques pondérées [Combettes et Vũ, 2013, 2014] (voir aussi [Becker et Fadili, 2012] en dimension finie). Nous listons dans cette section quelques propriétés utiles des opérateurs proximaux de fonctions de $\Gamma_0(\mathcal{H})$, et donnons leur équivalent, lorsque cela est possible, pour des fonctions non convexes quand $\mathcal{H} = \mathbb{R}^N$.

Dans cette section, $\mathcal{H}$ désigne soit un espace de Hilbert réel lorsque les fonctions étudiées sont convexes, soit $\mathbb{R}^N$ lorsque les fonction ne sont pas nécessairement convexes.

(i) **Caractérisation :**

Soient $\mathbf{U} \in \mathcal{S}^+(\mathcal{H})$, $\mathbf{x} \in \mathcal{H}$ et $\mathsf{f}\colon \mathcal{H} \to]-\infty, +\infty]$ une fonction propre, semi-continue inférieurement, telle que $\operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) \neq \varnothing$. On a alors

- Si $\mathbf{p} \in \operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x})$, alors $\mathbf{x} - \mathbf{p} \in \mathbf{U}^{-1}\partial\mathsf{f}(\mathbf{p})$.
- Si f est convexe, alors

$$\mathbf{p} = \operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) \quad \Leftrightarrow \quad \mathbf{x} - \mathbf{p} \in \mathbf{U}^{-1}\partial\mathsf{f}(\mathbf{p}). \tag{2.45}$$

(ii) **Point fixe :**

Soient $\mathbf{U} \in \mathcal{S}^+(\mathcal{H})$ et $\mathsf{f}\colon \mathcal{H} \to]-\infty, +\infty]$ une fonction propre, semi-continue inférieurement, telle que pour tout $\mathbf{x} \in \mathbb{R}^N$ $\operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) \neq \varnothing$. On a alors

- Si $\mathbf{x} \in \operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x})$, alors $\mathbf{x}$ est un point critique de f.
- Si f est convexe, alors

$$\mathbf{x} = \operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) \quad \Leftrightarrow \quad \mathbf{x} \in \operatorname{Argmin}\ \mathsf{f}.$$

(iii) **Translation :**

Soient $\mathbf{U} \in \mathcal{S}^+(\mathcal{H})$, $\mathbf{x}_0 \in \mathcal{H}$ et $\mathsf{g}\colon \mathcal{H} \to]-\infty, +\infty]$ une fonction propre, semi-continue inférieurement, telle que pour tout $\mathbf{x} \in \mathcal{H}$ $\operatorname{prox}_{\mathbf{U},\mathsf{g}}(\mathbf{x}) \neq \varnothing$. Soit $\mathsf{f} = \mathsf{f}(\cdot - \mathbf{x}_0)$. Pour tout $\mathbf{x} \in \mathcal{H}$, $\operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) \neq \varnothing$ et on a alors

$$\operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) = \{\mathbf{x}_0 + \mathbf{p} \mid \mathbf{p} \in \operatorname{prox}_{\mathbf{U},\mathsf{g}}(\mathbf{x} - \mathbf{x}_0)\} = \mathbf{x}_0 + \operatorname{prox}_{\mathbf{U},\mathsf{g}}(\mathbf{x} - \mathbf{x}_0).$$

(iv) **Changement d'échelle :**

Soient $\mathbf{U} \in \mathcal{S}^+(\mathcal{H})$, $\alpha \in \mathbb{R}^*$ et $\mathsf{g}\colon \mathcal{H} \to]-\infty,+\infty]$ une fonction propre, semi-continue inférieurement, telle que pour tout $\mathbf{x} \in \mathcal{H}$ $\operatorname{prox}_{\mathbf{U},\mathsf{g}}(\mathbf{x}) \neq \varnothing$. Soit $\mathsf{f} = \mathsf{g}(\cdot/\alpha)$. Pour tout $\mathbf{x} \in \mathcal{H}$, $\operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) \neq \varnothing$ et on a alors

$$\operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) = \alpha \operatorname{prox}_{\mathbf{U},\mathsf{g}/\alpha^2}(\mathbf{x}/\alpha).$$

(v) **Perturbation quadratique :**

Soient $\mathbf{U} \in \mathcal{S}^+(\mathcal{H})$, $\mathbf{A} \in \mathcal{S}^+(\mathcal{H})$ et $\mathsf{g}\colon \mathcal{H} \to]-\infty,+\infty]$ une fonction propre, semi-continue inférieurement, telle que pour tout $\mathbf{x} \in \mathcal{H}$ $\operatorname{prox}_{\mathbf{U},\mathsf{g}}(\mathbf{x}) \neq \varnothing$. Définissons $\mathsf{f} = \mathsf{g} + \dfrac{1}{2}\|\cdot\|_{\mathbf{A}}^2 + \beta\langle \cdot \mid \mathbf{x}_0\rangle + \gamma$, où $(\beta,\gamma) \in \mathbb{R}^2$ et $\mathbf{x}_0 \in \mathcal{H}$. On a alors

$$(\forall \mathbf{x} \in \mathcal{H}) \quad \operatorname{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) = \operatorname{prox}_{\mathbf{A}+\mathbf{U},\mathsf{g}}\Big((\mathbf{A}+\mathbf{U})^{-1}(\mathbf{U}\mathbf{x} - \beta\mathbf{x}_0)\Big). \tag{2.46}$$

Une démonstration de cette propriété est donnée dans l'annexe 2.A.

Remarquons que si $\mathbf{U} = \mathbf{Id}$, $\mathbf{A} = \alpha\mathbf{Id}$, où $\alpha \in]0,+\infty[$, et si $\mathsf{g} \in \Gamma_0(\mathcal{H})$ dans (2.46), on retrouve l'égalité énoncée dans [Combettes et Wajs, 2005] :

$$(\forall \mathbf{x} \in \mathcal{H}) \quad \operatorname{prox}_{\mathsf{f}}(\mathbf{x}) = \operatorname{prox}_{\mathsf{g}/(\alpha+1)}\left(\frac{\mathbf{x} - \beta\mathbf{x}_0}{\alpha+1}\right).$$

(vi) **Décomposition de Moreau :**

Soient $\mathsf{f} \in \Gamma_0(\mathcal{H})$ et $\mathbf{U} \in \mathcal{S}^+(\mathcal{H})$. On a alors [Combettes et Vũ, 2014, Ex. 3.9]

$$(\forall \mathbf{x} \in \mathcal{H}) \quad \mathbf{x} = \operatorname{prox}_{\mathbf{U},\mathsf{f}^*}(\mathbf{x}) + \mathbf{U}^{-1}\operatorname{prox}_{\mathbf{U}^{-1},\mathsf{f}}(\mathbf{U}\mathbf{x}). \tag{2.47}$$

Remarquons que lorsque $\mathbf{U} = \alpha\mathbf{Id}$, pour $\alpha \in]0,+\infty[$, dans l'égalité (2.47), on retrouve la décomposition de Moreau usuelle [Bauschke et Combettes, 2011, Thm. 14.3] :

$$(\forall \mathbf{x} \in \mathcal{H}) \quad \mathbf{x} = \operatorname{prox}_{\alpha\mathsf{f}}(\mathbf{x}) + \alpha\operatorname{prox}_{\alpha^{-1}\mathsf{f}^*}(\alpha^{-1}\mathbf{x}).$$

(vii) **Fonction additivement séparables :**

Soient $(\mathcal{H}_n)_{1\leqslant n\leqslant N}$ des espaces de Hilbert et $\boldsymbol{\mathcal{H}}$ leur somme directe. Soient, pour tout $n \in \{1,\dots,N\}$, $\mathsf{U}^{(n)} \in \mathcal{S}^+(\mathcal{H}_n)$, $\mathbf{x} = (\mathsf{x}^{(n)})_{1\leqslant n\leqslant N} \mapsto \mathbf{U}\mathbf{x} = (\mathsf{U}_n\mathsf{x}^{(n)})$, et $\mathsf{f}\colon \boldsymbol{\mathcal{H}} \to]-\infty,+\infty]$ une fonction additivement séparable, i.e. vérifiant

$$(\forall \mathbf{x} = (\mathsf{x}^{(n)})_{1\leqslant n\leqslant N} \in \boldsymbol{\mathcal{H}}) \quad \mathsf{f}(\mathbf{x}) = \sum_{n=1}^{N} \mathsf{f}_n(\mathsf{x}^{(n)}). \tag{2.48}$$

Supposons que, pour tout $n \in \{1,\dots,N\}$, $\mathsf{f}_n\colon \mathcal{H}_n \to]-\infty,+\infty]$ soit une fonction semi-continue inférieurement, propre, et telle que, pour tout $\mathsf{x} \in \mathcal{H}_n$, $\mathrm{prox}_{\mathsf{f}_n}(\mathsf{x}) \neq \varnothing$. On a alors

$$(\forall \mathbf{x} \in \boldsymbol{\mathcal{H}}) \quad \mathrm{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) = (\mathsf{p}^{(n)})_{1\leqslant n\leqslant N}, \tag{2.49}$$

où, pour tout $n \in \{1,\dots,N\}$, $\mathsf{p}^{(n)} = \mathrm{prox}_{\mathsf{U}_n,\mathsf{f}_n}(\mathsf{x}^{(n)})$.

Lorsque $\boldsymbol{\mathcal{H}} = \mathbb{R}^N$ et, pour tout $n \in \{1,\dots,N\}$, $\mathsf{f}_n \in \Gamma_0(\mathbb{R})$, un résultat plus général a été énoncé dans [Becker et Fadili, 2012] :

Proposition 2.8. [Becker et Fadili, 2012, Cor. 9] *Soient $\mathsf{f} \in \Gamma_0(\mathbb{R}^N)$ une fonction additivement séparable et $\mathbf{U} = \mathrm{Diag}\left(\mathsf{u}^{(1)},\dots,\mathsf{u}^{(n)}\right) + \mathbf{v}\mathbf{v}^\top$ où $\mathbf{v} \in \mathbb{R}^N$ et, pour tout $n \in \{1,\dots,N\}$, $\mathsf{u}^{(n)} > 0$. On a alors*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathrm{prox}_{\mathbf{U},\mathsf{f}}(\mathbf{x}) = (\mathsf{p}^{(n)})_{1\leqslant n\leqslant N}, \tag{2.50}$$

où

$$(\forall n \in \{1,\dots,N\}) \quad \mathsf{p}^{(n)} = \mathrm{prox}_{\mathsf{f}_n/\mathsf{u}^{(n)}}\left(\mathsf{x}^{(n)} - \frac{\alpha^*\mathsf{v}^{(n)}}{\mathsf{u}^{(n)}}\right), \tag{2.51}$$

et α^* est l'unique zéro de la fonction

$$\alpha \in \mathbb{R} \mapsto \left\langle \mathbf{v}, \mathbf{x} - \left(\mathrm{prox}_{\mathsf{f}_n/\mathsf{u}^{(n)}}(\mathsf{x}^{(n)} - \alpha\mathsf{v}^{(n)}/\mathsf{u}^{(n)})\right)\right\rangle + \alpha.$$

2.3.4.3 Cas particuliers

Dans cette section nous donnerons quelques exemples utiles d'opérateurs proximaux de fonctions usuelles pour la résolution de problèmes inverses.

(i) **Opérateur de projection :**

Définition 2.14. *Soit E un sous-ensemble de $\mathcal{H}$. La projection de $\mathbf{x} \in \mathcal{H}$ sur E est définie par*

$$\Pi_E(\mathbf{x}) \in \operatorname*{Argmin}_{\mathbf{y}\in E} \|\mathbf{y} - \mathbf{x}\|. \tag{2.52}$$

En d'autres termes, un élément de $\Pi_E(\mathbf{x})$ est un point de E tel que sa distance à $\mathbf{x}$ soit minimale.

Remarquons que l'opérateur proximal peut être vu comme une généralisation de l'opérateur de projection au sens où, pour tout sous-ensemble E de $\mathcal{H}$,

$$\Pi_E = \mathrm{prox}_{\iota_E}, \tag{2.53}$$

où ι_E désigne la fonction indicatrice de E.

Proposition 2.9. [Bauschke et Combettes, 2011, Thm 3.14] *Si C est un sous-ensemble convexe fermé non vide de $\mathcal{H}$, alors la projection $\Pi_C(\mathbf{x})$ de $\mathbf{x} \in \mathcal{H}$ sur C existe et est unique.*

(ii) **Opérateurs proximaux de fonctions puissances :**
Dans ce paragraphe, nous allons donner les opérateurs proximaux des fonctions

$$\mathsf{f}_\alpha \colon \mathbb{R}^N \to]-\infty, +\infty] \colon \mathbf{x} \mapsto \sum_{n=1}^{N} |\mathsf{x}^{(n)}|^\alpha, \tag{2.54}$$

pour différents $\alpha \in [0, +\infty[$. Notons que f étant additivement séparable, dans la suite nous n'étudierons que les opérateurs proximaux des fonctions $\psi_\alpha \colon \mathsf{x} \in \mathbb{R} \mapsto |\mathsf{x}|^\alpha$.

- Lorsque $\alpha \in \{0, 1\}$, l'opérateur proximal de ψ_α est appelé *opérateur de seuillage*. En particulier, lorsque $\alpha = 1$ nous parlerons de seuillage doux [Combettes et Pesquet, 2010], et lorsque $\alpha = 0$ (en utilisant la convention $0^0 = 0$), nous parlerons de seuillage dur [Blumensath et Davies, 2008] (voir figure 2.10).

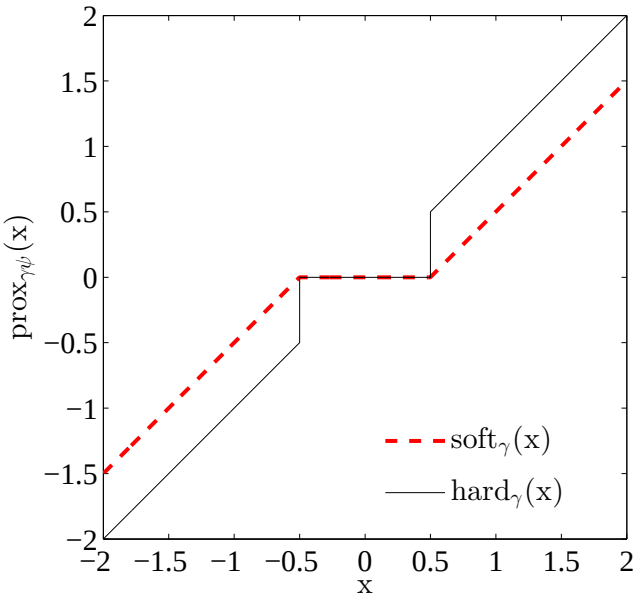


Figure 2.10 – *Opérateurs de seuillage doux (courbe rouge discontinue) et de seuillage dur (courbe noire continue).*

Définition 2.15.

(a) *Soit $\gamma > 0$. On définit l'opérateur de* seuillage doux *de seuil γ comme suit :*

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \mathrm{soft}_\gamma(\mathsf{x}) = \mathrm{sign}(\mathsf{x}) \max\{|\mathsf{x}| - \gamma, 0\} \tag{2.55}$$

$$= \begin{cases} \mathsf{x} - \gamma & \text{si } \mathsf{x} > \gamma, \\ 0 & \text{si } \mathsf{x} \in [-\gamma, \gamma], \\ \mathsf{x} + \gamma & \text{si } \mathsf{x} < -\gamma. \end{cases} \tag{2.56}$$

(b) *Soit $\gamma > 0$. On définit l'opérateur de* seuillage dur *de seuil γ comme suit :*

$$(\forall x \in \mathbb{R}) \quad \mathrm{hard}_\gamma(x) = \begin{cases} 0 & \text{si } |x| \leqslant \gamma, \\ x & \text{sinon.} \end{cases} \tag{2.57}$$

Le résultat suivant permet de construire d'autres opérateurs de seuillage à partir de l'opérateur de seuillage doux :

Proposition 2.10. [Combettes et Pesquet, 2007b, Prop. 3.6] *Soit $\varphi \in \Gamma_0(\mathbb{R})$ une fonction différentiable en 0 telle que $\psi'(0) = 0$. Soit*

$$\phi \colon \mathbb{R} \to]-\infty, +\infty] \colon x \mapsto \varphi(x) + \gamma|x|,$$

pour $\gamma > 0$. On a alors

$$\mathrm{prox}_\phi = \mathrm{prox}_\varphi \circ \mathrm{soft}_\gamma \,. \tag{2.58}$$

- Lorsque $\alpha \in [1, +\infty[$, l'opérateur proximal de ψ_α est donné par la proposition suivante.

Proposition 2.11. [Chaux *et al.*, 2007; Combettes et Pesquet, 2010] *Soient $\alpha \geqslant 1$ et $\gamma > 0$. L'opérateur proximal de ψ_α est donné par*

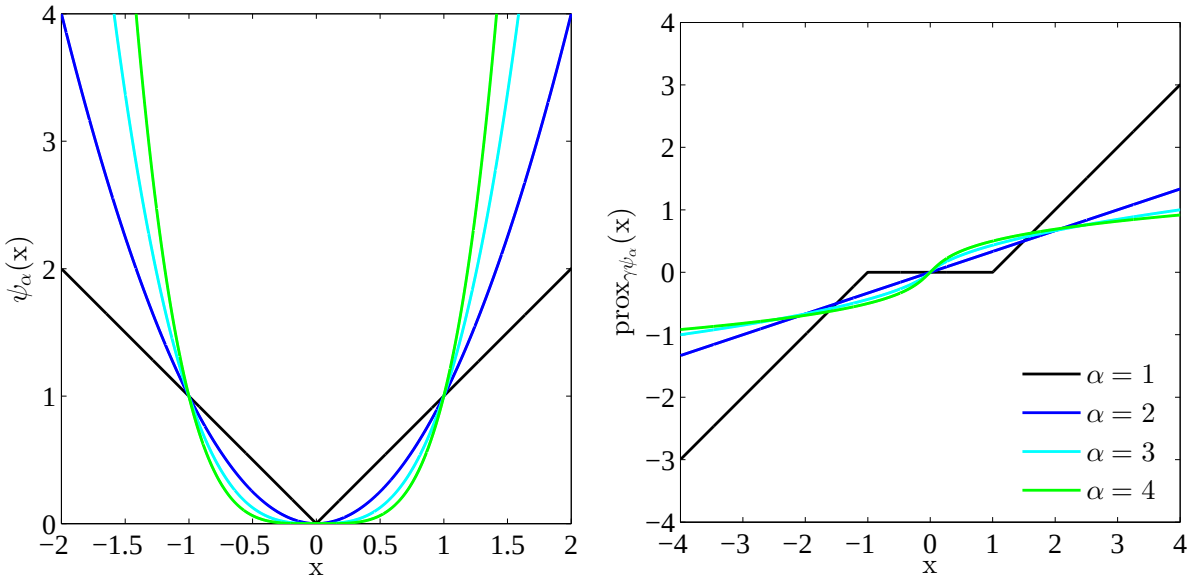
$$(\forall x \in \mathbb{R}) \quad \mathrm{prox}_{\gamma\psi_\alpha}(x) = \mathrm{sign}(x)p, \tag{2.59}$$

où $p \geqslant 0$ et satisfait $p + \alpha\gamma p^{\alpha-1} = |x|$.

Cette proposition permet d'obtenir une caractérisation de l'opérateur proximal de la fonction ψ_α, pour $\alpha \geqslant 1$. Cependant, seules certaines valeurs de α conduisent à une forme explicite de cet opérateur. En voici quelques exemples [Chaux *et al.*, 2007, Ex. 4.2 et 4.3] :

$$(\forall x \in \mathbb{R}) \quad \mathrm{prox}_{\gamma\psi_\alpha}(x) = \begin{cases} \mathrm{soft}_\gamma(x) & \text{si } \alpha = 1, \\ \dfrac{x}{1+2\gamma} & \text{si } \alpha = 2, \\ \mathrm{sign}(x)\dfrac{\sqrt{1+12\gamma|x|}-1}{6\gamma} & \text{si } \alpha = 3, \\ \left(\dfrac{t+x}{8\gamma}\right)^{1/3} - \left(\dfrac{t-x}{8\gamma}\right)^{1/3} & \\ \quad \text{où } t = \sqrt{x^2 + 1/(27\gamma)} & \text{si } \alpha = 4. \end{cases}$$

Remarquons en particulier que l'on retrouve l'opérateur de seuillage doux lorsque $\alpha = 1$. Une illustration de ces opérateurs proximaux ainsi que des fonctions associées est donnée dans la figure 2.11.



(a) (b)

Figure 2.11 – (a) *Fonction* ψ_α *pour* $\alpha \in \{1, 2, 3, 4\}$. (b) *Opérateur proximal* $\text{prox}_{\gamma\psi_\alpha}$ *pour* $\gamma = 1$ *et* $\alpha \in \{1, 2, 3, 4\}$.

- Lorsque $\alpha \in]0, 1[$, l'opérateur proximal de la fonction $|\cdot|^\alpha$ correspond à un seuillage et doit être calculé de façon itérative [Antoniadis *et al.*, 2002; Marjanovic et Solo, 2012].
- Nous allons maintenant introduire la fonction de Huber généralisée $\varphi_{\delta,\alpha} \colon \mathbb{R} \to \mathbb{R}^+$, pour $\alpha \in]0, 1]$ et $\delta > 0$, définie par

$$\varphi_{\delta,\alpha}(x) = \begin{cases} \dfrac{x^2}{2\delta} & \text{si } |x| \leqslant \delta^{1/(2-\alpha)}, \\ \dfrac{|x|^\alpha}{\alpha} - \left(\dfrac{1}{\alpha} - \dfrac{1}{2}\right)\delta^{\alpha/(2-\alpha)} & \text{sinon.} \end{cases} \tag{2.60}$$

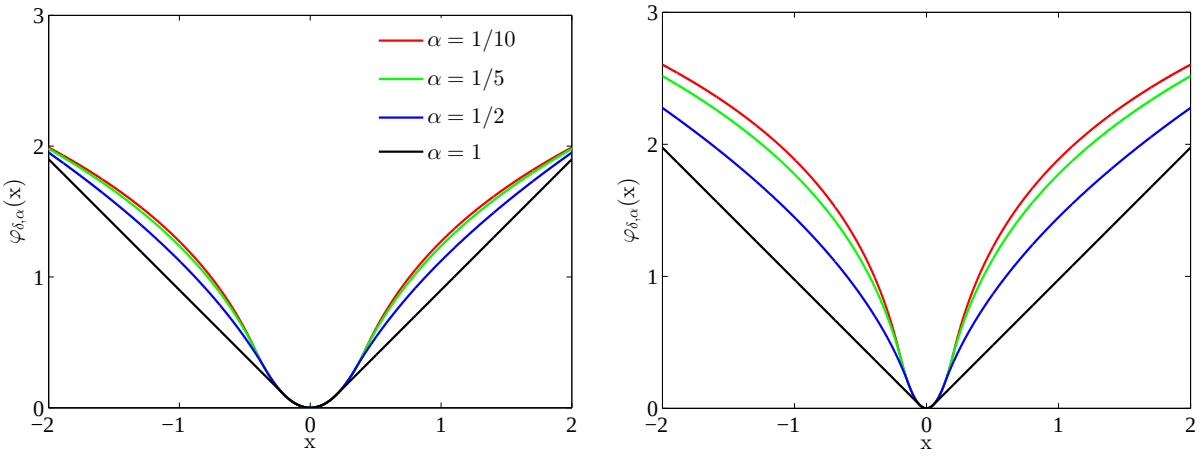
Remarquons que, si $\alpha = 1$, $\varphi_{\delta,1} = \varphi_\delta$ est la fonction de Huber définie par (2.28). De plus, la fonction de Huber généralisée peut être vue comme une approximation lisse de la fonction $\psi_\alpha = |\cdot|^\alpha$, pour $\alpha \in]0, 1[$, et, en particulier, elle n'est pas convexe. Une illustration de cette fonction est donnée dans la figure 2.12 pour différentes valeurs de α. Dans [Chartrand, 2012], cette fonction est utilisée pour définir un opérateur de seuillage, que nous utiliserons dans le chapitre 6 dans une application en reconstruction de données hyperspectrales. Ce seuillage est donné dans la proposition suivante.

Proposition 2.12. [Chartrand, 2012] *Soient* $\alpha \in]0, 1]$, $\delta > 0$, *et* $\varrho_{\delta,\alpha} \in \Gamma_0(\mathbb{R})$ *définie par*

$$(\forall x \in \mathbb{R}) \quad \varrho_{\delta,\alpha}(x) = \frac{x^2}{2} - \delta\varphi_{\delta,\alpha}(x). \tag{2.61}$$

L'opérateur proximal de la fonction $\phi_{\delta,\alpha} \colon \mathbb{R} \to]-\infty, +\infty]$ *définie par*

$$(\forall x \in \mathbb{R}) \quad \phi_{\delta,\alpha}(x) = \varrho^*_{\delta,\alpha}(x) - \frac{x^2}{2}, \tag{2.62}$$



(a) (b)

Figure 2.12 – *Fonction de Huber généralisée pour* $\alpha \in \{1/10, 1/5, 1/2, 1\}$*, avec* (a) $\delta = 1/5$ *et* (b) $\delta = 1/20$.

est un seuillage donné par

$$(\forall x \in \mathbb{R}) \quad \operatorname{prox}_{\phi_{\delta,\alpha}}(x) = \begin{cases} \max\left\{0, |x| - \delta|x|^{\alpha-1}\right\} \dfrac{x}{|x|} & \text{si } x \neq 0, \\ 0 & \text{si } x = 0. \end{cases} \tag{2.63}$$

Les fonctions $\varrho_{\delta,\alpha}$, $\phi_{\delta,\alpha}$ et $\operatorname{prox}_{\phi_{\delta,\alpha}}$, définies dans la proposition 2.12, sont tracées dans la figure 2.13.

2.3.5 Inégalité de Kurdyka-Łojasiewicz

2.3.5.1 Motivation

Soit $\mathsf{f}\colon \mathbb{R}^N \to \mathbb{R}$ une fonction différentiable. On considère la dynamique du gradient donnée par

$$\begin{cases} (\forall \mathsf{t} \in]0, +\infty[) \quad \dot{\mathbf{x}}(\mathsf{t}) = -\nabla \mathsf{f}(\mathbf{x}(\mathsf{t})), \\ \mathbf{x}(0) = \mathbf{x}_0. \end{cases}$$

La résolution de cette équation différentielle génère une trajectoire de gradient $\mathsf{t} \in]0, +\infty[\mapsto \mathbf{x}(\mathsf{t})$ partant de $\mathbf{x}_0$ et suivant une trajectoire changeant à chaque intersection avec une ligne de niveau de f de façon à prendre la direction orthogonale à la ligne de niveau croisée. La solution $\mathbf{x}(\mathsf{t})$ générée assure que $\mathsf{t} \in]0, +\infty[\mapsto \mathsf{f}(\mathbf{x}(\mathsf{t}))$ soit décroissante. Cette dynamique peut être vue comme une version continue de l'algorithme de descente de gradient donné par

$$(\forall k \in \mathbb{N}) \quad \mathbf{x}_{k+1} = \mathbf{x}_k - \mu_k \nabla \mathsf{f}(\mathbf{x}_k),$$

où $\mu_k > 0$.

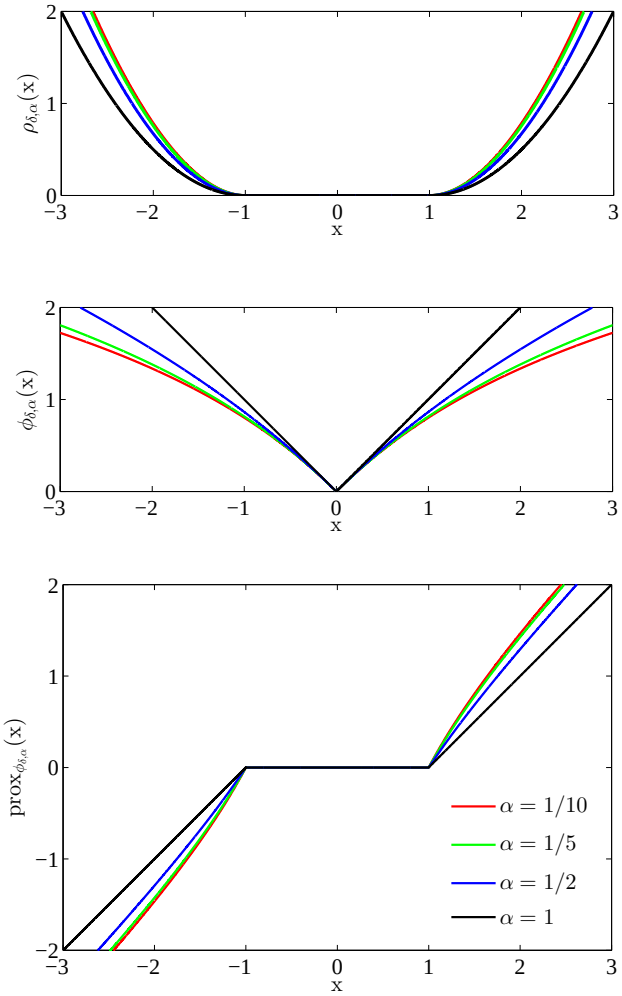


Figure 2.13 – *Fonctions* $\rho_{1,\alpha}$ *(haut),* $\phi_{1,\alpha}$ *(centre), et* $\mathrm{prox}_{\phi_{1,\alpha}}$ *(bas), pour* $\alpha \in \{1/10, 1/5, 1/2, 1\}$.

Considérons une trajectoire de gradient $\mathbf{x}(\mathsf{t})$ (resp. une suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$) bornée. On voudrait pouvoir assurer la convergence de $\mathbf{x}(\mathsf{t})$ (resp. une suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$) lorsque $\mathsf{t} \to +\infty$ (resp. $k \to \infty$), vers un point critique de f, i.e. assurer que la trajectoire $\mathbf{x}(\mathsf{t})$ soit de longueur finie.

Dans [Absil *et al.*, 2005; Palis et De Melo, 1982] (voir aussi [Bolte *et al.*, 2007, Ex. 1]), des contre-exemples ont été exhibés proposant des fonctions infiniment dérivables de $\mathbb{R}$ ou de $\mathbb{R}^2$ dont la courbe de gradient $\mathbf{x}(\mathsf{t})$ est bornée, mais de longueur infinie (elle tourne indéfiniment autour du cercle unité). Il a été démontré dans ces mêmes références que les fonctions vérifiant l'inégalité de Kurdyka-Łojasiewicz ne présentaient pas ce type de problème.

2.3.5.2 Inégalité de Łojasiewicz

L'inégalité de Kurdyka-Łojasiewicz (KL) apparaît ainsi comme un outil fondamental en optimisation non convexe qui permet de démontrer la convergence de suites minimisant des fonctions non nécessairement convexes. Cette inégalité a été proposée en premier dans [Łojasiewicz, 1963], et il a été démontré qu'elle est satisfaite par toute fonction analytique réelle :

Propriété 2.2. (Inégalité de Łojasiewicz) *Soient* $\mathsf{f}\colon \mathbb{R}^N \to \mathbb{R}$ *une fonction analytique réelle, et* $\widehat{\mathbf{x}}$ *un point critique de cette fonction. Il existe* $\theta \in [1/2, 1[$ *et* $\kappa \in]0, +\infty[$ *tels que* f *satisfait l'*inégalité de Łojasiewicz *dans un voisinage* $\mathcal{V}(\widehat{\mathbf{x}})$ *de* $\widehat{\mathbf{x}}$*, i.e. :*

$$(\forall \mathbf{x} \in \mathcal{V}(\widehat{\mathbf{x}})) \quad |\mathsf{f}(\mathbf{x}) - \mathsf{f}(\widehat{\mathbf{x}})|^{\theta} \leqslant \kappa \|\nabla \mathsf{f}(\mathbf{x})\|. \tag{2.64}$$

Cette inégalité a été utilisée pour démontrer la convergence d'algorithmes de descente de gradient pour la minimisation de fonctions différentiables non nécessairement convexe [Absil *et al.*, 2005], elle permet d'assurer que la courbe de gradient générée par ce type d'algorithme est de longueur finie [Absil *et al.*, 2005] (voir aussi [Bolte *et al.*, 2010]).

2.3.5.3 Inégalité de Kurdyka-Łojasiewicz

L'inégalité de Łojasiewicz a été généralisée dans [Kurdyka, 1998] pour les fonctions différentiables définissables dans une structure o-minimale (pour les structures o-minimales, voir par exemple [Coste, 1999; Van den Dries et Miller, 1996]). On parlera alors d'inégalité de KL. Plus récemment, dans [Bolte *et al.*, 2006a,b, 2007] il a été démontré que l'inégalité de KL peut être généralisée pour des fonctions non nécessairement différentiables en utilisant les sous-différentiel donné dans la définition 2.9. Cette inégalité, plus générale que (2.64), s'écrit de la façon suivante :

Définition 2.16. [Attouch *et al.*, 2010a, Déf. 7] *La fonction* $\mathsf{f}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *satisfait l'*inégalité de Kurdyka-Łojasiewicz *en* $\widehat{\mathbf{x}} \in \operatorname{dom} \partial \mathsf{f}$ *s'il existe* $\xi \in]0, +\infty]$*, un voisinage* $\mathcal{V}(\widehat{\mathbf{x}})$ *de* $\widehat{\mathbf{x}}$*, et une fonction* $\varphi\colon [0, \xi[\to [0, +\infty[$ *concave tels que*

(i) $\varphi(0) = 0$,

(ii) φ *est continue en* 0 *et continument différentiable sur* $]0, \xi[$,

(iii) *pour tout* $\mathsf{u} \in]0, \xi[$, $\varphi'(\mathsf{u}) > 0$,

(iv) $\varphi'\left(\mathsf{f}(\mathbf{x}) - \mathsf{f}(\widehat{\mathbf{x}})\right) \operatorname{dist}(\mathbf{0}, \partial \mathsf{f}(\mathbf{x})) \geqslant 1$, *pour tout* $\mathbf{x} \in E$ *tel que* $\mathsf{f}(\widehat{\mathbf{x}}) \leqslant \mathsf{f}(\mathbf{x}) \leqslant \mathsf{f}(\widehat{\mathbf{x}}) + \xi$, *où* $\operatorname{dist}(\mathbf{0}, \partial \mathsf{f}(\mathbf{x})) = \inf_{\mathbf{u} \in \partial \mathsf{f}(\mathbf{x})} \|\mathbf{u}\|$.

Cette propriété a permis, entre autres, de démontrer la convergence d'algorithmes proximaux [Attouch et Bolte, 2009], et d'algorithmes proximaux alternés [Attouch *et al.*, 2010a] pour des fonctions non nécessairement convexes.

2.3.5.4 Fonctions satisfaisant l'inégalité de Kurdyka-Łojasiewicz

Il a été démontré que l'inégalité de Kurdyka-Łojasiewicz est satisfaite par les fonctions définissables dans une structure o-minimale, et plus particulièrement par les fonctions semi-algébriques et analytiques réelles [Attouch *et al.*, 2010a; Bolte *et al.*, 2006a, 2007, 2010].

Les structures o-minimales réelles, introduites dans [Van den Dries, 1998], sont définies comme suit :

Définition 2.17. [Attouch *et al.*, 2010a, Déf. 13] *Soit $\mathcal{O} = \{\mathcal{O}_N\}_{N\in\mathbb{N}}$ tel que, pour tout $N \in \mathbb{N}$, $\mathcal{O}_N$ est une collection de sous-ensembles de $\mathbb{R}^N$. On dit que la famille $\mathcal{O}$ est une structure o-minimale sur $\mathbb{R}$ si les axiomes suivants sont vérifiés :*

(i) *Chaque ensemble $\mathcal{O}_N$ est une algèbre booléenne, i.e. $\varnothing \in \mathcal{O}_N$ et, pour tout A, B dans $\mathcal{O}_N$, les ensembles $A \cup B$, $A \cap B$, et $\mathbb{R}^N \setminus A$ sont dans $\mathcal{O}_N$.*

(ii) *Pour tout A dans $\mathcal{O}_N$, $A \times \mathbb{R}$ et $\mathbb{R} \times A$ sont dans $\mathcal{O}_{N+1}$.*

(iii) *Pour tout A dans $\mathcal{O}_{N+1}$, $\{(\mathsf{x}_1, \ldots, \mathsf{x}_N) \in \mathbb{R}^N \mid (\mathsf{x}_1, \ldots, \mathsf{x}_N, \mathsf{x}_{N+1}) \in A\} \in \mathcal{O}_N$.*

(iv) *Pour tout $(m, n) \in \{1, \ldots, N\}^2$ tel que $m \neq n$, $\{(\mathsf{x}_1, \ldots, \mathsf{x}_N) \in \mathbb{R}^N \mid \mathsf{x}_m = \mathsf{x}_n\} \in \mathcal{O}_N$.*

(v) *L'ensemble $\{(\mathsf{x}_1, \mathsf{x}_2) \in \mathbb{R}^2 \mid \mathsf{x}_1 < \mathsf{x}_2\}$ est dans $\mathcal{O}_2$.*

(vi) *$\mathcal{O}_1$ coïncide exactement avec l'ensemble des unions finies d'intervalles.*

Nous pouvons en déduire la définition d'ensembles et de fonctions définissables :

Définition 2.18. *Soit $\mathcal{O} = \{\mathcal{O}_N\}_{N\in\mathbb{N}}$ une structure o-minimale.*

(i) *Un ensemble E est* définissable dans $\mathcal{O}$ *si $E \subset \mathcal{O}_N$.*

(ii) *Une fonction $\mathsf{f} \colon \mathbb{R}^N \to]-\infty, +\infty]$ est* définissable dans $\mathcal{O}$ *si son graphe est définissable.*

Pour certains cas particuliers de structures o-minimales, la forme de la fonction φ de l'inégalité de KL est connue [Attouch *et al.*, 2010a, Sec. 4.3]. Plus particulièrement, dans la suite de cette thèse nous nous intéresserons aux fonctions semi-algébriques réelles (dont la définition est donnée ci-dessous) et analytiques réelles.

Définition 2.19. [Attouch *et al.*, 2011; Bochnak *et al.*, 1998; Shiota, 1997]

(i) *Un sous-ensemble S de $\mathbb{R}^N$ est un ensemble* semi-algébrique réel *s'il existe un nombre fini de fonctions polynomiales $\mathsf{p}_{i,j}, \mathsf{q}_{i,j} \colon \mathbb{R}^N \to \mathbb{R}$, pour $i \in \{1, \ldots, p\}$ et $j \in \{1, \ldots, q\}$, telles que*

$$S = \cup_{j=1}^{p} \cap_{i=1}^{q} \{\mathbf{x} \in \mathbb{R}^N \mid \mathsf{p}_{i,j}(\mathbf{x}) = 0,\ \mathsf{q}_{i,j}(\mathbf{x}) < 0\}.$$

(ii) *Une fonction $\mathsf{f} \colon \mathbb{R}^N \to]-\infty, +\infty]$ est* semi-algébrique *si son graphe est un sous-ensemble semi-algébrique de $\mathbb{R}^{N+1}$.*

Remarque 2.10. [Bolte *et al.*, 2006b] *La classe des ensembles semi-algébriques est stable pour les opérations de réunion finie, d'intersection finie, de produit cartésien, par passage au complémentaire et par projection (principe de Tarski-Seidenberg* [Bochnak *et al.*, 1998]*).*

Pour les fonctions semi-algébriques réelles et analytiques réelles, la fonction φ de l'inégalité de KL est de la forme $\varphi \colon u \in [0, \xi[\mapsto u^{1-\theta}$, pour $\theta \in [0, 1[$:

Définition 2.20. [Attouch et Bolte, 2009] *La fonction $\mathsf{f} \colon \mathbb{R}^N \to]-\infty, +\infty]$ satisfait l'inégalité de Kurdyka-Łojasiewicz si, pour tout $\xi \in \mathbb{R}$, et, pour tout sous-ensemble borné E de $\mathbb{R}^N$, il existe trois constantes $\kappa > 0$, $\zeta > 0$ et $\theta \in [0, 1[$ telles que*

$$(\forall \mathbf{t} \in \partial \mathsf{f}(\mathbf{x})) \qquad \|\mathbf{t}\| \geqslant \kappa |\mathsf{f}(\mathbf{x}) - \xi|^{\theta},$$

pour tout $\mathbf{x} \in E$ tel que $|\mathsf{f}(\mathbf{x}) - \xi| \leqslant \zeta$ (avec la convention $0^0 = 0$).

Cette forme de l'inégalité de KL est utilisée pour démontrer des résultats de convergence dans [Attouch et Bolte, 2009], et permet d'obtenir de plus des résultats de vitesse de convergence [Attouch et Bolte, 2009; Attouch *et al.*, 2010a; Frankel *et al.*, 2014].

Remarque 2.11. *Comme souligné dans* [Attouch *et al.*, 2010a]*, les fonctions logarithme et exponentielle sont définissables. Il existe donc une fonction φ telle que l'inégalité de KL dans sa forme la plus générale (définition 2.16) soit satisfaite. Cependant, ces fonctions ne sont ni semi-algébriques réelles, ni analytiques réelles, et elles ne satisfont pas l'inégalité donnée dans la définition 2.20.*

2.4 Algorithmes d'optimisation

Cette section a pour but de donner un rapide survol des méthodes d'optimisation de base permettant de résoudre des problèmes variationnels usuels.

2.4.1 Algorithmes de gradient et de sous-gradient

Dans un premier temps, nous allons nous intéresser au problème d'optimisation suivant :

$$\text{trouver } \widehat{\mathbf{x}} \in \underset{\mathbf{x}\in\mathcal{H}}{\operatorname{Argmin}} \ \mathsf{h}(\mathbf{x}), \tag{2.65}$$

où $\mathsf{h} \in \Gamma_0(\mathcal{H})$, et $\mathcal{H}$ est un espace de Hilbert réel. D'après la règle de Fermat, $\widehat{\mathbf{x}}$ est une solution de (2.65) si et seulement si $\mathbf{0} \in \partial\mathsf{h}(\widehat{\mathbf{x}})$.

Algorithme de gradient

Dans le cas particulier où h est une fonction différentiable et de gradient β-Lipschitz ($\beta > 0$), cette condition devient alors

$$\mathbf{0} = \nabla\mathsf{h}(\widehat{\mathbf{x}}). \tag{2.66}$$

On peut donc trouver une solution explicite du problème de minimisation (2.65) pour des fonctions h assez simples. Cependant, un tel $\widehat{\mathbf{x}}$ peut se révéler difficile à trouver dès lors que h a une forme plus compliquée et que l'on ne sait plus résoudre (2.66). Dans ce cas, il faut définir la solution du problème de minimisation comme étant la limite d'une suite construite itérativement, par le biais d'algorithmes. Une méthode permettant de trouver une solution itérative du problème de minimisation (2.65) est connue sous le nom d'*algorithme du gradient* (ou *de descente de gradient*) [Bertsekas, 1999, Chap. 1] et est donnée par l'algorithme 2.1.

Algorithme 2.1 Algorithme du gradient

Initialisation : Soient $\mathbf{x}_0 \in \mathbb{R}^N$ et, pour tout $k \in \mathbb{N}$, $\underline{\gamma} \leqslant \gamma_k \leqslant (2-\overline{\gamma})\beta^{-1}$, avec $(\underline{\gamma}, \overline{\gamma}) \in]0, +\infty[^2$.

Itérations :

Pour $k = 0, 1, \ldots$

$\quad \mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k \nabla\mathsf{h}(\mathbf{x}_k)$.

Algorithme de sous-gradient

Lorsque h n'est pas différentiable, une méthode permettant de résoudre le problème (2.65) consiste à utiliser un *algorithme de sous-gradient* [Bertsekas, 1999, Chap. 6], donné par l'algorithme 2.2.

Algorithme 2.2 Algorithme de sous-gradient

Initialisation : Soient $\mathbf{x}_0 \in \mathbb{R}^N$ et, pour tout $k \in \mathbb{N}$, $\gamma_k \in]0, +\infty[$ tels que $\sum_{k\in\mathbb{N}} \gamma_k = +\infty$ et $\sum_{k\in\mathbb{N}} \gamma_k^2 < +\infty$.

Itérations :

Pour $k = 0, 1, \ldots$

$\quad \mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k \mathbf{t}_k, \quad$ où $\mathbf{t}_k \in \partial \mathsf{h}(\mathbf{x}_k)$.

La convergence de la suite $(\mathsf{h}(\mathbf{x}_k))_{k\in\mathbb{N}}$ générée par cet algorithme, vers une solution du problème 2.65, est assurée lorsque le pas $(\gamma_k)_{k\in\mathbb{N}}$ satisfait une condition de carré sommable [Shor, 1985] (voir aussi [Bertsekas, 1999, Ex. 6.3.14] et [Boyd *et al.*, 2003]). Notons que cette hypothèse peut se révéler contraignante en pratique, puisqu'elle peut freiner la convergence et poser des problèmes numériques.

Algorithme proximal

Un moyen pour pallier le problème énoncé dans le paragraphe précédent, est de définir, à chaque itération $k \in \mathbb{N}$, $\mathbf{t}_k$ comme étant un élément du sous-différentiel de h au point $\mathbf{x}_{k+1}$:

$$(\forall k \in \mathbb{N}) \quad \mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k \mathbf{t}_k, \quad \text{où } \mathbf{t}_k \in \partial \mathsf{h}(\mathbf{x}_{k+1}), \tag{2.67}$$

ce qui peut sembler contre-intuitif puisque $\mathbf{x}_{k+1}$ n'est pas encore construit à l'itération k. De façon équivalente, l'équation (2.67) se réécrit

$$(\forall k \in \mathbb{N}) \quad \mathbf{x}_k - \mathbf{x}_{k+1} \in \gamma_k \partial \mathsf{h}(\mathbf{x}_{k+1}). \tag{2.68}$$

Autrement dit, d'après la propriété (2.45), pour trouver l'itérée $\mathbf{x}_{k+1}$ telle que (2.67) soit satisfaite, il suffit de la définir comme étant l'opérateur proximal de $\gamma_k \mathsf{h}$ en $\mathbf{x}_k$. Ainsi on obtient l'*algorithme du point proximal* [Rockafellar, 1976] donné par l'algorithme 2.3.

Algorithme 2.3 Algorithme du point proximal

Initialisation : Soient $\mathbf{x}_0 \in \mathcal{H}$ et, pour tout $k \in \mathbb{N}$, $\gamma_k \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$\quad \mathbf{x}_{k+1} \in \operatorname{prox}_{\gamma_k \mathsf{h}}(\mathbf{x}_k)$.

Lorsque h est convexe, rappelons que l'opérateur proximal de h est univalué. Un des résultats de convergence les plus généraux est alors donné par le théorème suivant :

Théorème 2.6. [Bauschke et Combettes, 2011, Thm. 27.1] *Soit* $\mathsf{h} \in \Gamma_0(\mathcal{H})$ *telle que* $\operatorname{Argmin} \mathsf{h} \neq \varnothing$. *Supposons que* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *est une suite générée par l'algorithme 2.3, et que* $(\gamma_k)_{k\in\mathbb{N}}$ *soit telle que* $\sum_{k\in\mathbb{N}} \gamma_k = +\infty$.*La suite* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *converge alors vers une solution du problème 2.65.*

Remarquons que ce résultat de convergence autorise l'emploi de pas $(\gamma_k)_{k\in\mathbb{N}}$ ne convergeant pas vers 0.

Il a été démontré récemment dans [Attouch *et al.*, 2011] que, lorsque $\mathcal{H} = \mathbb{R}^N$, la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ générée par l'algorithme 2.3 converge lorsque l'opérateur proximal est calculé de façon approchée et la fonction h n'est pas nécessairement convexe mais satisfait l'inégalité de KL :

Théorème 2.7. [Attouch *et al.*, 2011, Thm. 4.2] *Soit* $\mathsf{h}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *une fonction propre, s.c.i., et bornée inférieurement, satisfaisant l'inégalité de KL. On suppose de plus que* h *est continue sur son domaine. Si* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *est une suite bornée générée par l'algorithme 2.3, et* $(\gamma_k)_{k\in\mathbb{N}}$ *est tel que* $\underline{\gamma} \leqslant \gamma_k \leqslant \overline{\gamma}$, *pour* $(\underline{\gamma}, \overline{\gamma}) \in]0, +\infty[$, *alors*

(i) *la suite* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *converge vers un point critique* $\widehat{\mathbf{x}}$ *de* h ;

(ii) *la suite* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *est de longueur finie, i.e.*

$$\sum_{k\in\mathbb{N}} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| < +\infty. \tag{2.69}$$

2.4.2 Algorithme explicite-implicite

Algorithme

Dans la section précédente, nous avons présenté des algorithmes permettant de minimiser une fonction convexe, différentiable ou non. Nous allons maintenant nous intéresser au cas où la fonction à minimiser est éclatée en une somme de deux fonctions, l'une étant différentiable tandis que l'autre ne l'est pas nécessairement :

$$\text{trouver } \widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x}\in\mathcal{H}} \; (\mathsf{f}(\mathbf{x}) = \mathsf{h}(\mathbf{x}) + \mathsf{g}(\mathbf{x}))\,, \tag{2.70}$$

où $\mathsf{h} \in \Gamma_0(\mathcal{H})$ est différentiable de gradient β-Lipschitz, pour $\beta > 0$, $\mathsf{g} \in \Gamma_0(\mathcal{H})$, et $\mathcal{H}$ est un espace de Hilbert réel.

Proposition 2.13. [Combettes et Wajs, 2005, Prop. 3.1] *Soient* $\widehat{\mathbf{x}} \in \mathcal{H}$ *et* $\gamma \in]0, +\infty[$. *Les assertions suivantes sont équivalentes :*

(i) $\widehat{\mathbf{x}}$ *est une solution du problème* (2.70).

(ii) $\widehat{\mathbf{x}} = \operatorname{prox}_{\gamma\mathsf{g}}(\widehat{\mathbf{x}} - \gamma\nabla\mathsf{h}(\widehat{\mathbf{x}}))$.

D'après cette caractérisation de point fixe, une idée intuitive pour définir un algorithme itératif permettant de trouver une solution du problème (2.70) est :

$$(\forall k \in \mathbb{N}) \quad \mathbf{x}_{k+1} = \operatorname{prox}_{\gamma\mathsf{g}}(\mathbf{x}_k - \gamma\nabla\mathsf{h}(\mathbf{x}_k)), \tag{2.71}$$

pour un certain $\gamma > 0$ bien choisi. À chaque itération $k \in \mathbb{N}^*$, la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ est donc définie comme étant la composition d'une étape de gradient sur la fonction différentiable h (appelée étape *explicite*), et d'une étape proximale sur la fonction non lisse g (appelée étape *implicite*).

De manière plus rigoureuse, un algorithme permettant de résoudre le problème (2.70) est l'algorithme *explicite-implicite* (ou *forward-backward* en anglais) générant une suite de $\mathcal{H}$ notée $(\mathbf{x}_k)_{k\in\mathbb{N}}$, grâce aux itérations suivantes [Chen et Rockafellar, 1997; Tseng, 1998] :

Algorithme 2.4 Algorithme explicite-implicite

Initialisation : Soient $\mathbf{x}_0 \in \mathcal{H}$ et, pour tout $k \in \mathbb{N}$, $(\gamma_k, \lambda_k) \in]0, +\infty[^2$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} \mathbf{y}_k \in \operatorname{prox}_{\gamma_k \mathsf{g}}(\mathbf{x}_k - \gamma_k \nabla \mathsf{h}(\mathbf{x}_k)), \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k(\mathbf{y}_k - \mathbf{x}_k). \end{array} \right.$$

La suite des itérées $(\mathbf{x}_k)_{k\in\mathbb{N}}$ générée par l'algorithme 2.4 converge vers une solution du problème (2.70) :

Théorème 2.8. [Combettes et Wajs, 2005, Thm. 3.4] *Supposons que* $\mathsf{h} \in \Gamma_0(\mathcal{H})$ *est différentiable et de gradient* β*-Lipschitz, pour* $\beta > 0$*, et* $\mathsf{g} \in \Gamma_0(\mathcal{H})$*. Supposons de plus que* $\operatorname{Argmin}(\mathsf{h}+\mathsf{g}) \neq \varnothing$*. Soit* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *une suite générée par l'algorithme 2.4. Si*

- $0 < \inf_{l\in\mathbb{N}} \gamma_l \leqslant \sup_{l\in\mathbb{N}} \gamma_l < 2\beta^{-1}$,
- $(\forall k \in \mathbb{N})\ 0 < \inf_{l\in\mathbb{N}} \lambda_l \leqslant \lambda_k \leqslant 1$,

alors, $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *converge vers une solution du problème* (2.70)*.*

Récemment, dans le cas où $\mathcal{H} = \mathbb{R}^N$, les propriétés de convergence de l'algorithme explicite-implicite ont été étendues au cas où les fonctions h et g sont non nécessairement convexes dans [Attouch et Bolte, 2009; Attouch *et al.*, 2011], lorsque $\lambda_k \equiv 1$. Ces résultats de convergence reposent principalement sur l'hypothèse que la fonction f satisfait l'inégalité de KL :

Théorème 2.9. [Attouch *et al.*, 2011, Thm. 5.] *Supposons que* f *est une fonction propre, semi-continue inférieurement, bornée inférieurement, satisfaisant l'inégalité de KL. Supposons de plus que* $\mathsf{h} : \mathbb{R}^N \to \mathbb{R}$ *est différentiable, de gradient* β*-Lipschitz, pour* $\beta > 0$*, et que* $\mathsf{g} \colon \mathbb{R}^N \to]-\infty, +\infty]$ *est continue sur son domaine. Si* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *est une suite bornée générée par l'algorithme 2.4,* $\lambda_k \equiv 0$*, et* $(\gamma_k)_{k\in\mathbb{N}}$ *vérifie* $0 < \inf_{\ell\in\mathbb{N}} \gamma_\ell \leqslant \sum_{\ell\in\mathbb{N}} \gamma_\ell < \beta^{-1}$*, alors*

(i) *la suite* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *converge vers un point critique* $\widehat{\mathbf{x}}$ *de* f *;*

(ii) *la suite* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *est de longueur finie, i.e.*

$$\sum_{k\in\mathbb{N}} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| < +\infty. \tag{2.72}$$

Versions inexactes de l'algorithme explicite-implicite

En pratique, l'opérateur proximal de la fonction g n'a pas nécessairement une forme explicite et doit être calculé de façon itérative. Il existe plusieurs méthodes permettant de prendre en compte des termes d'erreurs éventuels.

L'une d'elles, donnée dans l'algorithme 2.5, consiste à considérer des termes d'erreurs sommables.

Algorithme 2.5 Algorithme explicite-implicite inexact (erreurs additives)

Initialisation : Soient $\mathbf{x}_0 \in \mathcal{H}$ et, pour tout $k \in \mathbb{N}$, $(\gamma_k, \lambda_k) \in]0, +\infty[^2$, $\mathbf{a}_k \in \mathcal{H}$ et $\mathbf{b}_k \in \mathcal{H}$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} \mathbf{y}_k = \mathbf{x}_k + \lambda_k \left(\mathrm{prox}_{\lambda_k \mathsf{g}} \left(\mathbf{x}_k - \gamma_k \nabla \mathsf{h}(\mathbf{x}_k) + \mathbf{b}_k \right) + \mathbf{a}_k - \mathbf{x}_k \right), \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k (\mathbf{y}_k - \mathbf{x}_k). \end{array} \right.$$

Dans cet algorithme, pour tout $k \in \mathbb{N}$, $\mathbf{a}_k \in \mathcal{H}$ est une terme d'erreur pouvant apparaitre lors du calcul de l'opérateur proximal, et $\mathbf{b}_k \in \mathcal{H}$ est un terme d'erreur pouvant apparaitre lors du calcul du gradient. Il a été démontré dans [Combettes et Wajs, 2005] que, lorsque h et g sont des fonctions convexes, si $\sum_{k \in \mathbb{N}} \|\mathbf{a}_k\| < +\infty$ et $\sum_{k \in \mathbb{N}} \|\mathbf{b}_k\| < +\infty$, alors le théorème 2.8 reste valide.

Une autre méthode modélisant un calcul approché de l'opérateur proximal de la fonction g est donnée dans l'algorithme 2.6.

Algorithme 2.6 Algorithme explicite-implicite inexact (erreurs relatives)

Initialisation : Soient $\mathbf{x}_0 \in \mathbb{R}^N$, $\mathsf{a} \in]\beta, +\infty[$ et $\mathsf{b} \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} \text{trouver } \mathbf{x}_{k+1} \in \mathbb{R}^N \text{ et } \mathbf{r}_{k+1} \in \partial \mathsf{g}(\mathbf{x}_{k+1}) \text{ tels que} \\ \quad \mathsf{g}(\mathbf{x}_{k+1}) + \langle \mathbf{x}_{k+1} - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k) \rangle + \dfrac{\mathsf{a}}{2} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 \leqslant \mathsf{g}(\mathbf{x}_k), \\ \quad \|\mathbf{r}_{k+1} + \nabla \mathsf{h}(\mathbf{x}_k)\| \leqslant \mathsf{b} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|. \end{array} \right.$$

Dans cet algorithme, la première inégalité est appelée *condition de décroissance suffisante*, et la seconde est appelée *condition d'optimalité*. Il a été démontré dans [Attouch *et al.*, 2011] que le théorème 2.9 reste valide pour l'algorithme 2.6, lorsque l'on considère des fonctions h et g non nécessairement convexes.

Remarque 2.12. *L'algorithme 2.6 peut être interprété comme une version inexacte de l'algorithme 2.4. En effet, d'une part, par définition de l'opérateur proximal, on a*

$$(\forall k \in \mathbb{N}) \quad \mathsf{g}(\mathbf{x}_{k+1}) + \langle \mathbf{x}_{k+1} - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k) \rangle + \frac{1}{2\gamma_k} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 \leqslant \mathsf{g}(\mathbf{x}_k), \tag{2.73}$$

ainsi la condition de décroissance suffisante est satisfaite pour $\mathsf{a} = (\sup_{\ell \in \mathbb{N}} \gamma_\ell)^{-1}$ *(puisque* $\gamma_k^{-1} \geqslant (\sup_{\ell \in \mathbb{N}} \gamma_\ell)^{-1} > \beta$*, où* β *est la constante de Lipschitz de* $\nabla \mathsf{h}$*). D'autre part, d'après la caractérisation de l'opérateur proximal, il existe* $\mathbf{r}_{k+1} \in \partial \mathsf{g}(\mathbf{x}_{k+1})$ *tel que*

$$\mathbf{r}_{k+1} = -\nabla \mathsf{h}(\mathbf{x}_k) + \gamma_k^{-1}(\mathbf{x}_k - \mathbf{x}_{k+1}). \tag{2.74}$$

Ainsi, la condition d'optimalité est obtenue pour $\mathsf{b} = (\inf_{\ell \in \mathbb{N}} \gamma_\ell)^{-1}$.

2.4.3 Principe de Majoration-Minimisation

Définitions et algorithme

Revenons au problème d'optimisation (2.65), où h est supposée différentiable. Un autre algorithme permettant d'aborder ce problème est l'*algorithme de Majoration-Minimisation* (MM), qui consiste à trouver une solution du problème (2.65) de façon itérative en minimisant à chaque itération $k \in \mathbb{N}$ une approximation *tangente majorante* de h.

Définition 2.21. *Soit* E *un sous-ensemble non vide de* $\mathbb{R}^N$*. Soient* $\mathsf{h} \colon \mathbb{R}^N \to]-\infty, +\infty]$ *et* $\mathbf{y} \in E$*. La fonction* $\mathsf{q} \colon \mathbb{R}^N \times \mathbb{R}^N \to]-\infty, +\infty]$ *est une* approximation tangente majorante *de la fonction* h *en* $\mathbf{y}$ *sur* E *si*

$$\begin{cases} (\forall \mathbf{x} \in E) \quad \mathsf{h}(\mathbf{x}) \leqslant \mathsf{q}(\mathbf{x}, \mathbf{y}) \\ \mathsf{h}(\mathbf{y}) = \mathsf{q}(\mathbf{y}, \mathbf{y}). \end{cases} \tag{2.75}$$

L'algorithme général MM proposé dans [Ortega et Rheinboldt, 1970] (voir aussi [Hunter et Lange, 2004]) est donné par l'algorithme 2.7.

Algorithme 2.7 Algorithme de Majoration-Minimisation

Initialisation : Soit $\mathbf{x}_0 \in \mathbb{R}^N$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\mathbf{x}_{k+1} \in \operatorname*{Argmin}_{\mathbf{x} \in \mathbb{R}^N} \mathsf{q}(\mathbf{x}, \mathbf{x}_k).$$

Une illustration graphique de cet algorithme pour $N = 1$ est donnée dans la figure 2.14.

L'avantage de cet algorithme est de pouvoir remplacer un problème d'optimisation qui peut être difficile à résoudre par une suite de problèmes plus simples, à partir du moment où la fonction q est facile à minimiser. Pour s'assurer que ce soit le cas, l'approximation tangente majorante q est souvent choisie quadratique :

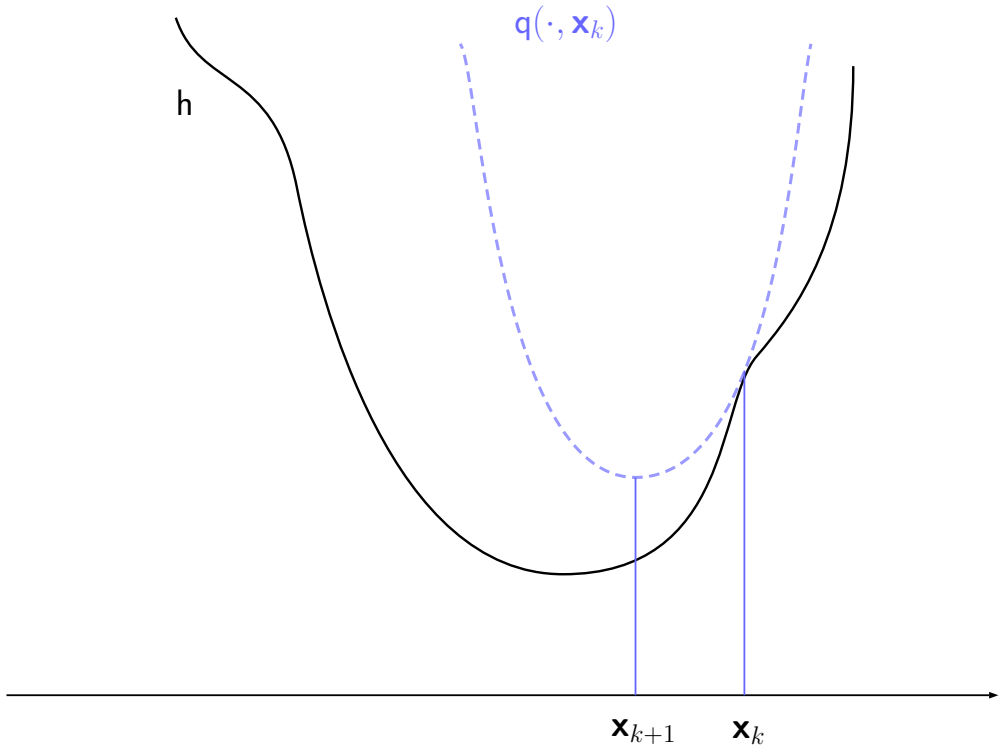


Figure 2.14 – *Illustration de l'algorithme de Majoration-Minimisation pour la minimisation d'une fonction* $\mathsf{h}\colon \mathbb{R} \to]-\infty, +\infty]$. *À une itération donnée* $k \in \mathbb{N}$, *on construit une majorante* $\mathsf{q}(\cdot, \mathbf{x}_k)$ *de* h *au point* $\mathbf{x}_k$, *puis on définit l'itérée* $\mathbf{x}_{k+1}$ *comme étant le minimiseur de cette majorante.*

Définition 2.22. *Soit* E *un sous-ensemble non vide de* $\mathbb{R}^N$. *Soient* $\mathsf{h}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *et* $\mathbf{y} \in E$. *La fonction quadratique définie par*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}(\mathbf{x}, \mathbf{y}) := \mathsf{h}(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla \mathsf{h}(\mathbf{y}) \rangle + \frac{1}{2} \|\mathbf{x} - \mathbf{y}\|^2_{\mathbf{A}(\mathbf{y})}, \tag{2.76}$$

où $\mathbf{A}(\mathbf{y}) \in \mathcal{S}_N^+$, *est une approximation majorante quadratique de* h *en* $\mathbf{y}$ *sur* E *si*

$$(\forall \mathbf{x} \in E) \quad \mathsf{h}(\mathbf{x}) \leqslant \mathsf{q}(\mathbf{x}, \mathbf{y}). \tag{2.77}$$

Remarquons que l'utilisation d'une majorante quadratique strictement convexe permet d'assurer l'unicité des solutions des sous-problèmes de l'algorithme 2.7. Dans ce cas, la minimisation de $\mathsf{q}(\cdot, \mathbf{x}_k)$, pour tout $k \in \mathbb{N}$, dans l'algorithme MM a une forme explicite, donnée dans l'algorithme 2.8.

Algorithme 2.8 Algorithme de Majoration-Minimisation Quadratique

Initialisation : Soit $\mathbf{x}_0 \in \mathbb{R}^N$.

Itérations :

Pour $k = 0, 1, \ldots$

$\quad \mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{A}(\mathbf{x}_k)^{-1} \nabla \mathsf{h}(\mathbf{x}_k)$.

Lorsque h est de gradient Lipschitz, l'existence de la matrice $\mathbf{A}(\mathbf{y})$ satisfaisant (2.76)-(2.77) est assurée par le lemme suivant :

Lemme 2.1. *Soit E un sous-ensemble convexe non vide de $\mathbb{R}^N$. Soient $\mathsf{h} \colon \mathbb{R}^N \to \mathbb{R}$ une fonction différentiable de gradient β-Lipschitz sur E, et $\mathbf{A}_k(\mathbf{y}) = \alpha\, \mathbf{I}_N$, où $\alpha \geqslant \beta$. La fonction $\mathsf{q} \colon \mathbb{R}^N \times \mathbb{R}^N \to]-\infty, +\infty]$ définie par (2.76) est une approximation majorante quadratique de h en $\mathbf{y}$ sur E.*

Démonstration. La fonction h étant de gradient β-Lipschitz sur l'ensemble convexe E, on peut appliquer le *Lemme de Descente* [Bertsekas, 1999, Prop.A.24], qui s'exprime par :

$$(\forall (\mathbf{x}, \mathbf{y}) \in E^2) \quad \mathsf{h}(\mathbf{x}) \leqslant \mathsf{h}(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla \mathsf{h}(\mathbf{y}) \rangle + \frac{\beta}{2} \|\mathbf{x} - \mathbf{y}\|^2.$$

Par conséquent, lorsque $\mathbf{A}_k(\mathbf{y}) = \alpha\, \mathbf{I}_N$, l'inégalité (2.77) est satisfaite. □

Le lemme précédent permet d'assurer l'existence des matrices $(\mathbf{A}(\mathbf{x}_k))_{k \in \mathbb{N}}$ et de les choisir facilement pour effectuer les itérations de l'algorithme 2.7. Cependant, intuitivement, à chaque itération $k \in \mathbb{N}$, plus la majorante quadratique $\mathsf{q}(\cdot, \mathbf{x}_k)$ sera proche de la fonction h dans le voisinage de $\mathbf{x}_k$, meilleure sera l'itérée suivante $\mathbf{x}_{k+1}$. En particulier, lorsque la fonction h est deux fois différentiable, ceci équivaut à choisir $\mathbf{A}(\mathbf{x}_k)$ proche de la matrice hessienne $\nabla^2 \mathsf{h}(\mathbf{x}_k)$, pour tout $k \in \mathbb{N}$.

Des techniques utiles de construction de telles matrices sont proposées dans [De Pierro, 1995; Erdogan et Fessler, 1999; Geman et Reynold, 1992; Geman et Yang, 1995; Hunter et Lange, 2004; Idier, 2001; Lange et Fessler, 1995; Sotthivirat et Fessler, 2002; Zheng *et al.*, 2004] pour certaines classes particulières de fonctions h. Nous en proposerons également dans les chapitres 4 et 6.

Par ailleurs, notons qu'il existe des algorithmes de Majoration-Minimisation faisant intervenir des majorantes non nécessairement quadratiques [Bolte et Pauwels, 2014; Fuchs, 2007; Ochs *et al.*, 2015].

Lien avec les algorithmes semi-quadratiques

On s'intéresse au cas où la fonction h est de la forme

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}(\mathbf{x}) = \frac{1}{2}\|\mathbf{H}\mathbf{x} - \mathbf{z}\|^2 + \sum_{p=1}^{P} \phi([\mathbf{V}\mathbf{x}]^{(p)}), \tag{2.78}$$

où $\mathbf{z} \in \mathbb{R}^M$, $\mathbf{H} \in \mathbb{R}^{M \times N}$, $\mathbf{V} \in \mathbb{R}^{P \times N}$ et $\phi \colon \mathbb{R} \to]-\infty, +\infty]$ est une fonction continument différentiable. On considère les majorantes quadratiques définies dans la proposition suivante :

Proposition 2.14. [Allain *et al.*, 2006; Chan et Mulet, 1999] *Soit* $\mathbf{y} \in \mathbb{R}^N$. *On considère la fonction quadratique*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathbf{q}(\mathbf{x}, \mathbf{y}) = \mathbf{h}(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla \mathbf{h}(\mathbf{y}) \rangle + \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|^2_{\mathbf{A}(\mathbf{y})}.$$

(i) *Si* ϕ *est de gradient Lipschitz, alors* $\mathsf{q}(\cdot, \mathbf{y})$ *est une fonction majorante de* h *en* $\mathbf{y}$ *pour*

$$\mathbf{A}(\mathbf{y}) = \mathbf{H}^\top \mathbf{H} + \frac{\mathbf{V}^\top \mathbf{V}}{\alpha},$$

avec $0 < \alpha < \dfrac{1}{\beta}$.

(ii) *Si* ϕ *est paire,* $\phi(\sqrt{\cdot})$ *est concave sur* $[0, +\infty[$, *et, pour tout* $\mathsf{t} \in \mathbb{R}$, $0 < \dot{\phi}(\mathsf{t})/\mathsf{t} < +\infty$, *alors* $\mathsf{q}(\cdot, \mathbf{y})$ *est une fonction majorante de* h *en* $\mathbf{y}$ *pour*

$$\mathbf{A}(\mathbf{y}) = \mathbf{H}^\top \mathbf{H} + \mathbf{V}^\top \operatorname{Diag}(\omega(\mathbf{V}\mathbf{y}))\, \mathbf{V},$$

avec, pour tout $\mathsf{t} \in \mathbb{R}$, $\omega(\mathsf{t}) = \dot{\phi}(\mathsf{t})/\mathsf{t}$.

L'algorithme 2.8 peut alors être interprété comme un algorithme semi-quadratique en remplaçant à chaque itération $k \in \mathbb{N}$, la matrice $\mathbf{A}(\mathbf{x}_k)$ par :

- la matrice donnée dans la proposition 2.14(i), on retrouve alors l'algorithme semi-quadratique de Geman & Yang [Geman et Yang, 1995].
- la matrice donnée dans la proposition 2.14(ii), on retrouve alors l'algorithme semi-quadratique de Geman & Reynolds [Geman et Reynold, 1992].

Le résultat de la proposition 2.14(ii) se généralise [Allain *et al.*, 2006; Chouzenoux *et al.*, 2011, 2013] à la minimisation de fonctions de la forme

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}(\mathbf{x}) = \psi(\mathbf{H}\mathbf{x} - \mathbf{z}) + \|\mathbf{V}_0\mathbf{x}\|^2 + \sum_{s=1}^{S} \phi_s(\|\mathbf{V}_s\mathbf{x} - \mathbf{c}_s\|), \tag{2.79}$$

où $\mathbf{z} \in \mathbb{R}^M$, $\mathbf{H} \in \mathbb{R}^{M \times N}$, $\psi \colon \mathbb{R}^M \to \mathbb{R}$, $\mathbf{V}_0 \in \mathbb{R}^{P_0 \times N}$, et, pour tout $s \in \{1, \ldots, S\}$, $\mathbf{V}_s \in \mathbb{R}^{P_s \times N}$, $\mathbf{c}_s \in \mathbb{R}^{P_s}$, et $\phi_s \colon \mathbb{R} \to \mathbb{R}$. On suppose que les assertions suivantes sont satisfaites :

- ψ est différentiable, de gradient β-Lipschitz,
- $(\forall s \in \{1, \ldots, S\})$ ϕ_s est différentiable,
- $(\forall s \in \{1, \ldots, S\})$ $\phi_s(\sqrt{\cdot})$ est concave sur $[0, +\infty[$,

- $(\forall s \in \{1,\dots,S\})$ il existe $\overline{\omega}_s \in [0,+\infty[$ tel que $(\forall \mathsf{t} \in]0,+\infty[)\ 0 \leqslant \omega_s(\mathsf{t}) \leqslant \overline{\omega}_s$, où $\omega_s(\mathsf{t}) = \dot{\phi}_s(\mathsf{t})/\mathsf{t}$,
- $\lim\limits_{\mathsf{t}\to 0, \mathsf{t}\neq 0} \omega_s(\mathsf{t}) \in \mathbb{R}$.

Pour tout $\mathbf{y} \in \mathbb{R}^N$, $\mathsf{q}(\cdot,\mathbf{y})$ est alors une fonction majorante quadratique de h en $\mathbf{y}$ pour

$$\mathbf{A}(\mathbf{y}) = \alpha \mathbf{H}^\top \mathbf{H} + 2\mathbf{V}_0^\top \mathbf{V}_0 + \mathbf{V}^\top \operatorname{Diag}(\mathbf{b}(\mathbf{y}))\, \mathbf{V}, \tag{2.80}$$

où $\alpha \in [\beta,+\infty[$, $\mathbf{V} = [\mathbf{V}_1|\dots|\mathbf{V}_S]$ et $\mathbf{b}(\mathbf{y}) = (\mathsf{b}_i(\mathbf{y}))_{1\leqslant i\leqslant SP}$ avec $P = \sum_{s=1}^S P_s$ et $(\forall i \in \{1,\dots,SP\})$ $\mathsf{b}_i(\mathbf{y}) \in \mathbb{R}$ est défini par

$$(\forall s \in \{1,\dots,S\})(\forall p \in \{1,\dots,P_s\}) \quad \mathsf{b}_{P_1+\dots+P_{s-1}+p}(\mathbf{y}) = \omega_s(\|\mathbf{V}_s\mathbf{y} - \mathbf{c}_s\|). \tag{2.81}$$

Lien avec l'algorithme explicite-implicite

Comme souligné dans [Beck et Teboulle, 2009; Chaâri *et al.*, 2011], notons qu'il existe un lien étroit entre l'algorithme MM et l'algorithme explicite-implicite.

Considérons l'algorithme 2.4 où les fonctions h et g ne sont pas nécessairement convexes, et $\lambda_k \equiv 1$. En utilisant la définition de l'opérateur proximal, on a

$$\begin{aligned}
(\forall k \in \mathbb{N}) \quad \mathbf{x}_{k+1} &\in \operatorname{prox}_{\gamma_k \mathsf{g}}(\mathbf{x}_k - \gamma_k \nabla \mathsf{h}(\mathbf{x}_k)) \\
&= \operatorname*{Argmin}_{\mathbf{x}\in\mathbb{R}^N}\ \mathsf{g}(\mathbf{x}) + \frac{1}{2\gamma_k}\|\mathbf{x} - \mathbf{x}_k + \gamma_k \nabla \mathsf{h}(\mathbf{x}_k)\|^2 \\
&= \operatorname*{Argmin}_{\mathbf{x}\in\mathbb{R}^N}\ \mathsf{g}(\mathbf{x}) + \frac{1}{2\gamma_k}\Big(\|\mathbf{x} - \mathbf{x}_k\|^2 + 2\gamma_k\langle \mathbf{x} - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k)\rangle + \|\gamma_k \nabla \mathsf{h}(\mathbf{x}_k)\|^2\Big) \\
&= \operatorname*{Argmin}_{\mathbf{x}\in\mathbb{R}^N}\ \mathsf{g}(\mathbf{x}) + \underbrace{\mathsf{h}(\mathbf{x}) + \langle \mathbf{x} - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k)\rangle + \frac{1}{2\gamma_k}\|\mathbf{x} - \mathbf{x}_k\|^2}_{:=\mathsf{q}_{\gamma_k}(\mathbf{x},\mathbf{x}_k)}.
\end{aligned} \tag{2.82}$$

D'après le théorème 2.9, si

$$(\forall k \in \mathbb{N}) \quad 0 < \inf_{\ell\in\mathbb{N}} \gamma_\ell \leqslant \gamma_k \leqslant \sup_{\ell\in\mathbb{N}} \gamma_\ell < \beta^{-1}, \tag{2.83}$$

où β est la constante de Lipschitz de h, alors la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ converge vers un point critique de $\mathsf{h}+\mathsf{g}$. Or, sous cette hypothèse, on a, pour tout $k \in \mathbb{N}$, $\gamma_k^{-1} > \beta$, et donc

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}_{\gamma_k}(\mathbf{x},\mathbf{x}_k) \geqslant \mathsf{h}(\mathbf{x}) + \langle \mathbf{x} - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k)\rangle + \frac{\beta}{2}\|\mathbf{x} - \mathbf{x}_k\|^2. \tag{2.84}$$

Donc, d'après le lemme 2.1, la fonction $\mathbf{x} \in \mathbb{R}^N \mapsto \mathsf{q}_{\gamma_k}(\mathbf{x},\mathbf{x}_k)$ majore la fonction h en $\mathbf{x}_k$:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}_{\gamma_k}(\mathbf{x},\mathbf{x}_k) \geqslant \mathsf{h}(\mathbf{x}). \tag{2.85}$$

Dans le chapitre 3, nous nous servirons de cette interprétation MM de l'algorithme explicite-implicite pour en accélérer la convergence.

2.4.4 Algorithmes primaux-duaux

Nous avons vu dans la section 2.2.4 que certaines fonctions de régularisation (par exemple, la variation totale) sont la composition d'une fonction non différentiable et d'un opérateur linéaire. Nous allons donc nous intéresser dans cette partie au problème de minimisation suivant :

$$\text{trouver } \widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x}\in\mathcal{H}} \ \mathsf{f}(\mathbf{x}) + \mathsf{h}(\mathbf{x}) + \mathsf{g}(\mathbf{L}\mathbf{x}), \tag{2.86}$$

où $\mathsf{f} \in \Gamma_0(\mathcal{H})$, $\mathsf{g} \in \Gamma_0(\mathcal{G})$, $\mathbf{L} \in \mathcal{B}(\mathcal{H},\mathcal{G})$, $\mathsf{h} \in \Gamma_0(\mathcal{H})$ est une fonction différentiable de gradient Lipschitz, $\mathcal{H}$ et $\mathcal{G}$ sont des espaces de Hilbert réels. Un tel problème peut être traité en utilisant, par exemple, l'algorithme explicite-implicite et en calculant l'opérateur proximal de $\mathsf{f} + \mathsf{g} \circ \mathbf{L}$ de façon approximative (à l'aide de sous-itérations). C'est ce que nous ferons dans la suite de cette thèse lorsque les fonctions ne sont pas nécessairement convexes. Cependant, lorsqu'elles sont convexes, ce problème peut être résolu en traitant à la fois le problème primal (2.86) et le problème dual correspondant [Komodakis et Pesquet, 2014] :

$$\text{trouver } \widehat{\mathbf{v}} \in \operatorname*{Argmin}_{\mathbf{v}\in\mathcal{G}} \ (\mathsf{f}^* \,\square\, \mathsf{h}^*)(-\mathbf{L}^*\mathbf{v}) + \mathsf{g}^*(\mathbf{v}). \tag{2.87}$$

Remarquons que les problèmes (2.86)-(2.87) peuvent être reformulés de façon à obtenir un problème de recherche de point selle du Lagrangien [Bauschke et Combettes, 2011, Chap. 19] :

$$\text{trouver } (\widehat{\mathbf{x}},\widehat{\mathbf{v}}) \in \operatorname*{Argmin}_{\mathbf{x}\in\mathcal{H}} \ \left(\operatorname*{Argmax}_{\mathbf{v}\in\mathcal{G}} \ \mathsf{h}(\mathbf{x}) + \mathsf{f}(\mathbf{x}) - \mathsf{g}^*(\mathbf{v}) + \langle \mathbf{L}\mathbf{x}, \mathbf{v}\rangle \right). \tag{2.88}$$

Ainsi, si le couple $(\widehat{\mathbf{x}},\widehat{\mathbf{v}})$ vérifie la condition de Karush-Kuhn-Tucker

$$-\mathbf{L}^*\widehat{\mathbf{v}} - \nabla\mathsf{h}(\widehat{\mathbf{x}}) \in \partial\mathsf{f}(\widehat{\mathbf{x}}) \quad \text{et} \quad \mathbf{L}\widehat{\mathbf{x}} \in \partial\mathsf{g}^*(\widehat{\mathbf{v}}), \tag{2.89}$$

alors $\widehat{\mathbf{x}}$ est une solution du problème primal (2.86), $\widehat{\mathbf{v}}$ est une solution du problème dual (2.87), et $(\widehat{\mathbf{x}},\widehat{\mathbf{v}})$ est une solution du problème (2.88).

D'autre part, dans la suite de cette section, nous supposerons que

(i) le problème (2.86) admet au moins une solution ;

(ii) $\operatorname{ri}(\operatorname{dom}\mathsf{g}) \cap \mathbf{L}(\operatorname{dom}\mathsf{f}) \neq \varnothing$ [4] (ou $\operatorname{dom}\mathsf{g} \cap \operatorname{ri}(\mathbf{L}(\operatorname{dom}\mathsf{f})) \neq \varnothing$) sont satisfaites.

Ces conditions permettent d'assurer l'existence d'une solution du problème dual (2.87) [Komodakis et Pesquet, 2014] (notons que d'autres conditions peuvent être considérées, c.f. [Bauschke et Combettes, 2011, Chap. 15]). De plus, l'égalité suivante sera satisfaite

$$\mathsf{f}(\widehat{\mathbf{x}}) + \mathsf{h}(\widehat{\mathbf{x}}) + \mathsf{g}(\mathbf{L}\widehat{\mathbf{x}}) = -\Big((\mathsf{f}^* \,\square\, \mathsf{h}^*)(-\mathbf{L}^*\widehat{\mathbf{v}}) + \mathsf{g}^*(\widehat{\mathbf{v}})\Big), \tag{2.90}$$

puisqu'il n'y a pas de saut de dualité.

4. $\operatorname{ri}(E)$ désigne l'intérieur relatif de l'ensemble E.

Les algorithmes proximaux primaux-duaux peuvent être décomposés en trois classes :

- des méthodes basées sur l'algorithme explicite-implicite [Chambolle et Pock, 2010; Condat, 2013; Pock *et al.*, 2009; Vũ, 2013] : ces méthodes alternent à chaque itération une étape de gradient et une étape proximale.
- des méthodes basées sur l'algorithme explicite-implicite-explicite [Boţ et Hendrich, 2014; Briceños Arias et Combettes, 2011; Combettes, 2013; Combettes et Pesquet, 2011] : en comparaison aux approches précédentes, ces méthodes ajoutent une étape de gradient supplémentaire à chaque itération. Ce sont cependant les premières méthodes primales-duales qui ont été proposées dans la littérature permettant de résoudre (2.86) en exploitant le caractère lisse de h.
- des méthodes basées sur des projections [Alotaibi *et al.*, 2014] : il s'agit de la classe la plus récente, dont le principal avantage, comparativement aux autres méthodes, est de ne pas avoir besoin de connaître la norme de l'opérateur linéaire $\mathbf{L}$.

Dans la suite, nous nous intéresserons à la première classe de ces méthodes.

Algorithme ADMM

Comme indiqué dans [Komodakis et Pesquet, 2014], l'algorithme ADMM (*Alternating Direction Method of Multipliers*) [Boyd *et al.*, 2011; Fortin et Glowinski, 1983; Gabay et Mercier, 1976] peut être vu comme un algorithme primal-dual. Cette méthode permet de résoudre les problèmes (2.86)-(2.87), lorsque $\mathbf{L}^*\mathbf{L}$ est un isomorphisme, grâce aux itérées rappelées dans l'algorithme 2.9.

Algorithme 2.9 Algorithme ADMM

Initialisation : Soient $(\mathbf{y}_0, \mathbf{z}_0) \in \mathcal{G} \times \mathcal{G}$ et $\gamma \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{aligned}
&\mathbf{x}_k = \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N} \frac{1}{2}\|\mathbf{L}\mathbf{x} - \mathbf{y}_k + \mathbf{z}_k\|^2 + \frac{1}{\gamma}\left(\mathsf{f}(\mathbf{x}) + \mathsf{h}(\mathbf{x})\right),\\
&\mathbf{s}_k = \mathbf{L}\mathbf{x}_k,\\
&\mathbf{y}_{k+1} = \operatorname{prox}_{\mathsf{g}/\gamma}(\mathbf{z}_k + \mathbf{s}_k),\\
&\mathbf{z}_{k+1} = \mathbf{z}_k + \mathbf{s}_k - \mathbf{y}_{k+1}.
\end{aligned}$$

On a alors le résultat de convergence suivant :

Théorème 2.10. [Komodakis et Pesquet, 2014] *Supposons que $\mathbf{L}^*\mathbf{L}$ soit un isomorphisme. La suite $(\mathbf{x}_k, \gamma\mathbf{z}_k)_{k\in\mathbb{N}}$ générée par l'algorithme 2.9 converge alors faiblement vers un couple de solutions $(\widehat{\mathbf{x}}, \widehat{\mathbf{v}})$ des problèmes* (2.86)-(2.87).

Remarquons que cet algorithme est équivalent à l'*algorithme de Douglas-Rachford* [Combettes et Pesquet, 2007a; Eckstein et Bertsekas, 1992] lorsque ce dernier est appliqué au problème dual (2.87).

Bien que cet algorithme soit beaucoup utilisé en traitement du signal [Afonso *et al.*, 2011; Figueiredo et Bioucas-Dias, 2010; Figueiredo et Nowak, 2009; Giovannelli et Coulais, 2005; Goldstein et Osher, 2009; Tran-Dinh *et al.*, 2014], en pratique, il peut être difficile à mettre en œuvre. En effet, lorsque la matrice $\mathbf{L}$ est de grande taille et/ou n'a pas une structure simple, le calcul de $\mathbf{x}_k$ à l'itération k peut se révéler coûteux. D'autre part, la différentiabilité de la fonction h n'est pas exploitée de façon explicite dans cette méthode.

Algorithme primal-dual de Condat-Vũ

Une autre méthode permettant de minimiser à la fois le problème primal (2.86) et le problème dual (2.87) est l'algorithme de Condat-Vũ proposé dans [Condat, 2013; Vũ, 2013]. Ses itérées sont données dans l'algorithme 2.10.

Algorithme 2.10 Algorithme primal-dual [Condat, 2013; Vũ, 2013]

Initialisation : Soient $(\mathbf{x}_0, \mathbf{v}_0) \in \mathcal{H} \times \mathcal{G}$, $(\sigma, \tau) \in]0, +\infty[^2$ et, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{bmatrix} \mathbf{y}_k = \operatorname{prox}_{\tau \mathsf{f}} \Big(\mathbf{x}_k - \tau(\nabla \mathsf{h}(\mathbf{x}_k) + \mathbf{b}_k + \mathbf{L}^* \mathbf{v}_k) \Big) + \mathbf{a}_k, \\ \mathbf{u}_k = \operatorname{prox}_{\sigma \mathsf{g}^*} \Big(\mathbf{v}_k + \sigma \mathbf{L}(2\mathbf{y}_k - \mathbf{x}_k) \Big) + \mathbf{c}_k, \\ \mathbf{x}_{k+1} = \lambda_k \mathbf{y}_k + (1 - \lambda_k) \mathbf{x}_k, \\ \mathbf{v}_{k+1} = \lambda_k \mathbf{u}_k + (1 - \lambda_k) \mathbf{v}_k. \end{bmatrix}$$

Remarquons que, contrairement à l'algorithme 2.9 présenté dans le paragraphe précédent, ici la matrice $\mathbf{L}$ n'a pas besoin de satisfaire de condition particulière.

Dans l'algorithme 2.10, pour tout $k \in \mathbb{N}$, $\mathbf{a}_k$, $\mathbf{b}_k$ et $\mathbf{c}_k$ représentent des termes d'erreurs pouvant apparaitre lors des calculs de l'opérateur proximal de $\tau \mathsf{f}$, du gradient de h et de l'opérateur proximal de $\sigma \mathsf{g}^*$. Remarquons que, lorsque $\mathbf{L} = \mathbf{0}$, et g^* est identiquement nulle, alors on retrouve l'algorithme explicite-implicite vu dans la section 2.4.2.

La convergence de la suite $(\mathbf{x}_k, \mathbf{v}_k)_{k \in \mathbb{N}}$ générée par l'algorithme 2.10 est assurée par le théorème suivant :

Théorème 2.11. [Condat, 2013, Thm. 3.1] *Supposons que les assertions suivantes soient vérifiées :*

(i) $\tau^{-1} - \sigma \|\mathbf{L}\|^2 > \beta/2$, *où* β *est la constante de Lipschitz de* h,

(ii) *pour tout* $k \in \mathbb{N}$, $\lambda_k \in]0, \overline{\lambda}[$, *où* $\overline{\lambda} = 2 - \beta(\tau^{-1} - \sigma\|\mathbf{L}\|^2)^{-1}/2 \in [1, 2[$,

(iii) $\sum_{k \in \mathbb{N}} \lambda_k (\overline{\lambda} - \lambda_k) = +\infty$,

(iv) $\sum_{k \in \mathbb{N}} \lambda_k \|\mathbf{a}_k\| < +\infty$, $\sum_{k \in \mathbb{N}} \lambda_k \|\mathbf{b}_k\| < +\infty$, *et* $\sum_{k \in \mathbb{N}} \lambda_k \|\mathbf{c}_k\| < +\infty$.

La suite $(\mathbf{x}_k, \mathbf{v}_k)_{k \in \mathbb{N}}$ *générée par l'algorithme 2.10 converge alors faiblement vers un couple de solutions* $(\widehat{\mathbf{x}}, \widehat{\mathbf{v}})$ *des problèmes* (2.86)-(2.87).

Notons que le problème primal (2.86) et le problème dual (2.87) ont une structure symétrique. Ainsi, une alternative à l'algorithme 2.10, ayant les mêmes propriétés de convergence que ce dernier, est obtenue en intervertissant les variables primales et duales. Cette méthode est donnée dans l'algorithme 2.11.

Algorithme 2.11 Algorithme primal-dual (forme symétrique) [Condat, 2013; Vũ, 2013]

Initialisation : Soient $(\mathbf{x}_0, \mathbf{v}_0) \in \mathbb{R}^N \times \mathbb{R}^M$, $(\sigma, \tau) \in]0, +\infty[^2$ et, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{bmatrix} \mathbf{u}_k = \mathrm{prox}_{\sigma \mathsf{g}^*}\Big(\mathbf{v}_k + \sigma \mathbf{L}\mathbf{x}_k\Big) + \mathbf{c}_k, \\ \mathbf{y}_k = \mathrm{prox}_{\tau \mathsf{f}}\Big(\mathbf{x}_k - \tau\Big(\nabla \mathsf{h}(\mathbf{x}_k) + \mathbf{b}_k + \mathbf{L}^*(2\mathbf{u}_k - \mathbf{v}_k)\Big)\Big) + \mathbf{a}_k, \\ \mathbf{x}_{k+1} = \lambda_k \mathbf{y}_k + (1 - \lambda_k)\mathbf{x}_k, \\ \mathbf{v}_{k+1} = \lambda_k \mathbf{u}_k + (1 - \lambda_k)\mathbf{v}_k. \end{bmatrix}$$

On peut remarquer que l'algorithme présenté dans [Chambolle et Pock, 2010; Pock *et al.*, 2009] est un cas particulier de l'algorithme 2.10. En effet, lorsque $\mathsf{h} \equiv 0$, et les termes d'erreurs $(\mathbf{a}_k)_{k\in\mathbb{N}}$, $(\mathbf{b}_k)_{k\in\mathbb{N}}$ et $(\mathbf{c}_k)_{k\in\mathbb{N}}$ sont tous nuls, l'algorithme 2.10 se réduit à l'algorithme 2.12.

Algorithme 2.12 Algorithme primal-dual simplifié [Chambolle et Pock, 2010]

Initialisation : Soient $(\mathbf{x}_0, \mathbf{v}_0) \in \mathcal{H} \times \mathcal{G}$, $(\sigma, \tau) \in]0, +\infty[^2$ et, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{bmatrix} \mathbf{y}_k = \mathrm{prox}_{\tau \mathsf{f}}\Big(\mathbf{x}_k - \tau \mathbf{L}^* \mathbf{v}_k\Big), \\ \mathbf{u}_k = \mathrm{prox}_{\sigma \mathsf{g}^*}\Big(\mathbf{v}_k + \sigma \mathbf{L}(2\mathbf{y}_k - \mathbf{x}_k)\Big), \\ \mathbf{x}_{k+1} = \lambda_k \mathbf{y}_k + (1 - \lambda_k)\mathbf{x}_k, \\ \mathbf{v}_{k+1} = \lambda_k \mathbf{u}_k + (1 - \lambda_k)\mathbf{v}_k. \end{bmatrix}$$

Une version de cet algorithme incluant des termes d'erreur pour le calcul des opérateurs proximaux est donné dans [Condat, 2013]. Notons de plus que, dans ce cas, les conditions de convergences sont moins restrictives que dans le théorème 2.11.

Théorème 2.12. [Condat, 2013, Thm. 3.3] *Supposons que les assertions suivantes soient vérifiées :*

(i) $\tau\sigma\|\mathsf{L}\|^2 \leqslant 1$

(ii) *pour tout* $k \in \mathbb{N}$, $\lambda_k \in [\varepsilon, 2-\varepsilon]$, *où* $\varepsilon > 0$.

La suite $(\mathbf{x}_k, \mathbf{v}_k)_{k\in\mathbb{N}}$ *générée par l'algorithme 2.12 converge alors faiblement vers un couple de solutions* $(\widehat{\mathbf{x}}, \widehat{\mathbf{v}})$ *des problèmes* (2.86)-(2.87).

De même, on peut réécrire l'algorithme 2.11 lorsque $\mathsf{h} \equiv 0$, les termes d'erreurs sont nuls et le paramètre de relaxation $\lambda_k \equiv 0$. On obtient alors l'algorithme 2.13, dont la convergence est assurée par le théorème 2.12.

Algorithme 2.13 Algorithme primal-dual simplifié (forme symétrique) [Chambolle et Pock, 2010]

Initialisation : Soient $(\mathbf{x}_0, \mathbf{v}_0) \in \mathcal{H} \times \mathcal{G}$, $(\sigma, \tau) \in]0, +\infty[^2$ et, pour tout $k \in \mathbb{N}$, $\gamma_k \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{bmatrix} \mathbf{v}_{k+1} = \mathrm{prox}_{\sigma \mathsf{g}^*}\Big(\mathbf{v}_k + \sigma \mathsf{L}\mathbf{x}_k\Big), \\ \mathbf{x}_{k+1} = \mathrm{prox}_{\tau \mathsf{f}}\Big(\mathbf{x}_k - \tau \mathsf{L}^*(2\mathbf{v}_{k+1} - \mathbf{v}_k)\Big). \end{bmatrix}$$

En utilisant le théorème de décomposition de Moreau, en effectuant le changement de variable $(\forall k \in \mathbb{N})$ $\mathbf{p}_k = \mathbf{v}_k/\sigma$ puis en réarrangeant les mises à jours des variables, on obtient alors l'algorithme 2.14 [Burger *et al.*, 2014; Esser *et al.*, 2010] lorsque $\theta = 1$ (lorsque $0 < \theta < 1$ on obtient une sous-relaxation de la variable $(\mathbf{z}_k)_{k\in\mathbb{N}}$ [Burger *et al.*, 2014]).

Algorithme 2.14 Algorithme primal-dual simplifié (forme symétrique modifiée) [Chambolle et Pock, 2010]

Initialisation : Soient $(\mathbf{x}_0, \mathbf{p}_0) \in \mathcal{H} \times \mathcal{G}$, $\mathbf{z}_0 = \mathbf{p}_0$, $(\tau, \sigma) \in]0, +\infty[^2$ et $\theta \in]0, 1]$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{bmatrix} \mathbf{x}_{k+1} = \mathrm{prox}_{\tau \mathsf{f}}(\mathbf{x}_k - \sigma\tau \mathsf{L}^* \mathbf{z}_k), \\ \mathbf{y}_{k+1} = \mathrm{prox}_{\mathsf{g}/\sigma}(\mathbf{p}_k + \mathsf{L}\mathbf{x}_{k+1}), \\ \mathbf{p}_{k+1} = \mathbf{p}_k + \mathsf{L}\mathbf{x}_{k+1} - \mathbf{y}_{k+1}, \\ \mathbf{z}_{k+1} = \mathbf{p}_{k+1} + \theta(\mathbf{p}_{k+1} - \mathbf{p}_k). \end{bmatrix}$$

On a alors le résultat de convergence suivant :

Corollaire 2.1. [Burger *et al.*, 2014] *Supposons que* $\tau\sigma\|\mathbf{L}\|^2 < 1$. *La suite* $(\mathbf{x}_k, \theta\mathbf{p}_k)_{k\in\mathbb{N}}$ *générée par l'algorithme 2.14 converge alors faiblement vers un couple de solutions* $(\widehat{\mathbf{x}}, \widehat{\mathbf{v}})$ *des problèmes* (2.86)-(2.87).

Nous utiliserons ce dernier algorithme dans le chapitre 8.

Par ailleurs, dans le chapitre 7, nous généraliserons ces algorithmes primaux-duaux au cadre stochastique. Nous revisiterons les conditions de convergence dans ce cadre plus général. Cela nous permettra de traiter une classe de problèmes plus large que (2.86)-(2.87).

2.5 Conclusion

Une méthode rigoureuse pour trouver une solution d'un problème inverse, est de la définir comme le minimiseur d'un problème variationnel faisant intervenir une fonction d'attache aux données et une fonction de régularisation. Comme nous l'avons vu dans ce chapitre, les fonctions à minimiser peuvent présenter des propriétés mathématiques différentes. Cela nécessite la construction d'algorithmes permettant d'exploiter ces différentes propriétés. En particulier, nous avons donné dans la section précédentes des algorithmes usuels permettant de minimiser des fonctions (non) différentiables et/ou (non) convexes.

Deux aspects importants sont à prendre en compte lors de l'utilisation de ces algorithmes pour des applications du traitement du signal et des images : leur vitesse de convergence et leur coût d'implémentation. L'objectif de cette thèse sera de proposer des algorithmes à la fois peu coûteux à implémenter et dont la convergence est assurée et rapide. Ainsi dans un premier temps nous proposerons dans le chapitre suivant une méthode d'accélération de l'algorithme explicite-implicite basée sur l'utilisation d'une métrique variable qui sera choisie grâce au principe MM.

2.A Démonstration de la propriété (v) de perturbation quadratique de l'opérateur proximal

Soient $\mathbf{x} \in \mathcal{H}$ et $\mathbf{p} = \operatorname{prox}_{\mathbf{A}+\mathbf{U},\mathsf{g}}\left((\mathbf{A}+\mathbf{U})^{-1}(\mathbf{U}\mathbf{x} - \beta\mathbf{x}_0)\right)$. D'après la définition 2.12 de l'opérateur proximal, on a

$$\mathbf{p} = \operatorname*{Argmin}_{\mathbf{y}\in\mathcal{H}} \ \mathsf{g}(\mathbf{y}) + \frac{1}{2}\mathsf{q}(\mathbf{y}),$$

où

$$\begin{aligned}\mathsf{q}(\mathbf{y}) &= \left\|\mathbf{y} - (\mathbf{A}+\mathbf{U})^{-1}(\mathbf{U}\mathbf{x} - \beta\mathbf{x}_0)\right\|^2_{\mathbf{A}+\mathbf{U}} \\ &= \left\langle \mathbf{y} - (\mathbf{A}+\mathbf{U})^{-1}(\mathbf{U}\mathbf{x} - \beta\mathbf{x}_0) \,\middle|\, (\mathbf{A}+\mathbf{U})\left(\mathbf{y} - (\mathbf{A}+\mathbf{U})^{-1}(\mathbf{U}\mathbf{x} - \beta\mathbf{x}_0)\right)\right\rangle.\end{aligned}$$

Afin de démontrer l'égalité (2.46), il suffit de montrer que

$$(\forall \mathbf{y} \in \mathcal{H}) \quad \mathsf{g}(\mathbf{y}) + \frac{1}{2}\mathsf{q}(\mathbf{y}) = \mathsf{f}(\mathbf{y}) + \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|_{\mathbf{U}}^2 + \kappa, \tag{2.91}$$

avec $\kappa \in \mathbb{R}$. Pour tout $\mathbf{y} \in \mathcal{H}$, on a

$$\begin{aligned}
\mathsf{q}(\mathbf{y}) &= \Big\langle \mathbf{y} - (\mathbf{A}+\mathbf{U})^{-1}\mathbf{U}\mathbf{x} + (\mathbf{A}+\mathbf{U})^{-1}\beta\mathbf{x}_0 \,\Big|\, (\mathbf{A}+\mathbf{U})\mathbf{y} - \mathbf{U}\mathbf{x} + \beta\mathbf{x}_0 \Big\rangle \\
&= \Big\langle \mathbf{y} - (\mathbf{A}+\mathbf{U})^{-1}\mathbf{U}\mathbf{x} \mid (\mathbf{A}+\mathbf{U})\mathbf{y} - \mathbf{U}\mathbf{x} \Big\rangle \\
&\quad + 2\beta\Big\langle (\mathbf{A}+\mathbf{U})^{-1}\mathbf{x}_0 \mid (\mathbf{A}+\mathbf{U})\mathbf{y} - \mathbf{U}\mathbf{x} \Big\rangle + \beta^2\Big\langle (\mathbf{A}+\mathbf{U})^{-1}\mathbf{x}_0 \mid \mathbf{x}_0 \Big\rangle \\
&= \Big\langle \mathbf{y} - (\mathbf{A}+\mathbf{U})^{-1}\mathbf{U}\mathbf{x} \mid (\mathbf{A}+\mathbf{U})\mathbf{y} - \mathbf{U}\mathbf{x} \Big\rangle + 2\beta\langle \mathbf{y} \mid \mathbf{x}_0 \rangle + \kappa_1,
\end{aligned} \tag{2.92}$$

où $\kappa_1 = -2\beta\big\langle (\mathbf{A}+\mathbf{U})^{-1}\mathbf{x}_0 \mid \mathbf{U}\mathbf{x}\big\rangle + \beta^2\big\langle (\mathbf{A}+\mathbf{U})^{-1}\mathbf{x}_0 \mid \mathbf{x}_0\big\rangle$ est une constante par rapport à la variable $\mathbf{y}$. Remarquons que

$$\begin{aligned}
&\Big\langle \mathbf{y} - (\mathbf{A}+\mathbf{U})^{-1}\mathbf{U}\mathbf{x} \mid (\mathbf{A}+\mathbf{U})\mathbf{y} - \mathbf{U}\mathbf{x} \Big\rangle \\
&= \Big\langle (\mathbf{y}-\mathbf{x}) + \Big(\mathbf{x} - (\mathbf{A}+\mathbf{U})^{-1}\mathbf{U}\mathbf{x}\Big) \,\Big|\, \mathbf{U}(\mathbf{y}-\mathbf{x}) + \mathbf{A}\mathbf{y} \Big\rangle \\
&= \|\mathbf{y}-\mathbf{x}\|_{\mathbf{U}}^2 + \|\mathbf{y}\|_{\mathbf{A}}^2 - \langle \mathbf{x} \mid \mathbf{A}\mathbf{y}\rangle + \Big\langle \mathbf{x} - (\mathbf{A}+\mathbf{U})^{-1}\mathbf{U}\mathbf{x} \mid (\mathbf{A}+\mathbf{U})\mathbf{y} \Big\rangle + \kappa_2,
\end{aligned} \tag{2.93}$$

où $\kappa_2 = \big\langle (\mathbf{A}+\mathbf{U})^{-1}\mathbf{U}\mathbf{x} - \mathbf{x} \mid \mathbf{U}\mathbf{x}\big\rangle$ est indépendant de $\mathbf{y}$. Remarquons de plus que

$$\begin{aligned}
\Big\langle \mathbf{x} - (\mathbf{A}+\mathbf{U})^{-1}\mathbf{U}\mathbf{x} \mid (\mathbf{A}+\mathbf{U})\mathbf{y} \Big\rangle &= \langle \mathbf{x} \mid \mathbf{A}\mathbf{y} + \mathbf{U}\mathbf{y}\rangle - \langle \mathbf{U}\mathbf{x} \mid \mathbf{y}\rangle \\
&= \langle \mathbf{x} \mid \mathbf{A}\mathbf{y}\rangle.
\end{aligned} \tag{2.94}$$

On peut alors déduire (2.91) en combinant les équations (2.92)-(2.94) et en posant $\kappa = \frac{1}{2}(\kappa_1 + \kappa_2) - \gamma$.

Chapitre 3

Algorithme explicite-implicite à métrique variable

Sommaire

3.1 Introduction

Dans ce chapitre, nous considérerons un problème de minimisation faisant intervenir une somme de deux fonctions dont l'une est différentiable de gradient Lipschitz, tandis que l'autre est convexe mais éventuellement non lisse. Dans ce cadre, une approche usuelle consiste à utiliser l'algorithme proximal explicite-implicite 2.4 [Chen et Rockafellar, 1997; Tseng, 1998]. Cette méthode alterne, à chaque itération, une étape de gradient sur la fonction différentiable et une étape proximale sur la seconde fonction. Récemment, la convergence de cet algorithme a été démontrée lorsque les fonctions à minimiser ne sont pas nécessairement convexes mais satisfont l'inégalité de KL (théorème 2.9).

Dans le contexte de problèmes de grande taille, tels que ceux rencontrés en restauration d'images, un des enjeux principaux est de pouvoir produire de bons résultats numériques en un temps de calcul raisonnable. L'algorithme explicite-implicite est caractérisé par un faible coût de calcul à chaque itération. Cependant, comme la plupart des méthodes de minimisation de premier ordre, il peut converger relativement lentement [Chen et Rockafellar, 1997]. Deux grandes familles de stratégies d'accélérations de cet algorithme se distinguent dans la littérature. La première repose sur une accélération de sous-espaces [Beck et Teboulle, 2009; Bioucas-Dias et Figueiredo, 2007; Kowalski, 2010; Nesterov, 2007; Ochs *et al.*, 2014] : c'est-à-dire que la vitesse de convergence va être accélérée en utilisant les informations obtenues lors des itérations précédentes pour construire la nouvelle estimée. La seconde stratégie d'accélération est basée sur l'introduction d'une métrique variant au cours des itérations [Becker et Fadili, 2012; Bonnans *et al.*, 1995; Burke et Qian, 1999; Chen et Rockafellar, 1997; Combettes et Vũ, 2014; Lotito *et al.*, 2009; Parente *et al.*, 2008]. Cette métrique est obtenue grâce à une matrice de préconditionnement. L'algorithme que l'on obtient alors est appelé algorithme explicite-implicite à métrique variable (ou *Variable Metric Forward-Backward* (VMFB)). La convergence de cet algorithme dans le cas où les fonctions à minimiser sont convexes a été démontrée dans [Combettes et Vũ, 2014; Lee *et al.*, 2012; Tran-Dinh *et al.*, 2014].

Dans ce chapitre, nous allons nous poser la question de l'extension de ces résultats de convergence au cadre non convexe. La démonstration de ces résultats reposera d'une part sur l'inégalité de KL, et d'autre part sur un choix original des matrices de préconditionnement basé sur une stratégie de Majoration-Minimisation. Enfin, comme nous l'avons vu dans le chapitre 2, l'opérateur proximal n'est pas toujours calculable de façon explicite. Nous rechercherons un algorithme robuste aux erreurs pouvant apparaître lors du calcul de l'opérateur proximal.

Le plan de ce chapitre est le suivant : nous allons tout d'abord présenter dans la section 3.2 le problème de minimisation étudié. Puis, dans la section 3.3, nous proposerons une version exacte et inexacte de l'algorithme VMFB. Enfin, nous donnerons les résultats de convergence ainsi que leurs démonstrations dans la section 3.4. Dans le chapitre 4, nous présenterons une application de cette méthode pour la résolution d'un problème inverse en traitement d'images.

3.2 Problème de Minimisation

Nous nous intéressons au problème de minimisation suivant :

$$\text{trouver} \quad \widehat{\mathbf{x}} \in \underset{\mathbf{x}\in\mathbb{R}^N}{\operatorname{Argmin}} \ \mathsf{f}(\mathbf{x}), \tag{3.1}$$

où $\mathsf{f} : \mathbb{R}^N \to]-\infty, +\infty]$ est une fonction coercive. De plus, nous supposons que f peut s'écrire de la façon suivante :

$$\mathsf{f} = \mathsf{h} + \mathsf{g}, \tag{3.2}$$

où h est une fonction différentiable et g est une fonction propre et s.c.i. Un algorithme permettant de résoudre ce problème est l'algorithme explicite-implicite donné par l'algorithme 2.4. Ses propriétés de convergence ont été données dans le chapitre 2.

Dans la suite de ce chapitre, nous nous intéresserons au problème de minimisation de la forme (3.1)-(3.2) où les fonctions h et g satisfont les hypothèses suivantes :

Hypothèse 3.1.

(i) $\mathsf{g}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *est une fonction convexe, propre, semi-continue inférieurement et dont la restriction à son domaine* $\operatorname{dom}\mathsf{g}$ *est continue.*

(ii) $\mathsf{h}\colon \mathbb{R}^N \to \mathbb{R}$ *est une fonction différentiable sur* $\mathbb{R}^N$ *de gradient* β*-Lipschitz sur* $\operatorname{dom}\mathsf{g}$*, où* $\beta > 0$.

(iii) f*, définie par* (3.2)*, est coercive.*

Les remarques suivantes concernant l'hypothèse 3.1 seront utiles par la suite.

Remarque 3.1.

(i) *L'hypothèse 3.1(ii) sur* h *est moins contraignante que l'hypothèse de gradient Lipschitz adoptée en général pour démontrer la convergence de l'algorithme explicite-implicite* [Attouch *et al.*, 2011; Combettes et Wajs, 2005]*. En particulier, si le domaine de* g *est compact et* h *est deux fois continument différentiable, alors l'hypothèse 3.1(ii) est satisfaite.*

(ii) *D'après l'hypothèse 3.1(ii),* $\operatorname{dom}\mathsf{g} \subset \operatorname{dom}\mathsf{h}$*. Par conséquent, en utilisant l'hypothèse 3.1(i),* $\operatorname{dom}\mathsf{f} = \operatorname{dom}\mathsf{g}$ *est non vide et convexe.*

(iii) *D'après l'hypothèse 3.1,* f *est une fonction propre et semi-continue inférieurement, dont la restriction à son domaine est continue. Ainsi, par coercivité de* f*, pour tout* $\mathbf{x} \in \operatorname{dom}\mathsf{g}$*,* $\operatorname{lev}_{\leqslant \mathsf{f}(\mathbf{x})}\mathsf{f}$ *est un ensemble compact. De plus, l'ensemble des minimiseurs de* f *est non-vide et compact.*

3.3 Méthode proposée

3.3.1 Algorithme explicite-implicite à métrique variable

L'algorithme explicite-implicite à métrique variable a été introduit dans [Becker et Fadili, 2012; Bonnans *et al.*, 1995; Burke et Qian, 1999; Chen et Rockafellar, 1997; Combettes et Vũ, 2014; Lotito *et al.*, 2009; Ochs *et al.*, 2014; Parente *et al.*, 2008] dans le but d'accélérer la convergence de l'algorithme (2.4).

Algorithme 3.1 Algorithme explicite-implicite à métrique variable

Initialisation : Soient $\mathbf{x}_0 \in \operatorname{dom} \mathsf{g}$ et $(\forall k \in \mathbb{N})\ (\gamma_k, \lambda_k) \in]0, +\infty[^2$ et $\mathbf{A}_k(\mathbf{x}_k) \in \mathcal{S}_N^+$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} \mathbf{y}_k = \operatorname{prox}_{\gamma_k^{-1}\boldsymbol{A}_k(\mathbf{x}_k),\mathsf{g}}(\mathbf{x}_k - \gamma_k \boldsymbol{A}_k(\mathbf{x}_k)^{-1}\nabla \mathsf{h}(\mathbf{x}_k)), \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k(\mathbf{y}_k - \mathbf{x}_k). \end{array} \right.$$

Remarque 3.2.

D'après l'hypothèse 3.1(i), $\operatorname{dom} \mathsf{g}$ *est convexe. Donc, pour tout* $k \in \mathbb{N}$*, si* $\lambda_k \in [0, 1]$*,* $\mathbf{x}_k$ *et* $\mathbf{y}_k$ *sont dans* $\operatorname{dom} \mathsf{g}$*.*

Certains liens avec des méthodes existantes peuvent être établies concernant l'algorithme 3.1. D'une part, lorsque $\mathbf{A}_k(\mathbf{x}_k) \equiv \mathbf{I}_N$, l'algorithme 3.1 est équivalent à l'algorithme explicite-implicite usuel donné par l'algorithme 2.4. D'autre part, lorsque $\mathsf{g} \equiv 0$, l'algorithme 3.1 correspond à un algorithme de gradient préconditionné.

La convergence de l'algorithme 3.1 a été étudiée dans divers travaux. Le résultat le plus général est donné dans [Combettes et Vũ, 2014] : les auteurs ont démontré la convergence de la suite des itérés $(\mathbf{x}_k)_{k\in\mathbb{N}}$ vers une solution du problème (3.1)-(3.2) lorsque h et g sont convexes et qu'il existe une suite $(\eta_k)_{k\in\mathbb{N}}$ positive bornée telle que, pour tout $k \in \mathbb{N}$,

$$(1 + \eta_k)\mathbf{A}_{k+1}(\mathbf{x}_{k+1}) \succcurlyeq \mathbf{A}_k(\mathbf{x}_k). \tag{3.3}$$

D'autres résultats de convergences existent dans la littérature dans le cas particulier où g est une fonction indicatrice sur un ensemble convexe [Bertsekas, 1982; Birgin *et al.*, 2000; Bonettini *et al.*, 2009] ou lorsque h est localement fortement convexe avec un hessien Lipschitz continu et g est convexe [Lee *et al.*, 2012]. Récemment la convergence de l'algorithme 3.1 a aussi été établie dans [Tran-Dinh *et al.*, 2014] lorsque g est convexe, h est une fonction lisse auto-concordante, et, pour tout $k \in \mathbb{N}$, $\mathbf{A}_k(\mathbf{x}_k)$ est soit le hessien de h en $\mathbf{x}_k$ (ou une version approchée de celui-ci), soit une matrice diagonale. Cependant, dans aucun des travaux précédents, l'étude de la convergence de l'algorithme 3.1 n'est menée dans le cas où la fonction h est non-convexe. C'est ce que nous nous proposons d'étudier.

3.3.2 Choix de la métrique

L'algorithme 3.1 que nous étudions dans ce chapitre est accéléré grâce à l'introduction d'une métrique variable induite par les matrices $(\mathbf{A}_k(\mathbf{x}_k))_{k\in\mathbb{N}}$. L'une des contributions de ce chapitre est la proposition d'une méthode constructive pour choisir les matrices permettant de définir cette métrique. Pour cela, nous allons nous baser sur une stratégie de type MM. Plus précisément, considérons une suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ de $\operatorname{dom}\mathsf{g}$. Nous définissons alors la suite de matrices $(\mathbf{A}_k(\mathbf{x}_k))_{k\in\mathbb{N}}$ dans $\mathcal{S}_N^+$ satisfaisant l'hypothèse suivante :

Hypothèse 3.2.

(i) *Pour tout $k \in \mathbb{N}$, la fonction quadratique définie par*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}(\mathbf{x}, \mathbf{x}_k) := \mathsf{h}(\mathbf{x}_k) + \langle \mathbf{x} - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k)\rangle + \frac{1}{2}\|\mathbf{x} - \mathbf{x}_k\|^2_{\mathbf{A}_k(\mathbf{x}_k)},$$

est une majorante de h en $\mathbf{x}_k$ sur $\operatorname{dom}\mathsf{g}$.

(ii) *Il existe $(\underline{\nu}, \overline{\nu}) \in]0, +\infty[^2$ tel que*

$$(\forall k \in \mathbb{N}) \quad \underline{\nu}\,\mathbf{I}_N \preccurlyeq \mathbf{A}_k(\mathbf{x}_k) \preccurlyeq \overline{\nu}\,\mathbf{I}_N\,.$$

Cette hypothèse permet de faire le lien entre l'algorithme 3.1 et le principe MM. En effet, remarquons que, d'après l'algorithme 3.1 et l'hypothèse 3.1(i), lorsque $(\forall k \in \mathbb{N})$ $\gamma_k = 1$, on a

$$\begin{aligned}
\mathbf{y}_k &= \operatorname{prox}_{\mathbf{A}_k(\mathbf{x}_k),\mathsf{g}}(\mathbf{x}_k - \mathbf{A}_k(\mathbf{x}_k)^{-1}\nabla \mathsf{h}(\mathbf{x}_k)) \\
&= \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N} \; \mathsf{g}(\mathbf{x}) + \frac{1}{2}\|\mathbf{x} - \mathbf{x}_k + \mathbf{A}_k(\mathbf{x}_k)^{-1}\nabla \mathsf{h}(\mathbf{x}_k)\|^2_{\mathbf{A}_k(\mathbf{x}_k)} \\
&= \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N} \; \mathsf{g}(\mathbf{x}) + \mathsf{q}(\mathbf{x}, \mathbf{x}_k),
\end{aligned}$$

où $\mathsf{q}(\cdot, \mathbf{x}_k)$ est la majorante quadratique de h définie par l'hypothèse 3.2(i).

Remarque 3.3. *D'après le lemme 2.1 du chapitre 2, l'existence de telles matrices $(\mathbf{A}_k(\mathbf{x}_k))_{k\in\mathbb{N}}$ est assurée. En effet, d'après l'hypothèse 3.1, $\operatorname{dom}\mathsf{g}$ est convexe et h est de gradient β-Lipschitz sur $\operatorname{dom}\mathsf{g}$. Par conséquent, si, pour tout $k \in \mathbb{N}$, $\mathbf{A}_k(\mathbf{x}_k) = \beta\,\mathbf{I}_N$, l'hypothèse 3.2 est satisfaite avec $\underline{\nu} = \overline{\nu} = \beta$.*

Bien que la remarque précédente permette de choisir facilement les matrices $(\mathbf{A}_k(\mathbf{x}_k))_{k\in\mathbb{N}}$, comme il l'a été précisé dans [Becker et Fadili, 2012; Lee *et al.*, 2012; Tran-Dinh *et al.*, 2014] (voir aussi le chapitre 2), l'obtention de bonnes performances de l'algorithme 3.1 dépend fortement de l'exactitude de l'approximation de second ordre caractérisant la métrique. Plus précisément, il faut construire $(\mathbf{A}_k(\mathbf{x}_k))_{k\in\mathbb{N}}$ de telle façon que, pour tout $k \in \mathbb{N}$, la fonction quadratique $\mathsf{q}(\cdot, \mathbf{x}_k)$ soit aussi proche que possible de h au voisinage de $\mathbf{x}_k$, tout en satisfaisant l'hypothèse 3.2. Comme nous l'avons remarqué dans la section 2.4.3 du chapitre 2, il existe des techniques de construction de telles métriques dans la littérature pour certaines classes particulières de fonctions h [Erdogan et Fessler, 1999; Geman et Reynold, 1992; Geman et Yang, 1995; Hunter et Lange, 2004].

3.3.3 Algorithme inexact à métrique variable

De manière générale, l'opérateur proximal relatif à une métrique quelconque n'a pas d'expression simple explicite. Afin de pallier cette difficulté, nous proposons de résoudre le problème (3.1) grâce à une version inexacte de l'algorithme explicite-implicite à métrique variable, s'inspirant de l'algorithme inexact explicite-implicite donné par l'algorithme 2.6, proposé dans [Attouch *et al.*, 2011, Sec. 5] :

Algorithme 3.2 Algorithme inexact explicite-implicite à métrique variable

Initialisation : Soient $\mathbf{x}_0 \in \operatorname{dom} \mathsf{g}$, $\alpha \in]0, +\infty[$ et $(\forall k \in \mathbb{N})$ $(\gamma_k, \lambda_k) \in]0, +\infty[^2$ et $\mathbf{A}_k(\mathbf{x}_k) \in \mathcal{S}_N^+$.

Itérations :

Pour $k = 0, 1, \ldots$

trouver $\mathbf{y}_k \in \mathbb{R}^N$ et $\mathbf{r}_k \in \partial \mathsf{g}(\mathbf{y}_k)$ tels que

$$\mathsf{g}(\mathbf{y}_k) + \langle \mathbf{y}_k - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k) \rangle + \gamma_k^{-1} \|\mathbf{y}_k - \mathbf{x}_k\|^2_{\mathbf{A}_k(\mathbf{x}_k)} \leqslant \mathsf{g}(\mathbf{x}_k),$$

$$\|\nabla \mathsf{h}(\mathbf{x}_k) + \mathbf{r}_k\| \leqslant \alpha \|\mathbf{y}_k - \mathbf{x}_k\|_{\mathbf{A}_k(\mathbf{x}_k)},$$

$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k(\mathbf{y}_k - \mathbf{x}_k)$.

De même que pour l'algorithme 2.6, dans l'algorithme 3.2, la première inégalité correspond à une condition de décroissance suffisante, et la seconde peut être interprétée comme une condition inexacte d'optimalité.

Nous supposerons que le pas $(\gamma_k)_{k\in\mathbb{N}}$ et le paramètre de relaxation $(\lambda_k)_{k\in\mathbb{N}}$ présents dans les algorithmes 3.1 et 3.2 satisfont les deux hypothèses suivante :

Hypothèse 3.3.

(i) *Il existe* $\underline{\lambda} \in]0, +\infty[$ *telle que, pour tout* $k \in \mathbb{N}$, $\underline{\lambda} \leqslant \lambda_k \leqslant 1$.

(ii) *Il existe* $(\underline{\eta}, \overline{\eta}) \in]0, +\infty[^2$ *tel que, pour tout* $k \in \mathbb{N}$, $\underline{\eta} \leqslant \lambda_k \gamma_k \leqslant 2 - \overline{\eta}$.

Hypothèse 3.4.

Il existe $\underline{\tau} \in]0, 1]$ *telle que, pour tout* $k \in \mathbb{N}$, *le paramètre de relaxation* λ_k *satisfait*

$$\mathsf{f}((1 - \lambda_k)\mathbf{x}_k + \lambda_k \mathbf{y}_k) \leqslant (1 - \underline{\tau})\mathsf{f}(\mathbf{x}_k) + \underline{\tau}\mathsf{f}(\mathbf{y}_k).$$

Remarque 3.4.

(i) *En posant* $\underline{\tau} = 1$, *on retrouve la condition de décroissance usuelle :*

$$(\forall k \in \mathbb{N}) \quad \mathsf{f}((1 - \lambda_k)\mathbf{x}_k + \lambda_k \mathbf{y}_k) \leqslant \mathsf{f}(\mathbf{y}_k). \tag{3.4}$$

(ii) *Dans la section suivante, nous montrerons que l'hypothèse 3.4 peut être simplifiée dans le cas où* f *est une fonction convexe. Cependant, notons que même dans le cas général, il existe au moins une suite* $(\lambda_k)_{k\in\mathbb{N}}$ *satisfaisant l'hypothèse 3.4 (il suffit de prendre, pour tout* $k \in \mathbb{N}$, $\lambda_k = \underline{\tau} = 1$*).*

Remarque 3.5.
Comme nous l'avons dit précédemment, sous condition que les hypothèses 3.1, 3.2, 3.3 et 3.4 soient satisfaites, l'algorithme 3.2 peut être interprété comme étant une version inexacte de l'algorithme 3.1. Soient $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *et* $(\mathbf{y}_k)_{k\in\mathbb{N}}$ *des suites générées par l'algorithme 3.1. Soit* $k \in \mathbb{N}$*, en utilisant à la fois la caractérisation variationnelle de l'opérateur proximal et la convexité de* g*, il existe* $\mathbf{r}_k \in \partial\mathsf{g}(\mathbf{y}_k)$ *tel que*

$$\begin{cases} \mathbf{r}_k = -\nabla\mathsf{h}(\mathbf{x}_k) + \gamma_k^{-1}\mathbf{A}_k(\mathbf{x}_k)(\mathbf{x}_k - \mathbf{y}_k), \\ \langle \mathbf{y}_k - \mathbf{x}_k, \mathbf{r}_k \rangle \geqslant \mathsf{g}(\mathbf{y}_k) - \mathsf{g}(\mathbf{x}_k), \end{cases} \tag{3.5}$$

ce qui conduit à

$$\mathsf{g}(\mathbf{y}_k) + \langle \mathbf{y}_k - \mathbf{x}_k, \nabla\mathsf{h}(\mathbf{x}_k)\rangle + \gamma_k^{-1}\|\mathbf{y}_k - \mathbf{x}_k\|^2_{\mathbf{A}_k(\mathbf{x}_k)} \leqslant \mathsf{g}(\mathbf{x}_k). \tag{3.6}$$

On obtient donc la condition de décroissance de l'algorithme 3.2.

D'autre part, d'après les hypothèses 3.2(ii) et 3.3, on a

$$\|\nabla\mathsf{h}(\mathbf{x}_k) + \mathbf{r}_k\| = \gamma_k^{-1}\|\mathbf{A}_k(\mathbf{x}_k)(\mathbf{x}_k - \mathbf{y}_k)\| \leqslant \underline{\gamma}^{-1}\overline{\nu}^{1/2}\|\mathbf{x}_k - \mathbf{y}_k\|_{\mathbf{A}_k(\mathbf{x}_k)}. \tag{3.7}$$

La condition d'optimalité de l'algorithme 3.2 est alors obtenue en posant $\alpha = \underline{\gamma}^{-1}\overline{\nu}^{1/2}$.

3.4 Analyse de convergence

3.4.1 Propriétés de décroissance

Dans cette section nous allons donner quelques résultats techniques sur le comportement des suites $\big(\mathsf{f}(\mathbf{x}_k)\big)_{k\in\mathbb{N}}$ et $\big(\mathsf{f}(\mathbf{y}_k)\big)_{k\in\mathbb{N}}$ générées par l'algorithme 3.2. Ces propriétés nous permettront ensuite d'étudier la convergence de cet algorithme.

Lemme 3.1. *Supposons que les hypothèses 3.1, 3.2 et 3.3 soient satisfaites. Il existe* $\mu_1 \in]0, +\infty[$ *tel que, pour tout* $k \in \mathbb{N}$*,*

$$\mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{f}(\mathbf{x}_k) - \frac{\mu_1}{2}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 \tag{3.8}$$

$$\leqslant \mathsf{f}(\mathbf{x}_k) - \underline{\lambda}^2\frac{\mu_1}{2}\|\mathbf{y}_k - \mathbf{x}_k\|^2, \tag{3.9}$$

où les suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *et* $(\mathbf{y}_k)_{k\in\mathbb{N}}$ *sont générées par l'algorithme 3.2.*

Démonstration. Pour tout $k \in \mathbb{N}$, la relaxation

$$\mathbf{x}_{k+1} = (1 - \lambda_k)\mathbf{x}_k + \lambda_k\mathbf{y}_k, \tag{3.10}$$

l'hypothèse 3.3(i) et la convexité de g nous donnent

$$\mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{h}(\mathbf{x}_{k+1}) + (1-\lambda_k)\mathsf{g}(\mathbf{x}_k) + \lambda_k \mathsf{g}(\mathbf{y}_k).$$

De plus, en utilisant l'hypothèse 3.2(i), on obtient

$$\mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{h}(\mathbf{x}_k) + \langle \mathbf{x}_{k+1} - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k)\rangle + \frac{1}{2}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2_{\boldsymbol{A}_k(\mathbf{x}_k)} + (1-\lambda_k)\mathsf{g}(\mathbf{x}_k) + \lambda_k \mathsf{g}(\mathbf{y}_k). \quad (3.11)$$

En reformulant (3.10), nous obtenons

$$\mathbf{x}_{k+1} - \mathbf{x}_k = \lambda_k(\mathbf{y}_k - \mathbf{x}_k). \quad (3.12)$$

En combinant la première inégalité de l'algorithme 3.2 et (3.12), on obtient

$$\langle \mathbf{x}_{k+1} - \mathbf{x}_k, \nabla \mathsf{h}(\mathbf{x}_k)\rangle \leqslant -\gamma_k^{-1}\lambda_k^{-1}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2_{\boldsymbol{A}_k(\mathbf{x}_k)} + \lambda_k\big(\mathsf{g}(\mathbf{x}_k) - \mathsf{g}(\mathbf{y}_k)\big). \quad (3.13)$$

Les équations (3.11) et (3.13) nous permettent de déduire que

$$\begin{aligned}\mathsf{f}(\mathbf{x}_{k+1}) &\leqslant \mathsf{f}(\mathbf{x}_k) - (\gamma_k^{-1}\lambda_k^{-1} - \frac{1}{2})\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2_{\boldsymbol{A}_k(\mathbf{x}_k)} \\ &\leqslant \mathsf{f}(\mathbf{x}_k) - \frac{1}{2}\frac{\overline{\eta}}{2-\overline{\eta}}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2_{\boldsymbol{A}_k(\mathbf{x}_k)},\end{aligned}$$

où la dernière inégalité est obtenue en utilisant l'hypothèse 3.3(ii). Par conséquent, la borne inférieure dans l'hypothèse 3.2(ii) nous permet de déduire l'équation (3.8) en posant $\mu_1 = \dfrac{\underline{\nu}\overline{\eta}}{2-\overline{\eta}} > 0$. L'inégalité (3.9) résulte ensuite de (3.12) et de l'hypothèse 3.3(i). □

Le corollaire suivant, permettant de réécrire l'hypothèse 3.4 sous une forme différente, est une conséquence du lemme 3.1.

Corollaire 3.1. *Supposons que les hypothèses 3.1, 3.2 et 3.3 soient satisfaites. Alors l'hypothèse 3.4 est satisfaite si et seulement si il existe* $\underline{\tau} \in]0,1]$ *tel que, pour tout* $k \in \mathbb{N}$,

$$(\exists \tau_k \in [\underline{\tau}, 1]) \quad \mathsf{f}(\mathbf{x}_{k+1}) \leqslant (1-\tau_k)\mathsf{f}(\mathbf{x}_k) + \tau_k \mathsf{f}(\mathbf{y}_k). \quad (3.14)$$

Démonstration.

(i) Supposons que l'hypothèse 3.4 soit satisfaite. En utilisant l'étape de relaxation donnée par (3.10), l'inégalité (3.14) est satisfaite en posant $\tau_k = \underline{\tau}$, pour tout $k \in \mathbb{N}$.

(ii) Montrons maintenant la réciproque du point précédent. Soit $k \in \mathbb{N}$, l'inégalité (3.14) est équivalente à

$$\tau_k(\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\mathbf{y}_k)) \leqslant \mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\mathbf{x}_{k+1}). \quad (3.15)$$

Si l'inégalité ci-dessus est satisfaite pour $\tau_k \in [\underline{\tau}, 1]$, alors on peut distinguer deux situations :

- *Cas où* $\mathsf{f}(\mathbf{x}_k) \leqslant \mathsf{f}(\mathbf{y}_k)$. D'après le lemme 3.1, on a $\mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{f}(\mathbf{x}_k)$. D'où

$$\underline{\tau}(\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\mathbf{y}_k)) \leqslant 0 \leqslant \mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\mathbf{x}_{k+1}).$$

- *Cas où* $\mathsf{f}(\mathbf{x}_k) \geqslant \mathsf{f}(\mathbf{y}_k)$. L'équation, (3.15) nous donne alors

$$\underline{\tau}(\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\mathbf{y}_k)) \leqslant \tau_k(\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\mathbf{y}_k)) \leqslant \mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\mathbf{x}_{k+1}).$$

Ce qui permet de montrer que, si (3.14) est satisfaite, alors, l'hypothèse 3.4 l'est aussi.

□

Le lemme qui suit permet de retrouver des hypothèses standards sur le paramètre de relaxation $(\lambda_k)_{k\in\mathbb{N}}$, sous certaines conditions :

Corollaire 3.2. *Supposons que les hypothèses 3.1, 3.2 et 3.3 soient satisfaites. Si* f *est convexe sur* $[\mathbf{x}_k, \mathbf{y}_k]$*, pour tout* $k \in \mathbb{N}$*, alors l'hypothèse 3.4 est satisfaite.*

Démonstration. D'après l'hypothèse 3.3, pout tout $k \in \mathbb{N}$, $\lambda_k \in]0, 1]$. Si f est convexe sur $[\mathbf{x}_k, \mathbf{y}_k]$ pour tout $k \in \mathbb{N}$, alors

$$\mathsf{f}\big((1-\lambda_k)\mathbf{x}_k + \lambda_k\mathbf{y}_k\big) \leqslant (1-\lambda_k)\mathsf{f}(\mathbf{x}_k) + \lambda_k\mathsf{f}(\mathbf{y}_k).$$

En utilisant le Corollaire 3.1 et le fait que, pour tout $k \in \mathbb{N}$, λ_k est bornée inférieurement par $\underline{\lambda} > 0$, on peut conclure que l'hypothèse 3.4 est satisfaite. □

Le résultat suivant va nous permettre d'évaluer les variations de f entre les points $\mathbf{x}_k$ et $\mathbf{y}_k$, à chaque itération $k \in \mathbb{N}$ de l'algorithme 3.2.

Lemme 3.2. *Supposons que les hypothèses 3.1, 3.2 et 3.3 soient satisfaites. Il existe* $\mu_2 \in \mathbb{R}$ *tel que*

$$(\forall k \in \mathbb{N}) \quad \mathsf{f}(\mathbf{y}_k) \leqslant \mathsf{f}(\mathbf{x}_k) - \mu_2\|\mathbf{y}_k - \mathbf{x}_k\|^2.$$

Démonstration. D'après l'hypothèse 3.2(i) et la première inégalité de l'algorithme 3.2, on a

$$\mathsf{h}(\mathbf{y}_k) \leqslant \mathsf{q}(\mathbf{y}_k, \mathbf{x}_k) \leqslant \mathsf{h}(\mathbf{x}_k) + \mathsf{g}(\mathbf{x}_k) - \mathsf{g}(\mathbf{y}_k) - (\gamma_k^{-1} - \frac{1}{2})\|\mathbf{y}_k - \mathbf{x}_k\|^2_{\boldsymbol{A}_k(\mathbf{x}_k)},$$

ce qui, en utilisant l'hypothèse 3.3, nous conduit à

$$\mathsf{f}(\mathbf{y}_k) \leqslant \mathsf{f}(\mathbf{x}_k) - (\frac{\underline{\lambda}}{2-\overline{\eta}} - \frac{1}{2})\|\mathbf{y}_k - \mathbf{x}_k\|^2_{\boldsymbol{A}_k(\mathbf{x}_k)}.$$

Le résultat découle ensuite de l'hypothèse 3.2(ii) en posant

$$\mu_2 = \begin{cases} \underline{\nu}\left(\dfrac{\underline{\lambda}}{2-\overline{\eta}} - \dfrac{1}{2}\right), & \text{si } 2\underline{\lambda} + \overline{\eta} \geqslant 2, \\ \overline{\nu}\left(\dfrac{\underline{\lambda}}{2-\overline{\eta}} - \dfrac{1}{2}\right), & \text{sinon.} \end{cases}$$

□

3.4.2 Résultat de convergence

Notre démonstration de convergence repose sur le résultat préliminaire suivant :

Lemme 3.3. *Soient* $(\mathsf{u}_k)_{k\in\mathbb{N}}$, $(\mathsf{v}_k)_{k\in\mathbb{N}}$, $(\mathsf{v}'_k)_{k\in\mathbb{N}}$ *et* $(\Delta_k)_{k\in\mathbb{N}}$ *des suites de réels positifs. Soit* $\theta \in]0,1[$. *Supposons que :*

(i) *Pour tout* $k \in \mathbb{N}$, $\mathsf{u}_k^2 \leqslant \mathsf{v}_k^\theta \Delta_k$.

(ii) $(\Delta_k)_{k\in\mathbb{N}}$ *est une suite sommable.*

(iii) *Pour tout* $k \in \mathbb{N}$, $\mathsf{v}_{k+1} \leqslant (1-\underline{\alpha})\mathsf{v}_k + \mathsf{v}'_k$, *avec* $\underline{\alpha} \in]0,1]$.

(iv) *Pour tout* $k \geqslant k^*$, $(\mathsf{v}'_k)^\theta \leqslant \varepsilon \mathsf{u}_k$, *où* $\varepsilon > 0$ *et* $k^* \in \mathbb{N}$.

La suite $(\mathsf{u}_k)_{k\in\mathbb{N}}$ *est sommable.*

Démonstration. En utilisant la sous-additivité de la fonction $\mathsf{t} \mapsto \mathsf{t}^\theta$, lorsque $\theta \in]0,1[$, d'après l'assertion (iii), pour tout $k \in \mathbb{N}$, on a

$$\mathsf{v}_{k+1}^\theta \leqslant (1-\underline{\alpha})^\theta \mathsf{v}_k^\theta + (\mathsf{v}'_k)^\theta.$$

De plus, en utilisant (iv), on obtient

$$(\forall k \geqslant k^*) \qquad \mathsf{v}_{k+1}^\theta \leqslant (1-\underline{\alpha})^\theta \mathsf{v}_k^\theta + \varepsilon \mathsf{u}_k,$$

ce qui implique que, pour tout $K > k^*$,

$$\sum_{k=k^*+1}^{K} \mathsf{v}_k^\theta \leqslant (1-\underline{\alpha})^\theta \sum_{k=k^*}^{K-1} \mathsf{v}_k^\theta + \varepsilon \sum_{k=k^*}^{K-1} \mathsf{u}_k, \tag{3.16}$$

et de manière équivalente, on obtient alors

$$\big(1-(1-\underline{\alpha})^\theta\big) \sum_{k=k^*}^{K-1} \mathsf{v}_k^\theta \leqslant \mathsf{v}_{k^*}^\theta - \mathsf{v}_K^\theta + \varepsilon \sum_{k=k^*}^{K-1} \mathsf{u}_k. \tag{3.17}$$

D'autre part, l'assertion (i) peut être reformulée de la façon suivante :

$$(\forall k \in \mathbb{N}) \qquad \mathsf{u}_k^2 \leqslant \Big(\varepsilon^{-1}\big(1-(1-\underline{\alpha})^\theta\big)\mathsf{v}_k^\theta\Big)\Big(\varepsilon\big(1-(1-\underline{\alpha})^\theta\big)^{-1}\Delta_k\Big).$$

En utilisant l'inégalité $(\forall (\mathsf{t},\mathsf{t}') \in [0,+\infty[^2)\ \sqrt{\mathsf{t}\mathsf{t}'} \leqslant (\mathsf{t}+\mathsf{t}')/2$ et puisque, pour tout $k \in \mathbb{N}$, $\mathsf{u}_k \geqslant 0$, on obtient

$$(\forall k \in \mathbb{N}) \qquad \mathsf{u}_k \leqslant \frac{1}{2}\Big(\varepsilon^{-1}\big(1-(1-\underline{\alpha})^\theta\big)\mathsf{v}_k^\theta + \varepsilon\big(1-(1-\underline{\alpha})^\theta\big)^{-1}\Delta_k\Big). \tag{3.18}$$

On peut alors déduire des équations (3.17) et (3.18) que, pour tout $K > k^*$,

$$\sum_{k=k^*}^{K-1} \mathsf{u}_k \leqslant \frac{1}{2}\Big(\sum_{k=k^*}^{K-1} \mathsf{u}_k + \varepsilon^{-1}(\mathsf{v}_{k^*}^\theta - \mathsf{v}_K^\theta) + \varepsilon\big(1-(1-\underline{\alpha})^\theta\big)^{-1} \sum_{k=k^*}^{K-1} \Delta_k\Big),$$

et par conséquent,

$$\sum_{k=k^*}^{K-1} \mathsf{u}_k \leqslant \varepsilon^{-1}\mathsf{v}_{k^*}^\theta + \varepsilon\big(1-(1-\underline{\alpha})^\theta\big)^{-1} \sum_{k=k^*}^{K-1} \Delta_k.$$

Il en découle la sommabilité de la suite $(\mathsf{u}_k)_{k\in\mathbb{N}}$ en utilisant (ii). □

Nous pouvons maintenant énoncer le résultat principal de ce chapitre concernant la convergence des suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ et $(\mathbf{y}_k)_{k\in\mathbb{N}}$ générées par l'algorithme 3.1 ou sa version inexacte, i.e. l'algorithme 3.2. Pour cela, en plus des hypothèses utilisées jusqu'ici dans ce chapitre, nous allons supposer que la fonction f satisfait l'inégalité de Kurdyka-Łojasiewicz (c.f. Chapitre 2).

Théorème 3.1. *Supposons que les hypothèses 3.1, 3.2 et 3.3 soient satisfaites. Supposons de plus que la fonction* f *satisfasse l'inégalité de Kurdyka-Łojasiewicz donnée dans la définition 2.20. Les assertions suivantes sont alors satisfaites :*

(i) *Les suites* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *et* $(\mathbf{y}_k)_{k\in\mathbb{N}}$ *générées par l'algorithme 3.1 ou l'algorithme 3.2 conver- gent toutes les deux vers un point critique* $\widehat{\mathbf{x}}$ *de* f.

(ii) *Ces deux suites satisfont*

$$\sum_{k=0}^{+\infty} \|\mathbf{x}_{k+1}-\mathbf{x}_k\| < +\infty \quad \text{et} \quad \sum_{k=0}^{+\infty} \|\mathbf{y}_{k+1}-\mathbf{y}_k\| < +\infty.$$

(iii) *Les suites* $\big(\mathsf{f}(\mathbf{x}_k)\big)_{k\in\mathbb{N}}$ *et* $\big(\mathsf{f}(\mathbf{y}_k)\big)_{k\in\mathbb{N}}$ *convergent vers* $\mathsf{f}(\widehat{\mathbf{x}})$. *De plus,* $\big(\mathsf{f}(\mathbf{x}_k)\big)_{k\in\mathbb{N}}$ *est décroissante.*

Remarque 3.6. *La démonstration de ce théorème est donnée dans l'annexe 3.A. Elle est constituée de trois étapes principales :*

(i) *La première étape consiste à se servir des propriétés de décroissances présentées dans la section 3.4.1 pour montrer que les suites* $\big(\mathsf{f}(\mathbf{x}_k)\big)_{k\in\mathbb{N}}$ *et* $\big(\mathsf{f}(\mathbf{y}_k)\big)_{k\in\mathbb{N}}$ *convergent vers un réel* ξ, *et que* $\big(\mathsf{f}(\mathbf{x}_k)\big)_{k\in\mathbb{N}}$ *est décroissante.*

(ii) *En utilisant l'inégalité de KL et le lemme 3.3, on montre dans la seconde étape que les suites* $\big(\|\mathbf{x}_{k+1}-\mathbf{x}_k\|\big)_{k\in\mathbb{N}}$ $\big(\|\mathbf{y}_{k+1}-\mathbf{y}_k\|\big)_{k\in\mathbb{N}}$ *sont sommables, et donc que les suites* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *et* $(\mathbf{y}_k)_{k\in\mathbb{N}}$ *sont de Cauchy et convergent vers un point* $\widehat{\mathbf{x}} \in \mathbb{R}^N$.

(iii) *La dernière étape consiste à montrer que* $\widehat{\mathbf{x}}$ *est un point critique de* f, *en utilisant la proposition 2.4.*

Le théorème de convergence précédent assure que les suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ et $(\mathbf{y}_k)_{k\in\mathbb{N}}$ convergent vers un point critique de f. De plus, la décroissance de la suite $(\mathsf{f}(\mathbf{x}_k))_{k\in\mathbb{N}}$ implique que ce point critique ne sera pas un maximum de f. Cependant, en théorie, les algorithmes 3.1 et 3.2 peuvent converger vers un minimum local de f, ou bien un point selle. Le corollaire qui suit nous permet d'affirmer que, si les algorithmes 3.1 et 3.2 sont initialisés avec un point $\mathbf{x}_0$ suffisamment proche d'un minimum global de f, alors les suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ et $(\mathbf{y}_k)_{k\in\mathbb{N}}$ convergeront vers cette solution.

Corollaire 3.3. *Supposons que les hypothèses 3.1, 3.2 et 3.3 soient satisfaites. Supposons de plus que la fonction* f *satisfait l'inégalité de Kurdyka-Łojasiewicz. Soient* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *et* $(\mathbf{y}_k)_{k\in\mathbb{N}}$ *des suites générées par l'algorithme 3.2. Il existe* $\upsilon > 0$ *tel que, si*

$$\mathsf{f}(\mathbf{x}_0) \leqslant \inf_{\mathbf{x}\in\mathbb{R}^N} \mathsf{f}(\mathbf{x}) + \upsilon, \tag{3.19}$$

alors les suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *et* $(\mathbf{y}_k)_{k\in\mathbb{N}}$ *convergent toutes les deux vers une solution du problème* (3.1).

Démonstration. D'après la remarque 3.1(iii), on a

$$\xi = \inf_{\mathbf{x}\in\mathbb{R}^N} \mathsf{f}(\mathbf{x}) < +\infty.$$

Soit $E = \mathrm{lev}_{\leqslant \xi+\delta}\mathsf{f}$, où $\delta > 0$. Cet ensemble est borné d'après l'hypothèse 3.1(iii). Puisque f satisfait l'inégalité de KL, il existe trois constantes $\kappa > 0$, $\zeta > 0$ and $\theta \in [0, 1[$ telles que

$$(\forall \mathbf{r} \in \partial \mathsf{g}(\mathbf{x})) \qquad \kappa|\mathsf{f}(\mathbf{x}) - \xi|^\theta \leqslant \|\nabla \mathsf{h}(\mathbf{x}) + \mathbf{r}\|, \tag{3.20}$$

pour tout $\mathbf{x} \in E$ tel que $|\mathsf{f}(\mathbf{x}) - \xi| \leqslant \zeta$. En d'autres termes, $\mathsf{f}(\mathbf{x}) \leqslant \xi + \zeta$ puisque, par définition de ξ, $\mathsf{f}(x) \geqslant \xi$ est toujours vraie. Nous posons maintenant $\upsilon = \min\{\delta, \zeta\} > 0$ et choisissons $\mathbf{x}_0$ de façon à satisfaire (3.19). Le théorème 3.1(iii) implique que, pour tout $k \in \mathbb{N}$,

$$\mathsf{f}(\mathbf{x}_k) \leqslant \xi + \upsilon.$$

La restriction de f à son domaine de définition étant continue, d'après le théorème 3.1(i), les suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ et $(\mathbf{y}_k)_{k\in\mathbb{N}}$ convergent vers $\widehat{\mathbf{x}}$, où $\widehat{\mathbf{x}}$ est tel que $\mathsf{f}(\widehat{\mathbf{x}}) \leqslant \xi + \upsilon$. L'inégalité de KL est donc satisfaite en $\widehat{\mathbf{x}}$:

$$(\forall \mathbf{t} \in \partial \mathsf{f}(\widehat{\mathbf{x}})) \quad \|\mathbf{t}\| \geqslant \kappa|\mathsf{f}(\widehat{\mathbf{x}}) - \xi|^\theta.$$

De plus, $\widehat{\mathbf{x}}$ étant un point critique de f, on a $\mathbf{0} \in \partial \mathsf{f}(\widehat{\mathbf{x}})$, et

$$|\mathsf{f}(\widehat{\mathbf{x}}) - \xi|^\theta \leqslant 0.$$

Pour conclure, nous obtenons $\mathsf{f}(\widehat{\mathbf{x}}) = \inf_{\mathbf{x}\in\mathbb{R}^N} \mathsf{f}(\mathbf{x})$. □

Remarque 3.7.

(i) *Remarquons que, bien que la constante de Lipschitz β n'apparaisse pas explicitement dans nos algorithmes, l'hypothèse de Lipschitz différentiabilité est indispensable dans les démonstrations de nos théorèmes de convergence.*

(ii) *Nous pouvons remarquer que la convergence de l'algorithme 3.2 a été démontrée dans* [Attouch *et al.*, 2011] *dans le cas non préconditionné, i.e. lorsque $\mathbf{A}_k(\mathbf{x}_k) \equiv \beta \mathbf{I}_N$ dans le théorème 2.9.*

3.5 Conclusion

Dans ce chapitre nous avons proposé de minimiser une fonction $\mathsf{f} = \mathsf{h} + \mathsf{g}$ où h est une fonction différentiable non-nécessairement convexe, g est une fonction convexe non-nécessairement différentiable, et f est une fonction coercive satisfaisant l'inégalité de KL. Pour cela, nous avons donné une version accélérée de l'algorithme explicite-implicite. Cette accélération est basée sur l'introduction d'une métrique variant à chaque itération, choisie grâce au principe de Majoration-Minimisation. Enfin, nous avons démontré la convergence de l'algorithme itératif proposé vers un point critique de la fonction à minimiser. Plus précisément, nous avons démontré :

- un résultat de convergence global, affirmant que les suites générées par l'algorithme explicite-implicite à métrique variable convergent vers un point critique de la fonction f,
- et un résultat de convergence local, montrant qu'en initialisant judicieusement l'algorithme, alors les suites générées convergent vers un minimum global de h.

Afin d'illustrer l'intérêt de l'utilisation de métriques variables, nous traiterons dans le chapitre 4 une application en reconstruction d'images.

3.A Démonstration du théorème 3.1

Dans cette annexe, nous allons établir le théorème 3.1 en montrant successivement les étapes (i)-(iii) décrites dans la remarque 3.6.

(i) D'après le lemme 3.1, on a

$$(\forall k \in \mathbb{N}) \quad \mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{f}(\mathbf{x}_k),$$

donc $(\mathsf{f}(\mathbf{x}_k))_{k\in\mathbb{N}}$ est une suite décroissante. De plus, les remarques 3.1(iii) et 3.2 assurent que les éléments des suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ et $(\mathbf{y}_k)_{k\in\mathbb{N}}$ sont dans un sous-ensemble compact E de $\mathrm{lev}_{\leqslant \mathsf{f}(\mathbf{x}_0)}\, \mathsf{f} \subset \mathrm{dom}\, \mathsf{g}$ et que f est bornée inférieurement. Par conséquent, $\big(\mathsf{f}(\mathbf{x}_k)\big)_{k\in\mathbb{N}}$ converge vers un réel ξ, et $\big(\mathsf{f}(\mathbf{x}_k) - \xi\big)_{k\in\mathbb{N}}$ est une suite positive convergeant vers 0.
En outre, le lemme 3.1 nous donne

$$(\forall k \in \mathbb{N}) \quad \underline{\lambda}^2 \frac{\mu_1}{2} \|\mathbf{y}_k - \mathbf{x}_k\|^2 \leqslant (\mathsf{f}(\mathbf{x}_k) - \xi) - (\mathsf{f}(\mathbf{x}_{k+1}) - \xi). \tag{3.21}$$

Ainsi, la suite $(\mathbf{y}_k - \mathbf{x}_k)_{k\in\mathbb{N}}$ converge vers $\mathbf{0}$.
D'autre part, la relaxation

$$(\forall k \in \mathbb{N}) \quad \mathbf{x}_{k+1} = (1 - \lambda_k)\mathbf{x}_k + \lambda_k \mathbf{y}_k, \tag{3.22}$$

et l'hypothèses 3.4 impliquent que, pour tout $k \in \mathbb{N}$,

$$\mathsf{f}(\mathbf{x}_{k+1}) - \xi \leqslant (1 - \underline{\alpha})(\mathsf{f}(\mathbf{x}_k) - \xi) + \underline{\alpha}(\mathsf{f}(\mathbf{y}_k) - \xi).$$

L'inégalité précédente combinée avec le lemme 3.2, nous donne alors, pour tout $k \in \mathbb{N}$,

$$\underline{\alpha}^{-1}(\mathsf{f}(\mathbf{x}_{k+1}) - \xi - (1 - \underline{\alpha})(\mathsf{f}(\mathbf{x}_k) - \xi)) \leqslant \mathsf{f}(\mathbf{y}_k) - \xi \leqslant \mathsf{f}(\mathbf{x}_k) - \xi - \mu_2 \|\mathbf{y}_k - \mathbf{x}_k\|^2.$$

Ainsi, puisque $(\|\mathbf{y}_k - \mathbf{x}_k\|)_{k\in\mathbb{N}}$ et $(\mathsf{f}(\mathbf{x}_k) - \xi)_{k\in\mathbb{N}}$ convergent vers 0, la suite $(\mathsf{f}(\mathbf{y}_k))_{k\in\mathbb{N}}$ converge vers ξ.

(ii) Revenons à l'équation (3.21). Soit $\psi\colon [0, +\infty[\to [0, +\infty[\colon \mathsf{t} \mapsto \mathsf{t}^{1/(1-\theta)}$, où $\theta \in [0, 1[$, une fonction convexe à laquelle nous appliquons l'inégalité de gradient suivante

$$(\forall (\mathsf{t}, \mathsf{t}') \in [0, +\infty[^2) \quad \psi(\mathsf{t}) - \psi(\mathsf{t}') \leqslant \dot{\psi}(\mathsf{t})(\mathsf{t} - \mathsf{t}').$$

Après un changement de variable, cette inégalité peut être réécrite comme suit

$$(\forall (\mathsf{t}, \mathsf{t}') \in [0, +\infty[^2) \quad \mathsf{t} - \mathsf{t}' \leqslant (1 - \theta)^{-1} \mathsf{t}^{\theta} (\mathsf{t}^{1-\theta} - \mathsf{t}'^{1-\theta}).$$

En prenant $\mathsf{t} = \mathsf{f}(\mathbf{x}_k) - \xi$ et $\mathsf{t}' = \mathsf{f}(\mathbf{x}_{k+1}) - \xi$, on a alors

$$(\forall k \in \mathbb{N}) \quad \big(\mathsf{f}(\mathbf{x}_k) - \xi\big) - \big(\mathsf{f}(\mathbf{x}_{k+1}) - \xi\big) \leqslant (1-\theta)^{-1}\big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{\theta}\Delta'_k,$$

où

$$(\forall k \in \mathbb{N}) \quad \Delta'_k = \big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{1-\theta} - \big(\mathsf{f}(\mathbf{x}_{k+1}) - \xi\big)^{1-\theta}.$$

En combinant l'inégalité ci-dessus avec (3.21) on obtient

$$(\forall k \in \mathbb{N}) \quad \|\mathbf{y}_k - \mathbf{x}_k\|^2 \leqslant 2\underline{\lambda}^{-2}\mu_1^{-1}(1-\theta)^{-1}\big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{\theta}\Delta'_k. \tag{3.23}$$

Par ailleurs, puisque E est borné, f satisfait l'inégalité de KL et comme h est différentiable, il existe trois constantes $\kappa > 0$, $\zeta > 0$ et $\theta \in [0, 1[$ telles que

$$\big(\forall \mathbf{r} \in \partial \mathsf{g}(\mathbf{x})\big) \qquad \kappa|\mathsf{f}(\mathbf{x}) - \xi|^{\theta} \leqslant \|\nabla \mathsf{h}(\mathbf{x}) + \mathbf{r}\|, \tag{3.24}$$

pour tout $\mathbf{x} \in E$ tel que $|\mathsf{f}(\mathbf{x}) - \xi| \leqslant \zeta$. La suite $\big(\mathsf{f}(\mathbf{y}_k)\big)_{k\in\mathbb{N}}$ convergeant vers ξ, il existe $k^* \in \mathbb{N}$, tel que, pour tout $k \geqslant k^*$, $|\mathsf{f}(\mathbf{y}_k) - \xi| < \zeta$. Ainsi, on a, pour tout $\mathbf{r}_k \in \partial \mathsf{g}(\mathbf{y}_k)$,

$$(\forall k \geqslant k^*) \quad \kappa|\mathsf{f}(\mathbf{y}_k) - \xi|^{\theta} \leqslant \|\nabla \mathsf{h}(\mathbf{y}_k) + \mathbf{r}_k\|.$$

Soit $\mathbf{r}_k$ donné dans l'algorithme 3.2. On a alors

$$\begin{aligned}
\kappa|\mathsf{f}(\mathbf{y}_k) - \xi|^{\theta} &\leqslant \|\nabla \mathsf{h}(\mathbf{y}_k) - \nabla \mathsf{h}(\mathbf{x}_k) + \nabla \mathsf{h}(\mathbf{x}_k) + \mathbf{r}_k\| \\
&\leqslant \|\nabla \mathsf{h}(\mathbf{y}_k) - \nabla \mathsf{h}(\mathbf{x}_k)\| + \|\nabla \mathsf{h}(\mathbf{x}_k) + \mathbf{r}_k\| \\
&\leqslant \|\nabla \mathsf{h}(\mathbf{y}_k) - \nabla \mathsf{h}(\mathbf{x}_k)\| + \tau\|\mathbf{x}_k - \mathbf{y}_k\|_{\boldsymbol{A}_k(\mathbf{x}_k)}.
\end{aligned} \tag{3.25}$$

Donc, en utilisant les hypothèses 3.1(ii) et 3.2(ii), on obtient

$$|\mathsf{f}(\mathbf{y}_k) - \xi|^{\theta} \leqslant \kappa^{-1}(\beta + \tau\sqrt{\overline{\nu}})\|\mathbf{x}_k - \mathbf{y}_k\|. \tag{3.26}$$

De plus, d'après l'hypothèse 3.4,

$$\mathsf{f}(\mathbf{x}_{k+1}) - \xi \leqslant (1 - \underline{\alpha})(\mathsf{f}(\mathbf{x}_k) - \xi) + |\mathsf{f}(\mathbf{y}_k) - \xi|. \tag{3.27}$$

Par ailleurs, nous pouvons remarquer que

$$\begin{aligned}
\sum_{k=k^*}^{+\infty} \Delta'_k &= \sum_{k=k^*}^{+\infty} \big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{1-\theta} - \big(\mathsf{f}(\mathbf{x}_{k+1}) - \xi\big)^{1-\theta} \\
&= \big(\mathsf{f}(\mathbf{x}_{k^*}) - \xi\big)^{1-\theta},
\end{aligned}$$

ce qui montre que $(\Delta'_k)_{k\in\mathbb{N}}$ est une suite sommable. Posons

$$(\forall k \in \mathbb{N}) \qquad \begin{cases}
\mathsf{u}_k = \|\mathbf{y}_k - \mathbf{x}_k\|, \\
\mathsf{v}_k = \mathsf{f}(\mathbf{x}_k) - \xi \geqslant 0, \\
\mathsf{v}'_k = |\mathsf{f}(\mathbf{y}_k) - \xi| \geqslant 0, \\
\varepsilon = \kappa^{-1}(\beta + \tau\sqrt{\overline{\nu}}) > 0, \\
\Delta_k = 2\underline{\lambda}^{-2}\mu_1^{-1}(1-\theta)^{-1}\Delta'_k.
\end{cases}$$

En utilisant l'équation (3.23), la sommabilité de $(\Delta'_k)_{k\in\mathbb{N}}$ et les équations (3.26) et (3.27), le lemme 3.3 nous permet d'affirmer que $(\|\mathbf{y}_k - \mathbf{x}_k\|)_{k\in\mathbb{N}}$ est une suite sommable dès que $\theta \neq 0$. Dans le cas où $\theta = 0$, puisque $\mathbf{x}_k - \mathbf{y}_k \to \mathbf{0}$, il existe $k^{**} \geqslant k^*$ tel que

$$(\forall k \geqslant k^{**}) \quad \kappa^{-1}(\beta + \tau\sqrt{\overline{\nu}})\|\mathbf{x}_k - \mathbf{y}_k\| < 1.$$

Ainsi, d'après l'équation (3.26) (rappelons que nous utilisons la convention $0^0 = 0$), on a nécessairement $\mathsf{f}(\mathbf{y}_k) = \xi$, pour tout $k \geqslant k^{**}$. Alors, d'après (3.23), pour tout $k \geqslant k^{**}$, $\mathbf{x}_k = \mathbf{y}_k$. Ceci prouve que $(\|\mathbf{y}_k - \mathbf{x}_k\|)_{k\in\mathbb{N}}$ est sommable.
D'autre part, en utilisant l'équation (3.22) et l'hypothèse 3.3(i), on a

$$(\forall k \in \mathbb{N}) \quad \|\mathbf{x}_{k+1} - \mathbf{x}_k\| = \lambda_k\|\mathbf{y}_k - \mathbf{x}_k\| \leqslant \|\mathbf{y}_k - \mathbf{x}_k\|.$$

Par conséquent, la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ satisfait

$$\sum_{k=0}^{+\infty} \|\mathbf{x}_{k+1} - \mathbf{x}_k\| < +\infty. \tag{3.28}$$

Cela implique que $(\mathbf{x}_k)_{k\in\mathbb{N}}$ est une suite de Cauchy. Elle converge donc vers un point que l'on notera $\widehat{\mathbf{x}}$. Puisque $(\mathbf{x}_k - \mathbf{y}_k)_{k\in\mathbb{N}}$ converge vers $\mathbf{0}$, $(\mathbf{y}_k)_{k\in\mathbb{N}}$ converge de même vers $\widehat{\mathbf{x}}$. De plus, on a

$$\begin{aligned}(\forall k \in \mathbb{N}) \quad \|\mathbf{y}_{k+1} - \mathbf{y}_k\| &\leqslant \|\mathbf{y}_{k+1} - \mathbf{x}_k\| + \|\mathbf{x}_k - \mathbf{y}_k\| \\ &\leqslant \|\mathbf{y}_{k+1} - \mathbf{x}_{k+1}\| + \|\mathbf{x}_{k+1} - \mathbf{x}_k\| + \|\mathbf{x}_k - \mathbf{y}_k\|,\end{aligned}$$

ainsi, $(\mathbf{y}_k)_{k\in\mathbb{N}}$ satisfait

$$\sum_{k=0}^{+\infty} \|\mathbf{y}_{k+1} - \mathbf{y}_k\| < +\infty. \tag{3.29}$$

(iii) Il ne nous reste plus qu'à démontrer que la limite $\widehat{\mathbf{x}}$ est un point critique de f. Pour cela, posons

$$(\forall k \in \mathbb{N}) \qquad \mathbf{t}_k = \nabla\mathsf{h}(\mathbf{y}_k) + \mathbf{r}_k,$$

tel que $(\mathbf{y}_k, \mathbf{t}_k) \in \operatorname{Graph}\partial\mathsf{f}$, où $\mathbf{r}_k$ est donné dans l'algorithme 3.2. En raisonnant de façon analogue au cheminement suivi pour l'obtention de l'équation (3.26), on a

$$(\forall k \in \mathbb{N}) \qquad \|\mathbf{t}_k\| \leqslant (\beta + \tau\sqrt{\overline{\nu}})\|\mathbf{x}_k - \mathbf{y}_k\|.$$

Étant donné que les suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ et $(\mathbf{y}_k)_{k\in\mathbb{N}}$ convergent toutes les deux vers $\widehat{\mathbf{x}}$, $(\mathbf{y}_k, \mathbf{t}_k)_{k\in\mathbb{N}}$ converge vers $(\widehat{\mathbf{x}}, \mathbf{0})$. Par ailleurs, la remarque 3.1(iii) nous permet d'affirmer que la restriction de f à son domaine est continue. Ainsi, puisque $(\forall k \in \mathbb{N})\ \mathbf{y}_k \in \operatorname{dom}\mathsf{f}$, la suite $(\mathsf{f}(\mathbf{y}_k))_{k\in\mathbb{N}}$ converge vers $\mathsf{f}(\widehat{\mathbf{x}})$. Au final, $\partial\mathsf{f}$ étant fermé (c.f. proposition 2.4), $(\widehat{\mathbf{x}}, \mathbf{0}) \in \operatorname{Graph}\partial\mathsf{f}$. Autrement dit, $\widehat{\mathbf{x}}$ est un point critique de f.

Chapitre 4

Algorithme explicite-implicite : application à un bruit gaussien dépendant

Sommaire

4.1 Introduction

Dans ce chapitre, nous nous intéressons à un problème de restauration/reconstruction d'image où l'objectif est de trouver une estimée d'une image originale à partir d'une version dégradée de celle-ci. Plus précisément, nous nous intéressons au cas où l'image observée peut s'écrire comme la somme de l'image originale dégradée par un opérateur linéaire et d'un bruit gaussien dépendant. Notons que ce modèle d'observation se rencontre dans de nombreux systèmes d'imagerie [Healey et Kondepudy, 1994; Janesick, 2007; Tian *et al.*, 2001] dans lesquels les acquisitions sont dégradées par un bruit gaussien dépendant lié au comptage de photons, et un bruit électronique indépendant. Le bruit gaussien dépendant peut être également vu comme une approximation au second ordre du bruit Poisson-gaussien fréquemment considéré en astronomie, biologie et imagerie médicale [Foi *et al.*, 2008; Jezierska *et al.*, 2012; Li *et al.*, 2012].

Nous verrons que le problème de minimisation associé à ce modèle, issu du MAP, n'est pas convexe. Nous proposerons donc d'utiliser l'algorithme explicite-implicite à métrique variable, étudié dans le chapitre précédent, pour trouver une solution de ce problème. Cela nous permettra de comparer, sur un exemple pratique, notre algorithme avec sa version non préconditionnée.

Le plan de ce chapitre sera le suivant : nous allons étudier le modèle de bruit gaussien dépendant dans la section 4.2. Puis nous appliquerons l'algorithme développé dans le chapitre précédent à ce modèle dans la section 4.3 et donnerons des résultats de simulations dans la section 4.4. Enfin, nous conclurons dans la section 4.5.

4.2 Bruit gaussien dépendant du signal

4.2.1 Estimateur du *Maximum A Posteriori*

On s'intéresse à l'estimation $\widehat{\mathbf{x}} \in \mathbb{R}^N$ d'une image originale $\overline{\mathbf{x}} \in \mathbb{R}^N$ à partir d'une image observée $\mathbf{z} \in \mathbb{R}^M$ de la forme

$$\begin{cases} \mathbf{z} = \mathbf{y} + \mathbf{w}, \\ \mathbf{y} = \mathbf{H}\overline{\mathbf{x}}, \end{cases} \tag{4.1}$$

où $\mathbf{H} \in \mathbb{R}^{M \times N}$ est une matrice modélisant la dégradation subie par l'image (par exemple, un opérateur de convolution) et $\mathbf{w} \in \mathbb{R}^M$ est une réalisation d'un vecteur aléatoire absolument continu $\boldsymbol{w}$ modélisant un bruit additif.

Supposons que $\mathbf{z}$ et $\overline{\mathbf{x}}$ soient des réalisations des vecteurs aléatoires $\boldsymbol{z}$ et $\overline{\boldsymbol{x}}$. Comme nous l'avons vu dans le chapitre 2, on peut alors utiliser une approche de type MAP pour définir l'estimée $\widehat{\mathbf{x}}$ de $\overline{\mathbf{x}}$:

$$\widehat{\mathbf{x}} \in \operatorname*{Argmax}_{\mathbf{x} \in \mathbb{R}^N} \ \mathsf{f}_{\overline{\boldsymbol{x}}|\boldsymbol{z}=\mathbf{z}}(\mathbf{x}), \tag{4.2}$$

où $\mathsf{f}_{\overline{\boldsymbol{x}}|\boldsymbol{z}=\mathbf{z}}$ est la densité de probabilité *a posteriori* de $\overline{\boldsymbol{x}}$ sachant $\boldsymbol{z} = \mathbf{z}$. Dans la propriété 2.1 nous avons exprimé le problème variationnel associé au modèle (4.1) lorsque le bruit est indépendant de l'image.

Bruit dépendant du signal. Dans ce chapitre, nous nous intéressons au cas où le bruit et l'image originale sont inter-dépendants. Dans ce cas, la proposition suivante est vérifiée :

Proposition 4.1. *Soit* $\mathsf{z} \in \mathbb{R}^M$ *défini par l'équation* (4.1). *Une solution* $\widehat{\mathsf{x}}$ *du problème* (4.2) *est donnée par*

$$\widehat{\mathsf{x}} \in \operatorname*{Argmin}_{\mathsf{x}\in\mathbb{R}^N} \ \mathsf{h}(\mathsf{x}) + \mathsf{g}(\mathsf{x}), \tag{4.3}$$

où $\mathsf{h}\colon \mathbb{R}^N \to]-\infty,+\infty]$ *et* $\mathsf{g}\colon \mathbb{R}^N \to]-\infty,+\infty]$ *sont respectivement les termes d'attache aux données et de régularisation définis par*

$$(\forall \mathsf{x} \in \mathbb{R}^N) \quad \begin{cases} \mathsf{h}(\mathsf{x}) = -\log \mathsf{f}_{\boldsymbol{w}|\boldsymbol{y}=\mathsf{H}\mathsf{x}}(\mathsf{z}-\mathsf{H}\mathsf{x}), \\ \mathsf{g}(\mathsf{x}) = -\log \mathsf{f}_{\overline{\boldsymbol{x}}}(\mathsf{x}), \end{cases} \tag{4.4}$$

où $\boldsymbol{y} = \mathsf{H}\overline{\boldsymbol{x}}$.

Démonstration. Il suffit de reprendre les équations (2.9) et (2.10) de la démonstration de la propriété 2.1. En les combinant, on obtient

$$\widehat{\mathsf{x}} \in \operatorname*{Argmin}_{\mathsf{x}\in\mathbb{R}^N} \ \left\{ -\log \mathsf{f}_{\boldsymbol{w}|\mathsf{H}\overline{\boldsymbol{x}}=\mathsf{H}\mathsf{x}}(\mathsf{z}-\mathsf{H}\mathsf{x}) - \log \mathsf{f}_{\overline{\boldsymbol{x}}}(\mathsf{x}) \right\}. \tag{4.5}$$

□

Bruit gaussien dépendant du signal. Supposons maintenant que la variable aléatoire $\boldsymbol{w}$, modélisant le bruit additif dans le modèle (4.1), suive une loi normale de moyenne nulle et de matrice de covariance $\boldsymbol{\Gamma}(\mathsf{y}) \in \mathbb{R}^{M\times M}$, telle que $\boldsymbol{\Gamma}(\mathsf{y}) \succ 0$, dépendant de la réalisation $\mathsf{y} = \mathsf{H}\overline{\mathsf{x}}$ du vecteur aléatoire $\boldsymbol{y} = \mathsf{H}\overline{\boldsymbol{x}}$:

$$\boldsymbol{w}|\boldsymbol{y} = \mathsf{y} \sim \mathcal{N}\left(\mathbf{0}_M, \boldsymbol{\Gamma}(\mathsf{y})\right). \tag{4.6}$$

Le problème variationnel associé à ce modèle s'écrit alors de la façon suivante :

Lemme 4.1. *On considère le modèle* (4.1), *où* w *est une réalisation du vecteur aléatoire* $\boldsymbol{w}$ *dont la loi sachant* $\boldsymbol{y} = \mathsf{y}$ *est donnée par* (4.6) *avec* $\boldsymbol{\Gamma}(\mathsf{y}) \in \mathbb{R}^{M\times M}$, *telle que* $\boldsymbol{\Gamma}(\mathsf{y}) \succ 0$. *Une solution* $\widehat{\mathsf{x}}$ *du problème* (4.2) *est alors donnée par*

$$\widehat{\mathsf{x}} \in \operatorname*{Argmin}_{\mathsf{x}\in\mathbb{R}^N} \ \frac{1}{2}(\mathsf{z}-\mathsf{H}\mathsf{x})^\top \boldsymbol{\Gamma}(\mathsf{H}\mathsf{x})^{-1}(\mathsf{z}-\mathsf{H}\mathsf{x}) + \frac{1}{2}\log\Big(\det\left(\boldsymbol{\Gamma}(\mathsf{H}\mathsf{x})\right)\Big) + \mathsf{g}(\mathsf{x}), \tag{4.7}$$

où $\mathsf{g}\colon \mathbb{R}^N \to]-\infty,+\infty]$ *est une fonction de régularisation dépendant des caractéristiques physiques de l'image originale.*

Démonstration. La fonction de densité de probabilité de $\boldsymbol{w}$ sachant $\boldsymbol{y} = \mathsf{y}$ est donnée par

$$(\forall \mathsf{w} \in \mathbb{R}^M) \quad \mathsf{f}_{\boldsymbol{w}|\boldsymbol{y}=\mathsf{y}}(\mathsf{w}) = \frac{1}{(2\pi)^{N/2}(\det \boldsymbol{\Gamma}(\mathsf{y}))^{1/2}} \exp\left(-\frac{1}{2}\mathsf{w}^\top \boldsymbol{\Gamma}(\mathsf{y})^{-1}\mathsf{w}\right). \tag{4.8}$$

Il suffit alors de combiner cette équation à la proposition 4.1. □

Bruit gaussien décorrélé et dépendant du signal. Dans la suite de ce chapitre, nous nous intéresserons au cas particulier où les coefficients de l'image originale $\overline{\mathbf{x}}$ et de la matrice de dégradation $\mathbf{H}$ sont positifs. De plus, nous supposerons que la matrice de covariance $\mathbf{\Gamma}(\mathbf{H}\overline{\mathbf{x}})$, dépendant de l'image $\mathbf{H}\overline{\mathbf{x}} = ([\mathbf{H}\overline{\mathbf{x}}]^{(m)})_{1\leqslant m\leqslant M} \in [0,+\infty[^M$, est une matrice diagonale donnée par

$$\mathbf{\Gamma}(\mathbf{H}\overline{\mathbf{x}}) = \mathrm{Diag}\left(\left(\sigma_m([\mathbf{H}\overline{\mathbf{x}}]^{(m)})^2\right)_{1\leqslant m\leqslant M}\right), \tag{4.9}$$

où, pour tout $m \in \{1,\dots,M\}$,

$$\begin{aligned}\sigma_m\colon [0,+\infty[&\to]0,+\infty[\\ \mathsf{y} &\mapsto \left(\alpha^{(m)}\mathsf{y}+\beta^{(m)}\right)^{1/2},\end{aligned} \tag{4.10}$$

avec $(\alpha^{(m)},\beta^{(m)}) \in [0,+\infty[\times]0,+\infty[$.

Dans ce cadre, le modèle (4.1) peut se réécrire

$$(\forall m \in \{1,\dots,M\}) \quad \mathsf{z}^{(m)} = [\mathbf{H}\overline{\mathbf{x}}]^{(m)} + \sigma_m\left([\mathbf{H}\overline{\mathbf{x}}]^{(m)}\right)\mathsf{v}^{(m)}, \tag{4.11}$$

où, pour tout $m \in \{1,\dots,M\}$, $\mathsf{v}^{(m)}$ est une réalisation d'une variable aléatoire $v^{(m)}$ suivant une loi normale de moyenne nulle et de variance égale à 1.

Remarque 4.1.

(i) *Lorsque, pour tout $m \in \{1,\dots,M\}$, $\alpha_m = 0$ et $\beta_m = \sigma^2 \in]0,+\infty[$, le bruit gaussien dépendant se réduit au bruit blanc gaussien (c.f. section 2.2.3 du chapitre 2).*

(ii) *En pratique, afin d'assurer la positivité de l'estimée $\widehat{\mathbf{x}}$, on peut utiliser comme terme de régularisation la fonction indicatrice sur l'orthant positif :*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}(\mathbf{x}) = \iota_{[0,+\infty[^N}(\mathbf{x}). \tag{4.12}$$

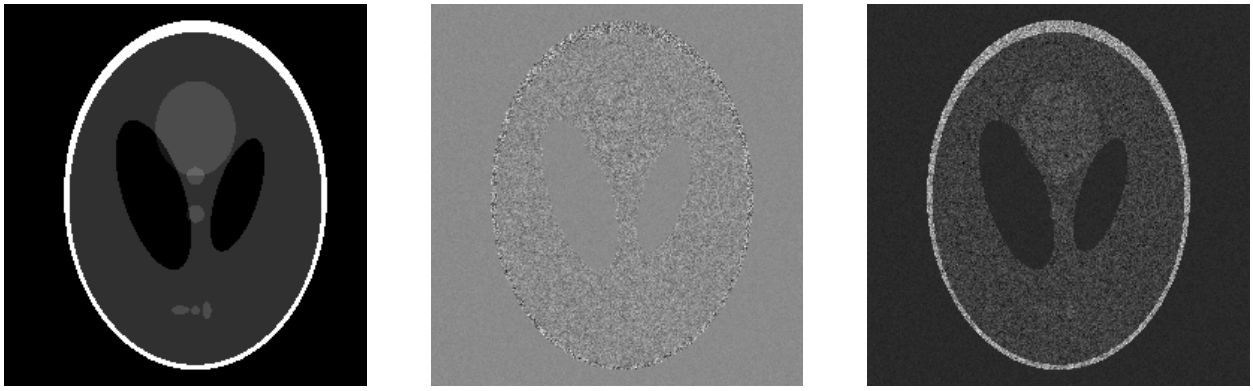
Dans la figure 4.1, une comparaison est faite entre une image bruitée par un bruit blanc gaussien (figures 4.1(b) et (c)) et une image bruitée par un bruit dépendant gaussien (figures 4.1(d) et (e)). L'image originale (figure 4.1(a)) correspond à l'image *phantom* générée à l'aide de Matlab.

Proposition 4.2. *On considère le modèle (4.11), où $\mathbf{H} \in [0,+\infty[^{M\times N}$ et $\overline{\mathbf{x}} \in [0,+\infty[^N$. L'estimée $\widehat{\mathbf{x}}$ de $\overline{\mathbf{x}}$ associée au MAP est définie par*

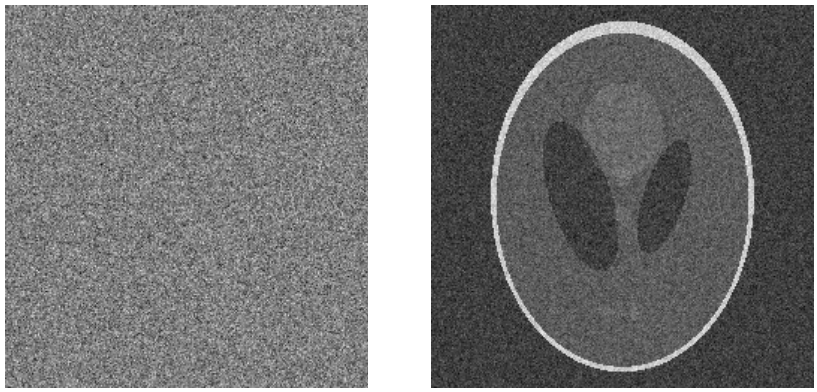
$$\widehat{\mathbf{x}} \in \underset{\mathbf{x}\in[0,+\infty[^N}{\mathrm{Argmin}} \ \{\mathsf{h}_1(\mathbf{x}) + \mathsf{h}_2(\mathbf{x}) + \mathsf{g}(\mathbf{x})\}, \tag{4.13}$$

où h_1 et h_2 sont des fonctions de $[0,+\infty[^N$ vers $]-\infty,+\infty]$, données par

$$(\forall \mathbf{x} \in [0,+\infty[^N) \quad \mathsf{h}_1(\mathbf{x}) = \sum_{m=1}^{M} \phi_m([\mathbf{H}\mathbf{x}]^{(m)}) \ \text{ et } \ \mathsf{h}_2(\mathbf{x}) = \sum_{m=1}^{M} \psi_m([\mathbf{H}\mathbf{x}]^{(m)}) \tag{4.14}$$



(a) (b) (c)



(d) (e)

Figure 4.1 – (a) *Image originale* phantom *de dimension* 256×256. (b) *Bruit gaussien dépendant de l'image originale avec* $\alpha^{(m)} \equiv 0.1$ *et* $\beta^{(m)} \equiv 0.001$. (c) *Image bruitée avec le bruit dépendant gaussien.* (d) *Bruit blanc gaussien de variance* $\sigma = 0.1154$. (e) *Image bruitée avec le bruit blanc gaussien. Les images bruitées* (c) *et* (e) *ont le même RSB de* 6.6 *dB.*

avec, pour tout $m \in \{1, \ldots, M\}$,

$$(\forall \mathsf{y} \in [0, +\infty[) \quad \phi_m(\mathsf{y}) = \frac{1}{2} \frac{(\mathsf{z}^{(m)} - \mathsf{y})^2}{\alpha^{(m)}\mathsf{y} + \beta^{(m)}}, \tag{4.15}$$

$$\psi_m(\mathsf{y}) = \frac{1}{2} \log\left(\alpha^{(m)}\mathsf{y} + \beta^{(m)}\right), \tag{4.16}$$

et $\mathsf{g}\colon \mathbb{R}^N \to]-\infty, +\infty]$ *une fonction de régularisation dépendant des caractéristiques physiques de l'image.*

Démonstration. Ce résultat est obtenu en injectant les équations (4.9) et (4.10) dans le lemme 4.1. □

4.2.2 Fonction d'attache aux données

Dans cette section, nous étudions les propriétés des fonctions h_1 et h_2 définies dans la proposition 4.2. Pour cela nous définissons les fonctions

$$(\forall \mathsf{y} \in [0,+\infty[) \quad \chi_m(\mathsf{y}) = \phi_m(\mathsf{y}) + \psi_m(\mathsf{y}), \tag{4.17}$$

$$(\forall \mathbf{H} \in [0,+\infty[^{M\times N})(\forall \mathbf{x} \in [0,+\infty[^N) \quad \mathsf{h}(\mathbf{x}) = \mathsf{h}_1(\mathbf{x}) + \mathsf{h}_2(\mathbf{x}) = \sum_{m=1}^{M} \chi_m([\mathbf{H}\mathbf{x}]^{(m)}). \tag{4.18}$$

Pour tout $m \in \{1,\dots,M\}$, les dérivées premières et secondes des fonctions ϕ_m et ψ_m en $\mathsf{y} \in [0,+\infty[$ [1] sont données par

$$\dot{\phi}_m(\mathsf{y}) = \frac{(\mathsf{y}-\mathsf{z}^{(m)})(\alpha^{(m)}\mathsf{y}+\alpha^{(m)}\mathsf{z}^{(m)}+2\beta^{(m)})}{2(\alpha^{(m)}\mathsf{y}+\beta^{(m)})^2}, \quad \ddot{\phi}_m(\mathsf{y}) = \frac{(\alpha^{(m)}\mathsf{z}^{(m)}+\beta^{(m)})^2}{(\alpha^{(m)}\mathsf{y}+\beta^{(m)})^3}, \tag{4.19}$$

$$\dot{\psi}_m(\mathsf{y}) = \frac{\alpha^{(m)}}{2(\alpha^{(m)}\mathsf{y}+\beta^{(m)})}, \qquad \ddot{\psi}_m(\mathsf{y}) = -\frac{(\alpha^{(m)})^2}{2(\alpha^{(m)}\mathsf{y}+\beta^{(m)})^2}. \tag{4.20}$$

Ainsi, les dérivées première et seconde de la fonction χ_m sont données par

$$\dot{\chi}_m(\mathsf{y}) = \frac{(\mathsf{y}-\mathsf{z}^{(m)})(\alpha^{(m)}\mathsf{y}+\alpha^{(m)}\mathsf{z}^{(m)}+2\beta^{(m)})}{2(\alpha^{(m)}\mathsf{y}+\beta^{(m)})^2} + \frac{\alpha^{(m)}}{2(\alpha^{(m)}\mathsf{y}+\beta^{(m)})}, \tag{4.21}$$

$$\ddot{\chi}_m(\mathsf{y}) = \frac{(\alpha^{(m)}\mathsf{z}^{(m)}+\beta^{(m)})^2}{(\alpha^{(m)}\mathsf{y}+\beta^{(m)})^3} - \frac{1}{2}\left(\frac{\alpha^{(m)}}{\alpha^{(m)}\mathsf{y}+\beta^{(m)}}\right)^2. \tag{4.22}$$

Convexité. D'une part, pour tout $m \in \{1,\dots,M\}$, d'après (4.19), on peut remarquer que la fonction $\ddot{\phi}_m$ est positive sur $[0,+\infty[$, et donc ϕ_m est convexe sur $[0,+\infty[$. Par conséquent, la fonction h_1 est convexe sur $[0,+\infty[^N$.

D'autre part, pour tout $m \in \{1,\dots,M\}$, d'après (4.20), la fonction $\ddot{\psi}_m$ est négative sur $[0,+\infty[$, et donc ψ_m est concave sur $[0,+\infty[$. Par conséquent, la fonction h_1 est concave sur $[0,+\infty[^N$.

Ainsi, la fonction h est une différence de fonctions convexes sur $[0,+\infty[^N$. Dans la proposition suivante, nous allons étudier plus en détails la courbure de la fonction χ_m.

Proposition 4.3. *Soit $m \in \{1,\dots,M\}$. Posons*

$$\mathsf{y}_{\max}^{(m)} = \frac{1}{\alpha^{(m)}}\left(2\Big(\frac{\alpha^{(m)}\mathsf{z}^{(m)}+\beta^{(m)}}{\alpha^{(m)}}\Big)^2 - \beta^{(m)}\right), \tag{4.23}$$

où $\alpha^{(m)}$, $\beta^{(m)}$ sont définis dans (4.10) *et $\mathsf{z}^{(m)}$ correspond au m-ème élément de l'image observée $\mathbf{z}$ définie par* (4.11)*. La fonction χ_m définie par* (4.17) *est convexe sur $[0,\mathsf{y}_{\max}^{(m)}]$ et concave sur $]\mathsf{y}_{\max}^{(m)},+\infty[$.*

1. On considèrera la dérivée à droite lorsque $\mathsf{y}=0$.

Démonstration. Soit $m \in \{1, \dots, M\}$. D'après (4.22), la dérivée seconde de χ_m est donnée par

$$(\forall \mathsf{y} \in [0, +\infty[) \quad \ddot{\chi}_m(\mathsf{y}) = \frac{2(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2 - (\alpha^{(m)})^2(\alpha^{(m)}\mathsf{y} + \beta^{(m)})}{2(\alpha^{(m)}\mathsf{y} + \beta^{(m)})^3}.$$

La fonction χ_m est donc convexe pour tout $\mathsf{y} \in [0, +\infty[$ satisfaisant $\ddot{\chi}_m(\mathsf{y}) \geqslant 0$, ce qui est équivalent à

$$2\left(\frac{\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)}}{\alpha^{(m)}}\right)^2 \geqslant \alpha^{(m)}\mathsf{y} + \beta^{(m)}.$$

Cette inégalité est vérifiée lorsque $\mathsf{y} \leqslant \mathsf{y}_{\max}^{(m)}$. □

Constantes de Lipschitz. Comme nous l'avons vu dans les chapitres précédents, une propriété intéressante permettant de construire des algorithmes d'optimisation est la propriété de gradient Lipschitz. C'est ce que nous allons étudier dans ce paragraphe.

Proposition 4.4. *Soit $m \in \{1, \dots, M\}$. La dérivée $\dot{\chi}_m$ donnée par (4.21) de χ_m est une fonction $\mu^{(m)}$-Lipschitzienne sur $[0, +\infty[$, où $\mu^{(m)}$ est donnée par*

$$\mu^{(m)} = \max\left\{\frac{1}{(\beta^m)^2}\left(\frac{(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2}{\beta^{(m)}} - \frac{(\alpha^{(m)})^2}{2}\right), \frac{1}{2}\frac{(\alpha^{(m)})^6}{27(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^4}\right\}. \tag{4.24}$$

La démonstration de cette proposition est donnée dans l'annexe 4.A. On peut en déduire la constante de Lipschitz du gradient de la fonction d'attache aux données h :

Proposition 4.5. *La fonction h est de gradient Lipschitz, de constante*

$$L = \|\mathbf{H}^\top \operatorname{Diag}(\boldsymbol{\mu})\mathbf{H}\|, \tag{4.25}$$

où $\boldsymbol{\mu} = (\mu^{(m)})_{1 \leqslant m \leqslant M}$, avec, pour tout $m \in \{1, \dots, M\}$, $\mu^{(m)}$ la constante donnée par (4.24).

4.3 Application de l'algorithme explicite-implicite à métrique variable

4.3.1 Problème d'optimisation

Dans cette section, nous considérons un problème inverse où une version dégradée $\mathbf{z} = (\mathsf{z}^{(m)})_{1 \leqslant m \leqslant M}$ d'une image originale $\overline{\mathbf{x}} \in [0, +\infty[^N$ est observée suivant le modèle (4.10)-(4.11), où $\mathbf{H} \in [0, +\infty[^M$. Notre objectif est de produire une extimation $\widehat{\mathbf{x}} \in [0, +\infty[^N$ de l'image originale $\overline{\mathbf{x}}$ à partir des données observées $\mathbf{z}$.

Comme nous l'avons vu dans la proposition 4.2, $\overline{\mathbf{x}}$ peut être estimée en résolvant le problème de minimisation (4.13). Ce problème faisant intervenir la fonction différentiable non convexe h définie par (4.18), nous proposons une solution utilisant l'algorithme 3.1 explicite-implicite à métrique variable.

Remarque 4.2.
D'après le théorème 3.1, afin d'assurer la convergence de l'algorithme 3.1 (ou de sa version inexacte donnée par l'algorithme 3.2), la fonction

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{f}(\mathbf{x}) = \mathsf{h}_1(\mathbf{x}) + \mathsf{h}_2(\mathbf{x}) + \mathsf{g}(\mathbf{x}), \tag{4.26}$$

où h_1, h_2 *sont définies par* (4.14), *doit satisfaire l'inégalité de Kurdyka-Łojasiewicz donnée dans la définition 2.20. Comme indiqué dans la remarque 2.11, ce n'est pas le cas de la fonction logarithme. Nous remplacerons donc, pour tout* $m \in \{1, \ldots, M\}$, *la fonction* ψ_m *donnée dans* (4.16) *par la fonction*

$$(\forall \mathsf{y} \in [0, +\infty[) \quad \widehat{\psi}_m(\mathsf{y}) = \frac{1}{2}\widehat{\log}(\alpha^{(m)}\mathsf{y} + \beta^{(m)}). \tag{4.27}$$

Dans l'équation (4.27), $\widehat{\log}$ *est une approximation semi-algébrique du logarithme définie sur* $]0, +\infty[$, *qui, comme la fonction originale, est concave et de gradient Lipschitz sur tout intervalle* $[\underline{\beta}, +\infty[$, *avec* $\underline{\beta} \in]0, +\infty[$. *Ce type d'approximation est souvent utilisée pour l'implémentation numérique de la fonction logarithme* [Dahlquist et Bjorck, 2003, Chap.4]. *Ainsi, la fonction* h *sera la somme de la fonction* h_1 *définie dans* (4.14) *et de la fonction*

$$(\forall \mathbf{x} \in [0, +\infty[^N) \quad \widehat{\mathsf{h}}_2(\mathbf{x}) = \sum_{m=1}^{M} \widehat{\psi}_m([\mathbf{H}\mathbf{x}]^{(m)}). \tag{4.28}$$

D'autre part, nous considérons une fonction de pénalisation hybride, composée de deux termes $\mathsf{g} = \mathsf{g}_1 + \mathsf{g}_2$. Pour le premier terme, afin de contraindre la dynamique de l'image originale $\overline{\mathbf{x}}$, nous définissons

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}_1(\mathbf{x}) = \iota_{[\mathsf{x}_{\min}, \mathsf{x}_{\max}]^N}(\mathbf{x}), \tag{4.29}$$

où $\mathsf{x}_{\min} \in [0, +\infty[$ et $\mathsf{x}_{\max} \in]\mathsf{x}_{\min}, +\infty[$ sont, respectivement, les valeurs minimales et maximales des composantes de $\overline{\mathbf{x}}$. Le second terme sera choisi, dans un premier exemple de simulation, comme étant une régularisation de type NLTV [Gilboa et Osher, 2008], puis dans un second exemple nous considèrerons une régularisation parcimonieuse dans une approche de type trame à l'analyse [Chaâri *et al.*, 2009; Elad *et al.*, 2007; Pustelnik *et al.*, 2012]. Comme nous l'avons vu dans la section 2.2.4.2, ces deux termes de régularisation s'écrivent

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}_2(\mathbf{x}) = \sum_{j=1}^{J} \mathsf{g}_{2,j}(\mathbf{F}_j\mathbf{x}), \tag{4.30}$$

où, pour tout $j \in \{1, \ldots, J\}$, $\mathsf{g}_{2,j}$ est une fonction semi-algébrique et $\mathbf{F}_j$ correspond soit à l'opérateur de gradient non local intervenant dans la NLTV, soit à un opérateur de trame ajusté. Ainsi, dans les deux cas, l'hypothèse 3.1 et l'inégalité de KL sont bien vérifiées.

4.3.2 Construction de la majorante

L'accélération des algorithmes 3.1 et 3.2 repose sur l'utilisation de majorantes quadratique approximant la fonction h autour de l'itérée courante, et dont les matrices de courbure $(\mathbf{A}_k(\mathbf{x}_k))_{k\in\mathbb{N}}$

sont simples à implémenter. Dans cette section, nous proposons une technique de majoration qui combine des stratégies proposées dans [Hunter et Lange, 2004] et [Erdogan et Fessler, 1999], afin de construire des matrices diagonales $(\mathbf{A}_k(\mathbf{x}_k))_{k\in\mathbb{N}}$ satisfaisant l'hypothèse 3.2.

Comme nous l'avons remarqué dans la section 4.2.2, sur son domaine de définition, h est la somme d'une fonction convexe h_1 définie par (4.14) et d'une fonction concave $\widehat{\mathsf{h}}_2$ définie par (4.28). Nous allons majorer ces deux termes séparément.

Pour tout $k \in \mathbb{N}$, soit $\mathbf{x}_k$ la k-ème itérée générée par l'algorithme 3.2. Une fonction majorant $\widehat{\mathsf{h}}_2$ sur $[0,+\infty[^N$ en $\mathbf{x}_k$ est

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}_2(\mathbf{x},\mathbf{x}_k) = \widehat{\mathsf{h}}_2(\mathbf{x}_k) + \left\langle \mathbf{x}-\mathbf{x}_k, \nabla\widehat{\mathsf{h}}_2(\mathbf{x}_k)\right\rangle, \tag{4.31}$$

où $\nabla\widehat{\mathsf{h}}_2(\mathbf{x}_k)$ désigne le gradient de $\widehat{\mathsf{h}}_2$ en $\mathbf{x}_k$.

Le lemme suivant nous permet de construire une majorante quadratique de la fonction h_1 en $\mathbf{x}_k$. Pour cela, nous avons besoin de définir, dans un premier temps, la fonction multivaluée

$$\Omega\colon [0,+\infty[^M \to \mathbb{R}^M\colon (\mathsf{y}^{(m)})_{1\leqslant m\leqslant M} \mapsto \big(\omega_m(\mathsf{y}^{(m)})\big)_{1\leqslant m\leqslant M},$$

où, pour tout $m \in \{1,\ldots,M\}$,

$$(\forall \mathsf{y} \in [0,+\infty[) \quad \omega_m(\mathsf{y}) = \begin{cases} \ddot{\phi}_m(0), & \text{si} \quad \mathsf{y}=0,\\ 2\,\frac{\phi_m(0)-\phi_m(\mathsf{y})+\mathsf{y}\dot{\phi}_m(\mathsf{y})}{\mathsf{y}^2}, & \text{si} \quad \mathsf{y}>0, \end{cases} \tag{4.32}$$

et ϕ_m est définie par (4.15).

Lemme 4.2. *Soit* h_1 *définie par* (4.14) *où* $\mathbf{H} = \left(\mathsf{H}^{(m,n)}\right)_{1\leqslant m\leqslant M,1\leqslant n\leqslant N} \in [0,+\infty[^{M\times N}$. *Pour tout* $k \in \mathbb{N}$, *on pose*

$$\mathbf{A}_k(\mathbf{x}_k) = \mathrm{Diag}\left(\mathbf{P}^\top \Omega(\mathbf{H}\mathbf{x}_k)\right) + \varepsilon\,\mathbf{I}_N, \tag{4.33}$$

où $\varepsilon \in [0,+\infty[$ *et* $\mathbf{P} = \left(\mathsf{P}^{(m,n)}\right)_{1\leqslant m\leqslant M,1\leqslant n\leqslant N}$ *est la matrice dont les éléments sont donnés par*

$$(\forall m \in \{1,\ldots,M\})(\forall n \in \{1,\ldots,N\}) \quad \mathsf{P}^{(m,n)} = \mathsf{H}^{(m,n)} \sum_{p=1}^{N} \mathsf{H}^{(m,p)}. \tag{4.34}$$

La fonction q_1 *définie par*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}_1(\mathbf{x},\mathbf{x}_k) = \mathsf{h}_1(\mathbf{x}_k) + \langle \mathbf{x}-\mathbf{x}_k, \nabla\mathsf{h}_1(\mathbf{x}_k)\rangle + \frac{1}{2}\|\mathbf{x}-\mathbf{x}_k\|^2_{\mathbf{A}_k(\mathbf{x}_k)}, \tag{4.35}$$

est alors une majorante quadratique de h_1 *sur* $[0,+\infty[^N$ *en* $\mathbf{x}_k$.

Démonstration. Pour tout $m \in \{1,\ldots,M\}$, ϕ_m est convexe et infiniment dérivable sur $[0,+\infty[$. On pose

$$(\forall (\mathsf{x},\mathsf{x}') \in [0,+\infty[^2) \quad \rho_m(\mathsf{x},\mathsf{x}') = \phi_m(\mathsf{x}') + (\mathsf{x}-\mathsf{x}')\dot{\phi}_m(\mathsf{x}') + \frac{1}{2}\omega_m(\mathsf{x}')\,(\mathsf{x}-\mathsf{x}')^2, \tag{4.36}$$

où les fonctions $\dot{\phi}_m$ et ω_m sont définies, respectivement, par (4.15) et (4.32).

Si $\alpha^{(m)} = 0$, alors ϕ_m est une fonction quadratique et on a

$$(\forall (\mathsf{x}, \mathsf{x}') \in [0, +\infty[^2) \quad \phi_m(\mathsf{x}) = \rho_m(\mathsf{x}, \mathsf{x}').$$

Supposons maintenant que $\alpha^{(m)} \in]0, +\infty[$. La dérivée troisième de ϕ_m est donnée par

$$(\forall \mathsf{x} \in [0, +\infty[) \quad \dddot{\phi}_m(\mathsf{x}) = -3\alpha^{(m)} \frac{(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2}{(\alpha^{(m)}\mathsf{x} + \beta^{(m)})^4},$$

et est strictement négative. Il apparaît alors que $\dot{\phi}_m$ est une fonction strictement concave sur $[0, +\infty[$. En utilisant une version simplifiée de la démonstration donnée dans [Erdogan et Fessler, 1999, Annexe B], nous allons démontrer la positivité de la fonction $\rho_m(\cdot, \mathsf{x}') - \phi_m$ sur $[0, +\infty[$, pour tout $\mathsf{x}' \in]0, +\infty[$. La dérivée seconde de cette dernière fonction est donnée par

$$(\forall \mathsf{x} \in [0, +\infty[) \quad \ddot{\rho}_m(\mathsf{x}, \mathsf{x}') - \ddot{\phi}_m(\mathsf{x}) = \omega_m(\mathsf{x}') - \ddot{\phi}_m(\mathsf{x}). \tag{4.37}$$

De plus, d'après la formule de Taylor de second ordre, il existe $\tilde{\mathsf{x}} \in]0, \mathsf{x}'[$ tel que

$$\phi_m(0) = \phi_m(\mathsf{x}') - \mathsf{x}'\dot{\phi}_m(\mathsf{x}') + \frac{1}{2}\mathsf{x}'^2 \ddot{\phi}_m(\tilde{\mathsf{x}}).$$

Ainsi, en utilisant les équations (4.32) et (4.37), on obtient

$$(\forall \mathsf{x} \in [0, +\infty[) \quad \ddot{\rho}_m(\mathsf{x}, \mathsf{x}') - \ddot{\phi}_m(\mathsf{x}) = \ddot{\phi}_m(\tilde{\mathsf{x}}) - \ddot{\phi}_m(\mathsf{x}). \tag{4.38}$$

Or, puisque $\dddot{\phi}_m$ est strictement négative, $\ddot{\phi}_m$ est strictement décroissante, et (4.38) implique que $\dot{\rho}_m(\cdot, \mathsf{x}') - \dot{\phi}_m$ est d'abord strictement décroissante sur $]0, \tilde{u}[$, puis strictement croissante sur $]\tilde{u}, +\infty[$. D'une part, (4.32) et (4.36) nous donnent

$$\begin{cases} \rho_m(\mathsf{x}', u') - \phi_m(u') &= 0, \\ \rho_m(0, u') - \phi_m(0) &= 0. \end{cases} \tag{4.39}$$

Ainsi, d'après le théorème des valeurs intermédiaires, il existe $\mathsf{x}^* \in]0, \mathsf{x}'[$ tel que $\dot{\rho}_m(\mathsf{x}^*, \mathsf{x}') - \dot{\phi}_m(\mathsf{x}^*) = 0$. D'autre part, d'après l'équation (4.36), on a

$$\dot{\rho}_m(\mathsf{x}', \mathsf{x}') - \dot{\phi}_m(\mathsf{x}') = 0.$$

Par conséquent, par monotonie de la fonction $\dot{\rho}_m(\cdot, \mathsf{x}') - \dot{\rho}_1^{(m)}$, on peut déduire que x^* est l'unique zéro de cette fonction sur $]0, u'[$, et

$$\begin{cases} \dot{\rho}_m(\mathsf{x}, \mathsf{x}') - \dot{\phi}_m(\mathsf{x}') &> 0, \quad \forall \mathsf{x} \in [0, \mathsf{x}^*[, \\ \dot{\rho}_m(\mathsf{x}, \mathsf{x}') - \dot{\phi}_m(\mathsf{x}') &< 0, \quad \forall \mathsf{x} \in]\mathsf{x}^*, \mathsf{x}'[, \\ \dot{\rho}_m(\mathsf{x}, \mathsf{x}') - \dot{\phi}_m(\mathsf{x}') &> 0, \quad \forall \mathsf{x} \in]\mathsf{x}', +\infty[. \end{cases} \tag{4.40}$$

L'équation (4.40) implique que $\rho_m(\cdot,\mathsf{x}') - \phi_m(\cdot)$ est strictement croissante sur $[0,\mathsf{x}^*[$, strictement décroissante sur $]\mathsf{x}^*,\mathsf{x}'[$ et strictement croissante sur $]\mathsf{x}',+\infty[$. Par conséquent, d'après (4.39),

$$(\forall(\mathsf{x},\mathsf{x}') \in [0,+\infty[\times]0,+\infty[) \quad \phi_m(\mathsf{x}) \leqslant \rho_m(\mathsf{x},\mathsf{x}'). \tag{4.41}$$

De plus, d'après l'expression de $\ddot{\phi}_m$ dans (4.19), on peut en déduire qu'elle est maximale en zéro, et les équations (4.32) et (4.36) montrent que

$$(\forall \mathsf{x} \in [0,+\infty[) \quad \phi_m(\mathsf{x}) \leqslant \rho_m(\mathsf{x},0). \tag{4.42}$$

Ainsi, en combinant les équations (4.41) et (4.42), on obtient

$$(\forall(\mathsf{x},\mathsf{x}') \in [0,+\infty[^2) \quad \phi_m(\mathsf{x}) \leqslant \rho_m(\mathsf{x},\mathsf{x}'), \tag{4.43}$$

qui, comme nous l'avons souligné précédemment, reste satisfaite lorsque $\alpha^{(m)} = 0$.

La majoration donnée dans (4.43) implique que, pour tout $k \in \mathbb{N}$,

$$(\forall \mathbf{x} \in [0,+\infty[^N) \quad \mathsf{h}_1(\mathbf{x}) \leqslant \mathsf{h}_1(\mathbf{x}_k) + \langle \mathbf{x}-\mathbf{x}_k, \nabla \mathsf{h}_1(\mathbf{x}_k)\rangle + \frac{1}{2}\left\langle \mathbf{H}\mathbf{x}-\mathbf{H}\mathbf{x}_k, \mathrm{Diag}\left(\Omega(\mathbf{H}\mathbf{x}_k)\right)(\mathbf{H}\mathbf{x}-\mathbf{H}\mathbf{x}_k)\right\rangle.$$

On définit la matrice $(\Lambda^{(m,n)})_{1\leqslant m\leqslant M, 1\leqslant n\leqslant N} \in [0,+\infty[^{M\times N}$ telle que

$$(\forall (m,n) \in \{1,\dots,M\}\times\{1,\dots,N\}) \quad \Lambda^{(m,n)} = \begin{cases} 0, & \text{si } \mathsf{H}^{(m,n)} = 0, \\ \dfrac{\mathsf{H}^{(m,n)}}{\sum_{p=1}^N \mathsf{H}^{(m,p)}}, & \text{si } \mathsf{H}^{(m,n)} > 0. \end{cases}$$

D'après l'inégalité de Jensen et (4.34), pour tout $m \in \{1,\dots,M\}$ et pour tout $\mathbf{x} \in [0,+\infty[^N$,

$$\left([\Lambda^{(m,1)},\dots,\Lambda^{(m,N)}](\mathbf{x}-\mathbf{x}_k)\right)^2 \leqslant \sum_{n=1}^N \Lambda^{(m,n)}\left(\mathsf{x}^{(n)}-\mathsf{x}_k^{(n)}\right)^2,$$

ce qui est équivalent à

$$\left([\mathsf{H}^{(m,1)},\dots,\mathsf{H}^{(m,N)}](\mathbf{x}-\mathbf{x}_k)\right)^2 \leqslant \sum_{n=1}^N \mathsf{P}^{(m,n)}\left(\mathsf{x}^{(n)}-\mathsf{x}_k^{(n)}\right)^2.$$

Puisque la convexité de ϕ_m, pour tout $m \in \{1,\dots,M\}$, implique la positivité de $\omega^{(m)}$ sur $[0,+\infty[$, on peut en déduire que

$$\left\langle \mathbf{H}\mathbf{x}-\mathbf{H}\mathbf{x}_k, \mathrm{Diag}\left(\Omega(\mathbf{H}\mathbf{x}_k)\right)(\mathbf{H}\mathbf{x}-\mathbf{H}\mathbf{x}_k)\right\rangle \leqslant \left\langle \mathbf{x}-\mathbf{x}_k, \mathrm{Diag}\left(\mathbf{P}^\top\Omega(\mathbf{H}\mathbf{x}_k)\right)(\mathbf{x}-\mathbf{x}_k)\right\rangle.$$

La fonction $\mathsf{q}_1(\cdot,\mathbf{x}_k)$ définie par (4.35) est donc une majorante quadratique de h_1 sur $[0,+\infty[^N$ en $\mathbf{x}_k$. □

Nous pouvons donc déduire de l'équation (4.31) et du lemme 4.2 que l'hypothèse 3.2(i) est satisfaite pour $\mathsf{q} = \mathsf{q}_1 + \mathsf{q}_2$.

Notons que, pour tout $m \in \{1, \dots, M\}$, la dérivée de ω_m en $\mathsf{x}' \in]0, +\infty[$ est donnée par

$$\begin{aligned}\dot{\omega}_m(\mathsf{x}') &= 2\frac{\mathsf{x}'^2\ddot{\phi}_m(\mathsf{x}') - 2\big(\phi_m(0) - \phi_m(\mathsf{x}') + \mathsf{x}'\dot{\phi}_m(\mathsf{x}')\big)}{\mathsf{x}'^3} \\ &= 2\frac{\ddot{\phi}_m(\mathsf{x}') - \ddot{\phi}_m(\tilde{\mathsf{x}})}{\mathsf{x}'},\end{aligned}$$

où $\tilde{\mathsf{x}} \in]0, \mathsf{x}'[$. Or, la dérivée troisième de ϕ_m étant négative, ω_m est une fonction positive strictement décroissante. De plus, d'après l'expression de la fonction g, $\operatorname{dom}\mathsf{g} = [\mathsf{x}_{\min}, \mathsf{x}_{\max}]^N$, donc son image $\{\mathbf{H}\mathbf{x} \mid \mathbf{x} \in \operatorname{dom}\mathsf{g}\}$ est un ensemble compact. Ainsi, pour tout $m \in \{1, \dots, M\}$ et pour tout $\mathbf{x} = (\mathsf{x}^{(m)})_{1 \leqslant m \leqslant M}$, la fonction $\mathsf{x}^{(m)} \mapsto \omega_m([\mathbf{H}\mathbf{x}]^{(m)})$ admet un minimum noté $\underline{\omega}_m > 0$ sur $\operatorname{dom}\mathsf{g}$. Ainsi, l'hypothèse 3.2(ii) est satisfaite pour les matrices $(\mathbf{A}_k(\mathbf{x}_k))_{k\in\mathbb{N}}$ définies par (4.33) avec

$$\begin{cases} \underline{\nu} = \varepsilon + \min\limits_{1 \leqslant n \leqslant N} \sum_{m=1}^{M} \mathsf{P}^{(m,n)}\, \underline{\omega}_m, \\ \overline{\nu} = \varepsilon + \max\limits_{1 \leqslant n \leqslant N} \sum_{m=1}^{M} \mathsf{P}^{(m,n)}\, \ddot{\phi}_m(0). \end{cases} \tag{4.44}$$

Remarque 4.3.

Si aucune des colonnes de la matrice $\mathbf{H}$ *n'est nulle, nous pouvons choisir* $\varepsilon = 0$ *dans l'équation* (4.44)*. Dans le cas contraire, il faut choisir* $\varepsilon > 0$.

4.3.3 Implémentation de l'étape proximale

Dans la section suivante nous donnerons deux exemples de simulations pour lesquelles, comme nous l'avons indiqué, nous utiliserons deux régularisations différentes. La première sera une régularisation NLTV et l'autre une régularisation de type trame à l'analyse. Dans les deux cas, à chaque itération $k \in \mathbb{N}$, il nous faut calculer la variable $\mathbf{y}_k$ de l'algorithme 3.1 correspondant à l'opérateur proximal de la fonction g au point $\widetilde{\mathbf{x}}_k = \mathbf{x}_k - \gamma_k \mathbf{A}_k(\mathbf{x}_k)\nabla\mathsf{h}(\mathbf{x}_k)$, relativement à la métrique induite par $\gamma_k^{-1}\mathbf{A}_k(\mathbf{x}_k)$, i.e.

$$\mathbf{y}_k = \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N} \ \Big\{ \sum_{j=1}^{J} \mathsf{g}_{2,j}(\mathbf{F}_j\mathbf{x}) + \iota_{[\mathsf{x}_{\min},\mathsf{x}_{\max}]^N}(\mathbf{x}) + \frac{1}{2}\|\mathbf{x} - \widetilde{\mathbf{x}}_k\|^2_{\gamma_k^{-1}\mathbf{A}_k(\mathbf{x}_k)} \Big\}. \tag{4.45}$$

En raison de la présence des opérateurs linéaires $(\mathbf{F}_j)_{1\leqslant j\leqslant J}$, il n'est pas possible d'obtenir une expression explicite de $\mathbf{y}_k$. Cependant, dans le chapitre 3, nous avons montré que la convergence de l'algorithme 3.1 reste assurée lorsque l'opérateur proximal est calculé de façon approchée. Ainsi, nous calculerons (4.45) à l'aide de sous-itérations. Pour cela, remarquons que l'équation (4.45) est équivalente à

$$\begin{aligned}\mathbf{y}_k = \gamma_k^{1/2}\mathbf{A}_k(\mathbf{x}_k)^{-1/2} \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N} \ \Big\{ &\frac{1}{2}\|\mathbf{x} - \gamma_k^{-1/2}\mathbf{A}_k(\mathbf{x}_k)^{1/2}\widetilde{\mathbf{x}}_k\|^2 \\ &+ \gamma_k^{1/2}\sum_{j=1}^{J} \mathsf{g}_{2,j}(\mathbf{F}_j\mathbf{A}_k(\mathbf{x}_k)^{-1/2}\mathbf{x}) + \iota_{[\mathsf{x}_{\min},\mathsf{x}_{\max}]^N}(\gamma_k^{1/2}\mathbf{A}_k(\mathbf{x}_k)^{-1/2}\mathbf{x}) \Big\}.\end{aligned}$$

Il existe divers algorithmes permettant de résoudre ce problème d'optimisation. Les résultats numériques présentés dans la section suivante ont été obtenus en utilisant l'algorithme explicite-implicite dual de [Combettes *et al.*, 2011].

4.4 Résultats de simulation

Nous allons maintenant évaluer les performances de notre algorithme sur deux exemples de simulations. Dans le premier exemple, nous traiterons une application en restauration d'image et, dans le second exemple, une application en reconstruction d'image en tomographie.

4.4.1 Restauration d'image

Dans le premier scénario considéré, l'image originale $\overline{\mathbf{x}}$ correspond à l'image standard *Peppers* de taille 256×256 (http://sipi.usc.edu/database/). Cette image est dégradée par un opérateur de flou $\mathbf{H}$ dont le noyau correspond à un flou gaussien d'écart type 1 et de taille 7×7. De plus, l'image résultante $\mathbf{H}\overline{\mathbf{x}}$ est bruitée par un bruit gaussien dépendant du signal tel que décrit par le modèle (4.11) où, pour tout $m \in \{1, \ldots, M\}$, $\alpha^{(m)} = 0.5$ et $\beta^{(m)} = 1$. Dans ce cas, $\mathsf{x}_{\min} = 0$, $\mathsf{x}_{\max} = 255$, et la constante de Lipschitz de $\nabla \mathsf{h}$ est égale à $L = 18668$.

Dans cette application, on considère comme terme de régularisation la NLTV [Gilboa et Osher, 2008] (voir aussi le paragraphe sur la TV dans la section 2.2.4.2). Ainsi, la fonction g_2 est donnée par (4.30) avec $J = N$ et

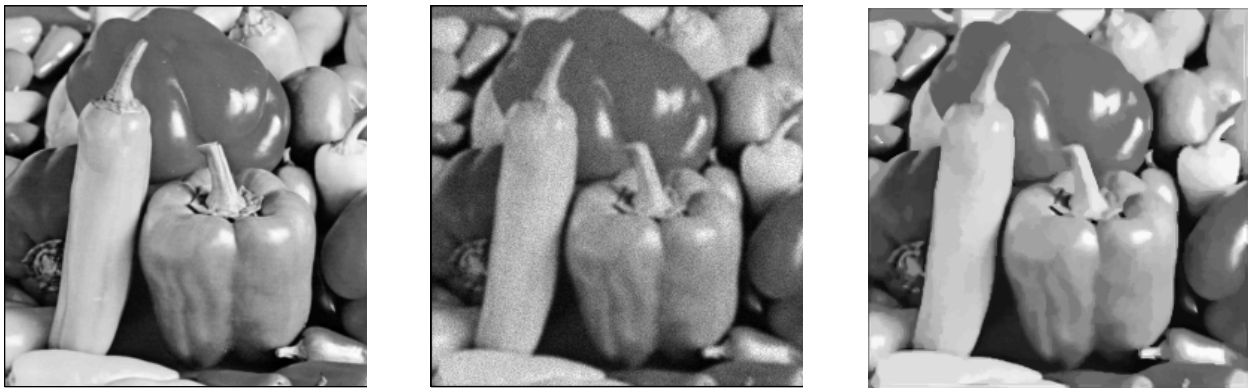
$$(\forall n \in \{1, \ldots, N\})(\forall \mathbf{x} \in \mathbb{R}^N) \quad \begin{cases} \mathsf{g}_{2,n}(\mathbf{x}) = \theta \ell_2(\mathbf{F}_n \mathbf{x}), \\ \mathbf{F}_n = [\omega_n^{(1)} \mathbf{F}_n^{(1)}, \ldots, \omega_n^{(J_n)} \mathbf{F}_n^{(J_n)}]^\top, \end{cases} \tag{4.46}$$

où $\theta \in]0, +\infty[$ est choisi de façon à maximiser le RSB entre $\overline{\mathbf{x}}$ et $\widehat{\mathbf{x}}$ défini par (2.20), $(\omega_n^{(j)})_{1 \leqslant j \leqslant J_n} \in \mathbb{R}^{J_n}$ sont des poids déterminés par un prétraitement [Bresson, 2009; Jezierska *et al.*, 2013], J_n détermine le nombre de voisins non locaux considérées au n-ème pixel, et ℓ_2 correspond à la norme euclidienne.

La figure 4.2(a) représente l'image originale et la figure 4.2(b) l'image dégradée correspondant au modèle décrit ci-dessus, dont le RSB est de 21.85 dB. Nous présentons en figure 4.2(c) l'image restaurée en utilisant la méthode proposée. Le RSB est égal à 27.11 dB.

Afin de montrer l'intérêt d'utiliser une métrique variable, nous proposons de comparer l'algorithme 3.1 avec sa version non préconditionnée donnée par l'algorithme 2.4 et une version visant à l'accélérer appelée FISTA (*Fast Iterative Shrinkage-Thresholding Algorithm*) développée par [Beck et Teboulle, 2009]. Notons que, bien que peu de résultats théoriques existent concernant la convergence de FISTA dans le cadre non convexe, nous n'avons pas observé de divergence de l'algorithme lors de nos simulations. Pour les algorithmes 3.1 et 2.4 nous avons choisi $\lambda_k \equiv 1$, et deux valeurs du pas $(\gamma_k)_{k \in \mathbb{N}}$ ont été testées : $\gamma_k \equiv 1$ et $\gamma_k \equiv 1.9$. Remarquons que pour ces valeurs de $(\lambda_k)_{k \in \mathbb{N}}$ et $(\gamma_k)_{k \in \mathbb{N}}$, l'hypothèse 3.3 est satisfaite. De plus, d'après la remarque 3.4(ii), l'hypothèse 3.4 est aussi vérifiée.

La figure 4.3 illustre les variations de $\big(\mathsf{f}(\mathbf{x}_k) - \mathsf{f}^*\big)_k$ et $\big(\|\mathbf{x}_k - \widehat{\mathbf{x}}\|\big)_k$ en fonction du temps de calcul, obtenues par les algorithmes 3.1, 2.4 et FISTA, pour des tests réalisés sur une machine Intel(R)



(a) (b) (c)

Figure 4.2 – ***Restauration :*** (a) *Image originale* Peppers. (b) *Image dégradée par un bruit gaussien dépendant du signal, RSB*=21.85 dB. (c) *Image reconstruite par la méthode proposée, RSB*=27.11 dB.

Xeon(R) CPU X5570 cadencé à 2.93GHz, implémenté à l'aide de Matlab 7. La solution optimale $\widehat{\mathbf{x}}$ a été calculée au préalable, pour chacun des algorithmes, avec un grand nombre d'itérations. D'autre part, f^* représente le minimum des (possibles) différentes valeurs $\mathsf{f}(\widehat{\mathbf{x}})$ obtenues pour chaque algorithme. Nous pouvons remarquer que notre méthode est la plus rapide dans cet exemple, autant pour la convergence du critère que pour la convergence des itérées.

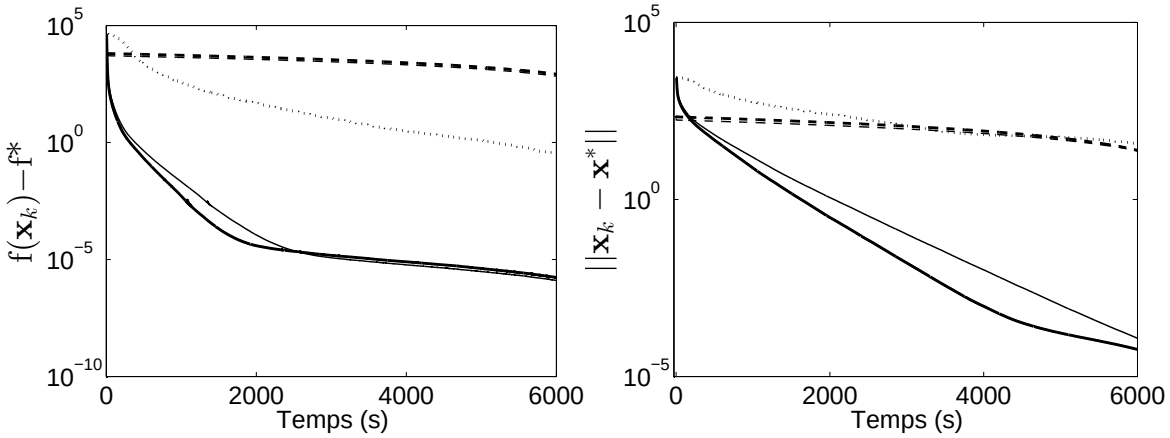


Figure 4.3 – ***Restauration :*** *Comparaison de la vitesse de convergence de l'algorithme 3.1 explicite-implicite à métrique variable avec* $\gamma_k \equiv 1$ *(lignes fines continues) et* $\gamma_k \equiv 1.9$ *(lignes épaisses continues), de l'algorithme 2.4 explicite-implicite avec* $\gamma_k \equiv 1$ *(lignes fines discontinues) et* $\gamma_k \equiv 1.9$ *(lignes épaisses discontinues), et de FISTA (lignes pointillés).*

4.4.2 Reconstruction d'image

Dans ce second scénario, $\overline{\mathbf{x}}$ correspond à une coupe du fantôme *Zubal* [Zubal *et al.*, 1994] de dimension 128×128, et $\mathbf{H}$ modélise $M = 16384$ projections parallèles à partir d'acquisitions de 128 lignes et 128 angles. Ce type de modèle se rencontre, par exemple, en tomographie par rayons X [Slaney et Kak, 1988]. Le vecteur $\mathbf{H}\overline{\mathbf{x}}$ est ensuite bruité par un bruit gaussien dépendant du signal d'après le modèle (4.11) où, pour tout $m \in \{1, \ldots, M\}$, $\alpha^{(m)} = 0.01$ et $\beta^{(m)} = 0.1$. Dans cette configuration, $\mathsf{x}_{\min} = 0$, $\mathsf{x}_{\max} = 1$, et la constante de Lipschitz de $\nabla \mathsf{h}$ est égale à $L = 6.0 \times 10^6$.

Nous utilisons une approche à l'analyse pour le terme de régularisation g_2 donné par (4.30) avec

$$(\forall j \in \{1, \ldots, j\}) \quad \begin{cases} \mathsf{g}_{2,j} = \theta^{(j)} |\cdot|, \\ \mathbf{F}_j = [\mathbf{W}\cdot]^{(j)}, \end{cases} \tag{4.47}$$

où $(\theta^{(j)})_{1 \leqslant j \leqslant J} \in]0, +\infty[^J$ et $\mathbf{W}$ correspond à une transformation redondante en ondelettes de Daubechies d'ordre 8 sur 3 niveaux de résolution. Cette décomposition correspond à une trame à l'analyse ajustée (*tight frame*) de constante égale à 64. Les paramètres $(\theta^{(j)})_{1 \leqslant j \leqslant J}$ sont choisis de façon à maximiser le RSB entre $\overline{\mathbf{x}}$ et $\widehat{\mathbf{x}}$.

La figure 4.4(a) représente l'image originale, (b) les projections bruités correspondantes, (c) l'image reconstruite par la méthode standard de rétro-projection filtrée [Slaney et Kak, 1988], et (d) l'image reconstruite par la méthode proposée.

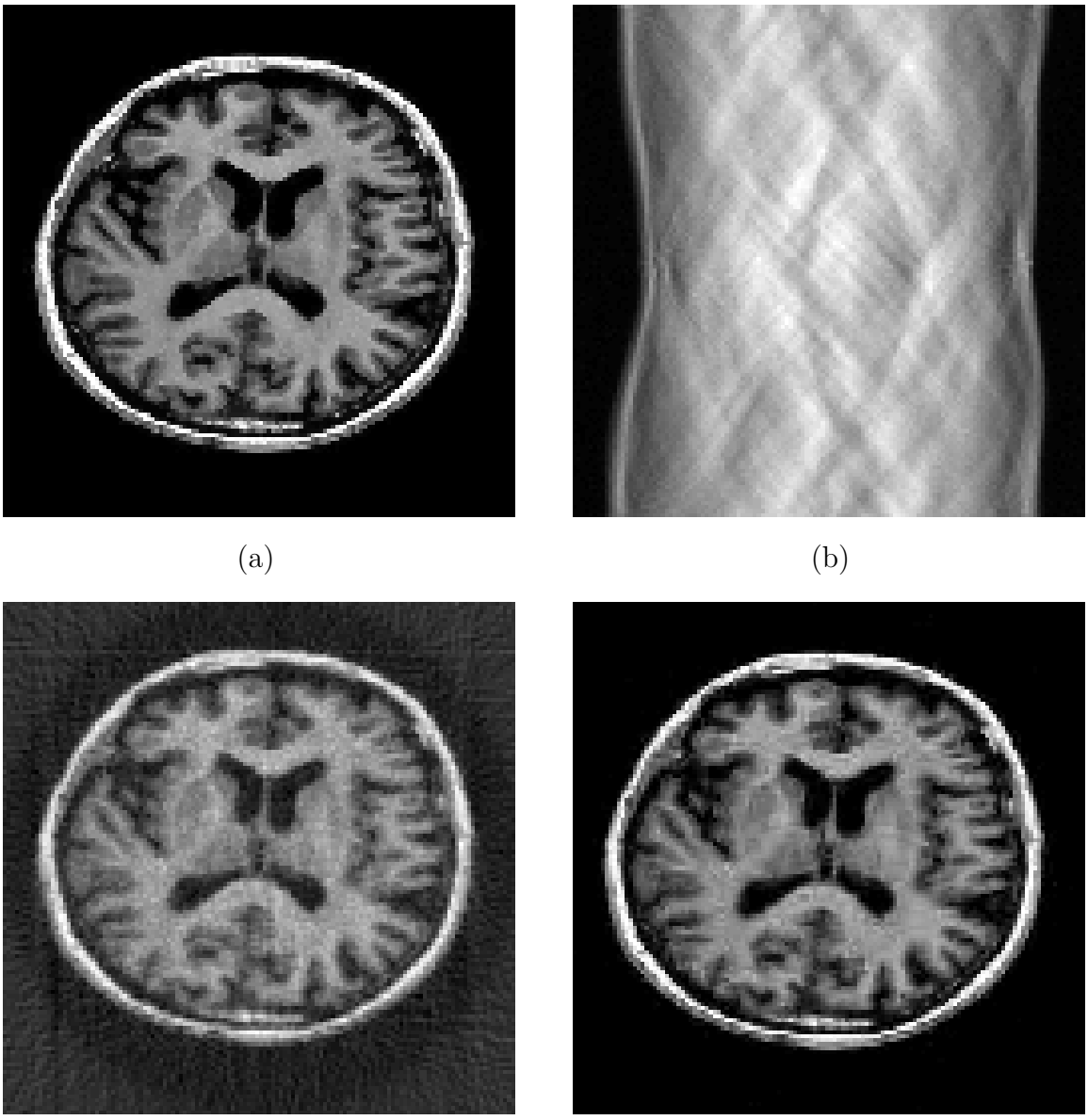
Comme dans l'exemple précédent, nous comparons la vitesse de convergence de la méthode que nous proposons, donnée par l'algorithme 3.1, avec sa version non préconditionnée, donnée dans l'algorithme 2.4, ainsi qu'avec FISTA. Nous choisissons le paramètre de relaxation λ_k égal à 1 pour tout $k \in \mathbb{N}$, et nous testons deux valeurs différentes pour le pas γ_k : $\gamma_k \equiv 1$ et $\gamma_k \equiv 1.9$.

La figure 4.5 illustre les variations de $\left(\mathsf{f}(\mathbf{x}_k) - \mathsf{f}^*\right)_k$ et $\left(\|\mathbf{x}_k - \widehat{\mathbf{x}}\|\right)_k$ en fonction du temps de calcul, en utilisant les trois algorithmes. Les solutions optimales $\widehat{\mathbf{x}}$ et f^* sont calculées de la même façon que dans la section précédente. Dans cet exemple, nous pouvons observer que FISTA et l'algorithme 2.4 ont des comportements similaires. Cependant, contrairement à l'exemple précédent, le choix du pas $(\gamma_k)_{k \in \mathbb{N}}$ influe de façon sensible sur la vitesse de convergence des algorithmes 3.1 et 2.4. Prendre $(\gamma_k)_{k \in \mathbb{N}}$ le plus proche possible de sa borne supérieure permet ici d'accélérer la convergence des critères et des itérées.

4.5 Conclusion

Dans ce chapitre nous nous sommes intéressés à la restauration d'une image dégradée par un bruit gaussien dépendant du signal. Pour cela nous avons étudié dans un premier temps les propriétés du terme d'attache aux données issu du MAP associé à ce modèle de bruit. En particulier, nous avons démontré la non-convexité et le caractère gradient Lipschitz de ce critère.

Dans un deuxième temps, nous avons traité deux exemples applicatifs en traitement des images. Le premier était une application en restauration et le second un problème de reconstruction en tomographie. Dans les deux cas, les images étaient dégradées par un opérateur linéaire puis bruitées par un bruit gaussien dépendant. Nous avons utilisé l'algorithme 3.1 explicite-implicite à métrique variable proposé dans le chapitre 3 pour trouver des solutions de ces problèmes inverses.



(a) (b)

(c) (d)

Figure 4.4 – ***Reconstruction :*** (a) *Image originale* Zubal. (b) *Image dégradée par un bruit gaussien dépendant du signal.* (c) *Image reconstruite par rétro-projection filtrée, RSB*=7 dB. (d) *Image reconstruite par la méthode proposée, RSB*=18.9 dB.

Des simulations ont montré que notre algorithme se compare très favorablement à l'algorithme explicite-implicite usuel et à ses formes accélérées telles que FISTA. En effet, nous avons vu que l'utilisation d'une métrique variable permet d'accélérer fortement la convergence du critère et des itérées.

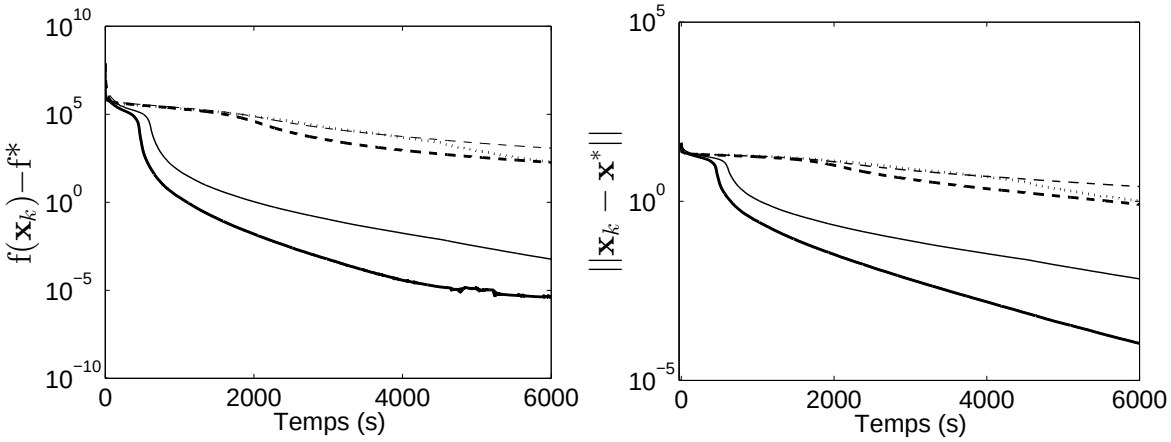


Figure 4.5 – ***Reconstruction :*** *Comparaison de la vitesse de convergence de l'algorithme 3.1 explicite-implicite à métrique variable avec $\gamma_k \equiv 1$ (lignes fines continues) et $\gamma_k \equiv 1.9$ (lignes épaisses continues), de l'algorithme 2.4 explicite-implicite avec $\gamma_k \equiv 1$ (lignes fines discontinues) et $\gamma_k \equiv 1.9$ (lignes épaisses discontinues), et de FISTA (lignes pointillés).*

Dans le chapitre 5, nous proposerons de combiner cette méthode d'accélération à une méthode de minimisation par bloc afin de réduire le coût de calcul par itération lorsque l'on traite des données de grande taille.

4.A Démonstration de la proposition 4.4

Soit $m \in \{1, \dots, M\}$. Rappelons que la constante de Lipschitz de $\dot{\chi}_m$ sur $[0, +\infty[$ est définie par

$$\mu^{(m)} = \sup_{\mathsf{y} \in [0, +\infty[} |\ddot{\chi}_m(\mathsf{y})|.$$

Or, d'après la proposition 4.3, la fonction χ_m est convexe sur $[0, \mathsf{y}_{\max}^{(m)}]$ et concave sur $]\mathsf{y}_{\max}^{(m)}, +\infty[$. Par conséquent, sa dérivée seconde vérifie $\ddot{\chi}_m(\mathsf{y}) \geqslant 0$ pour tout $\mathsf{y} \in [0, \mathsf{y}_{\max}^{(m)}]$, et $\ddot{\chi}_m(\mathsf{y}) \leqslant 0$ pour tout $\mathsf{y} \in]\mathsf{y}_{\max}^{(m)}, +\infty[$. Ainsi, on a

$$\mu^{(m)} = \max\{\mu_1^{(m)}, \mu_2^{(m)}\}, \tag{4.48}$$

où $\mu_1^{(m)}$ (resp. $\mu_2^{(m)}$) est la constante de Lipschitz de $\dot{\chi}_m$ sur $[0, \mathsf{y}_{\max}^{(m)}[$ (resp. sur $]\mathsf{y}_{\max}^{(m)}, +\infty[$) :

$$\begin{aligned}
\mu_1^{(m)} &= \sup_{\mathsf{y} \in [0, \mathsf{y}_{\max}^{(m)}[} \ddot{\chi}_m(\mathsf{y}) \\
\mu_2^{(m)} &= \sup_{\mathsf{y} \in]\mathsf{y}_{\max}^{(m)}, +\infty[} -\ddot{\chi}_m(\mathsf{y}) = - \inf_{\mathsf{y} \in]\mathsf{y}_{\max}^{(m)}, +\infty[} \ddot{\chi}_m(\mathsf{y}).
\end{aligned}$$

Nous allons donc étudier les variations de la fonction $\mathsf{y} \in [0,+\infty[\mapsto \ddot{\chi}_m(\mathsf{y})$. La dérivée de $\ddot{\chi}_m$ est donnée par

$$(\forall \mathsf{y} \in [0,+\infty[) \quad \dddot{\chi}_m(\mathsf{y}) = -3\alpha^{(m)} \frac{(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2}{(\alpha^{(m)}\mathsf{y} + \beta^{(m)})^4} + \left(\frac{\alpha^{(m)}}{\alpha^{(m)}\mathsf{y} + \beta^{(m)}}\right)^3.$$

On peut donc en déduire que la fonction $\mathsf{y} \in [0,+\infty[\mapsto \ddot{\chi}_m(\mathsf{y})$ est décroissante sur $\left[0, \theta^{(m)}\right]$, et croissante sur $\left[\theta^{(m)}, +\infty\right[$, avec

$$\theta^{(m)} = \frac{1}{\alpha^{(m)}} \left(3\Big(\frac{\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)}}{\alpha^{(m)}}\Big)^2 - \beta^{(m)}\right). \tag{4.49}$$

.

(i) Remarquons d'une part que

$$\begin{cases} \ddot{\chi}_m(0) = \dfrac{1}{(\beta^m)^2} \left(\dfrac{(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2}{\beta^{(m)}} - \dfrac{(\alpha^{(m)})^2}{2}\right) \\ \lim_{\mathsf{y}\to+\infty} \ddot{\chi}_m(\mathsf{y}) = 0. \end{cases}$$

Donc la borne supérieure de $\ddot{\chi}_m$ sur $[0,+\infty]$ (et donc sur $[0, \mathsf{y}^{(m)}_{\max}]$) est atteinte en 0, et on a alors

$$\mu_1^{(m)} = \ddot{\chi}_m(0) = \frac{1}{(\beta^m)^2} \left(\frac{(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2}{\beta^{(m)}} - \frac{(\alpha^{(m)})^2}{2}\right). \tag{4.50}$$

(ii) D'autre part, remarquons que $\theta^{(m)} \geqslant \mathsf{y}^{(m)}_{\max}$. Ainsi, la borne inférieure de $\ddot{\chi}_m$ est atteinte en $\theta^{(m)}$, et, d'après (4.22), on a alors

$$\begin{aligned} \mu_2^{(m)} &= -\ddot{\chi}_m(\theta^{(m)}) \\ &= -\frac{2(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2 - (\alpha^{(m)})^2(\alpha^{(m)}\theta^{(m)} + \beta^{(m)})}{2(\alpha^{(m)}\theta^{(m)} + \beta^{(m)})^3}. \end{aligned}$$

Or, d'après (4.49), on a

$$\alpha^{(m)}\theta^{(m)} + \beta^{(m)} = 3\Big(\frac{\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)}}{\alpha^{(m)}}\Big)^2.$$

En combinant les deux dernières équations, on obtient

$$\begin{aligned} \mu_2^{(m)} &= \frac{3(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2 - 2(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^2}{2\Big(3(\frac{\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)}}{\alpha^{(m)}})^2\Big)^3} \\ &= \frac{1}{2} \frac{(\alpha^{(m)})^6}{27(\alpha^{(m)}\mathsf{z}^{(m)} + \beta^{(m)})^4}. \end{aligned}$$

Chapitre 5

Algorithme explicite-implicite à métrique variable alterné par bloc

Sommaire

5.1 Introduction

Dans ce chapitre, nous considérons un problème d'optimisation consistant à minimiser un critère $\mathsf{f}\colon \mathbb{R}^N \to]-\infty,+\infty]$ qui peut s'écrire comme la somme d'une fonction h différentiable de gradient Lipschitz et d'une fonction g séparable par bloc et non nécessairement lisse (voir la section 2.3.3.2).

L'hypothèse de séparabilité suggère l'utilisation d'une stratégie d'optimisation alternée, où à chaque itération, un seul bloc est mis à jour tandis que les autres restent inchangés. Ce type de stratégie permet de réduire le coût de calcul par itération. La plus simple d'entre elles est l'algorithme de minimisation alternée par bloc (*Block Coordinate Descent* (BCD) algorithm) qui minimise, à chaque itération, de façon exacte la fonction f par rapport à un bloc donné [Bertsekas, 1999; Luenberger, 1973; Ortega et Rheinboldt, 1970; Zangwill, 1969]. Une autre méthode pouvant être utilisée est la version proximale de l'algorithme BCD, introduite par [Auslender, 1992], qui, à chaque itération, effectue une étape proximale sur la fonction f relativement à un bloc donné. Cependant, aucune de ces deux méthode ne permet d'exploiter la propriété de gradient Lipschitz de h. Une méthode pouvant en tirer parti est l'algorithme explicite-implicite alterné par bloc (*Block Coordinate Forward-Backward* (BC-FB) algorithm [Luo et Tseng, 1992a,b]). Notons que la convergence dans le cas où f n'est pas nécessairement convexe n'a été démontrée que pour l'algorithme proximal alterné par bloc et pour l'algorithme BC-FB (voir [Attouch *et al.*, 2010a, 2011] et [Bolte *et al.*, 2013; Xu et Yin, 2013] lorsque f satisfait l'inégalité de KL).

Dans ce chapitre nous nous intéressons plus particulièrement au troisième algorithme. Nous avons vu dans le chapitre 4 que l'utilisation d'une métrique variable permet d'accélérer la vitesse de convergence de l'algorithme explicite-implicite. Nous nous proposons d'étendre cette technique d'accélération au cas de l'algorithme BC-FB. Nous analyserons la convergence de cette méthode (ainsi que de sa version inexacte) dans le cas où la fonction f n'est pas nécessairement convexe.

Le reste du chapitre est organisé comme suit : nous formalisons le problème étudié et introduisons nos notations dans la section 5.2. Puis nous dressons, dans la section 5.3, un état de l'art des méthodes alternées par bloc afin de situer notre travail. Dans la section 5.4 nous proposons des versions exacte et inexacte de l'algorithme explicite-implicite à métrique variable alterné par bloc. Nous étudions leurs propriétés de convergence et de vitesse de convergence dans la section 5.5. Une conclusion sera fournie dans la section 5.6. Dans le chapitre 6 nous donnerons des exemples applicatifs de cet algorithme afin d'en illustrer son intérêt pratique.

5.2 Problème de minimisation

Dans ce chapitre, nous nous intéressons au problème d'optimisation suivant :

$$\text{trouver} \quad \widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x}\in\mathbb{R}^N} \ (\mathsf{f}(\mathbf{x}) = \mathsf{h}(\mathbf{x}) + \mathsf{g}(\mathbf{x})), \tag{5.1}$$

où $\mathsf{f} : \mathbb{R}^N \to]-\infty,+\infty]$ est une fonction coercive, h est une fonction différentiable et g est une fonction propre, semi-continue inférieurement et séparable par bloc. De façon formelle, on notera $(\mathbb{J}_j)_{1\leqslant j\leqslant J}$ une partition de $\{1,\dots,N\}$ formée de $J \geqslant 2$ sous-ensembles, et, pour tout

$j \in \{1, \dots, J\}$, $N_j \neq 0$ désignera le cardinal de $\mathbb{J}_j$. Tout vecteur $\mathbf{x} = (\mathsf{x}^{(n)})_{1 \leqslant n \leqslant N} \in \mathbb{R}^N$ peut alors se décomposer par blocs :

$$\mathbf{x} = \left(\mathbf{x}^{(j)}\right)_{1 \leqslant j \leqslant J} \in \mathbb{R}^{N_1} \times \ldots \times \mathbb{R}^{N_J}, \tag{5.2}$$

où, pour tout $j \in \{1, \dots, J\}$, $\mathbf{x}^{(j)} = \left(\mathsf{x}^{(n)}\right)_{n \in \mathbb{J}_j} \in \mathbb{R}^{N_j}$ et l'hypothèse de séparabilité sur g s'écrit

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}(\mathbf{x}) := \sum_{j=1}^{J} \mathsf{g}_j(\mathbf{x}^{(j)}), \tag{5.3}$$

où, pour tout $j \in \{1, \dots, J\}$, $\mathsf{g}_j \colon \mathbb{R}^{N_j} \to]-\infty, +\infty]$.

Dans le reste de ce chapitre, nous utiliserons les notations suivantes. Pour tout $j \in \{1, \dots, J\}$, on notera $\overline{\jmath}$ l'ensemble complémentaire de $\{j\}$ dans $\{1, \dots, J\}$, i.e. $\overline{\jmath} := \{1, \dots, J\} \setminus \{j\}$. Ainsi, pour tout $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{x}^{(\overline{\jmath})} := \left(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(j-1)}, \mathbf{x}^{(j+1)}, \dots, \mathbf{x}^{(J)}\right)$. De plus, pour tout $\mathbf{x}^{(\overline{\jmath})} \in \bigtimes_{i \in \overline{\jmath}} \mathbb{R}^{N_i}$, la fonction $\mathsf{h}_j(\cdot, \mathbf{x}^{(\overline{\jmath})}) \colon \mathbb{R}^{N_j} \to \mathbb{R}$ désignera la fonction h restreinte au j-ème bloc, en $\mathbf{x}^{(\overline{\jmath})}$, définie par

$$(\forall \mathbf{y} \in \mathbb{R}^{N_j}) \quad \mathsf{h}_j(\mathbf{y}, \mathbf{x}^{(\overline{\jmath})}) := \mathsf{h}(\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(j-1)}, \mathbf{y}, \mathbf{x}^{(j+1)}, \dots, \mathbf{x}^{(J)}). \tag{5.4}$$

Dans la suite, pour tout $\mathbf{x} \in \mathbb{R}^N$ et $j \in \{1, \dots, J\}$, on notera $\nabla_j \mathsf{h}(\mathbf{x}) \in \mathbb{R}^{N_j}$ le gradient partiel de h par rapport à $\mathbf{x}^{(j)}$ calculé en $\mathbf{x}$.

Le principe des méthodes alternées par blocs réside en la mise à jour de façon alternée au cours des itérations de chacun des blocs $(\mathbf{x}^{(j)})_{1 \leqslant j \leqslant J}$. Plus précisément, à chaque itération $k \in \mathbb{N}$ de l'algorithme, un indice $j_k \in \{1, \dots, J\}$ est sélectionné : le bloc $\mathbf{x}^{(j_k)}$ est alors mis à jour, tandis que tous les autres blocs $\mathbf{x}^{(\overline{\jmath}_k)}$ restent inchangés. Les principales règles de sélection des blocs sont données dans la définition suivante :

Définition 5.1.
Soit $(j_k)_{k \in \mathbb{N}}$ la suite d'indices des blocs mis à jour au cours des itérations $k \in \mathbb{N}$.

(i) *Les blocs sont mis à jour selon une* règle cyclique *si*

$$(\forall k \in \mathbb{N}) \quad j_k - 1 = k \mod (J). \tag{5.5}$$

(ii) *Les blocs sont mis à jour selon une* règle essentiellement cyclique *s'il existe une constante $K \geqslant J$ telle que,*

$$(\forall k \in \mathbb{N}) \quad \{1, \dots, j\} \subset \{j_k, \dots, j_{k+K-1}\}. \tag{5.6}$$

(iii) *Les blocs sont mis à jour selon une* règle aléatoire uniforme *si, pour tout $k \in \mathbb{N}$, j_k est une réalisation d'une variable aléatoire suivant une loi uniforme sur $\{1, \dots, J\}$.*

La *règle quasi-cyclique* donnée par la définition 5.1(ii) a été introduite par [Luo et Tseng, 1992a] et utilisée sous le nom de *règle essentiellement cyclique* dans [Tseng, 2001] pour la mise à jour des blocs de l'algorithme BC-FB. Elle peut être interprétée de la façon suivante : les blocs peuvent être mis à jour dans un ordre arbitraire à partir du moment où ils sont tous mis à jour au moins une fois sur un nombre fini d'itérations. En particulier, sous l'hypothèse de mise à jour essentiellement cyclique, les blocs peuvent être sélectionnés suivant un ordre arbitraire

changeant toutes les K itérations. Ainsi, on comprend que la règle cyclique donnée par (i) est un cas particulier de la règle (ii).

En outre, pour toute itération $k \in \mathbb{N}$ et pour tout indice de bloc $j \in \{1, \dots, J\}$, on notera

$$l_{k,j} = \min \{l \in \{0, \dots, K-1\} \mid j_{k+l} = j\}, \tag{5.7}$$

la première fois où le j-ème bloc est mis à jour après la k-ème itération. Par exemple, lorsque la règle cyclique est utilisée, on a $l_{k,j} = j \mod (J)$. Nous allons de plus introduire la permutation $\sigma_k \colon \{1, \dots, J\} \to \{1, \dots, J\}$ qui assure que la suite $(l_{k,\sigma_k(i)})_{1 \leqslant i \leqslant J}$ est croissante. Ces deux notations seront particulièrement utiles pour nous permettre de manipuler la règle de mise à jour essentiellement cyclique dans notre analyse de convergence.

5.3 État de l'art

Nous résumons dans cette section les principales méthodes alternées existants dans la littérature afin de pouvoir situer notre travail et souligner ses avantages par rapport à celles-ci.

5.3.1 Algorithme de minimisation alternée par bloc

Une approche standard pour résoudre le problème (5.1) est d'utiliser l'algorithme de minimisation alternée par bloc donné par l'algorithme 5.1.

Algorithme 5.1 Algorithme de minimisation alternée par bloc

Initialisation : Soit $\mathbf{x}_0 \in \mathbb{R}^N$.

Itérations :

Pour $k = 0, 1, \dots$

$$\begin{aligned} &\text{soit } j_k \in \{1, \dots, J\}, \\ &\mathbf{x}_{k+1}^{(j_k)} \in \operatorname*{Argmin}_{\mathbf{y} \in \mathbb{R}^{N_{j_k}}} \left(\mathsf{h}_{j_k}(\mathbf{y}, \mathbf{x}_k^{(\overline{\jmath}_k)}) + \mathsf{g}_{j_k}(\mathbf{y}) \right), \\ &\mathbf{x}_{k+1}^{(\overline{\jmath}_k)} = \mathbf{x}_k^{(\overline{\jmath}_k)}. \end{aligned}$$

Cette méthode peut aussi être vue comme une généralisation de l'algorithme de Gauss-Seidel permettant de résoudre les systèmes linéaires [Golub et Van Loan, 1996]. Elle est d'ailleurs désignée dans certains ouvrages comme *algorithme non-linéaire de Gauss-Seidel* (*nonlinear Gauss-Seidel method* [Bertsekas, 1999, Chap.2], [Ortega et Rheinboldt, 1970, Chap.7]). Dans la plupart des ouvrages de référence tels que [Bertsekas, 1999; Luenberger, 1973; Ortega et Rheinboldt, 1970; Zangwill, 1969], la mise à jour des blocs dans l'algorithme 5.1 se fait selon une *règle cyclique* (c.f. définition 5.1(i)). Cependant, l'analyse de convergence effectuée dans [Tseng, 2001]suppose seulement que les blocs sont mis à jour selon une *règle essentiellement cyclique* (c.f. définition 5.1(ii)). Les résultats de convergence présentés dans [Tseng, 2001] semblent de plus être les plus généraux concernant cet algorithme. Ils assurent la convergence de la suite $(\mathbf{x}_k)_{k \in \mathbb{N}}$ dès lors que

- la fonction f est quasi-convexe et hemivariée [1] en chaque bloc,
- soit la fonction f est pseudo-convexe [2] sur chaque paire de blocs, soit elle admet au moins un minimum par rapport à chacun des blocs.

Cependant, comme il est souligné dans [Tseng, 2001], la deuxième hypothèse peut être contraignante au sens où l'algorithme 5.1 risque de ne pas converger si l'on suppose uniquement la convexité de f par rapport à chaque bloc (c.f. [Powell, 1973] pour une illustration).

5.3.2 Algorithme proximal alterné par bloc

Pour pallier les problèmes de convergence rencontrés avec l'algorithme 5.1, une version proximale de ce dernier a été introduite par [Auslender, 1992]. Elle est définie par l'algorithme 5.2.

Algorithme 5.2 Algorithme proximal alterné par bloc

Initialisation : Soit $\mathbf{x}_0 \in \mathbb{R}^N$. Soient, pour tout $k \in \mathbb{N}$, $\gamma_k \in]0, +\infty[$ et $\mathbf{A}_{j_k}(\mathbf{x}_k) \in \mathcal{S}^+_{N_{j_k}}$.

Itérations :

Pour $k = 0, 1, \ldots$

$\quad$ soit $j_k \in \{1, \ldots, J\}$,

$\quad \mathbf{x}_{k+1}^{(j_k)} \in \operatorname{prox}_{\gamma_k^{-1}\mathbf{A}_{j_k}(\mathbf{x}_k),\mathsf{h}_{j_k}(\cdot,\mathbf{x}_k^{(\overline{\jmath}_k)})+\mathsf{g}_{j_k}} \left(\mathbf{x}_k^{(j_k)}\right)$,

$\quad \mathbf{x}_{k+1}^{(\overline{\jmath}_k)} = \mathbf{x}_k^{(\overline{\jmath}_k)}$.

La convergence de la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ générée par l'algorithme 5.2 vers une solution du problème (5.1) a été étudiée dans [Auslender, 1992] dans le cas où :

- la fonction h est une fonction différentiable, de gradient Lipschitz,
- les fonctions $(\mathsf{g}_j)_{1\leqslant j\leqslant J}$ sont convexes, propres, semi-continues inférieurement,
- pour tout $\ell \in \mathbb{N}$, $\mathbf{A}_{j_k}(\mathbf{x}_k) = \mathbf{I}_{N_{j_\ell}}$.

Récemment, la convergence vers un point critique de f, des itérées de l'algorithme 5.2 a été établie dans [Attouch *et al.*, 2010a] dans le cas où les fonctions h et $(\mathsf{g}_j)_{1\leqslant j\leqslant J}$ ne sont pas nécessairement convexes et, pour tout $k \in \mathbb{N}$, $\mathbf{A}_{j_k}(\mathbf{x}_k) = \mathbf{I}_{N_{j_k}}$. Ces résultats ont été généralisés dans [Attouch *et al.*, 2011] pour des matrices $(\mathbf{A}_{j_k}(\mathbf{x}_k))_{k\in\mathbb{N}}$ SDP quelconques. Notons que les études de convergences faites dans [Attouch *et al.*, 2010a, 2011] reposent sur l'hypothèse que la fonction f satisfait l'inégalité de Kurdyka-Łojasiewicz. Dans les trois études citées ci-dessus, la mise à jour des indices de blocs $(j_\ell)_{\ell\in\mathbb{N}}$ se fait selon la règle cyclique.

Notons que la convergence de l'algorithme 5.2 a été étudiée dans [Bauschke *et al.*, 2006] dans le cadre des opérateurs de projection de Bregman, dans le cas où $J = 2$, h est la distance de Bregman et g_1, g_2 sont des fonctions convexes. D'autre part, nous pouvons remarquer que

1. Une fonction $\mathsf{f}\colon D \subset \mathbb{R}^N \to \mathbb{R}$ est *hemivariée* sur un sous-ensemble D_0 de D si elle n'est constante sur aucun segment de D_0, i.e. il n'existe pas de couple $(\mathbf{x}, \mathbf{y}) \in D_0^2$ tel que $\mathbf{x} \neq \mathbf{y}$, et, pour tout $\mathsf{t} \in [0, 1]$, $(1-\mathsf{t})\mathbf{x} + \mathsf{t}\mathbf{y} \in D_0$ et $\mathsf{f}\big((1-\mathsf{t})\mathbf{x} + \mathsf{t}\mathbf{y}\big) = \mathsf{f}(\mathbf{x})$ [Ortega et Rheinboldt, 1970, Déf. 14.1.1].

2. Une fonction différentiable $\mathsf{f}\colon \mathbb{R}^N \to \mathbb{R}$ est *pseudo-convexe* si, pour tout $(\mathbf{x}, \mathbf{y}) \in (\mathbb{R}^N)^2$ tel que $\langle \mathbf{x} - \mathbf{y}, \nabla \mathsf{f}(\mathbf{x})\rangle \geqslant 0$, on a $\mathsf{f}(\mathbf{y}) \geqslant \mathsf{f}(\mathbf{x})$ [Mangasarian, 1969, Chap. 9.3]. Cette notion peut-être généralisée au cas où la fonction f n'est pas différentiable en considérant les éléments de son sous-différentiel [Hadjisavvas et Schaible, 2009].

le célèbre algorithme POCS (*Projection Onto Convex Sets* - algorithme de projection sur des ensembles convexes) introduit par [Bregman, 1965] peut être vu comme un cas particulier de l'algorithme 5.2 lorsque $\mathsf{h} \equiv 0$ et, pour tout $j \in \{1, \ldots, J\}$, g_j est une fonction indicatrice sur un ensemble convexe.

Une des difficultés majeures rencontrées lors de la mise en application de l'algorithme 5.2 est le calcul, à chaque itération $k \in \mathbb{N}$, de l'opérateur proximal de la fonction $\mathsf{h}_{j_k}(\cdot, \mathbf{x}_k^{(\overline{\jmath}_k)}) + \mathsf{g}_{j_k}$. En effet, même dans le cas où les opérateurs proximaux des fonctions $\mathsf{h}_{j_k}(\cdot, \mathbf{x}_k^{(\overline{\jmath}_k)})$ et g_{j_k} ont des formes simples, rien ne permet d'affirmer que l'opérateur proximal de $\mathsf{h}_{j_k}(\cdot, \mathbf{x}_k^{(\overline{\jmath}_k)}) + \mathsf{g}_{j_k}$ aura une forme explicite. Ainsi, [Attouch *et al.*, 2011] ont proposé une forme inexacte de l'algorithme 5.2 afin de calculer de façon approchée l'opérateur proximal.

5.3.3 Algorithme explicite-implicite alterné par bloc

Une autre stratégie permettant de pallier la possible difficulté rencontrée lors du calcul des opérateurs proximaux dans l'algorithme 5.2 est de remplacer à chaque itération l'étape proximale par une étape explicite-implicite (c.f. algorithme 2.4). Les itérées obtenues sont décrites dans l'algorithme 5.3.

Algorithme 5.3 Algorithme explicite-implicite alterné par bloc

Initialisation : Soient $\mathbf{x}_0 \in \mathbb{R}^N$ et, pour tout $k \in \mathbb{N}$, $\gamma_k \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$\quad$ soit $j_k \in \{1, \ldots, J\}$,

$\quad \mathbf{x}_{k+1}^{(j_k)} \in \operatorname{prox}_{\gamma_k \mathsf{g}_{j_k}} \left(\mathbf{x}_k^{(j_k)} - \gamma_k \nabla_{j_k} \mathsf{h}(\mathbf{x}_k) \right)$,

$\quad \mathbf{x}_{k+1}^{(\overline{\jmath}_k)} = \mathbf{x}_k^{(\overline{\jmath}_k)}$.

Cette méthode a tout d'abord été proposée par [Censor et Lent, 1987] afin de minimiser la fonction entropique de Burg sous des contraintes linéaires. Elle a ensuite été généralisée dans [Luo et Tseng, 1992a,b] au cas où la fonction h est convexe différentiable de gradient Lipschitz et les indices $(j_k)_{k\in\mathbb{N}}$ sont mis à jour selon la règle essentiellement cyclique.

Récemment, une étude de convergence de l'algorithme 5.3 a été menée dans [Combettes et Pesquet, 2015] dans le cas où h est convexe de gradient Lipschitz, g est convexe, les blocs sont mis à jour suivant une règle aléatoire, et plusieurs blocs peuvent être mis à jours simultanément à chaque itération. La suite des itérées générée par l'algorithme 5.3 est alors une suite de vecteurs aléatoires, et les auteurs en ont démontré la convergence presque sûre vers un minimiseur de f.

Finalement, les propriétés de convergence de l'algorithme 5.3 ont été étudiées dans [Xu et Yin, 2013] lorsque h est une fonction différentiable de gradient Lipschitz, g est non nécessairement lisse, et les deux fonctions sont convexes par rapport à chaque bloc. Les auteurs ont démontré que les suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ générées par l'algorithme 5.3 convergent vers un point critique de f. Ces résultats de convergence ont été généralisés dans [Bolte *et al.*, 2013] dans le cas où h et g ne sont pas nécessairement convexes. Dans cet article, l'algorithme 5.3 est dénommé algorithme *Proximal Alternating Linearized Minimization* (PALM). Dans les deux travaux [Bolte *et al.*, 2013; Xu et

Yin, 2013], les démonstrations de convergence reposent sur l'hypothèse que f satisfait l'inégalité de KL, et que les blocs sont mis à jour selon la règle cyclique.

5.4 Méthode proposée

5.4.1 Algorithme explicite-implicite à métrique variable alterné par bloc

Comme nous l'avons vu dans le chapitre 3, l'introduction d'une métrique variant au cours des itérations permet d'accélérer la vitesse de convergence de l'algorithme explicite-implicite. Ainsi, nous proposons de faire de même pour l'algorithme 5.3. Les itérées de cette méthode sont données dans l'algorithme 5.4, et est appelé algorithme explicite-implicite à métrique variable alterné par bloc (*Block Coordinate Variable Metric Forward-Backward* (BC-VMFB) algorithm).

Algorithme 5.4 Algorithme explicite-implicite à métrique variable alterné par bloc

Initialisation : Soient $\mathbf{x}_0 \in \mathbb{R}^N$, et, pour tout $k \in \mathbb{N}$, $\gamma_k \in]0, +\infty[$ et $\mathbf{A}_{j_k}(\mathbf{x}_k) \in \mathcal{S}^+_{N_{j_k}}$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} \text{soit } j_k \in \{1, \ldots, J\}, \\ \mathbf{x}_{k+1}^{(j_k)} \in \operatorname{prox}_{\gamma_k^{-1}\mathbf{A}_{j_k}(\mathbf{x}_k), \mathsf{g}_{j_k}} \left(\mathbf{x}_k^{(j_k)} - \gamma_k \left(\mathbf{A}_{j_k}(\mathbf{x}_k)\right)^{-1} \nabla_{j_k} \mathsf{h}(\mathbf{x}_k) \right), \\ \mathbf{x}_{k+1}^{(\overline{\jmath}_k)} = \mathbf{x}_k^{(\overline{\jmath}_k)}, \end{array} \right.$$

L'algorithme 5.4 (ainsi qu'une version inexacte de ce dernier) a été étudié dans [Richtárik et Talác, 2014] (resp. [Tappenden *et al.*, 2013]) lorsque les blocs sont mis à jour suivant une règle aléatoire uniforme (donnée dans la définition 5.1(iii)). Les auteurs ont établi la convergence de la suite $(\mathsf{f}(\mathbf{x}_k))_{k\in\mathbb{N}}$ lorsque les fonctions h et g_j sont convexes au sens où, pour tout $\delta \geqslant 0$ et $\epsilon \geqslant 0$, il existe $k_0 \in \mathbb{N}$ tel que la probabilité d'avoir $\mathsf{f}(\mathbf{x}_{k_0}) - \mathsf{f}(\widehat{\mathbf{x}}) \leqslant \epsilon$ est supérieure à $1 - \delta$.

Par ailleurs, comme il l'est indiqué dans [Razaviyayn *et al.*, 2013], lorsque les matrices de préconditionnement $(\mathbf{A}_{j_k}(\mathbf{x}_k))_{\ell\in\mathbb{N}}$ sont choisies judicieusement, l'algorithme 5.4 peut être vu comme un algorithme MM alterné par bloc. Ce type d'approche a été proposée dans [Fessler, 1997; Saquib *et al.*, 1995; Sotthivirat et Fessler, 2002] dans un contexte de reconstruction d'image. Ainsi, certaines propriétés de convergence de l'algorithme 5.4 peuvent être directement déduites de celles énoncées dans [Jacobson et Fessler, 2007] dans le cas où les fonctions g_j sont des fonctions indicatrices de sous-ensembles convexes fermés de $\mathbb{R}^{N_j}$, et de celles énoncées dans [Razaviyayn *et al.*, 2013] lorsque les fonctions g_j sont non lisses et convexes. Cependant, notons que dans [Jacobson et Fessler, 2007; Razaviyayn *et al.*, 2013], la convergence des itérées $(\mathbf{x}_k)_{k\in\mathbb{N}}$ vers une solution du problème (5.1) est démontrée seulement sous des hypothèses spécifiques. En particulier, il faut que la solution du problème (5.1) soit unique, et que chaque fonction $\mathsf{h}_j(\cdot, \mathbf{x}^{(\overline{\jmath})}) + \mathsf{g}_j$ n'admette qu'un unique minimiseur, pour $\mathbf{x}^{(\overline{\jmath})}$ fixé.

Remarque 5.1. *Lorsque $J = 1$ et $N_1 = N$, si $\mathsf{g} = \mathsf{g}_1$ est convexe, on retrouve l'algorithme 3.1 étudié dans le chapitre 3 dans le cas où il n'y a pas d'étape de relaxation dans ce dernier (i.e., pour tout $k \in \mathbb{N}$ dans l'algorithme 3.1, $\lambda_k = 1$).*

5.4.2 Hypothèses

Dans le reste de ce chapitre, nous nous intéresserons aux fonctions f, h et g satisfaisant les hypothèses suivantes :

Hypothèse 5.1.

(i) *Pour tout $j \in \{1, \dots, J\}$, $\mathsf{g}_j \colon \mathbb{R}^{N_j} \to]-\infty, +\infty]$ est une fonction propre, semi-continue inférieurement, bornée inférieurement par une fonction affine et sa restriction à son domaine de définition est continue.*

(ii) *La fonction $\mathsf{h} \colon \mathbb{R}^N \to \mathbb{R}$ est différentiable, et h est de gradient β-Lipschitzien sur $\operatorname{dom} \mathsf{g}$, avec $\beta > 0$.*

(iii) *La fonction f est coercive.*

La remarque suivante sera utile dans la suite du chapitre, en particulier pour l'étude de convergence de l'algorithme 5.4.

Remarque 5.2.

(i) *L'hypothèse 5.1(ii) est moins contraignante que l'hypothèse de Lipschitz différentiabilité de h utilisée en général pour démontrer la convergence de l'algorithme explicite-implicite* [Attouch *et al.*, 2011; Combettes et Wajs, 2005]. *En particulier, si $\operatorname{dom} \mathsf{g}$ est compact et h est deux fois continument différentiable, l'hypothèse 5.1(ii) est alors satisfaite.*

(ii) *D'après l'hypothèse 5.1(ii), $\operatorname{dom} \mathsf{g} \subset \operatorname{dom} \mathsf{h} = \mathbb{R}^N$. Ainsi, en conséquence de l'hypothèse 5.1(i), $\operatorname{dom} \mathsf{f} = \operatorname{dom} \mathsf{g}$ est non vide.*

(iii) *Si l'hypothèse 5.1 est satisfaite, alors f est une fonction propre, semi-continue inférieurement, et sa restriction à son domaine de définition est continue. Ainsi, la fonction f étant coercive, pour tout $\mathbf{x} \in \operatorname{dom} \mathsf{g}$, $\operatorname{lev}_{\leqslant \mathsf{f}(\mathbf{x})} \mathsf{f}$ est un ensemble compact. De plus, l'ensemble des minimiseurs de f est non vide et compact.*

(iv) *Si, pour tout $j \in \{1, \dots, J\}$, la fonction g_j est propre, semi-continue inférieurement et convexe, alors elle est bornée inférieurement par une fonction affine.*

Comme nous l'avons montré dans les chapitres 3 et 4, les matrices de préconditionnement $(\mathbf{A}_{j_k}(\mathbf{x}_k))_{k \in \mathbb{N}}$, induisant une métrique variable, jouent un rôle central dans l'algorithme 5.4. Nous proposons de les choisir suivant une stratégie MM. Plus précisément, soit $k \in \mathbb{N}$ une itération de l'algorithme 5.4, $\mathbf{x}_k \in \operatorname{dom} \mathsf{g}$ l'itérée associée, et $j_k \in \{1, \dots, J\}$ l'indice du bloc sélectionné. La matrice $\mathbf{A}_{j_k}(\mathbf{x}_k) \in \mathcal{S}^+_{N_{j_k}}$ sera alors choisie de façon à satisfaire la condition de majoration suivante :

Hypothèse 5.2.

(i) *La fonction quadratique définie par*

$$(\forall \mathbf{y} \in \mathbb{R}^{N_{j_k}}) \quad \mathsf{q}_{j_k}(\mathbf{y}, \mathbf{x}_k) := \mathsf{h}(\mathbf{x}_k) + \left\langle \mathbf{y} - \mathbf{x}_k^{(j_k)}, \nabla_{j_k} \mathsf{h}(\mathbf{x}_k) \right\rangle + \frac{1}{2} \left\langle \mathbf{y} - \mathbf{x}_k^{(j_k)}, \mathbf{A}_{j_k}(\mathbf{x}_k)(\mathbf{y} - \mathbf{x}_k^{(j_k)}) \right\rangle,$$

est une fonction majorante *de* $\mathsf{h}_{j_k}(\cdot, \mathbf{x}_k^{(\overline{\jmath}_k)})$ *en* $\mathbf{x}_k^{(j_k)}$ *sur* $\operatorname{dom} \mathsf{g}_{j_\ell}$, *i.e.*,

$$(\forall \mathbf{y} \in \operatorname{dom} \mathsf{g}_{j_k}) \quad \mathsf{h}_{j_k}(\mathbf{y}, \mathbf{x}_k^{(\overline{\jmath}_k)}) \leqslant \mathsf{q}_{j_k}(\mathbf{y}, \mathbf{x}_k).$$

(ii) *Il existe* $(\underline{\nu}, \overline{\nu}) \in]0, +\infty[^2$ *tel que*

$$(\forall \ell \in \mathbb{N}) \quad \underline{\nu}\, \mathbf{I}_{N_{j_\ell}} \preccurlyeq \mathbf{A}_{j_k}(\mathbf{x}_k) \preccurlyeq \overline{\nu}\, \mathbf{I}_{N_{j_\ell}}.$$

Remarque 5.3.

(i) *Remarquons qu'il n'est pas nécessaire de construire des majorantes quadratiques de* $\mathsf{h}_j(\cdot, \mathbf{x}^{(\overline{\jmath})})$ *sur* $\operatorname{dom} \mathsf{g}_j$ *pour tout* $j \in \{1, \ldots, J\}$ *et pour tout* $\mathbf{x}^{(\overline{\jmath})} \in \bigtimes_{i \in \overline{\jmath}} \operatorname{dom} \mathsf{g}_i$. *Il suffit que la majorante soit construite aux points* $(\mathbf{x}_k)_{k \in \mathbb{N}}$.

(ii) *Supposons que, pour tout* $\mathbf{x}' \in \operatorname{dom} \mathsf{g}$, *une fonction majorante quadratique de* h *sur* $\operatorname{dom} \mathsf{g}$ *soit donnée par*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}(\mathbf{x}, \mathbf{x}') := \mathsf{h}(\mathbf{x}') + \langle \mathbf{x} - \mathbf{x}', \nabla \mathsf{h}(\mathbf{x}') \rangle + \frac{1}{2} \langle \mathbf{x} - \mathbf{x}', \mathbf{B}(\mathbf{x}')(\mathbf{x} - \mathbf{x}') \rangle, \tag{5.8}$$

où $\mathbf{B}(\mathbf{x}') = \left(\mathsf{B}(\mathbf{x}_k)^{(n,n')}\right)_{(n,n') \in \{1,\ldots,N\}^2} \in \mathcal{S}_N^+$. *L'hypothèse 5.2(i) est alors satisfaite lorsque* $\mathbf{A}_{j_k}(\mathbf{x}_k) = \left(\mathsf{B}(\mathbf{x}_k)^{(n,n')}\right)_{(n,n') \in \mathbb{J}_{j_k}^2}$. *De plus, s'il existe* $(\underline{\nu}, \overline{\nu}) \in]0, +\infty[^2$ *tel que, pour tout* $\mathbf{x}' \in \operatorname{dom} \mathsf{g}$, $\underline{\nu} \mathbf{I}_N \preccurlyeq \mathbf{B}(\mathbf{x}') \preccurlyeq \overline{\nu} \mathbf{I}_N$, *alors l'hypothèse 5.2(ii) est aussi satisfaite.*

(iii) *D'après la lemme 2.1, si* $\operatorname{dom} \mathsf{g}$ *est convexe, l'existence de la fonction majorante* (5.8) *est assurée lorsque* h *satisfait l'hypothèse 5.1(ii).*

D'autre part, afin d'assurer que chaque bloc soit mis à jour une infinité de fois au cours des itérations de l'algorithme 5.4, nous allons considérer la règle de mise à jour des blocs suivante [Luo et Tseng, 1992b; Tseng, 2001] :

Hypothèse 5.3.
Les indices $(j_k)_{k \in \mathbb{N}}$ *des blocs sont mis à jour selon la règle essentiellement cyclique donnée dans la définition 5.1.*

Pour finir, nous supposerons que, pour tout $k \in \mathbb{N}$, le pas γ_k utilisé dans l'algorithme 5.4 satisfait l'hypothèse suivante :

Hypothèse 5.4.
L'une des assertions suivantes est vérifiée :

(i) *il existe* $(\underline{\gamma},\overline{\gamma}) \in]0,+\infty[^2$ *tel que, pour tout* $k \in \mathbb{N}$, $\underline{\gamma} \leqslant \gamma_k \leqslant 1-\overline{\gamma}$ *;*

(ii) *pour tout* $j \in \{1,\dots,J\}$, g_j *est une fonction convexe et il existe* $(\underline{\gamma},\overline{\gamma}) \in]0,+\infty[^2$ *tel que, pour tout* $k \in \mathbb{N}$, $\underline{\gamma} \leqslant \gamma_k \leqslant 2-\overline{\gamma}$.

5.4.3 Version inexacte de l'algorithme explicite-implicite à métrique variable alterné par bloc

De la même façon que dans le chapitre 3, nous proposons une version inexacte de l'algorithme 5.4 pour résoudre le problème (5.1) dans le cas où les opérateurs proximaux des fonctions $(\mathsf{g}_j)_{1\leqslant j\leqslant J}$ n'ont pas de forme explicite :

Algorithme 5.5 Algorithme inexact explicite-implicite à métrique variable alterné par bloc

Initialisation : Soient $\mathbf{x}_0 \in \mathbb{R}^N$, $\alpha_1 \in]1/2,+\infty[$, $\alpha_2 \in]0,+\infty[$. Soient, pour tout $k \in \mathbb{N}$, $\gamma_k \in]0,+\infty[$ et $\mathbf{A}_{j_k}(\mathbf{x}_k) \in \mathcal{S}^+_{N_{j_k}}$.

Itérations :

Pour $k = 0,1,\dots$

soit $j_k \in \{1,\dots,J\}$,

trouver $\mathbf{x}_{k+1}^{(j_k)} \in \mathbb{R}^{N_{j_k}}$ et $\mathbf{r}_{k+1}^{(j_k)} \in \partial \mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)})$ tels que

$$\mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)}) + \left\langle \mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}, \nabla_{j_k}\mathsf{h}(\mathbf{x}_k) \right\rangle + \alpha_1 \|\mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}\|^2_{\mathbf{A}_{j_k}(\mathbf{x}_k)} \leqslant \mathsf{g}_{j_k}(\mathbf{x}_k^{(j_k)}),$$

$$\|\nabla_{j_k}\mathsf{h}(\mathbf{x}_k) + \mathbf{r}_{k+1}^{(j_k)}\| \leqslant \alpha_2 \|\mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}\|_{\mathbf{A}_{j_k}(\mathbf{x}_k)},$$

$\mathbf{x}_{k+1}^{(\overline{j}_k)} = \mathbf{x}_k^{(\overline{j}_k)}$.

Dans l'algorithme 5.5, la première inégalité est une condition de décroissance suffisante et la seconde inégalité correspond à une condition d'optimalité.

La remarque suivante explique en quoi l'algorithme 5.5 peut être interprété comme une version inexacte de l'algorithme 5.3.

Remarque 5.4.
Soient $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *et* $(j_k)_{k\in\mathbb{N}}$ *des suites générées par l'algorithme* (5.3).

(i) *Dans un premier temps, supposons que l'hypothèse 5.4(i) soit satisfaite. D'une part, par définition de l'opérateur proximal, on a, pour tout* $k \in \mathbb{N}$,

$$\mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)}) + \left\langle \mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}, \nabla_{j_k}\mathsf{h}(\mathbf{x}_k) \right\rangle + \frac{\gamma_k^{-1}}{2} \|\mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}\|^2_{\mathbf{A}_{j_k}(\mathbf{x}_k)} \leqslant \mathsf{g}_{j_k}(\mathbf{x}_k^{(j_k)}),$$

ainsi, la condition de décroissance suffisante de l'algorithme 5.5 est satisfaite pour $\alpha_1 = (1-\overline{\gamma})^{-1}/2$ *(puisque* $\gamma_k^{-1} \geqslant (1-\overline{\gamma})^{-1} > 1$*). D'autre part, en considérant la caractérisation*

variationnelle de l'opérateur proximal, il existe $\mathbf{r}_{k+1}^{(j_k)} \in \partial \mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)})$ *tel que*

$$\mathbf{r}_{k+1}^{(j_k)} = -\nabla_{j_k} \mathsf{h}(\mathbf{x}_k) + \gamma_k^{-1} \mathbf{A}_{j_k}(\mathbf{x}_k)(\mathbf{x}_k^{(j_k)} - \mathbf{x}_{k+1}^{(j_k)}).$$

Sous les hypothèses 5.2(ii) et 5.4(i), on obtient alors

$$\|\mathbf{r}_{k+1}^{(j_k)} + \nabla_{j_k} \mathsf{h}(\mathbf{x}_k)\| = \gamma_k^{-1} \|\mathbf{A}_{j_k}(\mathbf{x}_k)(\mathbf{x}_k^{(j_k)} - \mathbf{x}_{k+1}^{(j_k)})\| \leqslant \underline{\gamma}^{-1} \sqrt{\overline{\nu}} \|\mathbf{x}_k^{(j_k)} - \mathbf{x}_{k+1}^{(j_k)}\|_{\mathbf{A}_{j_k}(\mathbf{x}_k)},$$

qui correspond à la condition d'optimalité donnée par la seconde inégalité de l'algorithme 5.5 pour $\alpha_2 = \underline{\gamma}^{-1} \sqrt{\overline{\nu}}$.

(ii) *Supposons maintenant que l'hypothèse 5.4(ii) soit satisfaite. Soit* $k \in \mathbb{N}$. *D'après la caractérisation variationnelle de l'opérateur proximal ainsi que la convexité de* g_{j_k}, *il existe* $\mathbf{r}_{k+1}^{(j_k)} \in \partial \mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)})$ *tel que*

$$\begin{cases} \mathbf{r}_{k+1}^{(j_k)} = -\nabla_{j_k} \mathsf{h}(\mathbf{x}_k) + \gamma_k^{-1} \mathbf{A}_{j_k}(\mathbf{x}_k)(\mathbf{x}_k^{(j_k)} - \mathbf{x}_{k+1}^{(j_k)}) \\ \left\langle \mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}, \mathbf{r}_{k+1}^{(j_k)} \right\rangle \geqslant \mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)}) - \mathsf{g}_{j_k}(\mathbf{x}_k^{(j_k)}), \end{cases}$$

ce qui implique

$$\mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)}) + \left\langle \mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}, \nabla_{j_k} \mathsf{h}(\mathbf{x}_k) \right\rangle + \gamma_k^{-1} \|\mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}\|_{\mathbf{A}_{j_k}(\mathbf{x}_k)}^2 \leqslant \mathsf{g}_{j_k}(\mathbf{x}_k^{(j_k)}),$$

ainsi, la condition suffisante de décroissance de l'algorithme 5.5 est satisfaite pour $\alpha_1 = (2 - \overline{\gamma})^{-1}$ *(puisque* $\gamma_k^{-1} \geqslant (2 - \overline{\gamma})^{-1} > 1/2$*). La condition d'optimalité inexacte est obtenue par un raisonnement similaire à celui de (i).*

5.5 Analyse de convergence

5.5.1 Propriétés de descente

Dans cette section, nous donnons des résultats techniques concernant le comportement asymptotique de la suite $\left(\mathsf{f}(\mathbf{x}_k)\right)_{k \in \mathbb{N}}$ générée par l'algorithme 5.5. Ces résultats préliminaires nous permettrons ensuite de démontrer la convergence de la méthode proposée.

Lemme 5.1. *Soit* $(\mathbf{x}_k)_{k \in \mathbb{N}}$ *une suite générée par l'algorithme 5.5. Supposons que les hypothèses 5.1 et 5.2 sont vérifiées. Il existe* $\mu \in]0, +\infty[$ *tel que, pour tout* $k \in \mathbb{N}$,

$$\mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{f}(\mathbf{x}_k) - \frac{\mu}{2} \|\mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}\|^2 = \mathsf{f}(\mathbf{x}_k) - \frac{\mu}{2} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2. \tag{5.9}$$

Démonstration. Soit $k \in \mathbb{N}$. On a

$$\mathsf{f}(\mathbf{x}_{k+1}) = \mathsf{h}(\mathbf{x}_{k+1}) + \mathsf{g}(\mathbf{x}_{k+1}).$$

D'une part, d'après l'hypothèse 5.2(i),

$$\mathsf{h}(\mathbf{x}_{k+1}) \leqslant \mathsf{h}(\mathbf{x}_k) + \left\langle \mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}, \nabla_{j_k}\mathsf{h}(\mathbf{x}_k) \right\rangle + \frac{1}{2}\|\mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}\|^2_{\mathbf{A}_{j_k}(\mathbf{x}_k)}. \tag{5.10}$$

D'autre part, d'après l'algorithme 5.5, on a

$$(\forall j \in \overline{\jmath}_k) \quad \mathbf{x}_{k+1}^{(j)} = \mathbf{x}_k^{(j)}, \tag{5.11}$$

ainsi

$$\begin{aligned}\mathsf{g}(\mathbf{x}_{k+1}) &= \mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)}) + \sum_{j\in\overline{\jmath}_k} \mathsf{g}_j(\mathbf{x}_{k+1}^{(j)}) \\ &= \mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)}) + \sum_{j\in\overline{\jmath}_k} \mathsf{g}_j(\mathbf{x}_k^{(j)}) \\ &= \mathsf{g}(\mathbf{x}_k) + \left(\mathsf{g}_{j_k}(\mathbf{x}_{k+1}^{(j_k)}) - \mathsf{g}_{j_k}(\mathbf{x}_k^{(j_k)})\right).\end{aligned}$$

Alors, la première inégalité de l'algorithme 5.5 nous donne

$$\mathsf{g}(\mathbf{x}_{k+1}) \leqslant \mathsf{g}(\mathbf{x}_k) - \left\langle \mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}, \nabla_{j_k}\mathsf{h}(\mathbf{x}_k) \right\rangle - \alpha_1\|\mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}\|^2_{\mathbf{A}_{j_k}(\mathbf{x}_k)}. \tag{5.12}$$

Par conséquent, en combinant (5.10) et (5.12), on obtient

$$\mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{f}(\mathbf{x}_k) - \left(\alpha_1 - \frac{1}{2}\right)\|\mathbf{x}_{k+1}^{(j_k)} - \mathbf{x}_k^{(j_k)}\|^2_{\mathbf{A}_{j_k}(\mathbf{x}_k)}. \tag{5.13}$$

Pour conclure, (5.9) découle de l'hypothèse 5.2(ii), en posant $\mu = \underline{\nu}(2\alpha_1 - 1)$ (puisque $\alpha_1 \in]1/2, +\infty[$), et en utilisant l'équation (5.11). □

Soit $(\boldsymbol{\chi}_k)_{k\in\mathbb{N}}$ la suite définie par

$$(\forall k \in \mathbb{N}) \quad \boldsymbol{\chi}_k = (\mathbf{x}_{k+l+1} - \mathbf{x}_{k+l})_{0\leqslant l\leqslant K-1} \in (\mathbb{R}^N)^K, \tag{5.14}$$

où $(\mathbf{x}_k)_{k\in\mathbb{N}}$ est une suite générée par l'algorithme 5.5 et K est la constante positive désignant le nombre d'itérations maximal au cours desquelles chaque bloc doit être mis à jour (d'après l'hypothèse 5.3). On a alors

$$\|\boldsymbol{\chi}_k\|^2 = \sum_{l=0}^{K-1} \|\mathbf{x}_{k+l+1} - \mathbf{x}_{k+l}\|^2,$$

et la propriété suivante est satisfaite.

Lemme 5.2. *Soit $(\mathbf{x}_k)_{k\in\mathbb{N}}$ une suite générée par l'algorithme 5.5. Supposons que les hypothèses 5.1, 5.2 et 5.3 sont satisfaites. Pour tout $k \in \mathbb{N}$,*

$$\mathsf{f}(\mathbf{x}_{k+K}) \leqslant \mathsf{f}(\mathbf{x}_k) - \frac{\mu}{2}\|\boldsymbol{\chi}_k\|^2,$$

où $\mu \in]0, +\infty[$ est la même constante que celle donnée dans le lemme 5.1.

Démonstration. Soit $k \in \mathbb{N}$. D'après le lemme 5.1, on a

$$\begin{aligned}
\mathsf{f}(\mathbf{x}_{k+K}) &\leqslant \mathsf{f}(\mathbf{x}_{k+K-1}) - \frac{\mu}{2}\|\mathbf{x}_{k+K} - \mathbf{x}_{k+K-1}\|^2 \\
&\leqslant \mathsf{f}(\mathbf{x}_{k+K-2}) - \frac{\mu}{2}\left(\|\mathbf{x}_{k+K-1} - \mathbf{x}_{k+K-2}\|^2 + \|\mathbf{x}_{k+K} - \mathbf{x}_{k+K-1}\|^2\right) \\
&\vdots \\
&\leqslant \mathsf{f}(\mathbf{x}_k) - \frac{\mu}{2}\sum_{l=0}^{K-1}\|\mathbf{x}_{k+l+1} - \mathbf{x}_{k+l}\|^2.
\end{aligned}$$

□

5.5.2 Théorème de convergence

Dans cette section nous allons énoncer deux théorèmes de convergence :

- Le premier assure la convergence de la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ générée par l'algorithme 5.5 vers un point critique de f.
- Le second est un résultat de convergence local assurant que si l'algorithme 5.5 est initialisé de façon judicieuse, alors la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ converge vers une solution du problème 5.1.

Pour cela, nous allons, dans un premier temps, donner deux lemmes qui nous permettrons de manipuler la règle de mise à jour des bloc essentiellement cyclique.

Lemme 5.3. *Soit $(\mathbf{x}_k)_{k\in\mathbb{N}}$ une suite générée par l'algorithme 5.5. Soit $k_0 \in \mathbb{N}$. On définit $\mathcal{J}_{k_0}$ comme étant un sous-ensemble de $\{1,\dots,J\}$ tel que $j_{k_0} \in \mathcal{J}_{k_0}$. On suppose que les hypothèses 5.1 et 5.2 sont satisfaites. Alors, on a*

$$\begin{aligned}
\sum_{j\in\mathcal{J}_{k_0}}\|\nabla_j\mathsf{h}(\mathbf{x}_{k_0+1}) + \mathbf{r}_{k_0+1}^{(j)}\|^2 \leqslant 2(\beta^2 + \alpha_2^2\overline{\nu})\|\mathbf{x}_{k_0+1} - \mathbf{x}_{k_0}\|^2 \\
+ 2\sum_{j\in\mathcal{J}_{k_0}\setminus\{j_{k_0}\}}\|\nabla_j\mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0}^{(j)}\|^2, \quad (5.15)
\end{aligned}$$

où $\mathbf{r}_{k_0+1}^{(j_{k_0})}$ est donné par l'algorithme 5.5 et, pour tout $j \in \mathcal{J}_{k_0}\setminus\{j_{k_0}\}$, $\mathbf{r}_{k_0+1}^{(j)} \in \partial\mathsf{g}_j(\mathbf{x}_{k_0+1}^{(j)})$ et $\mathbf{r}_{k_0}^{(j)} \in \partial\mathsf{g}_j(\mathbf{x}_{k_0}^{(j)})$.

Démonstration. Soit $k_0 \in \mathbb{N}$. D'après l'inégalité de Jensen, on a

$$\begin{aligned}
\sum_{j\in\mathcal{J}_{k_0}}\|\nabla_j\mathsf{h}(\mathbf{x}_{k_0+1}) + \mathbf{r}_{k_0+1}^{(j)}\|^2 \leqslant 2\sum_{j\in\mathcal{J}_{k_0}}\|\nabla_j\mathsf{h}(\mathbf{x}_{k_0+1}) - \nabla_j\mathsf{h}(\mathbf{x}_{k_0})\|^2 \\
+ 2\sum_{j\in\mathcal{J}_{k_0}}\|\nabla_j\mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0+1}^{(j)}\|^2. \quad (5.16)
\end{aligned}$$

D'une part, puisque $\sum_{j=1}^{J}\|\nabla_j\mathsf{h}(\mathbf{x}_{k_0+1}) - \nabla_j\mathsf{h}(\mathbf{x}_{k_0})\|^2 = \|\nabla\mathsf{h}(\mathbf{x}_{k_0+1}) - \nabla\mathsf{h}(\mathbf{x}_{k_0})\|^2$, d'après l'hypothèse 5.1(ii) on a

$$\sum_{j\in\mathcal{J}_{k_0}}\|\nabla_j\mathsf{h}(\mathbf{x}_{k_0+1}) - \nabla_j\mathsf{h}(\mathbf{x}_{k_0})\|^2 \leqslant \beta^2\|\mathbf{x}_{k_0+1} - \mathbf{x}_{k_0}\|^2. \quad (5.17)$$

D'autre part, puisque $j_{k_0} \in \mathcal{J}_{k_0}$,

$$\sum_{j\in\mathcal{J}_{k_0}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0+1}^{(j)}\|^2 = \|\nabla_{j_{k_0}} \mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0+1}^{(j_{k_0})}\|^2 + \sum_{j\in\mathcal{J}_{k_0}\setminus\{j_{k_0}\}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0+1}^{(j)}\|^2.$$

Par ailleurs, pour tout $j \in \mathcal{J}_{k_0} \setminus \{j_{k_0}\}$, on a $\mathbf{x}_{k_0+1}^{(j)} = \mathbf{x}_{k_0}^{(j)}$. Ainsi, d'après la deuxième inégalité de l'algorithme 5.5 et l'hypothèse 5.2(ii), on a

$$\sum_{j\in\mathcal{J}_{k_0}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0+1}^{(j)}\|^2 \leqslant \alpha_2^2\overline{\nu}\|\mathbf{x}_{k_0+1} - \mathbf{x}_{k_0}\|^2 + \sum_{j\in\mathcal{J}_{k_0}\setminus\{j_{k_0}\}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0}^{(j)}\|^2. \tag{5.18}$$

Pour conclure, (5.15) peut être déduite en combinant les équation (5.16), (5.17) et (5.18). □

Lemme 5.4. *Soit $(\mathbf{x}_k)_{k\in\mathbb{N}}$ une suite générée par l'algorithme 5.5. Soient $k_0 \in \mathbb{N}$ et $k_0' \in \mathbb{N}$ deux itérations telles que $k_0 \leqslant k_0'$. On définit le sous-ensemble $\mathcal{J}_{k_0,k_0'}$ de $\{1,\dots,J\}$ tel que, pour tout $k \in \{k_0,\dots,k_0'\}$, $j_k \in \mathcal{J}_{k_0,k_0'}$. On suppose que les hypothèses 5.1 et 5.2 sont satisfaites. On a alors*

$$\begin{aligned}\sum_{j\in\mathcal{J}_{k_0,k_0'}} &\|\nabla_j \mathsf{h}(\mathbf{x}_{k_0'+1}) + \mathbf{r}_{k_0'+1}^{(j)}\|^2 \\ &\leqslant (\beta^2 + \alpha_2^2\overline{\nu}) \sum_{k=k_0}^{k_0'} 2^{k_0'+1-k}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 + 2^{k_0'+1-k_0} \sum_{j\in\mathcal{J}_{k_0,k_0'}\setminus\{j_{k_0}\}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0}^{(j)}\|^2,\end{aligned}$$

où $\mathbf{r}_{k_0'+1}^{(j_{k_0'})}$ est donné par l'algorithme 5.5, pour tout $j \in \mathcal{J}_{k_0,k_0'} \setminus \{j_{k_0'}\}$, $\mathbf{r}_{k_0'+1}^{(j)} \in \partial \mathsf{g}_j(\mathbf{x}_{k_0'+1}^{(j)})$ et, pour tout $j \in \mathcal{J}_{k_0,k_0'} \setminus \{j_{k_0}\}$, $\mathbf{r}_{k_0}^{(j)} \in \partial \mathsf{g}_j(\mathbf{x}_{k_0}^{(j)})$.

Démonstration. Soit $(k_0, k_0') \in \mathbb{N}^2$ tel que $k_0 \leqslant k_0'$. Sous les hypothèses considérées, on peut

appliquer successivement le lemme 5.3 pour $k_0', k_0'-1, \ldots, k_0$. Ainsi, on obtient

$$\begin{aligned}
&\sum_{j\in\mathcal{J}_{k_0,k_0'}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0'+1}) + \mathbf{r}_{k_0'+1}^{(j)}\|^2 \\
&\quad\leqslant (\beta^2+\alpha_2^2\overline{\nu})2\|\mathbf{x}_{k_0'+1}-\mathbf{x}_{k_0'}\|^2 + 2\sum_{j\in\mathcal{J}_{k_0,k_0'}\setminus\{j_{k_0'}\}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0'}) + \mathbf{r}_{k_0'}^{(j)}\|^2 \\
&\quad\leqslant (\beta^2+\alpha_2^2\overline{\nu})2\|\mathbf{x}_{k_0'+1}-\mathbf{x}_{k_0'}\|^2 + 2\sum_{j\in\mathcal{J}_{k_0,k_0'}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0'}) + \mathbf{r}_{k_0'}^{(j)}\|^2 \\
&\quad\leqslant (\beta^2+\alpha_2^2\overline{\nu})\Big(2\|\mathbf{x}_{k_0'+1}-\mathbf{x}_{k_0'}\|^2 + 2^2\|\mathbf{x}_{k_0'}-\mathbf{x}_{k_0'-1}\|^2\Big) \\
&\qquad\qquad + 2^2\sum_{j\in\mathcal{J}_{k_0,k_0'}\setminus\{j_{k_0'-1}\}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0'-1}) + \mathbf{r}_{k_0'-1}^{(j)}\|^2 \\
&\quad\leqslant (\beta^2+\alpha_2^2\overline{\nu})\Big(2\|\mathbf{x}_{k_0'+1}-\mathbf{x}_{k_0'}\|^2 + 2^2\|\mathbf{x}_{k_0'}-\mathbf{x}_{k_0'-1}\|^2 + 2^3\|\mathbf{x}_{k_0'-1}-\mathbf{x}_{k_0'-2}\|^2\Big) \\
&\qquad\qquad + 2^3\sum_{j\in\mathcal{J}_{k_0,k_0'}\setminus\{j_{k_0'-2}\}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0'-2}) + \mathbf{r}_{k_0'-2}^{(j)}\|^2 \\
&\qquad\vdots \\
&\quad\leqslant (\beta^2+\alpha_2^2\overline{\nu})\sum_{k=k_0}^{k_0'} 2^{k_0'+1-k}\|\mathbf{x}_{k+1}-\mathbf{x}_k\|^2 \\
&\qquad\qquad + 2^{k_0'+1-k_0}\sum_{j\in\mathcal{J}_{k_0,k_0'}\setminus\{j_{k_0}\}} \|\nabla_j \mathsf{h}(\mathbf{x}_{k_0}) + \mathbf{r}_{k_0}^{(j)}\|^2.
\end{aligned}$$

□

Nous pouvons maintenant énoncer notre résultat principal concernant la convergence de l'algorithme 5.5 :

Théorème 5.1. *Soit $(\mathbf{x}_k)_{k\in\mathbb{N}}$ une suite générée par l'algorithme 5.5. Supposons que les hypothèses 5.1-5.3 sont satisfaites. De plus, supposons que la fonction f satisfait l'inégalité de Kurdyka-Łojasiewicz donnée dans la définition 2.20. Alors, les assertions suivantes sont vérifiées :*

(i) *La suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ converge vers un point critique $\widehat{\mathbf{x}}$ de f.*

(ii) *Cette suite vérifie*
$$\sum_{k=0}^{+\infty} \|\mathbf{x}_{k+1}-\mathbf{x}_k\| < +\infty.$$

(iii) *La suite $\big(\mathsf{f}(\mathbf{x}_k)\big)_{k\in\mathbb{N}}$ est décroissante et converge vers $\mathsf{f}(\widehat{\mathbf{x}})$.*

La démonstration du théorème 5.1 est donnée dans l'annexe 5.A. On peut en déduire le corollaire suivant, énonçant la convergence locale de l'algorithme 5.5 vers un minimiseur global de f.

Corollaire 5.1. *Soit $(\mathbf{x}_k)_{k\in\mathbb{N}}$ une suite générée par l'algorithme (5.5). Supposons que les hypothèses 5.1-5.3 sont satisfaites. De plus, supposons que f satisfait l'inégalité de Kurdyka-Łojasiewicz. Il existe $\upsilon \in]0,+\infty[$ tel que, si*

$$\mathsf{f}(\mathbf{x}_0) \leqslant \inf_{\mathbf{x}\in\mathbb{R}^N} \mathsf{f}(\mathbf{x}) + \upsilon,$$

alors, $(\mathbf{x}_k)_{k\in\mathbb{N}}$ converge vers une solution du problème (5.1).

Démonstration. La démonstration de ce corollaire se fait de façon analogue à la démonstration du Corollaire 3.3 du chapitre 3. □

5.5.3 Borne supérieure sur la vitesse de convergence

D'après le théorème 5.1, la limite $\widehat{\mathbf{x}}$ de la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ générée par l'algorithme (5.5) est un point critique de la fonction f, dès lors que les hypothèses 5.1-5.3 sont vérifiées et que f satisfait l'inégalité de KL. Ainsi, en procédant de la même façon que pour l'obtention de l'équation (5.29), il existe $\zeta \in]0,+\infty[$, $\kappa \in]0,+\infty[$ et $\theta \in [0,1[$ tels que ((ii)) soit vérifiée pour tout $\mathbf{x} \in \mathbb{R}^N$ satisfaisant $G(\mathbf{x}) \leqslant G(\widehat{\mathbf{x}}) + \zeta$. La valeur θ est alors appelée un *exposant de Łojasiewicz de G en $\widehat{\mathbf{x}}$*. Comme d'autres algorithmes basés sur l'inégalité de KL [Attouch et Bolte, 2009; Attouch *et al.*, 2010a], la vitesse de convergence locale de l'algorithme 5.5 dépend de cet exposant.

Le lemme ci-dessous nous sera utile pour déterminer la vitesse de convergence de l'algorithme 5.5 :

Lemme 5.5. [Attouch et Bolte, 2009] *Soit $(\Lambda_m)_{m\in\mathbb{N}}$ une suite positive réelle qui décroit vers 0. Supposons qu'il existe $m^* \in \mathbb{N}^*$ et $C \in]0,+\infty[$ tels que, pour tout $m \geqslant m^*$,*

$$\Lambda_m \leqslant (\Lambda_{m-1} - \Lambda_m) + C(\Lambda_{m-1} - \Lambda_m)^{\frac{1-\theta}{\theta}}, \tag{5.19}$$

où $\theta \in]0,1[$. On observe deux cas :

(i) *Si $\theta \in]\frac{1}{2},1[$, alors il existe $\lambda \in]0,+\infty[$ tel que*

$$(\forall m \geqslant 1) \quad \Lambda_m \leqslant \lambda m^{-\frac{1-\theta}{2\theta-1}}.$$

(ii) *Si $\theta \in]0,\frac{1}{2}]$, alors il existe $\lambda \in]0,+\infty[$ et $\tau \in [0,1[$ tels que*

$$(\forall m \in \mathbb{N}) \quad \Lambda_m \leqslant \lambda \tau^m.$$

Théorème 5.2. *Soit $(\mathbf{x}_k)_{k\in\mathbb{N}}$ une suite générée par l'algorithme (5.5). Supposons que les hypothèses 5.1-5.3 sont satisfaites et que f satisfait l'inégalité de KL. Soit θ l'exposant de Łojasiewicz de G au point limite $\widehat{\mathbf{x}}$ de la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$. Les propriétés suivantes sont alors satisfaites :*

(i) *Si $\theta \in]\frac{1}{2},1[$, alors il existe $(\lambda',\lambda'') \in]0,+\infty[^2$ tel que*

$$(\forall k > K) \quad \|\mathbf{x}_k - \widehat{\mathbf{x}}\| \leqslant \lambda' \Big(\frac{k}{K} - 1\Big)^{-\frac{1-\theta}{2\theta-1}}, \tag{5.20}$$

$$(\forall k > 2K) \quad G(\mathbf{x}_k) - G(\widehat{\mathbf{x}}) \leqslant \lambda'' \Big(\frac{k}{K} - 2\Big)^{-\frac{1-\theta}{\theta(2\theta-1)}}. \tag{5.21}$$

(ii) *Si* $\theta \in]0, \frac{1}{2}]$, *alors il existe* $(\lambda', \lambda'') \in]0, +\infty[^2$ *et* $\tau' \in [0, 1[$ *tels que*

$$(\forall k \in \mathbb{N}) \quad \|\mathbf{x}_k - \widehat{\mathbf{x}}\| \leqslant \lambda'(\tau')^k, \tag{5.22}$$

$$G(\mathbf{x}_k) - G(\widehat{\mathbf{x}}) \leqslant \lambda''(\tau')^{\frac{k}{\theta}}. \tag{5.23}$$

(iii) *Si* $\theta = 0$, *alors la suite* $(\mathbf{x}_k)_{k\in\mathbb{N}}$ *converge en un nombre fini d'itérations.*

La démonstration du théorème 5.2 est donnée dans l'annexe 5.B.

Remarque 5.5.

(i) *Remarquons que, lorsque* f *est fortement convexe, l'exposant de Łojasiewicz* θ *de* f *est égal à* $1/2$. *Dans ce cas,* $\widehat{\mathbf{x}}$ *est un minimiseur global de* f *et les suites* $(\|\mathbf{x}_k - \widehat{\mathbf{x}}\|)_{k\in\mathbb{N}}$ *et* $(\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}}))_{k\in\mathbb{N}}$ *convergent linéairement.*

(ii) *On remarque que, si* $\theta \in]0, 1/2]$, *alors, pour* m *suffisamment grand,* (5.19) *implique*

$$\Lambda_m \leqslant (1 + C)(\Lambda_{m-1} - \Lambda_m),$$

et donc la constante τ' *apparaissant dans les équations* (5.22)-(5.23) *peut être choisie égale à* $((1 + C)/(2 + C))^{1/K}$ *où* C *est donné par* (5.40).

Remarquons, d'autre part, que la convergence et la vitesse de convergence de l'algorithme 5.4 ont aussi été étudiés par [Frankel *et al.*, 2014]. Les auteurs ont démontré la convergence de l'algorithme pour la résolution d'une classe de problèmes non-convexes incluant celui étudié dans ce chapitre. La principale différence existant entre ces deux travaux réside dans la règle de mise à jour des blocs. En effet, dans [Frankel *et al.*, 2014], les auteurs présentent des propriétés de convergence de l'algorithme 5.4 lorsque les blocs sont mis à jour selon une règle cyclique. Les démonstrations de convergence ne sont donc pas les mêmes que celles présentées dans la section 5.5. De plus, nous verrons dans le chapitre 4 une application pour laquelle il s'est révélé très intéressant numériquement d'utiliser une règle essentiellement cyclique.

5.6 Conclusion

Dans ce chapitre nous avons proposé une méthode d'optimisation permettant de minimiser la somme d'une fonction différentiable h et d'une fonction non nécessairement lisse g additivement séparable par bloc. La méthode repose à la fois sur un algorithme explicite-implicite à métrique variable et sur une stratégie alternée par bloc. Nous avons vu dans les chapitres 3 et 4 que la métrique variable permet d'accélérer en pratique la convergence de l'algorithme explicite-implicite. Combiner cette méthode avec un principe de minimisation par bloc nous permet en plus de traiter des données de grande taille en réduisant le coup de calcul ainsi que l'allocation de mémoire par itération.

De plus, nous avons proposé une méthode basée sur le principe MM pour choisir la métrique et nous avons donné une règle générale acyclique de mise à jour des blocs. La convergence des

suites $(\mathbf{x}_k)_{k\in\mathbb{N}}$ générées par l'algorithme proposé, ainsi que par sa version inexacte, et du critère associé a été démontrée en utilisant l'inégalité de KL. Enfin, nous avons étudié la vitesse de convergence de $(\mathbf{x}_k)_{k\in\mathbb{N}}$ et du critère $(\mathsf{f}(\mathbf{x}_k))_{k\in\mathbb{N}}$ en fonction de l'exposant de Łojasiewicz de la fonction $\mathsf{f} = \mathsf{h} + \mathsf{g}$.

Afin de montrer l'intérêt pratique de l'algorithme 5.4 (et de sa version inexacte), nous en présenterons plusieurs exemples d'application dans le chapitre 6. En particulier, nous verrons que la mise à jour des blocs de façon acyclique permet dans certains cas d'accélérer de manière significative la convergence de l'algorithme.

5.A Démonstration du théorème 5.1

La démonstration du théorème 5.1 se fait en trois étapes : une première étape pour montrer que la suite $(\mathsf{f}(\mathbf{x}_k))_{k\in\mathbb{N}}$ est convergente, une seconde étape pour démontrer la convergence de la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ vers un point $\widehat{\mathbf{x}} \in \mathbb{R}^N$, et une dernière étape permettant de s'assurer que $\widehat{\mathbf{x}}$ est un point critique de f.

(i) D'après le lemme 5.1, on a

$$(\forall k \in \mathbb{N}) \quad \mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{f}(\mathbf{x}_k),$$

donc, $(\mathsf{f}(\mathbf{x}_k))_{k\in\mathbb{N}}$ est une suite décroissante. De plus, puisque $\mathbf{x}_0 \in \operatorname{dom}\mathsf{g}$, d'après la remarque 5.2(iii), les éléments de la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ sont dans le sous-ensemble compact $E = \operatorname{lev}_{\leqslant \mathsf{f}(\mathbf{x}_0)} \mathsf{f}$ de $\operatorname{dom}\mathsf{g}$, et f est bornée inférieurement. Ainsi, $(\mathsf{f}(\mathbf{x}_k))_{k\in\mathbb{N}}$ converge vers un réel noté ξ, et $(\mathsf{f}(\mathbf{x}_k) - \xi)_{k\in\mathbb{N}}$ est une suite positive convergeant vers 0.

(ii) Par ailleurs, d'après le lemme 5.2, on a

$$(\forall k \in \mathbb{N}) \quad \frac{\mu}{2}\|\boldsymbol{\chi}_k\|^2 \leqslant (\mathsf{f}(\mathbf{x}_k) - \xi) - (\mathsf{f}(\mathbf{x}_{k+K}) - \xi). \tag{5.24}$$

Soit $\psi\colon [0,+\infty[\to [0,+\infty[\colon \mathsf{u} \mapsto \mathsf{u}^{\frac{1}{1-\theta}}$ une fonction convexe, où $\theta \in [0,1[$. L'inégalité de gradient nous donne

$$(\forall (\mathsf{u},\mathsf{v}) \in [0,+\infty[^2) \quad \psi(\mathsf{u}) - \psi(\mathsf{v}) \leqslant \dot{\psi}(\mathsf{u})(\mathsf{u}-\mathsf{v}),$$

qui, après un changement de variable, peut être réécrite

$$(\forall (\mathsf{u},\mathsf{v}) \in [0,+\infty[^2) \quad \mathsf{u} - \mathsf{v} \leqslant (1-\theta)^{-1}\mathsf{u}^{\theta}(\mathsf{u}^{1-\theta} - \mathsf{v}^{1-\theta}).$$

En utilisant l'inégalité ci-dessus avec $\mathsf{u} = \mathsf{f}(\mathbf{x}_k) - \xi$ et $\mathsf{v} = \mathsf{f}(\mathbf{x}_{k+K}) - \xi$, on obtient

$$(\forall k \in \mathbb{N}) \quad \big(\mathsf{f}(\mathbf{x}_k) - \xi\big) - \big(\mathsf{f}(\mathbf{x}_{k+K}) - \xi\big) \leqslant (1-\theta)^{-1}\big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{\theta}\Delta_k,$$

où

$$(\forall k \in \mathbb{N}) \quad \Delta_k = \big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{1-\theta} - \big(\mathsf{f}(\mathbf{x}_{k+K}) - \xi\big)^{1-\theta}.$$

Donc, en combinant cette dernière inégalité avec l'équation (5.24), on a

$$(\forall k \in \mathbb{N}) \quad \|\boldsymbol{\chi}_k\|^2 \leqslant 2\mu^{-1}(1-\theta)^{-1}\big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{\theta}\Delta_k. \tag{5.25}$$

Posons

$$(\forall k \in \mathbb{N}) \qquad \mathbf{t}_k = \left(\nabla_j \mathsf{h}(\mathbf{x}_k) + \mathbf{r}_\ell^{(j)}\right)_{1 \leqslant j \leqslant J} \in \mathbb{R}^{N_1} \times \ldots \times \mathbb{R}^{N_J}, \tag{5.26}$$

où, pour tout $j \in \{1, \ldots, J\}$, $\mathbf{r}_k^{(j)} \in \partial \mathsf{g}_j(\mathbf{x}_k^{(j)})$. La règle de différentiabilité pour les fonctions séparables nous donne

$$\mathbf{r}_k = \left(\mathbf{r}_k^{(j)}\right)_{1 \leqslant j \leqslant J} \in \partial \mathsf{g}(\mathbf{x}_k). \tag{5.27}$$

Ainsi, pour tout $k \in \mathbb{N}$,

$$\mathbf{t}_k \in \partial \mathsf{f}(\mathbf{x}_k). \tag{5.28}$$

Puisque E est borné, d'après l'inégalité de KL, il existe trois constantes $\kappa > 0$, $\zeta > 0$ et $\theta \in [0, 1[$ telles que

$$(\forall \mathbf{t} \in \partial \mathsf{f}(\mathbf{x})) \quad \kappa |\mathsf{f}(\mathbf{x}) - \xi|^\theta \leqslant \|\mathbf{t}\|,$$

pour tout $\mathbf{x} \in E$ tel que $|\mathsf{f}(\mathbf{x}) - \xi| \leqslant \zeta$. La suite $\big(\mathsf{f}(\mathbf{x}_k)\big)_{k \in \mathbb{N}}$ convergeant vers ξ, il existe $k^* \in \mathbb{N}$, tel que, pour tout $k \geqslant k^*$, $|\mathsf{f}(\mathbf{x}_k) - \xi| < \zeta$. Par conséquent, on a

$$(\forall k \geqslant k^*) \quad \kappa |\mathsf{f}(\mathbf{x}_k) - \xi|^\theta \leqslant \|\mathbf{t}_k\|, \tag{5.29}$$

où $\mathbf{t}_k$ est donné par (5.26). Soit K, défini par la définition 5.1(ii), le nombre maximal d'itérations au cours desquelles chaque bloc doit être mis à jour au moins une fois. Pour tout $k \in \mathbb{N}$,

$$\|\mathbf{t}_{k+K}\|^2 = \| \left(\nabla_j \mathsf{h}(\mathbf{x}_{k+K}) + \mathbf{r}_{k+K}^{(j)}\right)_{1 \leqslant j \leqslant J} \|^2 = \sum_{j=1}^{J} \|\nabla_j \mathsf{h}(\mathbf{x}_{k+K}) + \mathbf{r}_{k+K}^{(j)}\|^2.$$

Soit $l_{k,\sigma_k(J)}$ donné par (5.7). Pour tout $l \in \{k + l_{l,\sigma_k(J)}, \ldots, k+K-1\}$, soit $\mathbf{r}_{l+1}^{(j_l)} \in \partial \mathsf{g}_{j_l}(\mathbf{x}_{l+1}^{(j_l)})$ défini par l'algorithme 5.5. On peut alors appliquer le lemme 5.4 avec $k_0 = k + l_{k,\sigma_k(J)}$, $k_0' = k + K - 1$ et $\mathcal{J}_{k_0,k_0'} = \{1, \ldots, J\}$:

$$\begin{aligned}\|\mathbf{t}_{k+K}\|^2 \leqslant (\beta^2 + \alpha_2^2 \overline{\nu}) & \sum_{l=k+l_{k,\sigma_k(J)}}^{k+K-1} 2^{k+K-l} \|\mathbf{x}_{l+1} - \mathbf{x}_l\|^2 \\ & + 2^{K - l_{k,\sigma_k(J)}} \sum_{\substack{j=1 \\ j \neq \sigma_k(J)}}^{J} \|\nabla_j \mathsf{h}(\mathbf{x}_{k+l_{k,\sigma_k(J)}}) + \mathbf{r}_{k+l_{k,\sigma_k(J)}}^{(j)}\|^2.\end{aligned}$$

En appliquant une nouvelle fois le lemme 5.4 à la somme

$$\sum_{\substack{j=1 \\ j \neq \sigma_k(J)}}^{J} \|\nabla_j \mathsf{h}(\mathbf{x}_{k+l_{k,\sigma_k(J)}}) + \mathbf{r}_{k+l_{k,\sigma_k(J)}}^{(j)}\|^2$$

avec $k_0 = k + l_{k,\sigma_k(J-1)}$, $k_0' = k + l_{k,\sigma_k(J)} - 1$ et $\mathcal{J}_{k_0,k_0'} = \{1,\dots,J\} \setminus \{\sigma_k(J)\}$, on obtient

$$\begin{aligned}\|\mathbf{t}_{k+K}\|^2 \leqslant{}& (\beta^2 + \alpha_2^2\overline{\nu}) \sum_{l=k+l_{k,\sigma_k(J)}}^{k+K-1} 2^{k+K-l}\|\mathbf{x}_{l+1} - \mathbf{x}_l\|^2 \\ &+ (\beta^2 + \alpha_2^2\overline{\nu}) \sum_{l=k+l_{k,\sigma_k(J-1)}}^{k+l_{k,\sigma_k(J)}-1} 2^{k+K-l}\|\mathbf{x}_{l+1} - \mathbf{x}_l\|^2 \\ &+ 2^{K-l_{k,\sigma_k(J-1)}} \sum_{\substack{j=1 \\ j\neq\sigma_k(i), i\in\{J-1,J\}}}^{J} \|\nabla_j \mathsf{h}(\mathbf{x}_{k+l_{k,\sigma_k(J-1)}}) + \mathbf{r}^{(j)}_{k+l_{k,\sigma_k(J-1)}}\|^2.\end{aligned}$$

En réitérant le procédé de façon similaire pour $i \in \{1,\dots,J-2\}$, on a

$$\begin{aligned}\|\mathbf{t}_{k+K}\|^2 \leqslant{}& (\beta^2 + \alpha_2^2\overline{\nu}) \sum_{l=k+l_{k,\sigma_k(J)}}^{k+K-1} 2^{k+K-l}\|\mathbf{x}_{l+1} - \mathbf{x}_l\|^2 \\ &+ (\beta^2 + \alpha_2^2\overline{\nu}) \sum_{i=1}^{J-1} \sum_{l=k+l_{k,\sigma_k(j)}}^{k+l_{k,\sigma_k(j+1)}-1} 2^{k+K-l}\|\mathbf{x}_{l+1} - \mathbf{x}_l\|^2, \quad (5.30)\end{aligned}$$

où l'on a utilisé le fait que $\{1,\dots,J\} \setminus \{\sigma_k(1),\dots,\sigma_k(J)\} = \varnothing$. Ainsi

$$\sum_{\substack{j=1 \\ j\neq\sigma_k(i), i\in\{1,\dots,J\}}}^{J} \|\nabla_j \mathsf{h}(\mathbf{x}_k) + \mathbf{r}^{(j)}_k\|^2 = 0.$$

Comme $l_{k,\sigma_k(1)} = 0$ et, pour tout $l \in \{k,\dots,k+K-1\}$, $2^{k+K-l} \leqslant 2^K$, il découle de (5.14) et (5.30) que

$$(\forall k \in \mathbb{N}) \quad \|\mathbf{t}_{k+K}\|^2 \leqslant 2^K(\beta^2 + \alpha_2^2\overline{\nu}) \sum_{l=k}^{k+K-1} \|\mathbf{x}_{l+1} - \mathbf{x}_l\|^2 = 2^K(\beta^2 + \alpha_2^2\overline{\nu})\|\boldsymbol{\chi}_k\|^2. \quad (5.31)$$

En combinant les équations (5.25), (5.29) et (5.31) on obtient

$$(\forall k \geqslant \max\{k^*, K\}) \quad \|\boldsymbol{\chi}_k\|^2 \leqslant 2\mu^{-1}(1-\theta)^{-1}\kappa^{-1}2^{K/2}(\beta^2 + \alpha_2^2\overline{\nu})^{1/2}\|\boldsymbol{\chi}_{k-K}\|\Delta_k.$$

En utilisant l'inégalité

$$(\forall (\mathsf{u},\mathsf{v}) \in [0,+\infty[^2) \quad (\mathsf{uv})^{1/2} \leqslant \frac{1}{2}(\mathsf{u}+\mathsf{v}),$$

et en posant $\mathsf{u} = \|\boldsymbol{\chi}_{k-K}\|$ et $\mathsf{v} = 2\mu^{-1}(1-\theta)^{-1}\kappa^{-1}2^{K/2}(\beta^2 + \alpha_2^2\overline{\nu})^{1/2}\Delta_k$, on obtient

$$(\forall k \geqslant \max\{k^*, K\}) \quad \|\boldsymbol{\chi}_k\| \leqslant \frac{1}{2}\|\boldsymbol{\chi}_{k-K}\| + \mu^{-1}(1-\theta)^{-1}\kappa^{-1}2^{K/2}(\beta^2 + \alpha_2^2\overline{\nu})^{1/2}\Delta_k. \quad (5.32)$$

Par ailleurs, remarquons que

$$\sum_{k=k^*}^{+\infty} \Delta_k = \sum_{k=k^*}^{+\infty} \big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{1-\theta} - \big(\mathsf{f}(\mathbf{x}_{k+K}) - \xi\big)^{1-\theta} = \sum_{k=k^*}^{k^*+K-1} \big(\mathsf{f}(\mathbf{x}_k) - \xi\big)^{1-\theta}.$$

On peut donc en déduire que $(\Delta_k)_{k\in\mathbb{N}}$ est une suite sommable. De plus, $(\|\boldsymbol{\chi}_k\|)_{k\geqslant\max\{k^*,K\}}$ satisfaisant l'inégalité (5.32), $(\|\boldsymbol{\chi}_k\|)_{k\in\mathbb{N}}$ est aussi une suite sommable. Enfin, d'après l'équation (5.14),

$$(\forall k \in \mathbb{N}) \quad \|\mathbf{x}_{k+1} - \mathbf{x}_k\| \leqslant \|\boldsymbol{\chi}_k\|,$$

donc $(\|\mathbf{x}_{k+1} - \mathbf{x}_k\|)_{k\in\mathbb{N}}$ est une suite sommable.

Par conséquent, la suite $(\mathbf{x}_k)_{k\in\mathbb{N}}$ satisfait l'assertion (ii). De plus, cette condition implique que $(\mathbf{x}_k)_{k\in\mathbb{N}}$ est une suite de Cauchy. Ainsi, on peut en déduire qu'elle converge vers un point $\widehat{\mathbf{x}}$.

(iii) Il nous reste à démontrer que la limite $\widehat{\mathbf{x}}$ est un point critique de f. D'après (5.28), on a, pour tout $k \in \mathbb{N}$,

$$(\mathbf{x}_k, \mathbf{t}_k) \in \operatorname{Graph} \partial\mathsf{f}.$$

Par ailleurs, la suite $(\|\boldsymbol{\chi}_k\|)_{k\in\mathbb{N}}$ étant sommable, elle converge vers 0. De plus, l'équation (5.31) implique que

$$(\forall k \in \mathbb{N}) \quad \|\mathbf{t}_k\| \leqslant 2^{K/2}(\beta^2 + \alpha_2^2\overline{\nu})^{1/2}\|\boldsymbol{\chi}_{k-K}\|,$$

donc $(\mathbf{x}_k, \mathbf{t}_k)_{k\in\mathbb{N}}$ converge vers $(\widehat{\mathbf{x}}, \mathbf{0})$. D'autre part, la remarque 5.2(iii) assure que la restriction de f à son domaine de définition est continue. Ainsi, puisque, pour tout $k \in \mathbb{N}$, $\mathbf{x}_k \in \operatorname{dom}\mathsf{f}$, la suite $(\mathsf{f}(\mathbf{x}_k))_{k\in\mathbb{N}}$ converge vers $\mathsf{f}(\widehat{\mathbf{x}})$. Finalement, en utilisant la propriété de fermeture de $\partial\mathsf{f}$ (c.f. remarque 2.4), on peut conclure que $(\widehat{\mathbf{x}}, \mathbf{0}) \in \operatorname{Graph} \partial\mathsf{f}$ i.e., $\widehat{\mathbf{x}}$ est un point critique de f.

5.B Démonstration du théorème 5.2

Nous utiliserons dans cette démonstration les mêmes notations que dans la démonstration du théorème 5.1.

Soit K donné par la définition 5.1(ii). Pour tout $k \in \mathbb{N}$, il existe $m \in \mathbb{N}$ et $l \in \{0, \ldots, K-1\}$ tels que $k = mK + l$. D'après l'inégalité triangulaire, on a alors

$$\|\mathbf{x}_k - \widehat{\mathbf{x}}\| \leqslant \|\mathbf{x}_{mK} - \widehat{\mathbf{x}}\| + \|\mathbf{x}_k - \mathbf{x}_{mK}\|. \tag{5.33}$$

En utilisant une nouvelle fois l'inégalité triangulaire, on obtient

$$\begin{aligned}\|\mathbf{x}_{mK}-\widehat{\mathbf{x}}\| &= \|\sum_{p=m}^{+\infty}\Big(\mathbf{x}_{(p+1)K}-\mathbf{x}_{pK}\Big)\| \\ &= \|\sum_{p=m}^{+\infty}\sum_{l'=0}^{K-1}\left(\mathbf{x}_{pK+l'+1}-\mathbf{x}_{pK+l'}\right)\| \\ &\leqslant \sum_{p=m}^{+\infty}\|\sum_{l'=0}^{K-1}\left(\mathbf{x}_{pK+l'+1}-\mathbf{x}_{pK+l'}\right)\|, \end{aligned} \tag{5.34}$$

et selon l'inégalité de Jensen et (5.14),

$$(\forall p\geqslant m)\quad \|\sum_{l'=0}^{K-1}\left(\mathbf{x}_{pK+l'+1}-\mathbf{x}_{pK+l'}\right)\|^2\leqslant K\|\boldsymbol{\chi}_{pK}\|^2. \tag{5.35}$$

Pour tout $m'\in\mathbb{N}$, soit $\Lambda_{m'}=\sum_{p=m'}^{+\infty}\|\boldsymbol{\chi}_{pK}\|$, qui est une somme finie d'après le théorème 5.1. Par conséquent, en combinant les deux inégalités ci-dessus, on obtient

$$\|\mathbf{x}_{mK}-\widehat{\mathbf{x}}\|\leqslant\sqrt{K}\Lambda_m. \tag{5.36}$$

En appliquant de nouveau l'inégalité de Jensen, on a

$$\begin{aligned}\|\mathbf{x}_{mK}-\mathbf{x}_k\|^2 &= \|\sum_{l'=0}^{l-1}\left(\mathbf{x}_{mK+l'+1}-\mathbf{x}_{mK+l'}\right)\|^2 \\ &\leqslant l\sum_{l'=0}^{l-1}\left\|\mathbf{x}_{mK+l'+1}-\mathbf{x}_{mK+l'}\right\|^2\leqslant(K-1)\|\boldsymbol{\chi}_{mK}\|^2. \end{aligned} \tag{5.37}$$

En combinant les équations (5.33), (5.36), et (5.37), on obtient alors

$$(\forall k\in\mathbb{N})\quad \|\mathbf{x}_k-\widehat{\mathbf{x}}\|\leqslant\sqrt{K}\Lambda_m+\sqrt{K-1}\|\boldsymbol{\chi}_{mK}\|\leqslant 2\sqrt{K}\Lambda_m. \tag{5.38}$$

D'après l'équation (5.32), on a, pour tout $m\geqslant\max\{k^*/K,1\}$,

$$\|\boldsymbol{\chi}_{mK}\|\leqslant\frac{1}{2}\|\boldsymbol{\chi}_{(m-1)K}\|+\mu^{-1}(1-\theta)^{-1}\kappa^{-1}2^{K/2}(\beta^2+\alpha_2^2\overline{\nu})^{1/2}\Delta_{mK},$$

où $\Delta_{mK}=\big(\mathsf{f}(\mathbf{x}_{mK})-\mathsf{f}(\widehat{\mathbf{x}})\big)^{1-\theta}-\big(\mathsf{f}(\mathbf{x}_{(m+1)K})-\mathsf{f}(\widehat{\mathbf{x}})\big)^{1-\theta}$. Ainsi, puisque $\big(\mathsf{f}(\mathbf{x}_k)-\mathsf{f}(\widehat{\mathbf{x}})\big)_{k\in\mathbb{N}}$ est une suite positive qui converge vers 0, on obtient

$$\Lambda_m\leqslant(\Lambda_{m-1}-\Lambda_m)+2\mu^{-1}(1-\theta)^{-1}\kappa^{-1}2^{K/2}(\beta^2+\alpha_2^2\overline{\nu})^{1/2}\big(\mathsf{f}(\mathbf{x}_{mK})-\mathsf{f}(\widehat{\mathbf{x}})\big)^{1-\theta}.$$

Supposons maintenant que $\theta\neq 0$. Selon les équations (5.29) et (5.31), on a

$$\kappa\big(\mathsf{f}(\mathbf{x}_{mK})-\mathsf{f}(\widehat{\mathbf{x}})\big)^{\theta}\leqslant\big(2^K(\beta^2+\alpha_2^2\overline{\nu})\big)^{1/2}\|\boldsymbol{\chi}_{(m-1)K}\|,$$

ce qui implique

$$\big(\mathsf{f}(\mathbf{x}_{mK})-\mathsf{f}(\widehat{\mathbf{x}})\big)^{1-\theta}\leqslant\kappa^{-\frac{1-\theta}{\theta}}\big(2^K(\beta^2+\alpha_2^2\overline{\nu})\big)^{\frac{1-\theta}{2\theta}}\|\boldsymbol{\chi}_{(m-1)K}\|^{\frac{1-\theta}{\theta}}. \tag{5.39}$$

Donc, en posant

$$C = 2\mu^{-1}(1-\theta)^{-1}\kappa^{-\frac{1}{\theta}}\big(2^K(\beta^2+\alpha_2^2\overline{\nu})\big)^{\frac{1}{2\theta}}, \tag{5.40}$$

nous obtenons, pour tout $m \geqslant \max\{k^*/K, 1\}$,

$$\Lambda_m \leqslant (\Lambda_{m-1} - \Lambda_m) + C\|\boldsymbol{\chi}_{(m-1)K}\|^{\frac{1-\theta}{\theta}}.$$

Ainsi, (5.19) est satisfaite.

Par conséquent, d'après le lemme 5.5 et l'équation (5.38), si $\theta \in]\frac{1}{2}, 1[$, il existe $\lambda \in]0, +\infty[$ tel que

$$(\forall k > K) \quad \|\mathbf{x}_k - \widehat{\mathbf{x}}\| \leqslant 2\sqrt{K}\lambda m^{-\frac{1-\theta}{2\theta-1}} \leqslant 2\sqrt{K}\lambda\Big(\frac{k}{K}-1\Big)^{-\frac{1-\theta}{2\theta-1}},$$

où m correspond à la partie entière inférieure de k/K. L'inégalité (5.20) est alors obtenue en posant $\lambda' = 2\sqrt{K}\lambda$. De façon analogue, si $\theta \in]0, \frac{1}{2}]$, alors il existe $\lambda \in]0, +\infty[$ et $\tau \in [0, 1[$ tels que

$$(\forall k > K) \quad \|\mathbf{x}_k - \widehat{\mathbf{x}}\| \leqslant 2\sqrt{K}\lambda\tau^m \leqslant 2\sqrt{K}\lambda\tau^{k/K-1}.$$

Ainsi, si $\tau \neq 0$, (5.22) est satisfaite en posant $\lambda' = 2\sqrt{K}\lambda/\tau$ et $\tau' = \tau^{1/K}$, tandis que si $\tau = 0$, (5.22) est vérifiée de façon immédiate.

De plus, comme $\big(\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}})\big)_{k\in\mathbb{N}}$ est une suite décroissante, pour tout $k \in \mathbb{N}$,

$$\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}}) \leqslant \mathsf{f}(\mathbf{x}_{mK}) - \mathsf{f}(\widehat{\mathbf{x}}),$$

où m correspond à la partie entière inférieure de k/K. D'après (5.39), si $m \geqslant \max\{k^*/K, 1\}$, alors

$$\begin{aligned}\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}}) &\leqslant \kappa^{-1/\theta}\left(2^K(\beta^2+\alpha_2^2\overline{\nu})\right)^{\frac{1}{2\theta}}\|\boldsymbol{\chi}_{(m-1)K}\|^{1/\theta}\\ &\leqslant \kappa^{-1/\theta}\left(2^K(\beta^2+\alpha_2^2\overline{\nu})\right)^{\frac{1}{2\theta}}\Lambda_{m-1}^{1/\theta}.\end{aligned}$$

Ainsi, si $\theta \in]\frac{1}{2}, 1[$, en appliquant de nouveau le lemme 5.5, il existe $\lambda \in]0, +\infty[$ tel que, pour tout $m > 2$,

$$\begin{aligned}\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}}) &\leqslant \kappa^{-1/\theta}\left(2^K(\beta^2+\alpha_2^2\overline{\nu})\right)^{\frac{1}{2\theta}}\lambda(m-1)^{-\frac{1-\theta}{\theta(2\theta-1)}}\\ &\leqslant \kappa^{-1/\theta}\left(2^K(\beta^2+\alpha_2^2\overline{\nu})\right)^{\frac{1}{2\theta}}\lambda\Big(\frac{k}{K}-2\Big)^{-\frac{1-\theta}{\theta(2\theta-1)}}.\end{aligned}$$

Par conséquent, on peut trouver $\lambda'' \in]0, +\infty[$ tel que, pour tout $k > 2K$, (5.21) soit vérifiée. D'autre part, d'après le lemme 5.5, si $\theta \in]0, \frac{1}{2}]$, il existe $\lambda \in]0, +\infty[$ et $\tau \in [0, 1[$ tels que

$$\begin{aligned}\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}}) &\leqslant \kappa^{-1/\theta}\left(2^K(\beta^2+\alpha_2^2\overline{\nu})\right)^{\frac{1}{2\theta}}\lambda\tau^{\frac{m-1}{\theta}}\\ &\leqslant \kappa^{-1/\theta}\left(2^K(\beta^2+\alpha_2^2\overline{\nu})\right)^{\frac{1}{2\theta}}\lambda\tau^{\frac{k/K-2}{\theta}}.\end{aligned}$$

On peut donc trouver $\lambda'' \in]0, +\infty[$ tel que, pour tout $k \in \mathbb{N}$, (5.23) soit vérifiée.

Nous allons maintenant démontrer la propriété (iii) en supposant que $\theta = 0$. Posons $\mathcal{L} = \{k \in \mathbb{N} \,|\, \mathbf{x}_k \neq \widehat{\mathbf{x}}\}$. Soit $k \in \mathcal{L}$ tel que $k \geqslant \max\{k^*, K\}$. D'après les lemmes 5.1 et 5.2,

$$\mathsf{f}(\mathbf{x}_{k+1}) \leqslant \mathsf{f}(\mathbf{x}_k) - \frac{\mu}{2}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 \leqslant \mathsf{f}(\mathbf{x}_{k-K}) - \frac{\mu}{2}\|\boldsymbol{\chi}_{k-K}\|^2.$$

En utilisant l'équation (5.31), on obtient alors

$$\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}}) - \frac{\mu}{2}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 \leqslant \mathsf{f}(\mathbf{x}_{k-K}) - \mathsf{f}(\widehat{\mathbf{x}}) - \frac{\mu'}{2}\|\mathbf{t}_k\|^2,$$

où $\mu' \in]0, +\infty[$. Puisque $\theta = 0$, en combinant l'inégalité ci-dessus avec (5.29), on a

$$\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}}) - \frac{\mu}{2}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 \leqslant \mathsf{f}(\mathbf{x}_{k-K}) - \mathsf{f}(\widehat{\mathbf{x}}) - \frac{\mu'}{2}\kappa^2|\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}})|^0,$$

c'est-à-dire,

$$\mathsf{f}(\mathbf{x}_k) - \mathsf{f}(\widehat{\mathbf{x}}) - \frac{\mu}{2}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2 \leqslant \mathsf{f}(\mathbf{x}_{k-K}) - \mathsf{f}(\widehat{\mathbf{x}}) - \frac{\mu'}{2}\kappa^2.$$

Comme $\lim_{k\to+\infty} \mathsf{f}(\mathbf{x}_k) = \mathsf{f}(\widehat{\mathbf{x}})$, cette dernière inégalité implique que $\mathcal{L}$ est un ensemble fini. La propriété (iii) en découle alors de façon immédiate.

Chapitre 6

Quelques applications de l'algorithme explicite-implicite à métrique variable alterné par bloc

Sommaire

6.1 Introduction

Nous avons proposé dans le chapitre précédent un algorithme permettant de minimiser la somme d'une fonction différentiable de gradient Lipschitz et d'une fonction séparable par bloc. Cette méthode combine une technique d'accélération utilisant une métrique variable et un principe de minimisation alternée par bloc. Nous allons maintenant appliquer cet algorithme à la résolution de problèmes inverses faisant intervenir un critère non lisse et non convexe, afin d'étudier son comportement pratique. En particulier nous allons montrer l'utilité de la métrique variable pour accélérer la vitesse de convergence de la méthode. Nous allons également étudier l'influence du choix de la taille des blocs ainsi que de l'ordre de leur mise à jour.

Pour cela, nous allons considérer trois exemples applicatifs différents. La première application présentée dans la section 6.2, est un problème de démélange spectral. Ensuite, la section 6.3 traitera d'une application en reconstruction de phase. Enfin, dans la section 6.4, nous présenterons un exemple de déconvolution aveugle de signaux sismiques. Nous conclurons dans la section 6.5.

6.2 Problème de démélange spectral

6.2.1 Formulation du problème

Considérons un ensemble de données hyperspectrales, noté $\mathbf{Y} \in \mathbb{R}^{S\times M}$, modélisant un ensemble d'images de $\mathbb{R}^M$ (réorganisées en colonnes avec $M = M_1 \times M_2$), acquises en S différentes bandes spectrales. On suppose que l'on observe le modèle de mélange linéaire suivant :

$$\mathbf{Y} = \overline{\mathbf{U}}\,\overline{\mathbf{V}} + \mathbf{E}, \tag{6.1}$$

où les colonnes de $\overline{\mathbf{U}} \in \mathbb{R}^{S\times P}$ représentent les spectres des P composantes distinctes présentes dans l'image, $\overline{\mathbf{V}} \in \mathbb{R}^{P\times M}$ leurs proportions respectives (abondances) en chaque pixel, et $\mathbf{E} \in \mathbb{R}^{S\times M}$ le bruit d'acquisition. Ce type de modèle se rencontre dans diverses applications telles que la télédétection [Asner et Heidebrecht, 2002], la planétologie [Themelis *et al.*, 2012], le contrôle alimentaire [Gowen *et al.*, 2007], ou la spectromicroscopie [Dobigeon et Brun, 2012]. Une illustration de ce modèle est donnée dans la figure 6.1. L'objectif du problème de démélange spectral est d'estimer $\overline{\mathbf{U}}$ et $\overline{\mathbf{V}}$ à partir des données observées $\mathbf{Y}$ [Bioucas-Dias *et al.*, 2012].

Les méthodes standards de démélange spectral reposent sur des approches supervisées, au sens où les spectres composant $\overline{\mathbf{U}}$ sont supposés être issus d'un dictionnaire connu ou bien avoir été préalablement estimés par un algorithme d'extraction de spectres [Bioucas-Dias *et al.*, 2012; Chouzenoux *et al.*, 2014a; Keshava et Mustard, 2002; Ma *et al.*, 2014; Moussaoui *et al.*, 2012]. Cependant, il existe un intérêt croissant pour des méthodes permettant d'estimer conjointement $\overline{\mathbf{U}}$ et $\overline{\mathbf{V}}$ basées sur la factorisation de matrices non-négative (*Nonnegative Matrix Factorization* (NMF)) [Ma *et al.*, 2014; Tomazeli Duarte *et al.*, 2014].

Dans l'approche standard de NMF, les estimations de $\overline{\mathbf{U}}$ et $\overline{\mathbf{V}}$ sont définies comme étant des solutions du problème de minimisation du critère de moindres carrés associé au modèle (6.1) sous

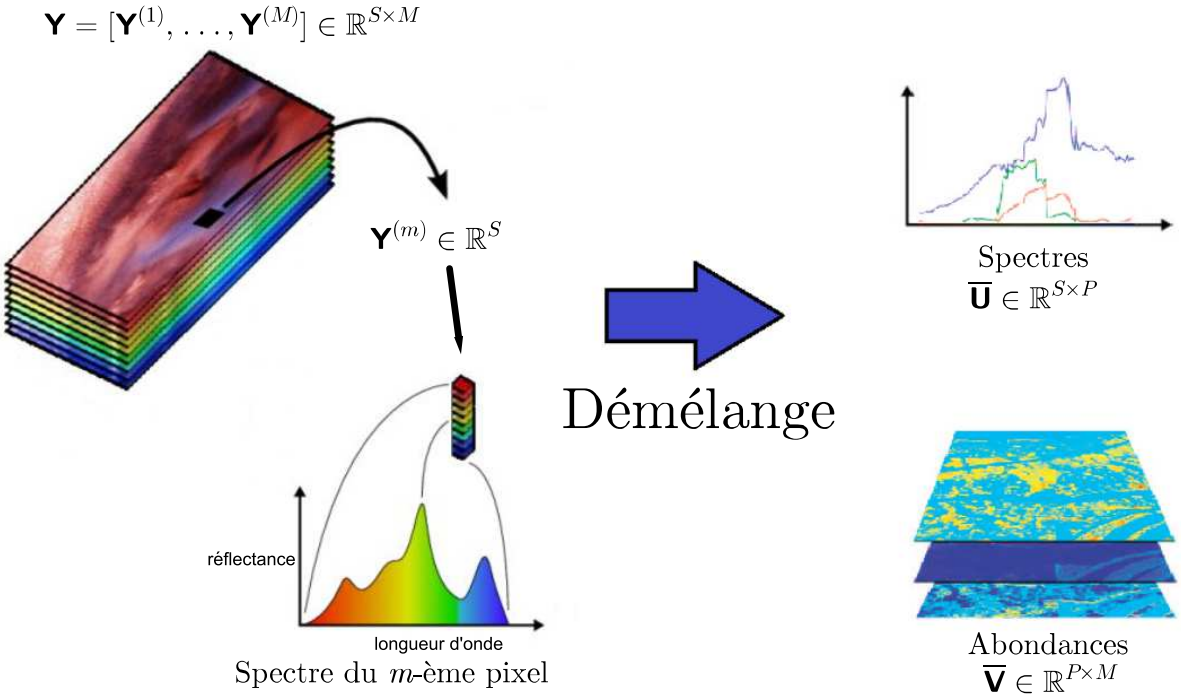


Figure 6.1 – *Illustration du problème de démélange spectral. L'objectif est d'estimer* $\overline{\mathbf{U}}$ *et* $\overline{\mathbf{V}}$ *à partir d'un ensemble de données hyperspectrales* $\mathbf{Y}$.

des contraintes de positivité sur les éléments des deux matrices [Paatero et Tapper, 1994] :

$$\text{trouver } (\widehat{\mathbf{U}}, \widehat{\mathbf{V}}) \in \underset{(\mathbf{U},\mathbf{V}) \in \mathbb{R}^{S \times P} \times \mathbb{R}^{P \times M}}{\operatorname{Argmin}} \left\{ \frac{1}{2} \|\mathbf{UV} - \mathbf{Y}\|^2 + \iota_{[0,+\infty[^{S \times P}}(\mathbf{U}) + \iota_{[0,+\infty[^{P \times M}}(\mathbf{V}) \right\}. \text{ [1]} \quad (6.2)$$

Le critère à minimiser n'est pas convexe et présente des minimas locaux [Lee et Seung, 2001], ainsi, les contraintes de positivité sur les matrices $\overline{\mathbf{U}}$ et $\overline{\mathbf{V}}$ ne sont pas toujours suffisantes pour obtenir des résultats corrects [Chu *et al.*, 2004]. Cependant, comme il est souligné dans [Jia et Qian, 2009; Ma *et al.*, 2014], une nette amélioration de leurs qualités peut être obtenue lorsque des connaissances *a priori* sur les matrices $\overline{\mathbf{U}}$ et $\overline{\mathbf{V}}$ sont introduites dans le problème de minimisation (6.2).

Dans cette section, nous nous focaliserons sur le cas où chaque spectre composant les colonnes de $\overline{\mathbf{U}}$ est une combinaison linéaire de quelques spectres de référence issus d'un grand dictionnaire de taille $Q > P$, que nous noterons $\mathbf{\Omega} \in \mathbb{R}^{S \times Q}$. Ainsi, le modèle d'observation (6.1) est ré-exprimé de la façon suivante :

$$\mathbf{Y} = \mathbf{\Omega}\,\overline{\mathbf{T}}\,\overline{\mathbf{V}} + \mathbf{E}, \quad (6.3)$$

où $\overline{\mathbf{T}} \in \mathbb{R}^{Q \times P}$ est une matrice supposée parcimonieuse donnant les coefficients de la combinaison linéaire permettant d'obtenir $\overline{\mathbf{U}} = \mathbf{\Omega}\,\overline{\mathbf{T}}$.

1. Ici $\|\cdot\|$ désigne la norme de Frobenius.

Nous définissons les estimées $\widehat{\mathbf{T}}$ et $\widehat{\mathbf{V}}$ de $\overline{\mathbf{T}}$ et $\overline{\mathbf{V}}$ comme étant des solutions du problème

$$\text{trouver } (\widehat{\mathbf{T}}, \widehat{\mathbf{V}}) \in \underset{(\mathbf{T},\mathbf{V})\in\mathbb{R}^{Q\times P}\times\mathbb{R}^{P\times M}}{\operatorname{Argmin}} \mathsf{h}(\mathbf{T},\mathbf{V}) + \mathsf{g}_1(\mathbf{T}) + \mathsf{g}_2(\mathbf{V}), \tag{6.4}$$

où $\mathsf{h}\colon \mathbb{R}^{Q\times P}\times\mathbb{R}^{P\times M}\to\mathbb{R}$ correspond au critère de moindre carré associé au modèle (6.3) défini par

$$(\forall (\mathbf{T},\mathbf{V})\in\mathbb{R}^{Q\times P}\times\mathbb{R}^{P\times M})\quad \mathsf{h}(\mathbf{T},\mathbf{V}) = \frac{1}{2}\|\mathbf{\Omega}\mathbf{T}\mathbf{V}-\mathbf{Y}\|^2, \tag{6.5}$$

et la fonction $\mathsf{g}_1\colon \mathbb{R}^{Q\times P}\to]-\infty,+\infty]$ (resp. $\mathsf{g}_2\colon \mathbb{R}^{P\times M}\to]-\infty,+\infty]$) est une fonction de régularisation propre et semi-continue inférieurement, incorporant des informations *a priori* sur $\widehat{\mathbf{T}}$ (resp. $\widehat{\mathbf{V}}$).

Afin de promouvoir la parcimonie de $\widehat{\mathbf{T}}$, nous proposons de choisir

$$(\forall\, \mathbf{T}=(\mathsf{T}^{(q,p)})_{1\leqslant q\leqslant Q,1\leqslant p\leqslant P}\in\mathbb{R}^{Q\times P})$$
$$\mathsf{g}_1(\mathbf{T}) = \sum_{q=1}^{Q}\sum_{p=1}^{P}\Big(\iota_{[\mathsf{T}_{\min},\mathsf{T}_{\max}]}(\mathsf{T}^{(q,p)}) + \phi_{\delta,\beta}(\mathsf{T}^{(q,p)})\Big), \tag{6.6}$$

où $(\delta,\mathsf{T}_{\min},\mathsf{T}_{\max})\in]0,+\infty[^3$ et $\phi_{\delta,\beta}$ est la fonction de régularisation donnée dans la proposition 2.12 [Chartrand, 2012] de paramètre $\beta\in]0,1]$. Rappelons que cette fonction est convexe si et seulement si $\beta=1$, i.e. lorsque $\phi_{\delta,\beta}$ correspond à la fonction valeur absolue. D'autre part, on définit

$$\mathsf{g}_2 = \iota_{\mathcal{V}}, \tag{6.7}$$

où $\mathcal{V}\subset\mathbb{R}^{P\times M}$ est donné par

$$\mathcal{V} = \Big\{\mathbf{V}=(\mathsf{V}^{(p,m)})_{1\leqslant p\leqslant P,1\leqslant m\leqslant M}\in\mathbb{R}^{P\times M}\Big|(\forall m\in\{1,\dots,M\})\quad \sum_{p=1}^{P}\mathsf{V}^{(p,m)}=1,$$
$$(\forall p\in\{1,\dots,P\})(\forall m\in\{1,\dots,M\})\quad \mathsf{V}^{(p,m)}\geqslant \mathsf{V}_{\min}\Big\},$$

avec $\mathsf{V}_{\min}>0$.

6.2.2 Mise en œuvre de l'algorithme

Nous proposons d'appliquer l'algorithme 5.4, donné dans le chapitre 5, au problème de minimisation (6.4). Pour cela, il nous faut définir, pour tout $(\widetilde{\mathbf{T}},\widetilde{\mathbf{V}})\in\operatorname{dom}\mathsf{g}_1\times\operatorname{dom}\mathsf{g}_2$, des fonctions quadratiques $\mathsf{q}_1(\cdot\,|\,\widetilde{\mathbf{T}},\widetilde{\mathbf{V}})$ et $\mathsf{q}_2(\cdot\,|\,\widetilde{\mathbf{T}},\widetilde{\mathbf{V}})$ majorant, respectivement, $\mathsf{h}(\cdot,\widetilde{\mathbf{V}})$ en $\widetilde{\mathbf{T}}$ et $\mathsf{h}(\widetilde{\mathbf{T}},\cdot)$ en $\widetilde{\mathbf{V}}$. D'après (6.6) et (6.7), les éléments des matrices $\widetilde{\mathbf{T}}$ et $\widetilde{\mathbf{V}}$ sont strictement positifs. Ainsi, on peut déduire des fonctions quadratiques majorantes proposées dans [Lee et Seung, 2001] dans le contexte de la NMF, les fonctions quadratiques majorantes données par la proposition suivante :

Proposition 6.1. [Lee et Seung, 2001] *Soit* $(\widetilde{\mathbf{T}},\widetilde{\mathbf{V}})\in\operatorname{dom}\mathsf{g}_1\times\operatorname{dom}\mathsf{g}_2$. *Une fonction quadratique majorant* $\mathsf{h}(\cdot,\widetilde{\mathbf{V}})$ *en* $\widetilde{\mathbf{T}}$ *est donnée par*

$$(\forall\,\mathbf{T}\in\mathbb{R}^{Q\times P})\quad \mathsf{q}_1(\mathbf{T}\,|\,\widetilde{\mathbf{T}},\widetilde{\mathbf{V}}) = \mathsf{h}(\widetilde{\mathbf{T}},\widetilde{\mathbf{V}}) + \operatorname{Tr}\Big((\mathbf{T}-\widetilde{\mathbf{T}})\nabla_1\mathsf{h}(\widetilde{\mathbf{T}},\widetilde{\mathbf{V}})^\top\Big)$$
$$+\frac{1}{2}\operatorname{Tr}\Big(((\mathbf{T}-\widetilde{\mathbf{T}})\odot\mathbf{A}_1(\widetilde{\mathbf{T}},\widetilde{\mathbf{V}}))(\mathbf{T}-\widetilde{\mathbf{T}})^\top\Big),$$

et une fonction quadratique majorant $\mathsf{h}(\widetilde{\mathbf{T}}, \cdot)$ *en* $\widetilde{\mathbf{V}}$ *est donnée par*

$$(\forall \mathbf{V} \in \mathbb{R}^{P\times M}) \quad \mathsf{q}_2(\mathbf{V} \,|\, \widetilde{\mathbf{T}}, \widetilde{\mathbf{V}}) = \mathsf{h}(\widetilde{\mathbf{T}}, \widetilde{\mathbf{V}}) + \operatorname{Tr}\left((\mathbf{V}-\widetilde{\mathbf{V}})\nabla_2\mathsf{h}(\widetilde{\mathbf{T}}, \widetilde{\mathbf{V}})^\top\right) + \frac{1}{2}\operatorname{Tr}\left(((\mathbf{V}-\widetilde{\mathbf{V}}) \odot \mathbf{A}_2(\widetilde{\mathbf{T}}, \widetilde{\mathbf{V}}))(\mathbf{V}-\widetilde{\mathbf{V}})^\top\right),$$

avec

$$\mathbf{A}_1(\widetilde{\mathbf{T}}, \widetilde{\mathbf{V}}) = \left((\mathbf{\Omega}^\top\mathbf{\Omega})\widetilde{\mathbf{T}}(\widetilde{\mathbf{V}}\widetilde{\mathbf{V}}^\top)\right) \oslash \widetilde{\mathbf{T}}, \tag{6.8}$$

$$\mathbf{A}_2(\widetilde{\mathbf{T}}, \widetilde{\mathbf{V}}) = \left((\mathbf{\Omega}\widetilde{\mathbf{T}})^\top\mathbf{\Omega}\widetilde{\mathbf{T}}\widetilde{\mathbf{V}}\right) \oslash \widetilde{\mathbf{V}}, \tag{6.9}$$

où $\operatorname{Tr}(\cdot)$ *désigne l'opérateur trace, et* $\odot$ *(resp.* $\oslash$*) le produit (resp. division) de Hadamard entre deux matrices de même taille.*

Dans ce cas particulier, l'algorithme 5.4 se réduit à l'algorithme 6.1.

Algorithme 6.1 Algorithme explicite-implicite à métrique variable alterné par bloc appliqué au problème de NMF

Initialisation : Soient $\mathbf{T}_0 \in \operatorname{dom}\mathsf{g}_1$, $\mathbf{V}_0 \in \operatorname{dom}\mathsf{g}_2$, et, pour tout $k \in \mathbb{N}$, $\gamma_k \in]0, +\infty[$, $\mathbf{A}_1(\mathbf{T}_k, \mathbf{V}_k)$ et $\mathbf{A}_2(\mathbf{T}_{k+1}, \mathbf{V}_k)$ définies par (6.8) et (6.9).

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{aligned}
&\widetilde{\mathbf{T}}_k = \mathbf{T}_k - \gamma_k\nabla_1\mathsf{h}(\mathbf{T}_k, \mathbf{V}_k) \oslash \mathbf{A}_1(\mathbf{T}_k, \mathbf{V}_k),\\
&\mathbf{T}_{k+1} = \operatorname{prox}_{\gamma_k^{-1}\mathbf{A}_1(\mathbf{T}_k,\mathbf{V}_k),\mathsf{g}_1}\left(\widetilde{\mathbf{T}}_k\right),\\
&\widetilde{\mathbf{V}}_k = \mathbf{V}_k - \gamma_k\nabla_2\mathsf{h}(\mathbf{T}_k, \mathbf{V}_k) \oslash \mathbf{A}_2(\mathbf{T}_{k+1}, \mathbf{V}_k),\\
&\mathbf{V}_{k+1} = \operatorname{prox}_{\gamma_k^{-1}\mathbf{A}_2(\mathbf{T}_{k+1},\mathbf{V}_k),\mathsf{g}_2}\left(\widetilde{\mathbf{V}}_k\right).
\end{aligned}$$

Dans l'algorithme 6.1, pour tout $k \in \mathbb{N}$, les matrices de préconditionnement $\mathbf{A}_1(\mathbf{T}_k, \mathbf{V}_k)$ et $\mathbf{A}_2(\mathbf{T}_{k+1}, \mathbf{V}_k)$ sont appliquées terme à terme. Ainsi, si les matrices $(\mathbf{T}_k)_{k\in\mathbb{N}}$ et $(\mathbf{V}_k)_{k\in\mathbb{N}}$ étaient réorganisées en vecteurs colonnes, les matrices $\mathbf{A}_1(\mathbf{T}_k, \mathbf{V}_k)$ et $\mathbf{A}_2(\mathbf{T}_{k+1}, \mathbf{V}_k)$ seraient des matrices diagonales. La fonction g_1 étant additivement séparable, son opérateur proximal (voir section 2.3.4.2 propriété (vii)) est donné par

$$(\forall k \in \mathbb{N}) \quad \mathbf{T}_{k+1} = (\mathsf{T}_{k+1}^{(q,p)})_{1\leqslant q\leqslant Q, 1\leqslant p\leqslant P} \tag{6.10}$$

avec

$$(\forall (q,p) \in \{1,\ldots,P\}\times\{1,\ldots,Q\}) \quad \mathsf{T}_{k+1}^{(q,p)} = \Pi_{[\mathsf{T}_{\min},\mathsf{T}_{\max}]}\left(\operatorname{prox}_{\gamma_k^{-1}\mathbf{A}_1(\mathbf{T}_k,\mathbf{V}_k)^{(q,p)},\phi_{\delta,\beta}}\left(\widetilde{\mathsf{T}}_k^{(q,p)}\right)\right), \tag{6.11}$$

où $\Pi_{[\mathsf{T}_{\min},\mathsf{T}_{\max}]}$ désigne l'opérateur de projection dans l'ensemble $[\mathsf{T}_{\min}, \mathsf{T}_{\max}]$, et l'opérateur proximal de la fonction $\phi_{\delta,\beta}$ est donné dans la proposition 2.12 (chapitre 2). D'autre part, pour tout $k \in \mathbb{N}$, l'opérateur proximal donnant $\mathbf{V}_{k+1}$ dans l'algorithme 6.1 correspond à la projection dans l'ensemble $\mathcal{V}$ relativement à la métrique $\gamma_k^{-1}\mathbf{A}_2(\mathbf{T}_{k+1}, \mathbf{V}_k)$. Cette projection peut être calculée de façon exacte grâce à l'algorithme itératif donné dans [Michelot, 1986].

6.2.3 Résultats de simulation

Afin de simuler des données hyperspectrales réalistes, nous composons le dictionnaire $\mathbf{\Omega}$ d'une sélection non redondante de $Q = 62$ spectres de $S = 224$ bandes spectrales, allant de 383 nm à 2508 nm, à partir de la bibliothèque disponible sur `http://www.lx.it.pt/~bioucas` (U.S. Geological Survey library [Clark *et al.*, 1993]). La matrice $\overline{\mathbf{U}}$ est composée de $P = 5$ spectres distincts résultant d'une combinaison pondérée de quelques (typiquement 3) spectres purs sélectionnés de façon aléatoire parmi les colonnes de $\mathbf{\Omega}$, les poids étant stockés dans la matrice $\overline{\mathbf{T}}$. Chaque

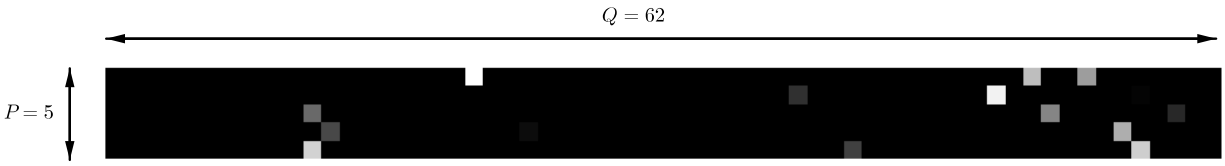


Figure 6.2 – *Transposée de la matrice $\overline{\mathbf{T}}$. Chaque ligne $p \in \{1, \dots, P\}$ contient les poids de la combinaison pondérée formant le p-ème spectre de $\overline{\mathbf{U}}$. Les carrés noirs correspondent aux coefficients nuls.*

ligne de la matrice d'abondance $\overline{\mathbf{V}}$ est construite comme étant la superposition de gaussiennes 2D composées de $M = 128 \times 128$ pixels, d'espérances et de variances aléatoires, normalisées afin d'assurer la contrainte de somme à un. Finalement, on définit la matrice $\mathbf{E}$ comme une réalisation d'un vecteur aléatoire gaussien centré et dont la variance est choisie de manière à ce que le RSB (c.f. équation (2.20)) soit égal à 20 dB. La figure 6.2 représente la matrice de coefficients $\overline{\mathbf{T}}$, et les abondances $\overline{\mathbf{V}}$ sont représentées dans la figure 6.3.

Les paramètres (δ, β) sont choisis de façon à maximiser le RSB des estimations $\widehat{\mathbf{T}}$ et $\widehat{\mathbf{V}}$. En particulier, nous avons pu observer que $\beta = 0.1$ donne les meilleurs résultats de reconstruction.

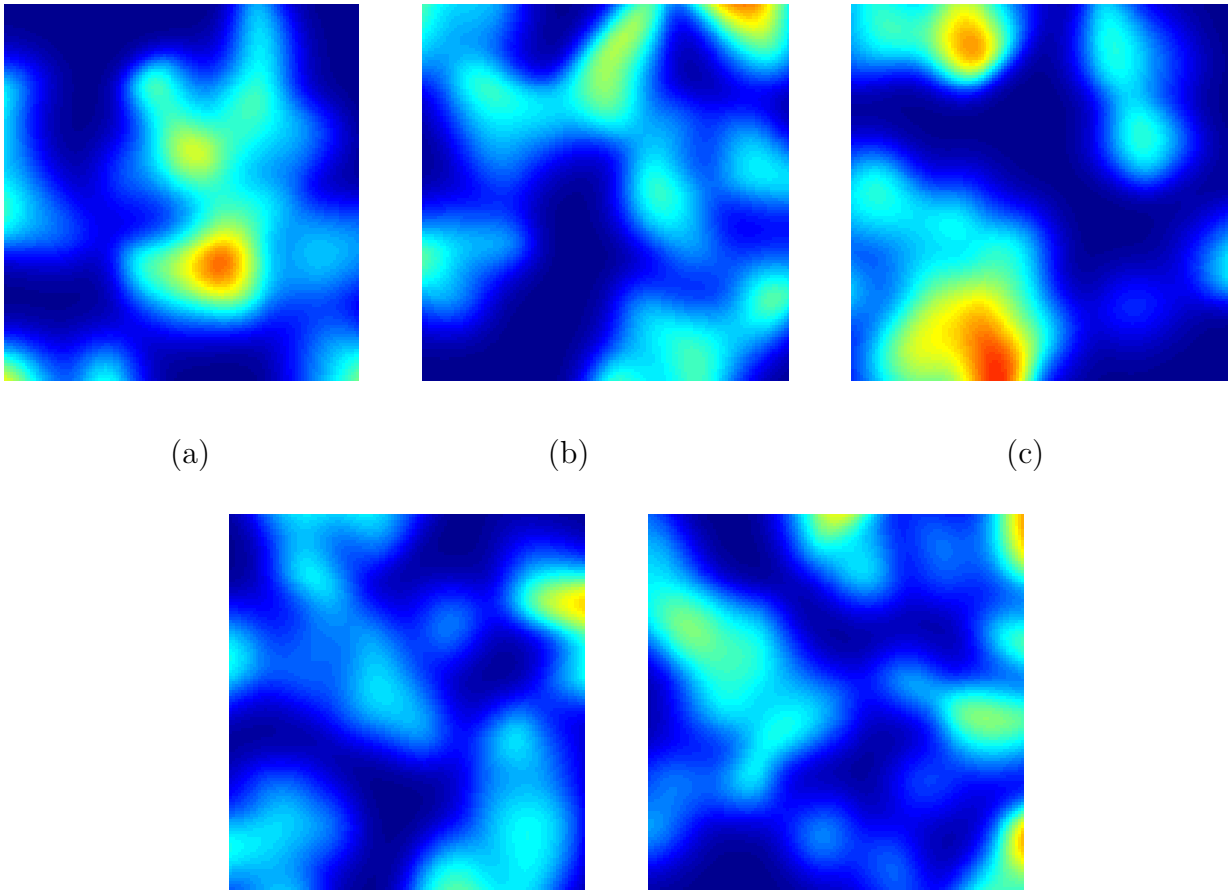
Dans la figure 6.4 nous donnons les spectres exacts $\overline{\mathbf{U}} = \mathbf{\Omega}\overline{\mathbf{T}}$ et reconstruits $\widehat{\mathbf{U}} = \mathbf{\Omega}\widehat{\mathbf{T}}$.

Dans cet exemple, la norme résiduelle $r(\widehat{\mathbf{T}})$ entre $\widehat{\mathbf{T}}$ et $\overline{\mathbf{T}}$ (resp. $r(\widehat{\mathbf{V}})$ entre $\widehat{\mathbf{V}}$ et $\overline{\mathbf{V}}$) est égal à 1.1×10^{-3} (resp. 7.5×10^{-5}). En comparaison, en appliquant la méthode de NMF proposée dans [Lee et Seung, 2001] au modèle (6.1) on obtient $r(\widehat{\mathbf{V}}) = 5 \times 10^{-4}$ (après renormalisation des colonnes de $\widehat{\mathbf{V}}$). La figure 6.5 illustre les variations de $(r(\mathbf{T}_k))_k$ en fonction du temps de calcul obtenues avec l'algorithme 6.1 pour $\gamma_k \equiv 0.99$, ou avec l'algorithme PALM proposé dans [Bolte *et al.*, 2013]. Pour obtenir l'algorithme PALM, on remplace, pour tout $k \in \mathbb{N}$, dans l'algorithme 6.1 les matrices $\mathbf{A}_1(\mathbf{T}_k, \mathbf{V}_k)$ et $\mathbf{A}_2(\mathbf{T}_{k+1}, \mathbf{V}_k)$ par

$$\mathbf{A}_1(\mathbf{T}_k, \mathbf{V}_k) = \|\mathbf{\Omega}\|^2 \|\mathbf{V}_k\|^2 \mathbf{1}_{Q \times P}, \tag{6.12}$$

$$\mathbf{A}_2(\mathbf{T}_{k+1}, \mathbf{V}_k) = \|\mathbf{\Omega}\mathbf{T}_{k+1}\|^2 \mathbf{1}_{P \times M}, \tag{6.13}$$

où, pour tout $(N, M) \in \mathbb{N}^* \times \mathbb{N}^*$, $\mathbf{1}_{N \times M}$ désigne la matrice de dimension $N \times M$ dont tous les coefficients sont égaux à 1. Nous pouvons observer que, dans cet exemple, la métrique variable permet d'accélérer la vitesse de convergence de l'algorithme 6.1 en terme de décroissance de l'erreur résiduelle.



(a) (b) (c)

(d) (e)

Figure 6.3 – *Cartes d'abondance* $\overline{\mathbf{V}}$. *Image correspondant à la première ligne* (a), *deuxième ligne* (b), *troisième ligne* (c), *quatrième ligne* (d) *et cinquième ligne* (e) *de la matrice* $\overline{\mathbf{V}}$.

6.3 Reconstruction de phase

6.3.1 Formulation du problème

6.3.1.1 État de l'art

Bien que la reconstruction de phase ne soit pas un problème inverse nouveau, il reste l'un de ceux les plus ardus à résoudre en traitement d'images. Il consiste à produire une estimation $\widehat{\mathbf{y}}$ d'un

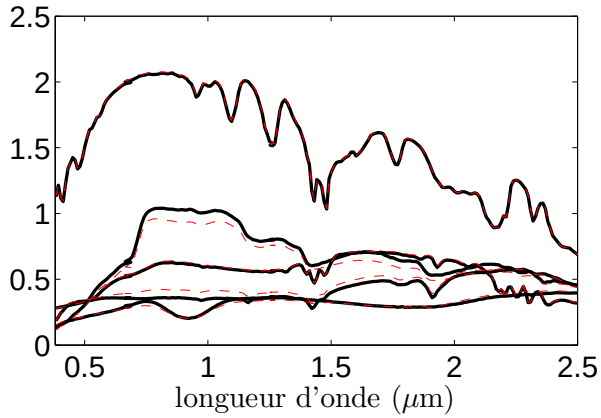


Figure 6.4 – *Spectres exacts* $\overline{\mathbf{U}} = \mathbf{\Omega}\overline{\mathbf{T}}$ *(lignes continues noires) et reconstruits* $\hat{\mathbf{U}} = \mathbf{\Omega}\hat{\mathbf{T}}$ *(lignes discontinues rouges).*

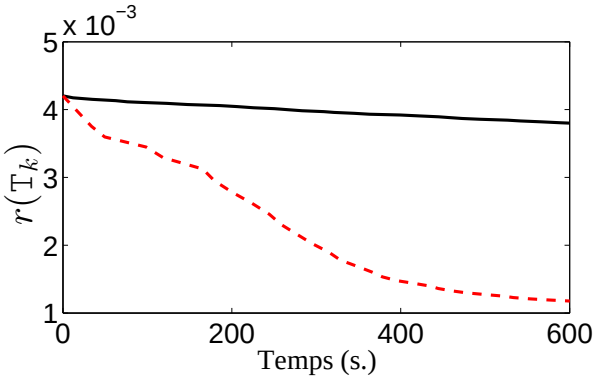


Figure 6.5 – *Comparaison de la vitesse de convergence de l'algorithme 6.1 (ligne discontinue rouge) et de l'algorithme PALM (ligne continue noire).*

signal original $\overline{\mathbf{y}}$ à partir de l'acquisition de son module dégradé $|\mathbf{H}\overline{\mathbf{y}}|$ (pouvant être bruité) où $\mathbf{H}$ est un opérateur linéaire à valeurs complexes. Un tel problème peut se rencontrer dans beaucoup de domaines d'applications en traitement d'images, en particulier en cristallographie [Harrison, 1993], en imagerie optique [Walther, 1963], en tomographie par contraste de phase [Bauschke *et al.*, 2005], et en imagerie par diffraction cohérente [Shechtman *et al.*, 2013].

Les méthodes les plus populaires permettant d'estimer $\overline{\mathbf{y}}$ sont probablement l'algorithme de *Gerchberg-Saxton* (GS) [Gerchberg et Saxton, 1972] ansi que sa version relaxée, l'algorithme de *Fienup* [Fienup, 1982]. Introduites initialement pour résoudre le cas particulier où $\mathbf{H}$ est une matrice modélisant une transformée de Fourier, ces algorithmes alternent entre une projection sur l'image de $\mathbf{H}$ et une projection sur l'ensemble non convexe des vecteurs ayant un module égal à $|\mathbf{H}\overline{\mathbf{y}}|$. Remarquons que, puisque cet ensemble est non convexe, les algorithmes de GS et Fienup ne bénéficient pas des garanties de convergence de l'algorithme POCS (*Projections Onto Convex Set*) [Bauschke *et al.*, 2002]. Des versions plus générales de ces algorithmes, plus rapides et donnant de meilleurs résultats de reconstruction, sont présentées dans [Bauschke *et al.*, 2003, 2005].

Des relaxations convexes du problème de reconstruction de phase basées sur des formulation de programmation semi-définie (*Semi-Definite-Programming*, SDP) ont été récemment proposées dans [Candès *et al.*, 2013; Waldspurger *et al.*, 2013]. Les algorithmes résultant sont appelés algorithme *PhaseLift* [Candès *et al.*, 2013] et algorithme *PhaseCut* [Waldspurger *et al.*, 2013]. Ces deux méthodes permettent d'obtenir de meilleurs résultats que les algorithmes GS et Fienup dans un certain nombre d'applications [Fogel *et al.*, 2013].

Les méthodes citées ci-dessus ont cependant tendance à ne pas être robustes lorsque les données sont bruitées et/ou à perdre en efficacité lorsque l'on traite des cas sous-déterminés [Shechtman *et al.*, 2014]. De telles difficultés sont souvent dues au caractère mal posé du problème de reconstruction de phase, et une méthode permettant de les contourner (ou les amoindrir) est d'incorporer des informations *a priori* dans le processus de reconstruction. Certains algorithmes basés sur la relaxation SDP [Jaganathan *et al.*, 2012], sur les projections alternées [Mukherjee et Seelamantula, 2012] et sur les méthodes gloutonnes [Shechtman *et al.*, 2014], ont récemment été proposées, pour résoudre le problème de reconstruction de phase sous l'hypothèse que le signal original $\overline{\mathbf{y}}$ a une représentation parcimonieuse dans un certain dictionnaire (qui peut être redondant). Cependant, à notre connaissance, ces approches peuvent rapidement devenir trop coûteuses lorsque la dimension du problème augmente. Cela est particulièrement vrai pour les méthodes SDP et de projections alternées lorsque la pseudo-inverse de $\mathbf{H}$ n'a pas de forme explicite (comme cela est souvent le cas pour des mesures qui ne sont pas faites dans le domaine de Fourier), et pour la méthode gloutonne de [Shechtman *et al.*, 2014] lorsque le degré de parcimonie des données n'est pas suffisant.

Dans cette section, nous allons introduire une nouvelle stratégie de reconstruction de phase reposant sur la minimisation d'un critère pénalisé. Ce critère est composé de *(i)* une différence de fonctions convexes réalisant une approximation lisse du terme d'attache aux données non convexe de moindres carrés usuel, et *(ii)* d'un terme de régularisation convexe non nécessairement lisse additivement séparable par bloc. Pour résoudre le problème variationnel résultant, nous utilisons l'algorithme 5.4 donné dans le chapitre 5.

6.3.1.2 Problème d'optimisation

Soient $\overline{\mathbf{y}} \in \mathbb{R}^M$ un signal original inconnu et $\mathbf{H} \in \mathbb{C}^{S \times M}$ une matrice d'observations dont les éléments sont à valeurs complexes. On suppose que les observations $\mathbf{z} \in [0, +\infty[^S$ sont liées au signal d'origine suivant le modèle :

$$\mathbf{z} = |\mathbf{H}\overline{\mathbf{y}}| + \mathbf{w}, \tag{6.14}$$

où $|\cdot|$ désigne l'opérateur de module terme à terme, et $\mathbf{w} \in [0, +\infty[^S$ est une réalisation d'un bruit additif.

Remarque 6.1. *Il est important de remarquer que le modèle* (6.14) *permet aussi de traiter le cas où le signal original* $\overline{\mathbf{y}}$ *est à valeurs complexes. En effet, dans ce cas, le vecteur d'observations s'écrit*

$$\mathbf{z} = \left| [\mathbf{H}_{\mathcal{R}} + \mathrm{i}\,\mathbf{H}_{\mathcal{I}} \mid -\mathbf{H}_{\mathcal{I}} + \mathrm{i}\,\mathbf{H}_{\mathcal{R}}] \begin{bmatrix} \overline{\mathbf{y}}_{\mathcal{R}} \\ \overline{\mathbf{y}}_{\mathcal{I}} \end{bmatrix} \right| + \mathbf{w}, \tag{6.15}$$

où $\mathrm{i}^2 = -1$ *et* $(\cdot)_{\mathcal{R}}$ *(resp.* $(\cdot)_{\mathcal{I}}$*) désigne la partie réelle (resp. imaginaire) de son argument.*

L'objectif est alors de produire une estimée $\widehat{\mathbf{y}} \in \mathbb{R}^M$ du signal original $\overline{\mathbf{y}}$ à partir des données observées $\mathbf{z}$.

Soit $\mathbf{W} \in \mathbb{R}^{M \times N}$, $M \leqslant N$, un opérateur de trame à la synthèse (par exemple, un opérateur d'ondelettes qui peut être redondant) [Mallat, 2009] tel que $\overline{\mathbf{y}} = \mathbf{W}\overline{\mathbf{x}}$, avec $\overline{\mathbf{x}} \in \mathbb{R}^N$ le vecteur contenant les coefficients de trame. En considérant une approche à la synthèse (voir section 2.2.4.2 du chapitre 2) le signal estimé s'écrit $\widehat{\mathbf{y}} = \mathbf{W}\widehat{\mathbf{x}}$ où $\widehat{\mathbf{x}} \in \mathbb{R}^N$ est obtenu en minimisant la somme d'une fonction d'attache aux données h et d'une fonction de régularisation g, i.e.

$$\widehat{\mathbf{x}} \in \underset{\mathbf{x} \in \mathbb{R}^N}{\operatorname{Argmin}} \ \{\mathsf{f}(\mathbf{x}) = \mathsf{h}(\mathbf{x}) + \mathsf{g}(\mathbf{x})\}. \tag{6.16}$$

Dans un contexte de reconstruction de phase, un choix usuel de terme d'attache aux données h est le critère non lisse et non convexe de moindres carrés [Fienup, 1982; Shechtman *et al.*, 2014] :

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}(\mathbf{x}) = \sum_{s=1}^{S} \varphi_s(|[\mathbf{H}\mathbf{W}\mathbf{x}]^{(s)}|), \tag{6.17}$$

où, pour tout $s \in \{1, \ldots, S\}$,

$$(\forall \mathsf{u} \in [0, +\infty[) \quad \varphi_s(\mathsf{u}) = \frac{1}{2}(\mathsf{u} - \mathsf{z}^{(s)})^2 = \frac{1}{2}\left(\mathsf{u}^2 + (\mathsf{z}^{(s)})^2\right) - \mathsf{u}\mathsf{z}^{(s)}. \tag{6.18}$$

Nous nous proposons de remplacer cette fonction par une approximation de façon à générer une approximation lisse de la fonction non lisse h. Nous définissons l'approximation de φ_s comme la somme d'une fonction convexe et d'une fonction concave, paramétrée par une constante $\delta > 0$:

$$(\forall \mathsf{u} \in [0, +\infty[) \quad \widehat{\varphi}_s(\mathsf{u}) = \varphi_{s,1}(\mathsf{u}) + \varphi_{s,2}(\mathsf{u}), \tag{6.19}$$

où, pour tout $s \in \{1, \ldots, S\}$,

$$(\forall \mathsf{u} \in [0, +\infty[) \quad \varphi_{s,1}(\mathsf{u}) = \frac{1}{2}\left(\mathsf{u}^2 + (z^{(s)})^2\right), \tag{6.20}$$

$$\varphi_{s,2}(\mathsf{u}) = -z^{(s)}\left(\mathsf{u}^2 + \delta^2\right)^{1/2}. \tag{6.21}$$

Pour tout $s \in \{1, \ldots, S\}$, les dérivées premières et secondes de $\varphi_{s,1}$ et $\varphi_{s,2}$ en $\mathsf{u} \in [0, +\infty[$ [2] sont données par

$$\begin{aligned} \dot{\varphi}_{s,1}(\mathsf{u}) &= \mathsf{u}, & \ddot{\varphi}_{s,1}(\mathsf{u}) &= 1, \\ \dot{\varphi}_{s,2}(\mathsf{u}) &= -\mathsf{z}^{(s)}\mathsf{u}(\mathsf{u}^2 + \delta^2)^{-1/2}, & \ddot{\varphi}_{s,2}(\mathsf{u}) &= -\mathsf{z}^{(s)}\delta^2(\mathsf{u}^2 + \delta^2)^{-3/2}. \end{aligned} \tag{6.22}$$

Ainsi, on obtient une approximation de la fonction d'attache aux données (6.17) séparée en deux fonctions $\widehat{\mathsf{h}} = \mathsf{h}_1 + \mathsf{h}_2$, définies par

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}_1(\mathbf{x}) = \sum_{s=1}^{S} \varphi_{s,1}(|[\mathbf{H}\mathbf{W}\mathbf{x}]^{(s)}|), \tag{6.23}$$

$$\mathsf{h}_2(\mathbf{x}) = \sum_{s=1}^{S} \varphi_{s,2}(|[\mathbf{H}\mathbf{W}\mathbf{x}]^{(s)}|). \tag{6.24}$$

2. On considèrera la dérivée à droite lorsque $\mathsf{u} = 0$.

Nous pouvons remarquer que dans le cas limite où $\delta = 0$, on retrouve la fonction de moindres carrés standard (6.18).

Concernant le terme de régularisation, nous nous focalisons sur le cas où g est une fonction additivement séparable par bloc. Plus précisément, définissons une partition $(\mathbb{J}_j)_{1\leqslant j\leqslant J}$ de l'ensemble des indices des coefficients de trames $\{1,\dots,N\}$ en $J \geqslant 2$ sous-ensembles de dimensions (non nulles) $(N_j)_{1\leqslant j\leqslant J}$. On suppose alors que la fonction g peut s'écrire

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}(\mathbf{x}) = \sum_{j=1}^{J} \mathsf{g}_j(\mathbf{x}^{(j)}), \tag{6.25}$$

où, pour tout $j \in \{1,\dots,J\}$, $\mathsf{g}_j \colon \mathbb{R}^{N_j} \to]-\infty,+\infty]$ est une fonction semi-algébrique satisfaisant l'hypothèse 5.1(i).

La structure du problème (6.16) nous permet donc d'utiliser l'algorithme 5.4 pour résoudre le problème de reconstruction de phase décrit dans la section 6.3.1.2. Dans la suite nous utiliserons les mêmes notations que dans la section 5.2. Par ailleurs, nous introduisons l'opérateur linéaire $\mathbf{T} = \mathbf{H}\mathbf{W} = (\mathsf{T}^{(s,n)})_{1\leqslant s\leqslant S, 1\leqslant n\leqslant N} \in \mathbb{C}^{S\times N}$.

6.3.2 Mise en œuvre de l'algorithme

Comme nous l'avons vu dans le chapitre 4, la vitesse de convergence de l'algorithme explicite-implicite à métrique variable repose sur le choix des matrices induisant la métrique. Il en va de même pour sa version alternée par bloc donnée par l'algorithme 5.4. L'efficacité des simulations numériques repose sur l'utilisation de majorantes quadratiques donnant, à chaque itération $k \in \mathbb{N}$ de l'algorithme 5.4, de bonnes approximations de la fonction $\mathsf{h}_{j_k}(\cdot, \mathbf{x}_k^{(\overline{\jmath}_k)})$ [3] en $\mathbf{x}_k^{(j_k)}$, dont les matrices de courbure $(\mathbf{A}_{j_k}(\mathbf{x}_k))_{k\in\mathbb{N}}$ sont simples à implémenter.

Définissons, pour tout $k \in \mathbb{N}$, les fonctions $\mathsf{h}_{1,j_k}(\cdot, \mathbf{x}_k^{\overline{\jmath}_k})$ et $\mathsf{h}_{2,j_k}(\cdot, \mathbf{x}_k^{\overline{\jmath}_k})$ respectivement associées à h_1 et h_2. Nous proposons de majorer ces deux fonctions séparément.

D'une part, nous avons remarqué que, pour tout $s \in \{1,\dots,S\}$, $\varphi_{s,2}$ est concave. Ainsi, pour tout $k \in \mathbb{N}$, $\mathsf{h}_{2,j_k}(\cdot, \mathbf{x}_k^{\overline{\jmath}_k})$ peut être majorée en $\mathbf{x}_k^{(j_k)}$ par son approximation de premier ordre :

$$(\forall \mathbf{y} \in \mathbb{R}^{N_{j_k}}) \quad \mathsf{q}_{2,j_k}(\mathbf{y}, \mathbf{x}_k) = \mathsf{h}_2(\mathbf{x}_k) + \left\langle \mathbf{y} - \mathbf{x}_k^{(j_k)}, \nabla_{j_k} \mathsf{h}_2(\mathbf{x}_k) \right\rangle. \tag{6.26}$$

Il nous reste donc à construire des matrices SDP $(\mathbf{A}_{j_k}(\mathbf{x}_k))_{k\in\mathbb{N}}$ telles que, pour tout $k \in \mathbb{N}$,

$$(\forall \mathbf{y} \in \mathbb{R}^{N_{j_k}}) \quad \mathsf{q}_{1,j_k}(\mathbf{y}, \mathbf{x}_k) = \mathsf{h}_1(\mathbf{x}_k) + \left\langle \mathbf{y} - \mathbf{x}_k^{(j_k)}, \nabla_{j_k} \mathsf{h}_1(\mathbf{x}_k) \right\rangle + \frac{1}{2} \|\mathbf{y} - \mathbf{x}_k^{(j_k)}\|^2_{\mathbf{A}_{j_k}(\mathbf{x}_k)} \tag{6.27}$$

soit une fonction majorant $\mathsf{h}_{1,j_k}(\cdot, \mathbf{x}_k^{\overline{\jmath}_k})$. Dans la proposition suivante, nous donnons une matrice $\mathbf{B} \in \mathcal{S}_N^+$ nous permettant de construire des approximations majorantes de h_1 en $\mathbf{x}_k$ pour tout $k \in \mathbb{N}$.

Proposition 6.2. *Soit $\widetilde{\mathbf{x}} \in \mathbb{R}^N$. Une fonction quadratique majorant h_1 en $\widetilde{\mathbf{x}}$ est donnée par*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}_1(\mathbf{x}, \widetilde{\mathbf{x}}) = \mathsf{h}_1(\widetilde{\mathbf{x}}) + \langle \mathbf{x} - \widetilde{\mathbf{x}}, \nabla \mathsf{h}_1(\widetilde{\mathbf{x}}) \rangle + \frac{1}{2} \|\mathbf{x} - \widetilde{\mathbf{x}}\|^2_{\mathbf{B}}, \tag{6.28}$$

3. Pour tout $k \in \mathbb{N}$, la fonction $\mathsf{h}_{j_k}(\cdot, \mathbf{x}_k^{(\overline{\jmath}_k)})$ est définie par l'équation (5.4)

où $\mathbf{B} = \mathrm{Diag}\left(\mathbf{\Omega}^\top \mathbf{1}_S\right) + \varepsilon \mathbf{I}_N$, $\mathbf{1}_S$ *étant le vecteur unité de* $\mathbb{R}^S$, $\varepsilon \geqslant 0$, *et* $\mathbf{\Omega} = \left(\Omega^{(s,n)}\right)_{1\leqslant s\leqslant S, 1\leqslant n\leqslant N} \in \mathbb{R}^{S\times N}$ *est définie par*

$$(\forall s \in \{1,\dots,S\})(\forall n \in \{1,\dots,N\})$$
$$\Omega^{(s,n)} = |\mathsf{T}_{\mathcal{R}}^{(s,n)}| \sum_{n'=1}^{N} |\mathsf{T}_{\mathcal{R}}^{(s,n')}| + |\mathsf{T}_{\mathcal{I}}^{(s,n)}| \sum_{n'=1}^{N} |\mathsf{T}_{\mathcal{I}}^{(s,n')}|. \quad (6.29)$$

Démonstration. Soit $\widetilde{\mathbf{x}} \in \mathbb{R}^N$. Pour tout $s \in \{1,\dots,S\}$, on a,

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad \varphi_{1,s}\left(|\mathbf{T}^{(s)}\mathbf{x}|\right) = \varphi_{1,s}\left(|\mathbf{T}^{(s)}\widetilde{\mathbf{x}}|\right) + \left\langle \mathbf{x}-\widetilde{\mathbf{x}}, \left[(\mathbf{T}^{(s)})^*\mathbf{T}^{(s)}\right]_{\mathcal{R}}\widetilde{\mathbf{x}}\right\rangle + \frac{1}{2}|\mathbf{T}^{(s)}(\mathbf{x}-\widetilde{\mathbf{x}})|^2,$$

où $\mathbf{T}^{(s)}$ désigne la ligne s de la matrice $\mathbf{T}$, et $(\cdot)^*$ est l'opérateur de conjugué de matrice. En sommant sur $s \in \{1,\dots,S\}$, on obtient alors

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}_1(\boldsymbol{x}) = \mathsf{h}_1(\widetilde{\mathbf{x}}) + \langle \mathbf{x}-\widetilde{\mathbf{x}}, \nabla \mathsf{h}_1(\widetilde{\mathbf{x}})\rangle + \frac{1}{2}|||\mathbf{T}(\mathbf{x}-\widetilde{\mathbf{x}})|||^2, \quad (6.30)$$

où $|||\cdot|||$ est la norme hermitienne de $\mathbb{C}^S$.

Soient $(\mathsf{V}_{\mathcal{R}}^{(s,n)})_{1\leqslant s\leqslant S, 1\leqslant n\leqslant N} \in [0,+\infty[^{S\times N}$ et $(\mathsf{V}_{\mathcal{I}}^{(s,n)})_{1\leqslant s\leqslant S, 1\leqslant n\leqslant N} \in [0,+\infty[^{S\times N}$ définis tels que, pour tout $s \in \{1,\dots,S\}$,

$$\sum_{n\in\mathcal{S}_{\mathcal{R}}^{(s)}} \mathsf{V}_{\mathcal{R}}^{(s,n)} = 1, \quad \text{et} \quad \sum_{n\in\mathcal{S}_{\mathcal{I}}^{(s)}} \mathsf{V}_{\mathcal{I}}^{(s,n)} = 1$$

où

$$\mathcal{S}_{\mathcal{R}}^{(s)} = \left\{n \in \{1,\dots,N\} \;\middle|\; \mathsf{V}_{\mathcal{R}}^{(s,n)} \neq 0\right\} = \left\{n \in \{1,\dots,N\} \;\middle|\; T_{\mathcal{R}}^{(s,n)} \neq 0\right\},$$
$$\mathcal{S}_{\mathcal{I}}^{(s)} = \left\{n \in \{1,\dots,N\} \;\middle|\; \mathsf{V}_{\mathcal{I}}^{(s,n)} \neq 0\right\} = \left\{n \in \{1,\dots,N\} \;\middle|\; T_{\mathcal{I}}^{(s,n)} \neq 0\right\}.$$

D'après l'inégalité de Jensen, pour tout $s \in \{1,\dots,S\}$, on a

$$\begin{aligned}
&\left|\sum_{n=1}^{N} \mathsf{T}^{(s,n)}(\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})\right|^2 \\
&= \left(\sum_{n=1}^{N} \mathsf{T}_{\mathcal{R}}^{(s,n)}(\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})\right)^2 + \left(\sum_{n=1}^{N} \mathsf{T}_{\mathcal{I}}^{(s,n)}(\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})\right)^2 \\
&= \left(\sum_{n\in\mathcal{S}_{\mathcal{R}}^{(s)}} \mathsf{V}_{\mathcal{R}}^{(s,n)} \left(\frac{\mathsf{T}_{\mathcal{R}}^{(s,n)}}{\mathsf{V}_{\mathcal{R}}^{(s,n)}}(\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})\right)\right)^2 + \left(\sum_{n\in\mathcal{S}_{\mathcal{I}}^{(s)}} \mathsf{V}_{\mathcal{I}}^{(s,n)} \left(\frac{\mathsf{T}_{\mathcal{I}}^{(s,n)}}{\mathsf{V}_{\mathcal{I}}^{(s,n)}}(\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})\right)\right)^2 \\
&\leqslant \sum_{n\in\mathcal{S}_{\mathcal{R}}^{(s)}} \frac{(\mathsf{T}_{\mathcal{R}}^{(s,n)})^2}{\mathsf{V}_{\mathcal{R}}^{(s,n)}}(\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})^2 + \sum_{n\in\mathcal{S}_{\mathcal{I}}^{(s)}} \frac{(\mathsf{T}_{\mathcal{I}}^{(s,n)})^2}{\mathsf{V}_{\mathcal{I}}^{(s,n)}}(\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})^2. \quad (6.31)
\end{aligned}$$

En particulier, nous pouvons choisir

$$(\forall (s,n) \in \{1,\dots,S\} \times \{1,\dots,N\}) \quad \mathsf{V}_{\mathcal{R}}^{(s,n)} = \begin{cases} 0 & \text{si } \mathsf{T}_{\mathcal{R}}^{(s,n)} = 0, \\ \dfrac{|\mathsf{T}_{\mathcal{R}}^{(s,n)}|}{\sum_{n'=1}^{N} |\mathsf{T}_{\mathcal{R}}^{(s,n')}|} & \text{sinon,} \end{cases}$$

$$\mathsf{V}_{\mathcal{I}}^{(s,n)} = \begin{cases} 0 & \text{si } \mathsf{T}_{\mathcal{I}}^{(s,n)}\} = 0, \\ \dfrac{|\mathsf{T}_{\mathcal{I}}^{(s,n)}|}{\sum_{n'=1}^{N} |\mathsf{T}_{\mathcal{I}}^{(s,n')}|} & \text{sinon.} \end{cases}$$

On peut alors déduire de (6.31) que, pour tout $s \in \{1,\dots,S\}$,

$$\left| \sum_{n=1}^{N} \mathsf{T}^{(s,n)} (\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)}) \right|^2 \leqslant \sum_{n=1}^{N} \left(|\mathsf{T}_{\mathcal{R}}^{(s,n)}| \sum_{n'=1}^{N} |\mathsf{T}_{\mathcal{R}}^{(s,n')}| \right) (\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})^2 + \sum_{n=1}^{N} \left(|\mathsf{T}_{\mathcal{I}}^{(s,n)}| \sum_{n'=1}^{N} |\mathsf{T}_{\mathcal{I}}^{(s,n')}| \right) (\mathsf{x}^{(n)} - \widetilde{\mathsf{x}}^{(n)})^2.$$

On obtient ainsi

$$|||\mathbf{T}(\mathbf{x} - \widetilde{\mathbf{x}})|||^2 \leqslant \left\langle \mathbf{x} - \widetilde{\mathbf{x}}, \mathrm{Diag}\left(\mathbf{\Omega}^\top \mathbf{1}_S\right)(\mathbf{x} - \widetilde{\mathbf{x}}) \right\rangle, \tag{6.32}$$

où $\mathbf{\Omega}$ est définie par (6.29). Pour finir, en combinant les équations (6.30) et (6.32), on en déduit que la fonction h_1 peut être bornée par la majorante (6.28). □

En utilisant le lemme 6.2 et la remarque 5.3(ii), nous pouvons maintenant construire, pour tout $k \in \mathbb{N}$, une majorante quadratique de $\mathsf{h}_{1,j_k}(\cdot, \overline{\mathbf{x}}_k^{j_k})$ en $\mathbf{x}_k$ de la forme de (6.27) avec

$$(\forall k \in \mathbb{N}) \quad \mathbf{A}_{j_k}(\mathbf{x}_k) = \mathrm{Diag}\left(\mathbf{\Omega}_{j_k}^\top \mathbf{1}_S\right) + \varepsilon\, \mathbf{I}_{N_{j_k}}, \tag{6.33}$$

où $\mathbf{\Omega}_{j_k} \in \mathbb{R}^{S \times N_{j_k}}$ est la matrice obtenue en extrayant de la matrice $\mathbf{\Omega}$ définie par (6.29) les colonnes dont les indices sont dans $\mathbb{J}_{j_k}$. Remarquons que l'hypothèse 5.2(ii) est satisfaite pour les matrices (6.33) avec

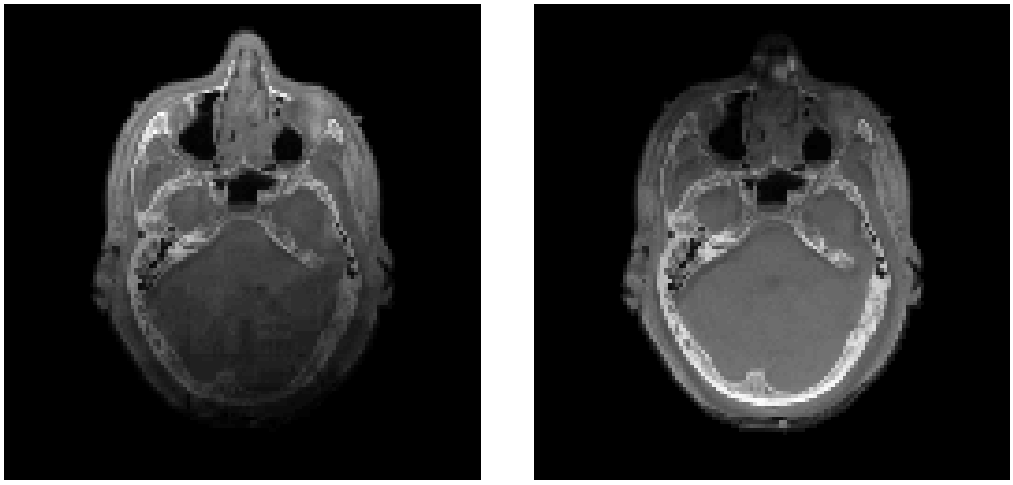
$$\begin{cases} \underline{\nu} = \varepsilon + \min\limits_{n \in \mathbb{J}_{j_k}} \sum_{s=1}^{S} \Omega^{(s,n)}, \\ \overline{\nu} = \varepsilon + \max\limits_{n \in \mathbb{J}_{j_k}} \sum_{s=1}^{S} \Omega^{(s,n)}. \end{cases} \tag{6.34}$$

Si aucune colonne de $\mathbf{T}$ n'est nulle, alors on peut choisir $\varepsilon = 0$ dans (6.34). Sinon, il faut prendre $\varepsilon > 0$.

6.3.3 Résultats de simulation

Nous allons maintenant présenter nos résultats de simulation. Considérons une image originale $\overline{\mathbf{y}} \in \mathbb{C}^M$ à valeurs complexes (voir figure 6.6) de dimension $M = 128 \times 128$. Nous observons une version dégradée $\mathbf{z} \in \mathbb{R}^S$ de cette image, suivant le modèle (6.15). La matrice de dégradation $\mathbf{H} \in \mathbb{C}^{S \times M}$ correspond à la composition de deux opérateurs, le premier étant une matrice de projection

modélisant $S = 23400$ projections de Radon à partir de 128 lignes d'acquisition parallèles et 180 angles équidistants sur $[0, \pi[$, et le second correspond à un opérateur de flou à valeurs complexes. Pour modéliser l'opérateur de flou, nous avons utilisé un noyau de convolution 1D de longueur 3, agissant séparément sur chaque angle d'acquisition. Ce modèle s'inspire de la version linéarisée d'un problème de tomographie par contraste de phase proposé dans [Davidoiu *et al.*, 2012; Guigay *et al.*, 2007] dans le cas 3D. D'autre part, on suppose que le vecteur $\mathbf{w}$ est une réalisation d'un vecteur aléatoire suivant une loi normale centrée et de variance égale à 0.025 (tronquée de façon à garantir la positivité des observations).



(a) (b)

Figure 6.6 – *Parties réelle* (a) *et imaginaire* (b) *de l'image originale* $\overline{\mathbf{y}}$

L'opérateur de trame $\mathbf{W} \in \mathbb{R}^{M \times N}$ est choisi de façon à ce que $N = 8M$ et que chaque vecteur $\mathbf{x} \in \mathbb{R}^N$ soit la concaténation de deux vecteurs $(\mathbf{x}_{\mathcal{R}}, \mathbf{x}_{\mathcal{I}}) \in \mathbb{R}^{4M} \times \mathbb{R}^{4M}$, où $\mathbf{x}_{\mathcal{R}} = (\mathrm{x}_{\mathcal{R}}^{(p)})_{1 \leqslant p \leqslant 4M}$ (resp. $\mathbf{x}_{\mathcal{I}} = (\mathrm{x}_{\mathcal{I}}^{(p)})_{1 \leqslant p \leqslant 4M}$) correspond à une décomposition de Haar redondante de $\mathbf{y}_{\mathcal{R}}$ (resp. $\mathbf{y}_{\mathcal{I}}$) sur un niveau de résolution. Un exemple de décomposition de la partie réelle de $\overline{\mathbf{y}}$ est donné dans la figure 6.7(a).

La fonction de régularisation g est définie par

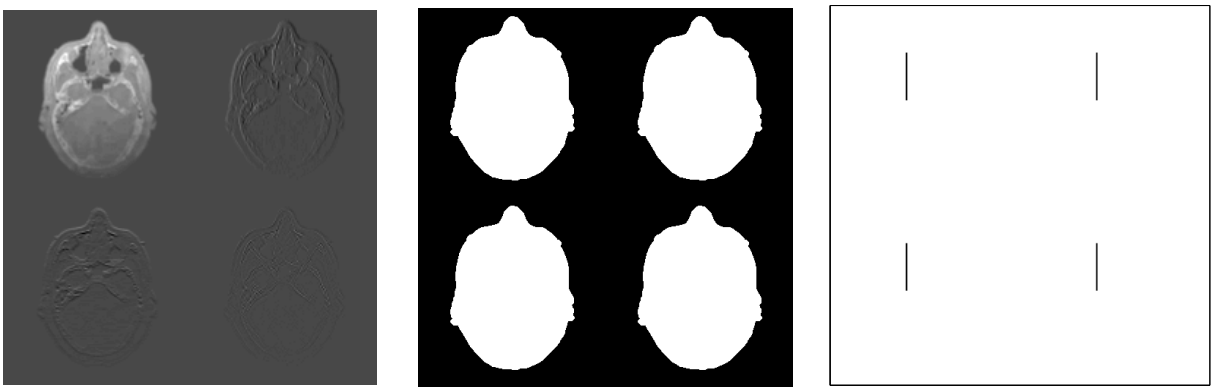
$$(\forall \mathbf{x} \in \mathbb{R}^{8M}) \quad \mathrm{g}(\mathbf{x}) = \sum_{p=1}^{4M} \mathrm{g}_p(\mathrm{x}_{\mathcal{R}}^{(p)}, \mathrm{x}_{\mathcal{I}}^{(p)}), \tag{6.35}$$

avec, pour tout $p \in \{1, \ldots, 4M\}$,

$$(\forall \mathbf{u} \in \mathbb{R}^2) \quad \mathrm{g}_p(\mathbf{u}) = \begin{cases} \vartheta_p \|\mathbf{u} - \boldsymbol{\omega}_p\|_2^{\kappa_p} & \text{si } p \notin \mathbb{E}, \\ 0 & \text{si } p \in \mathbb{E} \text{ et } \mathbf{u} = \mathbf{0}, \\ +\infty & \text{sinon}, \end{cases} \tag{6.36}$$

où $\mathbb{E}$ correspond à l'arrière plan de l'objet (donné dans la figure 6.7(b) pour les coefficients correspondants à la partie réelle de l'image). Afin de promouvoir la parcimonie des coefficients de

détails de $\widehat{\mathbf{x}}$ nous choisissons $\kappa_p = 1$ et $\vartheta_p = \vartheta^d \in]0, +\infty[$ lorsque $p \in \{M+1, \ldots, 4M\}$. Pour les coefficients d'approximation, nous choisissons $\kappa_p = 2$ et $\vartheta_p = \vartheta^a \in]0, +\infty[$ lorsque $p \in \{1, \ldots, M\}$. Le paramètre $\boldsymbol{\omega}_p \in \mathbb{R}^2$ correspond à une valeur moyenne des coefficients de trame. Dans nos simulations nous avons pris $\boldsymbol{\omega}_p = (0.4, 0.6)$ pour les coefficients d'approximation, et $\boldsymbol{\omega}_p = \mathbf{0}$ sinon. Ainsi, la fonction de régularisation donnée par (6.35)-(6.36) satisfait l'hypothèse 5.1(i) et est semi-algébrique. De plus, son opérateur proximal a une forme explicite (voir section 2.3.4).



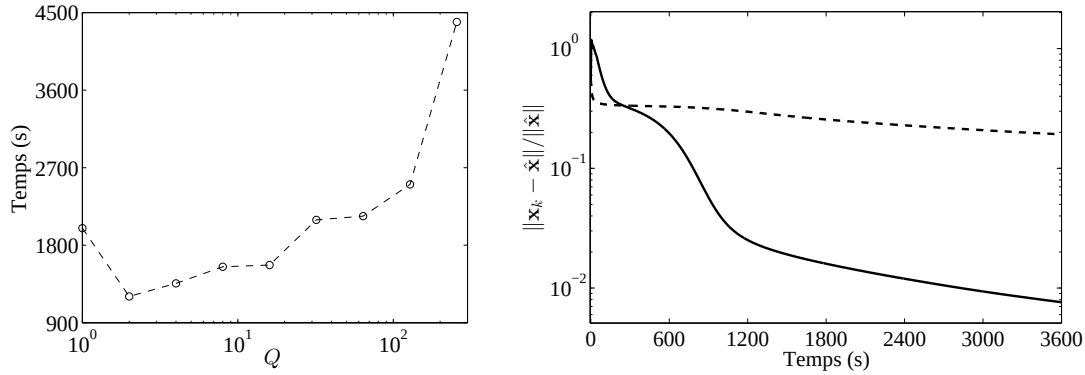
(a) (b) (c)

Figure 6.7 – *Seuls les coefficients de trame correspondants à la partie réelle de l'image* $\overline{\mathbf{y}}_{\mathcal{R}}$ *sont représentés dans les trois figures.* (a) *Exemple de décomposition de trame de* $\overline{\mathbf{y}}_{\mathcal{R}}$. (b) *Masque* $\mathbb{E}$ *permettant de forcer les coefficients de l'arrière plan de l'objet à être égaux à* 0. (c) *Indices d'un bloc* $\mathbf{x}^{(j)}$ *pour* $Q = 32$.

Dans nos simulations, les paramètres ϑ^a, ϑ^d et δ sont ajustés de façon à maximiser le RSB entre l'image originale $\overline{\mathbf{y}}$ et l'image reconstruite $\widehat{\mathbf{y}}$. Dans l'algorithme 5.4, pour tout j, $\mathbf{x}^{(j)}$ correspond à un vecteur, de dimension fixe $8Q$, faisant intervenir 8 blocs extraits à la fois des approximations et des détails des parties réelles et imaginaires de l'image. Une illustration pour la partie réelle est donnée dans la figure 6.7(c). À chaque itération $k \in \mathbb{N}$, j_k est choisi aléatoirement de façon à ce que chaque bloc soit mis à jour au moins une fois toutes les J itérations. En pratique, nous observons que le temps de reconstruction varie en fonction du paramètre Q. La figure 6.8(a) donne le temps de reconstruction nécessaire à l'algorithme 5.4 pour atteindre le critère d'arrêt $\|\mathbf{x}_k - \widehat{\mathbf{x}}\|/\|\widehat{\mathbf{x}}\| \leqslant 0.025$ en fonction du paramètre Q, où la solution optimale $\widehat{\mathbf{x}}$ a été calculée préalablement pour un grand nombre d'itérations. Dans cet exemple, le meilleur choix en terme de vitesse de convergence semble être $Q = 2$.

La figure 6.8(b) illustre la décroissance de $(\|\mathbf{x}_k - \widehat{\mathbf{x}}\|/\|\widehat{\mathbf{x}}\|)_k$ (en échelle logarithmique) obtenue en utilisant l'algorithme 5.4 et l'algorithme PALM de [Bolte *et al.*, 2013] avec $\gamma_k \equiv 1.9$ et $Q = 2$, où $\widehat{\mathbf{x}}$ correspond à la solution optimale obtenue pour chaque algorithme. Nous pouvons observer que, dans cet exemple, le préconditionnement permet d'accélérer la vitesse de convergence des itérées.

Dans la figure 6.9(a)-(b) nous donnons les parties réelle et imaginaire de l'image reconstruite en utilisant la méthode proposée dans cette section. Nous donnons aussi dans la figure 6.9(c)-(d) les parties réelle et imaginaire de l'image reconstruite obtenue en utilisant une version améliorée



(a) (b)

Figure 6.8 – (a) *Temps de reconstruction nécessaire pour satisfaire le critère d'arrêt* $\|\mathbf{x}_k - \widehat{\mathbf{x}}\|/\|\widehat{\mathbf{x}}\| \leqslant 0.025$ *avec l'algorithme 5.4 en fonction de* Q. (b) *Profil de convergence de l'algorithme 5.4 (ligne continue) et de l'algorithme PALM de* [Bolte *et al.*, 2013] *(ligne discontinue), obtenu en utilisant Matlab 7 sur une machine Intel(R) Core(TM) i7-3520M @ 2.9GHz.*

la méthode de projections alternée proposée par [Mukherjee et Seelamantula, 2012], où la non-inversibilité de la matrice $\mathbf{H}$ a été prise en compte. Remarquons de plus qu'il faut compter un temps 10 fois plus long avec la méthode de [Mukherjee et Seelamantula, 2012] que le temps mis avec l'algorithme 5.4, pour obtenir une valeur de RSB stable. Notons qu'en raison de la grande dimension du problème traité, les approches SDP de [Candès *et al.*, 2013; Jaganathan *et al.*, 2012; Waldspurger *et al.*, 2013] ainsi que la méthode gloutonne de [Shechtman *et al.*, 2014] ne peuvent pas être mises en œuvre. D'autre part, l'algorithme standard de GS ne donne pas des résultats de qualité acceptable.

6.4 Déconvolution aveugle de signaux sismiques

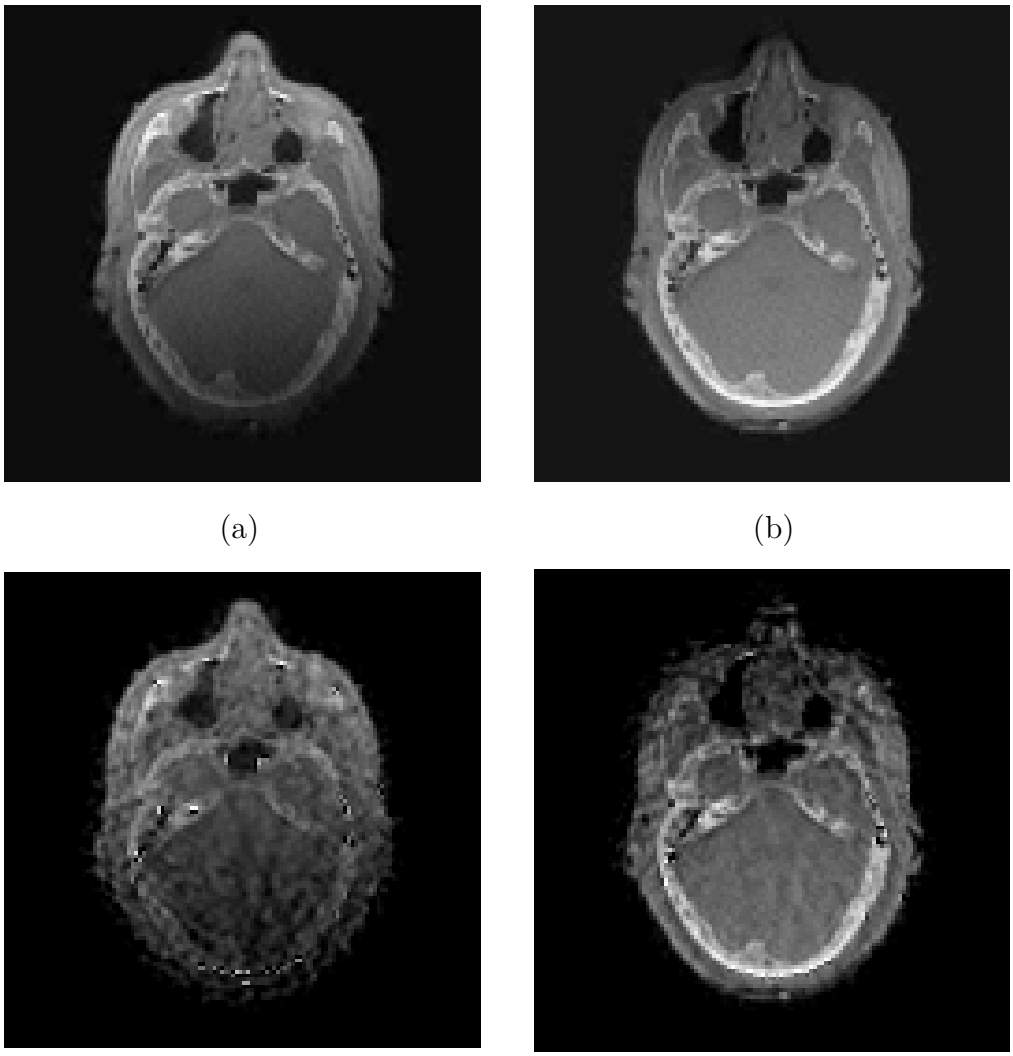
6.4.1 Formulation du problème

6.4.1.1 État de l'art

Dans cette section, nous cherchons à estimer un signal sismique inconnu $\overline{\mathbf{x}} \in \mathbb{R}^N$ à partir d'observations $\mathbf{z} \in \mathbb{R}^N$ obtenues par le modèle :

$$\mathbf{z} = \overline{\mathbf{k}} * \overline{\mathbf{x}} + \mathbf{w}, \tag{6.37}$$

où $\overline{\mathbf{k}} \in \mathbb{R}^S$ représente une réponse impulsionnelle (par exemple, la réponse linéaire d'un capteur, ou un noyau de convolution modélisant un flou), et $\mathbf{w} \in \mathbb{R}^N$ est une réalisation d'un vecteur aléatoire modélisant un bruit additif. Un exemple de signal sismique $\overline{\mathbf{x}} \in \mathbb{R}^N$ ainsi que de son observation faite au travers du modèle (6.37) sont donnés dans la figure 6.10.



(a) (b)

(c) (d)

Figure 6.9 – *Parties réelle* (a) *(resp.* (c)*) et imaginaire* (b) *(resp.* (d)*) de l'image reconstruite* $\hat{\mathbf{y}}$ *en utilisant l'algorithme 5.4, RSB =* 21.27 *dB (resp. l'algorithme de projections alternées régularisé de* [Mukherjee et Seelamantula, 2012]*, RSB =* 14.45 *dB).*

L'utilisation d'approches standards, telles que le filtre de Wiener [Wiener, 1949] ou des extensions de type SURELET [Pesquet *et al.*, 2009], permettent de trouver une estimée $\widehat{\mathbf{x}} \in \mathbb{R}^N$ du signal original $\overline{\mathbf{x}}$ en minimisant un critère basé sur le carré de la norme euclidienne $\|\cdot\|^2$. Cependant, d'une part l'utilisation seule d'une fonction d'attache aux données de moindres carrés ne permet pas de débruiter correctement le signal, et d'autre part, en ajoutant une régularisation $\|\cdot\|^2$ on obtient souvent des estimations trop lisses du signal de départ.

Dans cette section, nous nous intéresserons au cas de la déconvolution aveugle, c'est à dire lorsque le noyau $\overline{\mathbf{k}}$ est lui aussi inconnu, et doit être estimé conjointement au signal $\overline{\mathbf{x}}$. Ce type

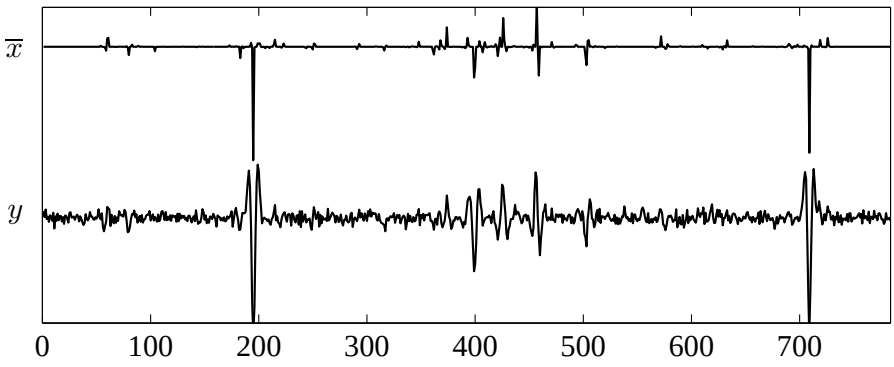


Figure 6.10 – *(Haut) Signal sismique inconnu* $\overline{\mathbf{x}}$*. (Bas) Observations* $\mathbf{z}$*.*

de problème peut se rencontrer dans diverses applications telles que les communications (égalisation ou estimation de canal) [Haykin, 1994], le contrôle non destructif [Nandi *et al.*, 1997], la géophysique [Kaaresen et Taxt, 1998; Pham *et al.*, 2014; Takahata *et al.*, 2012], le traitement des images [Ahmed *et al.*, 2014; Kato *et al.*, 1999; Kundur et Hatzinakos, 1996a,b], l'imagerie médicale et la télédétection Campisi et Egiazarian [2007]. Le problème de déconvolution aveugle étant un problème très sous-déterminé, il requiert l'utilisation d'hypothèses supplémentaires sur le signal et le noyau à estimer. Une approche usuelle est de définir les estimées $(\widehat{\mathbf{x}}, \widehat{\mathbf{k}}) \in \mathbb{R}^N \times \mathbb{R}^S$ de $(\overline{\mathbf{x}}, \overline{\mathbf{k}})$ comme les minimiseurs de la somme d'une fonction d'attache aux données et des termes de régularisation sur le signal et sur le noyau de convolution (voir section 2.2.1). Le problème de déconvolution aveugle étant sujet à des ambiguïtés d'échelles, et les fonctions de régularisation servant à la fois à prendre en compte des connaissances sur les objets originaux et assurant la stabilité de la solution, il peut être intéressant de les choisir invariantes par échelle [Comon, 1996; Moreau et Pesquet, 1997].

La fonction ℓ_1/ℓ_2, introduite dans [Gray, 1978] pour traiter des problèmes de déconvolution en géophysique, posséde de telles propriétés. Elle a été utilisée plus récemment comme mesure de parcimonie dans [Barak *et al.*, 2014; Hoyer, 2004; Hurley et Rickard, 2004; Zibulevsky et Pearlmutter, 2001] pour résoudre des problèmes de NMF (*Nonnegative Matrix Factorization*) [Mørup *et al.*, 2008], des problèmes de traitement d'image en utilisant une décomposition en ondelettes [Ji *et al.*, 2012], ou pour la reconstruction de signaux parcimonieux [Demanet et Hand, 2014]. Il est de plus souligné dans [Benichoux *et al.*, 2013] que cette fonction de régularisation peut être efficace pour les problèmes de déconvolution aveugle de signaux parcimonieux.

Récemment, [Krishnan *et al.*, 2011] a proposé un algorithme de minimisation alternée permettant d'utiliser la fonction de régularisation ℓ_1/ℓ_2. Les auteurs proposent de transformer le problème régularisé par la fonction non convexe ℓ_1/ℓ_2 en une famille de problèmes régularisés par la fonction convexe ℓ_1. Pour cela, à chaque itération, le résultat obtenu est pondéré en le divisant par la norme ℓ_2 de l'itérée précédente. Le problème régularisé par ℓ_1 peut être résolu en utilisant l'algorithme explicite-implicite (aussi appelé ISTA (*Iterative Shrinkage-Thresholding Algorithm*) lorsque la fonction non lisse est la norme ℓ_1 [Beck et Teboulle, 2009]). Bien que la convergence de cette méthode n'aie pas été établie, elle semble très efficace en pratique. Une autre méthode, basée sur un algorithme de gradient projeté avec changement d'échelle, a été proposée dans [Esser *et al.*, 2013] pour résoudre une approximation lisse de la fonction ℓ_1/ℓ_2. Cependant, cette approche ne permet de résoudre que des problèmes traitant des signaux parcimonieux à

valeurs positives.

L'objectif de cette section est de proposer un algorithme efficace permettant de prendre en compte une approximation lisse de la pénalisation ℓ_1/ℓ_2 pour des problèmes traitant des données à valeurs réelles signées.

6.4.1.2 Problème d'optimisation

On définit les estimées $(\widehat{\mathbf{x}}, \widehat{\mathbf{k}}) \in \mathbb{R}^N \times \mathbb{R}^S$ de $(\overline{\mathbf{x}}, \overline{\mathbf{k}})$ comme les minimiseurs de la fonction

$$(\forall (\mathbf{x}, \mathbf{k}) \in \mathbb{R}^N \times \mathbb{R}^S) \quad \mathsf{f}(\mathbf{x}, \mathbf{k}) = \mathsf{h}_1(\mathbf{x}, \mathbf{k}) + \mathsf{h}_2(\mathbf{x}) + \mathsf{g}_1(\mathbf{x}) + \mathsf{g}_2(\mathbf{k}), \tag{6.38}$$

où $\mathsf{h}_1 \colon \mathbb{R}^N \times \mathbb{R}^S \to]-\infty, +\infty]$ correspond au critère de moindres carrés associé au modèle (6.37) défini par

$$(\forall (\mathbf{x}, \mathbf{k}) \in \mathbb{R}^N \times \mathbb{R}^S) \quad \mathsf{h}_1(\mathbf{x}, \mathbf{k}) = \frac{1}{2} \|\mathbf{k} * \mathbf{x} - \mathbf{z}\|^2, \tag{6.39}$$

$\mathsf{g}_1 \colon \mathbb{R}^N \to]-\infty, +\infty]$ et $\mathsf{g}_2 \colon \mathbb{R}^S \to]-\infty, +\infty]$ sont respectivement des termes de régularisation sur $\mathbf{x}$ et $\mathbf{k}$ non nécessairement lisses, et $\mathsf{h}_2 \colon \mathbb{R}^N \to]-\infty, +\infty]$ est une fonction de régularisation différentiable de gradient Lipschitz sur $\mathbf{x}$. Nous supposons de plus que g_1 et g_2 sont des fonctions semi-algébriques satisfaisant l'hypothèse 5.1(i) donnée dans le chapitre 5, et nous notons, pour tout $(\mathbf{x}, \mathbf{k}) \in \mathbb{R}^N \times \mathbb{R}^S$, $\mathsf{h}(\mathbf{x}, \mathbf{k}) = \mathsf{h}_1(\mathbf{x}, \mathbf{k}) + \mathsf{h}_2(\mathbf{x})$ la partie différentiable du critère (6.38).

Concernant la fonction de régularisation h_2, cette dernière doit à la fois promouvoir la parcimonie de $\overline{\mathbf{x}}$, et être simple à implémenter numériquement. Nous avons vu que la fonction de régularisation non convexe ℓ_1/ℓ_2 [Slavakis *et al.*, 2013], définie comme le quotient des normes

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \ell_1(\mathbf{x}) = \sum_{n=1}^{N} |\mathsf{x}^{(n)}| \quad \text{et} \quad \ell_2(\mathbf{x}) = \left(\sum_{n=1}^{N} (\mathsf{x}^{(n)})^2 \right)^{1/2},$$

permet de traiter efficacement les problèmes de déconvolution aveugle. Cependant, n'étant ni différentiable, ni convexe, elle se révèle compliquée à optimiser.

Nous proposons donc de remplacer cette fonction non lisse par une approximation lisse de celle-ci qui sera plus facilement maniable. Pour cela, rappelons que des approximations lisses de ℓ_1 et ℓ_2, notées $\ell_{1,\alpha}$ (parfois appelée pénalisation ℓ_1-ℓ_2-hybride ou hyperbolique) et $\ell_{2,\eta}$, sont définies par

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \ell_{1,\alpha}(\mathbf{x}) = \sum_{n=1}^{N} \left(\sqrt{(\mathsf{x}^{(n)})^2 + \alpha^2} - \alpha \right) \quad \text{et} \quad \ell_{2,\eta}(\mathbf{x}) = \sqrt{\sum_{n=1}^{N} (\mathsf{x}^{(n)})^2 + \eta^2}, \tag{6.40}$$

où $(\alpha, \eta) \in]0, +\infty[^2$. Remarquons que dans le cas limite où $\alpha = \eta = 0$, on retrouve les normes usuelles ℓ_1 et ℓ_2. Nous définissons alors notre approximation lisse de la fonction ℓ_1/ℓ_2 de la façon suivante :

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{h}_2(\mathbf{x}) = \lambda \log \left(\frac{\ell_{1,\alpha}(\mathbf{x}) + \beta}{\ell_{2,\eta}(\mathbf{x})} \right), \tag{6.41}$$

avec $(\lambda, \beta, \alpha, \eta) \in]0, +\infty[^4$.

D'une part, remarquons que la fonction log permet non seulement de rendre la fonction plus facile à manier, mais aussi de mieux faire ressortir la parcimonie. Un exemple en est donné dans la figure 6.11 où h_2 est comparée à d'autres fonctions permettant de promouvoir la parcimonie.

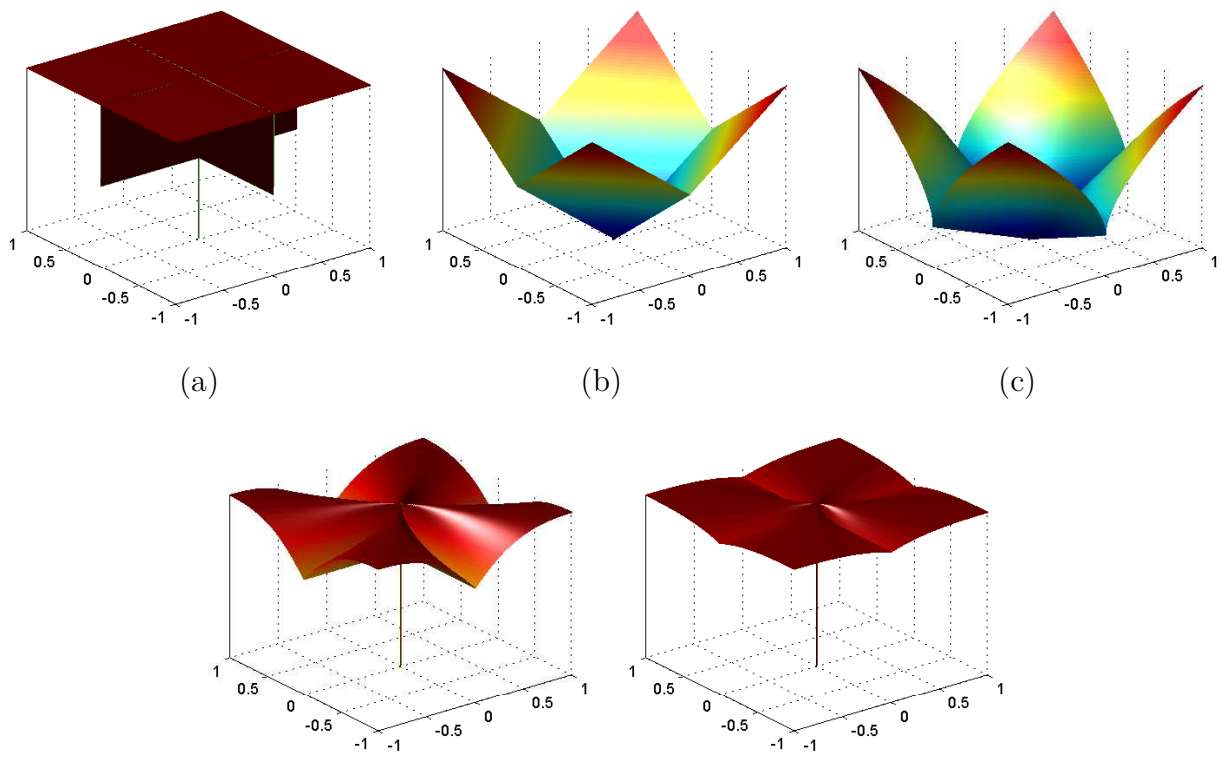


Figure 6.11 – (a) *Fonction* ℓ_0. (b) *Fonction* ℓ_1. (c) *Fonction* $\ell_{1/2}$. (d) *Fonction* $\ell_{1,\alpha}/\ell_{2,\eta}$. (e) *Fonction* $\mathsf{h}_2 = \log\big((\ell_{1,\alpha}+\beta)/\ell_{2,\eta}\big)$.

D'autre part, notons que f correspond à la fonction Lagrangienne associée à la minimisation de $\mathsf{h}_1 + \mathsf{g}_1 + \mathsf{g}_2$ sous la contrainte

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \log\left(\frac{\ell_{1,\alpha}(\mathbf{x})+\beta}{\ell_{2,\eta}(\mathbf{x})}\right) \leqslant \log(\vartheta), \tag{6.42}$$

pour un certain $\vartheta > 0$. D'après la monotonie de la fonction log, (6.42) est équivalente à

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \frac{\ell_{1,\alpha}(\mathbf{x})+\beta}{\ell_{2,\eta}(\mathbf{x})} \leqslant \vartheta, \tag{6.43}$$

qui, d'après (6.41), peut être interprétée comme une formulation sous contrainte d'une approximation lisse de la fonction ℓ_1/ℓ_2, dès que β est assez petit.

Pour finir, soulignons que la fonction h_2 est de gradient Lipschitz sur tout sous-ensemble compact de $\mathbb{R}^N$. Ainsi, la fonction h satisfait l'hypothèse 5.1(ii).

6.4.2 Mise en œuvre de l'algorithme

Pour minimiser la fonction (6.38), nous allons utiliser l'algorithme 5.4 en alternant sur les variables $\mathbf{x}$ et $\mathbf{k}$. Pour cela, nous avons besoin de définir, pour tout $(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) \in \mathbb{R}^N \times \mathbb{R}^S$, des matrices $\mathbf{A}_1(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) \in \mathcal{S}_N^+$ et $\mathbf{A}_2(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) \in \mathcal{S}_S^+$ satisfaisant l'hypothèse 5.2. De telles matrices sont données dans la proposition suivante.

Proposition 6.3. *Pour tout $(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) \in \mathbb{R}^N \times \mathbb{R}^S$, soient*

$$\mathbf{A}_1(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) = \left(L_1(\widetilde{\mathbf{k}}) + \frac{9\lambda}{8\eta^2} \right) \mathrm{I}_N + \frac{\lambda}{\ell_{1,\alpha}(\widetilde{\mathbf{x}}) + \beta} \mathbf{A}_{\ell_{1,\alpha}}(\widetilde{\mathbf{x}}),$$
$$\mathbf{A}_2(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) = (L_2(\widetilde{\mathbf{x}}) + \varepsilon)\, \mathrm{I}_S,$$

où

$$\mathbf{A}_{\ell_{1,\alpha}}(\widetilde{\mathbf{x}}) = \mathrm{Diag}\left(\left((\widetilde{\mathrm{x}}^{(n)})^2 + \alpha^2 \right)^{-1/2} \right)_{1 \leqslant n \leqslant N}, \tag{6.44}$$

$\varepsilon > 0$ et $L_1(\widetilde{\mathbf{k}})$ (resp. $L_2(\widetilde{\mathbf{x}})$) est une constante de Lipschitz de $\nabla_1 \mathsf{h}_1(\cdot, \widetilde{\mathbf{k}})$ (resp. $\nabla_2 \mathsf{h}_1(\widetilde{\mathbf{x}}, \cdot)$).[4] *La fonction définie par*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{q}(\mathbf{x} \,|\, \widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) = \mathsf{h}(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) + \langle \mathbf{x} - \widetilde{\mathbf{x}}, \nabla_1 \mathsf{h}(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) \rangle + \frac{1}{2} \|\mathbf{x} - \widetilde{\mathbf{x}}\|^2_{\mathbf{A}_1(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}})}, \tag{6.45}$$

est une fonction quadratique majorant $\mathsf{h}(\cdot, \widetilde{\mathbf{k}})$ en $\widetilde{\mathbf{x}}$, et

$$(\forall \mathbf{k} \in \mathbb{R}^N) \quad \mathsf{q}(\mathbf{k} \,|\, \widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) = \mathsf{h}(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) + \langle \mathbf{k} - \widetilde{\mathbf{k}}, \nabla_1 \mathsf{h}(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}}) \rangle + \frac{1}{2} \|\mathbf{k} - \widetilde{\mathbf{k}}\|^2_{\mathbf{A}_2(\widetilde{\mathbf{x}}, \widetilde{\mathbf{k}})}, \tag{6.46}$$

est une fonction quadratique majorant $\mathsf{h}(\widetilde{\mathbf{x}}, \cdot)$ en $\widetilde{\mathbf{k}}$.

Démonstration. Soit $\widetilde{\mathbf{x}} \in \mathbb{R}^N$. Décomposons h_2 en une somme de deux fonctions $\mathsf{h}_2^{(1)}$ et $\mathsf{h}_2^{(2)}$ où

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \begin{cases} \mathsf{h}_2^{(1)}(\mathbf{x}) = \lambda \log\left(\ell_{1,\alpha}(\mathbf{x}) + \beta \right), \\ \mathsf{h}_2^{(2)}(\mathbf{x}) = -\lambda \log\left(\ell_{2,\eta}(\mathbf{x}) \right). \end{cases} \tag{6.47}$$

Il nous suffit alors de montrer que

(i) la fonction

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \widetilde{\mathsf{q}}(\mathbf{x} \,|\, \widetilde{\mathbf{x}}) = \mathsf{h}_2^{(1)}(\widetilde{\mathbf{x}}) + \langle \mathbf{x} - \widetilde{\mathbf{x}}, \nabla \mathsf{h}_2^{(1)}(\widetilde{\mathbf{x}}) \rangle + \frac{1}{2} \|\mathbf{x} - \widetilde{\mathbf{x}}\|^2_{\mathbf{A}_{\mathsf{h}_2^{(1)}}(\widetilde{\mathbf{x}})}, \tag{6.48}$$

est une fonction majorante de $\mathsf{h}_2^{(1)}$ en $\widetilde{\mathbf{x}}$ pour

$$\mathbf{A}_{\mathsf{h}_2^{(1)}}(\widetilde{\mathbf{x}}) = \frac{\lambda}{\ell_{1,\alpha}(\widetilde{\mathbf{x}}) + \beta} \mathbf{A}_{\ell_{1,\alpha}}(\widetilde{\mathbf{x}}), \tag{6.49}$$

4. La fonction h_1 étant choisie quadratique, ces constantes de Lipschitz s'obtiennent facilement.

(ii) une constante de Lipschitz du gradient de $\mathsf{h}_2^{(2)}$ est égale à $\mu = \frac{9\lambda}{8\eta^2}$.

D'une part, posons

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{I}(\mathbf{x}) = \ell_{1,\alpha}(\mathbf{x}) + \beta. \tag{6.50}$$

D'après [Allain *et al.*, 2006], on a alors, pour tout $\mathbf{x} \in \mathbb{R}^N$,

$$\mathsf{I}(\mathbf{x}) \leqslant \mathsf{I}(\widetilde{\mathbf{x}}) + (\mathbf{x} - \widetilde{\mathbf{x}})^\top \nabla \mathsf{I}(\widetilde{\mathbf{x}}) + \frac{1}{2}\|\mathbf{x} - \widetilde{\mathbf{x}}\|^2_{\mathbf{A}_{\ell_{1,\alpha}}(\widetilde{\mathbf{x}})}, \tag{6.51}$$

où $\mathbf{A}_{\ell_{1,\alpha}}(\mathsf{I})$ est donnée par (6.44).

D'autre part, pour tout $(\mathsf{u}, \mathsf{v}) \in]0, +\infty[^2$,

$$\log \mathsf{v} \leqslant \log \mathsf{u} + \frac{\mathsf{v}}{\mathsf{u}} - 1 = \log \mathsf{u} + \frac{\mathsf{v} - \mathsf{u}}{\mathsf{u}}. \tag{6.52}$$

En posant $\mathsf{v} = \mathsf{I}(\mathbf{x}) > 0$ et $\mathsf{u} = \mathsf{I}(\widetilde{\mathbf{x}}) > 0$, et en combinant (6.51) et (6.52), on obtient

$$\mathsf{h}_2^{(1)}(\mathbf{x}) \leqslant \mathsf{h}_2^{(1)}(\widetilde{\mathbf{x}}) + \langle \mathbf{x} - \widetilde{\mathbf{x}}, \frac{\lambda}{\tau(\widetilde{\mathbf{x}})} \nabla \mathsf{I}(\widetilde{\mathbf{x}}) \rangle + \frac{1}{2} \langle \mathbf{x} - \widetilde{\mathbf{x}}, \frac{\lambda}{\tau(\mathbf{x})} \mathbf{A}_{\ell_{1,\alpha}}(\widetilde{\mathbf{x}})(\mathbf{x} - \widetilde{\mathbf{x}}) \rangle.$$

Ainsi, l'assertion (i) est vérifiée en remarquant que

$$\nabla \mathsf{h}_2^{(1)}(\widetilde{\mathbf{x}}) = \frac{\lambda}{\mathsf{I}(\widetilde{\mathbf{x}})} \nabla \mathsf{I}(\widetilde{\mathbf{x}}) \quad \text{et} \quad \mathbf{A}_{\mathsf{h}_2^{(1)}}(\widetilde{\mathbf{x}}) = \frac{\lambda}{\mathsf{I}(\widetilde{\mathbf{x}})} \mathbf{A}_{\ell_{1,\alpha}}(\widetilde{\mathbf{x}}).$$

D'autre part, le Hessien de $\mathsf{h}_2^{(2)}$ est donné par

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \nabla^2 \mathsf{h}_2^{(2)}(\mathbf{x}) = \frac{2\lambda}{\ell_{2,\eta}^4(\mathbf{x})} \mathbf{x}\mathbf{x}^\top - \frac{\lambda}{\ell_{2,\eta}^2(\mathbf{x})} \mathbf{I}_N \,.$$

En remarquant que, pour tout $\mathbf{x} \in \mathbb{R}^N$, $\ell_{2,\eta}^2(\mathbf{x}) = \|\mathbf{x}\|^2 + \eta^2$, et en appliquant l'inégalité triangulaire, on obtient

$$\|\nabla^2 \mathsf{h}_2^{(2)}(\mathbf{x})\| \leqslant \frac{2\lambda \|\mathbf{x}\|^2}{(\|\mathbf{x}\|^2 + \eta^2)^2} + \frac{\lambda}{\|\mathbf{x}\|^2 + \eta^2} = \chi(\|\mathbf{x}\|),$$

où

$$\chi \colon \mathsf{u} \in [0, +\infty[\mapsto \lambda \frac{3\mathsf{u}^2 + \eta^2}{(\mathsf{u}^2 + \eta^2)^2}.$$

La dérivée de χ est donnée, pour tout $\mathsf{u} \in [0, +\infty[$, par

$$\dot{\chi}(\mathsf{u}) = \lambda \frac{2\mathsf{u}}{(\mathsf{u}^2 + \eta^2)^3} (\eta^2 - 3\mathsf{u}^2),$$

donc χ est une fonction croissante sur $[0, \eta/\sqrt{3}]$ et décroissante sur $]\eta/\sqrt{3}, +\infty[$. De plus, on a $\sup_{\mathsf{u} \in [0,+\infty[} \chi(\mathsf{u}) = \chi\left(\eta/\sqrt{3}\right) = \frac{9\lambda}{8\eta^2}$. On obtient alors l'assertion (ii). □

Dans le cas particulier de la minimisation de la fonction (6.38), l'algorithme 5.4 se réduit à l'algorithme 6.2. Les propriétés de convergence de cet algorithme sont données dans le théorème 5.1. Remarquons que dans l'algorithme 6.2, pour tout $k \in \mathbb{N}$, les entiers I_k et J_k permettent de mettre à jour plusieurs fois de suite les vecteurs $\mathbf{x}_k$ et $\mathbf{k}_k$. Cela est possible grâce à l'hypothèse de mise à jour quasi-cyclique des blocs donnée dans la définition 5.1(ii). Nous montrerons l'intérêt de cette règle de mis à jour dans la section suivante.

Algorithme 6.2 Algorithme explicite-implicite à métrique variable alterné par bloc appliqué au problème de déconvolution aveugle.

Initialisation : Pour tout $k \in \mathbb{N}$, soient $J_k \in \mathbb{N}^*$, $I_k \in \mathbb{N}^*$ et soient $(\gamma_{\mathbf{x}}^{k,j})_{0 \leqslant j \leqslant J_k - 1} \in]0, +\infty[^{J_k}$ et $(\gamma_{\mathbf{k}}^{k,i})_{0 \leqslant i \leqslant I_k - 1} \in]0, +\infty[^{I_k}$. Soient $\mathbf{x}_0 \in \operatorname{dom} \mathsf{g}_1$ et $\mathbf{k}_0 \in \operatorname{dom} \mathsf{g}_2$.

Iterations :

Pour $k = 0, 1, \ldots$

$\mathbf{x}_{k,0} = \mathbf{x}_k$, $\mathbf{k}_{k,0} = \mathbf{k}_k$,

Pour $j = 0, \ldots, J_k - 1$

$\widetilde{\mathbf{x}}_{k,j} = \mathbf{x}_{k,j} - \gamma_{\mathbf{x}}^{k,j} \mathbf{A}_1(\mathbf{x}_{k,j}, \mathbf{k}_k)^{-1} \nabla_1 \mathsf{h}(\mathbf{x}_{k,j}, \mathbf{k}_k)$,

$\mathbf{x}_{k,j+1} = \operatorname{prox}_{(\gamma_{\mathbf{x}}^{k,j})^{-1} \mathbf{A}_1(\mathbf{x}_{k,j}, \mathbf{k}_k), \mathsf{g}_1} (\widetilde{\mathbf{x}}_{k,j})$,

$\mathbf{x}_{k+1} = \mathbf{x}_{k,J_k}$.

Pour $i = 0, \ldots, I_k - 1$

$\widetilde{\mathbf{k}}_{k,i} = \mathbf{k}_{k,i} - \gamma_{\mathbf{k}}^{k,i} \mathbf{A}_2(\mathbf{x}_{k+1}, \mathbf{k}_{k,i})^{-1} \nabla_2 \mathsf{h}(\mathbf{x}_{k+1}, \mathbf{k}_{k,i})$,

$\mathbf{k}_{k,i+1} = \operatorname{prox}_{(\gamma_{\mathbf{k}}^{k,i})^{-1} \mathbf{A}_2(\mathbf{x}_{k+1}, \mathbf{k}_{k,i}), \mathsf{g}_2} \left(\widetilde{\mathbf{k}}_{k,i}\right)$,

$\mathbf{k}_{k+1} = \mathbf{k}_{k,I_k}$.

6.4.3 Résultats de simulation

Le signal sismique parcimonieux $\overline{\mathbf{x}}$, de longueur $N = 784$, donné en haut de la figure 6.10, est composé d'une suite de "pics" appelés coefficients de réflectivité primaire [Walden et Hosken, 1986]. Ces séries de réflectivités indiquent le temps de parcours d'une onde sismique entre deux réflecteurs sismiques, et l'amplitude des évènements sismiques réfléchis vers le capteur. La trace sismique observée $\mathbf{z}$, donnée en bas de la figure 6.10, suit le modèle (6.37). Dans ce contexte, le noyau $\overline{\mathbf{k}}$ est lié à la source sismique générée. Nous utilisons ici une onde sismique passe-bande de "Ricker" (appelée aussi chapeau Mexicain [Ricker, 1940]) de taille $S = 41$ (voir figure 6.13(2)) dont les fréquences de spectre sont concentrées entre 10 et 40 Hz. Le bruit additif $\mathbf{w}$ est une réalisation d'un bruit blanc gaussien de variance σ^2.

Les séries de réflectivité étant parcimonieuses, mais limitées en amplitude, nous choisissons comme fonction de régularisation

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathsf{g}_1(\mathbf{x}) = \iota_{[\mathsf{x}_{\min}, \mathsf{x}_{\max}]^N}(\mathbf{x}), \tag{6.53}$$

où $(\mathsf{x}_{\min}, \mathsf{x}_{\max}) \in \mathbb{R}^2$ sont les amplitudes minimale et maximale de $\overline{\mathbf{x}}$. De même, l'onde sismique étant d'énergie finie, on choisit

$$(\forall \mathbf{k} \in \mathbb{R}^S) \quad \mathsf{g}_2(\mathbf{k}) = \iota_{\mathcal{C}}(\mathbf{k}), \tag{6.54}$$

où $\mathcal{C} = \{\mathbf{k} \in [\mathsf{k}_{\min}, \mathsf{k}_{\max}]^S \mid \|\mathbf{k}\| \leqslant \delta\}$, avec $\delta > 0$, et $\mathsf{k}_{\min}$ (resp. $\mathsf{k}_{\max}$) la valeur minimale (resp. maximale) de $\overline{\mathbf{k}}$.

La figure 6.12 donne les temps de reconstruction obtenus, en secondes, en utilisant l'algorithme 6.2, en fonction du nombre d'itérations internes $J_k \equiv J$, en fixant $I_k \equiv 1$, $\gamma_{\mathbf{x}}^{k,i} \equiv 1$, $\gamma_{\mathbf{k}}^{k,j} \equiv 1.9$ et le niveau de bruit $\sigma = 0.03$. Les temps de reconstruction donnés correspondent

au temps nécessaire pour que le critère d'arrêt $\|\mathbf{x}_k - \mathbf{x}_{k-1}\| \leqslant \sqrt{N} \times 10^{-6}$ soit vérifié. On peut observer que le meilleur compromis en terme de vitesse de convergence est obtenu pour une valeur intermédiaire d'itérations internes sur $\mathbf{x}_k$ égale à $J = 71$. D'autre part, nous avons pu remarquer que le choix de J influe très peu sur la qualité de reconstruction.

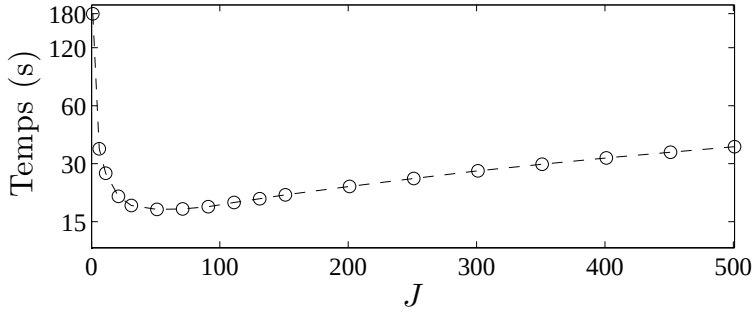


Figure 6.12 – *Temps de reconstruction en fonction du nombre d'itérations internes $J_k \equiv J$ effectuées dans l'algorithme 6.2 (moyennes obtenues pour* 30 *réalisations de bruit).*

La méthode proposée dans cette section est comparée à celle de [Krishnan *et al.*, 2011]. Les deux algorithmes sont initialisés de la même manière : $\mathbf{x}_0$ est un signal constant vérifiant $\|\mathbf{x}_0\| \leqslant \max\{|x_{\min}|, |x_{\max}|\}$, et $\mathbf{k}_0$ est un filtre gaussien centré appartenant à l'ensemble $\mathcal{C}$. Dans les tables 6.1 et 6.2 nous présentons, respectivement, les moyennes et variances obtenues pour trois niveaux de bruit $\sigma \in \{0.01, 0.02, 0.03\}$, moyennés sur 200 réalisations de bruit chacun. Les paramètres de régularisation de [Krishnan *et al.*, 2011] et $(\lambda, \alpha, \beta, \eta) \in]0, +\infty[^4$ apparaissant dans (6.41) sont choisis de manière à minimiser la norme ℓ_1 entre le signal original $\overline{\mathbf{x}}$ et le signal estimé $\widehat{\mathbf{x}}$. De plus, pour l'algorithme 6.2, nous prenons, pour tout $k \in \mathbb{N}$, $J_k = 71$ et $I_k = 1$. Bien que les deux méthodes donnent de bons résultats de reconstruction en terme d'erreurs ℓ_1 et ℓ_2, l'algorithme 6.2 donne de meilleurs résultats, pour tous les niveaux de bruits, autant pour la reconstruction de $\overline{\mathbf{x}}$ que de $\overline{\mathbf{h}}$. D'autre part, d'après la table 6.2, nous pouvons remarquer que les variances des erreurs ℓ_1 et ℓ_2 sont du même ordre de grandeur pour les deux méthodes (autour de 10% de l'erreur moyenne), ce qui permet de confirmer statistiquement la stabilité des résultats présentés dans la table 6.1. On peut remarquer de plus que, bien que les variances soient légèrement plus élevées avec l'algorithme 6.2 qu'avec [Krishnan *et al.*, 2011] pour l'estimation de $\overline{\mathbf{x}}$, le comportement inverse est observé pour l'estimation de $\overline{\mathbf{k}}$. Enfin, l'algorithme 6.2 est significativement plus rapide en moyenne, et sa vitesse de convergence est plus stable que [Krishnan *et al.*, 2011].

Finalement, nous donnons dans la figure 6.13 un résultat de reconstruction pour $\sigma = 0.03$. La figure 6.13(1) illustre l'erreur résiduelle de l'estimation parcimonieuse du signal $\overline{\mathbf{x}} - \widehat{\mathbf{x}}$, pour une réalisation de bruit donnée, où $\widehat{\mathbf{x}}$ est estimée par [Krishnan *et al.*, 2011] dans (a), et avec l'algorithme 6.2 dans (b). De même, nous donnons dans la figure 6.13(2) l'estimation de $\mathbf{k}$ pour la même réalisation de bruit.

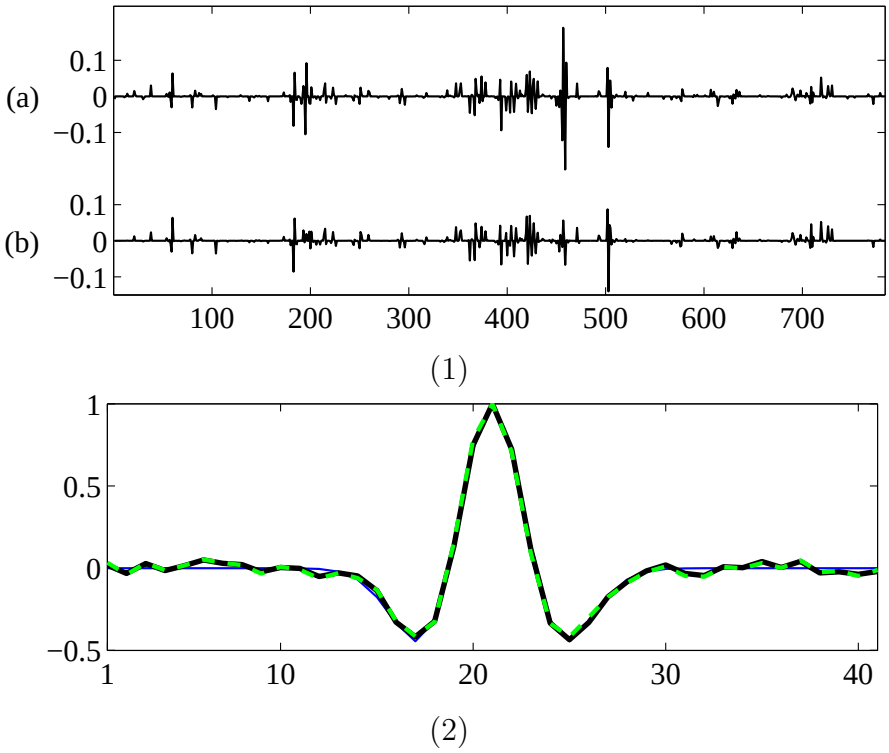


Figure 6.13 – (1) *Erreur résiduelle* $\overline{\mathbf{x}} - \hat{\mathbf{x}}$ *pour le signal estimé* $\hat{\mathbf{x}}$ *obtenu en utilisant* [Krishnan *et al.*, 2011] *(a) et l'algorithme 6.2 (b).* (2) *Noyau original* $\overline{\mathbf{k}}$ *(ligne fine continue bleue), estimé* $\hat{\mathbf{k}}$ *par l'algorithme 6.2 (ligne épaisse continue noire) et par* [Krishnan *et al.*, 2011] *(ligne discontinue épaisse verte).*

6.5 Conclusion

Dans ce chapitre nous avons traité trois problèmes inverses en traitement du signal, que nous avons résolus en utilisant l'algorithme explicite-implicite à métrique variable alterné par bloc présenté dans le chapitre 5. Nous avons vu que cet algorithme permet de résoudre des problèmes où deux objets sont à estimer conjointement (NMF et déconvolution aveugle), et permet également de traiter des problèmes de grande dimension tel que celui de reconstruction de phase présenté dans la section 6.3. Nous avons montré, sur les trois exemples de simulation, que notre méthode se compare favorablement à celles existantes. Nous avons montré de plus que, la métrique variable permet d'accélérer la convergence de l'algorithme explicite-implicite alterné par bloc dans les exemples traités. Pour finir, nous avons vu dans le dernier exemple l'intérêt d'utiliser la règle essentiellement cyclique pour la mise à jour des blocs, lorsque ces derniers ne sont pas de même nature.

Niveau de bruit (σ)			0.01	0.02	0.03
Erreur sur le signal observé $\mathbf{z}$		ℓ_2 ($\times 10^{-2}$)	7.14	7.35	7.68
		ℓ_1 ($\times 10^{-2}$)	2.85	3.44	4.09
Erreur sur le signal estimé $\widehat{\mathbf{x}}$	[Krishnan *et al.*, 2011]	ℓ_2 ($\times 10^{-2}$)	1.23	1.66	1.84
		ℓ_1 ($\times 10^{-3}$)	3.79	4.69	5.30
	Algorithme 6.2	ℓ_2 ($\times 10^{-2}$)	1.09	1.63	1.83
		ℓ_1 ($\times 10^{-3}$)	3.42	4.30	4.85
Erreur sur le noyau estimé $\widehat{\mathbf{k}}$	[Krishnan *et al.*, 2011]	ℓ_2 ($\times 10^{-2}$)	1.88	2.51	3.21
		ℓ_1 ($\times 10^{-2}$)	1.44	1.96	2.53
	Algorithme 6.2	ℓ_2 ($\times 10^{-2}$)	1.62	2.26	2.93
		ℓ_1 ($\times 10^{-2}$)	1.22	1.77	2.31
Temps de reconstruction (s)	[Krishnan *et al.*, 2011]		106	61	56
	Algorithme 6.2		56	22	18

Table 6.1 – *Moyennes sur* 200 *réalisations de bruit, pour chaque niveau de bruit, des résultats obtenus pour l'estimation de* $\overline{\mathbf{x}}$ *et* $\overline{\mathbf{k}}$ *en utilisant* [Krishnan *et al.*, 2011] *et l'algorithme 6.2 (Matlab 8 sur une machine Intel(R) Xeon(R) CPU E5-2609 v2@2.5GHz).*

Niveau de bruit (σ)			0.01	0.02	0.03
Erreur sur le signal observé $\mathbf{z}$		ℓ_2 ($\times 10^{-4}$)	3.52	7.02	10.42
		ℓ_1 ($\times 10^{-4}$)	2.98	5.55	8.06
Erreur sur le signal estimé $\widehat{\mathbf{x}}$	[Krishnan *et al.*, 2011]	ℓ_2 ($\times 10^{-3}$)	1.35	2.35	2.09
		ℓ_1 ($\times 10^{-4}$)	2.17	3.36	3.49
	Algorithme 6.2	ℓ_2 ($\times 10^{-3}$)	2.92	3.44	2.74
		ℓ_1 ($\times 10^{-4}$)	3.88	4.81	4.07
Erreur sur le noyau estimé $\widehat{\mathbf{k}}$	[Krishnan *et al.*, 2011]	ℓ_2 ($\times 10^{-3}$)	2.65	6.24	6.36
		ℓ_1 ($\times 10^{-3}$)	1.99	3.96	4.35
	Algorithme 6.2	ℓ_2 ($\times 10^{-3}$)	2.94	2.75	3.43
		ℓ_1 ($\times 10^{-3}$)	2.04	2.25	2.86
Temps de reconstruction (s)	[Krishnan *et al.*, 2011]		54	45	46
	Algorithme 6.2		11	4	6

Table 6.2 – *Variances sur* 200 *réalisations de bruit, pour chaque niveau de bruit, des résultats obtenus pour l'estimation de* $\overline{\mathbf{x}}$ *et* $\overline{\mathbf{k}}$ *en utilisant* [Krishnan *et al.*, 2011] *et l'algorithme 6.2.*

Chapitre 7

Algorithmes primaux-duaux alternés et distribués pour les problèmes d'inclusion monotone et d'optimisation convexe

Sommaire

7.1 Introduction

Ces dernières années, les méthodes primales-duales ont suscité un intérêt croissant. Elles permettent de minimiser des sommes de fonctions convexes (voir section 2.4.4 et [Komodakis et Pesquet, 2014]) mais aussi de trouver le zéro d'une somme d'opérateurs monotones [Bauschke et Combettes, 2011]. Lorsque plusieurs opérateurs linéaires apparaissent dans la formulation du problème étudié, résoudre conjointement les formes primales et duales permet de construire des stratégies d'optimisation où les opérateurs linéaires n'ont pas besoin d'être inversés, ce qui offre un avantage indéniable en terme de complexité algorithmique lorsque l'on traite des problèmes de grande taille (voir par exemple [Becker et Combettes, 2014; Couprie *et al.*, 2013; Harizanov *et al.*, 2013; Jezierska *et al.*, 2012; Pustelnik *et al.*, 2014; Repetti *et al.*, 2012; Teuber *et al.*, 2013]).

Dans la section 2.4.4, nous avons vu qu'il existe plusieurs classes d'algorithmes primaux-duaux. Nous nous concentrerons sur la première classe, basée sur l'algorithme explicite-implicite, déjà étudiée dans de nombreux travaux [Chambolle et Pock, 2010; Chen *et al.*, 2013; Combettes *et al.*, 2014, 2010; Combettes et Vũ, 2014; Condat, 2013; Esser *et al.*, 2010; Goldstein *et al.*, 2013; He et Yuan, 2012; Loris et Verhoeven, 2011; Pock et Chambolle, 2011; Vũ, 2013]. Nous avons vu que pour minimiser une somme de fonctions convexes, ces algorithmes exploitent les gradients des fonctions différentiables et les opérateurs proximaux des autres fonctions. Dans le cas des opérateurs monotones, cela signifie que l'opérateur est appliqué de façon explicite s'il est cocoercif, sinon sa résolvante est utilisée.

La plupart des algorithmes proposés dans les travaux mentionnés ci-dessus consistent à éclater le problème original en une somme de termes plus simples dont les opérateurs associés peuvent être traités individuellement, de façon parallèle, à chaque itération de l'algorithme. Notre objectif dans ce chapitre est de rendre plus flexible les algorithmes primaux-duaux existants, en s'autorisant à activer, à chaque itération, une sous-partie seulement des opérateurs impliqués. Pour cela, nous nous baserons sur les travaux récents de [Combettes et Pesquet, 2015], proposant des approches basées sur des techniques de balayage aléatoire applicables à des algorithmes générant des suites (quasi-)Fejér monotones. Les algorithmes résultants de ces approches aléatoires sont tolérants à des erreurs stochastiques satisfaisant une condition de sommabilité.

Dans ce chapitre, nous donnerons des versions alternées par bloc de deux variantes d'algorithmes primaux-duaux basés sur l'approche explicite-implicite. Comme nous l'avons vu dans les chapitres 5 et 6, les méthodes par bloc présentent des avantages algorithmiques en terme de vitesse de convergence et d'allocation de mémoire. Nous verrons de plus dans ce chapitre qu'elles permettent de développer des stratégies distribuées. Plus précisément, nous allons nous intéresser à des problèmes multi-agents où les mises à jours peuvent être effectuées de manière asynchrone sur le voisinage d'un nombre limité d'agents. Nous montrerons que les algorithmes développés permettent de traiter des problèmes variationnels et d'inclusion monotone, et nous ferons le lien avec les méthodes existantes. Notons que, dans le cas variationnel, des méthodes primales-duales distribuées existent, mettant en œuvre des étapes de sous-gradients [Chang *et al.*, 2014; Yuan *et al.*, 2011] (voir aussi [Towfic et Sayed, 2014] pour des applications à des réseaux de données). D'une manière générale, les méthodes de sous-gradients convergent sous la condition que le pas tend vers zero (voir section 2.4.1). Cependant, l'utilisation de l'opérateur proximal, qui peut être vu comme une étape implicite du sous-gradient, permet d'avoir des hypothèses moins restrictives sur le pas. Par exemple, la convergence des itérées peut être établie pour des pas constants.

La suite du chapitre est organisée de la façon suivante : dans la section 7.2 nous ferons des rappels sur la théorie des opérateurs monotones et donnerons nos notations. Dans la section 7.3, nous proposerons des algorithmes primaux-duaux alternés par bloc appliqués à des problèmes d'inclusion monotone et à des problèmes variationnels. Dans la section 7.4 nous traiterons une application en reconstruction de maillage 3D. Nous développerons ensuite des algorithmes distribués dans la section 7.5, permettant de résoudre des problèmes d'inclusion monotone et des problèmes variationnels. Finalement, nous donnerons quelques conclusions dans la section 7.6.

7.2 Opérateurs monotones et optimisation

Nous ne donnerons dans cette section que les définitions et propriétés qui seront utiles pour notre étude. Pour de plus amples informations sur les opérateurs monotones, le lecteur peut se référer à [Bauschke et Combettes, 2011], et pour les notions de probabilités, à [Fortet, 1995].

7.2.1 Notations

Dans ce chapitre, nous notons $(\Omega, \mathcal{F}, \mathsf{P})$ l'espace de probabilité sous-jacent. Soit $\mathcal{H}$ un espace de Hilbert réel séparable. On note $\mathcal{B}$ sa σ-algèbre de Borel. Une variable aléatoire de $\mathcal{H}$ est une application mesurable $x\colon (\Omega, \mathcal{F}) \to (\mathcal{H}, \mathcal{B})$. La plus petite σ-algèbre générée par une famille Φ de variables aléatoires est noté $\sigma(\Phi)$. De plus, l'espérance est notée $\mathsf{E}(\cdot)$.

Soit $\mathcal{G}$ un espace de Hilbert réel. Soit $\mathsf{L} \in \mathcal{B}(\mathcal{H}, \mathcal{G})$ [1], on note son opérateur adjoint L^*, son inverse L^{-1} et sa racine carrée $\mathsf{L}^{1/2}$. Si $\mathsf{L} \in \mathcal{S}^+(\mathcal{H})$ [2], alors L est un isomorphisme et son inverse L^{-1} est aussi dans $\mathcal{S}^+(\mathcal{H})$.

Nous notons $2^{\mathcal{H}}$ l'ensemble des parties de l'ensemble $\mathcal{H}$. Soit $\mathsf{A}\colon \mathcal{H} \to 2^{\mathcal{H}}$. Si, pour tout $\mathsf{x} \in \mathcal{H}$, Ax est un singleton, alors A est un opérateur univalué.

Définition 7.1. *Soit* $\mathsf{A}\colon \mathcal{H} \to 2^{\mathcal{H}}$.

(i) *L'*ensemble des zéros *de* A *est donné par*

$$\operatorname{zer} \mathsf{A} = \{\mathsf{x} \in \mathcal{H} \mid 0 \in \mathsf{Ax}\}.$$

(ii) *L'*inverse *de* A *est défini par*

$$\mathsf{A}^{-1}\colon \mathcal{H} \mapsto 2^{\mathcal{H}}\colon \mathsf{u} \mapsto \{\mathsf{x} \in \mathcal{H} \mid \mathsf{u} \in \mathsf{Ax}\}.$$

(iii) *On dit que* A *est un opérateur* monotone *si*

$$(\forall (\mathsf{x}, \mathsf{y}) \in \mathcal{H}^2)(\forall \mathsf{u} \in \mathsf{Ax})(\forall \mathsf{v} \in \mathsf{Ay}) \quad \langle \mathsf{x} - \mathsf{y} \mid \mathsf{u} - \mathsf{v} \rangle \geqslant 0.$$

1. Rappelons que $\mathcal{B}(\mathcal{H}, \mathcal{G})$ correspond à l'ensemble des opérateurs linéaires bornés de $\mathcal{H}$ vers $\mathcal{G}$.

2. Rappelons que $\mathcal{S}(\mathcal{H})$ correspond à l'ensemble des opérateurs $\mathsf{L} \in \mathcal{B}(\mathcal{H}) = \mathcal{B}(\mathcal{H}, \mathcal{H})$ auto-adjoints. De plus, $\mathcal{S}^+(\mathcal{H})$ est l'ensemble des opérateurs $\mathsf{L} \in \mathcal{S}(\mathcal{H})$ fortement positifs.

(iv) *On dit que* A *est un opérateur* maximal monotone *s'il est monotone et s'il n'existe pas d'autre opérateur monotone dont le graphe est inclus dans celui de* A.

(v) *On dit que* A *est un opérateur* β-fortement monotone, *pour* $\beta \in]0, +\infty[$, *si*

$$(\forall (\mathsf{x}, \mathsf{y}) \in \mathcal{H}^2)(\forall \mathsf{u} \in \mathsf{A}\mathsf{x})(\forall \mathsf{v} \in \mathsf{A}\mathsf{y}) \quad \langle \mathsf{x} - \mathsf{y} \mid \mathsf{u} - \mathsf{v} \rangle \geqslant \beta \|\mathsf{x} - \mathsf{y}\|^2.$$

(vi) *La* résolvante *de* A *est définie par*

$$\mathsf{J}_{\mathsf{A}} \colon \mathcal{H} \to 2^{\mathcal{H}} \colon \mathsf{x} \mapsto \mathsf{J}_{\mathsf{A}}\mathsf{x} = (\mathrm{Id} + \mathsf{A})^{-1}\mathsf{x}.$$

(vii) *Soit* $\mathsf{C} \colon \mathcal{H} \to 2^{\mathcal{H}}$. *La* somme parallèle *entre* A *et* C *est définie par*

$$\mathsf{A} \,\square\, \mathsf{C} = (\mathsf{A}^{-1} + \mathsf{C}^{-1})^{-1}.$$

L'élément neutre de la somme parallèle est le cône normal à $\{0\}$, *noté* $\mathsf{N}_{\{0\}}$.

Définition 7.2. *Soit* B *un opérateur univalué de* $\mathcal{H}$ *dans* $\mathcal{H}$.

(i) *L'opérateur* B *est dit* β-cocoercif, *pour* $\beta \in]0, +\infty[$, *si*

$$(\forall (\mathsf{x}, \mathsf{y}) \in \mathcal{H}^2) \quad \langle \mathsf{x} - \mathsf{y} \mid \mathsf{B}\mathsf{x} - \mathsf{B}\mathsf{y} \rangle \geqslant \beta \|\mathsf{B}\mathsf{x} - \mathsf{B}\mathsf{y}\|^2.$$

(ii) *L'opérateur* B *est dit* fermement contractant *s'il est* 1*-cocoercif.*

(iii) *L'opérateur* B *est dit* α-moyenné, *pour* $\alpha \in]0, 1[$, *si*

$$(\forall (\mathsf{x}, \mathsf{y}) \in \mathcal{H}^2) \quad \|\mathsf{B}\mathsf{x} - \mathsf{B}\mathsf{y}\|^2 \leqslant \|\mathsf{x} - \mathsf{y}\|^2 - \frac{1-\alpha}{\alpha}\|(\mathrm{Id} - \mathsf{B})\mathsf{x} - (\mathrm{Id} - \mathsf{B})\mathsf{y}\|^2$$

La propriété suivante permet de faire le lien entre certaines des définitions données ci-dessus.

Propriété 7.1.

(i) *Soient* $\mathsf{B} \colon \mathcal{H} \to \mathcal{H}$ *et* $\beta \in]0, +\infty[$. *L'opérateur* B *est* β*-cocoercif si et seulement si* $\mathsf{B}^{-1} \colon \mathcal{H} \to 2^{\mathcal{H}}$ *est* β*-fortement monotone.*

(ii) *Un opérateur* $\mathsf{A} \colon \mathcal{H} \to 2^{\mathcal{H}}$ *est maximal monotone si et seulement si sa résolvante est un opérateur fermement contractant de* $\mathcal{H}$ *vers* $\mathcal{H}$.

Dans le chapitre 2 nous avons énoncé le théorème de Moreau pour les fonctions de $\Gamma_0(\mathcal{H})$. Le théorème suivant le généralise au cas des opérateurs maximaux monotones.

Théorème 7.1. (Formule de décomposition de Moreau)[Combettes et Vũ, 2014, Ex. 3.9] *Soient* $\mathsf{A} \colon \mathcal{H} \to 2^{\mathcal{H}}$ *un opérateur maximal monotone,* $\mathsf{U} \in \mathcal{S}^+(\mathcal{H})$ *et* $\gamma \in]0, +\infty[$. *On note alors* $\mathsf{J}_{\gamma\mathsf{U}\mathsf{A}} \colon \mathcal{H} \to \mathcal{H}$ *la résolvante de* A *relativement à* $\gamma\mathsf{U}$, *et on a*

$$\mathsf{J}_{\gamma\mathsf{U}\mathsf{A}} = \mathsf{U}^{1/2}\mathsf{J}_{\gamma\mathsf{U}^{1/2}\mathsf{A}\mathsf{U}^{1/2}}\mathsf{U}^{-1/2} = \mathrm{Id} - \gamma\mathsf{U}\mathsf{J}_{\gamma^{-1}\mathsf{U}^{-1}\mathsf{A}^{-1}}(\gamma^{-1}\mathsf{U}^{-1}\cdot). \tag{7.1}$$

Nous allons maintenant nous intéresser aux liens existants entre les opérateurs monotones et les fonctions convexes.

Propriété 7.2. *Soit* $\mathsf{f}\colon \mathcal{H} \to]-\infty, +\infty]$.

(i) *Si* $\mathsf{f} \in \Gamma_0(\mathcal{H})$, *le sous-différentiel de Moreau* $\partial\mathsf{f}$ *donné par la Définition 2.8 est un opérateur maximal monotone.*

(ii) *Si* f *est une fonction propre et* β*-fortement convexe, pour* $\beta \in]0, +\infty[$, *alors* $\partial\mathsf{f}$ *est* β*-fortement monotone.*

(iii) **Théorème de Baillon-Haddad.** *Soit* $\beta \in]0, +\infty[$. *Si* f *est différentiable et convexe, alors le gradient de* f, $\nabla\mathsf{f}$, *est* β^{-1}*-Lipschitz si et seulement s'il est* β*-cocoercif.*

(iv) *Soit* $\mathsf{U} \in \mathcal{S}^+(\mathcal{H})$. *Si* $\mathsf{f} \in \Gamma_0(\mathcal{H})$, *alors* $\operatorname{prox}_{\mathsf{U},\mathsf{f}} = \mathsf{J}_{\mathsf{U}^{-1}\partial\mathsf{f}}$.

Soient $(\mathcal{G}_i)_{1\leqslant i\leqslant m}$ des espaces de Hilbert réels. Nous notons $\boldsymbol{\mathcal{G}} = \mathcal{G}_1 \oplus \cdots \oplus \mathcal{G}_m$ leur somme directe de Hilbert, i.e. l'espace produit muni du produit scalaire :

$$(\forall \mathbf{x} = (\mathsf{x}^{(i)})_{1\leqslant i\leqslant m} \in \boldsymbol{\mathcal{G}})(\forall \mathbf{y} = (\mathsf{y}^{(i)})_{1\leqslant i\leqslant m} \in \boldsymbol{\mathcal{G}}) \quad (\mathbf{x}, \mathbf{y}) \mapsto \sum_{i=1}^{m} \left\langle \mathsf{x}^{(i)} \mid \mathsf{y}^{(i)} \right\rangle,$$

où, pour tout $i \in \{1, \ldots, m\}$, $\mathsf{x}^{(i)} \in \mathcal{G}_i$ et $\mathsf{y}^{(i)} \in \mathcal{G}_i$.

De plus, nous définissons $\mathbb{D}_m = \{0,1\}^m \smallsetminus \{\mathbf{0}\}$ l'ensemble des codes binaires non nuls de longueur m.

7.2.2 Algorithme explicite-implicite alterné aléatoirement par bloc

Dans un premier temps nous allons rappeler les résultats principaux présentés dans [Combettes et Pesquet, 2015] sur lesquels nous nous baserons dans ce chapitre.

On suppose que $\mathcal{K}_1, \ldots, \mathcal{K}_m$ sont des espaces de Hilbert réels séparables, où $m \in \mathbb{N}^*$, et $\boldsymbol{\mathcal{K}} = \mathcal{K}_1 \oplus \cdots \oplus \mathcal{K}_m$ correspond à leur somme directe de Hilbert.

On définit, pour tout $k \in \mathbb{N}$,

- $\mathbf{S}_k\colon \boldsymbol{\mathcal{H}} \to \boldsymbol{\mathcal{H}}$ un opérateur β_k-moyenné, pour $\beta_k \in]0,1[$,
- et $\mathbf{T}_k\colon \boldsymbol{\mathcal{H}} \to \boldsymbol{\mathcal{H}}\colon \mathbf{z} \mapsto (\mathsf{T}_{i,k}\mathbf{z})_{1\leqslant i\leqslant m}$ un opérateur α_k-moyenné, pour $\alpha_k \in]0,1[$, avec $(\forall i \in \{1, \ldots, m\})$ $\mathsf{T}_{i,k}\colon \boldsymbol{\mathcal{H}} \to \mathcal{H}_i$.

L'objectif est de trouver

$$\mathbf{z} \in \widetilde{\mathbf{Z}} = \bigcap_{k\in\mathbb{N}} \operatorname{Fix}\left(\mathbf{T}_k \circ \mathbf{S}_k\right). \tag{7.2}$$

Une méthode permettant de résoudre le problème (7.2) est donnée par l'algorithme 7.1.

Algorithme 7.1 [Combettes et Pesquet, 2015, Algo. (4.1)]

Initialisation : Soient $\boldsymbol{z}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{K}}$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_m$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0, +\infty[$.

Itérations :

Pour $k = 0, 1, \ldots$

$\quad \boldsymbol{r}_k = \mathbf{S}_k \boldsymbol{z}_k + \boldsymbol{b}_k,$

$\quad$ pour $i = 1, \ldots, m$

$\quad\quad z_{k+1}^{(i)} = z_k^{(i)} + \lambda_k \varepsilon_k^{(i)} \left(\mathsf{T}_{i,k} \boldsymbol{r}_k + a_k^{(i)} - z_k^{(i)} \right).$

Le théorème suivant, issu de [Combettes et Pesquet, 2015, Thm. 4.1], donne les garanties de convergence de l'algorithme 7.1.

Théorème 7.2. [Combettes et Pesquet, 2015, Thm. 4.1] *Soit $(\boldsymbol{z}_k)_{k\in\mathbb{N}}$ la suite générée par l'algorithme 7.1. Supposons que $\widetilde{\mathbf{Z}}$ est non vide, $\sup_{k\in\mathbb{N}} \alpha_k < 1$ et $\sup_{k\in\mathbb{N}} \beta_k < 1$. De plus, supposons que, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0, 1]$ et que $\inf_{k\in\mathbb{N}} \lambda_k > 0$. Posons, pour tout $k \in \mathbb{N}$, $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ et $\boldsymbol{\mathcal{Z}}_k = \sigma(\boldsymbol{z}_0, \ldots, \boldsymbol{z}_k)$. Si*

(i) $\sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{a}_k\|^2 \,|\, \boldsymbol{\mathcal{Z}}_k)} < +\infty$ *et* $\sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2 \,|\, \boldsymbol{\mathcal{Z}}_k)} < +\infty$ P*-presque sûrement,*

(ii) *pour tout $k \in \mathbb{N}$, $\boldsymbol{\mathcal{E}}_k$ et $\boldsymbol{\mathcal{Z}}_k$ sont indépendants, et* $(\forall i \in \{1, \ldots, m\})$ $\mathsf{P}[\varepsilon_0^{(i)} = 1] > 0$,

on a alors P-presque sûrement

$$(\forall \mathbf{z} \in \widetilde{\mathbf{Z}}) \quad \mathbf{T}_k(\mathbf{S}_k \boldsymbol{z}_k) - \mathbf{S}_k \boldsymbol{z}_k + \mathbf{S}_k \mathbf{z} \to \mathbf{z}, \tag{7.3}$$

$$(\forall \mathbf{z} \in \widetilde{\mathbf{Z}}) \quad \boldsymbol{z}_k - \mathbf{S}_k \boldsymbol{z}_k + \mathbf{S}_k \mathbf{z} \to \mathbf{z}. \tag{7.4}$$

De plus, si

(iii) *l'ensemble des points de faible adhérence séquentielle de $(\boldsymbol{z}_k)_{k\in\mathbb{N}}$ est inclus dans $\widetilde{\mathbf{Z}}$ P-presque sûrement,*

alors, $(\boldsymbol{z}_k)_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire à valeurs dans $\widetilde{\mathbf{Z}}$.

Les algorithmes présentés dans ce chapitre sont issus des itérations de l'algorithme explicite-implicite pour la résolution de problèmes d'inclusions monotones [Combettes et Wajs, 2005] (voir [Attouch *et al.*, 2010b] pour des exemples de problèmes pouvant être résolus par cette méthode). Une version alternée aléatoirement par bloc a récemment été proposée dans [Combettes et Pesquet, 2015, Sec. 5.2] (une version pour le cas variationnel a aussi été proposée dans [Necoara et Patrascu, 2014; Richtárik et Talác, 2014]). Nous allons montrer comment inclure des opérateurs de préconditionnement dans l'algorithme explicite-implicite alterné aléatoirement par bloc grâce à un changement de métrique.

Soient $\mathbf{Q}\colon \mathcal{K} \to 2^{\mathcal{K}}$ un opérateur maximal monotone et $\mathbf{R}\colon \mathcal{K} \to \mathcal{K}$ un opérateur cocoercif. On veut trouver

$$\mathbf{z} \in \mathbf{Z} = \operatorname{zer}(\mathbf{Q} + \mathbf{R}).$$

Nous proposons d'utiliser l'algorithme 7.2 ci-dessous pour résoudre ce problème d'inclusion, où nous posons, pour tout $\mathbf{V} \in \mathcal{S}^+(\mathcal{K})$ et pour tout $k \in \mathbb{N}$, $\mathbf{J}_{\gamma_k \mathbf{V}\mathbf{Q}}\colon \mathbf{z} \mapsto (\mathsf{T}_{i,k}\mathbf{z})_{1\leqslant i\leqslant m}$ avec $(\forall i \in \{1,\dots,m\})$ $\mathsf{T}_{i,k}\colon \mathcal{K} \to \mathcal{K}_i$.

Algorithme 7.2 Algorithme préconditionné explicite-implicite alterné aléatoirement par bloc

Initialisation : Soient $\boldsymbol{z}_0$, $(\boldsymbol{s}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{t}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\mathcal{K}$ et $\mathbf{V} \in \mathcal{S}^+(\mathcal{K})$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_m$. Soient, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$ et $\gamma_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

$\boldsymbol{r}_k = \mathbf{V}\mathbf{R}\boldsymbol{z}_k$,

pour $i = 1, \dots, m$

$z_{k+1}^{(i)} = z_k^{(i)} + \lambda_k \varepsilon_k^{(i)} \left(\mathsf{T}_{i,k}(\boldsymbol{z}_k - \gamma_k \boldsymbol{r}_k + \boldsymbol{s}_k) + t_k^{(i)} - z_k^{(i)}\right)$.

Les garanties de convergence de l'algorithme 7.2 sont données dans la proposition suivante, dont la démonstration se déduit du théorème 7.2 et de la démonstration de [Combettes et Pesquet, 2015, Prop. 5.9].

Proposition 7.1. *Soit $(\boldsymbol{z}_k)_{k\in\mathbb{N}}$ la suite générée par l'algorithme 7.2. Supposons que $\mathbf{Z}$ est non vide, et que $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ est ϑ-cocoercif, avec $\vartheta \in]0,+\infty[$. Supposons, de plus, que $\inf_{k\in\mathbb{N}} \gamma_k > 0$, $\sup_{k\in\mathbb{N}} \gamma_k < 2\vartheta$, et que, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,1]$ et $\inf_{k\in\mathbb{N}} \lambda_k > 0$. Posons $(\forall k \in \mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ et $\boldsymbol{\mathcal{Z}}_k = \sigma(\boldsymbol{z}_0, \dots, \boldsymbol{z}_k)$. Si*

(i) $\sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{s}_k\|^2 \,|\, \boldsymbol{\mathcal{Z}}_k)} < +\infty$ *et* $\sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{t}_k\|^2 \,|\, \boldsymbol{\mathcal{Z}}_k)} < +\infty$ P*-presque sûrement,*

(ii) *pour tout $k \in \mathbb{N}$, $\boldsymbol{\mathcal{E}}_k$ et $\boldsymbol{\mathcal{Z}}_k$ sont indépendants et $(\forall i \in \{1,\dots,m\})$ $\mathsf{P}[\varepsilon_0^{(i)} = 1] > 0$,*

alors $(\boldsymbol{z}_k)_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire à valeurs dans $\mathbf{Z}$.

Démonstration. On a $\mathbf{Z} = \operatorname{zer}(\mathbf{V}\mathbf{Q} + \mathbf{V}\mathbf{R}) \neq \varnothing$. Puisque $\mathbf{V} \in \mathcal{S}^+(\mathcal{K})$, on peut changer la métrique de la norme de $\mathcal{K}$:

$$(\forall \mathbf{z} \in \mathcal{K}) \qquad \|\mathbf{z}\|_{\mathbf{V}^{-1}} = \sqrt{\left\langle \mathbf{z} \mid \mathbf{V}^{-1}\mathbf{z} \right\rangle}.$$

Notons de même $\langle \cdot \mid \cdot \rangle_{\mathbf{V}^{-1}}$ le produit scalaire associé. Dans cet espace renormé, $\mathbf{V}\mathbf{Q}$ est maximal

monotone. De plus,

$$\begin{aligned}(\forall (\mathbf{z},\mathbf{z}') \in \mathcal{K}^2) \quad \|\mathbf{V}\mathbf{R}\mathbf{z}-\mathbf{V}\mathbf{R}\mathbf{z}'\|_{\mathbf{V}^{-1}}^2 &= \|\mathbf{V}^{1/2}\mathbf{R}\mathbf{z}-\mathbf{V}^{1/2}\mathbf{R}\mathbf{z}'\|^2\\ &\leqslant \vartheta^{-1}\left\langle \mathbf{V}^{-1/2}\mathbf{z}-\mathbf{V}^{-1/2}\mathbf{z}' \mid \mathbf{V}^{1/2}\mathbf{R}\mathbf{z}-\mathbf{V}^{1/2}\mathbf{R}\mathbf{z}'\right\rangle\\ &= \vartheta^{-1}\langle \mathbf{z}-\mathbf{z}' \mid \mathbf{R}\mathbf{z}-\mathbf{R}\mathbf{z}'\rangle\\ &= \vartheta^{-1}\langle \mathbf{z}-\mathbf{z}' \mid \mathbf{V}\mathbf{R}\mathbf{z}-\mathbf{V}\mathbf{R}\mathbf{z}'\rangle_{\mathbf{V}^{-1}},\end{aligned}$$

ce qui montre que $\mathbf{V}\mathbf{R}$ est ϑ-cocoercif dans $(\mathcal{K}, \|\cdot\|_{\mathbf{V}^{-1}})$. On peut alors utiliser un algorithme explicite-implicite pour trouver un élément de $\mathbf{Z}$ en itérant, pour tout $k \in \mathbb{N}$,

$$\boldsymbol{z}_{k+1} = \mathbf{J}_{\gamma_k \mathbf{V}\mathbf{Q}}\left(\boldsymbol{z}_k - \gamma_k \mathbf{V}\mathbf{R}\boldsymbol{z}_k\right).$$

Dans $(\mathcal{K}, \|\cdot\|_{\mathbf{V}^{-1}})$, l'opérateur $\mathbf{T}_k = \mathbf{J}_{\gamma_k \mathbf{V}\mathbf{Q}}$ est fermement contractant (et donc 1/2-moyenné), l'opérateur $\mathbf{S}_k = \mathbf{Id} - \gamma_k \mathbf{V}\mathbf{R}$ est $\gamma_k/(2\vartheta)$-moyenné [Bauschke et Combettes, 2011, Prop. 4.33], et on a $\mathbf{Z} = \bigcap_{k\in\mathbb{N}} \operatorname{Fix} \mathbf{T}_k \circ \mathbf{S}_k$ [Bauschke et Combettes, 2011, Prop. 25.1(iv)]. L'algorithme 7.2 apparaît alors comme un cas particulier de l'algorithme 7.1. Remarquons que la condition (i) induit

$$\begin{aligned}\sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{s}_k\|_{\mathbf{V}^{-1}}^2 \,|\, \mathcal{Z}_k)} &\leqslant \sqrt{\|\mathbf{V}^{-1}\|} \sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{s}_k\|^2 \,|\, \mathcal{Z}_k)} < +\infty\\ \sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{t}_k\|_{\mathbf{V}^{-1}}^2 \,|\, \mathcal{Z}_k)} &\leqslant \sqrt{\|\mathbf{V}^{-1}\|} \sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{t}_k\|^2 \,|\, \mathcal{Z}_k)} < +\infty\end{aligned}$$

et que la convergence faible au sens de $\langle\cdot \mid \cdot\rangle$ et $\langle\cdot \mid \cdot\rangle_{\mathbf{V}^{-1}}$ sont équivalentes.

Afin d'appliquer le théorème 7.2, il reste à démontrer la condition (iii) de ce théorème. Pour cela, il suffit d'utiliser un raisonnement analogue à celui de la démonstration de [Combettes et Pesquet, 2015, Thm. 5.9]. Les résultats (7.3) et (7.4) assurent l'existence de $\widetilde{\Omega} \in \mathcal{F}$ vérifiant $\mathsf{P}(\widetilde{\Omega}) = 1$ et

$$(\forall \omega \in \widetilde{\Omega})(\forall \mathbf{z} \in \widetilde{\mathbf{Z}}) \quad \begin{cases}\mathbf{T}_k(\mathbf{S}_k \boldsymbol{z}_k(\omega)) - \mathbf{S}_k \boldsymbol{z}_k(\omega) + \mathbf{S}_k \mathbf{z} \to \mathbf{z}\\ \mathbf{S}_k \boldsymbol{z}_k(\omega) - \boldsymbol{z}_k(\omega) - \mathbf{S}_k \mathbf{z} \to -\mathbf{z}.\end{cases}$$

Par conséquent, puisque $\inf_{k\in\mathbb{N}} \gamma_k > 0$ et $\mathbf{V} \in \mathcal{S}^+(\mathcal{K})$,

$$(\forall \omega \in \widetilde{\Omega})(\forall \mathbf{z} \in \widetilde{\mathbf{Z}}) \quad \begin{cases}\mathbf{J}_{\gamma_k \mathbf{V}\mathbf{Q}}(\boldsymbol{z}_k(\omega) - \gamma_k \mathbf{V}\mathbf{R}\boldsymbol{z}_k(\omega)) - \boldsymbol{z}_k(\omega) = \mathbf{T}_k(\mathbf{S}_k \boldsymbol{z}_k(\omega)) - \boldsymbol{z}_k(\omega) \to \mathbf{0}\\ \mathbf{R}\boldsymbol{z}_k(\omega) \to \mathbf{R}\mathbf{z}.\end{cases} \tag{7.5}$$

Posons, de plus,

$$(\forall k \in \mathbb{N}) \quad \begin{cases}\boldsymbol{y}_k = \mathbf{J}_{\gamma_k \mathbf{V}\mathbf{Q}}(\boldsymbol{z}_k - \gamma_k \mathbf{V}\mathbf{R}\boldsymbol{z}_k)\\ \boldsymbol{u}_k = \gamma_k^{-1}\mathbf{V}^{-1}(\boldsymbol{z}_k - \boldsymbol{y}_k) - \mathbf{R}\boldsymbol{z}_k.\end{cases} \tag{7.6}$$

Nous pouvons alors déduire de (7.5) que

$$(\forall \omega \in \widetilde{\Omega})(\forall \mathbf{z} \in \widetilde{\mathbf{Z}}) \quad \begin{cases}\boldsymbol{z}_k(\omega) - \boldsymbol{y}_k(\omega) \to \mathbf{0}\\ \boldsymbol{u}_k(\omega) \to -\mathbf{R}\mathbf{z}.\end{cases} \tag{7.7}$$

Pour établir la condition (iii) il est alors suffisant de fixer $\mathbf{z} \in \mathbf{Z}$, $\widehat{\mathbf{z}} \in \mathcal{K}$, de définir une sous-suite $(n_k)_{k\in\mathbb{N}}$ de $\mathbb{N}$ strictement croissante, de poser $\omega \in \widetilde{\Omega}$ tel que $\boldsymbol{z}_{n_k}(\omega) \rightharpoonup \widehat{\mathbf{z}}$ et de montrer que $\widehat{\mathbf{z}} \in \mathbf{Z}$.

Nous déduisons de (7.5) que $\mathbf{R}\boldsymbol{z}_{n_k}(\omega) \rightharpoonup \mathbf{R}\mathbf{z}$. Ainsi, d'après [Bauschke et Combettes, 2011, Ex. 20.28] $\mathbf{R}$ étant maximal monotone, on peut déduire de [Bauschke et Combettes, 2011, Prop. 20.33(ii)] que $\mathbf{R}\mathbf{z} = \mathbf{R}\widehat{\mathbf{z}}$. Nous pouvons de plus déduire de (7.7) que $\boldsymbol{y}_{k_n}(\omega) \rightharpoonup \widehat{\mathbf{z}}$ et $\boldsymbol{u}_{n_k} \rightharpoonup -\mathbf{R}\widehat{\mathbf{z}} = -\mathbf{R}\mathbf{z}$. L'équation (7.6) assure que $(\forall k \in \mathbb{N})$ $(\boldsymbol{y}_{n_k}(\omega), \boldsymbol{u}_{n_k}(\omega)) \in \operatorname{Graph}\mathbf{Q}$. On déduit alors de [Bauschke et Combettes, 2011, Prop. 20.33(ii)] que $-\mathbf{R}\widehat{\mathbf{z}} \in \mathbf{Q}\widehat{\mathbf{z}}$, i.e. $\widehat{\mathbf{z}} \in \mathbf{Z}$. □

Remarque 7.1.

(i) *Soient $\widetilde{\mathcal{K}}$ un espace de Hilbert réel séparable, $\mathbf{L} \in \mathcal{B}(\mathcal{K}, \widetilde{\mathcal{K}})$, et $\widetilde{\mathbf{R}}\colon \widetilde{\mathcal{K}} \to \widetilde{\mathcal{K}}$ un opérateur $\widetilde{\vartheta}$-cocoercif, avec $\widetilde{\vartheta} \in]0, +\infty[$. Si $\mathbf{R} = \mathbf{L}^*\widetilde{\mathbf{R}}\mathbf{L}$, alors $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ est ϑ-cocoercif pour tout $\mathbf{V} \in \mathcal{S}^+(\mathcal{K})$ vérifiant*

$$\vartheta\|\mathbf{L}\mathbf{V}\mathbf{L}^*\| = \widetilde{\vartheta}.$$

En effet, si $\widetilde{\mathbf{R}}$ est $\widetilde{\vartheta}$-cocoercif alors pour tout $(\mathbf{x}, \mathbf{y}) \in \mathcal{K}^2$,

$$\begin{aligned} &\left\langle \mathbf{x} - \mathbf{y} \mid \widetilde{\mathbf{R}}\mathbf{x} - \widetilde{\mathbf{R}}\mathbf{y} \right\rangle \geqslant \widetilde{\vartheta}\|\widetilde{\mathbf{R}}\mathbf{x} - \widetilde{\mathbf{R}}\mathbf{y}\|^2 \\ \Leftrightarrow \quad &\left\langle \mathbf{L}\mathbf{V}^{1/2}\mathbf{x} - \mathbf{L}\mathbf{V}^{1/2}\mathbf{y} \mid \widetilde{\mathbf{R}}\mathbf{L}\mathbf{V}^{1/2}\mathbf{x} - \widetilde{\mathbf{R}}\mathbf{L}\mathbf{V}^{1/2}\mathbf{y} \right\rangle \geqslant \widetilde{\vartheta}\|\widetilde{\mathbf{R}}\mathbf{L}\mathbf{V}^{1/2}\mathbf{x} - \widetilde{\mathbf{R}}\mathbf{L}\mathbf{V}^{1/2}\mathbf{y}\|^2. \end{aligned}$$

On obtient donc

$$\begin{aligned} &\left\langle \mathbf{x} - \mathbf{y} \mid \mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}\mathbf{x} - \mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}\mathbf{y} \right\rangle \\ &\geqslant \vartheta\|\mathbf{V}^{1/2}\mathbf{L}^*\|^2\|\widetilde{\mathbf{R}}\mathbf{L}\mathbf{V}^{1/2}\mathbf{x} - \widetilde{\mathbf{R}}\mathbf{L}\mathbf{V}^{1/2}\mathbf{y}\|^2 \\ &\geqslant \vartheta\|\mathbf{V}^{1/2}\mathbf{L}^*(\widetilde{\mathbf{R}}\mathbf{L}\mathbf{V}^{1/2}\mathbf{x} - \widetilde{\mathbf{R}}\mathbf{L}\mathbf{V}^{1/2}\mathbf{y})\|^2 = \vartheta\|\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}\mathbf{x} - \mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}\mathbf{y}\|^2. \end{aligned}$$

(ii) *Soit $k \in \mathbb{N}$ une itération de l'algorithme 7.2. Les variables aléatoires $\boldsymbol{s}_k$ et $\boldsymbol{t}_k$ peuvent être vues comme des termes d'erreurs apparaissant dans les calculs respectifs de $\mathbf{R}$ et $\mathbf{J}_{\gamma_k\mathbf{V}\mathbf{Q}}$. Notons que la possibilité de considérer des termes d'erreurs stochastiques aléatoires offre des degrés de liberté plus importants que l'hypothèse d'erreur déterministe sommable généralement faite dans la littérature (voir section 2.4.2). Remarquons cependant qu'il existe aussi des modèles prenant en compte des erreurs relatives* [Monteiro et Svaiter, 2010; Solodov et Svaiter, 2001; Svaiter, 2014].

(iii) *Soit $k \in \mathbb{N}^*$. D'après l'algorithme 7.2, $\boldsymbol{\mathcal{E}}_k$ et $\boldsymbol{\mathcal{Z}}_k$ sont indépendants si $\boldsymbol{\varepsilon}_k$ est indépendant de $(\boldsymbol{z}_0, (\boldsymbol{\varepsilon}_{k'}, \boldsymbol{s}_{k'}, \boldsymbol{t}_{k'})_{0\leqslant k'<k})$.*

7.3 Algorithmes primaux-duaux alternés aléatoirement par bloc

Soient p et q des entiers strictement positifs, et $(\mathcal{H}_j)_{1\leqslant j\leqslant p}$, $(\mathcal{G}_l)_{1\leqslant l\leqslant q}$ des espaces de Hilbert réels séparables. De plus, on note respectivement $\boldsymbol{\mathcal{H}} = \mathcal{H}_1 \oplus \cdots \oplus \mathcal{H}_p$ et $\boldsymbol{\mathcal{G}} = \mathcal{G}_1 \oplus \cdots \oplus \mathcal{G}_q$ les

sommes directes de Hilbert de $(\mathcal{H}_j)_{1\leqslant j\leqslant p}$ et $(\mathcal{G}_l)_{1\leqslant l\leqslant q}$. Enfin, notons $\boldsymbol{\mathcal{K}}$ l'espace produit $\boldsymbol{\mathcal{H}}\oplus\boldsymbol{\mathcal{G}}$.

7.3.1 Problème

Le problème suivant, mettant en œuvre des opérateurs monotones, a été beaucoup étudié ces dernières années (voir, par exemple dans [Boţ et Hendrich, 2013; Briceño-Arias, 2012; Combettes, 2013; Combettes et Pesquet, 2011; Pesquet et Pustelnik, 2012; Raguet *et al.*, 2013.]). Il jouera un rôle central tout au long de ce chapitre.

Problème 7.1. *Pour tout $j\in\{1,\dots,p\}$ et pour tout $l\in\{1,\dots,q\}$, soient $\mathsf{A}_j\colon\mathcal{H}_j\to 2^{\mathcal{H}_j}$ un opérateur maximal monotone, $\mathsf{C}_j\colon\mathcal{H}_j\to\mathcal{H}_j$ un opérateur cocoercif, $\mathsf{B}_l\colon\mathcal{G}_l\to 2^{\mathcal{G}_l}$ un opérateur maximal monotone, $\mathsf{D}_l\colon\mathcal{G}_l\to 2^{\mathcal{G}_l}$ un opérateur maximal et fortement monotone, et $\mathsf{L}_{l,j}\in\mathcal{B}(\mathcal{H}_j,\mathcal{G}_l)$. Supposons que*

$$(\forall l\in\{1,\dots,q\})\qquad \mathbb{L}_l=\big\{j\in\{1,\dots,p\}\;\big|\;\mathsf{L}_{l,j}\neq 0\big\}\neq\varnothing, \tag{7.8}$$

$$(\forall j\in\{1,\dots,p\})\qquad \mathbb{L}_j^*=\big\{l\in\{1,\dots,q\}\;\big|\;\mathsf{L}_{l,j}\neq 0\big\}\neq\varnothing, \tag{7.9}$$

et que l'ensemble $\mathbf{F}$ de solutions du problème

trouver $\mathsf{x}^{(1)}\in\mathcal{H}_1,\dots,\mathsf{x}^{(p)}\in\mathcal{H}_p$ tels que

$$(\forall j\in\{1,\dots,p\})\quad 0\in\mathsf{A}_j\mathsf{x}^{(j)}+\mathsf{C}_j\mathsf{x}^{(j)}+\sum_{l=1}^{q}\mathsf{L}_{l,j}^*(\mathsf{B}_l\,\square\,\mathsf{D}_l)\Big(\sum_{j'=1}^{p}\mathsf{L}_{l,j'}\mathsf{x}^{(j')}\Big) \tag{7.10}$$

est non vide. On note, de plus, $\mathbf{F}^*$ l'ensemble de solutions du problème dual :

trouver $\mathsf{v}^{(1)}\in\mathcal{G}_1,\dots,\mathsf{v}^{(q)}\in\mathcal{G}_q$ tels que

$$(\forall l\in\{1,\dots,q\})\;0\in-\sum_{j=1}^{p}\mathsf{L}_{l,j}(\mathsf{A}_j^{-1}\,\square\,\mathsf{C}_j^{-1})\Big(-\sum_{l'=1}^{q}\mathsf{L}_{l',j}^*\mathsf{v}^{(l')}\Big)+\mathsf{B}_l^{-1}\mathsf{v}^{(l)}+\mathsf{D}_l^{-1}\mathsf{v}^{(l)}. \tag{7.11}$$

L'objectif est de trouver un couple $(\widehat{\boldsymbol{x}},\widehat{\boldsymbol{v}})$ de variables aléatoires à valeurs dans $\mathbf{F}\times\mathbf{F}^*$.

Le problème 7.1 peut être réécrit comme la recherche du zéro d'une somme de deux opérateurs maximaux monotones dans l'espace produit $\boldsymbol{\mathcal{K}}$ comme indiqué ci-dessous.

Proposition 7.2. [Vũ, 2013] *Posons*

$$\mathbf{A}\colon\boldsymbol{\mathcal{H}}\to 2^{\boldsymbol{\mathcal{H}}}\colon\mathbf{x}\mapsto\bigtimes_{j=1}^{p}\mathsf{A}_j\mathsf{x}^{(j)}\qquad \mathbf{B}\colon\boldsymbol{\mathcal{G}}\to 2^{\boldsymbol{\mathcal{G}}}\colon\mathbf{v}\mapsto\bigtimes_{l=1}^{q}\mathsf{B}_l\mathsf{v}^{(l)}$$

$$\mathbf{C}\colon\boldsymbol{\mathcal{H}}\to\boldsymbol{\mathcal{H}}\colon\mathbf{x}\mapsto(\mathsf{C}_j\mathsf{x}^{(j)})_{1\leqslant j\leqslant p}\qquad \mathbf{D}\colon\boldsymbol{\mathcal{G}}\to 2^{\boldsymbol{\mathcal{G}}}\colon\mathbf{v}\mapsto\bigtimes_{l=1}^{q}\mathsf{D}_l\mathsf{v}^{(l)}$$

$$\mathbf{L}\colon\boldsymbol{\mathcal{H}}\to\boldsymbol{\mathcal{G}}\colon\mathbf{x}\mapsto\Big(\sum_{j=1}^{p}\mathsf{L}_{l,j}\mathsf{x}^{(j)}\Big)_{1\leqslant l\leqslant q}$$

et introduisons les opérateurs

$$\begin{aligned} \mathbf{Q}\colon \quad & \mathcal{K} \to 2^{\mathcal{K}} \\ & (\mathbf{x}, \mathbf{v}) \mapsto (\mathbf{A}\mathbf{x} + \mathbf{L}^*\mathbf{v}) \times (-\mathbf{L}\mathbf{x} + \mathbf{B}^{-1}\mathbf{v}) \end{aligned} \tag{7.12}$$

et

$$\begin{aligned} \mathbf{R}\colon \quad & \mathcal{K} \to \mathcal{K} \\ & (\mathbf{x}, \mathbf{v}) \mapsto (\mathbf{C}\mathbf{x}, \mathbf{D}^{-1}\mathbf{v}). \end{aligned} \tag{7.13}$$

Les assertions suivantes sont alors vérifiées

(i) $\mathbf{Q}$ *est un opérateur maximal monotone et* $\mathbf{R}$ *est un opérateur cocoercif.*

(ii) $\mathbf{Z} = \operatorname{zer}(\mathbf{Q} + \mathbf{R})$ *est non vide.*

(iii) *Un couple* $(\widehat{\boldsymbol{x}}, \widehat{\boldsymbol{v}})$ *de variables aléatoires est une solution du problème 7.1 si et seulement si* $(\widehat{\boldsymbol{x}}, \widehat{\boldsymbol{v}})$ *est à valeurs dans* $\mathbf{Z}$.

D'après la propriété ci-dessus, nous pouvons donc utiliser l'algorithme 7.2 pour résoudre le problème 7.1. De plus, plusieurs algorithmes peuvent en être déduits selon le choix fait pour les opérateurs de préconditionnement.

Dans la suite, $\mathbf{L} \in \mathcal{B}(\boldsymbol{\mathcal{H}}, \boldsymbol{\mathcal{G}})$ sera défini comme dans la proposition 7.2.

7.3.2 Première sous-classe d'algorithmes

Nous donnons dans un premier temps deux résultats préliminaires qui nous seront utiles pour développer les algorithmes proposés dans cette section.

Lemme 7.1. *Soient* $\mathbf{W} \in \mathcal{S}^+(\boldsymbol{\mathcal{H}})$ *et* $\mathbf{U} \in \mathcal{S}^+(\boldsymbol{\mathcal{G}})$ *tels que* $\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\| < 1$.

(i) *L'opérateur défini par*

$$\begin{aligned} \mathbf{V}'\colon \quad & \mathcal{K} \to \mathcal{K} \\ & (\mathbf{x}, \mathbf{v}) \mapsto (\mathbf{W}^{-1}\mathbf{x} - \mathbf{L}^*\mathbf{v}, -\mathbf{L}\mathbf{x} + \mathbf{U}^{-1}\mathbf{v}) \end{aligned} \tag{7.14}$$

appartient à $\mathcal{S}^+(\mathcal{K})$. *Son inverse donné par*

$$\begin{aligned} \mathbf{V}\colon \quad & \mathcal{K} \to \mathcal{K} \\ & (\mathbf{x}, \mathbf{v}) \mapsto \big((\mathbf{W}^{-1} - \mathbf{L}^*\mathbf{U}\mathbf{L})^{-1}\mathbf{x} + \mathbf{W}\mathbf{L}^*(\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}\mathbf{v}, \\ & \qquad (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}(\mathbf{L}\mathbf{W}\mathbf{x} + \mathbf{v})\big) \end{aligned} \tag{7.15}$$

est aussi un opérateur de $\mathcal{S}^+(\mathcal{K})$.

(ii) *Soient* $\mathbf{C}\colon \boldsymbol{\mathcal{H}} \to \boldsymbol{\mathcal{H}}$, $\mathbf{D}\colon \boldsymbol{\mathcal{G}} \to 2^{\boldsymbol{\mathcal{G}}}$, *et* $\mathbf{R}\colon \boldsymbol{\mathcal{K}} \to \boldsymbol{\mathcal{K}}$ *les opérateurs définis dans la proposition 7.2. Si* $\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}$ *est* μ*-cocoercif, avec* $\mu \in \left]0, +\infty\right[$, *et* $\mathbf{U}^{1/2}\mathbf{D}^{-1}\mathbf{U}^{1/2}$ *est* ν*-cocoercif, avec* $\nu \in \left]0, +\infty\right[$, *alors, pour tout* $\alpha \in \left]0, +\infty\right[$, $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ *est* ϑ_α*-cocoercif, avec*

$$\vartheta_\alpha = (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2) \min\left\{\mu(1 + \alpha\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|)^{-1}, \nu(1 + \alpha^{-1}\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|)^{-1}\right\}. \tag{7.16}$$

La démonstration du lemme 7.1 est donnée dans l'annexe 7.A.

Remarque 7.2.

(i) *Dans* (7.16)*, un choix simple est de prendre* $\alpha = 1$*, ce qui conduit à la constante de cocoercivité*

$$\vartheta_1 = (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|) \min\{\mu, \nu\}.$$

Un autre choix possible de cette constante est $\vartheta_{\widehat{\alpha}}$ *où* $\widehat{\alpha}$ *est le maximiseur de* $\alpha \mapsto \vartheta_\alpha$ *sur* $\left]0, +\infty\right[$*. On obtient alors*

$$\widehat{\alpha} = \frac{\mu - \nu + \sqrt{(\mu - \nu)^2 + 4\mu\nu\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2}}{2\nu\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|}.$$

(ii) *Lorsque* $\mathbf{D}^{-1} = \mathbf{0}$*, la constante positive* ν *peut être choisie arbitrairement grande. Une constante de cocoercivité de* $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ *est alors donnée par*

$$\lim_{\substack{\alpha \to 0 \\ \alpha > 0}} \vartheta_\alpha = (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)\mu.$$

Lemme 7.2. *Soient* $\mathbf{A}\colon \boldsymbol{\mathcal{H}} \to 2^{\boldsymbol{\mathcal{H}}}$, $\mathbf{B}\colon \boldsymbol{\mathcal{G}} \to 2^{\boldsymbol{\mathcal{G}}}$, $\mathbf{C}\colon \boldsymbol{\mathcal{H}} \to \boldsymbol{\mathcal{H}}$, $\mathbf{D}\colon \boldsymbol{\mathcal{G}} \to 2^{\boldsymbol{\mathcal{G}}}$, $\mathbf{Q}\colon \boldsymbol{\mathcal{K}} \to 2^{\boldsymbol{\mathcal{K}}}$, *et* $\mathbf{R}\colon \boldsymbol{\mathcal{K}} \to \boldsymbol{\mathcal{K}}$*. Soient* $\mathbf{W} \in \mathcal{S}^+(\boldsymbol{\mathcal{H}})$ *et* $\mathbf{U} \in \mathcal{S}^+(\boldsymbol{\mathcal{G}})$ *tels que* $\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\| < 1$*. Soit* $\mathbf{V} \in \mathcal{S}^+(\boldsymbol{\mathcal{K}})$ *défini par* (7.15)*. Pour tout* $\mathbf{z} = (\mathbf{x}, \mathbf{v}) \in \boldsymbol{\mathcal{K}}$ *et* $(\mathbf{c}, \mathbf{e}) \in \boldsymbol{\mathcal{K}}$*, on pose*

$$\begin{cases} \mathbf{y} = \mathbf{J}_{\mathbf{W}\mathbf{A}}(\mathbf{x} - \mathbf{W}(\mathbf{L}^*\mathbf{v} + \mathbf{C}\mathbf{x} + \mathbf{c})) \\ \mathbf{u} = \mathbf{J}_{\mathbf{U}\mathbf{B}^{-1}}(\mathbf{v} + \mathbf{U}(\mathbf{L}(2\mathbf{y} - \mathbf{x}) - \mathbf{D}^{-1}\mathbf{v} + \mathbf{e})). \end{cases} \tag{7.17}$$

On a alors, $(\mathbf{y}, \mathbf{u}) = \mathbf{J}_{\mathbf{V}\mathbf{Q}}(\mathbf{z} - \mathbf{V}\mathbf{R}\mathbf{z} + \mathbf{s})$ *où*

$$\mathbf{s} = \left((\mathbf{W}^{-1} - \mathbf{L}^*\mathbf{U}\mathbf{L})^{-1}(\mathbf{L}^*\mathbf{U}\mathbf{e} - \mathbf{c}), (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}(\mathbf{e} - \mathbf{L}\mathbf{W}\mathbf{c})\right).$$

Démonstration. Soient $\mathbf{z} = (\mathbf{x}, \mathbf{v}) \in \mathcal{K}$ et $\mathbf{s} = (\mathbf{c}', \mathbf{e}') \in \mathcal{K}$. On a les équivalences suivantes :

$$
\begin{aligned}
& (\mathbf{y}, \mathbf{u}) = \mathbf{J}_{\mathbf{VQ}}(\mathbf{z} - \mathbf{VRz} + \mathbf{s}) \\
\Leftrightarrow \quad & \mathbf{z} - \mathbf{VRz} + \mathbf{s} \in (\mathbf{Id} + \mathbf{VQ})(\mathbf{y}, \mathbf{u}) \\
\Leftrightarrow \quad & \mathbf{V}^{-1}(\mathbf{z} + \mathbf{s} - (\mathbf{y}, \mathbf{u})) - \mathbf{Rz} \in \mathbf{Q}(\mathbf{y}, \mathbf{u}) \\
\Leftrightarrow \quad & \begin{cases} \mathbf{W}^{-1}(\mathbf{x} - \mathbf{y} + \mathbf{c}') - \mathbf{L}^*(\mathbf{v} + \mathbf{e}') - \mathbf{Cx} \in \mathbf{Ay} \\ \mathbf{U}^{-1}(\mathbf{v} - \mathbf{u} + \mathbf{e}') + \mathbf{L}(2\mathbf{y} - \mathbf{x} - \mathbf{c}') - \mathbf{D}^{-1}\mathbf{v} \in \mathbf{B}^{-1}\mathbf{u} \end{cases} \\
\Leftrightarrow \quad & \begin{cases} \mathbf{x} + \mathbf{c}' - \mathbf{W}(\mathbf{L}^*(\mathbf{v} + \mathbf{e}') + \mathbf{Cx}) \in (\mathbf{Id} + \mathbf{WA})\mathbf{y} \\ \mathbf{v} + \mathbf{e}' + \mathbf{U}(\mathbf{L}(2\mathbf{y} - \mathbf{x} - \mathbf{c}') - \mathbf{D}^{-1}\mathbf{v}) \in (\mathbf{Id} + \mathbf{UB}^{-1})\mathbf{u} \end{cases} \\
\Leftrightarrow \quad & \begin{cases} \mathbf{y} = \mathbf{J}_{\mathbf{WA}}(\mathbf{x} + \mathbf{c}' - \mathbf{W}(\mathbf{L}^*(\mathbf{v} + \mathbf{e}') + \mathbf{Cx})) \\ \mathbf{u} = \mathbf{J}_{\mathbf{UB}^{-1}}(\mathbf{v} + \mathbf{e}' + \mathbf{U}(\mathbf{L}(2\mathbf{y} - \mathbf{x} - \mathbf{c}') - \mathbf{D}^{-1}\mathbf{v})), \end{cases}
\end{aligned}
\tag{7.18}
$$

où, pour obtenir (7.18), on a utilisé l'expression de $\mathbf{Q}$ donnée par (7.12) et l'expression de $\mathbf{V}$ donnée par (7.14).

Afin de conclure, remarquons que, puisque $\|\mathbf{U}^{1/2}\mathbf{LW}^{1/2}\| < 1$, nous avons montré dans la démonstration du lemme 7.1(i) que $\mathbf{W}^{-1} - \mathbf{L}^*\mathbf{UL}$ et $\mathbf{U}^{-1} - \mathbf{LWL}^*$ sont des isomorphismes (en conséquence de (7.73) et (7.74)). Ainsi, pour tout $(\mathbf{c}, \mathbf{e}) \in \mathcal{K}$,

$$
\begin{cases} \mathbf{Wc} = \mathbf{WL}^*\mathbf{e}' - \mathbf{c}' \\ \mathbf{Ue} = \mathbf{e}' - \mathbf{ULc}' \end{cases}
\quad \Leftrightarrow \quad
\begin{cases} \mathbf{c}' = (\mathbf{W}^{-1} - \mathbf{L}^*\mathbf{UL})^{-1}(\mathbf{L}^*\mathbf{Ue} - \mathbf{c}) \\ \mathbf{e}' = (\mathbf{U}^{-1} - \mathbf{LWL}^*)^{-1}(\mathbf{e} - \mathbf{LWc}). \end{cases}
$$

□

Nous pouvons maintenant donner un premier algorithme primal-dual alterné aléatoirement par bloc pour résoudre le problème 7.1. Ses itérations sont décrites dans l'algorithme 7.3.

Algorithme 7.3 Algorithme primal-dual alterné aléatoirement par bloc (version 1)

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{H}}$, et soient $\boldsymbol{v}_0$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$. Pour tout $j \in \{1,\dots,p\}$, soit $\mathsf{W}_j \in \mathcal{S}^+(\mathcal{H}_j)$, et, pour tout $l \in \{1,\dots,q\}$, soit $\mathsf{U}_l \in \mathcal{S}^+(\mathcal{G}_l)$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{p+q}$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

pour $j = 1,\dots,p$

$$y_k^{(j)} = \varepsilon_k^{(j)}\Big(\mathsf{J}_{\mathsf{W}_j\mathsf{A}_j}\Big(x_k^{(j)} - \mathsf{W}_j\big(\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^* v_k^{(l)} + \mathsf{C}_j x_k^{(j)} + c_k^{(j)}\big)\Big) + a_k^{(j)}\Big),$$

$$x_{k+1}^{(j)} = x_k^{(j)} + \lambda_k\varepsilon_k^{(j)}(y_k^{(j)} - x_k^{(j)}),$$

pour $l = 1,\dots,q$

$$u_k^{(l)} = \varepsilon_k^{(p+l)}\Big(\mathsf{J}_{\mathsf{U}_l\mathsf{B}_l^{-1}}\Big(v_k^{(l)} + \mathsf{U}_l\big(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}(2y_k^{(j)} - x_k^{(j)}) - \mathsf{D}_l^{-1}v_k^{(l)} + d_k^{(l)}\big)\Big) + b_k^{(l)}\Big),$$

$$v_{k+1}^{(l)} = v_k^{(l)} + \lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)} - v_k^{(l)}).$$

En utilisant les lemmes 7.1 et 7.2, nous obtenons la proposition suivante établissant la convergence de l'algorithme 7.3.

Proposition 7.3. *Considérons l'algorithme 7.3. Supposons que, pour tout $j \in \{1,\dots,p\}$, $\mathsf{W}_j^{1/2}\mathsf{C}_j\mathsf{W}_j^{1/2}$ est μ_j-cocoercif, avec $\mu_j \in]0,+\infty[$, et, pour tout $l \in \{1,\dots,q\}$, $\mathsf{U}_l^{1/2}\mathsf{D}_l^{-1}\mathsf{U}_l^{1/2}$ est ν_l-cocoercif, avec $\nu_l \in]0,+\infty[$. Supposons que*

$$(\exists\alpha \in]0,+\infty[) \quad 2\vartheta_\alpha > 1 \tag{7.19}$$

où ϑ_α est défini par (7.16) avec $\mu = \min\{\mu_1,\dots,\mu_p\}$, $\nu = \min\{\nu_1,\dots,\nu_q\}$ et

$$\mathbf{W}\colon \boldsymbol{\mathcal{H}} \to \boldsymbol{\mathcal{H}}\colon \mathbf{x} \mapsto (\mathsf{W}_1\mathsf{x}^{(1)},\dots,\mathsf{W}_p\mathsf{x}^{(p)}) \quad \text{et} \quad \mathbf{U}\colon \boldsymbol{\mathcal{G}} \to \boldsymbol{\mathcal{G}}\colon \mathbf{v} \mapsto (\mathsf{U}_1\mathsf{v}^{(1)},\dots,\mathsf{U}_q\mathsf{v}^{(q)}). \tag{7.20}$$

Supposons que les éléments de la suite $(\lambda_k)_{k\in\mathbb{N}}$ sont dans $]0,1]$ et que $\inf_{k\in\mathbb{N}}\lambda_k > 0$. Posons $(\forall k \in \mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ et $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant k'\leqslant k}$, et supposons de plus que les assertions suivantes sont vérifiées :

(i) $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{a}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$, $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$, $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{c}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$, *et* $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{d}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$ P*-presque sûrement,*

(ii) *pour tout $k \in \mathbb{N}$, $\boldsymbol{\mathcal{E}}_k$ et $\boldsymbol{\mathcal{X}}_k$ sont indépendants, et $(\forall l \in \{1,\dots,q\})$ $\mathsf{P}[\varepsilon_0^{(p+l)} = 1] > 0$,*

(iii) *pour tout $j \in \{1,\dots,p\}$ et $k \in \mathbb{N}$, $\bigcup_{l\in\mathbb{L}_j^*}\{\omega\in\Omega \mid \varepsilon_k^{(p+l)}(\omega) = 1\} \subset \{\omega\in\Omega \mid \varepsilon_k^{(j)}(\omega) = 1\}$.*

La suite $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ converge alors faiblement P-presque sûrement vers une variable aléatoire à valeurs dans $\mathbf{F}$, et $(\boldsymbol{v}_k)_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire à valeurs dans $\mathbf{F}^*$.

La démonstration de la proposition 7.3 est donnée dans l'annexe 7.B. Un certain nombre d'observations peuvent être faites sur cette proposition.

Remarque 7.3.

(i) *Les variables aléatoires booléennes* $(\varepsilon_k^{(i)})_{1\leqslant i\leqslant p+q}$ *indiquent les variables* $(x_k^{(j)})_{1\leqslant j\leqslant p}$ *et* $(v_k^{(l)})_{1\leqslant l\leqslant q}$ *qui sont activées à chaque itération* k *de l'algorithme 7.3. D'un point de vue algorithmique, lorsque certaines de ces variables sont nulles, les variables associées n'ont pas besoin d'être mises à jour. Remarquons que, en accord avec la condition (iii), pour tout* $j \in \{1,\dots,p\}$, $y_k^{(j)}$ *n'est pas seulement mis à jour lorsque* $x_k^{(j)}$ *est activée, mais aussi lorsqu'il existe* $l \in \{1,\dots,q\}$ *tel que* $v_k^{(l)}$ *est activé et* $\mathsf{L}_{l,j} \neq 0$.

(ii) *Pour tout* $n \in \mathbb{N}$, $j \in \{1,\dots,p\}$, *et* $l \in \{1,\dots,q\}$, $a_k^{(j)}$, $b_k^{(l)}$, $c_k^{(j)}$, *et* $d_k^{(l)}$ *sont des erreurs stochastiques pouvant apparaitre, à l'itération* k *de l'algorithme 7.3, lors des calculs respectifs de* $\mathsf{J}_{\mathsf{W}_j\mathsf{A}_j}$, $\mathsf{J}_{\mathsf{U}_l\mathsf{B}_l^{-1}}$, C_j, *et* D_l^{-1}.

(iii) *En utilisant l'inégalité triangulaire et l'inégalité de Cauchy-Schwarz, on obtient*

$$
\begin{aligned}
(\mathsf{x}\in\mathcal{H}) \qquad \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\mathsf{x}\|^2 &= \sum_{l=1}^{q}\Big\|\sum_{j=1}^{p}\mathsf{U}_l^{1/2}\mathsf{L}_{l,j}\mathsf{W}_j^{1/2}\mathsf{x}^{(j)}\Big\|^2\\
&\leqslant \sum_{l=1}^{q}\Big(\sum_{j=1}^{p}\|\mathsf{U}_l^{1/2}\mathsf{L}_{l,j}\mathsf{W}_j^{1/2}\|\|\mathsf{x}^{(j)}\|\Big)^2\\
&\leqslant \sum_{l=1}^{q}\Big(\sum_{j=1}^{p}\|\mathsf{U}_l^{1/2}\mathsf{L}_{l,j}\mathsf{W}_j^{1/2}\|^2\Big)\Big(\sum_{j=1}^{p}\|\mathsf{x}^{(j)}\|^2\Big),
\end{aligned}
$$

ce qui implique que

$$
\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\| \leqslant \Big(\sum_{j=1}^{p}\sum_{l=1}^{q}\|\mathsf{U}_l^{1/2}\mathsf{L}_{l,j}\mathsf{W}_j^{1/2}\|^2\Big)^{1/2}. \tag{7.21}
$$

Pour tout $j \in \{1,\ldots,p\}$, soit $\widetilde{\mu}_j \in \left]0,+\infty\right[$ une constante de cocoercivité de C_j et, pour tout $l \in \{1,\ldots,q\}$, soit $\widetilde{\nu}_k \in \left]0,+\infty\right[$ une constante de forte monotonicité de D_l. On peut alors choisir

$$\mu = \min\{(\|\mathsf{W}_j\|^{-1}\widetilde{\mu}_j)_{1\leqslant j\leqslant p}\}, \qquad \nu = \min\{(\|\mathsf{U}_l\|^{-1}\widetilde{\nu}_l)_{1\leqslant l\leqslant q}\}. \tag{7.22}$$

Par conséquent, en utilisant la remarque 7.2(i), une condition suffisante pour que (7.19) *soit satisfaite avec $\alpha = 1$ est*

$$\left(1 - \Big(\sum_{j=1}^{p}\sum_{l=1}^{q}\|\mathsf{U}_l^{1/2}\mathsf{L}_{l,j}\mathsf{W}_j^{1/2}\|^2\Big)^{1/2}\right)\min\{(\|\mathsf{W}_j\|^{-1}\widetilde{\mu}_j)_{1\leqslant j\leqslant p}, (\|\mathsf{U}_l\|^{-1}\widetilde{\nu}_l)_{1\leqslant l\leqslant q}\} > \frac{1}{2}. \tag{7.23}$$

Lorsque $\mathbf{D}^{-1} = \mathbf{0}$, en accord avec la remarque 7.2(ii), cette condition peut être remplacée par

$$\left(1 - \sum_{j=1}^{p}\sum_{l=1}^{q}\|\mathsf{U}_l^{1/2}\mathsf{L}_{l,j}\mathsf{W}_j^{1/2}\|^2\right)\min\{(\|\mathsf{W}_j\|^{-1}\widetilde{\mu}_j)_{1\leqslant j\leqslant p}\} > \frac{1}{2}. \tag{7.24}$$

(iv) *L'algorithme 7.3 généralise plusieurs résultats existants dans le cas déterministe, lorsque $p = 1$ et que la mise à jour des variables duales n'est pas aléatoire. Dans la plupart de ces travaux, $\mathsf{W}_1 = \tau\mathrm{Id}$ et, pour tout $l \in \{1,\ldots,q\}$, $\mathsf{U}_l = \rho_l\mathrm{Id}$ avec $(\tau,\rho_1,\ldots,\rho_q) \in \left]0,+\infty\right[^{q+1}$. En particulier, dans* [Vũ, 2013]*, une condition suffisante pour que* (7.23) *soit satisfaite est utilisée, tandis que dans* [Condat, 2013] *une condition similaire à* (7.24) *est utilisée dans un cadre variationnel qui correspondrait au cas où $\mathbf{D}^{-1} = \mathbf{0}$. L'algorithme proposé généralise aussi les résultats donnés dans* [Combettes et Vũ, 2014, Sec. 6] *lorsqu'une métrique constante est considérée.*

Les problèmes 7.10 et 7.11 étant symétriques, les rôles des variables primales et des variables duales des deux problèmes peuvent être échangées. Cela nous conduit à une forme symétrique de l'algorithme 7.3, donnée dans l'algorithme 7.4.

Algorithme 7.4 Algorithme primal-dual alterné aléatoirement par bloc (version 1 symétrique)

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{H}}$, et soient $\boldsymbol{v}_0$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$. Pour tout $j \in \{1,\dots,p\}$, soit $\mathsf{W}_j \in \mathcal{S}^+(\mathcal{H}_j)$, et, pour tout $l \in \{1,\dots,q\}$, soit $\mathsf{U}_l \in \mathcal{S}^+(\mathcal{G}_l)$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{p+q}$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

pour $l = 1,\dots,q$

$$u_k^{(l)} = \varepsilon_k^{(p+l)}\Big(\mathsf{J}_{\mathsf{U}_l\mathsf{B}_l^{-1}}\big(v_k^{(l)} + \mathsf{U}_l(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}x_k^{(j)} - \mathsf{D}_l^{-1}v_k^{(l)} + d_k^{(l)})\big) + b_k^{(l)}\Big),$$

$$v_{k+1}^{(l)} = v_k^{(l)} + \lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)} - v_k^{(l)}),$$

pour $j = 1,\dots,p$

$$y_k^{(j)} = \varepsilon_k^{(j)}\Big(\mathsf{J}_{\mathsf{W}_j\mathsf{A}_j}\big(x_k^{(j)} - \mathsf{W}_j(\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^*(2u_k^{(l)} - v_k^{(l)}) + \mathsf{C}_j x_k^{(j)} + c_k^{(j)})\big) + a_k^{(j)}\Big),$$

$$x_{k+1}^{(j)} = x_k^{(j)} + \lambda_k\varepsilon_k^{(j)}(y_k^{(j)} - x_k^{(j)}).$$

Proposition 7.4. *Considérons l'algorithme 7.4,* $\mathbf{W}$ *et* $\mathbf{U}$ *définis par* (7.20) *et* (μ,ν) *donnés dans la proposition 7.3. Supposons que* (7.19) *est vérifiée où* ϑ_α *est défini par* (7.16)*. Supposons que les éléments de la suite* $(\lambda_k)_{k\in\mathbb{N}}$ *sont dans* $]0,1]$ *et que* $\inf_{k\in\mathbb{N}}\lambda_k > 0$*. De plus, supposons que la condition (i) de la proposition 7.3 est vérifiée où* $(\forall k\in\mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ *et* $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant k'\leqslant k}$*, et que les assertions suivantes sont satisfaites :*

(ii) *pour tout* $k\in\mathbb{N}$, $\boldsymbol{\mathcal{E}}_k$ *et* $\boldsymbol{\mathcal{X}}_k$ *sont indépendants, et* $(\forall j\in\{1,\dots,p\})$ $\mathsf{P}[\varepsilon_0^{(j)} = 1] > 0$.

(iii) *pour tout* $l\in\{1,\dots,q\}$ *et* $k\in\mathbb{N}$, $\bigcup_{j\in\mathbb{L}_l}\{\omega\in\Omega \mid \varepsilon_k^{(j)}(\omega) = 1\} \subset \{\omega\in\Omega \mid \varepsilon_k^{(p+l)}(\omega) = 1\}$.

La suite $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ *converge alors faiblement* P*-presque sûrement vers une variable aléatoire à valeurs dans* $\mathbf{F}$*, et* $(\boldsymbol{v}_k)_{k\in\mathbb{N}}$ *converge faiblement* P*-presque sûrement vers une variable aléatoire à valeurs dans* $\mathbf{F}^*$.

7.3.3 Seconde sous-classe d'algorithmes

Dans cette section, nous considérons que l'opérateur $\mathbf{V}$ a une forme diagonale, et nous procédons par une approche similaire à celle suivie dans la section 7.3.2.

Lemme 7.3. *Soient* $\mathbf{W} \in \mathcal{S}^+(\mathcal{H})$ *et* $\mathbf{U} \in \mathcal{S}^+(\mathcal{G})$ *tels que* $\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\| < 1$.

(i) *L'opérateur défini par*

$$\begin{aligned} \mathbf{V}\colon \quad & \mathcal{K} \to \mathcal{K} \\ & (\mathbf{x}, \mathbf{v}) \mapsto (\mathbf{W}\mathbf{x}, (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}\mathbf{v}) \end{aligned} \tag{7.25}$$

est dans $\mathcal{S}^+(\mathcal{K})$.

(ii) *Soient* $\mathbf{C}\colon \mathcal{H} \to \mathcal{H}$, $\mathbf{D}\colon \mathcal{G} \to 2^{\mathcal{G}}$, *et* $\mathbf{R}\colon \mathcal{K} \to \mathcal{K}$ *les opérateurs définis dans la proposition 7.2. Si* $\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}$ *est* μ*-cocoercif, avec* $\mu \in]0, +\infty[$, *et* $\mathbf{U}^{1/2}\mathbf{D}^{-1}\mathbf{U}^{1/2}$ *est* ν*-cocoercif, avec* $\nu \in]0, +\infty[$, *alors* $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ *est* ϑ*-cocoercif, où*

$$\vartheta = \min\{\mu, \nu(1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)\}. \tag{7.26}$$

Démonstration.
(i) Tout d'abord, remarquons que, comme nous l'avons montré dans (7.74), si $\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\| < 1$, alors $\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*$ est un opérateur fortement positif de $\mathcal{B}(\mathcal{G})$ et est donc un isomorphisme. Ainsi, pour tout $(\mathbf{x}, \mathbf{v}) \in \mathcal{K}$,

$$\begin{aligned} \langle (\mathbf{x}, \mathbf{v}) \mid \mathbf{V}(\mathbf{x}, \mathbf{v}) \rangle &= \langle \mathbf{x} \mid \mathbf{W}\mathbf{x} \rangle + \Big\langle \mathbf{v} \mid (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}\mathbf{v} \Big\rangle \\ &\geqslant \|\mathbf{W}^{-1}\|^{-1}\|\mathbf{x}\|^2 + \|\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*\|^{-1}\|\mathbf{v}\|^2 \\ &\geqslant \min\{\|\mathbf{W}^{-1}\|^{-1}, \|\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*\|^{-1}\}(\|\mathbf{x}\|^2 + \|\mathbf{v}\|^2). \end{aligned}$$

Donc, $\mathbf{V} \in \mathcal{S}^+(\mathcal{K})$.
(ii) Soient $\mathbf{z} = (\mathbf{x}, \mathbf{v}) \in \mathcal{K}$ et $\mathbf{z}' = (\mathbf{x}', \mathbf{v}') \in \mathcal{K}$. On a

$$\begin{aligned} &\|\mathbf{R}\mathbf{z} - \mathbf{R}\mathbf{z}'\|_{\mathbf{V}}^2 \\ &\quad = \langle \mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}' \mid \mathbf{W}(\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}') \rangle + \Big\langle \mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}' \mid (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}(\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}') \Big\rangle \\ &\quad \leqslant \|\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}'\|_{\mathbf{W}}^2 + \|(\mathbf{Id} - \mathbf{U}^{1/2}\mathbf{L}\mathbf{W}\mathbf{L}^*\mathbf{U}^{1/2})^{-1}\|\|\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}'\|_{\mathbf{U}}^2 \\ &\quad \leqslant \mu^{-1}\langle \mathbf{x} - \mathbf{x}' \mid \mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}' \rangle + \nu^{-1}(1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)^{-1}\Big\langle \mathbf{v} - \mathbf{v}' \mid \mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}' \Big\rangle \\ &\quad \leqslant \max\{\mu^{-1}, \nu^{-1}(1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)^{-1}\}\langle \mathbf{z} - \mathbf{z}' \mid \mathbf{R}\mathbf{z} - \mathbf{R}\mathbf{z}' \rangle, \end{aligned}$$

ce qui, en utilisant la remarque faite au début de la démonstration du lemme 7.1(ii), montre que $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ est ϑ-cocoercif. □

Lemme 7.4. *Soient* $\mathbf{B}\colon \mathcal{G} \to 2^{\mathcal{G}}$, $\mathbf{C}\colon \mathcal{H} \to \mathcal{H}$, $\mathbf{D}\colon \mathcal{G} \to 2^{\mathcal{G}}$, $\mathbf{Q}\colon \mathcal{K} \to 2^{\mathcal{K}}$, *et* $\mathbf{R}\colon \mathcal{K} \to \mathcal{K}$. *On suppose que l'opérateur* $\mathbf{A}$ *défini dans la proposition 7.2 est nul. Soient* $\mathbf{W} \in \mathcal{S}^+(\mathcal{H})$ *et* $\mathbf{U} \in \mathcal{S}^+(\mathcal{G})$ *tels que* $\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\| < 1$. *Soit* $\mathbf{V} \in \mathcal{S}^+(\mathcal{K})$ *défini par* (7.25). *Pour tout* $\mathbf{z} = (\mathbf{x}, \mathbf{v}) \in \mathcal{K}$ *et* $(\mathbf{e}_1, \mathbf{e}_2) \in \mathcal{K}$, *posons*

$$\begin{cases} \mathbf{u} = \mathbf{J}_{\mathbf{U}\mathbf{B}^{-1}}\Big(\mathbf{v} + \mathbf{U}(\mathbf{L}(\mathbf{x} - \mathbf{W}(\mathbf{C}\mathbf{x} + \mathbf{L}^*\mathbf{v})) - \mathbf{D}^{-1}\mathbf{v} + \mathbf{e}_2)\Big) \\ \mathbf{y} = \mathbf{x} - \mathbf{W}(\mathbf{C}\mathbf{x} + \mathbf{L}^*\mathbf{u} + \mathbf{e}_1). \end{cases} \tag{7.27}$$

On a alors $(\mathbf{y}, \mathbf{u}) = \mathbf{J}_{\mathbf{V}\mathbf{Q}}(\mathbf{z} - \mathbf{V}\mathbf{R}\mathbf{z} + \mathbf{s})$ *où*

$$\mathbf{s} = (\, -\mathbf{W}\mathbf{e}_1, (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}(\mathbf{e}_2 + \mathbf{L}\mathbf{W}\mathbf{e}_1)). \tag{7.28}$$

Démonstration. Soient $\mathbf{z} = (\mathbf{x}, \mathbf{v}) \in \mathcal{K}$ et $\mathbf{s} = (\mathbf{e}_1', \mathbf{e}_2') \in \mathcal{K}$. On a les équivalences suivantes :

$$\begin{aligned} & (\mathbf{y}, \mathbf{u}) = \mathbf{J}_{\mathbf{V}\mathbf{Q}}(\mathbf{z} - \mathbf{V}\mathbf{R}\mathbf{z} + \mathbf{s}) \\ \Leftrightarrow \quad & \mathbf{V}^{-1}(\mathbf{z} + \mathbf{s} - (\mathbf{y}, \mathbf{u})) - \mathbf{R}\mathbf{z} \in \mathbf{Q}(\mathbf{y}, \mathbf{u}) \\ \Leftrightarrow \quad & \begin{cases} \mathbf{W}^{-1}(\mathbf{x} - \mathbf{y} + \mathbf{e}_1') - \mathbf{L}^*\mathbf{u} - \mathbf{C}\mathbf{x} = \mathbf{0} \\ (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)(\mathbf{v} - \mathbf{u} + \mathbf{e}_2') + \mathbf{L}\mathbf{y} - \mathbf{D}^{-1}\mathbf{v} \in \mathbf{B}^{-1}\mathbf{u} \end{cases} \\ \Leftrightarrow \quad & \begin{cases} \mathbf{y} = \mathbf{x} - \mathbf{W}(\mathbf{C}\mathbf{x} + \mathbf{L}^*\mathbf{u}) + \mathbf{e}_1' \\ \mathbf{v} + \mathbf{e}_2' + \mathbf{U}\left(\mathbf{L}(\mathbf{x} - \mathbf{W}(\mathbf{C}\mathbf{x} + \mathbf{L}^*\mathbf{v} + \mathbf{L}^*\mathbf{e}_2') + \mathbf{e}_1') - \mathbf{D}^{-1}\mathbf{v}\right) \in (\mathbf{Id} + \mathbf{U}\mathbf{B}^{-1})\mathbf{u} \end{cases} \\ \Leftrightarrow \quad & \begin{cases} \mathbf{u} = \mathbf{J}_{\mathbf{U}\mathbf{B}^{-1}}\left(\mathbf{v} + \mathbf{e}_2' + \mathbf{U}\left(\mathbf{L}(\mathbf{x} - \mathbf{W}(\mathbf{C}\mathbf{x} + \mathbf{L}^*\mathbf{v} + \mathbf{L}^*\mathbf{e}_2') + \mathbf{e}_1') - \mathbf{D}^{-1}\mathbf{v}\right)\right) \\ \mathbf{y} = \mathbf{x} - \mathbf{W}(\mathbf{C}\mathbf{x} + \mathbf{L}^*\mathbf{u}) + \mathbf{e}_1', \end{cases} \end{aligned}$$

qui implique (7.27) pour

$$\begin{cases} -\mathbf{W}\mathbf{e}_1 = \mathbf{e}_1' \\ \mathbf{U}\mathbf{e}_2 = \mathbf{U}\mathbf{L}(\mathbf{e}_1' - \mathbf{W}\mathbf{L}^*\mathbf{e}_2') + \mathbf{e}_2'. \end{cases}$$

Puisque $\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*$ est un isomorphisme, ces égalités sont équivalentes à (7.28). □

D'après les lemmes 7.3 et 7.4, on peut déduire un second type d'algorithme pour résoudre le problème 7.1 lorsque $\mathbf{A} = \mathbf{0}$.

Soient $\mathbf{W}$ et $\mathbf{U}$ définis par (7.20). Nous proposons de résoudre le problème 7.1 lorsque $\mathbf{A} = \mathbf{0}$ en utilisant l'algorithme 7.5.

Algorithme 7.5 Algorithme primal-dual alterné aléatoirement par bloc (version 2)

Initialisation : Soient $\boldsymbol{x}_0$, et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{H}}$, et soient $\boldsymbol{v}_0$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$. Pour tout $j \in \{1,\dots,p\}$, soit $\mathsf{W}_j \in \mathcal{S}^+(\mathcal{H}_j)$, et, pour tout $l \in \{1,\dots,q\}$, soit $\mathsf{U}_l \in \mathcal{S}^+(\mathcal{G}_l)$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{p+q}$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

pour $j = 1,\dots,p$

$\eta_k^{(j)} = \max\left\{\varepsilon_k^{(p+l)} \;\middle|\; l \in \mathbb{L}_j^*\right\}$

$w_k^{(j)} = \eta_k^{(j)}\Big(x_k^{(j)} - \mathsf{W}_j(\mathsf{C}_j x_k^{(j)} + c_k^{(j)})\Big)$

pour $l = 1,\dots,q$

$u_k^{(l)} = \varepsilon_k^{(p+l)}\Big(\mathsf{J}_{\mathsf{U}_l\mathsf{B}_l^{-1}}\Big(v_k^{(l)} + \mathsf{U}_l(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}(w_k^{(j)} - \mathsf{W}_j\sum_{l'\in\mathbb{L}_j^*}\mathsf{L}_{l',j}^* v_k^{(l')}) - \mathsf{D}_l^{-1}v_k^{(l)} + d_k^{(l)})\Big) + b_k^{(l)}\Big)$

$v_{k+1}^{(l)} = v_k^{(l)} + \lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)} - v_k^{(l)})$

pour $j = 1,\dots,p$

$x_{k+1}^{(j)} = x_k^{(j)} + \lambda_k\varepsilon_k^{(j)}\Big(w_k^{(j)} - \mathsf{W}_j\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^* u_k^{(l)} - x_k^{(j)}\Big)$.

Proposition 7.5. *Considérons l'algorithme 7.5, (μ,ν) donnés dans la proposition 7.3, et $\mathbf{W}$ défini par* (7.20)*. Supposons que*

$$\min\{\mu, \nu(1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)\} > \frac{1}{2}, \tag{7.29}$$

et que les éléments de la suite $(\lambda_k)_{k\in\mathbb{N}}$ sont dans $]0,1]$ et vérifient $\inf_{k\in\mathbb{N}}\lambda_k > 0$. Posons $(\forall k\in\mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ et $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant k'\leqslant k}$, et supposons de plus que l'assertion suivante est vérifiée

(i) $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$, $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{c}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$, *et* $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{d}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$ P*-presque sûrement,*

et que les condition (ii)-(iii) de la proposition 7.4 sont satisfaites.
Si, dans le problème 7.1, $(\forall j\in\{1,\dots,p\})$ $\mathsf{A}_j = 0$, alors $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire à valeurs dans $\mathbf{F}$, et $(\boldsymbol{v}_k)_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire à valeurs dans $\mathbf{F}^*$.

La démonstration de la proposition 7.5 est donnée dans l'annexe 7.C.

Remarque 7.4.

(i) *Pour tout* $j \in \{1,\dots,p\}$, *une constante de cocoercivité de* C_j *est donnée par* $\widetilde{\mu}_j \in]0,+\infty[$ *et, pour tout* $l \in \{1,\dots,q\}$, *une constante de forte monotonie de* D_l *est donnée par* $\widetilde{\nu}_l \in]0,+\infty[$. *En utilisant* (7.21)-(7.22), *une condition nécessaire pour que* (7.29) *soit satisfaite est*

$$\min\Big\{(\|\mathsf{W}_j\|^{-1}\widetilde{\mu}_j)_{1\leqslant j\leqslant p}, \Big(1-\sum_{j=1}^{p}\sum_{l=1}^{q}\|\mathsf{U}_l^{1/2}\mathsf{L}_{l,j}\mathsf{W}_j^{1/2}\|^2\Big)(\|\mathsf{U}_l\|^{-1}\widetilde{\nu}_k)_{1\leqslant l\leqslant q}\Big\} > \frac{1}{2}. \tag{7.30}$$

(ii) *Dans le cas où, pour tout* $l \in \{1,\dots,q\}$, $\mathsf{D}_l^{-1} = 0$, *nous pouvons utiliser une condition moins restrictive que la condition* (7.29) *donnée par*

$$\begin{cases} \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2 < 1 \\ \mu > \frac{1}{2}. \end{cases} \tag{7.31}$$

De plus, une condition nécessaire pour que (7.30) *soit satisfaite est alors*

$$\begin{cases} \displaystyle\sum_{j=1}^{p}\sum_{l=1}^{q}\|\mathsf{U}_l^{1/2}\mathsf{L}_{l,j}\mathsf{W}_j^{1/2}\|^2 < 1 \\ \min\big\{(\|\mathsf{W}_j\|^{-1}\widetilde{\mu}_j)_{1\leqslant j\leqslant p}\big\} > \frac{1}{2}. \end{cases}$$

Notons que cette condition est moins restrictive que (7.24).

De la même façon que dans la section 7.3.2, remarquons nous pourrions trouver une forme symétrique de l'algorithme 7.5. En effet, les rôles des variables primales et duales des problèmes (7.10) et (7.11) peuvent être échangées en choisissant, dans le problème dual (7.11), $(\forall l \in \{1,\dots,q\})$ $\mathsf{B}_l^{-1} = 0$ à la place de $(\forall j \in \{1,\dots,p\})$ $\mathsf{A}_j = 0$ dans le problème primal (7.10).

7.3.4 Application aux problèmes variationnels

Dans cette section nous allons montrer que les résultats obtenus dans la section 7.3 nous permettent de construire des algorithmes proximaux primaux-duaux alternés par bloc pour résoudre un ensemble de problèmes d'optimisation convexe non nécessairement lisses.

Nous considérons la classe de problèmes d'optimisation suivante :

Problème 7.2. *Pour tout* $j \in \{1,\dots,p\}$, *soit* $\mathsf{f}_j \in \Gamma_0(\mathcal{H}_j)$, *soit* $\mathsf{h}_j \in \Gamma_0(\mathcal{H}_j)$ *une fonction différentiable de gradient Lipschitz, et, pour tout* $l \in \{1,\dots,q\}$, *soit* $\mathsf{g}_l \in \Gamma_0(\mathcal{G}_l)$, *soit* $\mathsf{l}_l \in \Gamma_0(\mathcal{G}_l)$ *une fonction fortement convexe, et soit* $\mathsf{L}_{l,j} \in \mathcal{B}(\mathcal{H}_j,\mathcal{G}_l)$. *Supposons que* (7.8) *et* (7.9) *sont satisfaites, et qu'il existe* $(\overline{\mathsf{x}}^{(1)},\dots,\overline{\mathsf{x}}^{(p)}) \in \mathcal{H}_1 \oplus \cdots \oplus \mathcal{H}_p$ *tels que*

$$(\forall j \in \{1,\dots,p\}) \quad 0 \in \partial\mathsf{f}_j(\overline{\mathsf{x}}^{(j)}) + \nabla\mathsf{h}_j(\overline{\mathsf{x}}^{(j)}) + \sum_{l=1}^{q}\mathsf{L}_{l,j}^*(\partial\mathsf{g}_l \,\square\, \partial\mathsf{l}_l)\Big(\sum_{j'=1}^{p}\mathsf{L}_{l,j'}\overline{\mathsf{x}}^{(j')}\Big). \tag{7.32}$$

Soit $\widetilde{\mathsf{F}}$ *l'ensemble défini par*

$$\widetilde{\mathsf{F}} = \underset{\mathsf{x}^{(1)}\in\mathcal{H}_1,\ldots,\mathsf{x}^{(p)}\in\mathcal{H}_p}{\operatorname{Argmin}} \sum_{j=1}^{p}\big(\mathsf{f}_j(\mathsf{x}^{(j)})+\mathsf{h}_j(\mathsf{x}^{(j)})\big)+\sum_{l=1}^{q}(\mathsf{g}_l\,\square\,\mathsf{l}_l)\Big(\sum_{j=1}^{p}\mathsf{L}_{l,j}\mathsf{x}^{(j)}\Big) \tag{7.33}$$

et soit $\widetilde{\mathsf{F}}^*$ *l'ensemble des solutions du problème dual associé défini par :*

$$\widetilde{\mathsf{F}}^* = \underset{\mathsf{v}^{(1)}\in\mathcal{G}_1,\ldots,\mathsf{v}^{(q)}\in\mathcal{G}_q}{\operatorname{Argmin}} \sum_{j=1}^{p}(\mathsf{f}_j^*\,\square\,\mathsf{h}_j^*)\Big(-\sum_{l=1}^{q}\mathsf{L}_{l,j}^*\mathsf{v}^{(l)}\Big)+\sum_{l=1}^{q}\big(\mathsf{g}_l^*(\mathsf{v}^{(l)})+\mathsf{l}_l^*(\mathsf{v}^{(l)})\big). \tag{7.34}$$

L'objectif est de trouver un couple de variables aléatoires $(\widehat{\boldsymbol{x}},\widehat{\boldsymbol{v}})$ *tel que* $\widehat{\boldsymbol{x}}$ *soit à valeurs dans* $\widetilde{\mathsf{F}}$ *et* $\widehat{\boldsymbol{v}}$ *soit à valeurs dans* $\widetilde{\mathsf{F}}^*$.

Notons que sous certaines hypothèses peu contraignantes, on peut assurer que la condition d'inclusion dans le problème 7.2 est satisfaite :

Proposition 7.6. [Combettes, 2013, Prop. 5.3] *Considérons le problème 7.2, et supposons que* (7.33) *admet une solution. L'existence de* $(\overline{\mathsf{x}}^{(1)},\ldots,\overline{\mathsf{x}}^{(p)})\in\mathcal{H}_1\oplus\cdots\oplus\mathcal{H}_p$ *vérifiant* (7.32) *est garantie dans chacun des cas suivants :*

(i) *Pour tout* $j\in\{1,\ldots,p\}$, f_j *est à valeurs réelles et, pour tout* $l\in\{1,\ldots,q\}$, $(\mathsf{x}^{(j)})_{1\leqslant j\leqslant p}\mapsto\sum_{j=1}^{p}\mathsf{L}_{l,j}\mathsf{x}^{(j)}$ *est surjective.*

(ii) *Pour tout* $l\in\{1,\ldots,q\}$, g_l *ou* l_l *est à valeurs réelles.*

(iii) $(\mathcal{H}_j)_{1\leqslant j\leqslant p}$ *et* $(\mathcal{G}_l)_{1\leqslant l\leqslant q}$ *sont de dimension finie et* $(\forall j\in\{1,\ldots,p\})$ $(\exists\,\mathsf{x}^{(j)}\in\operatorname{ri}\operatorname{dom}\mathsf{f}_j)$ *tel que* $(\forall l\in\{1,\ldots,q\})$ $\sum_{j=1}^{p}\mathsf{L}_{l,j}\mathsf{x}^{(j)}\in\operatorname{ri}\operatorname{dom}\mathsf{g}_l+\operatorname{ri}\operatorname{dom}\mathsf{l}_l$.

Nous proposons de résoudre le problème 7.2 en utilisant l'algorithme 7.6 déduit de l'algorithme 7.3.

Algorithme 7.6 Algorithme primal-dual (version 1) appliqué dans un cadre variationnel

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{H}}$, et soient $\boldsymbol{v}_0$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$. Pour tout $j \in \{1,\dots,p\}$, soit $\mathsf{W}_j \in \mathcal{S}^+(\mathcal{H}_j)$, et, pour tout $l \in \{1,\dots,q\}$, soit $\mathsf{U}_l \in \mathcal{S}^+(\mathcal{G}_l)$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{p+q}$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

pour $j = 1,\dots,p$

$$y_k^{(j)} = \varepsilon_k^{(j)}\Big(\operatorname{prox}_{\mathsf{W}_j^{-1},\mathsf{f}_j}\Big(x_k^{(j)} - \mathsf{W}_j\big(\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^* v_k^{(l)} + \nabla\mathsf{h}_j(x_k^{(j)}) + c_k^{(j)}\big)\Big) + a_k^{(j)}\Big),$$

$$x_{k+1}^{(j)} = x_k^{(j)} + \lambda_k\varepsilon_k^{(j)}(y_k^{(j)} - x_k^{(j)}),$$

pour $l = 1,\dots,q$

$$u_k^{(l)} = \varepsilon_k^{(p+l)}\Big(\operatorname{prox}_{\mathsf{U}_l^{-1},\mathsf{g}_l^*}\Big(v_k^{(l)} + \mathsf{U}_l\big(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}(2y_k^{(j)} - x_k^{(j)}) - \nabla\mathsf{l}_l^*(v_k^{(l)}) + d_k^{(l)}\big)\Big) + b_k^{(l)}\Big),$$

$$v_{k+1}^{(l)} = v_k^{(l)} + \lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)} - v_k^{(l)}).$$

Le résultat de convergence suivant, concernant l'algorithme 7.6, se déduit de la proposition 7.3 :

Proposition 7.7. *Considérons l'algorithme 7.6 et* $\mathbf{W}$ *et* $\mathbf{U}$ *définis par* (7.20)*. Supposons que, pour tout* $j \in \{1,\dots,p\}$*, le gradient de* $\mathsf{h}_j \circ \mathsf{W}_j^{1/2}$ *est* μ_j^{-1}*-Lipschitz, avec* $\mu_j \in]0,+\infty[$*, et que, pour tout* $l \in \{1,\dots,q\}$*, le gradient de* $\mathsf{l}_l^* \circ \mathsf{U}_l^{1/2}$ *est* ν_l^{-1}*-Lipschitz, avec* $\nu_l \in]0,+\infty[$*. Supposons de plus que* (7.19) *est satisfaite où* ϑ_α *est défini par* (7.16)*,* $\mu = \min\{\mu_1,\dots,\mu_p\}$*, et* $\nu = \min\{\nu_1,\dots,\nu_q\}$*. Supposons que les éléments de* $(\lambda_k)_{k\in\mathbb{N}}$ *sont dans* $]0,1]$ *et vérifient* $\inf_{k\in\mathbb{N}}\lambda_k > 0$*. Posons* $(\forall k \in \mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ *et* $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant k'\leqslant k}$ *et supposons que les conditions (i)-(iii) de la proposition 7.3 sont vérifiées.*

La suite $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ *converge alors faiblement* P*-presque sûrement vers une variable aléatoire à valeurs dans* $\widetilde{\mathbf{F}}$*, et* $(\boldsymbol{v}_k)_{k\in\mathbb{N}}$ *converge faiblement* P*-presque sûrement vers une variable aléatoire à valeurs dans* $\widetilde{\mathbf{F}}^*$*.*

La démonstration de la proposition 7.7 est donnée dans l'annexe 7.D.

De même que pour obtenir l'algorithme 7.4, les problèmes (7.33) et (7.34) étant symétriques, les rôles des variables primales et duales peuvent être échangées dans les deux problèmes. On obtient alors l'algorithme 7.7, dont la démonstration de convergence se déduit en remplaçant les conditions (ii)-(iii) de la proposition 7.3 par les conditions (ii)-(iii) de la proposition 7.4.

Algorithme 7.7 Algorithme primal-dual (version 1 symétrique) appliqué dans un cadre variationnel

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{H}}$, et soient $\boldsymbol{v}_0$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$. Pour tout $j \in \{1,\dots,p\}$, soit $\mathsf{W}_j \in \mathcal{S}^+(\mathcal{H}_j)$, et, pour tout $l \in \{1,\dots,q\}$, soit $\mathsf{U}_l \in \mathcal{S}^+(\mathcal{G}_l)$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{p+q}$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

pour $l = 1,\dots,q$

$$u_k^{(l)} = \varepsilon_k^{(p+l)}\Big(\operatorname{prox}_{\mathsf{U}_l^{-1},\mathsf{g}_l^*}\Big(v_k^{(l)} + \mathsf{U}_l\big(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}x_k^{(j)} - \nabla\mathsf{l}_l^* v_k^{(l)} + d_k^{(l)}\big)\Big) + b_k^{(l)}\Big),$$

$$v_{k+1}^{(l)} = v_k^{(l)} + \lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)} - v_k^{(l)}),$$

pour $j = 1,\dots,p$

$$y_k^{(j)} = \varepsilon_k^{(j)}\Big(\operatorname{prox}_{\mathsf{W}_j^{-1},\mathsf{f}_j}\Big(x_k^{(j)} - \mathsf{W}_j\big(\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^*(2u_k^{(l)} - v_k^{(l)}) + \nabla\mathsf{h}_j x_k^{(j)} + c_k^{(j)}\big)\Big) + a_k^{(j)}\Big),$$

$$x_{k+1}^{(j)} = x_k^{(j)} + \lambda_k\varepsilon_k^{(j)}(y_k^{(j)} - x_k^{(j)}).$$

Proposition 7.8. *Considérons l'algorithme 7.7, et* $\mathbf{W}$ *et* $\mathbf{U}$ *définis par* (7.20) *et supposons que, pour tout* $j \in \{1,\dots,p\}$, *le gradient de* $\mathsf{h}_j\circ\mathsf{W}_j^{1/2}$ *est* μ_j^{-1}*-Lipschitz, avec* $\mu_j \in]0,+\infty[$, *et que, pour tout* $l \in \{1,\dots,q\}$, *le gradient de* $\mathsf{l}_l^*\circ\mathsf{U}_l^{1/2}$ *est* ν_l^{-1}*-Lipschitz, avec* $\nu_l \in]0,+\infty[$. *Supposons de plus que* (7.19) *est satisfaite où* ϑ_α *est défini par* (7.16), $\mu = \min\{\mu_1,\dots,\mu_p\}$, *et* $\nu = \min\{\nu_1,\dots,\nu_q\}$. *Supposons que les éléments de* $(\lambda_k)_{k\in\mathbb{N}}$ *sont dans* $]0,1]$ *et vérifient* $\inf_{k\in\mathbb{N}}\lambda_k > 0$. *Posons* $(\forall k\in\mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ *et* $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant k'\leqslant k}$ *et supposons que la condition (i) de la proposition 7.3 et les conditions (ii)-(iii) de la proposition 7.4 sont vérifiées.*
La suite $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ *converge alors faiblement* P*-presque sûrement vers une variable aléatoire à valeurs dans* $\widetilde{\mathbf{F}}$, *et* $(\boldsymbol{v}_k)_{k\in\mathbb{N}}$ *converge faiblement* P*-presque sûrement vers une variable aléatoire à valeurs dans* $\widetilde{\mathbf{F}}^*$.

De façon similaire, l'algorithme 7.5 nous permet d'obtenir l'algorithme 7.8, et nous pouvons déduire de la proposition 7.5 le résultat de convergence suivant.

Proposition 7.9. *Considérons l'algorithme 7.8,* $\mathbf{W}$ *et* $\mathbf{U}$ *définis par* (7.20) *et* (μ,ν) *définis par la proposition 7.7. Supposons que la condition* (7.29) *est satisfaite et que les éléments de* $(\lambda_k)_{k\in\mathbb{N}}$ *sont dans* $]0,1]$ *et vérifient* $\inf_{k\in\mathbb{N}}\lambda_k > 0$. *Posons* $(\forall k\in\mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ *et* $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant k'\leqslant k}$ *et supposons que la condition (i) de la proposition 7.5, les conditions (ii)-(iii) de la proposition 7.4 sont satisfaites.*
Si, dans le problème 7.2, $(\forall j\in\{1,\dots,p\})$ $\mathsf{f}_j = 0$, *alors* $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ *converge faiblement* P*-presque*

sûrement vers une variable aléatoire à valeurs dans $\widetilde{\mathsf{F}}$*, et* $(\boldsymbol{v}_k)_{k\in\mathbb{N}}$ *converge faiblement* P*-presque sûrement vers une variable aléatoire à valeurs dans* $\widetilde{\mathsf{F}}^*$*.*

Algorithme 7.8 Algorithme primal-dual (version 2) appliqué dans un cadre variationnel

Initialisation : Soient $\boldsymbol{x}_0$, et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{H}}$, et soient $\boldsymbol{v}_0$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$. Pour tout $j\in\{1,\dots,p\}$, soit $\mathsf{W}_j\in\mathcal{S}^+(\mathcal{H}_j)$, et, pour tout $l\in\{1,\dots,q\}$, soit $\mathsf{U}_l\in\mathcal{S}^+(\mathcal{G}_l)$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{p+q}$. Soit, pour tout $k\in\mathbb{N}$, $\lambda_k\in\left]0,+\infty\right[$.

Itérations :

Pour $k=0,1,\dots$

pour $j=1,\dots,p$

$\eta_k^{(j)}=\max\left\{\varepsilon_k^{(p+l)}\;\middle|\; l\in\mathbb{L}_j^*\right\}$

$w_k^{(j)}=\eta_k^{(j)}\Big(x_k^{(j)}-\mathsf{W}_j(\nabla\mathsf{h}_j(x_k^{(j)})+c_k^{(j)})\Big)$

pour $l=1,\dots,q$

$$u_k^{(l)}=\varepsilon_k^{(p+l)}\Bigg(\operatorname{prox}_{\mathsf{U}_l^{-1},\mathsf{g}_l^*}\Big(v_k^{(l)}+\mathsf{U}_l\Big(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}\Big(w_k^{(j)}-\mathsf{W}_j\sum_{l'\in\mathbb{L}_j^*}\mathsf{L}_{l',j}^*v_k^{(l')}\Big)-\nabla\mathsf{l}_l^*(v_k^{(l)})+d_k^{(l)}\Big)\Big)+b_k^{(l)}\Bigg)$$

$v_{k+1}^{(l)}=v_k^{(l)}+\lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)}-v_k^{(l)})$

pour $j=1,\dots,p$

$$x_{k+1}^{(j)}=x_k^{(j)}+\lambda_k\varepsilon_k^{(j)}\Big(w_k^{(j)}-\mathsf{W}_j\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^*u_k^{(l)}-x_k^{(j)}\Big).$$

Dans la remarque suivante nous donnons les liens existants entre les algorithmes proximaux alternés par bloc proposés dans cette section et les travaux existants dans la littérature.

Remarque 7.5.

(i) *En pratique, il est souvent intéressant de considérer le cas particulier du problème* (7.2) *où* $(\forall l\in\{1,\dots,q\})\ \mathsf{l}_l=\iota_{\{0\}}$*, i.e.* $\mathsf{l}_l^*=0$*. Dans ce cas, l'ensemble* $\widetilde{\mathsf{F}}$ *se réécrit*

$$\widetilde{\mathsf{F}}=\operatorname*{Argmin}_{\mathsf{x}^{(1)}\in\mathcal{H}_1,\dots,\mathsf{x}^{(p)}\in\mathcal{H}_p}\ \sum_{j=1}^{p}\big(\mathsf{f}_j(\mathsf{x}^{(j)})+\mathsf{h}_j(\mathsf{x}^{(j)})\big)+\sum_{l=1}^{q}\mathsf{g}_l\Big(\sum_{j=1}^{p}\mathsf{L}_{l,j}\mathsf{x}^{(j)}\Big).$$

(ii) *L'algorithme 7.6 généralise les approches déterministes proposées dans* [Chambolle et Pock, 2010; Condat, 2013; Esser *et al.*, 2010; He et Yuan, 2012; Vũ, 2013] *présentant le cas où* $p = 1$*, en y introduisant un balayage aléatoire des blocs et en considérant des erreurs stochastiques.*

De façon similaire, l'algorithme 7.8 généralise les algorithmes présentés dans [Chen *et al.*, 2013; Loris et Verhoeven, 2011]*, développés dans le cas déterministe ne prenant en compte aucun terme d'erreur, dans le cas particulier où* $p = q = 1$*,* $\mathcal{H}_1$ *et* $\mathcal{G}_1$ *sont des espaces de dimension finie,* $\mathsf{l}_1 = \iota_{\{0\}}$*,* $\mathsf{W}_1 = \tau\mathrm{Id}$ $\mathsf{U}_1 = \rho\mathrm{Id}$*, avec* $(\tau, \rho) \in]0, +\infty[^2$*, sans relaxation* ($\lambda_k \equiv 1$) *ou bien avec une relaxation constante* ($\lambda_k \equiv \lambda_0 < 1$)*.*

Récemment, ces travaux ont été généralisés au cas où les espaces de Hilbert peuvent être de dimension infinie, pour $p = 1$ *et* $q > 1$*, en utilisant des opérateurs de préconditionnement arbitraires, et en prenant en compte des termes d'erreurs déterministes et sommables* [Combettes *et al.*, 2014].

L'intérêt pratique d'utiliser des opérateurs de préconditionnement pour des méthodes primales-duales, afin d'en accélérer la convergence, a été démontré dans [Combettes *et al.*, 2014; Pock et Chambolle, 2011; Repetti *et al.*, 2012].

(iii) *Dans* [Combettes et Pesquet, 2015, Cor. 5.5]*, un autre algorithme primal-dual alterné aléatoirement par bloc a été proposé pour résoudre un cas particulier du problème 7.2, obtenu lorsque* $(\forall j \in \{1, \ldots, p\})$ $\mathsf{h}_j = 0$ *et* $(\forall l \in \{1, \ldots, q\})$ $\mathsf{l}_l = \iota_{\{0\}}$*. Cet algorithme est obtenu en se basant sur les itérations de l'algorithme de Douglas-Rachford. On peut aussi mentionner l'algorithme ADMM aléatoire développé pour des espaces de dimensions finies dans* [Iutzeler *et al.*, 2013]*. Remarquons cependant que l'algorithme donné dans* [Combettes et Pesquet, 2015, Cor. 5.5] *requiert l'inversion de* $\mathbf{Id} + \mathbf{L}\mathbf{L}^*$ *ou* $\mathbf{Id} + \mathbf{L}^*\mathbf{L}$ *(voir* [Combettes et Pesquet, 2015, Rmq. 5.4]*), contrairement aux algorithmes 7.6 et 7.8 qui ne nécessitent aucune inversion d'opérateur linéaire.*

Le diagramme donné dans la figure 7.5, présentée à la fin de la section 7.4, résume les liens existants entre les algorithmes proposés dans les sections 7.3 et 7.4.

7.4 Application de l'algorithme primal-dual alterné aléatoirement par bloc

7.4.1 Description du problème

Considérons un maillage surfacique représenté par ses p nœuds de coordonnées spatiales $\overline{\mathbf{x}} = (\overline{\mathbf{x}}^{(j)})_{1 \leqslant j \leqslant p} \in \mathbb{R}^{p \times 3}$, où, pour tout $j \in \{1, \ldots, p\}$, $\overline{\mathbf{x}}^{(j)} = (\overline{\mathbf{x}}^{(\mathcal{X}_j)}, \overline{\mathbf{x}}^{(\mathcal{Y}_j)}, \overline{\mathbf{x}}^{(\mathcal{Z}_j)}) \in \mathbb{R}^3$ représente les

coordonnées spatiales du j-ème nœud, ainsi que par sa matrice d'adjacence[3] $\mathbf{A} \in \mathbb{R}^{p \times p}$.

Nous nous intéressons au problème d'estimation de ce maillage à partir d'une observation inexacte de celui-ci, ayant la même "topologie" (i.e. la même matrice d'adjacence $\mathbf{A}$), mais dont la position des nœuds est connue de façon imprécise selon le modèle

$$\mathbf{z} = (\mathbf{z}^{(\mathcal{X})}, \mathbf{z}^{(\mathcal{Y})}, \mathbf{z}^{(\mathcal{Z})}) = \overline{\mathbf{x}} + \mathbf{b}, \tag{7.35}$$

où $\mathbf{b} = (\mathsf{b}_j)_{1 \leqslant j \leqslant p} \in \mathbb{R}^{p \times 3}$ est une réalisation d'une variable aléatoire modélisant les incertitudes des positions spatiales des nœuds du maillage. Une méthode permettant de produire une estimée $\widehat{\mathbf{x}} \in \mathbb{R}^{p \times 3}$ de $\overline{\mathbf{x}}$, est de la définir comme une solution du problème (7.2), en prenant $(\forall l \in \{1, \ldots, q\})$ $\mathsf{l}_l \equiv \iota_{\{0\}}$, avec les fonctions $(\mathsf{f}_j)_{1 \leqslant j \leqslant p}$, $(\mathsf{h}_j)_{1 \leqslant j \leqslant p}$ et $(\mathsf{g}_l)_{1 \leqslant l \leqslant q}$ choisies de façon à incorporer des informations concernant le modèle d'observation ainsi que des connaissances *a priori* sur la position des nœuds du maillage [Couprie *et al.*, 2013; Shuman *et al.*, 2013].

Nous proposons d'utiliser un terme d'attache aux données nous permettant d'obtenir une estimation robuste, correspondant à la somme sur $j \in \{1, \ldots, p\}$ des fonctions définies par :

$$(\forall \mathsf{x}^{(j)} = (\mathsf{x}^{(\mathcal{D}_j)})_{\mathcal{D} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{Z}\}} \in \mathbb{R}^3) \quad \mathsf{h}_j(\mathsf{x}^{(j)}) = \sum_{\mathcal{D} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{Z}\}} \varphi_j(\mathsf{x}^{(\mathcal{D}_j)} - \mathsf{z}^{(\mathcal{D}_j)}), \tag{7.36}$$

où, pour tout $j \in \{1, \ldots, p\}$, φ_j est la fonction de Huber paramétrée par $\delta^{(j)} \in]0, +\infty]$, définie par (2.28), qui est différentiable et de gradient Lipschitz.

De plus, afin d'obtenir un maillage estimé lisse, nous proposons d'utiliser comme fonction de régularisation une variation totale isotrope [Elmoataz *et al.*, 2008; Grady et Polimeni, 2010] définie, pour tout $\mathbf{x} \in \mathbb{R}^{p \times 3}$, par

$$\sum_{l=1}^{p} \mathsf{g}_l(\mathsf{L}_l \mathbf{x}) = \sum_{l=1}^{p} \beta^{(l)} \sum_{\mathcal{D} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{Z}\}} \|(\mathsf{x}^{(\mathcal{D}_l)} - \mathsf{x}^{(\mathcal{D}_i)})_{i \in \mathcal{V}_l}\|_2, \tag{7.37}$$

avec, en particulier,

$$(\forall l \in \{1, \ldots, p\}) \quad \mathsf{L}_l \colon \mathbb{R}^{p \times 3} \to \mathbb{R}^{\kappa_l \times 3} \colon \mathbf{x} \mapsto \sum_{j=1}^{p} \mathsf{L}_{l,j} \mathsf{x}^{(j)} = \left(\mathsf{x}^{(l)} - \mathsf{x}^{(i)}\right)_{i \in \mathcal{V}_l}, \tag{7.38}$$

$$\mathsf{g}_l \colon \mathbb{R}^{\kappa_l \times 3} \to \mathbb{R} \colon \mathsf{u} \to \beta^{(l)} \sum_{\mathcal{D} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{Z}\}} \|\mathsf{u}^{\mathcal{D}}\|_2, \tag{7.39}$$

où, pour tout $l \in \{1, \ldots, p\}$, $\beta^{(l)} \in]0, +\infty[$, et $\mathcal{V}_l$, de cardinal κ_l, désigne l'ensemble des voisins du nœud l (c'est-à-dire l'ensemble des indices $i \in \{1, \ldots, p\}$ tels que $\mathsf{A}^{(i,l)} = 1$). Les ensembles définis par (7.8) et (7.9) sont alors

$$\begin{aligned} &(\forall l \in \{1, \ldots, p\}) \quad \mathbb{L}_l = \{l\} \cup \mathcal{V}_l, \\ &(\forall j \in \{1, \ldots, p\}) \quad \mathbb{L}_j^* = \{l \in \{1, \ldots, p\} \mid l = j \text{ ou } j \in \mathcal{V}_l\}. \end{aligned}$$

Finalement, nous ajoutons un terme de contrainte de façon à s'assurer que les nœuds estimés soient compris dans un hypercube paramétré par des positions spatiales minimales et maximales données $(\mathsf{x}_{\min}^{(\mathcal{D})}, \mathsf{x}_{\max}^{(\mathcal{D})})_{\mathcal{D} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{Z}\}} \in (\mathbb{R}^3)^2$:

$$(\forall j \in \{1, \ldots, p\}) \quad \mathsf{f}_j(\mathsf{x}^{(j)}) = \sum_{\mathcal{D} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{Z}\}} \iota_{[\mathsf{x}_{\min}^{(\mathcal{D})}, \mathsf{x}_{\max}^{(\mathcal{D})}]}(\mathsf{x}^{(\mathcal{D}_j)}). \tag{7.40}$$

3. La matrice d'adjacence $\mathbf{A} = (\mathbf{A}^{(j,j')})_{1 \leqslant j,j' \leqslant p} \in \mathbb{R}^{p \times p}$ vérifie $\mathbf{A}^{(j,j')} = 1$ lorsque les nœuds j et j' (pour $j \neq j'$) sont connectés, et $\mathbf{A}^{(j,j')} = 0$ sinon.

7.4.2 Méthodes proposées

Nous proposons de tester indépendamment les deux versions des algorithmes que nous avons proposées dans la section 7.3.4 afin de résoudre le problème décrit ci-dessus. Pour cela, nous allons proposons des versions simplifiées des algorithmes 7.4 et 7.8 définies respectivement par les algorithmes 7.9 et 7.10.

7.4.2.1 Première méthode

Dans un premier temps, nous proposons d'utiliser l'algorithme 7.9.

Algorithme 7.9 Algorithme primal-dual (version 1 symétrique simplifiée) appliqué dans un cadre variationnel

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{H}}$, et soient $\boldsymbol{v}_0$, et $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$. Pour tout $j \in \{1,\dots,p\}$, soit $\mathsf{W}_j \in \mathcal{S}^+(\mathcal{H}_j)$, et, pour tout $l \in \{1,\dots,q\}$, soit $\mathsf{U}_l \in \mathcal{S}^+(\mathcal{G}_l)$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_p$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

pour $l = 1,\dots,q$

$\eta_k^{(l)} = \max_{1\leqslant j\leqslant p}\left\{\varepsilon_k^{(j)} \;\middle|\; l \in \mathbb{L}_j^*\right\}$

$u_k^{(l)} = \eta_k^{(l)}\Big(\operatorname{prox}_{\mathsf{U}_l^{-1},\mathsf{g}_l^*}\Big(v_k^{(l)} + \mathsf{U}_l\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}x_k^{(j)}\Big) + b_k^{(l)}\Big)$,

$v_{k+1}^{(l)} = v_k^{(l)} + \lambda_k\eta_k^{(l)}(u_k^{(l)} - v_k^{(l)})$,

pour $j = 1,\dots,p$

$y_k^{(j)} = \varepsilon_k^{(j)}\Big(\operatorname{prox}_{\mathsf{W}_j^{-1},\mathsf{f}_j}\Big(x_k^{(j)} - \mathsf{W}_j(\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^*(2u_k^{(l)} - v_k^{(l)}) + \nabla\mathsf{h}_j x_k^{(j)} + c_k^{(j)})\Big) + a_k^{(j)}\Big)$,

$x_{k+1}^{(j)} = x_k^{(j)} + \lambda_k\varepsilon_k^{(j)}(y_k^{(j)} - x_k^{(j)})$.

Notons que, pour tout $k \in \mathbb{N}$ et pour tout $l \in \{1,\dots,q\}$, le choix

$$\eta_k^{(l)} = \max_{1\leqslant j\leqslant p}\{\varepsilon_k^{(j)} \mid l \in \mathbb{L}_j^*\}$$

dans l'algorithme 7.9 permet d'assurer que la condition (iii) de la Proposition 7.4 soit satisfaite. Ainsi, la convergence de l'algorithme 7.9 est assurée par la Proposition 7.8.

Cet algorithme requiert de choisir les matrices $(\mathsf{W}_j)_{1\leqslant j\leqslant p}$ et $(\mathsf{U}_l)_{1\leqslant l\leqslant q}$. Dans notre exemple nous prendrons simplement $q = p$, pour tout $j \in \{1,\dots,p\}$, $\mathsf{W}_j \equiv \tau\,\mathbf{I}_3$ et, pour tout $l \in \{1,\dots,p\}$, $\mathsf{U}_l \equiv \rho\,\mathbf{I}_{\kappa_l}$, où (τ,ρ) sont des constantes positives vérifiant la condition de convergence

$$2\min\{(\mu_j)_{1\leqslant j\leqslant p}\}\left(1 - \|\mathbf{W}^{1/2}\mathbf{L}\mathbf{U}^{1/2}\|^2\right) = 2\min\{(\mu_j)_{1\leqslant j\leqslant p}\}\left(1 - \tau\rho\|\mathbf{L}\|^2\right) > 1, \tag{7.41}$$

obtenue lorsque $\mathsf{l}_l \equiv \iota_{\{0\}}$ et où $\mathbf{L}\colon \mathbf{x} \in \mathbb{R}^{p\times 3} \mapsto (\mathsf{L}_l\mathbf{x})_{1\leqslant l\leqslant p}$. De plus, si, pour tout $j \in \{1,\dots,p\}$, $\widetilde{\mu}_j^{-1}$ est la constante de Lipschitz de h_j, alors la contrainte ci-dessus est satisfaite en choisissant $(\tau,\rho) \in]0,+\infty[^2$ vérifiant

$$\min\{(\widetilde{\mu}_j)_{1\leqslant j\leqslant p}\}(\tau^{-1} - \rho\|\mathbf{L}\|^2) > 1/2. \tag{7.42}$$

7.4.2.2 Seconde méthode

Algorithme 7.10 Algorithme primal-dual (version 2 simplifiée) appliqué dans un cadre variationnel

Initialisation : Soient $\boldsymbol{x}_0$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{H}}$, et soient $\boldsymbol{v}_0$ et $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$. Pour tout $j \in \{1,\dots,p\}$, soit $\mathsf{W}_j \in \mathcal{S}^+(\mathcal{H}_j)$, et, pour tout $l \in \{1,\dots,q\}$, soit $\mathsf{U}_l \in \mathcal{S}^+(\mathcal{G}_l)$. Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{p+q}$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

pour $j = 1,\dots,p$

$\eta_k^{(j)} = \max\left\{\varepsilon_k^{(p+l)} \;\middle|\; l \in \mathbb{L}_j^*\right\}$

$w_k^{(j)} = \eta_k^{(j)}\Big(x_k^{(j)} - \mathsf{W}_j(\nabla \mathsf{h}_j(x_k^{(j)}) + c_k^{(j)})\Big)$

pour $l = 1,\dots,q$

$\varepsilon_k^{(p+l)} = \max\left\{\varepsilon_k^{(j)} \;\middle|\; j \in \mathbb{L}_l\right\}$

$u_k^{(l)} = \varepsilon_k^{(p+l)}\Big(\operatorname{prox}_{\mathsf{U}_l^{-1},\mathsf{g}_l^*}\Big(v_k^{(l)} + \mathsf{U}_l(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}(w_k^{(j)} - \mathsf{W}_j\sum_{l'\in\mathbb{L}_j^*}\mathsf{L}_{l',j}^* v_k^{(l')})\Big) + b_k^{(l)}\Big)$

$v_{k+1}^{(l)} = v_k^{(l)} + \lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)} - v_k^{(l)})$

pour $j = 1,\dots,p$

$x_{k+1}^{(j)} = x_k^{(j)} + \lambda_k\varepsilon_k^{(j)}\Big(w_k^{(j)} - \mathsf{W}_j\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^* u_k^{(l)} - x_k^{(j)}\Big)$.

Dans l'algorithme 7.10, pour tout $k \in \mathbb{N}$ et pour tout $l \in \{1,\dots,q\}$, la condition $\varepsilon_k^{(p+l)} = \max\{\varepsilon_k^{(j)} \mid j \in \mathbb{L}_l\}$ permet d'assurer que la condition (iii) de la proposition 7.4 est satisfaite. Ainsi, la convergence de l'algorithme 7.10 est assurée par la proposition 7.9.

Afin d'appliquer ce second algorithme au problème décrit dans la section 7.4.1, nous posons $q = 2p$ et nous définissons, pour tout $l \in \{1,\dots,p\}$, $(\forall \mathbf{x} \in \mathbb{R}^{p\times 3})$ $\mathsf{g}_{p+l}(\sum_{j=1}^p \mathsf{L}_{p+l,j}\mathsf{x}^{(j)}) = \mathsf{f}_l(\mathsf{x}^{(l)})$, où f_l est définie par (7.40). Donc, pour tout $l \in \{1,\dots,p\}$,

$$\mathsf{g}_{p+l} = \mathsf{f}_l \quad \text{et} \quad (\forall j \in \{1,\dots,p\}) \quad \mathsf{L}_{p+l,j} = \begin{cases} \mathbf{I}_3 & \text{si } j = l \\ \mathbf{0}_3 & \text{sinon.} \end{cases}$$

Dans cet algorithme, nous choisissons, pour tout $j \in \{1,\dots,p\}$, $\mathsf{W}_j \equiv \tau\,\mathbf{I}_3$, $\mathsf{U}_j \equiv \rho_1\,\mathbf{I}_{\kappa_j}$ et $\mathsf{U}_{p+j} \equiv \rho_2\,\mathbf{I}_3$, où (τ,ρ_1,ρ_2) sont des constantes positives vérifiant la condition de convergence (7.31) donnée dans la remarque 7.4, obtenue lorsque $\mathsf{l}_l \equiv \iota_{\{0\}}$:

$$\begin{cases} \dfrac{\tau}{2} < \min\{(\widetilde{\mu}_j)_{1\leqslant j\leqslant p}\} \\ \tau\left(\rho_1\|\mathbf{L}\|^2+\rho_2\right) < 1, \end{cases} \tag{7.43}$$

où $(\forall l \in \{1,\dots,p\})$ L_l est donnée par (7.38) et, $(\forall j \in \{1,\dots,p\})$ $\widetilde{\mu}_j^{-1}$ est la constante de Lipschitz de h_j.

7.4.2.3 Implémentation

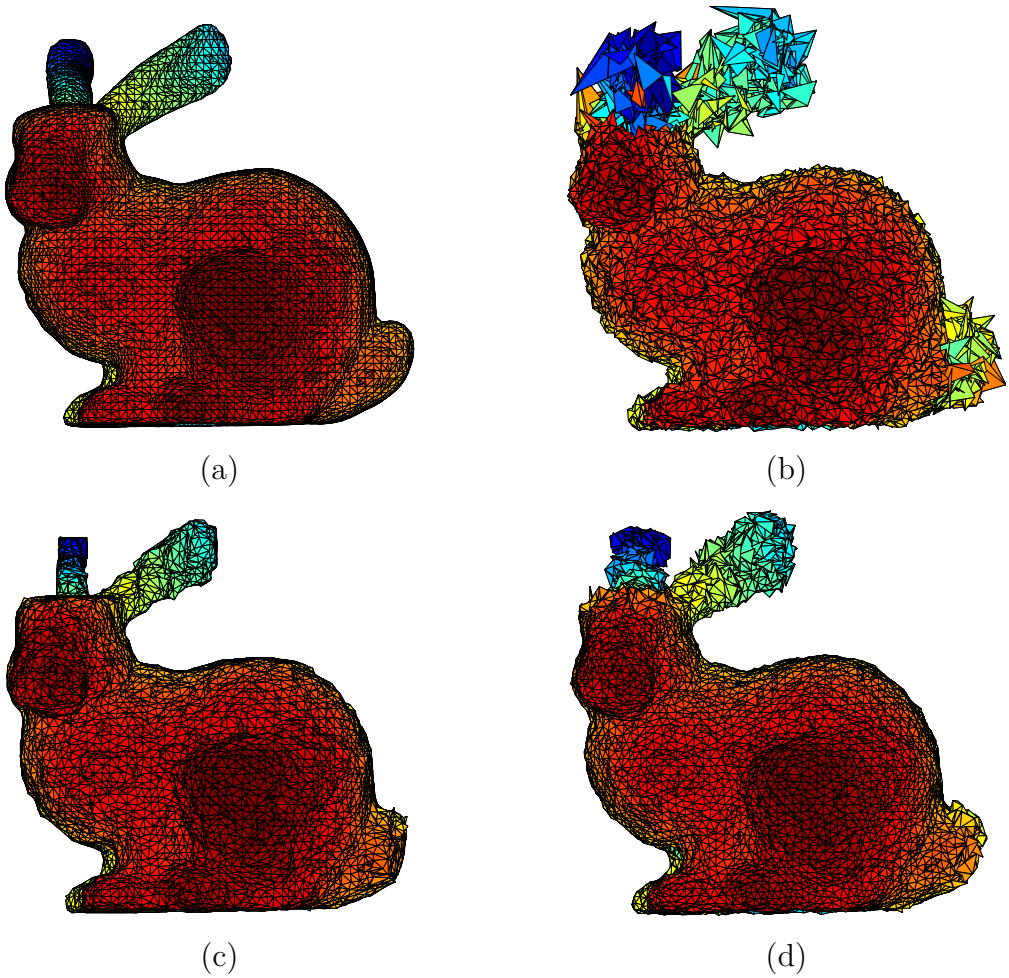
Puisque, pour tout $l \in \{1,\dots,p\}$, g_l correspond à une somme de normes ℓ_2, $\mathrm{prox}_{\mathsf{U}_l^{-1},\mathsf{g}_l^*}$ a une forme explicite [Combettes et Pesquet, 2010]. De plus, pour tout $j \in \{1,\dots,p\}$, l'opérateur proximal de f_j relatif à la métrique induite par W_j^{-1} correspond à un simple opérateur de projection sur $[\mathsf{x}_{\min}^{(\mathcal{X})},\mathsf{x}_{\max}^{(\mathcal{X})}] \times [\mathsf{x}_{\min}^{(\mathcal{Y})},\mathsf{x}_{\max}^{(\mathcal{Y})}] \times [\mathsf{x}_{\min}^{(\mathcal{Z})},\mathsf{x}_{\max}^{(\mathcal{Z})}]$.

Remarquons que, pour les deux algorithmes, les boucles internes sur les $l \in \{1,\dots,q\}$ et les $j \in \{1,\dots,q\}$ peuvent être calculées en parallèle. D'autre part, lorsque l'on traite de très grandes quantités de données, nous risquons d'être rapidement limités en terme de mémoire, et le temps de calcul peut devenir secondaire. Cette limitation de mémoire implique que seule une partie des variables globales peuvent être manipulées à chaque itération de l'algorithme.

7.4.3 Résultats numériques

7.4.3.1 Premier exemple

Dans ce premier exemple numérique, nous avons utilisé l'algorithme 7.13 pour résoudre le problème décrit dans la section 7.4.1. Dans nos simulations nous avons utilisé le maillage surfacique `Bunny`, disponible sur `http://graphics.stanford.edu/data/3Dscanrep/`, composé de $p = 8171$ nœuds, illustré en Figure 7.1(a). Le maillage bruité $\mathbf{z}$, représenté dans la Figure 7.1(b), résulte du modèle d'observation décrit dans la section 7.4.1. Plus précisément, nous définissons deux ensembles de nœuds $\mathcal{N}_1$ et $\mathcal{N}_2$, de cardinaux respectifs N_1 et N_2, vérifiant $N_1 + N_2 = p$. On suppose que les éléments de $(\mathsf{b}_j)_{j\in\mathcal{N}_1}$ modélisent un bruit blanc gaussien de moyenne nulle et d'écart-type $\sigma_1 = 10^{-3}$, tandis que les éléments de $(\mathsf{b}_j)_{j\in\mathcal{N}_2}$ sont indépendants et identiquement distribués selon une loi normale mixte, comprenant deux composantes de mélange pondérées par des poids $(\pi, 1-\pi)$ $(\pi = 0.98)$, de moyennes nulles, et d'écart-types respectifs $\sigma_2 = 5\times10^{-3}$ et $\sigma_2' = 1.5\times10^{-2}$. Dans notre application, nous avons choisit $\mathcal{N}_2$, de cardinal $N_2 = 1327$, comme étant l'ensemble des nœuds situés au niveau des oreilles et de la queue du lapin. Nous montrons dans la Figure 7.1(c) le maillage restauré obtenu en utilisant l'algorithme 7.9 avec $\lambda_k \equiv 1$. Les paramètres de régularisation $(\beta^{(l)})_{1\leqslant l\leqslant p}$ et $(\delta^{(j)})_{1\leqslant j\leqslant p}$ sont choisis différemment suivant qu'ils correspondent à l'ensemble $\mathcal{N}_1$ ou $\mathcal{N}_2$, de façon à minimiser l'EQM (défini par (2.20)) du maillage restauré. De plus, nous avons comparé le maillage obtenu après reconstruction par la méthode proposée, avec le maillage restauré par une méthode de filtrage de l'état de l'art [Taubin *et al.*, 1996], donné dans la Figure 7.1(d). Notons que, comme il l'a été remarqué dans [Couprie *et al.*,



(a) (b)

(c) (d)

Figure 7.1 – (a) *Maillage original* $\overline{\mathbf{x}}$, (b) *maillage bruité* $\mathbf{z}$ *avec EQM* $= 5.45 \times 10^{-6}$, *et maillage reconstruit* $\widehat{\mathbf{x}}$ *en utilisant* (c) *l'algorithme 7.9 avec EQM* $= 8.89 \times 10^{-7}$, *et* (d) *une méthode de lissage du Laplacien avec EQM* $= 1.29 \times 10^{-6}$.

2013], l'utilisation d'une méthode régularisée permet d'obtenir une amélioration en terme d'EQM par rapport à la méthode de [Taubin *et al.*, 1996].

L'algorithme 7.9 permet de sélectionner, à chaque itération, de façon flexible, les nœuds qui sont mis à jour. Afin de prendre en compte les variations spatiales du bruit, nous proposons de choisir, pour tout $k \in \mathbb{N}$ et $j \in \{1, \ldots, p\}$, $\varepsilon_k^{(j)}$ comme étant une variable aléatoire suivant une loi de Bernoulli de probabilité p_j telle que $\mathsf{p}_j = 1$ si $j \in \mathcal{N}_2$, et $\mathsf{p}_j = \mathsf{p} \in]0, 1]$ sinon. Remarquons que, pour $\mathsf{p} = 1$, on retrouve l'algorithme déterministe primal-dual de [Condat, 2013; Vũ, 2013]. Pour différentes valeurs de p, on évalue l'indicateur de performance défini par

$$\mathsf{C}(\mathsf{p}) = \overline{k}\,\frac{\mathsf{p}\mathcal{N}_1 + \mathcal{N}_2}{\mathcal{N}_1 + \mathcal{N}_2}, \tag{7.44}$$

où $\overline{k} \in \mathbb{N}^*$ correspond à la première itération de l'algorithme 7.9 pour laquelle

$$\|\boldsymbol{x}_{\overline{k}} - \boldsymbol{x}_{\overline{k}-1}\| \leqslant 10^{-6}\sqrt{3p}.$$

Pour une probabilité p fixée, $\mathsf{C}(\mathsf{p})$ peut être interprété comme le nombre d'itérations nécessaires pour atteindre la convergence, pondéré par le nombre de nœuds mis à jour. Nous montrons dans la Figure 7.2 l'évolution de l'indicateur $\mathsf{C}(\mathsf{p})$, moyenné sur dix exécutions de l'algorithme 7.9 (correspondant à dix réalisations différentes des variables aléatoires $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$). Nous observons alors que la valeur minimale de $\mathsf{C}(\mathsf{p})$ est obtenue pour $\mathsf{p} = 0.33$, ce qui montre que la stratégie optimale de l'algorithme 7.9 en terme de profil de convergence est obtenue lorsque seule une fraction des nœuds de l'ensemble $\mathcal{N}_1$ (correspondant aux nœuds dont la position est affectée par le bruit le plus faible) est mise à jour à chaque itération.

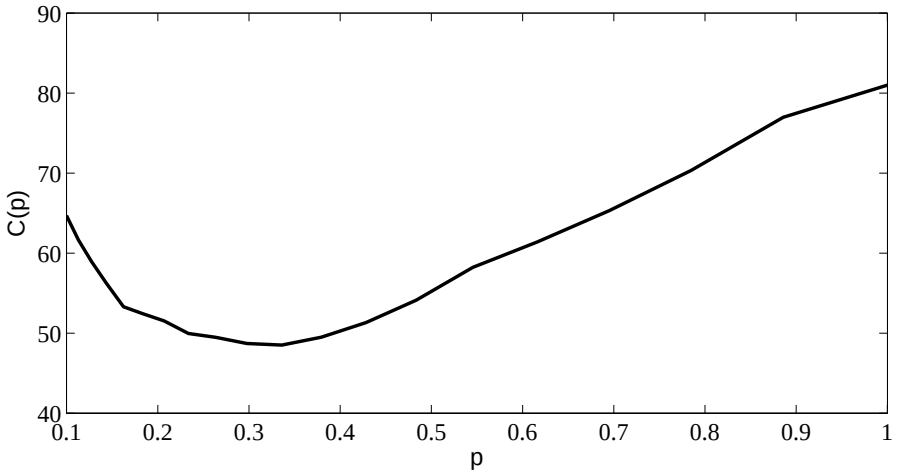


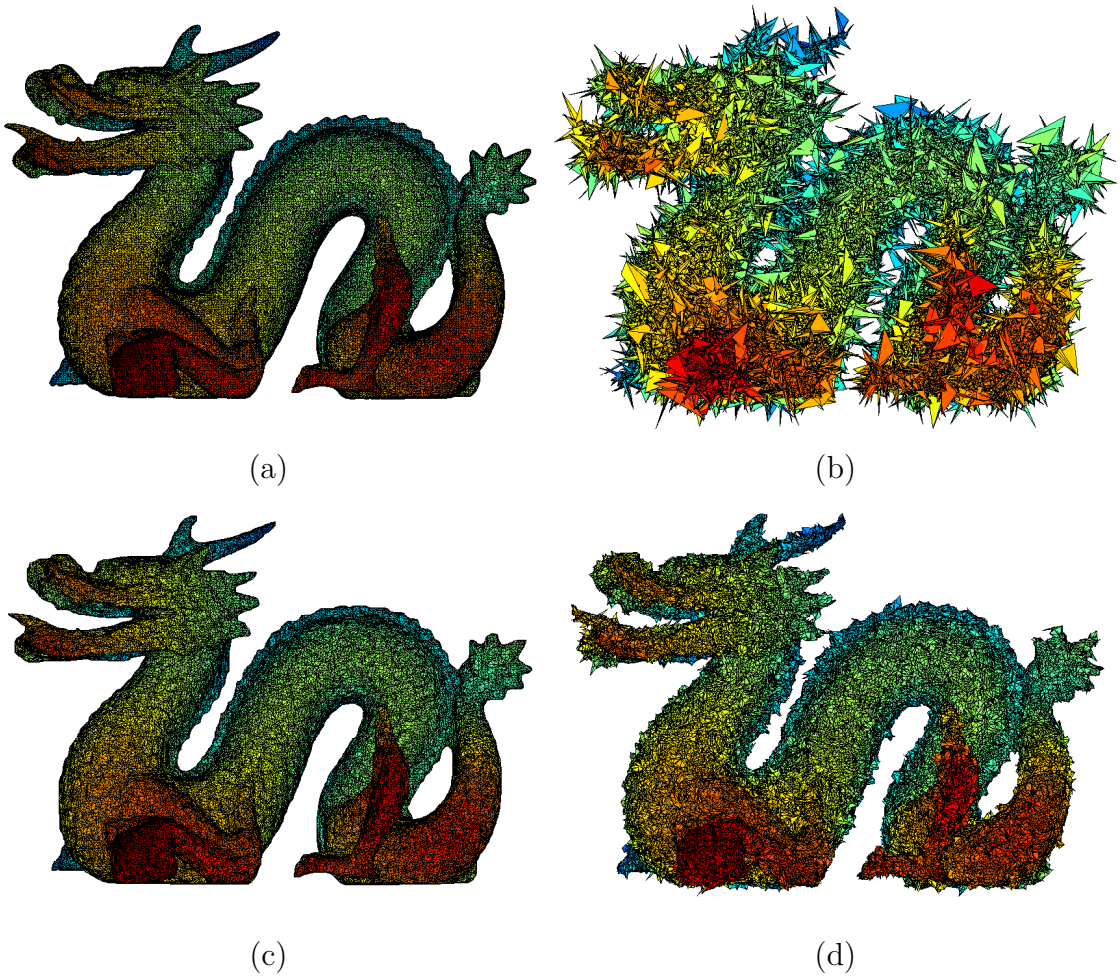
Figure 7.2 – *Indicateur* $\mathsf{C}(\mathsf{p})$ *obtenu pour différentes valeurs de la probabilité* p *(moyenné sur dix exécutions de l'algorithme 7.9).*

7.4.3.2 Second exemple

Dans ce second exemple numérique, nous avons utilisé l'algorithme 7.10 pour résoudre le problème décrit dans la section 7.4.1. Nous avons considéré le maillage surfacique `Dragon` de plus grande dimension disponible sur `http://graphics.stanford.edu/data/3Dscanrep/`, composé de $p = 100250$ nœuds, illustré en Figure 7.3(a). Les coordonnées du maillage original sont dégradées par un bruit distribué suivant un mélange de gaussiennes. Le maillage observé est donné dans la figure 7.3(b). Le maillage restauré obtenu en utilisant l'algorithme 7.10, avec $\lambda_k \equiv 1$ est représenté dans la figure 7.3(c). Les paramètres de régularisation $(\beta^{(l)})_{1\leqslant l\leqslant q}$ et $(\delta^{(j)})_{1\leqslant j\leqslant p}$ sont choisis de façon à minimiser l'EQM du maillage restauré. Afin d'avoir un élément de comparaison en terme de qualité de restauration, nous donnons dans la figure 7.3(d) le maillage restauré en utilisant la méthode de [Taubin *et al.*, 1996].

Dans cette application, nous divisons le maillage en une partition de p/r blocs, où chaque bloc est composé de $r \leqslant p$ nœuds. Nous choisissons, à chaque itération $k \in \mathbb{N}$, les variables booléennes $(\varepsilon_k^{(j)})_{1\leqslant j\leqslant p}$ de façon à ce que seuls les éléments liés aux r nœuds sélectionnés soient mis à jour à la k-ème itération. Ainsi, $(\varepsilon_k^{(j)})_{1\leqslant j\leqslant p}$ satisfait $\sum_{j=1}^{p} \varepsilon_k^{(j)} = r$.

La figure 7.4 représente à la fois le temps nécessaire pour atteindre le critère d'arrêt $\|\boldsymbol{x}_k - \widehat{\boldsymbol{x}}\| \leqslant 10^{-3}\|\widehat{\boldsymbol{x}}\|$, où $\widehat{\boldsymbol{x}}$ a été calculé au préalable pour un grand nombre d'itérations, et la mémoire



(a) (b)

(c) (d)

Figure 7.3 – (a) *Maillage original* $\overline{\mathbf{x}}$, (b) *maillage bruité* $\mathbf{z}$ *avec* $EQM = 2.89 \times 10^{-6}$, *et maillage reconstruit* $\widehat{\mathbf{x}}$ *en utilisant* (c) *l'algorithme 7.10 avec* $EQM = 8.09 \times 10^{-8}$, *et* (d) *une méthode de lissage du Laplacien avec* $EQM = 5.23 \times 10^{-7}$.

nécessaire pour l'exécution de l'algorithme avec Matlab R2013a, en fonction du nombre de blocs p/r. Nous pouvons remarquer que le temps minimum de reconstruction (égal à 5s.) est obtenu lorsque $p = 1$, i.e. avec une stratégie déterministe. Cependant, choisir des blocs plus petits permet de réduire l'occupation mémoire, au dépend d'une vitesse d'exécution plus lente. L'avantage de la méthode présentée ici pour des données de grande dimension, est qu'elle permet à l'utilisateur de choisir la stratégie qui lui permettra d'obtenir un temps de reconstruction minimal, sans dépasser la capacité de mémoire disponible sur sa propre architecture d'ordinateur.

Remarque 7.6. *Le diagramme donné dans la figure 7.5 résume les liens existants entre les différents algorithmes développés dans les sections 7.3 et 7.4. Bien que seul l'algorithme 7.9 ait été présenté avec un choix simplifié des variables booléennes* $(\varepsilon_k)_{k\in\mathbb{N}}$, *un choix similaire peut être appliqué aux autres algorithmes. Les cadres verts renvoient aux problèmes traités (problèmes 7.1*

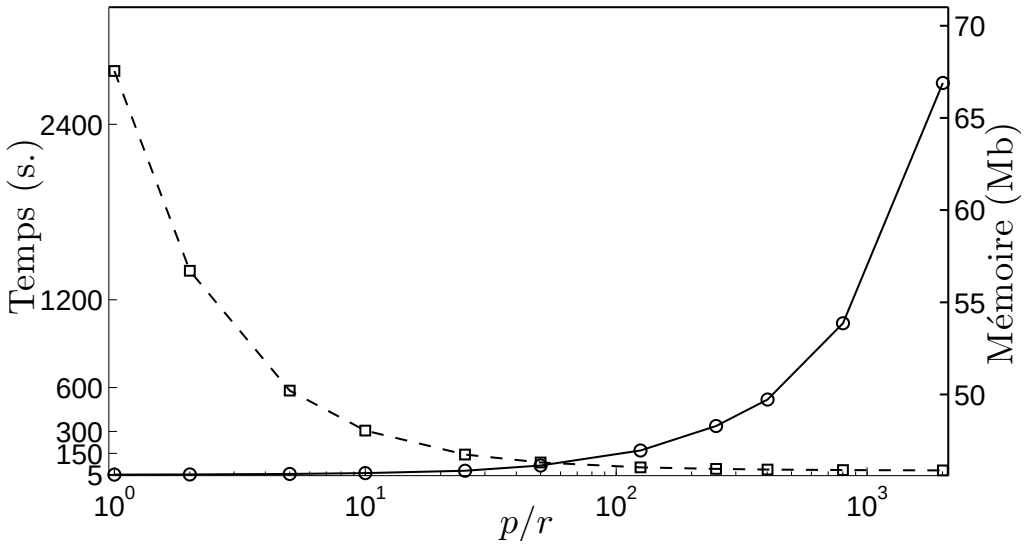


Figure 7.4 – *Temps de reconstruction (cercles) et mémoire nécessaire (carrés) pour différents nombres de blocs p/r.*

et 7.2) ainsi qu'à leurs cas particuliers (vert clair), et les ellipses bleues donnent les algorithmes permettant de les traiter.

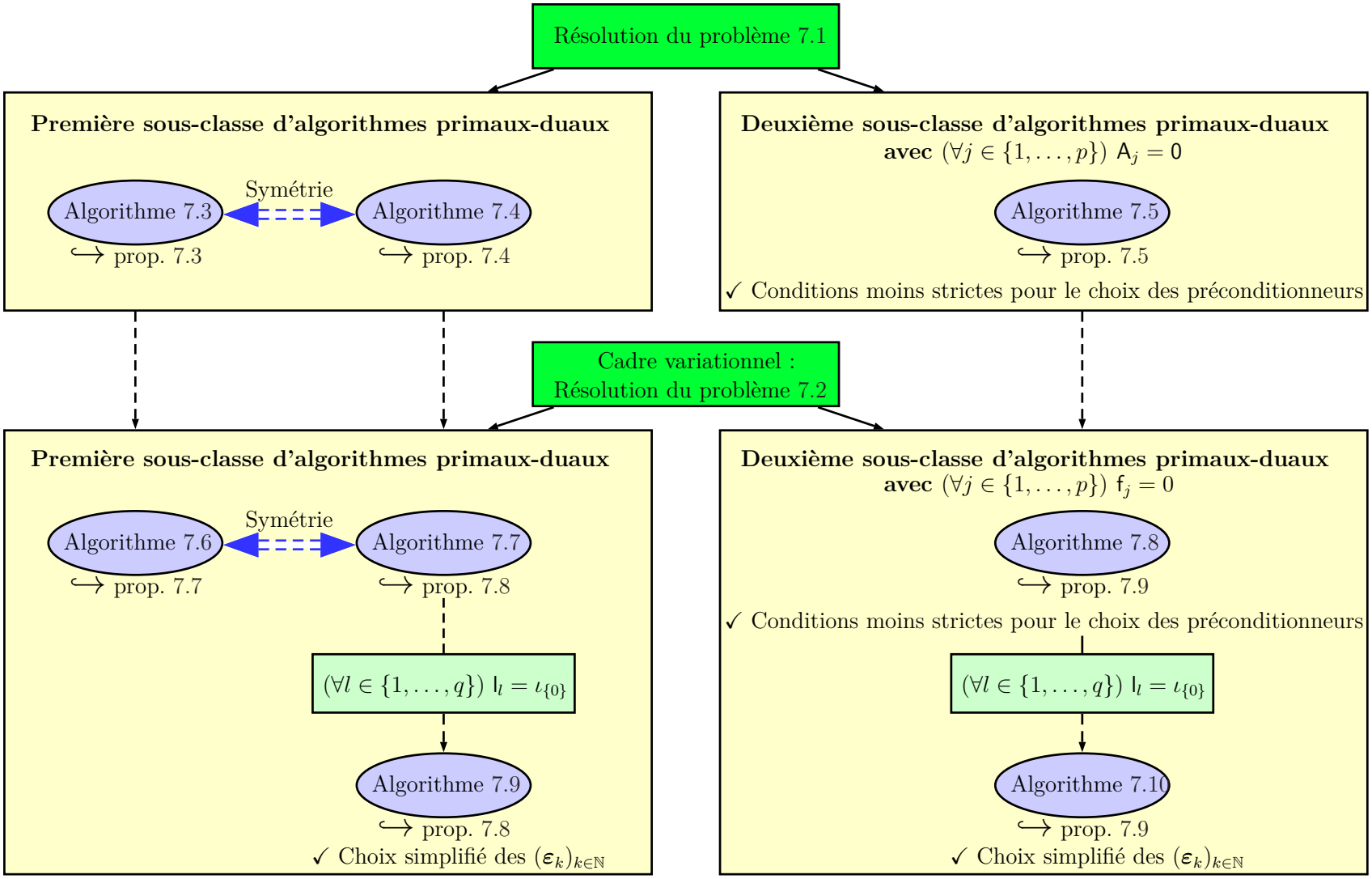


Figure 7.5 – *Diagramme résumant les algorithmes primaux-duaux alternés par bloc développés dans les sections 7.3 et 7.4 pour résoudre les problèmes 7.1 et 7.2.*

7.5 Algorithmes distribués

7.5.1 Problème et notations

Dans cette section, $\mathcal{H}$, $\mathcal{G}_1, \ldots, \mathcal{G}_m$ sont des espaces de Hilbert réels séparables, et on note $\boldsymbol{\mathcal{G}} = \mathcal{G}_1 \oplus \cdots \oplus \mathcal{G}_m$. On considère le problème suivant :

Problème 7.3. *Pour tout $i \in \{1, \ldots, m\}$, soient $\mathsf{A}_i \colon \mathcal{H} \to 2^{\mathcal{H}}$ un opérateur maximal monotone, $\mathsf{C}_i \colon \mathcal{H} \to \mathcal{H}$ un opérateur cocoercif, $\mathsf{B}_i \colon \mathcal{G}_i \to 2^{\mathcal{G}_i}$ un opérateur maximal monotone, $\mathsf{D}_i \colon \mathcal{G}_i \to 2^{\mathcal{G}_i}$ un opérateur maximal et fortement monotone, et M_i un opérateur non nul de $\mathcal{B}(\mathcal{H}, \mathcal{G}_i)$. On suppose que l'ensemble $\widehat{\mathsf{F}}$ de solutions du problème :*

$$\text{trouver } \mathsf{x} \in \mathcal{H} \text{ tel que } 0 \in \sum_{i=1}^{m} \mathsf{A}_i\mathsf{x} + \mathsf{C}_i\mathsf{x} + \mathsf{M}_i^*(\mathsf{B}_i \,\square\, \mathsf{D}_i)(\mathsf{M}_i\mathsf{x}) \tag{7.45}$$

est non vide. Notre objectif est de trouver une variable aléatoire $\widehat{x}$ à valeurs dans $\widehat{\mathsf{F}}$.

Afin d'obtenir des algorithmes parallèles servant à trouver le zéro d'une somme d'opérateurs maximaux monotones [Combettes et Pesquet, 2008; Pesquet et Pustelnik, 2012], ou bien pour traiter des problèmes de concensus [Boyd *et al.*, 2011; Nedić et Ozdaglar, 2010], le problème (7.45) peut être reformulé dans l'espace produit $\mathcal{H}^m$ comme suit

$$\text{trouver } (\mathsf{x}^{(1)}, \ldots, \mathsf{x}^{(m)}) \in \Lambda_m \text{ tel que } 0 \in \sum_{i=1}^{m} \mathsf{A}_i\mathsf{x}^{(i)} + \mathsf{C}_i\mathsf{x}^{(i)} + \mathsf{M}_i^*(\mathsf{B}_i \,\square\, \mathsf{D}_i)(\mathsf{M}_i\mathsf{x}^{(i)}) \tag{7.46}$$

où

$$\Lambda_m = \{(\mathsf{x}^{(1)}, \ldots, \mathsf{x}^{(m)}) \in \mathcal{H}^m \mid \mathsf{x}^{(1)} = \ldots = \mathsf{x}^{(m)}\}.$$

Pour construire des algorithmes distribués, la contrainte linéaire apparaissant dans le problème (7.46) est divisée en un ensemble de contraintes similaires, chacune d'entre elles faisant intervenir un sous ensemble réduit de variables. Chaque indice $i \in \{1, \ldots, m\}$ correspond alors à un agent distinct, et il s'agit de modéliser les relations topologiques existantes entre chaque agent. Pour ce faire, on définit des sous-ensembles non vides $(\mathbb{V}_\ell)_{1 \leqslant \ell \leqslant r}$ de $\{1, \ldots, m\}$, de cardinaux $(\kappa^{(\ell)})_{1 \leqslant \ell \leqslant r}$, vérifiant l'hypothèse suivante :

Hypothèse 7.1. *Pour tout $\mathbf{x} = (\mathsf{x}^{(i)})_{1 \leqslant i \leqslant m} \in \mathcal{H}^m$,*

$$\mathbf{x} \in \Lambda_m \quad \Leftrightarrow \quad (\forall \ell \in \{1, \ldots, r\}) \quad (\mathsf{x}^{(i)})_{i \in \mathbb{V}_\ell} \in \Lambda_{\kappa^{(\ell)}}.$$

Remarque 7.7. *Notons que l'hypothèse 7.1 est satisfaite dans les cas particuliers suivants*

- *si $r = 1$ et $\mathbb{V}_1 = \{1, \ldots, m\}$,*
- *si $r = m - 1$ et $(\forall \ell \in \{1, \ldots, m-1\})$ $\mathbb{V}_\ell = \{\ell, \ell + 1\}$.*

De manière plus générale, si les ensembles $(\mathbb{V}_\ell)_{1 \leqslant \ell \leqslant r}$ correspondent aux hyper-arêtes d'un hypergraphe de sommets $\{1, \ldots, m\}$, alors l'hypothèse 7.1 est équivalente à supposer que l'hypergraphe est connecté.

Pour la suite de cette section, nous avons besoin d'introduire les notations suivantes :

$$\boldsymbol{\mathcal{H}} = \mathcal{H}^{\kappa^{(1)}} \oplus \cdots \oplus \mathcal{H}^{\kappa^{(r)}}, \qquad \boldsymbol{\Lambda} = \Lambda_{\kappa^{(1)}} \oplus \cdots \oplus \Lambda_{\kappa^{(r)}},$$
$$\mathbf{A} = \bigtimes_{i=1}^{m} \mathsf{A}_i, \qquad \mathbf{C} = \bigtimes_{i=1}^{m} \mathsf{C}_i,$$
$$\mathbf{B} = \bigtimes_{i=1}^{m} \mathsf{B}_i, \qquad \mathbf{D} = \bigtimes_{i=1}^{m} \mathsf{D}_i,$$
$$\mathbf{S} \colon \mathcal{H}^m \to \boldsymbol{\mathcal{H}} \colon \mathbf{x} \mapsto (\mathsf{S}_\ell \mathbf{x})_{1 \leqslant \ell \leqslant r}, \qquad \mathbf{M} \colon \mathcal{H}^m \to \boldsymbol{\mathcal{G}} \colon \mathbf{x} \mapsto (\mathsf{M}_i \mathsf{x}^{(i)})_{1 \leqslant i \leqslant m},$$

où, pour tout $\ell \in \{1, \ldots, r\}$,

$$\mathsf{S}_\ell \colon \mathcal{H}^m \to \mathcal{H}^{\kappa^{(\ell)}} \colon \mathbf{x} \mapsto (\mathsf{x}^{(i)})_{i \in \mathbb{V}_\ell} = (\mathsf{x}^{(\mathsf{i}(\ell,j))})_{1 \leqslant j \leqslant \kappa^{(\ell)}} \tag{7.47}$$

et $\mathsf{i}(\ell, 1), \ldots, \mathsf{i}(\ell, \kappa_\ell)$ désignent les éléments de $\mathbb{V}_\ell$ ordonnés dans un ordre croissant. Notons que, pour tout $\ell \in \{1, \ldots, r\}$, l'opérateur adjoint de S_ℓ est donné par

$$\mathsf{S}_\ell^* \colon \mathcal{H}^{\kappa^{(\ell)}} \to \mathcal{H}^m \colon \mathsf{z}^{(\ell)} = (\mathsf{z}^{(\ell,j)})_{1 \leqslant j \leqslant \kappa^{(\ell)}} \mapsto (\mathsf{x}^{(i)})_{1 \leqslant i \leqslant m} \tag{7.48}$$

où

$$(\forall i \in \{1, \ldots, m\}) \qquad \mathsf{x}^{(i)} = \begin{cases} \mathsf{z}^{(\ell,j)} & \text{si } i = \mathsf{i}(\ell, j) \text{ avec } j \in \{1, \ldots, \kappa^{(\ell)}\} \\ 0 & \text{sinon.} \end{cases} \tag{7.49}$$

L'opérateur adjoint de $\mathbf{S}$ est alors donné par

$$\mathbf{S}^* \colon \boldsymbol{\mathcal{H}} \to \mathcal{H}^m \colon (\mathsf{z}^{(\ell)})_{1 \leqslant \ell \leqslant r} \mapsto \sum_{\ell=1}^{r} \mathsf{S}_\ell^* \mathsf{z}^{(\ell)} = (\mathsf{x}^{(i)})_{1 \leqslant i \leqslant m} \tag{7.50}$$

où, pour tout $i \in \{1, \ldots, m\}$,

$$\mathsf{x}^{(i)} = \sum_{(\ell,j) \in \mathbb{V}_i^*} \mathsf{z}^{(\ell,j)} \tag{7.51}$$

avec

$$\mathbb{V}_i^* = \big\{(\ell, j) \in \{1, \ldots, r\} \times \{1, \ldots, \kappa^{(\ell)}\} \;\big|\; \mathsf{i}(\ell, j) = i\big\}.$$

En conséquence de l'hypothèse 7.1, le cardinal de $\mathbb{V}_i^*$ (correspondant au nombre d'ensembles $(\mathbb{V}_\ell)_{1 \leqslant \ell \leqslant r}$ qui contiennent l'indice i) est non nul.

Un exemple d'hypergraphe connecté est donné dans la figure 7.6. Il est composé de $m = 6$ nœuds (ou agents) et $r = 4$ hyper-arêtes modélisant les relations topologiques entre ces agents.

Le lien entre les problèmes 7.3 et 7.1 est mis en évidence dans la proposition suivante :

Proposition 7.10. *Sous l'hypothèse 7.1, le problème* (7.46) *est équivalent à*

$$\textit{trouver } \mathbf{x} \in \mathcal{H}^m \textit{ tel que } \mathbf{0} \in \mathbf{A}\mathbf{x} + \mathbf{C}\mathbf{x} + \mathbf{M}^*(\mathbf{B} \,\square\, \mathbf{D})(\mathbf{M}\mathbf{x}) + \mathbf{S}^* \mathbf{N}_{\boldsymbol{\Lambda}}(\mathbf{S}\mathbf{x}). \tag{7.52}$$

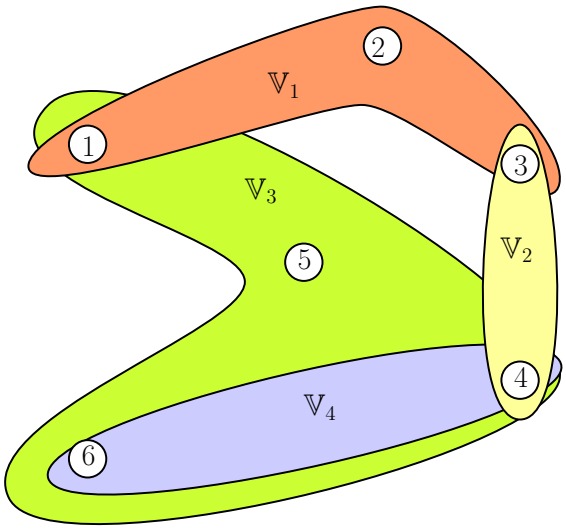


Figure 7.6 – *Exemple d'hypergraphe connecté avec $r = 4$ et $m = 6$. Ici, par exemple, le nœud 3 est à la fois dans les hyper-arêtes $\mathbb{V}_1 = \{\mathsf{i}(1,1), \mathsf{i}(1,2), \mathsf{i}(1,3)\}$ et $\mathbb{V}_2 = \{\mathsf{i}(2,1), \mathsf{i}(2,2)\}$. De plus, les voisins du nœud $3 = \mathsf{i}(1,3) = \mathsf{i}(2,1)$ sont les nœuds $1 = \mathsf{i}(1,1) = \mathsf{i}(3,1)$, $2 = \mathsf{i}(1,2)$ et $4 = \mathsf{i}(2,2) = \mathsf{i}(3,2) = \mathsf{i}(4,1)$, et on a $\mathbb{V}_3^* = \{(1,3), (2,1)\}$.*

Démonstration. Pour tout $\mathbf{x} \in \mathcal{H}^m$, on a les équivalences suivantes :

$$
\begin{aligned}
& \begin{cases} 0 \in \sum_{i=1}^m \mathsf{A}_i \mathsf{x}^{(i)} + \mathsf{C}_i \mathsf{x}^{(i)} + \mathsf{M}_i^* (\mathsf{B}_i \,\square\, \mathsf{D}_i)(\mathsf{M}_i \mathsf{x}^{(i)}) \\ \mathbf{x} \in \Lambda_m \end{cases} \\
\Leftrightarrow \quad & \begin{cases} (\forall i \in \{1, \dots, m\}) \quad 0 \in \mathsf{A}_i \mathsf{x}^{(i)} + \mathsf{C}_i \mathsf{x}^{(i)} + \mathsf{M}_i^* (\mathsf{B}_i \,\square\, \mathsf{D}_i)(\mathsf{M}_i \mathsf{x}^{(i)}) + \mathsf{u}^{(i)} \\ \mathbf{x} \in \Lambda_m \\ \sum_{i=1}^m \mathsf{u}^{(i)} = 0 \end{cases} \\
\Leftrightarrow \quad & \begin{cases} \mathbf{0} \in \mathbf{A}\mathbf{x} + \mathbf{C}\mathbf{x} + \mathbf{M}^* (\mathbf{B} \,\square\, \mathbf{D})(\mathbf{M}\mathbf{x}) + \mathbf{u} \\ \mathbf{x} \in \Lambda_m \\ \mathbf{u} \in \Lambda_m^\perp \end{cases} \\
\Leftrightarrow \quad & \mathbf{0} \in \mathbf{A}\mathbf{x} + \mathbf{C}\mathbf{x} + \mathbf{M}^* (\mathbf{B} \,\square\, \mathbf{D})(\mathbf{M}\mathbf{x}) + \mathbf{S}^* \mathbf{N}_{\mathbf{\Lambda}}(\mathbf{S}\mathbf{x}),
\end{aligned}
$$

où l'on a utilisé le fait que $\mathbf{N}_{\Lambda_m} = \partial \iota_{\Lambda_m} = \partial(\iota_{\mathbf{\Lambda}} \circ \mathbf{S}) = \mathbf{S}^* \partial \iota_{\mathbf{\Lambda}} \mathbf{S} = \mathbf{S}^* \mathbf{N}_{\mathbf{\Lambda}} \mathbf{S}$ puisque $\mathbf{\Lambda} + \operatorname{ran}(\mathbf{S})$ est un sous-espace fermé de $\boldsymbol{\mathcal{H}}$ [Bauschke et Combettes, 2011, Prop. 6.19 & 16.42]. □

Remarque 7.8. *Si nous reformulons (7.52) avec les notations du problème 7.1, nous pouvons voir que les problèmes 7.3 et 7.1 sont équivalent en posant $p = m$, $q = m+r$, $\mathcal{H}_1 = \ldots = \mathcal{H}_m = \mathcal{H}$,*

$(\forall \ell \in \{1,\ldots,r\})$ $\mathcal{G}_{m+\ell} = \mathcal{H}^{\kappa(\ell)}$, $\mathsf{B}_{m+\ell} = \mathsf{N}_{\Lambda_{\kappa(\ell)}}$, $\mathsf{D}_{m+\ell} = \mathsf{N}_{\{0\}}$, *et*

$$(\forall (k,i) \in \{1,\ldots,m\}^2) \qquad \mathsf{L}_{k,i} = \begin{cases} \mathsf{M}_i & \text{si } k = i \\ 0 & \text{sinon,} \end{cases}$$

$$(\forall \ell \in \{1,\ldots,r\})(\forall \mathbf{x} \in \mathcal{H}^m) \qquad \sum_{i=1}^{m} \mathsf{L}_{m+\ell,i}\, \mathsf{x}^{(i)} = \mathsf{S}_\ell \mathbf{x}$$

(ainsi, (7.8) *et* (7.9) *sont satisfaites).*

7.5.2 Algorithme distribués asynchrones pour les problèmes d'inclusion monotone

Notre objectif est maintenant de résoudre le problème 7.3. Pour cela, nous allons développer des algorithmes distribués asynchrones au sens où, à chaque itération, seul un nombre limité d'opérateurs $(\mathsf{A}_i)_{1\leqslant i\leqslant m}$, $(\mathsf{B}_i)_{1\leqslant i\leqslant m}$, $(\mathsf{C}_i)_{1\leqslant i\leqslant m}$,et $(\mathsf{D}_i)_{1\leqslant i\leqslant m}$ est activé, et ce, de façon aléatoire. Nous définissons donc l'algorithme 7.11.

La convergence faible presque sûre de l'algorithme 7.11 vers une solution du problème 7.3 est assurée par la proposition 7.11. Ce résultat est obtenu en se basant sur la remarque 7.8 et en utilisant l'algorithme 7.4 et la proposition 7.4.

Proposition 7.11. *Considérons l'algorithme 7.11. Supposons que, pour tout* $i \in \{1,\ldots,m\}$, $\mathsf{W}_i^{1/2}\mathsf{C}_i\mathsf{W}_i^{1/2}$ *est* μ_i*-cocoercif, avec* $\mu_i \in]0,+\infty[$, *et* $\mathsf{U}_i^{1/2}\mathsf{D}_i^{-1}\mathsf{U}_i^{1/2}$ *est* ν_i*-cocoercif, avec* $\nu_i \in]0,+\infty[$. *Posons*

$$\overline{\theta}_i = \sum_{\ell\in\{\ell'\in\{1,\ldots,r\}\,|\,i\in\mathbb{V}_{\ell'}\}} \theta^{(\ell)}. \tag{7.53}$$

Supposons que

$$(\exists \alpha \in]0,+\infty[) \qquad (1-\chi)\min\{\mu(1+\alpha\sqrt{\chi})^{-1}, \nu(1+\alpha^{-1}\sqrt{\chi})^{-1}\} > \frac{1}{2} \tag{7.54}$$

où

$$\chi = \max_{i\in\{1,\ldots,m\}} \|\mathsf{U}_i^{1/2}\mathsf{M}_i\mathsf{W}_i^{1/2}\|^2 + \overline{\theta}_i\|\mathsf{W}_i\|, \tag{7.55}$$

avec $\mu = \min\{\mu_1,\ldots,\mu_m\}$, *et* $\nu = \min\{\nu_1,\ldots,\nu_m\}$. *Supposons que les éléments de la suite* $(\lambda_k)_{k\in\mathbb{N}}$ *soient dans* $]0,1]$ *et vérifient* $\inf_{k\in\mathbb{N}}\lambda_k > 0$. *Posons* $(\forall k \in \mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ *et* $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant k'\leqslant k}$, *et supposons de plus que les assertions suivantes sont vérifiées :*

(i) $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{a}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$, $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$, $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{c}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$, *et* $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{d}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)} < +\infty$ P*-presque sûrement.*

(ii) *Pour tout* $k \in \mathbb{N}$, $\boldsymbol{\mathcal{E}}_k$ *et* $\boldsymbol{\mathcal{X}}_k$ *sont indépendants, et* $(\forall i \in \{1,\ldots,m\})$ $\mathsf{P}[\varepsilon_0^{(i)} = 1] > 0$.

(iii) *Pour tout* $k \in \mathbb{N}$,

$$(\forall i \in \{1,\ldots,m\}) \qquad \{\omega\in\Omega \mid \varepsilon_k^{(i)}(\omega) = 1\} \subset \{\omega\in\Omega \mid \varepsilon_k^{(m+i)}(\omega) = 1\} \tag{7.56}$$

Algorithme 7.11 Algorithme distribué asynchrone (version 1)

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\mathcal{H}^m$, et soient $(v_0^{(i)})_{1\leqslant i\leqslant m}$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$.
Pour tout $\ell\in\{1,\ldots,r\}$, soit $v_0^{(m+\ell)}=(v_0^{(m+\ell,j)})_{1\leqslant j\leqslant\kappa^{(\ell)}}$ une variable aléatoire à valeurs dans $\mathcal{H}^{\kappa^{(\ell)}}$, $\overline{x}_0^{(\ell)}=\frac{1}{\kappa^{(\ell)}}\sum_{i\in\mathbb{V}_\ell}x_0^{(i)}$, et $\theta^{(\ell)}\in\left]0,+\infty\right[$.
Soient, pour tout $i\in\{1,\ldots,m\}$, $\mathsf{W}_i\in\mathcal{S}^+(\mathcal{H})$ et $\mathsf{U}_i\in\mathcal{S}^+(\mathcal{G}_i)$.
Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{2m+r}$.
Soit, pour tout $k\in\mathbb{N}$, $\lambda_k\in\left]0,+\infty\right[$.

Itérations :

Pour $k=0,1,\ldots$

pour $\ell=1,\ldots,r$

$\overline{u}_k^{(\ell)}=\varepsilon_k^{(2m+\ell)}\Big(\dfrac{1}{\kappa^{(\ell)}}\displaystyle\sum_{j=1}^{\kappa^{(\ell)}}v_k^{(m+\ell,j)}+\theta^{(\ell)}\,\overline{x}_k^{(\ell)}\Big)$,

pour $j=1,\ldots,\kappa^{(\ell)}$

$w_k^{(\ell,j)}=\varepsilon_k^{(2m+\ell)}\big(2(\theta^{(\ell)}\,x_k^{(\mathrm{i}(\ell,j))}-\overline{u}_k^{(\ell)})+v_k^{(m+\ell,j)}\big)$,

pour $i=1,\ldots,m$

$u_k^{(i)}=\varepsilon_k^{(m+i)}\Big(\mathsf{J}_{\mathsf{U}_i\mathsf{B}_i^{-1}}\big(v_k^{(i)}+\mathsf{U}_i\big(\mathsf{M}_ix_k^{(i)}-\mathsf{D}_i^{-1}v_k^{(i)}+d_k^{(i)}\big)\big)+b_k^{(i)}\Big)$,

$y_k^{(i)}=\varepsilon_k^{(i)}\Big(\mathsf{J}_{\mathsf{W}_i\mathsf{A}_i}\big(x_k^{(i)}-\mathsf{W}_i(\mathsf{M}_i^*(2u_k^{(i)}-v_k^{(i)})+\displaystyle\sum_{(\ell,j)\in\mathbb{V}_i^*}w_k^{(\ell,j)}+\mathsf{C}_ix_k^{(i)}+c_k^{(i)})\big)+a_k^{(i)}\Big)$,

$v_{k+1}^{(i)}=v_k^{(i)}+\lambda_k\varepsilon_k^{(m+i)}(u_k^{(i)}-v_k^{(i)})$,

$x_{k+1}^{(i)}=x_k^{(i)}+\lambda_k\varepsilon_k^{(i)}\,(y_k^{(i)}-x_k^{(i)})$,

pour $\ell=1,\ldots,r$

$v_{k+1}^{(m+\ell)}=v_k^{(m+\ell)}+\dfrac{\lambda_k}{2}\varepsilon_k^{(2m+\ell)}(w_k^{(\ell)}-v_k^{(m+\ell)})$,

$\eta_k^{(\ell)}=\max\big\{\varepsilon_k^{(i)}\;\big|\;i\in\mathbb{V}_\ell\big\}$,

$\overline{x}_{k+1}^{(\ell)}=\overline{x}_k^{(\ell)}+\eta_k^{(\ell)}\Big(\dfrac{1}{\kappa^{(\ell)}}\displaystyle\sum_{i\in\mathbb{V}_\ell}x_{k+1}^{(i)}-\overline{x}_k^{(\ell)}\Big)$.

et

$$(\forall\ell\in\{1,\ldots,r\})\quad\bigcup_{i\in\mathbb{V}_\ell}\{\omega\in\Omega\mid\varepsilon_k^{(i)}(\omega)=1\}\subset\{\omega\in\Omega\mid\varepsilon_k^{(2m+\ell)}(\omega)=1\}.\tag{7.57}$$

Sous l'hypothèse 7.1, pour tout $i\in\{1,\ldots,m\}$, $(x_k^{(i)})_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire $\widehat{x}$ à valeurs dans $\widehat{\mathsf{F}}$ et, pour tout $\ell\in\{1,\ldots,r\}$, $(\overline{x}_k^{(\ell)})_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers $\widehat{x}$.

La démonstration de la proposition 7.11 est donnée dans l'annexe 7.E.

Remarque 7.9.

(i) *La k-ème itération de l'algorithme* (7.11) *est composé de deux types d'opérations : d'une part, la mise à jour de certaines des variables $(x_k^{(i)})_{1\leqslant i\leqslant m}$ et $(v_k^{(i)})_{1\leqslant i\leqslant m}$ en utilisant les opérateurs $(\mathsf{J}_{\mathsf{W}_i\mathsf{A}_i})_{1\leqslant i\leqslant m}$, $(\mathsf{J}_{\mathsf{U}_i\mathsf{B}_i^{-1}})_{1\leqslant i\leqslant m}$, $(\mathsf{C}_i)_{1\leqslant i\leqslant m}$, et $(\mathsf{D}_i^{-1})_{1\leqslant i\leqslant m}$, d'autre part, des étapes de fusion effectuées sur les ensembles $(\mathbb{V}_\ell)_{1\leqslant\ell\leqslant r}$. Dans ce contexte, un choix simple pour que les variables aléatoires booléennes $(\varepsilon_k^{(l)})_{m+1\leqslant l\leqslant 2m+r}$ satisfassent la condition (iii) est de prendre, pour tout $k\in\mathbb{N}$,*

$$(\forall i\in\{1,\dots,m\})\quad \varepsilon_k^{(m+i)}=\varepsilon_k^{(i)}, \tag{7.58}$$

$$(\forall \ell\in\{1,\dots,r\})\quad \varepsilon_k^{(2m+\ell)}=\eta_k^{(\ell)}=\max\{\varepsilon_k^{(i)}\mid i\in\mathbb{V}_\ell\}. \tag{7.59}$$

(ii) *D'après* (7.92)*, nous pouvons remarquer que, pour tout $k\in\mathbb{N}$ et $\ell\in\{1,\dots,r\}$, $\Pi_{\Lambda_{\kappa(\ell)}}u_k^{(m+\ell)}=0$, ce qui implique que la relation de récursivité suivante est satisfaite :*

$$\sum_{j=1}^{\kappa^{(\ell)}}v_{k+1}^{(m+\ell,j)}=(1-\lambda_k\varepsilon_k^{(2m+\ell)})\sum_{j=1}^{\kappa^{(\ell)}}v_k^{(m+\ell,j)}. \tag{7.60}$$

En particulier, si l'initialisation $(v_0^{(m+\ell)})_{1\leqslant\ell\leqslant r}$ est choisie de façon à satisfaire la condition suivante :

$$(\forall\ell\in\{1,\dots,r\})\qquad \sum_{j=1}^{\kappa^{(\ell)}}v_0^{(m+\ell,j)}=0, \tag{7.61}$$

alors l'algorithme 7.11 peut se simplifier et on obtient l'algorithme 7.12.

(iii) *De la même manière que pour la remarque 7.3(iii), une condition suffisante pour que* (7.54) *soit satisfaite est obtenue en posant $\alpha=1$:*

$$(1-\sqrt{\chi})\min\{\mu,\nu\}>\frac{1}{2}. \tag{7.62}$$

(iv) *Lorsque, pour tout $i\in\{1,\dots,m\}$, $\mathsf{D}_i^{-1}=\mathsf{0}$, une condition moins contraignante est de prendre*

$$(1-\chi)\mu>\frac{1}{2}. \tag{7.63}$$

Algorithme 7.12 Algorithme distribué asynchrone (version 1 simplifiée)

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\mathcal{H}^m$, et soient $(v_0^{(i)})_{1\leqslant i\leqslant m}$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$.
Pour tout $\ell\in\{1,\dots,r\}$, soit $v_0^{(m+\ell)}=(v_0^{(m+\ell,j)})_{1\leqslant j\leqslant\kappa^{(\ell)}}$ une variable aléatoire à valeurs dans $\mathcal{H}^{\kappa^{(\ell)}}$ vérifiant $\sum_{j=1}^{\kappa^{(\ell)}} v_0^{(m+\ell,j)}=0$, soit $\overline{x}_0^{(\ell)}=\frac{1}{\kappa^{(\ell)}}\sum_{i\in\mathbb{V}_\ell} x_0^{(i)}$, et soit $\theta^{(\ell)}\in\left]0,+\infty\right[$.
Soient, pour tout $i\in\{1,\dots,m\}$, $\mathsf{W}_i\in\mathcal{S}^+(\mathcal{H})$ et $\mathsf{U}_i\in\mathcal{S}^+(\mathcal{G}_i)$.
Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{2m+r}$.
Soit, pour tout $k\in\mathbb{N}$, $\lambda_k\in\left]0,+\infty\right[$.

Itérations :

Pour $k=0,1,\dots$

pour $\ell=1,\dots,r$

pour $j=1,\dots,\kappa^{(\ell)}$

$w_k^{(\ell,j)}=\varepsilon_k^{(2m+\ell)}\Big(2\theta^{(\ell)}\,(x_k^{(\mathrm{i}(\ell,j))}-\overline{x}_k^{(\ell)})+v_k^{(m+\ell,j)}\Big)$,

pour $i=1,\dots,m$

$u_k^{(i)}=\varepsilon_k^{(m+i)}\Big(\mathsf{J}_{\mathsf{U}_i\mathsf{B}_i^{-1}}\Big(v_k^{(i)}+\mathsf{U}_i\Big(\mathsf{M}_i x_k^{(i)}-\mathsf{D}_i^{-1}v_k^{(i)}+d_k^{(i)}\Big)\Big)+b_k^{(i)}\Big)$,

$y_k^{(i)}=\varepsilon_k^{(i)}\Big(\mathsf{J}_{\mathsf{W}_i\mathsf{A}_i}\Big(x_k^{(i)}-\mathsf{W}_i\big(\mathsf{M}_i^*(2u_k^{(i)}-v_k^{(i)})+\sum_{(\ell,j)\in\mathbb{V}_i^*} w_k^{(\ell,j)}+\mathsf{C}_i x_k^{(i)}+c_k^{(i)}\big)\Big)+a_k^{(i)}\Big)$,

$v_{k+1}^{(i)}=v_k^{(i)}+\lambda_k\varepsilon_k^{(m+i)}(u_k^{(i)}-v_k^{(i)})$,

$x_{k+1}^{(i)}=x_k^{(i)}+\lambda_k\varepsilon_k^{(i)}\,(y_k^{(i)}-x_k^{(i)})$,

pour $\ell=1,\dots,r$

$v_{k+1}^{(m+\ell)}=v_k^{(m+\ell)}+\dfrac{\lambda_k}{2}\varepsilon_k^{(2m+\ell)}(w_k^{(\ell)}-v_k^{(m+\ell)})$,

$\eta_k^{(\ell)}=\max\Big\{\varepsilon_k^{(i)}\;\Big|\;i\in\mathbb{V}_\ell\Big\}$,

$\overline{x}_{k+1}^{(\ell)}=\overline{x}_k^{(\ell)}+\eta_k^{(\ell)}\Big(\dfrac{1}{\kappa^{(\ell)}}\sum_{i\in\mathbb{V}_\ell} x_{k+1}^{(i)}-\overline{x}_k^{(\ell)}\Big)$.

De plus, si $(\forall i\in\{1,\dots,m\})$ $\mathsf{B}_i=\mathsf{0}$, *les* $(\|\mathsf{M}_i\|)_{1\leqslant i\leqslant m}$ *peuvent être choisis aussi petits que l'on veut, et ainsi on peut poser*

$$\chi=\max_{i\in\{1,\dots,m\}}\overline{\theta}_i\|\mathsf{W}_i\|. \tag{7.64}$$

Dans ce cas, l'algorithme 7.12 peut être simplifié, en remarquant que, pour tout $k\in\mathbb{N}$, *les variables* $(u_k^{(i)})_{1\leqslant i\leqslant m}$ *et* $(v_k^{(i)})_{1\leqslant i\leqslant m}$ *ne sont plus utiles. En imposant les conditions* (7.59) *et* (7.61), *et en posant*

$$(\forall n\in\mathbb{N})(\forall\ell\in\{1,\dots,r\})\quad \widetilde{v}_k^{(\ell)}=v_k^{(m+\ell)}, \tag{7.65}$$

on obtient l'algorithme 7.13

Algorithme 7.13 Algorithme distribué asynchrone (version 1 simplifiée dans le cas où ($\forall i \in \{1,\dots,m\}$) $\mathsf{D}_i^{-1} = \mathsf{0}$ et $\mathsf{B}_i = \mathsf{0}$)

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\mathcal{H}^m$. Pour tout $\ell \in \{1,\dots,r\}$, soit $\widetilde{v}_0^{(\ell)} = (\widetilde{v}_0^{(\ell,j)})_{1\leqslant j\leqslant \kappa^{(\ell)}}$ une variable aléatoire à valeurs dans $\mathcal{H}^{\kappa^{(\ell)}}$ vérifiant $\sum_{j=1}^{\kappa^{(\ell)}} \widetilde{v}_0^{(\ell,j)} = 0$, soit $\overline{x}_0^{(\ell)} = \frac{1}{\kappa^{(\ell)}} \sum_{i\in\mathbb{V}_\ell} x_0^{(i)}$, et soit $\theta^{(\ell)} \in]0,+\infty[$.
Soient, pour tout $i \in \{1,\dots,m\}$, $\mathsf{W}_i \in \mathcal{S}^+(\mathcal{H})$.
Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_m$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :
Pour $k = 0, 1, \dots$

pour $\ell = 1,\dots,r$
- $\eta_k^{(\ell)} = \max\left\{\varepsilon_k^{(i)} \;\middle|\; i \in \mathbb{V}_\ell\right\}$
- pour $j = 1,\dots,\kappa^{(\ell)}$
  - $w_k^{(\ell,j)} = \eta_k^{(\ell)}\big(2\theta^{(\ell)}(x_k^{(\mathrm{i}(\ell,j))} - \overline{x}_k^{(\ell)}) + \widetilde{v}_k^{(\ell,j)}\big)$

pour $i = 1,\dots,m$
- $y_k^{(i)} = \varepsilon_k^{(i)}\Big(\mathsf{J}_{\mathsf{W}_i\mathsf{A}_i}\big(x_k^{(i)} - \mathsf{W}_i(\sum_{(\ell,j)\in\mathbb{V}_i^*} w_k^{(\ell,j)} + \mathsf{C}_i x_k^{(i)} + c_k^{(i)})\big) + a_k^{(i)}\Big)$
- $x_{k+1}^{(i)} = x_k^{(i)} + \lambda_k\varepsilon_k^{(i)}\,(y_k^{(i)} - x_k^{(i)})$

pour $\ell = 1,\dots,r$
- $\widetilde{v}_{k+1}^{(\ell)} = \widetilde{v}_k^{(\ell)} + \dfrac{\lambda_k}{2}\eta_k^{(\ell)}(w_k^{(\ell)} - \widetilde{v}_k^{(\ell)})$
- $\overline{x}_{k+1}^{(\ell)} = \overline{x}_k^{(\ell)} + \eta_k^{(\ell)}\Big(\dfrac{1}{\kappa^{(\ell)}}\displaystyle\sum_{i\in\mathbb{V}_\ell} x_{k+1}^{(i)} - \overline{x}_k^{(\ell)}\Big)$.

Nous pouvons déduire de la section 7.3.3 un autre type d'algorithme distribué, lorsque, pour tout $i \in \{1,\dots,m\}$, $\mathsf{A}_i = \mathsf{0}$. En se basant sur l'algorithme 7.5, nous obtenons alors l'algorithme 7.14, dont la convergence vers une solution du problème 7.3, lorsque $\mathsf{A}_i = \mathsf{0}$ pour tout $i \in \{1,\dots,m\}$, est assurée par la proposition 7.12. La démonstration de ce résultat est basée sur la proposition 7.5.

Proposition 7.12. *Considérons l'algorithme 7.14. Supposons que, pour tout $i \in \{1,\dots,m\}$, $\mathsf{W}_i^{1/2}\mathsf{C}_i\mathsf{W}_i^{1/2}$ est μ_i-cocoercif, avec $\mu_i \in]0,+\infty[$, et $\mathsf{U}_i^{1/2}\mathsf{D}_i^{-1}\mathsf{U}_i^{1/2}$ est ν_i-cocoercif, avec $\nu_i \in]0,+\infty[$. Soient μ, ν, et χ définis par la proposition 7.11, et supposons que*

$$\min\{\mu, \nu(1-\chi)\} > \frac{1}{2}. \tag{7.66}$$

Supposons que les éléments de la suite $(\lambda_k)_{k\in\mathbb{N}}$ sont dans $]0,1]$ et vérifient $\inf_{k\in\mathbb{N}} \lambda_k > 0$. De plus, posons ($\forall k \in \mathbb{N}$) $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_n)$ and $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant n'\leqslant n}$ et supposons que

Algorithme 7.14 Algorithme distribué asynchrone (version 2 simplifiée)

Initialisation : Soient $\boldsymbol{x}_0$, et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\mathcal{H}^m$, et soient $(v_0^{(i)})_{1\leqslant i\leqslant m}$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$.
Pour tout $\ell\in\{1,\ldots,r\}$, soit $v_0^{(m+\ell)}=(v_0^{(m+\ell,j)})_{1\leqslant j\leqslant\kappa^{(\ell)}}$ une variable aléatoire à valeurs dans $\mathcal{H}^{\kappa^{(\ell)}}$ vérifiant $\sum_{j=1}^{\kappa^{(\ell)}} v_0^{(m+\ell,j)}=0$, et soit $\theta^{(\ell)}\in\left]0,+\infty\right[$.
Soient, pour tout $i\in\{1,\ldots,m\}$, $\mathsf{W}_i\in\mathcal{S}^+(\mathcal{H})$ et $\mathsf{U}_i\in\mathcal{S}^+(\mathcal{G}_i)$.
Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{2m+r}$.
Soit, pour tout $k\in\mathbb{N}$, $\lambda_k\in\left]0,+\infty\right[$.
Itérations :
Pour $k=0,1,\ldots$
pour $i=1,\ldots,m$
$\eta_k^{(i)}=\max\Big\{\varepsilon_k^{(m+i)},(\varepsilon_k^{(2m+\ell)})_{\ell\in\{\ell'\in\{1,\ldots,r\}\,|\,i\in\mathbb{V}_{\ell'}\}}\Big\}$
$w_k^{(i)}=\eta_k^{(i)}\Big(x_k^{(i)}-\mathsf{W}_i(\mathsf{C}_i x_k^{(i)}+c_k^{(i)})\Big)$
$\widetilde{w}_k^{(i)}=\eta_k^{(i)}\Big(w_k^{(i)}-\mathsf{W}_i(\mathsf{M}_i^* v_k^{(i)}+\sum_{(\ell,j)\in\mathbb{V}_i^*} v_k^{(m+\ell,j)})\Big)$
$u_k^{(i)}=\varepsilon_k^{(m+i)}\Big(\mathsf{J}_{\mathsf{U}_i\mathsf{B}_i^{-1}}\Big(v_k^{(i)}+\mathsf{U}_i(\mathsf{M}_i\widetilde{w}_k^{(i)}-\mathsf{D}_i^{-1}v_k^{(i)}+d_k^{(i)})\Big)+b_k^{(i)}\Big)$
$v_{k+1}^{(i)}=v_k^{(i)}+\lambda_k\varepsilon_k^{(m+i)}(u_k^{(i)}-v_k^{(i)})$
pour $\ell=1,\ldots,r$
$\overline{w}_k^{(\ell)}=\varepsilon_k^{(2m+\ell)}\,\dfrac{\theta^{(\ell)}}{\kappa^{(\ell)}}\sum_{i\in\mathbb{V}_\ell}\widetilde{w}_k^{(i)}$
pour $j=1,\ldots,\kappa^{(\ell)}$
$u_k^{(m+\ell,j)}=\varepsilon_k^{(2m+\ell)}\Big(v_k^{(m+\ell,j)}+\theta^{(\ell)}\,\widetilde{w}_k^{(\mathrm{i}(j,\ell))}-\overline{w}_k^{(\ell)}\Big)$
$v_{k+1}^{(m+\ell)}=v_k^{(m+\ell)}+\lambda_k\varepsilon_k^{(2m+\ell)}(u_k^{(m+\ell)}-v_k^{(m+\ell)})$
pour $i=1,\ldots,m$
$x_{k+1}^{(i)}=x_k^{(i)}+\lambda_k\varepsilon_k^{(i)}\Big(w_k^{(i)}-\mathsf{W}_i\big(\mathsf{M}_i^* u_k^{(i)}+\sum_{(\ell,j)\in\mathbb{V}_i^*} u_k^{(m+\ell,j)}\big)-x_k^{(i)}\Big)$.

(i) $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)}<+\infty$, $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{c}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)}<+\infty$, *et* $\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{d}_k\|^2\,|\,\boldsymbol{\mathcal{X}}_k)}<+\infty$ P-*presque sûrement*

et que les conditions (ii)-(iii) de la proposition 7.11 sont satisfaites.
Sous l'hypothèse 7.1, si $(\forall i\in\{1,\ldots,m\})$ $\mathsf{A}_i=0$ *dans le problème 7.3, alors, pour tout* $i\in\{1,\ldots,m\}$, $(x_k^{(i)})_{k\in\mathbb{N}}$ *converge faiblement* P*-presque sûrement vers une variable aléatoire* $\widehat{x}$ *à valeurs dans* $\widehat{\mathsf{F}}$.

La démonstration de la proposition 7.12 est donnée dans l'annexe 7.F.

Remarque 7.10. *Lorsque* $(\forall i\in\{1,\ldots,m\})$ $\mathsf{D}_i^{-1}=0$, *la condition* (7.66) *peut être réécrite de la*

façon suivante

$$(\forall i \in \{1,\dots,m\}) \quad \|\mathsf{U}_i^{1/2}\mathsf{M}_i\mathsf{W}_i^{1/2}\|^2 + \overline{\theta}_i\|\mathsf{W}_i\| < 1 \quad \textit{et} \quad \mu_i > 1/2. \tag{7.67}$$

7.5.3 Application aux problèmes variationnels

Nous proposons maintenant d'appliquer les résultats présentés dans la section 7.5.2 à la résolution de problèmes variationnels pouvant s'exprimer de la façon suivante :

Problème 7.4. *Pour tout $i \in \{1,\dots,m\}$, soient $\mathsf{f}_i \in \Gamma_0(\mathcal{H})$, $\mathsf{h}_i \in \Gamma_0(\mathcal{H})$ une fonction différentiable de gradient Lipschitz, $\mathsf{g}_i \in \Gamma_0(\mathcal{G}_i)$, $\mathsf{l}_i \in \Gamma_0(\mathcal{G}_i)$ une fonction fortement convexe, et M_i un opérateur non nul de $\mathcal{B}(\mathcal{H},\mathcal{G}_i)$. Supposons qu'il existe $\overline{\mathsf{x}} \in \mathcal{H}$ vérifiant*

$$0 \in \sum_{i=1}^{m} \partial\mathsf{f}_i(\overline{\mathsf{x}}) + \nabla\mathsf{h}_i(\overline{\mathsf{x}}) + \mathsf{M}_i^*(\partial\mathsf{g}_i \,\square\, \partial\mathsf{l}_i)(\mathsf{M}_i\overline{\mathsf{x}}). \tag{7.68}$$

Soit $\check{\mathsf{F}}$ l'ensemble défini par

$$\check{\mathsf{F}} = \operatorname*{Argmin}_{\mathsf{x}\in\mathcal{H}} \sum_{i=1}^{m} \mathsf{f}_i(\mathsf{x}) + \mathsf{h}_i(\mathsf{x}) + (\mathsf{g}_i \,\square\, \mathsf{l}_i)(\mathsf{M}_i\mathsf{x}). \tag{7.69}$$

L'objectif est de trouver une variable aléatoire $\widehat{x}$ à valeurs dans $\check{\mathsf{F}}$.

Une méthode proximale, déduite de l'algorithme 7.12, permettant de résoudre le problème 7.4 est donnée dans l'algorithme 7.15. Ses résultats de convergence, donnés dans la proposition 7.13, découlent de la proposition 7.11.

Proposition 7.13. *Considérons l'algorithme 7.15. Supposons que, pour tout $i \in \{1,\dots,m\}$, le gradient $\mathsf{h}_i \circ \mathsf{W}_i^{1/2}$ est μ_i^{-1}-Lipschitz, avec $\mu_i \in \left]0,+\infty\right[$, et $\mathsf{l}_i^* \circ \mathsf{U}_i^{1/2}$ est ν_i^{-1}-Lipschitz, avec $\nu_i \in \left]0,+\infty\right[$. Soient $\mu = \min\{\mu_1,\dots,\mu_m\}$, $\nu = \min\{\nu_1,\dots,\nu_m\}$, soit χ défini par la proposition 7.11, et supposons que*(7.54) *est vérifiée. Supposons que les éléments de la suite $(\lambda_k)_{k\in\mathbb{N}}$ sont dans $]0,1]$ et vérifient $\inf_{k\in\mathbb{N}} \lambda_k > 0$. Posons, $(\forall k \in \mathbb{N})$ $\boldsymbol{\mathcal{E}}_k = \sigma(\boldsymbol{\varepsilon}_k)$ et $\boldsymbol{\mathcal{X}}_k = \sigma(\boldsymbol{x}_{k'}, \boldsymbol{v}_{k'})_{0\leqslant k'\leqslant k}$ et supposons que les conditions (i)-(iii) de la proposition 7.11 sont vérifiées.*
Sous l'hypothèse 7.1, pour tout $i \in \{1,\dots,m\}$, $(x_k^{(i)})_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire $\widehat{x}$ à valeurs dans $\check{\mathsf{F}}$ et, pour tout $\ell \in \{1,\dots,r\}$, $(\overline{x}_k^{(\ell)})_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers $\widehat{x}$.

Démonstration. Dans la proposition 7.11, pour tout $i \in \{1,\dots,m\}$, posons $\mathsf{A}_i = \partial\mathsf{f}_i$, $\mathsf{B}_i = \partial\mathsf{g}_i$, $\mathsf{C}_i = \nabla\mathsf{h}_i$, et $\mathsf{D}_i^{-1} = \nabla\mathsf{l}_i^*$. D'après (7.68) et [Bauschke et Combettes, 2011, Prop. 16.5], on a

$$0 \in \sum_{i=1}^{m} \mathsf{A}_i\overline{\mathsf{x}} + \mathsf{C}_i\overline{\mathsf{x}} + \mathsf{M}_i^*(\mathsf{B}_i \,\square\, \mathsf{D}_i)(\mathsf{M}_i\overline{\mathsf{x}}) \subset \partial\Big(\sum_{i=1}^{m} \mathsf{f}_i + \mathsf{h}_i + (\mathsf{g}_i \,\square\, \mathsf{l}_i) \circ \mathsf{M}_i\Big)(\overline{\mathsf{x}}), \tag{7.70}$$

ce qui montre que $\varnothing \neq \widehat{\mathsf{F}} \subset \check{\mathsf{F}}$. Nous pouvons donc conclure en appliquant la proposition 7.11 et en utilisant la remarque 7.9(ii). □

Algorithme 7.15 Algorithme distribué asynchrone (version 1 simplifiée) appliqué dans un cadre variationnel

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\mathcal{H}^m$, et soient $(v_0^{(i)})_{1\leqslant i\leqslant m}$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$.
Pour tout $\ell\in\{1,\dots,r\}$, soit $v_0^{(m+\ell)}=(v_0^{(m+\ell,j)})_{1\leqslant j\leqslant\kappa^{(\ell)}}$ une variable aléatoire à valeurs dans $\mathcal{H}^{\kappa^{(\ell)}}$ vérifiant $\sum_{j=1}^{\kappa^{(\ell)}} v_0^{(m+\ell,j)}=0$, soit $\overline{x}_0^{(\ell)}=\frac{1}{\kappa^{(\ell)}}\sum_{i\in\mathbb{V}_\ell}x_0^{(i)}$, et soit $\theta^{(\ell)}\in\left]0,+\infty\right[$.
Soient, pour tout $i\in\{1,\dots,m\}$, $\mathsf{W}_i\in\mathcal{S}^+(\mathcal{H})$ et $\mathsf{U}_i\in\mathcal{S}^+(\mathcal{G}_i)$.
Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{2m+r}$.
Soit, pour tout $k\in\mathbb{N}$, $\lambda_k\in\left]0,+\infty\right[$.

Itérations :

Pour $k=0,1,\dots$

pour $\ell=1,\dots,r$

pour $j=1,\dots,\kappa^{(\ell)}$

$w_k^{(\ell,j)}=\varepsilon_k^{(2m+\ell)}\Big(2\theta^{(\ell)}\,(x_k^{(\mathrm{i}(\ell,j))}-\overline{x}_k^{(\ell)})+v_k^{(m+\ell,j)}\Big)$,

pour $i=1,\dots,m$

$u_k^{(i)}=\varepsilon_k^{(m+i)}\Big(\operatorname{prox}_{\mathsf{U}_i^{-1},\mathsf{g}_i^*}\Big(v_k^{(i)}+\mathsf{U}_i\Big(\mathsf{M}_i x_k^{(i)}-\nabla\mathsf{l}_i^*(v_k^{(i)})+d_k^{(i)}\Big)\Big)+b_k^{(i)}\Big)$,

$y_k^{(i)}=\varepsilon_k^{(i)}\Big(\operatorname{prox}_{\mathsf{W}_i^{-1},\mathsf{f}_i}\Big(x_k^{(i)}-\mathsf{W}_i(\mathsf{M}_i^*(2u_k^{(i)}-v_k^{(i)})+\sum_{(\ell,j)\in\mathbb{V}_i^*}w_k^{(\ell,j)}$
$+\nabla\mathsf{h}_i(x_k^{(i)})+c_k^{(i)})\Big)+a_k^{(i)}\Big)$,

$v_{k+1}^{(i)}=v_k^{(i)}+\lambda_k\varepsilon_k^{(m+i)}(u_k^{(i)}-v_k^{(i)})$,

$x_{k+1}^{(i)}=x_k^{(i)}+\lambda_k\varepsilon_k^{(i)}\,(y_k^{(i)}-x_k^{(i)})$,

pour $\ell=1,\dots,r$

$v_{k+1}^{(m+\ell)}=v_k^{(m+\ell)}+\dfrac{\lambda_k}{2}\varepsilon_k^{(2m+\ell)}(w_k^{(\ell)}-v_k^{(m+\ell)})$,

$\eta_k^{(\ell)}=\max\Big\{\varepsilon_k^{(i)}\;\Big|\;i\in\mathbb{V}_\ell\Big\}$,

$\overline{x}_{k+1}^{(\ell)}=\overline{x}_k^{(\ell)}+\eta_k^{(\ell)}\Big(\dfrac{1}{\kappa^{(\ell)}}\sum_{i\in\mathbb{V}_\ell}x_{k+1}^{(i)}-\overline{x}_k^{(\ell)}\Big)$.

Remarque 7.11.

(i) *De façon similaire, un second algorithme distribué d'optimisation convexe peut être déduit de la proposition 7.12.*

(ii) *Dans le cas où, pour tout $\ell\in\{1,\dots,r\}$, $\kappa^{(\ell)}=2$, les variables $(\overline{x}_k^{(\ell)})_{1\leqslant\ell\leqslant r}$ sont mises à jour à chaque itération $k\in\mathbb{N}$, i.e.*

$$(\forall k\in\mathbb{N})(\forall\ell\in\{1,\dots,r\})\quad \overline{x}_k^{(\ell)}=\frac{1}{2}(x_k^{(\mathrm{i}(\ell,1))}+x_k^{(\mathrm{i}(\ell,2))}),$$

et (7.58)-(7.59) *sont satisfaites, une version simplifiée de l'algorithme 7.15 est donnée par l'algorithme 7.16.*

Algorithme 7.16 Algorithme distribué (version 1 simplifiée) sur un graphe non orienté

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\mathcal{H}^m$, et soient $(v_0^{(i)})_{1\leqslant i\leqslant m}$, $(\boldsymbol{b}_k)_{k\in\mathbb{N}}$, et $(\boldsymbol{d}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\boldsymbol{\mathcal{G}}$.
Pour tout $\ell \in \{1,\dots,r\}$, soient $\theta^{(\ell)} \in]0,+\infty[$, $v_0^{(m+\ell,1)}$ une variable aléatoire à valeurs dans $\mathcal{H}$ et $v_0^{(m+\ell,2)} = -v_0^{(m+\ell,1)}$.
Soient, pour tout $i \in \{1,\dots,m\}$, $\mathsf{W}_i \in \mathcal{S}^+(\mathcal{H})$, $\mathsf{U}_i \in \mathcal{S}^+(\mathcal{G}_i)$, et $\overline{\theta}_i$ défini par (7.53).
Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_{2m+r}$.
Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.

Itérations :

Pour $k = 0, 1, \dots$

pour $\ell = 1,\dots,r$

$$\eta_k^{(\ell)} = \max\Big\{\varepsilon_k^{(i)} \;\Big|\; i \in \mathbb{V}_\ell\Big\},$$
$$v_{k+1}^{(\ell,1)} = v_k^{(m+\ell,1)} + \frac{\lambda_k}{2}\eta_k^{(\ell)}\theta^{(\ell)}\,(x_k^{(\mathrm{i}(\ell,1))} - x_k^{(\mathrm{i}(\ell,2))}),$$
$$v_{k+1}^{(m+\ell,2)} = -v_{k+1}^{(m+\ell,1)},$$

pour $i = 1,\dots,m$

$$u_k^{(i)} = \varepsilon_k^{(i)}\Big(\operatorname{prox}_{\mathsf{U}_i^{-1},\mathsf{g}_i^*}\Big(v_k^{(i)} + \mathsf{U}_i\Big(\mathsf{M}_i x_k^{(i)} - \nabla \mathsf{l}_i^*(v_k^{(i)}) + d_k^{(i)}\Big)\Big) + b_k^{(i)}\Big),$$
$$y_k^{(i)} = \varepsilon_k^{(i)}\Big(\operatorname{prox}_{\mathsf{W}_i^{-1},\mathsf{f}_i}\Big((1-\mathsf{W}_i\overline{\theta}_i)x_k^{(i)} - \mathsf{W}_i(\mathsf{M}_i^*(2u_k^{(i)} - v_k^{(i)}) + \sum_{(\ell,j)\in\mathbb{V}_i^*}(v_k^{(m+\ell,j)} - \theta^{(\ell)}x_k^{(\mathrm{i}(\ell,\overline{\jmath}))}) + \nabla \mathsf{h}_i(x_k^{(i)}) + c_k^{(i)})\Big) + a_k^{(i)}\Big),$$
$$v_{k+1}^{(i)} = v_k^{(i)} + \lambda_k\varepsilon_k^{(i)}(u_k^{(i)} - v_k^{(i)}),$$
$$x_{k+1}^{(i)} = x_k^{(i)} + \lambda_k\varepsilon_k^{(i)}\,(y_k^{(i)} - x_k^{(i)}),$$

En effet dans ce cas, pour tout $k \in \mathbb{N}$ et $\ell \in \{1,\dots,r\}$, on a

$$(\forall j \in \{1,2\}) \quad w_k^{(\ell,j)} = \eta_k^{(\ell)}\Big(\theta^{(\ell)}(x_k^{(\mathrm{i}(\ell,j))} - x_k^{(\mathrm{i}(\ell,\overline{\jmath}))}) + v_k^{(m+\ell,j)}\Big),$$

où, pour tout $j \in \{1,2\}$, $\overline{\jmath} = 3 - j$. Par conséquent, puisque (7.61) *est satisfaite,*

$$(\forall k \in \mathbb{N})(\forall \ell \in \{1,\dots,r\}) \quad v_{k+1}^{(m+\ell,1)} = v_k^{(m+\ell,1)} + \frac{\lambda_k}{2}\eta_k^{(l)}\theta^{(\ell)}(x_k^{(\mathrm{i}(\ell,1))} - x_k^{(\mathrm{i}(\ell,2))}),$$
$$v_{k+1}^{(m+\ell,2)} = -v_{k+1}^{(m+\ell,1)}.$$

En remarquant que puisque $\kappa_\ell \equiv 2$, pour tout $i \in \{1,\dots,m\}$, pour tout $(\ell,j) \in \mathbb{V}_i^*$, $x_k^{(\mathrm{i}(\ell,j))} =$

$x_k^{(i)}$, *et en utilisant* (7.53), *on a*

$$(\forall i \in \{1,\dots,m\}) \quad \varepsilon_k^{(i)} \sum_{(\ell,j)\in\mathbb{V}_i^*} w_k^{(\ell,j)} = \varepsilon_k^{(i)}\Big(\sum_{(\ell,j)\in\mathbb{V}_i^*} (v_k^{(m+\ell,j)} + \theta^{(\ell)}(x_k^{(\mathrm{i}(\ell,j))} - x_k^{(\mathrm{i}(\ell,\overline{j}))})\Big)$$
$$= \varepsilon_k^{(i)}\Big(\overline{\theta}_i x_k^{(i)} + \sum_{(\ell,j)\in\mathbb{V}_i^*} (v_k^{(m+\ell,j)} - \theta^{(\ell)} x_k^{(\mathrm{i}(\ell,\overline{j}))})\Big).$$

Ainsi, en réarrangeant les différentes étapes de l'algorithme 7.15, on obtient l'algorithme 7.16. Dans ce cas, les ensembles $(\mathbb{V}_\ell)_{1\leqslant\ell\leqslant r}$ *peuvent être vus comme les arêtes d'un graphe connecté non orienté dont les nœuds sont indicés par* $i \in \{1,\dots,m\}$.

(iii) *Si* $(\forall i \in \{1,\dots,m\})$ $\mathsf{g}_i = 0$ *et* $\mathsf{l}_i = \iota_{\{0\}}$, $(\forall \ell \in \{1,\dots,r\})$ $\kappa^{(\ell)} = 2$, *et* (7.59) *est satisfaite, alors on peut prendre les* $(\mathsf{M}_i)_{1\leqslant i\leqslant m}$ *aussi petit qu'on le souhaite et l'algorithme 7.16 se réduit à l'algorithme 7.17. Pour cela, nous posons* $(\forall i \in \{1,\dots,m\})$ $v_0^{(i)} = 0$, $(\forall k \in \mathbb{N})$ $\widetilde{\boldsymbol{v}}_k = (\widetilde{v}_k^{(\ell)})_{1\leqslant\ell\leqslant r} = (v_k^{(m+\ell)})_{1\leqslant\ell\leqslant r}$, *et* $\boldsymbol{b}_k = \boldsymbol{d}_k = \boldsymbol{0}$. *Notons que, dans ce cas particulier,*

Algorithme 7.17 Algorithme distribué (version 1 simplifiée) sur un graphe non orienté lorsque $(\forall i \in \{1,\dots,m\})$ $\mathsf{g}_i = 0$ et $\mathsf{l}_i = \iota_{\{0\}}$

Initialisation : Soient $\boldsymbol{x}_0$, $(\boldsymbol{a}_k)_{k\in\mathbb{N}}$ et $(\boldsymbol{c}_k)_{k\in\mathbb{N}}$ des variables aléatoires à valeurs dans $\mathcal{H}^m$.
Pour tout $\ell \in \{1,\dots,r\}$, soient $\theta^{(\ell)} \in]0,+\infty[$, $\widetilde{v}_0^{(\ell,1)}$ une variable aléatoire à valeurs dans $\mathcal{H}$ et $\widetilde{v}_0^{(\ell,2)} = -\widetilde{v}_0^{(\ell,1)}$.
Soient, pour tout $i \in \{1,\dots,m\}$, $\mathsf{W}_i \in \mathcal{S}^+(\mathcal{H})$ et $\overline{\theta}_i$ défini par (7.53).
Soient $(\boldsymbol{\varepsilon}_k)_{k\in\mathbb{N}}$ des variables aléatoires identiquement distribuées à valeurs dans $\mathbb{D}_m$. Soit, pour tout $k \in \mathbb{N}$, $\lambda_k \in]0,+\infty[$.
Itérations :
Pour $k = 0, 1, \dots$
  pour $\ell = 1,\dots,r$
    $\eta_k^{(\ell)} = \max\Big\{\varepsilon_k^{(i)} \;\Big|\; i \in \mathbb{V}_\ell\Big\}$,
    $\widetilde{v}_{k+1}^{(\ell,1)} = \widetilde{v}_k^{(\ell,1)} + \dfrac{\lambda_k}{2}\eta_k^{(\ell)}\theta^{(\ell)}\,(x_k^{(\mathrm{i}(\ell,1))} - x_k^{(\mathrm{i}(\ell,2))})$,
    $\widetilde{v}_{k+1}^{(\ell,2)} = -\widetilde{v}_{k+1}^{(\ell,1)}$,
  pour $i = 1,\dots,m$
    $y_k^{(i)} = \varepsilon_k^{(i)}\Big(\mathrm{prox}_{\mathsf{W}_i^{-1},\mathsf{f}_i}\Big(\big(1-\mathsf{W}_i\overline{\theta}_i\big)x_k^{(i)} - \mathsf{W}_i\Big(\sum_{(\ell,j)\in\mathbb{V}_i^*}(\widetilde{v}_k^{(\ell,j)} - \theta^{(\ell)}\,x_k^{(\mathrm{i}(\ell,\overline{j}))}),$
    $\qquad\qquad +\nabla\mathsf{h}_i(x_k^{(i)}) + c_k^{(i)}\Big)\Big) + a_k^{(i)}\Big)$
    $x_{k+1}^{(i)} = x_k^{(i)} + \lambda_k\varepsilon_k^{(i)}\,(y_k^{(i)} - x_k^{(i)})$.

d'après la remarque 7.9(iv), il suffit que les conditions (7.63) *et* (7.64) *soient satisfaites pour que* (7.54) *soit vérifiée.*

Notons, pour tout $i \in \{1,\dots,m\}$, κ_i^* *le cardinal de l'ensemble* $\mathbb{V}_i^*$ *(ce qui correspond au degré du nœud* i*). Dans le cas particulier où* $\mathcal{H}$ *est un espace euclidien, qu'aucun terme d'erreur n'est pris en compte, que*

$$\begin{aligned} (\forall k \in \mathbb{N}) \quad & \lambda_k = 1, \\ (\forall \ell \in \{1,\dots,r\}) \quad & \theta^{(\ell)} = \theta \in]0,+\infty[, \\ (\forall i \in \{1,\dots,m\}) \quad & \mathsf{W}_i = \tau_i \mathrm{Id} \end{aligned}$$

avec

$$(\forall i \in \{1,\dots,m\}) \quad \tau_i = \tau/\kappa_i^*,$$

et $\tau \in]0,+\infty[$*, on retrouve l'algorithme distribué développé dans* [Bianchi *et al.*, 2014]*. D'après* (7.53)*, pour tout* $i \in \{1,\dots,m\}$, $\overline{\theta}_i = \kappa_i^*\theta$*. L'expression* (7.64) *s'écrit alors*

$$\chi = \max_{i\in\{1,\dots,m\}} (\theta\kappa_i^*\tau_i) = \theta \max_{i\in\{1,\dots,m\}} (\kappa_i^*\tau/\kappa_i^*) = \theta\tau.$$

De plus, pour tout $i \in \{1,\dots,m\}$, μ_i^{-1} *étant une constante de Lipschitz du gradient de* $\mathsf{h}_i \circ \mathsf{W}_i^{1/2}$*, la constante de Lipschitz de* h_i *est alors égale à* $\overline{\mu}_i = \kappa_i^*/(\mu_i\tau)$*. Ainsi* $\mu_i = \kappa_i^*/(\overline{\mu}_i\tau)$*, et la condition* (7.63) *se réécrit*

$$(\forall i \in \{1,\dots,m\}) \quad \tau^{-1} - \theta > \frac{1}{2\min_{i\in\{1,\dots,m\}}(\kappa_i^*/\overline{\mu}_i)}. \tag{7.71}$$

La condition de convergence donnée dans [Bianchi *et al.*, 2014, Thm. 6] *est*

$$\tau^{-1} - \theta > \frac{\overline{\mu}}{2\underline{\kappa}^*}, \tag{7.72}$$

où $\overline{\mu} = \max_{i\in\{1,\dots,m\}} \overline{\mu}_i$ *et* $\underline{\kappa}^* = \min_{i\in\{1,\dots,m\}} \kappa_i^*$*. Or, puisque*

$$\min_{i\in\{1,\dots,m\}} (\kappa_i^*/\overline{\mu}_i) \geqslant \big(\min_{i\in\{1,\dots,m\}} \kappa_i^*\big)\big(\min_{i\in\{1,\dots,m\}} 1/\overline{\mu}_i\big) = \frac{\underline{\kappa}^*}{\overline{\mu}},$$

on a alors

$$\frac{\overline{\mu}}{2\underline{\kappa}^*} \geqslant \frac{1}{2\min_{i\in\{1,\dots,m\}}(\kappa_i^*/\overline{\mu}_i)}.$$

On en déduit ainsi que la condition (7.72) *donnée dans* [Bianchi *et al.*, 2014, Thm. 6] *est plus restrictive que la condition* (7.71) *que nous déduisons de nos résultats de convergence.*

Remarque 7.12. *Le diagramme représenté dans la figure 7.7 donne les différents algorithmes distribués développés dans cette section. Les cadres verts foncés renvoient aux problèmes traités (problèmes 7.3 et 7.4), les cadres verts clairs indiquent les cas particuliers de ces problèmes, et les ellipses bleues donnent les algorithmes permettant de les traiter.*

7.6 Conclusion

Dans ce chapitre nous avons proposé plusieurs algorithmes permettant de résoudre des problèmes d'inclusion monotone ou des problèmes de minimisation de fonctions convexes. Ces algorithmes combinent des méthodes primales-duales, utiles pour traiter des sommes d'opérateurs (ou de fonctions) composés avec des opérateurs linéaires, et des stratégies alternées aléatoirement par blocs proposées dans [Combettes et Pesquet, 2015]. Notons que dans aucun des algorithmes proposés il n'est nécessaire d'inverser des opérateurs linéaires, ce qui peut se révéler utile en pratique lorsque ces derniers sont mal conditionnés ou de très grande dimension. De plus, les algorithmes primaux-duaux proposés peuvent être accélérés grâce à l'utilisation d'opérateurs de préconditionnement. Nous avons montré les performances de la stratégie alternée aléatoirement par bloc au travers d'un exemple applicatif de reconstruction d'un maillage 3D. Les divers algorithmes alternés aléatoirement par bloc sont donnés dans le diagramme de la figure 7.5 Enfin, nous avons déduit de ces algorithmes primaux-duaux alternés par bloc des algorithmes distribués permettant de résoudre des problèmes d'inclusion monotone (ou des problèmes de minimisation de fonctions convexe) en utilisant une topologie générale d'hypergraphes. Un diagramme résumant les liens existants entre les différents algorithmes distribués proposés est donné dans le figure 7.7.

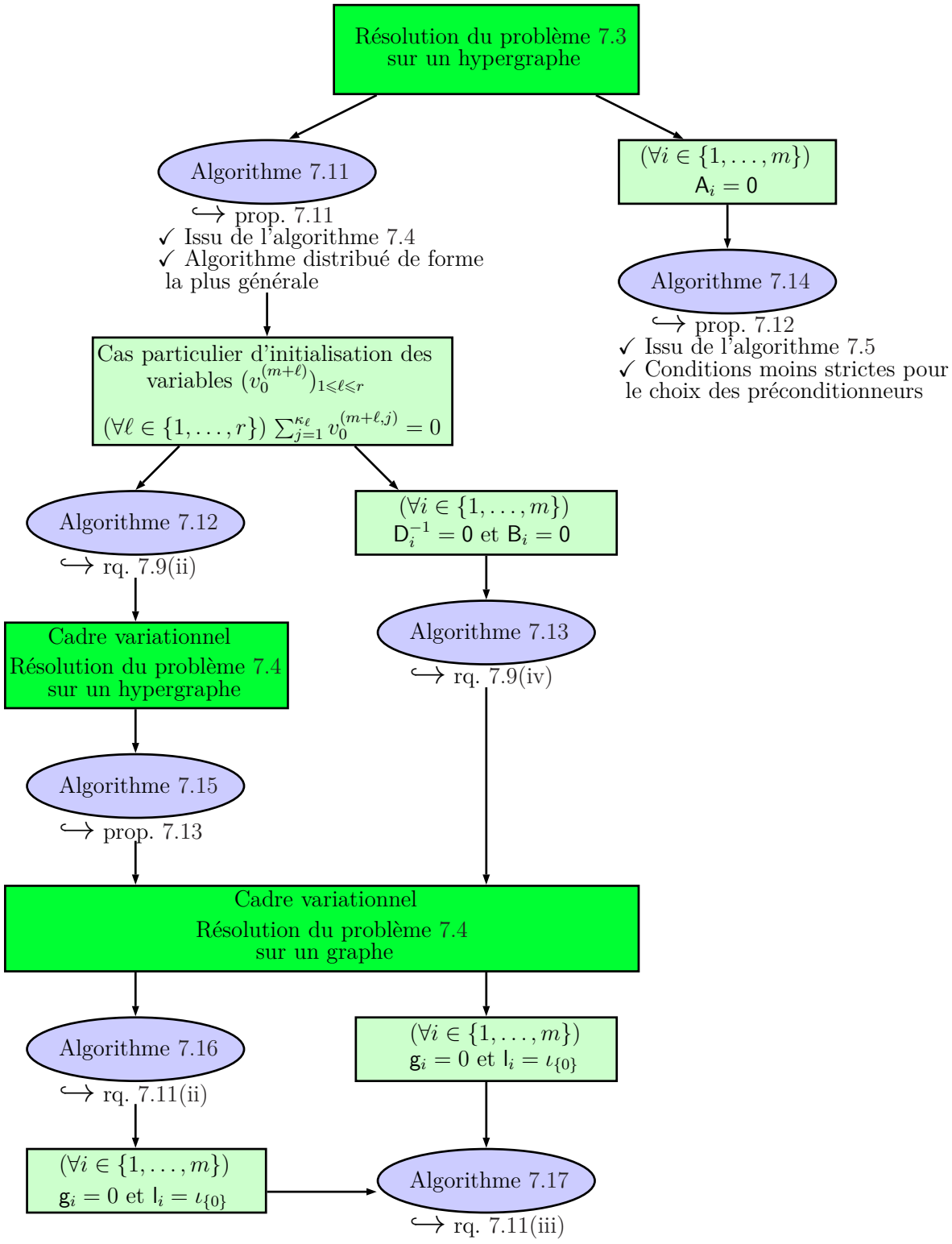


Figure 7.7 – *Diagramme résumant les algorithmes développés dans la section 7.5.*

7.A Démonstration du lemme 7.1

(i) Les opérateurs $\mathbf{W}^{-1}$ et $\mathbf{U}^{-1}$ étant linéaires, bornés et auto-adjoints, $\mathbf{V}'$ est aussi linéaire, borné et auto-adjoint. De plus, pour tout $(\mathbf{x}, \mathbf{v}) \in \mathcal{K}$,

$$\begin{aligned}\Big\langle \mathbf{x} \mid (\mathbf{W}^{-1} - \mathbf{L}^*\mathbf{U}\mathbf{L})\mathbf{x} \Big\rangle &= \Big\langle \mathbf{W}^{-1/2}\mathbf{x} \mid (\mathbf{Id} - \mathbf{W}^{1/2}\mathbf{L}^*\mathbf{U}\mathbf{L}\mathbf{W}^{1/2})\mathbf{W}^{-1/2}\mathbf{x} \Big\rangle \\ &= \Big\langle \mathbf{x} \mid \mathbf{W}^{-1}\mathbf{x} \Big\rangle - \Big\langle \mathbf{W}^{-1/2}\mathbf{x} \mid \mathbf{W}^{1/2}\mathbf{L}^*\mathbf{U}\mathbf{L}\mathbf{W}^{1/2}\mathbf{W}^{-1/2}\mathbf{x} \Big\rangle \\ &\geqslant (1 - \|\mathbf{W}^{1/2}\mathbf{L}^*\mathbf{U}\mathbf{L}\mathbf{W}^{1/2}\|)\Big\langle \mathbf{x} \mid \mathbf{W}^{-1}\mathbf{x} \Big\rangle \\ &\geqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)\|\mathbf{W}\|^{-1}\|\mathbf{x}\|^2 \end{aligned} \tag{7.73}$$

et

$$\begin{aligned}\Big\langle \mathbf{v} \mid (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)\mathbf{v} \Big\rangle &\geqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}\mathbf{L}^*\mathbf{U}^{1/2}\|)\Big\langle \mathbf{v} \mid \mathbf{U}^{-1}\mathbf{v} \Big\rangle \\ &\geqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)\|\mathbf{U}\|^{-1}\|\mathbf{v}\|^2. \end{aligned} \tag{7.74}$$

On peut donc en déduire que

$$\begin{aligned}\big\langle (\mathbf{x}, \mathbf{v}) \mid \mathbf{V}'(\mathbf{x}, \mathbf{v}) \big\rangle &= \Big\langle \mathbf{x} - \mathbf{W}\mathbf{L}^*\mathbf{v} \mid \mathbf{W}^{-1}(\mathbf{x} - \mathbf{W}\mathbf{L}^*\mathbf{v}) \Big\rangle + \Big\langle \mathbf{v} \mid (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)\mathbf{v} \Big\rangle \\ &\geqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)\|\mathbf{U}\|^{-1}\|\mathbf{v}\|^2 \end{aligned}$$

et, de façon similaire,

$$\big\langle (\mathbf{x}, \mathbf{v}) \mid \mathbf{V}'(\mathbf{x}, \mathbf{v}) \big\rangle \geqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)\|\mathbf{W}\|^{-1}\|\mathbf{x}\|^2.$$

Les deux inégalités ci-dessus nous donnent

$$\begin{aligned}\big\langle (\mathbf{x}, \mathbf{v}) \mid \mathbf{V}'(\mathbf{x}, \mathbf{v}) \big\rangle &\geqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2) \min\{\|\mathbf{W}\|^{-1}, \|\mathbf{U}\|^{-1}\} \max\{\|\mathbf{x}\|^2, \|\mathbf{v}\|^2\} \\ &\geqslant \frac{1}{2}(1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2) \min\{\|\mathbf{W}\|^{-1}, \|\mathbf{U}\|^{-1}\}(\|\mathbf{x}\|^2 + \|\mathbf{v}\|^2), \end{aligned}$$

Ce qui montre que $\mathbf{V}'$ est un opérateur fortement positif. C'est donc un isomorphisme et son inverse appartient à $\mathcal{S}^+(\mathcal{K})$.

D'autre part, (7.73) (resp. (7.74)) montre que $\mathbf{W}^{-1} - \mathbf{L}^*\mathbf{U}\mathbf{L}$ (resp. $\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*$) est un isomorphisme puisqu'il appartient à $\mathcal{S}^+(\mathcal{H})$ (resp. $\mathcal{S}^+(\mathcal{G})$). L'expression de l'inverse de $\mathbf{V}'$ peut être obtenue par un calcul direct.

(ii) Soit $\alpha \in]0, +\infty[$. Montrer que $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ est ϑ_α-cocoercif est équivalent à montrer que

$$\begin{aligned}(\forall (\mathbf{z}, \mathbf{z}') \in \mathcal{K}^2) \quad &\Big\langle \mathbf{z} - \mathbf{z}' \mid \mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}\mathbf{z} - \mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}\mathbf{z}' \Big\rangle \\ &\qquad \geqslant \vartheta_\alpha \|\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}\mathbf{z} - \mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}\mathbf{z}'\|^2 \\ \Leftrightarrow (\forall (\mathbf{z}, \mathbf{z}') \in \mathcal{K}^2) \quad &\big\langle \mathbf{z} - \mathbf{z}' \mid \mathbf{R}\mathbf{z} - \mathbf{R}\mathbf{z}' \big\rangle \geqslant \vartheta_\alpha \|\mathbf{R}\mathbf{z} - \mathbf{R}\mathbf{z}'\|_{\mathbf{V}}^2. \end{aligned}$$

Soit $\mathbf{z} = (\mathbf{x}, \mathbf{v}) \in \mathcal{K}$ et $\mathbf{z}' = (\mathbf{x}', \mathbf{v}') \in \mathcal{K}$. On a, d'une part

$$\begin{aligned}\|\mathbf{R}\mathbf{z} - \mathbf{R}\mathbf{z}'\|_{\mathbf{V}}^2 =& \Big\langle \mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}' \mid (\mathbf{W}^{-1} - \mathbf{L}^*\mathbf{U}\mathbf{L})^{-1}(\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}') \Big\rangle \\ &+ \Big\langle \mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}' \mid (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}(\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}') \Big\rangle \\ &+ 2\Big\langle \mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}' \mid \mathbf{W}\mathbf{L}^*(\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}(\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}') \Big\rangle. \end{aligned} \tag{7.75}$$

D'autre part,

$$\begin{aligned}&\Big\langle \mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}' \mid (\mathbf{W}^{-1} - \mathbf{L}^*\mathbf{U}\mathbf{L})^{-1}(\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}') \Big\rangle \\ &\quad = \Big\langle \mathbf{W}^{1/2}(\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}') \mid (\mathbf{Id} - \mathbf{W}^{1/2}\mathbf{L}^*\mathbf{U}\mathbf{L}\mathbf{W}^{1/2})^{-1}\mathbf{W}^{1/2}(\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}') \Big\rangle \\ &\quad \leqslant \|(\mathbf{Id} - \mathbf{W}^{1/2}\mathbf{L}^*\mathbf{U}\mathbf{L}\mathbf{W}^{1/2})^{-1}\| \, \|\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}'\|_{\mathbf{W}}^2 \\ &\quad = (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)^{-1}\|\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}'\|_{\mathbf{W}}^2 , \end{aligned} \tag{7.76}$$

$$\begin{aligned}&\Big\langle \mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}' \mid (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1})(\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}') \Big\rangle \\ &\quad \leqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)^{-1}\|\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}'\|_{\mathbf{U}}^2 , \end{aligned} \tag{7.77}$$

$$\begin{aligned}&\Big\langle \mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}' \mid \mathbf{W}\mathbf{L}^*(\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1})(\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}') \Big\rangle \\ &\quad \leqslant \|\mathbf{W}^{1/2}(\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}')\|\|\mathbf{W}^{1/2}\mathbf{L}^*\mathbf{U}^{1/2}(\mathbf{Id} - \mathbf{U}^{1/2}\mathbf{L}\mathbf{W}\mathbf{L}^*\mathbf{U}^{1/2})^{-1}\| \\ &\qquad\qquad \times \|\mathbf{U}^{1/2}(\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}')\| \\ &\quad \leqslant \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|(1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)^{-1}\|\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}'\|_{\mathbf{W}}\|\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}'\|_{\mathbf{U}} \\ &\quad \leqslant \frac{1}{2}\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|(1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)^{-1} \\ &\qquad\qquad \times (\alpha\|\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}'\|_{\mathbf{W}}^2 + \alpha^{-1}\|\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}'\|_{\mathbf{U}}^2). \end{aligned} \tag{7.78}$$

En combinant les équations (7.75) à (7.78) et en utilisant la cocoercivité de $\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}$ et de $\mathbf{U}^{1/2}\mathbf{D}^{-1}\mathbf{U}^{1/2}$, on obtient la suite d'inégalités :

$$\begin{aligned}\|\mathbf{R}\mathbf{z} - \mathbf{R}\mathbf{z}'\|_{\mathbf{V}}^2 &\leqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)^{-1}\big((1 + \alpha\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|)\|\mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}'\|_{\mathbf{W}}^2 \\ &\quad + (1 + \alpha^{-1}\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|)\|\mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}'\|_{\mathbf{U}}^2\big) \\ &\leqslant (1 - \|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|^2)^{-1}\big(\mu^{-1}(1 + \alpha\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|)\langle \mathbf{x} - \mathbf{x}' \mid \mathbf{C}\mathbf{x} - \mathbf{C}\mathbf{x}'\rangle \\ &\quad + \nu^{-1}(1 + \alpha^{-1}\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|)\Big\langle \mathbf{v} - \mathbf{v}' \mid \mathbf{D}^{-1}\mathbf{v} - \mathbf{D}^{-1}\mathbf{v}' \Big\rangle\big) \\ &\leqslant \vartheta_\alpha^{-1}\langle \mathbf{z} - \mathbf{z}' \mid \mathbf{R}\mathbf{z} - \mathbf{R}\mathbf{z}'\rangle. \end{aligned}$$

7.B Démonstration de la proposition 7.3

D'après la condition (iii), pour tout $j \in \{1, \dots, p\}$, $\max\{\varepsilon_k^{(j)}, (\varepsilon_k^{(p+l)})_{l \in \mathbb{L}_j^*}\} = \varepsilon_k^{(j)}$. De plus, pour tout $l \in \{1, \dots, q\}$, $j \in \mathbb{L}_l \Leftrightarrow l \in \mathbb{L}_j^*$. Les itérations de l'algorithme 7.3 sont donc équivalentes

à

$$
\begin{array}{l}
\text{pour } k=0,1,\ldots\\
\quad\left\lfloor\begin{array}{l}
\text{pour } j=1,\ldots,p\\
\quad\left\lfloor\begin{array}{l}
\eta_k^{(j)}=\max\big\{\varepsilon_k^{(j)},(\varepsilon_k^{(p+l)})_{l\in\mathbb{L}_j^*}\big\}\\
y_k^{(j)}=\eta_k^{(j)}\Big(\mathsf{J}_{\mathsf{W}_j\mathsf{A}_j}\big(x_k^{(j)}-\mathsf{W}_j\big(\sum_{l=1}^{q}\mathsf{L}_{l,j}^*v_k^{(l)}+\mathsf{C}_jx_k^{(j)}+c_k^{(j)}\big)\big)+a_k^{(j)}\Big)\\
x_{k+1}^{(j)}=x_k^{(j)}+\lambda_k\varepsilon_k^{(j)}(y_k^{(j)}-x_k^{(j)})
\end{array}\right.\\
\text{pour } l=1,\ldots,q\\
\quad\left\lfloor\begin{array}{l}
u_k^{(l)}=\varepsilon_k^{(p+l)}\Big(\mathsf{J}_{\mathsf{U}_l\mathsf{B}_l^{-1}}\big(v_k^{(l)}+\mathsf{U}_l\big(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}\big(2y_k^{(j)}-x_k^{(j)}\big)-\mathsf{D}_l^{-1}v_k^{(l)}+d_k^{(l)}\big)\big)+b_k^{(l)}\Big)\\
v_{k+1}^{(l)}=v_k^{(l)}+\lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)}-v_k^{(l)}).
\end{array}\right.
\end{array}\right.
\end{array}
\tag{7.79}
$$

D'autre part, d'après la proposition 7.2(i)-(ii), $\mathbf{Q}$ est maximal monotone, $\mathbf{R}$ est cocoercif, et $\mathbf{Z}=\operatorname{zer}(\mathbf{Q}+\mathbf{R})\neq\varnothing$. On peut remarquer que (7.16) et (7.19) impliquent que $\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|<1$. Ainsi, d'après le lemme 7.2, l'algorithme (7.79) peut être réécrit sous la forme de l'algorithme 7.2, où $m=p+q$, $\mathbf{V}$ est défini par (7.15) et, pour tout $k\in\mathbb{N}$,

$$\boldsymbol{z}_k=(\boldsymbol{x}_k,\boldsymbol{v}_k), \tag{7.80}$$
$$\gamma_k=1, \tag{7.81}$$
$$\mathbf{J}_{\mathbf{VQ}}\colon\mathsf{z}\mapsto(\mathsf{T}_{i,k}\mathsf{z})_{1\leqslant i\leqslant m}, \tag{7.82}$$
$$(\forall j\in\{1,\ldots,p\})\qquad\mathsf{T}_{j,k}\colon\boldsymbol{\mathcal{K}}\to\mathcal{H}_j, \tag{7.83}$$
$$(\forall l\in\{1,\ldots,q\})\qquad\mathsf{T}_{p+l,k}\colon\boldsymbol{\mathcal{K}}\to\mathcal{G}_l, \tag{7.84}$$
$$\boldsymbol{t}_k=(\boldsymbol{a}_k,\boldsymbol{b}_k), \tag{7.85}$$
$$\boldsymbol{s}_k=\big((\mathbf{W}^{-1}-\mathbf{L}^*\mathbf{U}\mathbf{L})^{-1}(\mathbf{L}^*\mathbf{U}\boldsymbol{e}_k-\boldsymbol{c}_k),(\mathbf{U}^{-1}-\mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}(\boldsymbol{e}_k-\mathbf{L}\mathbf{W}\boldsymbol{c}_k)\big), \tag{7.86}$$
$$\boldsymbol{e}_k=2\mathbf{L}\boldsymbol{a}_k+\boldsymbol{d}_k. \tag{7.87}$$

Puisque $\mathbf{W}\in\mathcal{S}^+(\boldsymbol{\mathcal{H}})$ et $\mathbf{U}\in\mathcal{S}^+(\boldsymbol{\mathcal{G}})$ vérifient $\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|<1$, le lemme 7.1(i) assure que $\mathbf{V}\in\mathcal{S}^+(\boldsymbol{\mathcal{K}})$. De plus, pour tout $(\mathbf{x},\widetilde{\mathbf{x}})\in\boldsymbol{\mathcal{H}}^2$,

$$
\begin{aligned}
\Big\langle\mathbf{x}-\widetilde{\mathbf{x}}\mid\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}\mathbf{x}-\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}\widetilde{\mathbf{x}}\Big\rangle&=\sum_{j=1}^{p}\Big\langle\mathsf{x}^{(j)}-\widetilde{\mathsf{x}}^{(j)}\mid\mathsf{W}_j^{1/2}\mathsf{C}_j\mathsf{W}_j^{1/2}\mathsf{x}^{(j)}-\mathsf{W}_j^{1/2}\mathsf{C}_j\mathsf{W}_j^{1/2}\widetilde{\mathsf{x}}^{(j)}\Big\rangle\\
&\geqslant\sum_{j=1}^{p}\mu_j\|\mathsf{W}_j^{1/2}\mathsf{C}_j\mathsf{W}_j^{1/2}\mathsf{x}^{(j)}-\mathsf{W}_j^{1/2}\mathsf{C}_j\mathsf{W}_j^{1/2}\widetilde{\mathsf{x}}^{(j)}\|^2\\
&\geqslant\mu\|\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}\mathbf{x}-\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}\widetilde{\mathbf{x}}\|^2.
\end{aligned}
$$

Donc $\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}$ est μ-cocoercif, et en procédant de façon similaire, on déduit que $\mathbf{U}^{1/2}\mathbf{D}^{-1}\mathbf{U}^{1/2}$ est ν-cocoercif. En conséquence du lemme 7.1(ii), $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ est donc ϑ_α-cocoercif, et nos hypothèses garantissent que $1=\sup_{k\in\mathbb{N}}\gamma_k<2\vartheta_\alpha$. D'autre part, nous pouvons déduire de la condi-

tion (i) et des équations (7.85)-(7.87) que

$$\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{t}_k\|^2\,|\,\mathfrak{X}_k)}\leqslant\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{a}_k\|^2\,|\,\mathfrak{X}_k)}+\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2\,|\,\mathfrak{X}_k)}<+\infty,$$
$$\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{s}_k\|^2\,|\,\mathfrak{X}_k)}$$
$$\leqslant\|(\mathbf{W}^{-1}-\mathbf{L}^*\mathbf{U}\mathbf{L})^{-1}\|\Big(\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{c}_k\|^2\,|\,\mathfrak{X}_k)}+2\|\mathbf{L}^*\mathbf{U}\mathbf{L}\|\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{a}_k\|^2\,|\,\mathfrak{X}_k)}$$
$$+\|\mathbf{L}^*\mathbf{U}\|\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{d}_k\|^2\,|\,\mathfrak{X}_k)}\Big)+\|(\mathbf{U}^{-1}-\mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}\|\Big(2\|\mathbf{L}\|\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{a}_k\|^2\,|\,\mathfrak{X}_k)}$$
$$+\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{d}_k\|^2\,|\,\mathfrak{X}_k)}+\|\mathbf{L}\mathbf{W}\|\sum_{k\in\mathbb{N}}\sqrt{\mathsf{E}(\|\boldsymbol{c}_k\|^2\,|\,\mathfrak{X}_k)}\Big)<+\infty.$$

De plus, puisque nous avons supposé que, pour tout $j\in\{1,\ldots,p\}$, $\mathbb{L}_j^*\neq\varnothing$, (ii) et (iii) garantissent que la condition (ii) de la proposition 7.1 est aussi vérifiée. Par conséquent, toutes les hypothèses de la proposition 7.1 sont satisfaites, ce qui nous permet d'établir la convergence presque sûre de $(\boldsymbol{x}_k,\boldsymbol{v}_k)_{k\in\mathbb{N}}$ vers une variable aléatoire à valeurs dans $\mathbf{Z}$. Finalement, la proposition 7.2(iii) nous assure que la limite est à valeurs dans $\mathbf{F}\times\mathbf{F}^*$.

7.C Démonstration de la proposition 7.5

Tout d'abord, remarquons que, d'après la condition (iii) de la proposition 7.4, puisque $(\forall j\in\{1,\ldots,p\})\ \mathbb{L}_j^*\neq\varnothing$, les itérations de l'algorithme 7.5 sont équivalentes à

$$\begin{array}{l}
\text{pour } k=0,1,\ldots\\
\quad\text{pour } l=1,\ldots,q\\
\quad\quad\zeta_k^{(l)}=\max\{\varepsilon_k^{(p+l)},(\varepsilon_k^{(j)})_{j\in\mathbb{L}_l}\}\\
\quad\text{pour } j=1,\ldots,p\\
\quad\quad\eta_k^{(j)}=\max\{\varepsilon_k^{(j)},(\zeta_k^{(l)})_{l\in\mathbb{L}_j^*}\}\\
\quad\quad w_k^{(j)}=\eta_k^{(j)}\big(x_k^{(j)}-\mathsf{W}_j(\mathsf{C}_jx_k^{(j)}+c_k^{(j)})\big)\\
\quad\text{pour } l=1,\ldots,q\\
\quad\quad u_k^{(l)}=\zeta_k^{(l)}\Big(\mathsf{J}_{\mathsf{U}_l\mathsf{B}_l^{-1}}\big(v_k^{(l)}+\mathsf{U}_l\big(\sum_{j\in\mathbb{L}_l}\mathsf{L}_{l,j}\big(w_k^{(j)}-\mathsf{W}_j\sum_{l'\in\mathbb{L}_j^*}\mathsf{L}_{l',j}^*v_k^{(l')}\big)\\
\qquad\qquad\qquad\qquad-\mathsf{D}_l^{-1}v_k^{(l)}+d_k^{(l)}\big)\big)+b_k^{(l)}\Big)\\
\quad\quad v_{k+1}^{(l)}=v_k^{(l)}+\lambda_k\varepsilon_k^{(p+l)}(u_k^{(l)}-v_k^{(l)})\\
\quad\text{pour } j=1,\ldots,p\\
\quad\quad x_{k+1}^{(j)}=x_k^{(j)}+\lambda_k\varepsilon_k^{(j)}\Big(w_k^{(j)}-\mathsf{W}_j\sum_{l\in\mathbb{L}_j^*}\mathsf{L}_{l,j}^*u_k^{(l)}-x_k^{(j)}\Big).
\end{array}\tag{7.88}$$

De plus, la condition (7.29) implique que $\|\mathbf{U}^{1/2}\mathbf{L}\mathbf{W}^{1/2}\|<1$. Ainsi, le lemme 7.4 nous assure l'équivalence entre les algorithmes (7.88) et 7.2 lorsque $\mathbf{V}$ est donné par (7.25), $\mathbf{Q}$ est donné par

(7.12) (avec $\mathbf{A} = \mathbf{0}$), et $\mathbf{R}$ est donné par (7.13), dès lors que, pour tout $k \in \mathbb{N}$, (7.80)-(7.84) sont satisfaites et que

$$\begin{aligned}
&\boldsymbol{t}_k = (\mathbf{0}, \boldsymbol{b}_k),\\
&\boldsymbol{s}_k = \big(-\mathbf{W}\boldsymbol{e}_{1,k}, (\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}(\mathbf{L}\mathbf{W}\mathbf{L}^*\boldsymbol{b}_k + \boldsymbol{d}_k)\big),\\
&\boldsymbol{e}_{1,k} = \mathbf{L}^*\boldsymbol{b}_k + \boldsymbol{c}_k.
\end{aligned}$$

Dans la démonstration de la proposition 7.3, nous avons vu que $\mathbf{W}^{1/2}\mathbf{C}\mathbf{W}^{1/2}$ est μ-cocoercif et $\mathbf{U}^{1/2}\mathbf{D}^{-1}\mathbf{U}^{1/2}$ est ν-cocoercif. D'après le lemme 7.3(ii), $\mathbf{V}^{1/2}\mathbf{R}\mathbf{V}^{1/2}$ est donc ϑ-cocoercif, où ϑ est donné par (7.26), et (7.29) assure que $1 = \sup_{k\in\mathbb{N}} \gamma_k < 2\vartheta$. De plus,

$$\begin{aligned}
&\sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{t}_k\|^2 \,|\, \mathcal{X}_k)} = \sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2 \,|\, \mathcal{X}_k)} < +\infty,\\
&\sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{s}_k\|^2 \,|\, \mathcal{X}_k)}\\
&\quad\leqslant \|\mathbf{L}\mathbf{W}\mathbf{L}^*\| \sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2 \,|\, \mathcal{X}_k)} + \|\mathbf{W}\| \sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{c}_k\|^2 \,|\, \mathcal{X}_k)}\\
&\quad+ \|(\mathbf{U}^{-1} - \mathbf{L}\mathbf{W}\mathbf{L}^*)^{-1}\| \Big(\|\mathbf{L}\mathbf{W}\mathbf{L}^*\| \sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{b}_k\|^2 \,|\, \mathcal{X}_k)} + \sum_{k\in\mathbb{N}} \sqrt{\mathsf{E}(\|\boldsymbol{d}_k\|^2 \,|\, \mathcal{X}_k)} \Big) < +\infty.
\end{aligned}$$

Puisque nous avons supposé que, pour tout $l \in \{1, \ldots, q\}$, $\mathbb{L}_l \neq \varnothing$, les conditions (ii)-(iii) de la proposition 7.4 assurent que la condition (ii) de la proposition 7.1 est satisfaite. La convergence est obtenue en appliquant cette proposition.

7.D Démonstration de la proposition 7.7

Posons

$$\begin{array}{llll}
(\forall j \in \{1, \ldots, p\}) & \mathsf{A}_j = \partial \mathsf{f}_j, & \mathsf{C}_j = \nabla \mathsf{h}_j,\\
(\forall l \in \{1, \ldots, q\}) & \mathsf{B}_l = \partial \mathsf{g}_l, & \mathsf{D}_l^{-1} = \nabla \mathsf{l}_l^*.
\end{array}$$

On peut alors remarquer que, pour tout $j \in \{1, \ldots, p\}$ et $l \in \{1, \ldots, q\}$,

$$\mathsf{J}_{\mathsf{W}_j\mathsf{A}_j} = \mathrm{prox}_{\mathsf{W}_j^{-1},\mathsf{f}_j} \quad \text{et} \quad \mathsf{J}_{\mathsf{U}_l\mathsf{B}_k^{-1}} = \mathrm{prox}_{\mathsf{U}_l^{-1},\mathsf{g}_l^*},$$

et que les hypothèses de Lipschitz-différentiabilité faites sur h_j et l_l^* sont équivalentes aux hypothèses que $\mathsf{W}_j^{1/2}\mathsf{C}_j\mathsf{W}_j^{1/2}$ est μ_j-cocoercif et $\mathsf{U}_l^{1/2}\mathsf{D}_l^{-1}\mathsf{U}_l^{1/2}$ est ν_l-cocoercif [Bauschke et Combettes, 2011, Cor. 16.42 & 18.16]. La proposition 7.3 nous assure donc que $(\boldsymbol{x}_k)_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire à valeurs dans $\mathbf{F}$ et que $(\boldsymbol{v}_k)_{k\in\mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire à valeurs dans $\mathbf{F}^*$, où $\mathbf{F}$ et $\mathbf{F}^*$ sont définis dans le problème 7.1.

Il nous reste à montrer que la première limite est une variable aléatoire à valeurs dans $\widetilde{\mathbf{F}}$, et que la seconde est une variable aléatoire à valeurs dans $\widetilde{\mathbf{F}}^*$. On définit les fonctions séparables

$\mathbf{f} \in \Gamma_0(\boldsymbol{\mathcal{H}})$, $\mathbf{h} \in \Gamma_0(\boldsymbol{\mathcal{H}})$, $\mathbf{g} \in \Gamma_0(\boldsymbol{\mathcal{G}})$, et $\mathbf{l} \in \Gamma_0(\boldsymbol{\mathcal{G}})$ par

$$\mathbf{f}\colon \mathbf{x} \mapsto \sum_{j=1}^{p} \mathsf{f}_j(\mathsf{x}^{(j)}), \qquad \mathbf{h}\colon \mathbf{x} \mapsto \sum_{j=1}^{p} \mathsf{h}_j(\mathsf{x}^{(j)}),$$
$$\mathbf{g}\colon \mathbf{v} \mapsto \sum_{l=1}^{q} \mathsf{g}_l(\mathsf{v}^{(l)}), \qquad \mathbf{l}\colon \mathbf{v} \mapsto \sum_{l=1}^{q} \mathsf{l}_l(\mathsf{v}^{(l)}).$$

D'après [Bauschke et Combettes, 2011, Prop. 16.8], (7.32) peut être réécrite comme suit

$$\mathbf{0} \in \partial\mathbf{f}(\overline{\mathbf{x}}) + \nabla\mathbf{h}(\overline{\mathbf{x}}) + \mathbf{L}^*(\partial\mathbf{g} \,\square\, \partial\mathbf{l})(\mathbf{L}\overline{\mathbf{x}}). \tag{7.89}$$

Par [Bauschke et Combettes, 2011, Prop. 16.38 & 17.26], puisque $\operatorname{dom}\mathbf{h} = \boldsymbol{\mathcal{H}}$, on a $\partial\mathbf{f} + \nabla\mathbf{h} = \partial(\mathbf{f}+\mathbf{h})$, et puisque $\operatorname{dom}\mathbf{l}^* = \boldsymbol{\mathcal{G}}$, d'après [Bauschke et Combettes, 2011, Prop. 24.27] on a $\partial\mathbf{g} \,\square\, \partial\mathbf{l} = \partial(\mathbf{g} \,\square\, \mathbf{l})$. L'équation (7.89) implique que $\mathbf{L}(\operatorname{dom}(\mathbf{f}+\mathbf{h})) \cap \operatorname{dom}(\mathbf{g} \,\square\, \mathbf{l}) \neq \varnothing$ (d'après [Bauschke et Combettes, 2011, Prop. 16.3(i)]) et on peut déduire de [Bauschke et Combettes, 2011, Prop. 16.5] que

$$(\forall \mathbf{x} \in \boldsymbol{\mathcal{H}}) \qquad \partial\mathbf{f}(\mathbf{x}) + \nabla\mathbf{h}(\mathbf{x}) + \mathbf{L}^*(\partial\mathbf{g} \,\square\, \partial\mathbf{l})(\mathbf{L}\mathbf{x}) \subset \partial(\mathbf{f} + \mathbf{h} + (\mathbf{g} \,\square\, \mathbf{l}) \circ \mathbf{L})(\mathbf{x}).$$

En utilisant (7.10) et la règle de Fermat [Bauschke et Combettes, 2011, Thm. 16.2], nous pouvons conclure que

$$\mathbf{F} = \operatorname{zer}\left(\partial\mathbf{f} + \nabla\mathbf{h} + \mathbf{L}^*(\partial\mathbf{g} \,\square\, \partial\mathbf{l})\mathbf{L}\right) \subset \operatorname{zer}\left(\partial(\mathbf{f} + \mathbf{h} + (\mathbf{g} \,\square\, \mathbf{l}) \circ \mathbf{L})\right) = \widetilde{\mathbf{F}}.$$

Par des arguments similaires, le fait que $\mathbf{F}^* = \operatorname{zer}\left(-\mathbf{L}(\partial\mathbf{f}^* \,\square\, \partial\mathbf{h}^*)(-\mathbf{L}^*) + \partial\mathbf{g}^* + \nabla\mathbf{l}^*\right) \neq \varnothing$ nous permet de montrer que $\mathbf{F}^* \subset \widetilde{\mathbf{F}}^*$.

7.E Démonstration de la proposition 7.11

En utilisant la proposition 7.10, la remarque 7.8, les équations (7.47), (7.50)-(7.51), en posant

$$(\forall \ell \in \{1, \dots, r\}) \quad \mathsf{U}_{m+\ell} = \theta^{(\ell)}\mathrm{Id} \tag{7.90}$$
$$(\forall k \in \mathbb{N}) \quad b_k^{(m+\ell)} = d_k^{(m+\ell)} = 0, \tag{7.91}$$

et en remarquant que $\mathsf{J}_{\mathsf{U}_{m+\ell}\mathsf{N}^{-1}_{\Lambda_{\kappa(\ell)}}} = \mathrm{Id} - \theta^{(\ell)}\Pi_{\Lambda_{\kappa(\ell)}}(\cdot/\theta^{(\ell)}) = \mathrm{Id} - \Pi_{\Lambda_{\kappa(\ell)}}$ (d'après la formule de Moreau (7.1)), l'algorithme 7.4 pour résoudre le problème (7.46) se réécrit

$$
\begin{array}{l}
\text{pour } k = 0, 1, \ldots \\
\quad \text{pour } i = 1, \ldots, m \\
\quad\quad u_k^{(i)} = \varepsilon_k^{(m+i)}\Big(\mathsf{J}_{\mathsf{U}_i\mathsf{B}_i^{-1}}(v_k^{(i)} + \mathsf{U}_i(\mathsf{M}_i x_k^{(i)} - \mathsf{D}_i^{-1} v_k^{(i)} + d_k^{(i)})) + b_k^{(i)}\Big) \\
\quad\quad v_{k+1}^{(i)} = v_k^{(i)} + \lambda_k \varepsilon_k^{(m+i)}(u_k^{(i)} - v_k^{(i)}) \\
\quad \text{pour } \ell = 1, \ldots, r \\
\quad\quad u_k^{(m+\ell)} = \varepsilon_k^{(2m+\ell)}(v_k^{(m+\ell)} + \theta^{(\ell)}\,(x_k^{(i)})_{i\in\mathbb{V}_\ell} - \Pi_{\Lambda_{\kappa(\ell)}}(v_k^{(m+\ell)} + \theta^{(\ell)}\,(x_k^{(i)})_{i\in\mathbb{V}_\ell})) \\
\quad\quad v_{k+1}^{(m+\ell)} = v_k^{(m+\ell)} + \lambda_k \varepsilon_k^{(2m+\ell)}\big(u_k^{(m+\ell)} - v_k^{(m+\ell)}\big) \\
\quad \text{pour } i = 1, \ldots, m \\
\quad\quad y_k^{(i)} = \varepsilon_k^{(i)}\Big(\mathsf{J}_{\mathsf{W}_i\mathsf{A}_i}(x_k^{(i)} - \mathsf{W}_i(\mathsf{M}_i^*(2u_k^{(i)} - v_k^{(i)}) + \sum_{(\ell,j)\in\mathbb{V}_i^*}(2u_k^{(m+\ell,j)} - v_k^{(m+\ell,j)}) \\
\qquad\qquad\qquad\qquad + \mathsf{C}_i x_k^{(i)} + c_k^{(i)})) + a_k^{(i)}\Big) \\
\quad\quad x_{k+1}^{(i)} = x_k^{(i)} + \lambda_k \varepsilon_k^{(i)}\,(y_k^{(i)} - x_k^{(i)}).
\end{array}
\tag{7.92}
$$

Or, pour tout $\ell \in \{1, \ldots, r\}$, la projection dans l'espace vectoriel $\Lambda_{\kappa(\ell)}$ a une forme explicite donnée par

$$
(\forall k \in \mathbb{N}) \quad \Pi_{\Lambda_{\kappa(\ell)}}(v_k^{(m+\ell)} + \theta^{(\ell)}\,(x_k^{(i)})_{i\in\mathbb{V}_\ell}) = \frac{1}{\kappa^{(\ell)}}\Big(\sum_{j=1}^{\kappa^{(\ell)}} v_k^{(m+\ell,j)} + \theta^{(\ell)}\sum_{i\in\mathbb{V}_\ell} x_k^{(i)}\Big).
$$

Ainsi, l'algorithme (7.92) se réécrit

$$
\begin{array}{l}
\text{pour } k = 0, 1, \ldots \\
\quad \text{pour } i = 1, \ldots, m \\
\quad\quad u_k^{(i)} = \varepsilon_k^{(m+i)}\Big(\mathsf{J}_{\mathsf{U}_i\mathsf{B}_i^{-1}}(v_k^{(i)} + \mathsf{U}_i(\mathsf{M}_i x_k^{(i)} - \mathsf{D}_i^{-1} v_k^{(i)} + d_k^{(i)})) + b_k^{(i)}\Big) \\
\quad\quad v_{k+1}^{(i)} = v_k^{(i)} + \lambda_k \varepsilon_k^{(m+i)}(u_k^{(i)} - v_k^{(i)}) \\
\quad \text{pour } \ell = 1, \ldots, r \\
\quad\quad \overline{u}_k^{(\ell)} = \varepsilon_k^{(2m+\ell)}\frac{1}{\kappa^{(\ell)}}\Big(\sum_{j=1}^{\kappa^{(\ell)}} v_k^{(m+\ell,j)} + \theta^{(\ell)}\sum_{i\in\mathbb{V}_\ell} x_k^{(i)}\Big) \\
\quad\quad \text{pour } j = 1, \ldots, \kappa^{(\ell)} \\
\quad\quad\quad u_k^{(m+\ell,j)} = \varepsilon_k^{(2m+\ell)}(v_k^{(m+\ell,j)} + \theta^{(\ell)}\,x_k^{(\mathrm{i}(\ell,j))} - \overline{u}_k^{(\ell)}) \\
\quad\quad v_{k+1}^{(m+\ell)} = v_k^{(m+\ell)} + \lambda_k \varepsilon_k^{(2m+\ell)}\big(u_k^{(m+\ell)} - v_k^{(m+\ell)}\big) \\
\quad \text{pour } i = 1, \ldots, m \\
\quad\quad y_k^{(i)} = \varepsilon_k^{(i)}\Big(\mathsf{J}_{\mathsf{W}_i\mathsf{A}_i}(x_k^{(i)} - \mathsf{W}_i(\mathsf{M}_i^*(2u_k^{(i)} - v_k^{(i)}) + \sum_{(\ell,j)\in\mathbb{V}_i^*}(2u_k^{(m+\ell,j)} - v_k^{(m+\ell,j)}) \\
\qquad\qquad\qquad\qquad + \mathsf{C}_i x_k^{(i)} + c_k^{(i)})) + a_k^{(i)}\Big) \\
\quad\quad x_{k+1}^{(i)} = x_k^{(i)} + \lambda_k \varepsilon_k^{(i)}\,(y_k^{(i)} - x_k^{(i)}).
\end{array}
\tag{7.93}
$$

Posons, pour tout $k \in \mathbb{N}$ et pour tout $\ell \in \{1, \dots, r\}$,

$$\overline{x}_k^{(\ell)} = \frac{1}{\kappa^{(\ell)}} \sum_{i \in \mathbb{V}_\ell} x_k^{(i)}, \tag{7.94}$$

$$w_k^{(\ell)} = \varepsilon_k^{(2m+\ell)} \big(2u_k^{(m+\ell)} - v_k^{(m+\ell)}\big), \tag{7.95}$$

$$\eta_k^{(\ell)} = \max\{\varepsilon_k^{(i)} \mid i \in \mathbb{V}_\ell\}. \tag{7.96}$$

En utilisant (7.57) et l'étape de mise à jour

$$\overline{x}_{k+1}^{(\ell)} = \overline{x}_k^{(\ell)} + \eta_k^{(\ell)} \Big(\frac{1}{\kappa^{(\ell)}} \sum_{i \in \mathbb{V}_\ell} x_{k+1}^{(i)} - \overline{x}_k^{(\ell)} \Big),$$

les itérations de l'algorithme 7.11 sont obtenues en réordonnant les étapes dans (7.93).

Afin d'appliquer la proposition 7.4, il nous faut maintenant démontrer que la condition (7.19), où ϑ_α est défini par (7.16), est satisfaite. Soit

$$\mathbf{W} \colon \mathcal{H}^m \to \mathcal{H}^m \colon \mathbf{x} \mapsto (\mathsf{W}_i \mathsf{x}^{(i)})_{1 \leqslant i \leqslant m} \quad \text{et} \quad \mathbf{U} \colon \boldsymbol{\mathcal{G}} \oplus \boldsymbol{\mathcal{H}} \to \boldsymbol{\mathcal{G}} \oplus \boldsymbol{\mathcal{H}} \colon \mathbf{v} \mapsto (\mathsf{U}_l \mathsf{v}^{(l)})_{1 \leqslant l \leqslant m+r}.$$

D'après la remarque 7.8 et l'équation (7.90), on a

$$(\forall \mathbf{x} \in \mathcal{H}^m) \quad \mathbf{U}^{1/2} \mathbf{L} \mathbf{W}^{1/2} \mathbf{x} = (\mathbf{U}_1^{1/2} \mathbf{M} \mathbf{W}^{1/2} \mathbf{x}, \mathbf{U}_2^{1/2} \mathbf{S} \mathbf{W}^{1/2} \mathbf{x})$$

où $\mathbf{U}_1 \colon \boldsymbol{\mathcal{G}} \to \boldsymbol{\mathcal{G}} \colon (\mathsf{v}^{(i)})_{1 \leqslant i \leqslant m} \mapsto (\mathsf{U}_i \mathsf{v}^{(i)})_{1 \leqslant i \leqslant m}$ et $\mathbf{U}_2 \colon \boldsymbol{\mathcal{H}} \to \boldsymbol{\mathcal{H}} \colon (\mathsf{v}^{(m+\ell)})_{1 \leqslant \ell \leqslant r} \mapsto (\theta^{(\ell)} \mathsf{v}^{(m+\ell)})_{1 \leqslant \ell \leqslant r}$. On en déduit que

$$\begin{aligned} \|\mathbf{U}^{1/2} \mathbf{L} \mathbf{W}^{1/2} \mathbf{x}\|^2 &= \|\mathbf{U}_1^{1/2} \mathbf{M} \mathbf{W}^{1/2} \mathbf{x}\|^2 + \|\mathbf{U}_2^{1/2} \mathbf{S} \mathbf{W}^{1/2} \mathbf{x}\|^2 \\ &= \sum_{i=1}^m \|\mathsf{U}_i^{1/2} \mathsf{M}_i \mathsf{W}_i^{1/2} \mathsf{x}^{(i)}\|^2 + \Big\langle \mathbf{W}^{1/2} \mathbf{x} \mid \mathbf{S}^* \mathbf{U}_2 \mathbf{S} \mathbf{W}^{1/2} \mathbf{x} \Big\rangle. \end{aligned}$$

En utilisant (7.47) et (7.50)-(7.51), on peut de plus remarquer que

$$\mathbf{S}^* \mathbf{U}_2 \mathbf{S} \colon \mathcal{H}^m \to \mathcal{H}^m \colon (\mathsf{x}^{(i)})_{1 \leqslant i \leqslant m} \mapsto (\overline{\theta}_i \mathsf{x}^{(i)})_{1 \leqslant i \leqslant m}$$

ce qui implique

$$\begin{aligned} \|\mathbf{U}^{1/2} \mathbf{L} \mathbf{W}^{1/2} \mathbf{x}\|^2 &= \sum_{i=1}^m \|\mathsf{U}_i^{1/2} \mathsf{M}_i \mathsf{W}_i^{1/2} \mathsf{x}^{(i)}\|^2 + \sum_{i=1}^m \overline{\theta}_i \|\mathsf{W}_i^{1/2} \mathsf{x}^{(i)}\|^2 \\ &\leqslant \sum_{i=1}^m (\|\mathsf{U}_i^{1/2} \mathsf{M}_i \mathsf{W}_i^{1/2}\|^2 + \overline{\theta}_i \|\mathsf{W}_i\|) \|\mathsf{x}^{(i)}\|^2 \leqslant \chi \|\mathbf{x}\|^2, \end{aligned}$$

et on en déduit donc que $\|\mathbf{U}^{1/2} \mathbf{L} \mathbf{W}^{1/2}\|^2 \leqslant \chi$. Ainsi, (7.54) implique (7.19).

De plus, la condition (iii) de la proposition 7.4 se réécrit sous la forme de la condition (iii) de l'actuelle proposition. On peut donc déduire des propositions 7.4 et 7.10 que, pour tout $i \in \{1, \dots, m\}$, $(x_k^{(i)})_{k \in \mathbb{N}}$ converge faiblement P-presque sûrement vers une variable aléatoire $\widehat{x}$ à valeurs dans $\widehat{\mathsf{F}}$. En conséquence de (7.94), pour tout $\ell \in \{1, \dots, r\}$, $(\overline{x}_k^{(\ell)})_{k \in \mathbb{N}}$ converge aussi faiblement P-presque sûrement vers $\widehat{x}$.

7.F Démonstration de la proposition 7.12

En choisissant $(\mathsf{U}_{m+\ell})_{1\leqslant\ell\leqslant r}$ comme dans (7.90) et en considérant des termes d'erreur nuls comme dans (7.91), l'algorithme 7.5 pour résoudre le problème (7.46) peut être réécrit de la façon suivante :

pour $k = 0, 1, \ldots$

  pour $i = 1, \ldots, m$

    $\eta_k^{(i)} = \max\big\{\varepsilon_k^{(m+i)}, (\varepsilon_k^{(2m+\ell)})_{\ell\in\{\ell'\in\{1,\ldots,r\}\,|\,i\in\mathbb{V}_{\ell'}\}}\big\}$

    $w_k^{(i)} = \eta_k^{(i)}\big(x_k^{(i)} - \mathsf{W}_i(\mathsf{C}_i x_k^{(i)} + c_k^{(i)})\big)$

    $\widetilde{w}_k^{(i)} = \eta_k^{(i)}\big(w_k^{(i)} - \mathsf{W}_i(\mathsf{M}_i^* v_k^{(i)} + \sum_{(\ell,j)\in\mathbb{V}_i^*} v_k^{(m+\ell,j)})\big)$

    $u_k^{(i)} = \varepsilon_k^{(m+i)}\Big(\mathsf{J}_{\mathsf{U}_i\mathsf{B}_i^{-1}}\big(v_k^{(i)} + \mathsf{U}_i(\mathsf{M}_i\widetilde{w}_k^{(i)} - \mathsf{D}_i^{-1}v_k^{(i)} + d_k^{(i)})\big) + b_k^{(i)}\Big)$

    $v_{k+1}^{(i)} = v_k^{(i)} + \lambda_k\varepsilon_k^{(m+i)}(u_k^{(i)} - v_k^{(i)})$

  pour $\ell = 1, \ldots, r$

    $u_k^{(m+\ell)} = \varepsilon_k^{(2m+\ell)}\big(v_k^{(m+\ell)} + \theta^{(\ell)}\,(\widetilde{w}_k^{(i)})_{i\in\mathbb{V}_\ell} - \Pi_{\Lambda_{\kappa(\ell)}}(v_k^{(m+\ell)} + \theta^{(\ell)}\,(\widetilde{w}_k^{(i)})_{i\in\mathbb{V}_\ell})\big)$

    $v_{k+1}^{(m+\ell)} = v_k^{(m+\ell)} + \lambda_k\varepsilon_k^{(2m+\ell)}(u_k^{(m+\ell)} - v_k^{(m+\ell)})$

  pour $i = 1, \ldots, m$

    $x_{k+1}^{(i)} = x_k^{(i)} + \lambda_k\varepsilon_k^{(i)}\Big(w_k^{(i)} - \mathsf{W}_i\big(\mathsf{M}_i^* u_k^{(i)} + \sum_{(\ell,j)\in\mathbb{V}_i^*} u_k^{(m+\ell,j)}\big) - x_k^{(i)}\Big)$.

Le reste de la démonstration est identique à celle de la proposition 7.11.

Chapitre 8

Opérateur proximal de fonctions quotient

Sommaire

8.1 Introduction

Le but du travail présenté dans ce chapitre est l'optimisation des requêtes dans les *Systèmes de Gestion de Bases de Données* (SGBD ou DBMS en anglais pour *Database Management Systems*). Dans ce contexte, le plan d'exécution des requêtes optimales dépend de la précision de l'estimation des sélectivités (la sélectivité correspond à la proportion de n-uplets satisfaisants les prédicats d'une requête). Les modèles pour l'estimation des n-uplets, tels que ceux présentés dans [Markl *et al.*, 2007], requièrent la résolution d'un problème de faisabilité. Plus précisément, soient $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{x} \in \mathbb{R}^N$ et $\mathbf{b} \in \mathbb{R}^N$. Nous considérons un système linéaire sous-déterminé $\mathbf{Ax} = \mathbf{b}$ n'admettant pas de solution positive ou nulle. Nous allons alors rechercher à remplacer $\mathbf{b}$ par un élément $\widehat{\mathbf{b}}$ qui soit "correct" au sens où le système $\mathbf{Ax} = \widehat{\mathbf{b}}$ admette une solution positive ou nulle. Bien que les problèmes de faisabilité aient été étudiés dans divers travaux (voir [Chinneck, 2008; Combettes, 1996], et les références associées), notre approche sera différente de celles existantes par le critère que l'on va proposer de minimiser. Les approches existantes s'intéressent à la minimisation des différences des éléments $\widehat{\mathsf{b}}^{(n)}$ et $\mathsf{b}^{(n)}$, pour tout $n \in \{1, \ldots, N\}$, tandis que nous minimiserons leurs quotients $\max\{\widehat{\mathsf{b}}^{(n)}/\mathsf{b}^{(n)}, \mathsf{b}^{(n)}/\widehat{\mathsf{b}}^{(n)}\}$. L'intérêt d'utiliser les quotients par rapport aux différences a été mis en évidence dans [Moerkotte *et al.*, 2009]. Ainsi, dans ce chapitre, nous proposons d'étudier la somme de ces quotients, désignée par $\mathbf{q}_1$, ainsi que leur maximum, noté $\mathbf{q}_\infty$.

Nous proposons d'appliquer un algorithme primal-dual pour résoudre les problèmes faisant intervenir les fonctions quotient $\mathbf{q}_\nu$, $\nu = 1, \infty$. Pour cela, nous allons dans un premier temps analyser les opérateurs proximaux des fonctions quotient $\mathbf{q}_\nu$, $\nu = 1, \infty$. Nous démontrerons que l'opérateur proximal de la somme des quotients, $\mathbf{q}_1$, est une fonction de seuillage. Nous verrons que cette fonction est un peu plus complexe que la fonction de seuillage doux, correspondant à l'opérateur proximal de la norme ℓ_1, car son calcul fait intervenir une équation du troisième degré. Concernant l'opérateur proximal du maximum des quotients $\mathbf{q}_\infty$, nous montrerons qu'il peut être calculé de façon itérative grâce à un algorithme primal-dual en utilisant des projections épigraphiques terme à terme [Chierchia *et al.*, 2013a]. Nous utiliserons les résultats obtenus pour résoudre le problème de faisabilité décrit ci-dessus, et donnerons un exemple d'application numérique.

Le reste du chapitre sera organisé comme suit : Dans la section 8.2, nous introduirons les distances quotient entre des éléments positifs puis nous les utiliserons pour définir les fonctions quotient de vecteurs d'éléments positifs. Ensuite, dans la section 8.3, nous nous intéresserons à la détermination de l'opérateur proximal des fonctions quotient. Nous proposerons dans la section 8.4 une méthode utilisant ces opérateurs proximaux pour résoudre des problèmes de faisabilité apparaissant, notamment, dans les estimations de n-uplets nécessaires pour l'optimisation de requêtes en SGBD. Après avoir décrit le problème d'estimation des n-uplets, nous proposerons un algorithme de minimisation primal-dual pour le résoudre et illustrerons ses performances par un exemple numérique. Nous conclurons dans la section 8.5.

8.2 Fonctions quotient

Nous définissons la fonction quotient q par

$$\begin{aligned} \mathsf{q}\colon \quad]0,+\infty[^2 &\to [0,+\infty[\\ (\mathsf{x},\mathsf{y}) &\mapsto \mathsf{q}(\mathsf{x},\mathsf{y}) := \frac{\max(\mathsf{x},\mathsf{y})}{\min(\mathsf{x},\mathsf{y})}. \end{aligned} \tag{8.1}$$

Remarquons que la fonction $(\mathsf{x},\mathsf{y}) \in]0,+\infty[^2 \mapsto \mathsf{q}(\mathsf{x},\mathsf{y}) - 1$ vérifie deux des conditions définissant une distance :

- elle est symétrique $((\forall (\mathsf{x},\mathsf{y}) \in]0,+\infty[^2)\ \mathsf{q}(\mathsf{x},\mathsf{y}) = \mathsf{q}(\mathsf{y},\mathsf{x}))$,
- elle vérifie $\mathsf{q}(\mathsf{x},\mathsf{y}) - 1 = 0$ si et seulement si $\mathsf{x} = \mathsf{y}$ (puisque $\mathsf{q}(\mathsf{x},\mathsf{y}) - 1 = \frac{|\mathsf{x}-\mathsf{y}|}{\min(\mathsf{x},\mathsf{y})}$).

Cependant, elle ne satisfait pas l'inégalité triangulaire. Une fonction similaire à la fonction quotient, appelée *distance relative généralisée*, définie par

$$(\mathsf{x},\mathsf{y}) \in \mathbb{R}^* \times \mathbb{R}^* \mapsto \frac{|\mathsf{x}-\mathsf{y}|}{\max(\mathsf{x},\mathsf{y})}, \tag{8.2}$$

a été utilisée dans [Coutinho et Figueiredo, 2005; Griesel, 1974; Metcalf, 1976; Ziv, 1982]. Le lien entre ces deux fonctions (8.1) et (8.2) a été étudié dans [Setzer *et al.*, 2010]. D'autre part, la fonction quotient (8.1) a été utilisée comme une mesure de contraste en traitement d'image dans [Palma-Amestoy *et al.*, 2009] en raison de sa propriété d'homogénéité, i.e.

$$(\forall (\mathsf{x},\mathsf{y}) \in]0,+\infty[^2)(\forall \lambda > 0) \quad \mathsf{q}(\lambda\mathsf{x},\lambda\mathsf{y}) = \mathsf{q}(\mathsf{x},\mathsf{y}) \tag{8.3}$$

Soit $\mathsf{b} \in]0,+\infty[$ fixé, on se propose d'étendre la fonction $\mathsf{q}(\cdot,\mathsf{b})$ à l'ensemble de l'axe des réels de la façon suivante :

$$\begin{aligned} \mathsf{q}(\cdot,\mathsf{b})\colon \quad \mathbb{R} &\to [0,+\infty[\\ \mathsf{x} &\mapsto \mathsf{q}(\mathsf{x},\mathsf{b}) := \begin{cases} \frac{\mathsf{x}}{\mathsf{b}} & \text{si } \mathsf{b} \leqslant \mathsf{x}, \\ \frac{\mathsf{b}}{\mathsf{x}} & \text{si } 0 < \mathsf{x} < \mathsf{b}, \\ +\infty & \text{sinon.} \end{cases} \end{aligned} \tag{8.4}$$

Notons que cette fonction est convexe et semi-continue inférieurement, et que, de plus, d'après (8.3), on a

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \mathsf{q}(\mathsf{x},\mathsf{b}) = \mathsf{q}(\mathsf{x}/\mathsf{b},1). \tag{8.5}$$

Dans la suite du chapitre nous noterons

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \theta(\mathsf{x}) := \mathsf{q}(\mathsf{x},1). \tag{8.6}$$

Soit $\mathbf{b} = (\mathsf{b}^{(n)})_{1\leqslant n\leqslant N} \in]0,+\infty[^N$. Nous nous intéressons d'une part à la somme de fonctions quotient $\mathbf{q}_1(\cdot,\mathbf{b})\colon \mathbb{R}^N \to [0,+\infty]$ et d'autre part à leur maximum $\mathbf{q}_\infty(\cdot,\mathbf{b})\colon \mathbb{R}^N \to [0,+\infty]$. Ces deux fonctions sont définies par

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathbf{q}_1(\mathbf{x},\mathbf{b}) := \sum_{n=1}^{N} \mathsf{q}(\mathsf{x}^{(n)},\mathsf{b}^{(n)}), \tag{8.7}$$

$$\mathbf{q}_\infty(\mathbf{x},\mathbf{b}) := \max_{1\leqslant n\leqslant N} \mathsf{q}(\mathsf{x}^{(n)},\mathsf{b}^{(n)}). \tag{8.8}$$

De plus, pour $\nu \in \{1, \infty\}$, nous posons $(\forall \mathbf{x} \in \mathbb{R}^N)$ $\boldsymbol{\theta}_\nu(\mathbf{x}) := \mathbf{q}_\nu(\mathbf{x}, \mathbf{1}_N)$, où $\mathbf{1}_N$ est le vecteur unitaire de $\mathbb{R}^N$. Notons $\mathbf{B} = \mathrm{Diag}(\mathbf{b})$ la matrice diagonale de $\mathbb{R}^{N\times N}$ dont les éléments diagonaux correspondent aux éléments du vecteur $\mathbf{b} = (\mathsf{b}^{(n)})_{1\leqslant n\leqslant N}$. D'après (8.5), on a alors

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathbf{q}_\nu(\mathbf{x}, \mathbf{b}) = \boldsymbol{\theta}_\nu\left(\mathbf{B}^{-1}\mathbf{x}\right), \tag{8.9}$$

8.3 Opérateur proximal de fonctions quotient

Dans cette section, nous nous intéressons aux opérateurs proximaux des fonctions convexes $\mathbf{q}_1(\cdot, \mathbf{b})$ et $\mathbf{q}_\infty(\cdot, \mathbf{b})$, pour $\mathbf{b} \in]0, +\infty[^N$ fixé. D'après l'équation (8.9) on a, pour $\nu \in \{1, \infty\}$,

$$\begin{aligned}
(\forall \gamma \in]0, +\infty[)(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathrm{prox}_{\gamma\mathbf{q}_\nu(\cdot,\mathbf{b})}(\mathbf{x}) &= \operatorname*{argmin}_{\mathbf{y}\in\mathbb{R}^N} \ \mathbf{q}_\nu(\mathbf{y}, \mathbf{b}) + \frac{1}{2\gamma}\|\mathbf{x} - \mathbf{y}\|^2 \\
&= \operatorname*{argmin}_{\mathbf{y}\in\mathbb{R}^N} \ \boldsymbol{\theta}_\nu\left(\mathbf{B}^{-1}\mathbf{y}\right) + \frac{1}{2\gamma}\|\mathbf{x} - \mathbf{y}\|^2 \\
&= \mathbf{B} \operatorname*{argmin}_{\mathbf{y}'\in\mathbb{R}^N} \ \boldsymbol{\theta}_\nu(\mathbf{y}') + \frac{1}{2\gamma}\left\langle \mathbf{B}^{-1}\mathbf{x} - \mathbf{y}' \,\middle|\, \mathbf{B}^2(\mathbf{B}^{-1}\mathbf{x} - \mathbf{y}')\right\rangle \\
&= \mathbf{B} \operatorname*{argmin}_{\mathbf{y}'\in\mathbb{R}^N} \ \boldsymbol{\theta}_\nu(\mathbf{y}') + \frac{1}{2\gamma}\|\mathbf{B}^{-1}\mathbf{x} - \mathbf{y}'\|^2_{\mathbf{B}^2} \\
&= \mathbf{B}\,\mathrm{prox}_{\gamma^{-1}\mathbf{B}^2,\boldsymbol{\theta}_\nu}\left(\mathbf{B}^{-1}\mathbf{x}\right),
\end{aligned} \tag{8.10}$$

où, pour tout $\alpha \in \mathbb{R}$, $\mathbf{B}^\alpha = \mathrm{Diag}\left(((\mathsf{b}^{(n)})^\alpha)_{1\leqslant n\leqslant N}\right)$. Par conséquent, il nous suffit d'étudier l'opérateur proximal :

$$(\forall \gamma \in]0, +\infty[)(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathrm{prox}_{\mathbf{U},\boldsymbol{\theta}_\nu}(\mathbf{x}) = \operatorname*{argmin}_{\mathbf{y}\in\mathbb{R}^N} \ \boldsymbol{\theta}_\nu(\mathbf{y}) + \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|^2_{\mathbf{U}}, \tag{8.11}$$

où $\nu \in \{1, \infty\}$ et $\mathbf{U} = \mathrm{Diag}(\mathbf{u})$ pour $\mathbf{u} \in]0, +\infty[^N$.

8.3.1 Opérateur proximal de la somme des fonctions quotient

Pour $\nu = 1$, la fonction $\boldsymbol{\theta}_1$ étant additivement séparable, l'argument minimal de (8.11) peut être calculé terme à terme, i.e.,

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathrm{prox}_{\mathbf{U},\boldsymbol{\theta}_1}(\mathbf{x}) = \left(\mathrm{prox}_{\theta/\mathsf{u}^{(n)}}(\mathsf{x}^{(n)})\right)_{1\leqslant n\leqslant N}. \tag{8.12}$$

Ainsi, il nous suffit d'étudier

$$(\forall \mathsf{x} \in \mathbb{R}^N) \quad \mathrm{prox}_{\gamma\theta}(\mathsf{x}) = \operatorname*{argmin}_{\mathsf{y}\in\mathbb{R}} \ \theta(\mathsf{y}) + \frac{1}{2\gamma}(\mathsf{x} - \mathsf{y})^2, \tag{8.13}$$

pour $\gamma > 0$. L'expression de cet opérateur proximal est donnée dans la proposition suivante.

Proposition 8.1. *Soient $\gamma > 0$ et $\mathsf{x} \in \mathbb{R}$. L'opérateur proximal de la fonction quotient définie par* (8.4)-(8.6) *est donné par*

$$\operatorname{prox}_{\gamma\theta}(\mathsf{x}) = \begin{cases} \zeta^* & \text{si } \mathsf{x} < 1-\gamma, \\ 1 & \text{si } \mathsf{x} \in [1-\gamma, 1+\gamma], \\ \mathsf{x}-\gamma & \text{si } \mathsf{x} > 1+\gamma, \end{cases} \tag{8.14}$$

où ζ^* est l'unique solution dans $]0,1]$ du polynôme de degré trois

$$\mathsf{r}(\zeta) = \zeta^3 - \mathsf{x}\zeta^2 - \gamma = 0. \tag{8.15}$$

Démonstration. Soit $\gamma \in]0,+\infty[$. Nous allons appliquer la proposition 2.10 affirmant que, pour toute fonction $\psi \in \Gamma_0(\mathbb{R})$ différentiable en 0 telle que $\dot{\psi}(0) = 0$, on a $\operatorname{prox}_{\psi+\gamma|\cdot|} = \operatorname{prox}_{\psi} \circ \operatorname{soft}_{\gamma}$ [1]. Pour cela, nous allons décomposer la fonction θ de la façon suivante :

$$(\forall \mathsf{y} \in \mathbb{R}) \quad \theta(\mathsf{y}) = 1 + \phi(\mathsf{y}-1), \tag{8.16}$$

où $(\forall \mathsf{z} \in \mathbb{R})$ $\phi(\mathsf{z}) := \psi(\mathsf{z}) + |\mathsf{z}|$ avec

$$\psi(\mathsf{z}) := \begin{cases} +\infty & \text{si } \mathsf{z} \leqslant -1, \\ \mathsf{z} + \dfrac{1}{1+\mathsf{z}} - 1 & \text{si } \mathsf{z} \in]-1, 0], \\ 0 & \text{si } \mathsf{z} > 0. \end{cases}$$

Notons que la fonction ψ est dans $\Gamma_0(\mathbb{R})$, elle est différentiable en zéro et vérifie $\dot{\psi}(0) = 0$. Ainsi, $(\forall \mathsf{z} \in \mathbb{R})$ $\psi(\mathsf{z}) \geqslant 0$. Intéressons nous à l'opérateur proximal de $\gamma\psi$:

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \operatorname{prox}_{\gamma\psi}(\mathsf{x}) = \operatorname*{argmin}_{\mathsf{z}\in\mathbb{R}} \ \{\varphi(\mathsf{z}) := \psi(\mathsf{z}) + \frac{1}{2\gamma}(\mathsf{z}-\mathsf{x})^2\}. \tag{8.17}$$

- Étudions le cas où $\mathsf{x} \in]0,+\infty[$. D'une part, la fonction $\mathsf{z} \in \mathbb{R} \mapsto \psi(\mathsf{z})$ est convexe et sa valeur minimale, égale à 0, est atteinte dès que $\mathsf{z} \in]0,+\infty[$. D'autre part, la fonction $\mathsf{z} \in \mathbb{R} \mapsto \frac{1}{2}(\mathsf{z}-\mathsf{x})^2$ atteint son minimum en $\mathsf{z} = \mathsf{x} \in]0,+\infty[$, et sa valeur minimale est égale à zéro. Ainsi, la fonction $\mathsf{z} \in \mathbb{R} \mapsto \varphi(\mathsf{z})$ est strictement convexe et $(\forall \mathsf{z} \in \mathbb{R})$ $\varphi(\mathsf{z}) \geqslant 0 = \varphi(\mathsf{x})$.
- Étudions le cas où $\mathsf{x} \in]-\infty, 0]$. L'expression de φ est donnée par

$$(\forall \mathsf{z} \in \mathbb{R}) \quad \varphi(\mathsf{z}) = \begin{cases} +\infty & \text{si } \mathsf{z} \leqslant -1, \\ \mathsf{z} + \dfrac{1}{1+\mathsf{z}} - 1 + \dfrac{1}{2\gamma}(\mathsf{z}-\mathsf{x})^2 & \text{si } -1 < \mathsf{z} \leqslant 0, \\ \dfrac{1}{2\gamma}(\mathsf{z}-\mathsf{x})^2 & \text{si } 0 < \mathsf{z}. \end{cases} \tag{8.18}$$

1. $\operatorname{soft}_{\gamma}$ désigne la fonction de seuillage doux définie par (2.56).

Cette fonction est strictement convexe, et sa dérivée est donnée par

$$(\forall \mathsf{z} \in \mathbb{R}) \quad \dot{\varphi}(\mathsf{z}) = \begin{cases} +\infty & \text{si } \mathsf{z} \leqslant -1, \\ 1 - \dfrac{1}{(1+\mathsf{z})^2} + \dfrac{\mathsf{z}-\mathsf{x}}{\gamma} & \text{si } -1 < \mathsf{z} \leqslant 0, \\ \dfrac{\mathsf{z}-\mathsf{x}}{\gamma} & \text{si } 0 < \mathsf{z}. \end{cases} \tag{8.19}$$

Ainsi φ est strictement décroissante sur $]-1, \mathsf{z}^*[$, strictement croissante sur $]\mathsf{z}^*, 0]$, et strictement croissante sur $]0, +\infty[$, où $\mathsf{z}^* \in]-1, 0]$ est la solution de

$$0 = (1+\mathsf{z}^*)^3 - (\mathsf{x}+1-\gamma)(1+\mathsf{z}^*)^2 - \gamma. \tag{8.20}$$

On peut donc en déduire que

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \mathsf{z}^* = \mathrm{prox}_{\gamma\psi}(\mathsf{x}) = \operatorname*{argmin}_{\mathsf{z} \in]-1,0]} \ \varphi(\mathsf{z}) \tag{8.21}$$

est l'unique solution de (8.20) sur $]-1, 0]$.

On peut donc en conclure que

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \mathrm{prox}_{\gamma\psi}(\mathsf{x}) = \begin{cases} \mathsf{x} & \text{si } \mathsf{x} \geqslant 0, \\ \mathsf{z}^* & \text{sinon}, \end{cases} \tag{8.22}$$

où $\mathsf{z}^* \in]-1, 0]$ est l'unique solution de (8.20).

D'après la proposition 2.10, on peut alors en déduire que

$$(\forall \mathsf{x} \in \mathbb{R}) \quad \mathrm{prox}_{\gamma\phi}(\mathsf{x}) = \begin{cases} \mathsf{x} - \gamma & \text{si } \mathsf{x} > \gamma, \\ 0 & \text{si } \mathsf{x} \in [-\gamma, \gamma], \\ \mathsf{t}^* & \text{si } \mathsf{x} < -\gamma, \end{cases} \tag{8.23}$$

où t^* est solution de

$$(1+\mathsf{t}^*)^3 - (\mathsf{x}+1)(1+\mathsf{t}^*)^2 - \gamma = 0. \tag{8.24}$$

Par conséquent, en utilisant l'équation (8.16) on obtient

$$\begin{aligned} \mathrm{prox}_{\gamma\theta}(\mathsf{x}) &= \operatorname*{argmin}_{\mathsf{y} \in \mathbb{R}} \ 1 + \phi(\mathsf{y}-1) + \frac{1}{2\gamma}(\mathsf{y}-\mathsf{x})^2 \\ &= 1 + \operatorname*{argmin}_{\mathsf{y} \in \mathbb{R}} \ \phi(\mathsf{y}) + \frac{1}{2\gamma}(\mathsf{y}-(\mathsf{x}-1))^2 \\ &= 1 + \mathrm{prox}_{\gamma\phi}(\mathsf{x}-1) \\ &= \begin{cases} \mathsf{x} - \gamma & \text{si } \mathsf{x} > 1+\gamma, \\ 1 & \text{si } \mathsf{x} \in [1-\gamma, 1+\gamma], \\ 1+\mathsf{y}^* & \text{si } \mathsf{x} < 1-\gamma, \end{cases} \end{aligned} \tag{8.25}$$

où y^* est l'unique solution dans $]0,1]$ de

$$(1+y^*)^3 - (1+y^*)^2 x - \gamma = 0.$$

En posant $\zeta := 1+y$ on retrouve alors (8.14)-(8.15). □

Remarque 8.1.

(i) *La racine $\zeta^* \in]0,1[$ du polynôme de troisième degré r, donné dans (8.15), peut être calculée en utilisant la méthode de Newton [Hanke-Bourgeois, 2002], ou bien, de façon alternative, en utilisant la formule de Cardan [Tignol, 2001].*

(ii) *Une illustration graphique de la fonction $\mathrm{prox}_{\gamma\theta}$, pour plusieurs valeurs de γ, est donnée en figure 8.1.*

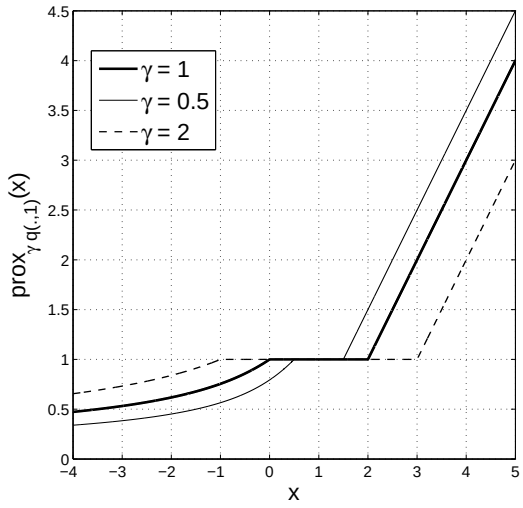


Figure 8.1 – *Fonction $\mathrm{prox}_{\gamma q(\cdot,1)}$ pour différents γ.*

Notons que, par un raisonnement similaire à celui utilisé pour obtenir (8.10), on a

$$(\forall \gamma \in]0,+\infty[)(\forall x \in \mathbb{R}) \quad \mathrm{prox}_{\gamma q(\cdot,b)}(x) = b\,\mathrm{prox}_{\gamma b^{-2}\theta}(x/b), \tag{8.26}$$

où $b \in]0,+\infty[$ est fixé. Ainsi, le corollaire suivant, qui donne l'opérateur proximal de $q(\cdot,b)$, pour $b \in]0,+\infty[$, est une conséquence directe de la proposition 8.1 :

Corollaire 8.1. *Soient $\gamma > 0$, $b \in]0,+\infty[$ et $x \in \mathbb{R}$. L'opérateur proximal de $q(\cdot,b)$ en x est donné par*

$$\mathrm{prox}_{\gamma q(\cdot,b)}(x) = \begin{cases} x - \frac{\gamma}{b} & \text{si } x > b + \frac{\gamma}{b}, \\ b & \text{si } x \in [b - \frac{\gamma}{b}, b + \frac{\gamma}{b}], \\ \zeta^* & \text{si } x < b - \frac{\gamma}{b}, \end{cases}$$

où ζ^* est l'unique solution sur $]0,b]$ du polynôme de troisième degré

$$\zeta^3 - x\zeta^2 - \gamma b = 0.$$

8.3.2 Opérateur proximal du maximum des fonctions quotient

Nous allons maintenant nous intéresser à

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \operatorname{prox}_{\mathbf{U},\boldsymbol{\theta}_\infty}(\mathbf{x}) = \operatorname*{argmin}_{\mathbf{y}\in\mathbb{R}^N} \ \boldsymbol{\theta}_\infty(\mathbf{y}) + \frac{1}{2}\|\mathbf{x}-\mathbf{y}\|_{\mathbf{U}}^2. \tag{8.27}$$

Nous proposons de réécrire ce problème de minimisation sous la forme d'un problème de minimisation sous contrainte épigraphique. Ce type d'approche a été introduite dans [Chierchia *et al.*, 2013a] et a été utilisée, par exemple, dans [Chierchia *et al.*, 2013b; Harizanov *et al.*, 2013].

En utilisant la définition 2.3(vi) de l'épigraphe, le problème de minimisation (8.27) peut se réécrire comme suit :

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \operatorname{prox}_{\mathbf{U},\boldsymbol{\theta}_\infty}(\mathbf{x}) = \operatorname*{argmin}_{\mathbf{y}\in\mathbb{R}^N,\, \xi\in\mathbb{R}} \ \xi + \frac{1}{2}\|\mathbf{x}-\mathbf{y}\|_{\mathbf{U}}^2$$
$$\text{tel que} \quad (\forall n \in \{1,\dots,N\}) \quad (\mathsf{y}^{(n)}, \xi) \in \operatorname{epi}\theta. \tag{8.28}$$

Le problème sous contraintes épigraphiques (8.28) peut être réécrit en faisant intervenir la fonction indicatrice donnée dans la définition 2.3(viii) :

$$\operatorname{prox}_{\mathbf{U},\boldsymbol{\theta}_\infty}(\mathbf{x}) = \operatorname*{argmin}_{(\mathbf{y},\xi)\in\mathbb{R}^{N+1},\, (\mathbf{s},\boldsymbol{\eta})\in\mathbb{R}^{2N}} \ \xi + \frac{1}{2}\|\mathbf{x}-\mathbf{y}\|_{\mathbf{U}}^2 + \sum_{n=1}^{N} \iota_{\operatorname{epi}\theta}(\mathsf{s}^{(n)}, \eta^{(n)})$$
$$\text{tel que} \quad \mathbf{s} = \mathbf{y} \ \text{ et } \ \xi\mathbf{1}_N = \boldsymbol{\eta}. \tag{8.29}$$

Le problème (8.29) peut être résolu, par exemple, en utilisant ADMM, décrit dans le chapitre 2 par l'algorithme 2.9. Les itérations de cette méthode appliquées au problème (8.29) sont données dans l'algorithme 8.1.

Algorithme 8.1 ADMM pour le calcul de $\operatorname{prox}_{\gamma\boldsymbol{\theta}_\infty}(\mathbf{x})$, où $\mathbf{x} \in \mathbb{R}^N$

Initialisation : Soient $\mu > 0$, $\begin{pmatrix} \mathbf{s}_0 \\ \boldsymbol{\eta}_0 \end{pmatrix} \in \mathbb{R}^{2N}$ et $\begin{pmatrix} \mathbf{p}_{\mathbf{y},0} \\ \mathbf{p}_{\xi,0} \end{pmatrix} \in \mathbb{R}^{2N}$.

Itérations :

Pour $k = 0, 1, \dots$

$$\begin{pmatrix} \mathbf{y}_{k+1} \\ \xi_{k+1} \end{pmatrix} = \operatorname*{argmin}_{(\mathbf{y},\xi)\in\mathbb{R}^{N+1}} \ \xi + \frac{1}{2}\|\mathbf{x}-\mathbf{y}\|_{\mathbf{U}}^2 + \frac{\mu}{2}\left(\|\mathbf{y} - \mathbf{s}_k + \mathbf{p}_{\mathbf{y},k}\|^2 + \|\xi\mathbf{1}_N - \boldsymbol{\eta}_k + \mathbf{p}_{\xi,k}\|^2\right),$$

pour $n = 1, \dots, N$

$$\begin{pmatrix} \mathsf{s}_{k+1}^{(n)} \\ \eta_{k+1}^{(n)} \end{pmatrix} = \Pi_{\operatorname{epi}\theta}\left(\begin{pmatrix} \mathsf{y}_{k+1}^{(n)} + \mathsf{p}_{\mathbf{y},k+1}^{(n)} \\ \xi_{k+1}^{(n)} + \mathsf{p}_{\xi,k+1}^{(n)} \end{pmatrix} \right),$$

$$\begin{pmatrix} \mathbf{p}_{\mathbf{y},k+1} \\ \mathbf{p}_{\xi,k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{p}_{\mathbf{y},k} \\ \mathbf{p}_{\xi,k} \end{pmatrix} + \begin{pmatrix} \mathbf{y}_{k+1} \\ \xi_{k+1}\mathbf{1}_N \end{pmatrix} - \begin{pmatrix} \mathbf{s}_{k+1} \\ \boldsymbol{\eta}^{k+1} \end{pmatrix}.$$

Notons que dans la première étape de l'algorithme 8.1, on peut effectuer la minimisation séparément par rapport à la variable $\mathbf{y}$ et par rapport à la variable ξ :

$$(\forall k \in \mathbb{N}) \quad \begin{cases} \mathbf{y}_{k+1} = \underset{\mathbf{y}\in\mathbb{R}^N}{\operatorname{argmin}} \ \frac{1}{2}\|\mathbf{x}-\mathbf{y}\|_{\mathbf{U}}^2 + \frac{\mu}{2}\|\mathbf{y}-\mathbf{s}_k+\mathbf{p}_{\mathbf{y},k}\|^2 \\ \xi_{k+1} = \underset{\xi\in\mathbb{R}}{\operatorname{argmin}} \ \xi + \frac{\mu}{2}\|\xi\mathbf{1}_N - \boldsymbol{\eta}_k + \mathbf{p}_{\xi,k}\|^2. \end{cases} \tag{8.30}$$

On veut donc trouver, pour tout $k \in \mathbb{N}$, $(\mathbf{y}_{k+1}, \xi_{k+1})$ tels que

$$\begin{cases} \mathbf{0}_N = \mathbf{U}(\mathbf{y}_{k+1}-\mathbf{x}) + \mu(\mathbf{y}_{k+1}-\mathbf{s}_k+\mathbf{p}_{\mathbf{y},k}) \\ 0 = 1 + N\mu\xi_{k+1} - \mu\sum_{n=1}^{N}(\eta_k^{(n)} - \mathsf{p}_{\xi,k}^{(n)}), \end{cases} \tag{8.31}$$

où $\mathbf{0}_N$ est le vecteur nul de $\mathbb{R}^N$. On obtient alors

$$\begin{cases} \mathbf{y}_{k+1} = \left(\mathbf{I}_N + \mu\mathbf{U}^{-1}\right)^{-1}\left(\mathbf{x} + \gamma\mu(\mathbf{s}_k - \mathbf{p}_{\mathbf{y},k})\right) \\ \xi_{k+1} = \dfrac{1}{N}\left(\displaystyle\sum_{n=1}^{N}(\eta_k^{(n)} - \mathsf{p}_{\xi,k}^{(n)}) - \frac{1}{\mu}\right). \end{cases} \tag{8.32}$$

Rappelons que $\mathbf{U} = \operatorname{Diag}(\mathbf{u})$ avec $\mathbf{u} = (\mathsf{u}^{(n)})_{1\leqslant n\leqslant N} \in]0,+\infty[^N$. Ainsi, pour tout $n \in \{1,\dots,N\}$, on a

$$\mathsf{y}_{k+1}^{(n)} = \frac{1}{1+\mu(\mathsf{u}^{(n)})^{-1}}\left(\mathsf{x}^{(n)} + \mu(\mathsf{u}^{(n)})^{-1}(\mathsf{s}_k^{(n)} - \mathsf{p}_{\mathbf{y},k}^{(n)})\right). \tag{8.33}$$

D'autre part, pour tout $k \in \mathbb{N}$, et pour chacune des composantes $n \in \{1,\dots,N\}$, nous devons projeter $\left(\mathsf{y}_{k+1}^{(n)} + \mathsf{p}_{\mathbf{y},k}^{(n)}, \xi_{k+1} + \mathsf{p}_{\xi,k}^{(n)}\right)$ dans l'ensemble $\operatorname{epi}\theta$. Cette projection est donnée par la proposition suivante :

Proposition 8.2. *Soit* $\mathsf{b} > 0$. *La projection* $\Pi_{\operatorname{epi}\mathsf{q}(\cdot,\mathsf{b})}(\mathsf{u},\zeta)$ *de* $(\mathsf{u},\zeta) \in \mathbb{R}^2$ *dans l'épigraphe de* $\mathsf{q}(\cdot,\mathsf{b})$ *est donné par*

$$\Pi_{\operatorname{epi}\mathsf{q}(\cdot,\mathsf{b})}(\mathsf{u},\zeta) := \begin{cases} (\mathsf{u},\zeta) & \text{si } \mathsf{u} > 0 \text{ et } \max\{\frac{\mathsf{u}}{\mathsf{b}}, \frac{\mathsf{b}}{\mathsf{u}}\} \leqslant \zeta, \\ \frac{\mathsf{bu}+\zeta}{1+\mathsf{b}^2}(\mathsf{b},1) & \text{si } 1+\mathsf{b}^2-\mathsf{bu} < \zeta < \frac{\mathsf{u}}{\mathsf{b}}, \\ (\mathsf{b},1) & \text{si } \zeta \leqslant \min\{1+\mathsf{b}^2-\mathsf{bu}, 1-\mathsf{b}^2+\mathsf{bu}\}, \\ (\mathsf{t}^*, \frac{\mathsf{b}}{\mathsf{t}^*}) & \text{si } 1-\mathsf{b}^2+\mathsf{bu} < \zeta \text{ et } \zeta < \frac{\mathsf{b}}{\mathsf{u}} \text{ pour } \mathsf{u} > 0, \end{cases} \tag{8.34}$$

où t^* *est l'unique racine dans* $]0,\mathsf{b}[$ *du polynôme du quatrième ordre*

$$\mathsf{r}(\mathsf{t}) = \mathsf{t}^4 - \mathsf{ut}^3 + \zeta\mathsf{bt} - \mathsf{b}^2. \tag{8.35}$$

Démonstration. Soit $\mathsf{b} > 0$. Afin de définir la projection sur l'épigraphe de $\mathsf{q}(\cdot,\mathsf{b})$, nous devons trouver les fonctions perpendiculaires [2] aux fonctions

$$\begin{cases} \mathsf{u} \in]0,\mathsf{b}] & \mapsto \quad \mathsf{q}_1(\mathsf{u},\mathsf{b}) := \mathsf{b}/\mathsf{u} \\ \mathsf{u} \in [\mathsf{b},+\infty[& \mapsto \quad \mathsf{q}_2(\mathsf{u},\mathsf{b}) := \mathsf{u}/\mathsf{b} \end{cases} \tag{8.36}$$

au point $(\mathsf{b},1)$.

2. Une fonction π est dite perpendiculaire à une fonction $\mathsf{q}\colon E \to \mathbb{R}$, pour $E \subset \mathbb{R}$, en un point $(\mathsf{b},\zeta) \in E \times \mathbb{R}$ si les vecteurs directeurs de leurs tangentes au point (b,ζ) sont orthogonaux.

(i) Soit $\mathsf{u} \in]0, \mathsf{b}]$. La fonction perpendiculaire à $\mathsf{q}_1(\mathsf{u}, \mathsf{b})$ en $(\mathsf{b}, 1)$, que nous noterons $\pi_1(\mathsf{u}) := \alpha_1 \mathsf{u} + \beta_1$, où $(\alpha_1, \beta_1) \in \mathbb{R}^2$, satisfait

$$\begin{cases} \langle (\mathsf{u}, \zeta) - (\mathsf{t}, \varrho), (1, -\mathsf{b}/\mathsf{t}^2) \rangle = 0, \\ (\mathsf{t}, \varrho) = (\mathsf{b}, 1), \\ \zeta = \pi_1(\mathsf{u}), \end{cases}$$

ce qui implique

$$\langle (\mathsf{u}, \zeta) - (\mathsf{b}, 1), (1, -1/\mathsf{b}) \rangle = 0,$$

soit, de façon équivalente,

$$\mathsf{b}\mathsf{u} - \mathsf{b}^2 + 1 = \zeta = \pi_1(\mathsf{u}).$$

Ainsi, la perpendiculaire π_1 est donnée par

$$\pi_1(\mathsf{u}) = \mathsf{b}\mathsf{u} + 1 - \mathsf{b}^2.$$

(ii) Soit $\mathsf{u} \in [\mathsf{b}, +\infty[$. La fonction perpendiculaire à $\mathsf{q}_2(\mathsf{u}, \mathsf{b})$ en $(\mathsf{b}, 1)$, notée $\pi_2(\mathsf{u})$ satisfait

$$\begin{cases} \pi_2(\mathsf{u}) = -\mathsf{b}\mathsf{u} + \beta_2, \\ \pi_2(\mathsf{b}) = 1, \end{cases}$$

où $\beta_2 \in \mathbb{R}$. Donc

$$\begin{cases} \pi_2(\mathsf{u}) = -\mathsf{b}\mathsf{u} + \beta_2, \\ -\mathsf{b}^2 + \beta_2 = 1, \end{cases}$$

et de façon équivalente

$$\pi_2(\mathsf{u}) = -\mathsf{b}\mathsf{u} + 1 + \mathsf{b}^2.$$

D'après (i) et (ii), les ensembles sont alors définis par

$$\begin{aligned} A_1 &= \{(\mathsf{u}, \zeta) \mid \mathsf{u} > 0 \text{ et } \zeta \leqslant \max\{\mathsf{u}/\mathsf{b}, \mathsf{b}/\mathsf{u}\}\}, \\ A_2 &= \{(\mathsf{u}, \zeta) \mid 1 + \mathsf{b}^2 - \mathsf{b}\mathsf{u} < \zeta < \mathsf{u}/\mathsf{b}\}, \\ A_3 &= \{(\mathsf{u}, \zeta) \mid \zeta \leqslant \min\{1 - \mathsf{b}^2 + \mathsf{b}\mathsf{u}, 1 + \mathsf{b}^2 - \mathsf{b}\mathsf{u}\}\}, \\ A_3 &= \{(\mathsf{u}, \zeta) \mid \zeta > 1 - \mathsf{b}^2 + \mathsf{b}\mathsf{u} \text{ et } (\zeta < \mathsf{b}/\mathsf{u} \text{ si } \mathsf{u} > 0)\}. \end{aligned}$$

Ces ensembles sont représentés sur la figure 8.2 dans le cas où $\mathsf{b} = 1$.

- **Cas où** $(\mathsf{u}, \zeta) \in A_1$. On remarque que $A_1 = \operatorname{epi} \mathsf{q}(\cdot, \mathsf{b})$. Ainsi,

$$(\forall (\mathsf{u}, \zeta) \in A_1) \quad \Pi_{\operatorname{epi} \mathsf{q}(\cdot, \mathsf{b})}(\mathsf{u}, \zeta) = (\mathsf{u}, \zeta). \tag{8.37}$$

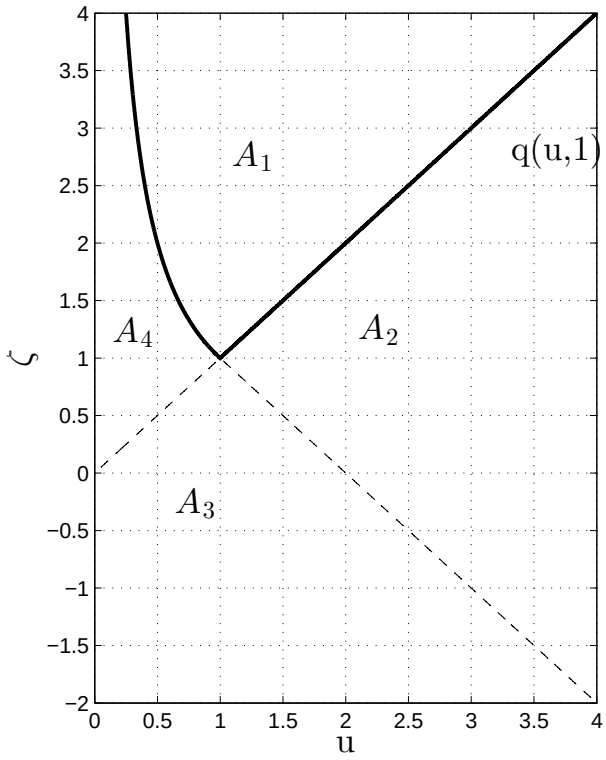


Figure 8.2 – *Division de* $\mathbb{R}^2$ *en quatre sous-espaces pour calculer la projection épigraphique dans* $\operatorname{epi} \mathsf{q}(\cdot, \mathsf{b})$ *pour* $\mathsf{b} = 1$.

- **Cas où** $(\mathsf{u}, \zeta) \in A_2$. La projection orthogonale $(\mathsf{t}, \varrho) = \Pi_{\operatorname{epi} \mathsf{q}(\cdot,\mathsf{b})}(\mathsf{u}, \zeta)$ satisfait

$$\begin{cases} \varrho = \mathsf{t}/\mathsf{b}, \\ \left\langle \begin{pmatrix} \mathsf{u} \\ \zeta \end{pmatrix} - \begin{pmatrix} \mathsf{t} \\ \varrho \end{pmatrix}, \begin{pmatrix} 1 \\ 1/\mathsf{b} \end{pmatrix} \right\rangle = 0. \end{cases}$$

  Donc, $\mathsf{t} = \frac{\mathsf{b}}{1+\mathsf{b}^2}(\mathsf{bu} + \zeta)$.
- **Cas où** $(\mathsf{u}, \zeta) \in A_3$. On remarque que A_3 correspond au cône normal de $\operatorname{epi} \mathsf{q}(\cdot, \mathsf{b})$ en $(\mathsf{b}, 1)$. Ainsi,

$$(\forall (u, \zeta) \in A_3) \quad \Pi_{\operatorname{epi} q(\cdot,\mathsf{b})}(u, \zeta) = (\mathsf{b}, 1). \tag{8.38}$$

- **Cas où** $(\mathsf{u}, \zeta) \in A_4$. Dans ce cas, la projection orthogonale se fait sur $\mathsf{t} \in]0, \mathsf{b}[\mapsto \tau(\mathsf{t}) := (\mathsf{t}, \mathsf{b}/\mathsf{t})$. La direction de la tangente de τ est $(1, -\mathsf{b}/\mathsf{t}^2)$, donc $(\mathsf{t}, \varrho) = \Pi_{\operatorname{epi} \mathsf{q}(\cdot,\mathsf{b})}(\mathsf{u}, \zeta)$ satisfait

$$\begin{cases} \varrho = \mathsf{t}/\mathsf{b}, \\ \left\langle \begin{pmatrix} \mathsf{u} \\ \zeta \end{pmatrix} - \begin{pmatrix} \mathsf{t} \\ \varrho \end{pmatrix}, \begin{pmatrix} 1 \\ 1/\mathsf{b} \end{pmatrix} \right\rangle = 0. \end{cases}$$

Par conséquent, $\Pi_{\mathrm{epi}\,\mathsf{q}(\cdot,\mathsf{b})}(\mathsf{u},\zeta) = (\mathsf{t},\mathsf{t}/\mathsf{b})$, où $\mathsf{t} \in]0,\mathsf{b}[$ est la solution unique de $\mathsf{r}(\mathsf{t}) = 0$, r étant le polynôme de quatrième ordre défini par (8.35). L'unicité est garantie puisque $\mathsf{q}(\cdot,\mathsf{b})$ étant convexe, son épigraphe est un ensemble convexe fermé non vide (proposition 2.1). Par conséquent, la projection sur cet ensemble est unique (proposition 2.9).

□

8.4 Application à l'estimation de la sélectivité

Le but de cette section est de résoudre le problème de minimisation suivant

$$\text{trouver } \widehat{\mathbf{x}} = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{C}} \ \mathbf{q}_\nu\left(\mathbf{A}\mathbf{x},\mathbf{b}\right), \tag{8.39}$$

où $\nu \in \{1,\infty\}$, $\mathbf{A} \in \mathbb{R}^{M\times N}$, $\mathbf{b} \in \mathbb{R}^M$, et

$$\mathcal{C} = \{\mathbf{x} = (\mathsf{x}^{(n)})_{1\leqslant n\leqslant N} \in [0,+\infty[^N \mid \sum_{n=1}^{N} \mathsf{x}^{(n)} \leqslant 1\}. \tag{8.40}$$

Ce type de problème apparaît, notamment, dans l'estimation des sélectivités pour l'optimisation des requêtes basées sur les coûts en SGBD (c.f. [Markl *et al.*, 2007; Moerkotte *et al.*, 2009]). Une brève description de cette application est donnée dans le paragraphe suivant.

8.4.1 Estimation de la sélectivité en SGBD

La sélectivité indique la proportion de n-uplets dans une base de donnée satisfaisant les prédicats dans une requête. La précision de l'estimation des sélectivités est cruciale pour la conception de plans d'exécution des requêtes optimales. Cependant, en pratique, cette estimation est faite de façon inexacte, introduisant des erreurs dans le modèle. Une question se pose alors : Comment ces erreurs influencent l'optimisation des plans de requêtes ? Dans [Moerkotte *et al.*, 2009], l'erreur de propagation de mauvaises estimations de sélectivités a été examinée au moyen d'une requête optimisée appropriée. Il est apparu que l'erreur de propagation est multiplicative (voir aussi [Ioannidis et Christodoulakis, 1991]). Il a de plus été démontré que les pires bornes d'erreurs pouvant être atteintes par la fonction de coût d'un plan optimal basé sur des sélectivités erronées, sont liées à la fois à la fonction de coût d'un plan optimal basé sur des sélectivités exactes, et au quotient des sélectivités erronées et exactes. Par conséquent, il semble que la fonction quotient d'erreur de sélectivités donnera de meilleures estimations que d'autres mesures d'erreurs (en particulier, on aura de meilleurs résultats que pour les erreurs additives).

8.4.2 Formulation du problème

Soit N le nombre de prédicats possibles d'une requête, $\mathcal{P}_N = 2^{\mathcal{I}_N}$ l'ensemble des parties de $\mathcal{I}_N = \{1,\ldots,N\}$. On suppose que l'on connaît une famille $\{\mathsf{p}_n \mid n \in \mathcal{I}_N\}$ de prédicats simples.

De la même manière que dans [Markl *et al.*, 2007], nous allons modéliser les sélectivités de prédicats par une distribution de probabilités. Nous allons donc les représenter sous *Forme Normale Disjonctive* (FND), dont les principaux symboles sont résumés dans la Table 8.1.

Symbole logique	définition
$\wedge$	et
$\vee$	ou
$\neg$	non

Table 8.1 – *Grammaire pour la FND*

Exemple 8.1. *Soit $N = 3$, les prédicats $\mathtt{p}_1$ et $\mathtt{p}_1 \wedge \mathtt{p}_2$ ont pour FND :*

$$\mathtt{p}_1 = (\mathtt{p}_1 \wedge \neg\mathtt{p}_2 \wedge \neg\mathtt{p}_3) \vee (\mathtt{p}_1 \wedge \mathtt{p}_2 \wedge \neg\mathtt{p}_3) \vee (\mathtt{p}_1 \wedge \neg\mathtt{p}_2 \wedge \mathtt{p}_3) \vee (\mathtt{p}_1 \wedge \mathtt{p}_2 \wedge \mathtt{p}_3),$$
$$\mathtt{p}_1 \wedge \mathtt{p}_2 = (\mathtt{p}_1 \wedge \mathtt{p}_2 \wedge \neg\mathtt{p}_3) \vee (\mathtt{p}_1 \wedge \mathtt{p}_2 \wedge \mathtt{p}_3).$$

Pour la FND du prédicat $\mathtt{p}_1 \wedge \mathtt{p}_2$, nous rappelons que :

- $\mathtt{p}_1 \wedge \mathtt{p}_2$ *est appelée* conjonction de prédicats *(ici des prédicats $\mathtt{p}_1$ et $\mathtt{p}_2$).*
- $\mathtt{v} = (\mathtt{p}_1 \wedge \mathtt{p}_2 \wedge \neg\mathtt{p}_3)$ *est une proposition de la FND du prédicat $\mathtt{p}_1 \wedge \mathtt{p}_2$.*

Nous utilisons les notations suivantes :

(i) Nous représentons chaque ensemble de $\mathcal{P}_N$ par une chaîne binaire $\boldsymbol{\beta} \in \{0,1\}^N$, où, pour tout $n \in \mathcal{I}_N$, $\beta^{(n)} = 1$ si et seulement si n est dans l'ensemble considéré. Alors, $\boldsymbol{\beta} = \mathbf{0}_N$ correspond à l'ensemble vide, tandis que $\boldsymbol{\beta} = \mathbf{1}_N$ correspond à l'ensemble $\mathcal{I}_N$ complet. On définit de plus $\mathcal{P}_N^* = \{0,1\}^N \backslash \{\mathbf{0}_N\}$.

(ii) Nous indexons les sélectivités $\mathtt{s}_{\boldsymbol{\beta}}$ de prédicats par les étiquettes binaires $\boldsymbol{\beta} \in \{0,1\}^N$, où, pour tout $n \in \mathcal{I}_N$, $\beta^{(n)} = 1$ si et seulement si le prédicat $\mathtt{p}_n$ est dans la conjonction.

(iii) Les sélectivités $\mathtt{x}_{\boldsymbol{\beta}}$ des propositions $\mathtt{v}$ de la FND sont indexées par des chaînes binaires $\boldsymbol{\beta} \in \{0,1\}^N$, où, pour tout $n \in \mathcal{I}_N$, $\beta^{(n)} = 1$ si et seulement si $\mathtt{p}_n \in \mathtt{v}$ et $\beta^{(n)} = 0$ si $\neg\mathtt{p}_n \in \mathtt{v}$.

Exemple 8.2. *Les FND de l'exemple 8.1 sont représentées de la façon suivante :*

$$\mathtt{s}_{100} = \mathtt{x}_{100} + \mathtt{x}_{110} + \mathtt{x}_{101} + \mathtt{x}_{111},$$
$$\mathtt{s}_{110} = \mathtt{x}_{110} + \mathtt{x}_{111}.$$

Les prédicats étant des probabilités, nous avons alors

$$\sum_{\boldsymbol{\beta} \in \{0,1\}^N} \mathtt{x}_{\boldsymbol{\beta}} = 1.$$

Les données $\mathtt{x}_{\boldsymbol{\beta}}$ peuvent être interprétées comme les probabilités d'apparition des propositions correspondantes. Notons que $\mathtt{x}_{\mathbf{0}_N}$ n'est un terme de la somme de $\mathtt{s}_{\boldsymbol{\beta}}$ que lorsque $\boldsymbol{\beta} = \mathbf{0}_N$.

Remarquons que seule une petite partie des sélectivités $\mathsf{s}_{\boldsymbol{\beta}}$, où $\boldsymbol{\beta} \in \mathcal{J} \subset \mathcal{P}_N^*$, tel que $M = \sharp\,\mathcal{J} \ll 2^N$, des conjonctions de prédicats peut être mémorisée comme statistiques multivariées en SGBD.

Définissons alors $\mathbf{b} = (\mathsf{s}_{\boldsymbol{\beta}})_{\boldsymbol{\beta}\in\mathcal{J}}$, $\mathbf{x} = (\mathsf{x}_{\boldsymbol{\beta}})_{\boldsymbol{\beta}\in\mathcal{P}_N^*}$ et $\mathbf{A} = (\mathsf{a}_{\boldsymbol{\beta},\boldsymbol{\beta}'})_{\boldsymbol{\beta}\in\mathcal{J},\boldsymbol{\beta}'\in\mathcal{P}_N^*} \in \{0,1\}^{M\times N}$, où

$$(\forall(\boldsymbol{\beta}\in\mathcal{J},\boldsymbol{\beta}'\in\mathcal{P}_N^*)) \quad \mathsf{a}_{\boldsymbol{\beta},\boldsymbol{\beta}'} = \begin{cases} 1 & \text{si } (\forall\, i \in \mathcal{I}_n) \quad \beta_i = 1 \Rightarrow \beta'_i = 1, \\ 0 & \text{sinon.} \end{cases} \tag{8.41}$$

Si, pour tout $\boldsymbol{\beta} \in \mathcal{J}$, $\mathsf{s}_{\boldsymbol{\beta}}$ était connu de façon précise, les $\mathsf{x}_{\boldsymbol{\beta}}$ satisferaient alors $\mathbf{Ax} = \mathbf{b}$. Dans [Markl *et al.*, 2007], les auteurs proposent d'estimer $(\mathsf{x}_{\boldsymbol{\beta}})_{\boldsymbol{\beta}\in\mathcal{P}_N^*}$ (et par conséquent, toutes les sélectivités) en maximisant l'entropie

$$\max_{\mathbf{x}\in\mathbb{R}^N} \sum_{\boldsymbol{\beta}\in\{0,1\}^N} -\mathsf{x}_{\boldsymbol{\beta}} \log \mathsf{x}_{\boldsymbol{\beta}} \text{ tel que } \begin{cases} \mathbf{Ax} = \mathbf{b}, \\ \mathbf{x} \geqslant \mathbf{0}, \\ \displaystyle\sum_{\boldsymbol{\beta}\in\{0,1\}^N} \mathsf{x}_{\boldsymbol{\beta}} = 1. \end{cases} \tag{8.42}$$

Si ce problème d'optimisation convexe est réalisable, i.e. admet une solution, il est alors possible de le résoudre en utilisant diverses méthodes. Par exemple, on peut utiliser la méthode de Newton en l'appliquant au problème dual (c.f. [Fletcher, 2000, p. 222-223]), ou bien une méthode itérative avec changement d'échelle comme il l'a été proposé dans [Markl *et al.*, 2007].

Cependant, en pratique, les inexactitudes présentes dans les sélectivités $\mathsf{s}_{\boldsymbol{\beta}}$ mémorisées rendent le problème (8.42) non réalisable, au sens où $\mathbf{Ax} = \mathbf{b}$ n'admet pas de solution $\mathbf{x} \in \mathbb{R}^N$ vérifiant $\mathbf{x} \geqslant \mathbf{0}$. Une solution pour pallier ce problème pourrait être d'ajouter un nouveau terme de régularisation au terme d'entropie à maximiser, pénalisant l'erreur entre $\mathbf{Ax}$ et $\mathbf{b}$. Néanmoins, cela introduirai un paramètre de régularisation qu'il faudrait optimiser par la suite. Par conséquent, nous proposons de traiter ce problème en deux étapes. Dans un premier temps, pour les raisons que nous avons données dans la section 8.4.1, nous allons minimiser le quotient de $\mathbf{Ax}$ par $\mathbf{b}$, i.e., nous allons résoudre le problème (8.39) de façon à rendre le problème réalisable. Nous pourrons alors résoudre le problème (8.42) dans un deuxième temps en remplaçant la variable $\mathbf{b}$ apparaissant dans ce problème par $\mathbf{A}\widehat{\mathbf{x}}$, où $\widehat{\mathbf{x}}$ est est la solution du problème (8.39), et en utilisant les méthodes précédemment citées. Dans ce chapitre, notre étude se focalisera sur la première étape.

8.4.3 Solution du problème en SGBD

Nous allons maintenant nous intéresser à la résolution du problème (8.39). Une approche, déjà présentée dans [Setzer *et al.*, 2010], est d'utiliser une méthode SOCP (*Second Order Cone Programming* [Lobo *et al.*, 1998]). Cependant, dans cette section nous proposons de résoudre ce problème à l'aide d'un algorithme primal-dual. Ce type d'algorithme présente l'avantage de pouvoir effectuer certaines étapes, telles que le calcul de l'opérateur de seuillage $\mathrm{prox}_{\mathsf{q}(\cdot,\mathbf{b})}$ ou des projections épigraphiques, en parallèle à chaque itération.

8.4.3.1 Algorithmes proposés

Pour $\nu \in \{1, +\infty\}$, nous reformulons le problème (8.39) comme suit :

$$\text{trouver } \widehat{\mathbf{x}} = \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N,\mathbf{y}\in\mathbb{R}^M} \mathbf{q}_\nu(\mathbf{y},\mathbf{b}) + \iota_{\mathcal{C}}(\mathbf{x}) \text{ tel que } \mathbf{Ax} = \mathbf{y}. \tag{8.43}$$

- Lorsque $\nu = 1$, le problème d'optimisation (8.43) peut être résolu par divers algorithmes primaux-duaux (voir section 2.4.4).
  Dans notre approche, nous proposons d'utiliser l'algorithme 2.14 donné dans la section 2.4.4, qui se réécrit sous la forme de l'algorithme 8.2 pour résoudre le problème

$$\text{trouver } \widehat{\mathbf{x}} = \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N, \mathbf{y}\in\mathbb{R}^M} \mathbf{q}_1(\mathbf{y}, \mathbf{b}) + \iota_{\mathcal{C}}(\mathbf{x}) \text{ tel que } \mathbf{A}\mathbf{x} = \mathbf{y}. \tag{8.44}$$

Algorithme 8.2 Algorithme primal-dual pour résoudre le problème (8.44)

Initialisation : Soient $\mathbf{x}_0 \in \mathbb{R}^N$, $\mathbf{p}_0 = \mathbf{z}_0 \in \mathbb{R}^M$, $(\mu, \sigma) \in]0, +\infty[^2$, et $\lambda \in]0, 1]$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{aligned}
\mathbf{x}_{k+1} &= \Pi_{\mathcal{C}}\Big(\mathbf{x}_k - \mu\sigma\mathbf{A}^\top \mathbf{z}_k\Big) \\
\mathbf{y}_{k+1} &= \operatorname{prox}_{\mathbf{q}_1(\cdot,\mathbf{b})/\sigma}(\mathbf{p}_k + \mathbf{A}\mathbf{x}_{k+1}) \\
\mathbf{p}_{k+1} &= \mathbf{p}_k + \mathbf{A}\mathbf{x}_{k+1} - \mathbf{y}_{k+1} \\
\mathbf{z}_{k+1} &= \mathbf{p}_{k+1} + \lambda\Big(\mathbf{p}_{k+1} - \mathbf{p}_k\Big)
\end{aligned}$$

Les propriétés de convergence de cet algorithme ont été énoncées dans le corollaire 2.1. L'étape de mise à jour des $(\mathbf{y}_k)_{k\in\mathbb{N}}$ nécessite le calcul de l'opérateur proximal de la fonction $\mathbf{q}_1(\cdot, \mathbf{b})$. Ce dernier peut-être effectué en calculant terme à terme, pour tout $m \in \{1, \ldots, M\}$, le seuillage de $\mathsf{q}(\cdot, \mathsf{b}^{(m)})$ décrit dans le corollaire 8.1.

- Lorsque $\nu = \infty$, pour $\mathbf{b} = (\mathsf{b}^{(m)})_{1\leqslant m\leqslant M}$, le problème (8.43) se réécrit comme suit

$$\text{trouver } (\widehat{\mathbf{x}}, \widehat{\xi}, \widehat{\mathbf{y}}, \widehat{\boldsymbol{\eta}}) \in \operatorname*{Argmin}_{(\mathbf{x},\xi)\in\mathbb{R}^{N+1}, (\mathbf{y},\boldsymbol{\eta})\in\mathbb{R}^{2M}} \xi + \sum_{m=1}^{M} \iota_{\operatorname{epi}\mathsf{q}(\cdot,\mathsf{b}^{(m)})}(\mathsf{y}^{(m)}, \eta^{(m)}) + \iota_{\mathcal{C}}(\mathbf{x}) \quad \text{tel que } \begin{cases} \mathbf{A}\mathbf{x} = \mathbf{y}, \\ \xi\mathbf{1}_M = \boldsymbol{\eta}. \end{cases} \tag{8.45}$$

Pour résoudre ce problème, on peut de nouveau utiliser l'algorithme 2.14. On obtient alors les itérations données dans l'algorithme 8.3, où

$$(\forall (\mathbf{x}, \xi) \in \mathbb{R}^{N+1}) \quad \mathsf{f}(\mathbf{x}, \xi) = \xi + \iota_{\mathcal{C}}(\mathbf{x}).$$

Algorithme 8.3 Algorithme primal-dual pour résoudre le problème (8.45)

Initialisation : Soient $\mathbf{x}_0 \in \mathbb{R}^N$, $\xi_0 \in \mathbb{R}$, $\mathbf{p}_{\mathbf{x},0} = \mathbf{z}_{\mathbf{x},0} \in \mathbb{R}^M$, $\mathbf{p}_{\xi,0} = \mathbf{z}_{\xi,0} \in \mathbb{R}^M$, $(\mu, \sigma) \in]0, +\infty[^2$, et $\lambda \in]0, 1]$.

Itérations :

Pour $k = 0, 1, \ldots$

$$\begin{pmatrix} \mathbf{x}_{k+1} \\ \xi_{k+1} \end{pmatrix} = \mathrm{prox}_{\mu \mathsf{f}} \left(\begin{pmatrix} \mathbf{x}_k \\ \xi_k \end{pmatrix} - \mu\sigma \begin{pmatrix} \mathbf{A}^\top \mathbf{z}_{\mathbf{x},k} \\ \mathbf{1}_M^\top \mathbf{z}_{\xi,k} \end{pmatrix} \right)$$

pour $m = 1, \ldots, M$

$$\begin{pmatrix} \mathsf{y}_{k+1}^{(m)} \\ \eta_{k+1}^{(m)} \end{pmatrix} = \Pi_{\mathrm{epi}\, \mathsf{q}(\cdot, \mathsf{b}^{(m)})/\sigma} \begin{pmatrix} \mathsf{p}_{\mathbf{x},k}^{(m)} + [\mathbf{A}\mathbf{x}_{k+1}]^{(m)} \\ \mathsf{p}_{\xi,k}^{(m)} + \xi_{k+1} \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{p}_{\mathbf{x},k+1} \\ \mathbf{p}_{\xi,k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{p}_{\mathbf{x},k} + \mathbf{A}\mathbf{x}_{k+1} - \mathbf{y}_{k+1} \\ \mathbf{p}_{\xi,k} + \mathbf{1}_M \xi_{k+1} - \boldsymbol{\eta}_{k+1} \end{pmatrix}$$

$$\begin{pmatrix} \mathbf{z}_{\mathbf{x},k+1} \\ \mathbf{z}_{\xi,k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{p}_{\mathbf{x},k} + \lambda \left(\mathbf{p}_{\mathbf{x},k} - \mathbf{p}_{\mathbf{x},k+1} \right) \\ \mathbf{p}_{\xi,k} + \lambda \left(\mathbf{p}_{\xi,k} - \mathbf{p}_{\xi,k+1} \right) \end{pmatrix}$$

Soit $k \in \mathbb{N}$ une itération de l'algorithme 8.3. Dans la première étape, $\mathbf{x}_{k+1}$ est donné par la projection de $\mathbf{x}_k - \mu\sigma \mathbf{A}^\top \mathbf{z}_{\mathbf{x},k}$ sur le simplexe $\mathcal{C}$, et ξ_{k+1} est donné par

$$\xi_{k+1} = \xi_k - \mu\sigma \mathbf{1}_M^\top \mathbf{z}_{\xi,k} - \mu. \tag{8.46}$$

La deuxième étape requiert la projection épigraphique calculée terme à terme du n-uplet $\left([\mathbf{p}_{\mathbf{x},k} + \mathbf{A}\mathbf{x}_{k+1}]^{(m)}, [\mathbf{p}_{\xi,k} + \mathbf{1}_M \xi_{k+1}]^{(m)} \right)$ sur l'épigraphe de $\mathsf{q}(\cdot, \mathsf{b}^{(m)})$, pour tout $m \in \{1, \ldots, M\}$. Cette dernière peut être calculée en exploitant le corollaire 8.2.

8.4.3.2 Exemple numérique

Nous allons maintenant présenter un exemple numérique afin d'illustrer notre méthode. Considérons le système d'équations suivant :

$$\begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} \mathsf{x}_{001} \\ \mathsf{x}_{010} \\ \mathsf{x}_{011} \\ \mathsf{x}_{100} \\ \mathsf{x}_{101} \\ \mathsf{x}_{110} \\ \mathsf{x}_{111} \end{pmatrix} = \begin{pmatrix} 0.2114 \\ 0.6331 \\ 0.6312 \\ 0.5182 \\ 0.9337 \\ 0.0035 \end{pmatrix} = \begin{pmatrix} \mathsf{s}_{001} \\ \mathsf{s}_{010} \\ \mathsf{s}_{100} \\ \mathsf{s}_{011} \\ \mathsf{s}_{101} \\ \mathsf{s}_{110} \end{pmatrix} \tag{8.47}$$

n'admettant pas de solution $\mathbf{x}$ positive. On résout le problème (8.39) pour $\nu = 1$ en utilisant l'algorithme 8.2, et pour $\nu = \infty$ avec l'algorithme 8.3. Nous utilisons comme critère d'arrêt $\|\mathbf{A}\mathbf{x}_k - \mathbf{y}_k\|_\infty < \epsilon$, où la matrice $\mathbf{A} \in \mathbb{R}^{M \times N}$ est donnée par (8.47) et $\epsilon = 10^{-5}$ (resp. $\epsilon = 10^{-4}$) pour l'algorithme 8.2 (resp. pour l'algorithme 8.3).

Remarquons que dans l'algorithme 8.2, le calcul de l'opérateur proximal de la fonction $\mathbf{q}_1(\cdot, \mathbf{b})$ (resp. dans l'algorithme 8.3, le calcul de la projection épigraphique) nécessite de trouver le zéro d'un polynôme du troisième degré (resp. du quatrième degré). Bien qu'il soit possible de trouver analytiquement ces zéros, dans nos simulations nous utilisons la méthode de Newton qui semble être plus rapide en pratique (comme nous l'avons vu dans la section 2.4.4, l'algorithme primal-dual utilisé est robuste aux erreurs numériques pouvant être introduites par cette méthode de calcul).

Nous comparerons nos solutions à celles obtenues en résolvant le problème suivant, faisant intervenir une norme d'une fonction additive :

$$\text{trouver } \widehat{\mathbf{x}} \in \operatorname*{Argmin}_{\mathbf{x} \in \mathcal{C}} \ \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_\alpha, \tag{8.48}$$

où $\alpha \in \{1, 2\}$ en utilisant, respectivement, des méthodes de programmation linéaire et de programmation quadratique de l'outil MOSEK [MOS].

Pour chaque solution $\widehat{\mathbf{x}}$ obtenue avec chacun des quatre algorithmes, nous calculons ensuite $\widehat{\mathbf{b}} = \mathbf{A}\widehat{\mathbf{x}}$. Notons que le vecteur $\widehat{\mathbf{x}}$ n'est ici d'aucun intérêt puisque, comme nous l'avons expliqué à la fin de la section 8.4.2, il sera estimé au cours d'une seconde étape (par exemple, dans une application en base de données, par minimisation entropique), non traitée dans ce chapitre. L'erreur $\mathbf{q}_\infty(\widehat{\mathbf{b}}, \mathbf{b})$ pour les quatre vecteurs résultants $\widehat{\mathbf{b}}$ est donnée dans la table 8.2.

méthode	(8.39) pour $\nu = \infty$	(8.39) pour $\nu = 1$	(8.48) pour $\alpha = 1$	(8.48) pour $\alpha = 2$
$\mathbf{q}_\infty(\widehat{\mathbf{b}}, \mathbf{b})$	2.61	3.65	75.65	74.85

Table 8.2 – *Erreur* $\mathbf{q}_\infty$ *obtenue pour l'exemple* (8.47).

Ces résultats reflètent de manière qualitative ce qui a pu être observé sur divers autres exemples. Pour des raisons évidentes, la méthode (8.39) pour $\nu = \infty$ donne la valeur minimale de l'erreur $\mathbf{q}_\infty(\widehat{\mathbf{b}}, \mathbf{b})$, puisque la méthode a été conçue dans le but de minimiser cette erreur. Le modèle (8.39) avec $\nu = 1$ donne une erreur $\mathbf{q}_\infty(\widehat{\mathbf{b}}, \mathbf{b})$ plus élevée mais cependant très proche de l'erreur minimale. Puisque cette méthode, n'ayant pas recourt aux projections épigraphiques, est plus rapide que la première, elle peut apparaitre comme un bon compromis. Pour finir, les méthodes (8.48) pour $\alpha \in \{1, 2\}$ conduisent toutes les deux à des erreurs $\mathbf{q}_\infty(\widehat{\mathbf{b}}, \mathbf{b})$ beaucoup plus importantes.

Dans cet exemple, nous ne donnons que la valeur de l'erreur $\mathbf{q}_\infty$, puisque, comme nous l'avons expliqué précédemment, c'est celle qui nous intéresse dans cette application, de par son influence sur la conception des demandes de plans d'exécution.

8.5 Conclusion

Dans ce chapitre nous avons d'une part déterminé l'opérateur proximal de la somme de fonctions quotients de vecteurs positifs, et d'autre part donné une méthode pour calculer l'opérateur proximal du maximum de ces fonctions quotients. Ces opérateurs proximaux peuvent être utilisés pour trouver les solutions de problèmes divers. Ici, nous avons considéré un problème de faisabilité intervenant dans l'estimation des sélectivités pour l'optimisation de requêtes. Dans ce type

d'application, les fonctions quotient se révèlent d'un intérêt évident en tant que mesure d'erreur lorsqu'elles sont comparées avec des mesures d'erreur additives. Afin de résoudre le problème sous-jacent, nous avons proposé d'utiliser un algorithme primal-dual, permettant de traiter les opérateurs proximaux des fonctions quotients.

Dans le cadre de l'optimisation des requêtes dans les SGBD, il serait intéressant de mettre en œuvre une implémentation parallèle de certaines étapes de l'algorithme primal-dual ainsi que de l'algorithme permettant de résoudre le problème (8.42). De plus, une comparaison détaillée de plusieurs méthodes, en particulier en ce qui concerne les temps d'exécution, doit être faite. Ces deux points pourront être traités dans de futurs travaux. Nous souhaiterions par ailleurs appliquer les fonctions quotients dans des problèmes pouvant apparaître en traitement d'images, comme par exemple pour la correction d'illumination dans le modèle *Retinex*, dans lesquels une image est supposée être le résultat d'un produit terme à terme de l'illumination et de la réflectivité d'une scène (c.f. [Gonzalez et Woods, 2002]).

Chapitre 9

Conclusions et perspectives

9.1 Bilan

Dans cette thèse, nous avons mis en œuvre des méthodes d'optimisation permettant de résoudre des problèmes faisant intervenir de grands volumes de données. En particulier, nous nous sommes intéressés à des problèmes inverses intervenant en traitement du signal et des images. Nous avons vu que ces problèmes peuvent être résolus en passant par la minimisation d'un critère s'écrivant comme la somme de fonctions non nécessairement convexes, non nécessairement différentiables, et pouvant être composées avec des opérateurs linéaires non nécessairement inversibles.

Dans un premier temps, nous nous sommes focalisés sur la minimisation d'une somme de deux fonctions, dont l'une était différentiable et l'autre était convexe.
Une méthode qui permet de résoudre ce problème, pour des données de grande taille, est l'algorithme explicite-implicite, qui alterne, à chaque itération une étape de gradient sur la fonction différentiable et une étape proximale sur la deuxième fonction. Cependant, en pratique, cet algorithme peut converger très lentement. Cela est dû, notamment, au pas apparaissant dans l'algorithme qui dépend de l'inverse de la constante de Lipschitz du gradient du terme différentiable. Ainsi, lorsque cette constante est très grande, le pas devient très petit et freine la convergence.
Une méthode permettant d'accélérer la convergence de l'algorithme explicite-implicite consiste à remplacer, à chaque itération, le pas par une matrice de préconditionnement. L'algorithme résultant correspond alors à l'algorithme explicite-implicite à métrique variable, ainsi nommé car à chaque itération la métrique sous-jacente est adaptée à l'itérée en cours. Ce type d'algorithme a déjà été proposé dans la littérature pour minimiser une somme de fonctions convexes. Cependant nous avons généralisé cette approche au cadre non convexe, et nous avons, de plus, proposé une méthode, basée sur le principe de majoration-minimisation, permettant de choisir efficacement la métrique à chaque itération.
Nous avons démontré la convergence des itérées de l'algorithme proposé vers un point critique du critère grâce à l'utilisation de l'inégalité de Kurdyka-Łojasievicz. De plus, nous avons montré que la convergence restait assurée lorsqu'une étape de relaxation convexe était incorporée dans la méthode.

Lorsque l'on est amenés à gérer un très grand nombre de données, il peut être intéressant, du point de vue de l'implémentation de l'algorithme, de ne traiter à chaque itération que des sous-parties de ces données, sous la forme de blocs. Les algorithmes adoptant une telle stratégie permettent de ne réaliser, à chaque itération, qu'un nombre réduit d'opérations correspondant à la dimension du bloc traité.
Dans une seconde partie de cette thèse, nous avons proposé de combiner cette stratégie alternée par bloc à l'algorithme explicite-implicite à métrique variable. Nous avons démontré la convergence des itérées de l'algorithme résultant, vers un point critique du critère à minimiser. Dans ce travail, nous nous sommes intéressés au cas où ce critère s'écrit comme la somme de deux fonctions dont l'une est différentiable non nécessairement convexes et l'autre n'est ni nécessairement différentiable, ni nécessairement convexe.
La stratégie de minimisation alternée proposée permet de mettre à jour les blocs selon une règle acyclique, autorisant de choisir les blocs suivant un ordre arbitraire et de mettre à jour certains d'entre eux plus souvent que d'autres. Notons que cette règle de mise à jour est beaucoup plus flexible que la règle cyclique qui est généralement adoptée lors de la mise en place de stratégies alternées par bloc.

Dans une troisième partie de cette thèse, nous nous sommes intéressés au cas où le critère à minimiser s'écrit comme la somme de plusieurs fonctions dont certaines ne sont pas nécessairement différentiables et peuvent être composées avec des opérateurs linéaires.
Ce type de critère ne peut pas être minimisé par un algorithme de type explicite-implicite sans faire appel à des sous-itérations. Cependant, lorsque toutes les fonctions intervenant dans le critère sont convexes, il existe des méthodes efficaces de type primal-dual permettant de résoudre ce problème.
Nous avons proposé de combiner ces algorithmes primaux-duaux à une stratégie alternée par bloc. Les méthodes que nous avons développées permettent de ne mettre à jour, à chaque itération, qu'un certain nombre de blocs, choisis de façon aléatoire suivant une loi arbitraire. L'un des avantages de cette règle de mise à jour aléatoire, par rapport à la règle acyclique, réside en la possibilité de mettre à jour plusieurs blocs en même temps en parallèle, à chaque itération.
Notons de plus que les algorithmes primaux-duaux alternés par bloc que nous avons proposés s'apparentent à des méthodes stochastiques permettant de prendre en compte des erreurs aléatoires.
Nous avons démontré la convergence presque sûre de ces algorithmes vers un couple de variables aléatoires solution du problème primal et du problème dual.
Enfin, les algorithmes primaux-duaux alternés aléatoirement par bloc nous ont permis de développer des algorithmes stochastiques distribués, dans lesquels les étapes de calculs sont réparties sur différents agents communiquant entre eux.

Les contributions méthodologiques de cette thèse ont été étayées par divers tests numériques sur des applications du traitement du signal et des images.

Dans un premier temps nous nous sommes intéressés à la restauration d'images dégradées par un bruit gaussien dépendant du signal, apparaissant dans certains systèmes d'imagerie. Nous avons étudié ce modèle de bruit à travers une interprétation MAP. Nous avons montré que le problème de minimisation qui en résulte n'est pas convexe, et nous avons appliqué l'algorithme explicite-implicite à métrique variable afin de le résoudre. Au travers de cet exemple nous avons pu montrer numériquement que l'introduction d'une métrique variable permet d'accélérer de manière significative la convergence de l'algorithme explicite-implicite.

Nous avons ensuite appliqué la version alternée par bloc de l'algorithme explicite-implicite pour traiter trois exemples différents en traitement du signal.
Dans le premier exemple, nous nous sommes intéressés à un problème de démélange spectral, s'apparentant à un problème de NMF (*nonnegative matrix factorization* ou factorisation de matrices positives), où le but était d'estimer simultanément deux ensembles de données : les spectres apparaissant dans l'image hyperspectrale d'origine ainsi que leur abondance. Nous avons écrit ce problème sous la forme d'une minimisation d'un critère non différentiable et non convexe. Nous avons montré au travers d'un exemple numérique que notre méthode permettait de résoudre efficacement ce problème, et que la métrique variable permettait d'accélérer considérablement la vitesse de convergence de l'algorithme.
Nous avons ensuite présenté une application en reconstruction de phase pour un problème de tomographie. Dans cet exemple, l'image d'origine considérée ainsi que la matrice d'observation étaient à valeurs complexes, cependant seul le module des observations était accessible. Ainsi, le problème associé au modèle considéré était non convexe. De plus, les données étant de grande dimension, nous les avons divisées en blocs de dimensions plus petites. Nous nous sommes intéressés à l'influence du choix de la taille de ces blocs sur la vitesse de convergence. Nous avons montré, grâce à un exemple numérique, que l'utilisation de blocs combinée avec celle d'une stratégie de préconditionnement permettait d'accélérer significativement la vitesse de convergence.
Finalement, nous avons proposé un exemple de déconvolution aveugle pour la restauration de signaux sismiques parcimonieux. Dans cette application nous avons reconstruit de façon simultanée le signal sismique et le noyau de convolution. Afin de modéliser la parcimonie du signal estimé, nous avons proposé d'utiliser une fonction de régularisation non convexe s'écrivant comme le logarithme du quotient d'une norme ℓ_1 et d'une norme ℓ_2. Dans cette application, nous avons montré, en particulier, l'intérêt pratique d'utiliser une règle acyclique de mise à jour des blocs. En effet, nous avons pu observer que la possibilité de mettre à jour plusieurs fois de suite le signal pour une seule mise à jour du noyau rendait la convergence de l'algorithme plus rapide. Ceci peut s'expliquer, d'une part, par le fait que la taille du signal à estimer était beaucoup plus grande que celle du noyau, et d'autre part par le fait que le signal était plus compliqué à estimer que le noyau du fait de sa parcimonie.

Les algorithmes primaux-duaux alternés par blocs nous ont permis de développer des méthodes algorithmiques permettant de reconstruire des maillages 3D lorsque la position des nœuds de ces derniers est connue de façon approximative. Ce type de problème peut s'écrire sous la forme d'un problème inverse où l'objectif est d'estimer la position des nœuds du maillage original à partir d'une version dégradée par un bruit additif. Dans cet exemple nous avons montré l'intérêt d'utiliser des méthodes par blocs en terme d'occupation de la mémoire. En effet, ce type de méthode permet de limiter le nombre de calculs qui est fait à chaque itération, en fonction de l'architecture de calcul dont on dispose.

Finalement, nous nous sommes intéressés au problème d'optimisation de requêtes dans les systèmes de gestion de bases de données. Dans ce type d'application, des norme ℓ_1 et ℓ_∞ de quotients de fonctions sont utilisées afin de définir des problèmes réalisables. Nous avons proposé des méthodes basées sur les algorithmes primaux-duaux pour résoudre ce type de problèmes. Pour cela, nous avons montré que l'opérateur proximal de la norme ℓ_1 d'un quotient de fonctions correspond à un opérateur de seuillage. Concernant la norme ℓ_∞, son opérateur proximal n'a pas de forme explicite. Nous avons donc reformulé le problème de façon à faire intervenir des projections épigraphiques, et nous avons proposé un algorithme permettant de le résoudre.

9.2 Perspectives

Certains travaux futurs, se plaçant dans la continuité de ce qui a été présenté dans cette thèse, peuvent être envisagés.

Choisir une métrique dans les algorithmes primaux-duaux

Dans les chapitres 3 et 5, nous avons proposé des algorithmes de type explicite-implicite accélérés grâce à l'introduction d'une métrique variant au cours des itérations. La méthode que nous avons donné pour choisir la métrique se base sur une interprétation MM de ce dernier de l'algorithme explicite-implicite. Bien que les algorithmes primaux-duaux proposés dans le chapitre 7 ne fassent intervenir qu'une métrique constante au cours des itérations, il serait possible d'y faire apparaitre une métrique variable. Cependant un problème majeur concerne le choix de cette métrique dans ce type d'algorithmes [Repetti *et al.*, 2012].
Une perspective intéressante est alors de proposer une méthode permettant de choisir une métrique (constante ou variant au cours des itérations) appropriée pour le préconditionnement des algorithmes primaux-duaux. Nous pouvons supposer que trouver un lien entre les algorithmes primaux-duaux et les stratégies MM permettrait de développer une méthode efficace pour le choix des préconditionneurs dans les méthodes primales-duales.

Proposer une version stochastique de l'algorithme explicite-implicite pour l'optimisation non convexe

Dans cette thèse, nous avons proposé des algorithmes alternés par bloc où les blocs mis à jours à chaque itération sont choisis selon deux règles différentes. Dans le cas de l'algorithme explicite-implicite alterné par bloc, les blocs sont prédéfinis de façon à former une partition des données globales. Au cours des itérations, les blocs sont ensuite choisis un à un pour être mis à jour selon une règle acyclique. Dans le cas des algorithmes stochastiques primaux-duaux, les blocs sont aussi prédéfinis, mais plusieurs d'entre eux peuvent être choisis aléatoirement à chaque itération. Ainsi, la mise à jour peut se faire de façon plus générale qu'avec la règle acyclique puisque les blocs peuvent se superposer et être traités en parallèles pendant une même itération. Il serait donc intéressant d'unifier ces résultats et d'étudier la convergence presque sûre d'algorithmes explicite-implicite stochastiques à métrique variable dans le cadre non convexe, ce qui permettrait de mettre à jour en parallèle les blocs sélectionnés à chaque itération.

Étendre les travaux réalisés sur les algorithmes primaux-duaux

Dans le chapitre 7, deux types d'algorithmes ont été proposés : des algorithmes alternés par blocs, et des algorithmes distribués. Les deux permettent de traiter des problèmes de minimisation convexe faisant intervenir un opérateur linéaire grâce à une formulation primale-duale du problème. Certaines perspectives concernant ces algorithmes peuvent être dressées :

- Dans le chapitre 8, nous avons traité un problème d'optimisation de requêtes dans les systèmes de gestion de bases de données en utilisant un algorithme primal-dual usuel [Chambolle et Pock, 2010]. L'exemple qui a été traité est de très petite dimension, et une extension naturelle serait de considérer un plus grand nombre de requêtes, pour des bases de données plus conséquentes, et d'appliquer un algorithme primal-dual alterné par bloc à ce problème.

- Nous avons montré dans la remarque 7.11 que les algorithmes distribués que nous avons proposés sont plus généraux que les méthodes récemment proposées dans la littérature dans le domaine de l'apprentissage. Ainsi, une seconde perspective concernant ce travail serait d'appliquer nos algorithmes distribués à des problèmes d'apprentissage régularisés tels que ceux abordés dans des cas particuliers dans [Bianchi *et al.*, 2014; Jaggi *et al.*, 2014].

- Les problèmes de minimisation pouvant être résolus par les algorithmes distribués que nous avons proposés sont de la forme

$$\text{trouver } \widehat{\mathsf{x}} \in \operatorname*{Argmin}_{\mathsf{x}\in\mathcal{H}} \sum_{i=1}^{m} \phi_i(\mathsf{x}),$$

  où $\mathcal{H}$ est un espace de Hilbert réel, et $(\forall i \in \{1,\dots,m\})$ les fonctions ϕ_i sont convexes, et peuvent s'écrire comme une somme de fonctions (non) différentiables qui peuvent être composées avec des opérateurs linaires. Ainsi, nous pouvons voir que lorsque les ϕ_i correspondent à des indicatrices d'ensembles convexes fermés non vides $C_i \subset \mathcal{H}$, le problème de

minimisation peut être reformulé de la façon suivante :

$$\text{trouver } \widehat{\mathsf{x}} \in \bigcap_{i=1}^{m} C_i,$$

ou, de façon plus générale,

$$\text{trouver } \widehat{\mathsf{x}} \text{ tel que } (\forall i \in \{1,\dots,m\}) \quad \mathsf{L}_i\widehat{\mathsf{x}} \in C_i,$$

où $(\forall i \in \{1,\dots,m\})$ L_i est un opérateur linéaire. Ce type de problème peut être résolu grâce à des algorithmes de projections alternées. Une perspective intéressante serait donc de faire le lien entre les algorithmes distribués proposés dans le chapitre 7 et les méthodes de projections alternées existantes (e.g. [Bauschke et Borwein, 1996; Combettes, 1994; Pesquet et Combettes, 1996]).

Généraliser la régularisation ℓ_1/ℓ_2

Plusieurs extensions peuvent être envisagées concernant la pénalisation de type ℓ_1/ℓ_2 proposée dans le chapitre 6.

- La première serait d'appliquer notre algorithme de minimisation alternée à métrique variable à des problèmes de déconvolution aveugle dans le cadre de la restauration d'images. La pénalisation ℓ_1/ℓ_2 permettant de promouvoir la parcimonie du signal auquel on l'applique, il suffirait dans ce cas de l'appliquer aux gradients verticaux et horizontaux de l'image à estimer (à l'instar de la variation totale). Cette méthode a été proposée dans [Krishnan *et al.*, 2011]. Les auteurs de cet article ont montré son efficacité sur plusieurs exemples de reconstruction d'image. Notre méthode offrant de solides résultats de convergence et se comparant favorablement à la méthode proposée dans [Krishnan *et al.*, 2011] pour des signaux 1D, il serait intéressant de comparer les deux méthodes pour des signaux 2D. On pourrait de plus envisager d'améliorer la méthode de [Krishnan *et al.*, 2011] en appliquant la pénalisation ℓ_1/ℓ_2, par exemple, à des gradients de l'image calculés sur un voisinage non local.

- Le problème de déconvolution aveugle est un problème non convexe, et introduire dans celui-ci une pénalisation non convexe rend le problème très difficile à résoudre. Nous avons pu remarquer, en particulier, que certains "pics" du signal estimé sont décalés, indifféremment vers la droite ou vers la gauche, par rapport aux "pics" du signal original. Les signaux reconstruits correspondent alors à des minimiseurs locaux du problème.
  Ainsi, un second point important qui pourrait être développé concerne l'initialisation de l'algorithme, qui permettrait d'augmenter les chances de trouver des minimiseurs globaux.

- Enfin, il pourrait être intéressant d'étudier une version plus générale de la régularisation ℓ_1/ℓ_2 afin de développer une classe de pénalisations qui pourraient s'adapter au mieux à d'autres applications. L'algorithme de minimisation alternée à métrique variable utilisé dans le chapitre 6 permet de traiter des fonctions diversifiées qui peuvent être ni convexes,

ni différentiables. Le but serait alors de développer une méthode efficace permettant d'utiliser un large spectre de pénalisations servant à promouvoir la parcimonie, en donnant des formules explicites afin de calculer facilement les métriques variables.

Traiter des problèmes de dominance stochastique

Nous avons montré que les algorithmes stochastiques présentés dans le chapitre 7 sont robustes aux erreurs pouvant apparaitre lors des calculs des gradients et des opérateurs proximaux. De plus, nous avons montré que ces termes d'erreurs sont des variables aléatoires. Dans les applications qui ont été présentées dans le chapitre 7, l'avantage de ces erreurs aléatoires n'a pas été exploité.
Une perspective intéressante serait d'étudier un problème pratique où ce type d'erreurs est utile. C'est en particulier le cas pour les problèmes de dominance stochastique [Dentcheva et Ruszczyǹski, 2014; Shapiro *et al.*, 2009] apparaissant, par exemple, en apprentissage ou pour la gestion de portefeuille, où le critère à minimiser fait intervenir des espérances probabilistes.

Bibliographie

The MOSEK Optimization Toolbox. http://www.mosek.com.

Absil, P.-A., Mahony, R. et Andrews, B. : Convergence of the iterates of descent methods for analytic cost functions. *SIAM J. Optim.*, 6:531–547, 2005.

Afonso, M., Bioucas-Dias, J. M. et Figueiredo, M. A. T. : An augmented lagrangian approach to the constrained optimization formulation of imaging inverse problems. *IEEE Trans. Image Process.*, 20(3):681–695, Mar. 2011.

Ahmed, A., Recht, B. et Romberg, J. : Blind deconvolution using convex programming. *IEEE Trans. Inform. Theory*, 60(3):1711–1732, Mar. 2014.

Allain, M., Idier, J. et Goussard, Y. : On global and local convergence of half-quadratic algorithms. *IEEE Trans. Image Process.*, 15(5):1130–1142, May 2006.

Alotaibi, A., Combettes, P. L. et Shahzad, N. : Solving coupled composite monotone inclusions by successive Fejér approximations of their Kuhn-Tucker set. *SIAM J. Optim.*, 24(4):2076–2095, Dec. 2014.

Antoniadis, A., Leporini, D. et Pesquet, J.-C. : Wavelet thresholding for some classes of non-gaussian noise. *Statist. Neerlandica*, 56(4):434–453, 2002.

Asner, G. P. et Heidebrecht, K. B. : Spectral unmixing of vegetation, soil and dry carbon cover in arid regions : comparing multispectral and hyperspectral observations. *Int. J. Remote Sens.*, 23(19):3939–3958, Oct. 2002.

Attouch, H. et Bolte, J. : On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Math. Program.*, 116:5–16, Jun. 2009.

Attouch, H., Bolte, J., Redont, P. et Soubeyran, A. : Proximal alternating minimization and projection methods for nonconvex problems. An approach based on the Kurdyka-Łojasiewicz inequality. *Math. Oper. Res.*, 35(2):438–457, 2010a.

Attouch, H., Bolte, J. et Svaiter, B. F. : Convergence of descent methods for semi-algebraic and tame problems : proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. *Math. Program.*, 137:91–129, Feb. 2011.

Attouch, H., Briceño Arias, L. M. et Combettes, P. L. : A parallel splitting method for coupled monotone inclusions. *SIAM J. Control Optim.*, 48:3246–3270, 2010b.

Auslender, A. : Asymptotic properties of the Fenchel dual functional and applications to decomposition problems. *J. Optim. Theory Appl.*, 73(3):427–449, Jun. 1992.

Barak, B., Kelner, J. et Steurer, D. : Rounding sum-of-squares relaxations. May 31-Jun. 3 2014.

Bauschke, H. H. et Borwein, J. : On projection algorithms for solving convex feasibility problems. *SIAM Review*, 38(3):367–426, 1996.

Bauschke, H. H. et Combettes, P. L. : *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, New York, 2011.

Bauschke, H. H., Combettes, P. L. et Luke, D. R. : Phase retrieval, error reduction algorithm, and Fienup variants : a view from convex optimization. *J. Opt. Soc. Amer. A*, 19(7):1334–1345, Jul. 2002.

Bauschke, H. H., Combettes, P. L. et Luke, D. R. : Hybrid projection-reflection method for phase retrieval. *J. Opt. Soc. Amer. A*, 20(6):1025–1034, June 2003.

Bauschke, H. H., Combettes, P. L. et Luke, D. R. : A new generation of iterative transform algorithms for phase contrast tomography. *In Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, vol. 4, p. 89–92, Philadelphia, PA, 19-23 Mar. 2005.

Bauschke, H. H., Combettes, P. L. et Noll, D. : Joint minimization with alternating Bregman proximity operators. *Pac. J. Optim.*, 2(3):401–424, Sep. 2006.

Beck, A. et Teboulle, M. : Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Trans. Image Process.*, 18(11):2419 –2434, Nov. 2009.

Becker, S. et Fadili, J. : A quasi-Newton proximal splitting method. Rap. tech., Jun. 2012. http://arxiv.org/abs/1206.1156.

Becker, S. R. et Combettes, P. L. : An algorithm for splitting parallel sums of linearly composed monotone operators, with applications to signal recovery. *J. Nonlinear Convex Anal.*, 15(1): 137–159, Jan. 2014.

Bect, J., Blanc-Féraud, L., Aubert, G. et Chambolle, A. : A l1-unified variational framework for image restoration. *In* T. Pajdla, J. M., éd. : *8th European Conference on Computer Vision (ECCV 2004)*, vol. 3024 de *Lecture Notes in Computer Science*, p. 1–13, Prague, Czech Republic, mai 2004. Springer.

Benichoux, A., Vincent, E. et Gribonval, R. : A fundamental pitfall in blind deconvolution with sparse and shift-invariant priors. *In Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Vancouver, BC, Canada, 26-31 May 2013.

Bertsekas, D. P. : Projected Newton methods for optimization problems with simple constraints. *SIAM J. Control Optim.*, 20:221–246, 1982.

Bertsekas, D. P. : *Nonlinear Programming*. Athena Scientific, Belmont, MA, 2nd édn, 1999.

Bianchi, P., Hachem, W. et Iutzeler, F. : A stochastic coordinate descent primal-dual algorithm and applications to large-scale composite optimization. Rap. tech., 2014. http://arxiv.org/abs/1407.0898.

Bioucas-Dias, J. M. et Figueiredo, M. A. : A new TwIST : Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Trans. Image Process.*, 16:2992–3004, Dec. 2007.

Bioucas-Dias, J. M., Plaza, A., Dobigeon, N., Parente, M., Qian, D., Gader, P. et Chanussot, J. : Hyperspectral unmixing overview : Geometrical, statistical, and sparse regression-based approaches. *IEEE J. Sel. Topics Appl. Earth Observ.*, 5(2):354–379, Apr. 2012.

Birgin, E. G., Martínez, J. M. et Raydan, M. : Nonmonotone spectral projected gradient methods on convex sets. *SIAM J. Optim.*, 10(4):1196–1211, 2000.

Blumensath, T. et Davies, M. : Iterative thresholding for sparse approximations. *J. Fourier Anal. Appl.*, 14(5):629–654, 2008.

Bochnak, J., Coste, M. et Roy, M.-F. : *Real Algebraic Geometry*. Springer, 1998.

Boţ, R. I. et Hendrich, C. : A Douglas-Rachford type primal-dual method for solving inclusions with mixtures of composite and parallel-sum type monotone operators. *SIAM J. Optim.*, 23 (4):2541–2565, Dec. 2013.

Boţ, R. I. et Hendrich, C. : Convergence analysis for a primal-dual monotone + skew splitting algorithm with applications to total variation minimization. *J. Math. Imaging Vision*, 49 (3):551–568, Jul. 2014.

Bolte, J., Daniilidis, A. et Lewis, A. : The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM J. Optim.*, 17:1205–1223, 2006a.

Bolte, J., Daniilidis, A. et Lewis, A. : A nonsmooth morse-sard theorem for subanalytic functions. *J. Math. Anal. Appl.*, 321:729–740, 2006b.

Bolte, J., Daniilidis, A., Lewis, A. et Shiota, M. : Clarke subgradients of stratifiable functions. *SIAM J. Optim.*, 18(2):556–572, 2007.

Bolte, J., Daniilidis, A., Ley, O. et Mazet, L. : Characterizations of Łojasiewicz inequalities : subgradient flows, talweg, convexity. *Trans. Amer. Math. Soc.*, 362(6):3319–3363, 2010.

Bolte, J. et Pauwels, E. : Majorization-minimization procedures and convergence of sqp methods for semi-algebraic and tame programs. Rap. tech., 2014. http://arxiv.org/pdf/1409.8147.pdf.

Bolte, J., Sabach, S. et Teboulle, M. : Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *To appear in Math. Program.*, 2013.

Bonettini, S., Zanella, R. et Zanni, L. : A scaled gradient projection method for constrained image deblurring. *Inverse Probl.*, 25(1):015002+, 2009.

Bonnans, J. F., Gilbert, J. C., Lemaréchal, C. et Sagastizábal, C. A. : A family of variable metric proximal methods. *Math. Program.*, 68:15–47, 1995.

Bouman, C. et Sauer, K. : A unified approach to statistical tomography using coordinate descent optimization. *IEEE Trans. Image Process.*, 5(3):480–492, 1996.

Boyd, S., Parikh, N., Chu, E., Peleato, B. et Eckstein, J. : Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Machine Learn.*, 8 (1):1–122, 2011.

Boyd, S., Xiao, L. et Mutapcic : Subgradient methods. *Lecture notes of EE392o, Stanford University*, 2003.

Bregman, L. M. : The method of successive projection for finding a common point of convex sets. *Soviet Math. Dokl.*, 6:688–692, 1965.

Bresson, X. : A short note for nonlocal tv minimization. Rap. tech., 2009. http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=582C006CFE9B89E3DB4BD6FEEB588A5B?doi=10.1.1.210.471&rep=rep1&type=pdf.

Briceño-Arias, L. M. : A Douglas-Rachford splitting method for solving equilibrium problems. *Nonlinear Anal.*, 75(16):6053–6059, Nov. 2012.

Briceños Arias, L. M. et Combettes, P. L. : A monotone + skew splitting model for composite monotone inclusions in duality. *SIAM J. Optim.*, 21(4):1230–1250, Oct. 2011.

Burger, M., Sawatzky, A. et Steidl, G. : First order algorithms in variational image processing. Rap. tech., 2014. http://www.mathematik.uni-kl.de/fileadmin/image/steidl/publications/algs_book_revision_01.pdf.

Burke, J. V. et Qian, M. : A variable metric proximal point algorithm for monotone operators. *SIAM J. Control Optim.*, 37:353–375, 1999.

Campisi, P. et Egiazarian, K., éds. *Blind Image Deconvolution : Theory and Applications*. CRC Press, 2007.

Candès, E., Strohmer, T. et Voroninski, V. : Phaselift : Exact and stable signal recovery from magnitude measurements via convex programming. *Comm. Pure Appl. Math.*, 66(8):1241–1274, 2013.

Censor, Y. et Lent, A. : Optimization of $\log x$ entropy over linear equality constraints. *SIAM J. Control Optim.*, 25(4):921–933, juillet 1987.

Chaâri, L., Chouzenoux, E., Pustelnik, N., Chaux, C. et Moussaoui, S. : OPTIMED : optimisation itérative pour la résolution de problèmes inverses de grande taille. *Traitement du Signal*, 28 (3-4):329–374, 2011.

Chaâri, L., Pustelnik, N., Chaux, C. et Pesquet, J.-C. : Solving inverse problems with overcomplete transforms and convex optimization techniques. *In SPIE, Wavelets XIII*, vol. 7446, San Diego, CA, USA, 2-6 Aug. 2009.

Chambolle, A. : An algorithm for total variation minimization and applications. *J. Math. Imag. Vision*, 20(1-2):89–97, 2004.

Chambolle, A. et Pock, T. : A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imag. Vision*, 40(1):120–145, 2010.

Chan, T. F. et Mulet, P. : On the convergence of the lagged diffusivity fixed point method in total variation image restoration. *SIAM J. Num. Anal.*, 2(36):354–367, 1999.

Chang, T., Nedić, A. et Scaglione, A. : Distributed constrained optimization by consensus-based primal-dual perturbation method. *IEEE Trans. Automat. Control*, 59(6):1524–1538, Jun. 2014.

Chartrand, R. : Nonconvex splitting for regularized low-rank + sparse decomposition. *IEEE Trans. Signal Process.*, 60:5810–5819, 2012.

Chaux, C., Combettes, P. L., Pesquet, J.-C. et Wajs, V. R. : A variational formulation for frame-based inverse problems. *Inverse Probl.*, 23(4):1495–1518, 2007.

Chaux, C., Duval, L., Benazza-Benyahia, A. et Pesquet, J.-C. : A nonlinear stein based estimator for multichannel image denoising. *IEEE Trans. Image Process.*, 56(8):3855–3870, Aug. 2008.

Chen, G. H.-G. et Rockafellar, R. T. : Convergence rates in forward-backward splitting. *SIAM J. Optim.*, 7:421–444, 1997.

Chen, P., Huang, J. et Zhang, X. : A primal-dual fixed point algorithm for convex separable minimization with applications to image restoration. *Inverse Problems*, 29(2):025011, 2013.

Chierchia, G., Pustelnik, N., J.-C., P. et Pesquet-Popescu, B. : Epigraphical projection and proximal tools for solving constrained convex optimization problems : Part i. Rap. tech., 2013a. http://arxiv.org/abs/1210.5844.

Chierchia, G., Pustelnik, N., Pesquet, J.-C. et Pesquet-Popescu, B. : An epigraphical convex optimization approach for multicomponent image restoration using non-local structure tensor. *In Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, p. 1359–1363, 26–31 May 2013b.

Chinneck, J. W. : *Feasibility and Infeasibility in Optimization*. Springer, New York, 2008.

Chouzenoux, E., Idier, J. et Moussaoui, S. : A Majorize-Minimize strategy for subspace optimization applied to image restoration. *IEEE Trans. Image Process.*, 20(18):1517–1528, Jun. 2011.

Chouzenoux, E., Jezierska, A., Pesquet, J.-C. et Talbot, H. : A Majorize-Minimize subspace approach for $\ell_2 - \ell_0$ image regularization. *SIAM J. Imaging Sci.*, 28(1):563–591, 2013.

Chouzenoux, E., Legendre, M., Moussaoui, S. et Idier, J. : Fast constrained least squares spectral unmixing using primal-dual interior-point optimization. *IEEE J. Sel. Topics Appl. Remote Sens.*, 7(1):59–69, 2014a.

Chouzenoux, E., Pesquet, J.-C. et Repetti, A. : Variable metric forward-backward algorithm for minimizing the sum of a differentiable function and a convex function. *J. Optim. Theory Appl.*, 162(1), Jul. 2014b.

Chu, M., Diele, F., Plemmons, R. et Ragni, S. : Optimality, computation, and interpretation of nonnegative matrix factorizations. *SIAM J. Matrix Anal. Appl.*, p. 4–8030, 2004.

Clark, R. N., Swayze, G. A., Gallagher, A., King, T. V. et Calvin, W. M. : The U.S. geological survey digital spectral library : version 1 : 0.2 to 3.0 μm. *U.S. Geological Survey, Denver, CO, Open File Rep. 93-592*, 1993.

Combettes, P. L. : Inconsistent signal feasibility problems : least-squares solutions in a product space. *IEEE Trans. Signal Process.*, 42(11):2955–2966, Nov. 1994.

Combettes, P. L. : The convex feasibility problem in image recovery. *In* Hawkes, P., éd. : *Advances in Imaging and Electron Physics, vol 96*, p. 155–270. Academic Press, New York, 1996.

Combettes, P. L. : Systems of structured monotone inclusions : duality, algorithms, and applications. *SIAM J. Optim.*, 23(4):2420–2447, Dec. 2013.

Combettes, P. L., Condat, L., Pesquet, J.-C. et Vũ, B. C. : A forward-backward view of some primal-dual optimization methods in image recovery. *In Proc. IEEE Int. Conf. Image Process.*, p. 4141–4145, Paris, France, 27-30 Oct. 2014.

Combettes, P. L., Dũng, D. et Vũ, B. C. : Dualization of signal recovery problems. *Set-Valued Var. Anal.*, 18:373–404, Dec. 2010.

Combettes, P. L., Dũng, D. et Vũ, B. C. : Proximity for sums of composite functions. *J. Math. Anal. Appl.*, 380(2):680–688, Aug. 2011.

Combettes, P. L. et Pesquet, J.-C. : A Douglas-Rachford splitting approach to nonsmooth convex variational signal recovery. *IEEE J. Selected Topics Signal Process.*, 1(4):564–574, Dec. 2007a.

Combettes, P. L. et Pesquet, J.-C. : Proximal thresholding algorithm for minimization over orthonormal bases. *SIAM J. Optim.*, 18(4):1351–1376, Nov. 2007b.

Combettes, P. L. et Pesquet, J.-C. : A proximal decomposition method for solving convex variational inverse problems. *Inverse Probl.*, 24(6), Dec. 2008.

Combettes, P. L. et Pesquet, J.-C. : Proximal splitting methods in signal processing. *In* Bauschke, H. H., Burachik, R., Combettes, P. L., Elser, V., Luke, D. R. et Wolkowicz, H., éds : *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, p. 185–212. Springer-Verlag, New York, 2010.

Combettes, P. L. et Pesquet, J.-C. : Primal-dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators. *Set-Valued Var. Anal.*, p. 1–24, 2011.

Combettes, P. L. et Pesquet, J.-C. : Stochastic quasi-Fejér block-coordinate fixed point iterations with random sweeping. *To appear in SIAM J. Optim.*, 2015.

Combettes, P. L. et Vũ, B. C. : Variable metric quasi-Fejér monotonicity. *Nonlinear Anal.*, 78:17–31, 2013.

Combettes, P. L. et Vũ, B. C. : Variable metric forward-backward splitting with applications to monotone inclusions in duality. *Optimization*, 63(9):1289–1318, Sept. 2014.

Combettes, P. L. et Wajs, V. R. : Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4(4):1168–1200, Nov. 2005.

Comon, P. : Contrasts for multichannel blind deconvolution. *Signal Process. Lett.*, 3(7):209–211, Jul. 1996.

Condat, L. : A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *J. Optim. Theory Appl.*, 158(2):460–479, Aug. 2013.

Coste, M. : An introduction to o-minimal geometry. *RAAG Notes, 81 pages, Institut de Recherche Mathématiques de Rennes*, Nov. 1999.

Couprie, C., Grady, L., Najman, L., Pesquet, J.-C. et Talbot, H. : Dual constrained TV-based regularization on graphs. *SIAM J. Imaging Sci.*, 6:1246–1273, 2013.

Coutinho, D. et Figueiredo, M. : Information theoretic text classification using the Ziv-Merhav method. *In Proc. 2nd Iberian Conf. on Pattern Recognition and Image Analysis (IbPRIA)*, p. 355–362, Estoril, Portugal, 2005.

Dahlquist, G. et Bjorck, A. : *Numerical Methods*. Dover Publication, Mineola, N.Y., 2003.

Davidoiu, V., Sixou, B., Langer, M. et Peyrin, F. : Nonlinear phase retrieval using projection operator and iterative wavelet thresholding. *IEEE Signal Process. Lett.*, 19(9):579–582, Sept 2012.

De Pierro, A. R. : A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography. *IEEE Trans. Med. Imag.*, 14(1):132–137, Mar. 1995.

Deledalle, C.-A., Vaiter, S., Fadili, J. M. et Peyré, G. : Stein COnsistent Risk Estimator (SCORE) for hard thresholding. *In Proc. SPARS'13*, 8–11 Jul. 2013.

Deledalle, C.-A., Vaiter, S., Fadili, J. M. et Peyré, G. : Stein Unbiased GrAdient estimator of the Risk (SUGAR) for multiple parameter selection. Aug. 2014.

Demanet, L. et Hand, P. : Scaling law for recovering the sparsest element in a subspace. *Information and Inference*, 2014.

Demoment, G. : Image reconstruction and restoration : Overview of common estimation structure and problems. *IEEE Trans. Acoust., Speech, Signal Process.*, 37(12):2024–2036, Dec. 1989.

Dentcheva, D. et Ruszczyǹski, A. : Risk-averse portfolio optimization via stochastic dominance constraints. *In* Lee, C.-F. et Lee, J. C., éds : *Handbook of Financial Econometrics and Statistics*, p. 2281–2302. Springer, 2014.

Dobigeon, N. et Brun, N. : Spectral mixture analysis of eels spectrum-images. *Ultramicroscopy*, 120:25–34, Sept. 2012.

Donoho, D. L. : Compressed sensing. *IEEE Trans. Inf. Theory*, 52(4):1289–1306, 2006.

Donoho, D. L., Johnstone, I. M., Kerkyacharian, G. et Picard, D. : Wavelet shrinkage : Asymptopia ? *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 57(2):301–369, 1995.

Eckstein, J. et Bertsekas, D. P. : On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Math. Program.*, 55(1-3):293–318, 1992.

Elad, M., Milanfar, P. et Ron, R. : Analysis versus synthesis in signal priors. *Inverse Problems*, 23(3):947–968, Jun. 2007.

Elmoataz, A., Lezoray, O. et Bougleux, S. : Nonlocal discrete regularization on weighted graphs : A framework for image and manifold processing. *IEEE Trans. Image Process.*, 17:1047–1060, 2008.

Erdogan, H. et Fessler, J. A. : Monotonic algorithms for transmission tomography. *IEEE Trans. Med. Imag.*, 18(9):801–814, Sept. 1999.

Esser, E., Lou, Y. et Xin, J. : A method for finding structured sparse solutions to non-negative least squares problems with applications. *SIAM J. Imaging Sci.*, 6(4):2010–2046, 2013.

Esser, E., Zhang, X. et Chan, T. : A general framework for a class of fist order primal-dual algorithms for convex optimization in imaging science. *SIAM J. Imaging Sci.*, 3(4):1015–1046, 2010.

Fessler, J. A. : Grouped coordinate ascent algorithms for penalized-likelihood transmission image reconstruction. *IEEE Trans. Med. Imag.*, 16:166–175, 1997.

Fienup, J. R. : Phase retrieval algorithms : A comparison. *Appl. Opt.*, 21:2758–2769, Aug. 1982.

Figueiredo, M. A. T. et Bioucas-Dias, J. M. : Restoration of Poissonian images using alternating direction optimization. *IEEE Trans. Image Process.*, 19(12):3133–3145, Dec. 2010.

Figueiredo, M. A. T. et Nowak, R. D. : Deconvolution of Poissonian images using variable splitting and augmented Lagrangian optimization. *In Proc. IEEE Workshop Stat. Sign. Proc.*, p. x+4, Cardiff, United Kingdom, 31 Aug. - 3 Sept. 2009.

Figueiredo, M. A. T., Nowak, R. D. et Wright, S. : Gradient projection for sparse reconstruction : Application to compressed sensing and other inverse problems. *IEEE J. Sel. Topics Signal Process.*, 1(4):586–597, Dec. 2007.

Fletcher, R. : *Practical Methods of Optimization*. Wiley, 2nd édn, 2000.

Fogel, F., Waldspurger, I. et D'Aspremont, A. : Phase retrieval for imaging problems. Rap. tech., Apr. 2013. http://arxiv.org/abs/1304.7735/.

Foi, A., Trimeche, M., Katkovnik, V. et Egiazarian, K. : Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Trans. Image Process.*, 17(10):1737–1754, Oct. 2008.

Fortet, R. M. : *Fonctions et Distributions Aléatoires dans les Espaces de Hilbert*. Hermès, Paris, 1995.

Fortin, M. et Glowinski, R., éds. *Augmented Lagrangian Methods : Applications to the Numerical Solution of Boundary-Value Problems*. Elsevier Science Ltd, Amsterdam : North-Holland, 1983.

Frankel, P., Garrigos, G. et Peypouquet, J. : Splitting methods with variable metric for Kurdyka-Łojasiewicz functions and general convergence rates. *To appear in J. Optim. Theory Appl.*, 2014.

Fuchs, J.-J. : Convergence of a sparse representations algorithm applicable to real or complex data. *IEEE J. Sel. Topics Signal Process.*, 1(14):598–605, Dec. 2007.

Gabay, D. et Mercier, B. : A dual algorithm for the solution of nonlinear variational problems via finite elements approximations. *Comput. Math. Appl.*, 2:17–40, 1976.

Geman, D. et Reynold, G. : Constrained restoration and the recovery of discontinuities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(3):367–383, Mar. 1992.

Geman, D. et Yang, C. : Nonlinear image recovery with half-quadratic regularization. *IEEE Trans. Image Process.*, 4(7):932–946, July 1995.

Gerchberg, R. W. et Saxton, W. O. : A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35:237–246, 1972.

Gilboa, G. et Osher, S. : Nonlocal operators with applications to image processing. *Multiscale Model. Simul.*, 7(3):1005–1028, 2008.

Giovannelli, J.-F. et Coulais, A. : Positive deconvolution for superimposed extended source and point sources. *Astron. Astrophys.*, 439:401–412, 2005.

Goldstein, T., Esser, E. et Baraniuk, R. : Adaptive primal-dual hybrid gradient methods for saddle-point problems. 2013.

Goldstein, T. et Osher, S. : The split Bregman method for ℓ_1-regularized problems. *SIAM J. Imaging Sci.*, 2:323–343, 2009.

Golub, G. H. et Van Loan, C. F. : *Matrix Computations*. Johns Hopkins University Press, Baltimore, Maryland, 3rd édn, 1996.

Gonzalez, R. C. et Woods, R. E. : *Digital Image Processing*. Addison–Wesley, Reading, second édn, 2002.

Gowen, A., O ?Donnell, C., Cullen, P., Downey, G. et Frias, J. : Hyperspectral imaging : an emerging process analytical tool for food quality and safety control. *Trends in Food Science and Technology*, 18(12):590–598, 2007.

Grady, L. J. et Polimeni, J. R. : *Discrete Calculus : Applied Analysis on Graphs for Computational Science*. Springer-Verlag, New York, 2010.

Gray, W. C. : Variable norm deconvolution. Rap. tech., 1978. http://sepwww.stanford.edu/oldreports/sep14/14_19.pdf.

Griesel, M. A. : A linear Remes-type algorithm for relative error approximation. *SIAM Journal on Numerical Analysis*, 11(1):170–173, 1974.

Guigay, J. P., Langer, M., Boistel, R. et Cloetens, P. : Mixed transfer function and transport of intensity approach for phase retrieval in the Fresnel region. *Opt. Lett.*, 32(12):1617–1619, Jun 2007.

Hadjisavvas, N. et Schaible, S. : Generalized monotone multivalued maps. *In* Floudas, C. A. et Pardalos, P. M., éds : *Encyclopedia of Optimization*, p. 1193–1197. Springer, New York, 2nd édn, 2009.

Hanke-Bourgeois, M. : *Grundlagen der Numerischen Mathematik und des Wissenschaftlichen Rechnens*. Teubner, Stuttgart, 2002.

Harizanov, S., Pesquet, J.-C. et Steidl, G. : Epigraphical projection for solving least squares anscombe transformed constrained optimization problems. *In* et al., A. K., éd. : *Scale-Space and Variational Methods in Computer Vision. Lecture Notes in Computer Science, SSVM 2013, LNCS 7893*, p. 125–136, Berlin, 2013. Springer.

Harrison, R. : Phase problem in crystallography. *J. Opt. Soc. Amer. A*, 10(5):1046–1055, 1993.

Haykin, S., éd. *Blind Deconvolution*. Prentice Hall, 1994.

He, B. et Yuan, X. : Convergence analysis of primal-dual algorithms for a saddle-point problem : from contraction perspective. *SIAM J. Imaging Sci.*, 5(1):119–149, 2012.

Healey, G. E. et Kondepudy, R. : Radiometric CCD camera calibration and noise estimation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(3):267–276, Mar. 1994.

Hiriart-Urruty, J.-B. et Lemaréchal, C. : *Convex Analysis and Minimization Algorithms*. Springer-Verlag, New York, 1993.

Hoyer, P. : Non-negative matrix factorization with sparseness constraints. *J. Mach. Learn. Res.*, 5:1457–1469, 2004.

Huber, P. J. : *Robust Statistics*. John Wiley, New York, NY, USA, 1981.

Hunter, D. R. et Lange, K. : A tutorial on MM algorithms. *Amer. Stat.*, 58(1):30–37, Feb. 2004.

Hurley, N. et Rickard, S. : Comparing measures of sparsity. *IEEE Trans. Inf. Theory*, 55(10):4723–4741, Oct. 2004.

Idier, J. : Convex half-quadratic criteria and interacting auxiliary variables for image restoration. *IEEE Trans. Image Process.*, 10(7):1001–1009, July 2001.

Ioannidis, Y. E. et Christodoulakis, S. : On the propagation of errors in the size of join results. *SIGMOD*, 1991.

Iutzeler, F., Bianchi, P., Ciblat, P. et Hachem, W. : Asynchronous distributed optimization using a randomized alternating direction method of multipliers. *In Proc. 52nd Conf. Decision Control*, p. 3671–3676, Florence, Italy, 10-13 Dec. 2013.

Jacobson, M. W. et Fessler, J. A. : An expanded theoretical treatment of iteration-dependent majorize-minimize algorithms. *IEEE Trans. Image Process.*, 16(10):2411–2422, Oct. 2007.

Jaganathan, K., Oymak, S. et Hassibi, B. : Recovery of sparse 1-D signals from the magnitudes of their Fourier transform. *In Proc. IEEE Int. Symp. Inf. Theory (ISIT 2012)*, p. 1473–1477, Cambridge, MA, 1-6 Jul. 2012.

Jaggi, M., Smith, V., Takáč, M., Terhorst, J., Krishnan, S., Hofmann, T. et Jordan, M. I. : Communication-efficient distributed dual coordinate ascent. *In Proc. Ann. Conf. Neur. Inform. Proc. Syst.*, p. 3068–3076, Montreal, Canada, 8-11 Dec. 2014.

Janesick, J. R. : *Photon Transfer*, vol. PM170. SPIE Press Book, Bellingham, WA, 2007.

Jezierska, A., Chouzenoux, E., Pesquet, J.-C. et Talbot, H. : A primal-dual proximal splitting approach for restoring data corrupted with poisson-gaussian noise. *In Proc. Int. Conf. Acoust., Speech Signal Process.*, p. 1085–1088, Kyoto, Japan, 25–30 Mar. 2012.

Jezierska, A., Chouzenoux, E., Pesquet, J.-C. et Talbot, H. : A convex approach for image restoration with exact poisson-gaussian likelihood. Rap. tech., 2013. https://hal.archives-ouvertes.fr/hal-00922151.

Ji, H., Li, J., Shen, Z. et Wang, K. : Image deconvolution using a characterization of sharp images in wavelet domain. *Appl. Comp. Harm. Analysis*, 32(2):295–304, 2012.

Jia, S. et Qian, Y. : Constrained nonnegative matrix factorization for hyperspectral unmixing. *IEEE Trans. Geosci. Remote Sens.*, 47(1):161–173, 2009.

Kaaresen, K. F. et Taxt, T. : Multichannel blind deconvolution of seismic signals. *Geophysics*, 63(6):2093–2107, Nov. 1998.

Kato, M., Yamada, I. et Sakaniwa, K. : A set-theoretic blind image deconvolution based on hybrid steepest descent method. *IEICE Trans. Fund. Electron. Comm. Comput. Sci.*, E82-A (8):1443–1449, Aug. 1999.

Keshava, N. et Mustard, J. F. : Spectral unmixing. *IEEE Signal Process. Mag.*, 19(1):44–57, Jan. 2002.

Komodakis, N. et Pesquet, J.-C. : Playing with duality : An overview of recent primal-dual approaches for solving large-scale optimization problems. *To appear in IEEE Signal Process. Mag.*, 2014.

Kowalski, M. : Proximal algorithm meets a conjugate descent. Rap. tech., Jun. 2010. http://hal.archives-ouvertes.fr/docs/00/50/57/33/PDF/proxconj.pdf.

Krishnan, D., Tay, T. et Fergus, R. : Blind deconvolution using a normalized sparsity measure. *In Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, p. 233–240, Colorado Springs, CO, USA, Jun. 21-25, 2011.

Kundur, D. et Hatzinakos, D. : Blind image deconvolution. *IEEE Signal Process. Mag.*, 13 (3):43–64, May 1996a.

Kundur, D. et Hatzinakos, D. : Blind image deconvolution revisited. *IEEE Signal Process. Mag.*, 13(6):61–63, Nov. 1996b.

Kurdyka, K. : On gradients of functions definable in o-minimal structures. *Ann. Inst. Fourier*, 48(3):769–783, 1998.

Lange, K. et Fessler, J. A. : Globally convergent algorithms for maximum a posteriori transmission tomography. *IEEE Trans. Image Process.*, 4(10):1430–1438, Oct. 1995.

Lee, D. D. et Seung, H. S. : Algorithms for non-negative matrix factorization. *In Advances in Neural and Information Processing Systems*, vol. 13, p. 556–562, 2001.

Lee, J. D., Sun, Y. et Saunders, M. A. : Proximal Newton-type methods for convex optimization. *In* Bartlett, P., Pereira, F., Burges, C., Bottou, L. et Weinberger, K., éds : *Advances in Neural Information Processing Systems (NIPS)*, vol. 25, p. 827 – 835, 2012.

Li, J., Shen, Z., Jin, R. et Zhang, X. : A reweighted ℓ_2 method for image restoration with Poisson and mixed Poisson-Gaussian noise. Rap. tech., 2012. UCLA Preprint, ftp://ftp.math.ucla.edu/pub/camreport/cam12-84.pdf.

Lobo, M. S., Vandenberghe, L., Boyd, S. et Lebret, H. : Applications of second order cone programming. *Linear Algebra and its Applications*, 284:193–228, 1998.

Łojasiewicz, S. : *Une propriété topologique des sous-ensembles analytiques réels*, p. 87–89. Editions du centre National de la Recherche Scientifique, 1963.

Loris, I. et Verhoeven, C. : On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty. *Inverse Problems*, 27(12):125007, 2011.

Lotito, P. A., Parente, L. A. et Solodov, M. V. : A class of variable metric decomposition methods for monotone variational inclusions. *J. Convex Anal.*, 16:857–880, 2009.

Luenberger, D. G. : *Linear and Nonlinear Programming*. Addison-Wesley, Reading, Massachusetts, 1973.

Luo, Z. Q. et Tseng, P. : On the convergence of the coordinate descent method for convex differentiable minimization. *J. Optim. Theory Appl.*, 72(1):7–35, 1992a.

Luo, Z. Q. et Tseng, P. : On the linear convergence of descent methods for convex essentially smooth minimization. *SIAM J. Control Optim.*, 30(2):408–425, 1992b.

Ma, W.-K., Bioucas-Dias, J. M., Tsung-Han, C., Gillis, N., Gader, P., Plaza, A., Ambikapathi, A. et Chong-Yung, C. : A signal processing perspective on hyperspectral unmixing : Insights from remote sensing. *IEEE Signal Process. Mag.*, 31(1):67–81, Jan. 2014.

Mallat, S. : *A Wavelet Tour of Signal Processing*. Academic Press, Burlington, MA, 2rd édn, 2009.

Mangasarian, O. L. : *Nonlinear Programming*. McGraw-Hill, New York, NY, 1969.

Marjanovic, G. et Solo, V. : On l_q optimization and matrix completion. *IEEE Trans. Image Process.*, 60(11):5714–5724, Nov. 2012.

Markl, V., Haas, P., Kutsch, M., Megiddo, N., Srivastava, U. et Tran, T. : Consistent selectivity estimation via maximum entropy. *VLDB Journal*, 16(1):55–76, Jan. 2007.

Metcalf, F. T. : Error measures and their associated means. *Journal of Approximation Theory*, 17:57–65, 1976.

Michelot, C. : A finite algorithm for finding the projection of a point onto the canonical simplex of $\mathbb{R}^n$. *J. Optim. Theory Appl.*, 50(1):195–200, Jul. 1986.

Moerkotte, G., Neumann, T. et Steidl, G. : Preventing bad plans by bounding the impact of cardinality estimation errors. *Proc. of the VLDB*, 2(1):982–993, 2009.

Monteiro, R. D. C. et Svaiter, B. F. : Convergence rate of inexact proximal point methods with relative error criteria for convex optimization. Rap. tech., 2010. http://www.optimization-online.org/DB_HTML/2010/08/2714.html.

Mordukhovich, B. S. : *Variational Analysis and Generalized Differentiation. Vol. I : Basic theory*, vol. 330 de *Series of Comprehensive Studies in Mathematics*. Springer-Verlag, Berlin-Heidelberg, 2006.

Moreau, E. et Pesquet, J.-C. : Generalized contrasts for multichannel blind deconvolution of linear systems. *IEEE Signal Process. Lett.*, 4(6):182–183, Jun. 1997.

Moreau, J. J. : Proximité et dualité dans un espace hilbertien. *Bull. Soc. Math. France*, 93:273–299, 1965.

Mørup, M., Madsen, K. H. et Hansen, L. K. : Approximate L_0 constrained non-negative matrix and tensor factorization. *In Proc. Int. Symp. Circuits Syst.*, p. 1328–1331, May 2008.

Moussaoui, S., Chouzenoux, E. et Idier, J. : Primal-dual interior point optimization for penalized least squares estimation of abundance maps in hyperspectral imaging. *In Proc. IEEE Workshop on Hyperspectral Image and Signal Processing : Evolution in Remote Sensing (WHISPERS ?12)*, p. 1–4, June 2012.

Mukherjee, S. et Seelamantula, C. : An iterative algorithm for phase retrieval with sparsity constraints : application to frequency domain optical coherence tomography. *In Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP 2012)*, p. 553–556, Kyoto, Japan, 25-30 Mar. 2012.

Nandi, A. K., Mampel, D. et Roscher, B. : Blind deconvolution of ultrasonic signals in nondestructive testing applications. *IEEE Trans. Signal Process.*, 45(5):1382–1390, 1997.

Necoara, I. et Patrascu, A. : A random coordinate descent algorithm for optimization problems with composite objective function and linear coupled constraints. *Comput. Optim. Appl.*, 57 (2):307–337, Mar. 2014.

Nedić, A. et Ozdaglar, A. : Cooperative distributed multi-agent optimization. *In* Palomar, D. P. et Eldar, Y. C., éds : *Convex Optimization in Signal Processing and Communications*, chap. 10, p. 340–386. Cambridge University Press, Cambridge, UK, 2010.

Nesterov, Y. : Gradient methods for minimizing composite objective function. 2007.

Ochs, P., Chen, Y., Brox, T. et Pock, T. : iPiano : inertial proximal algorithm for non-convex optimization. *SIAM J. Imaging Sci.*, 7(2):1388–1419, 2014.

Ochs, P., Dosovitskiy, A., Brox, T. et Pock, T. : On iteratively reweighted algorithms for nonsmooth non-convex optimization in computer vision. *SIAM J. Imaging Sci.*, 8(1):331–372, 2015.

Ortega, J. M. et Rheinboldt, W. C. : *Iterative Solution of Nonlinear Equations in Several Variables.* Academic Press, New York, NY, 1970.

Paatero, P. et Tapper, U. : Positive matrix factorization : a nonnegative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5:111–126, 1994.

Palis, J. et De Melo, W. : *Geometric Theory of Dynamical Systems. An Introduction.* Springer-Verlag, NewYork–Berlin, 1982.

Palma-Amestoy, R., Provenzi, E., Bertalmio, M. et Caselles, V. : A perceptually inspired variational framework for color enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(3):458–474, 2009.

Parente, L. A., Lotito, P. A. et Solodov, M. V. : A class of inexact variable metric proximal point algorithms. *SIAM J. Optim.*, 19:240–260, 2008.

Parikh, N. et Boyd, S. : Proximity algorithms. *Foundations and Trends in Optimization*, 1 (3):123–231, 2013.

Pesquet, J.-C., Benazza-Benyahia, A. et Chaux, C. : A SURE approach for digital signal/image deconvolution problems. *IEEE Trans. Signal Process.*, 57(12):4616–4632, Dec 2009.

Pesquet, J.-C. et Combettes, P. : Wavelet synthesis by alternating projections. *IEEE Trans. Signal Process.*, 44(3):728–732, Mar. 1996.

Pesquet, J.-C. et Pustelnik, N. : A parallel inertial proximal optimization method. *Pac. J. Optim.*, 8(2):273–305, Apr. 2012.

Peyré, G. : A review of adaptive image representations. *IEEE J. Selected Topics Signal Process.*, 5(5):896–911, 2011.

Pham, M. Q., Duval, L., Chaux, C. et Pesquet, J.-C. : A primal-dual proximal algorithm for sparse template-based adaptive filtering : Application to seismic multiple removal. *IEEE Trans. Signal Process.*, 62(16):4256–4269, Aug. 2014.

Pock, T. et Chambolle, A. : Diagonal preconditioning for first order primal-dual algorithms in convex optimization. *In Proc. IEEE Int. Conf. Comput. Vis.*, p. 1762–1769, Barcelona, Spain, 6-13 Nov. 2011.

Pock, T., Chambolle, A., Cremers, D. et Bischof, H. : A convex relaxation approach for computing minimal partitions. *IEEE Conf. Computer Vision and Pattern Recognition*, p. 810–817, 2009.

Powell, M. J. D. : On search directions for minimization algorithms. *Math. Program.*, 4:193–201, 1973.

Pustelnik, N., Borgnat, P. et Flandrin, P. : Empirical Mode Decomposition revisited by multicomponent nonsmooth convex optimization. *Signal Process.*, 102:313–331, Sept. 2014.

Pustelnik, N., Pesquet, J.-C. et Chaux, C. : Relaxing tight frame condition in parallel proximal methods for signal restoration. *IEEE Trans. Signal Process.*, 60(2):968–973, Feb. 2012.

Raguet, H., Fadili, J., et Peyré, G. : A generalized forward-backward splitting. *SIAM J. Imaging Sci.*, 6(3):1199–1226, 2013.

Razaviyayn, M., Hong, M. et Luo, Z. : A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM J. Optim.*, 23(2):1126–1153, 2013.

Repetti, A., Chouzenoux, E. et Pesquet, J.-C. : A penalized weighted least squares approach for restoring data corrupted with signal-dependent noise. *In Proc. Eur. Sig. and Image Proc. Conference*, p. 1553–1557, Bucharest, Romania, 27-31 Aug. 2012.

Richtárik, P. et Talác, M. : Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Math. Program.*, 144(1-2):1–38, Apr. 2014.

Ricker, N. : The form and nature of seismic waves and the structure of seismograms. *Geophysics*, 5(4):348–366, 1940.

Rockafellar, R. T. : *Convex Analysis*. Princeton University Press, 1970.

Rockafellar, R. T. : *Conjugate Duality and optimization*. SIAM, 1974.

Rockafellar, R. T. : Monotone operators and the proximal point algorithm. *SIAM J. Control Optim.*, 14:877–898, 1976.

Rockafellar, R. T. et Wets, R. J.-B. : *Variational Analysis*. Springer-Verlag, 1st édn, 1997.

Rudin, L. I., Osher, S. et Fatemi, E. : Nonlinear total variation based noise removal algorithms. *Phys. D*, 60:259–268, 1992.

Saquib, S., Zheng, J., Bouman, C. A. et Sauer, K. D. : Parallel computation of sequential pixel updates in statistical tomographic reconstruction. *Proc. IEEE Int. Conf. Image Processing*, 2:93–96, 1995.

Setzer, S., Steidl, G., Teuber, T. et Moerkotte, G. : Approximation related to quotient functionals. *Journal of Approximation Theory*, 162(3):545–558, 2010.

Shapiro, A., Dentcheva, D. et Ruszczyǹski, A. : *Lectures on Stochastic Programming : Modeling and Theory*. Society for Industrial and Applied Mathematics and the Mathematical Programming Society, Philadelphia, USA, 2009.

Shechtman, Y., Beck, A. et Eldar, Y. : GESPAR : Efficient phase retrieval of sparse signals. *IEEE Trans. Signal Process.*, 62(4):928–938, Feb 2014.

Shechtman, Y., Eldar, Y., Szameit, A. et Segev, M. : Efficient coherent diffractive imaging for sparsely varying objects. *Optics Express*, 21(5):6327–6338, 2013.

Shiota, M. : *Geometry of Subanalytic and emialgebraic Sets*, vol. 140. Progress in Mathematics, 1997.

Shor, N. Z. : *Minimization Methods for Non-differentiable Functions*. Springer-Verlag, 1985.

Shuman, D. I., Narang, S. K., Forssard, P., Ortega, A. et Vandergheynst, P. : The emerging field of signal processing on graphs : Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, 2013.

Slaney, M. et Kak, A. : Principles of computerized tomographic imaging. *SIAM, Philadelphia*, 1988.

Slavakis, K., Kopsinis, Y., Theodoridis, S. et McLaughlin, S. : Generalized thresholding and online sparsity-aware learning in a union of subspaces. *IEEE Trans. Signal Process.*, 61(15):3760–3773, Aug. 2013.

Solodov, M. V. et Svaiter, B. F. : A unified framework for some inexact proximal point algorithms. *Numer. Funct. Anal. Optim.*, 22:1013–1035, 2001.

Sotthivirat, S. et Fessler, J. A. : Image recovery using partitioned-separable paraboloidal surrogate coordinate ascent algorithms. *IEEE Trans. Signal Process.*, 11(3):306–317, 2002.

Svaiter, B. F. : A class of Fejér convergent algorithms, approximate resolvents and the hybrid proximal-extragradient method. *J. Optim. Theory Appl.*, 162:133–153, July 2014.

Takahata, A. K., Nadalin, E. Z., Ferrari, R., Duarte, L. T., Suyama, R., Lopes, R. R., Romano, J. M. T. et Tygel, M. : Unsupervised processing of geophysical signals : A review of some key aspects of blind deconvolution and blind source separation. *IEEE Signal Process. Mag.*, 29 (4):27–35, Jul. 2012.

Tappenden, R., Richtárik, P. et Gondzio, J. : Inexact coordinate descent : Complexity and preconditioning. Rap. tech., 2013. http://arxiv.org/abs/1304.5530.

Taubin, G., Zhang, T. et Golub, G. : Optimal surface smoothing as filter design. *In* Buxton, B. et Cipolla, R., éds : *Computer Vision ECCV '96*, vol. 1064 de *Lecture Notes in Computer Science*, p. 283–292. Springer Berlin Heidelberg, 1996.

Teuber, T., Steidl, G. et Chan, R.-H. : Minimization and parameter estimation for seminorm regularization models with I-divergence constraints. *Inverse Problems*, 29:035007, Mar. 2013.

Themelis, K. E., Schmidt, F., Sykioti, O., Rontogiannis, A. A., Koutroumbas, K. D. et Daglis, I. A. : On the unmixing of MEx/OMEGA hyperspectral data. *Planetary and Space Science*, 68(1):34–41, 2012.

Tian, H., Fowler, B. et Gamal, A. E. : Analysis of temporal noise in CMOS photodiode active pixel sensor. *IEEE Solid State Circuits Mag.*, 36(1):92–101, Jan. 2001.

Tignol, J. P. : *Galois' Theory of Algebraic Equations*. World Scientific, Singapore, 2001.

Tomazeli Duarte, L., Moussaoui, S. et Jutten, C. : Source separation in chemical analysis : Recent achievements and perspectives. *IEEE Signal Processing Magazine*, 31(3):135–146, May 2014.

Towfic, Z. J. et Sayed, A. H. : Stability and performance limits of adaptive primal-dual networks. Rap. tech., 2014. http://arxiv.org/pdf/1408.3693.pdf.

Tran-Dinh, Q., Kyrillidis, A. et Cevher, V. : Composite self-concordant minimization. *To appear in J. Mach. Learn. Res.*, 2014.

Tseng, P. : A modified forward-backward splitting method for maximal monotone mappings. *SIAM J. Control Optim.*, 38(2):431–446, 1998.

Tseng, P. : Convergence of a block coordinate descent method for nondifferentiable minimization. *J. Optim. Theory Appl.*, 109(3):475–494, Jun. 2001.

Vũ, B. C. : A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Advances in Computational Mathematics*, 38(3):667–681, 2013.

Van den Dries, L. : Tame topology and o-minimal structures. vol. 248 de *London Mathematical Society Lecture Note Series*. Cambridge University Press, 1998.

Van den Dries, L. et Miller, C. : Geometries categories and o-minimal structures. *Duke Math. J.*, 84:497–540, 1996.

Walden, A. T. et Hosken, J. W. J. : The nature of the non-Gaussianity of primary reflection coefficients and its significance for deconvolution. *Geophys. Prospect.*, 34(7):1038–1066, 1986.

Waldspurger, I., d'Aspremont, A. et Mallat, S. : Phase recovery, maxcut and complex semidefinite programming. *To appear in Math. Program.*, 2013.

Walther, A. : The question of phase retrieval in optics. *J. Modern Opt.*, 10(1):41–49, 1963.

Werlberger, M., Pock, T. et Bischof, H. : Motion estimation with non-local total variation regularization. *In IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, p. 2464–2471, 2010.

Wiener, N. : *Extrapolation, interpolation, and smoothing of stationary time series*. MIT Press, 1949.

Xu, Y. et Yin, W. : A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM J. Imaging Sci.*, 6(3):1758–1789, 2013.

Yuan, D., Xu, S. et Zhao, H. : Distributed primal-dual subgradient method for multiagent optimization via consensus algorithms. *IEEE Trans. Systems, Man, and Cybernetics- Part B*, 41(6):1715 ?–1724, Dec. 2011.

Zangwill, W. I. : *Nonlinear Programming.* Prentice-Hall, Englewood Cliffs, New Jersey, 1969.

Zheng, J., Saquib, S. S., Sauer, K. et Bouman, C. A. : Parallelizable bayesian tomography algorithms with rapid, guarenteed convergence. *SIAM J. Optim.*, 14(4):1043–1056, 2004.

Zibulevsky, M. et Pearlmutter, B. A. : Blind source separation by sparse decomposition in a signal dictionary. *Neural Comput.*, 13(4):863–882, Apr. 2001.

Ziv, A. : Relative distance - an error measure in round-off error analysis. *Mathematics of Computation*, 39(160):563–569, 1982.

Zubal, I. G., Harrell, C. R., Smith, E. O., Rattner, Z., Gindi, G. et Hoffer, P. B. : Computerized three-dimensional segmented human anatomy. *Med. Phys.*, 21:299–302, 1994.